사용자 가이드

Amazon Inspector

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Inspector: 사용자 가이드

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon의 상표 및 브랜드 디자인은 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

Amazon Inspector란 무엇인가? 1
특징 1
Amazon Inspector 액세스 3
시작하기 5
Amazon Inspector를 활성화하기 전에 5
시작하기 자습서: Amazon Inspector 활성화 6
자동 스캔 8
Amazon Inspector 스캔 유형 개요 8
스캔 유형 활성화 9
스캔 활성화 10
Amazon EC2 인스턴스 스캔 11
에이전트 기반 스캔 12
에이전트 없는 스캔 16
스캔 모드 관리 17
Amazon Inspector 스캔에서 인스턴스 제외 18
지원되는 운영 체제 19
Linux 인스턴스에 대한 심층 검사 19
스캔 Windows EC2 인스턴스 24
Amazon ECR 컨테이너 이미지 스캔 27
Amazon 스캔을 위한 ECR 스캔 동작 28
지원되는 운영 체제 및 미디어 유형 29
Amazon ECR리포지토리에 대한 향상된 스캔 구성 29
ECR 재스캔 기간 30
Lambda 함수 스캔 32
Lambda 함수 스캔의 스캔 동작 32
지원되는 런타임 및 함수 33
Amazon Inspector Lambda 표준 스캔 34
Amazon Inspector Lambda 코드 스캔 35
스캔 유형 비활성화 37
스캔 비활성화 38
CIS 스캔 39
Amazon Inspector 스캔에 대한 Amazon EC2 인스턴스 요구 사항 CIS 40
프라이빗 Amazon 인스턴스에서 스캔을 실행하기 위한 Amazon Virtual Private Cloud 엔드포인트 요구 사항 CIS EC2 40

CIS 스캔 실행 ... 41
를 사용하여 Amazon Inspector CIS 스캔을 관리하기 위한 고려 사항 AWS Organizations ... 42
Amazon Inspector가 소유한 Amazon S3 버킷으로 Amazon Inspector CIS 스캔에 사용 ... 43
CIS 스캔 구성 생성 ... 45
CIS 스캔 결과 보기 ... 46
CIS 스캔 구성 편집 ... 47
CIS 스캔 결과 다운로드 ... 47
결과 이해 ... 49
결과 유형 ... 50
패키지 취약성 ... 50
코드 취약성 ... 50
네트워크 연결성 ... 51
결과 보기 ... 52
결과 세부 정보 보기 ... 53
Amazon Inspector 점수 보기 ... 56
Amazon Inspector 점수 ... 56
취약성 인텔리전스 ... 58
조사 결과의 심각도 수준 이해 ... 59
소프트웨어 패키지 취약성 심각도 ... 60
코드 취약성 심각도 ... 61
네트워크 연결성 심각도 ... 60
결과 관리 ... 63
조사 결과 필터링 ... 63
Amazon Inspector 콘솔에서 필터 생성 ... 63
조사 결과 안 보이게 하기 ... 64
억제 규칙 생성 ... 64
숨겨진 결과 보기 ... 65
억제 규칙 변경 ... 65
억제 규칙 삭제 ... 66
결과 보고서 내보내기 ... 66
1단계: 권한 확인 ... 67
2단계: S3 버킷 구성 ... 69
3단계: AWS KMS key구성 ... 72
4단계: 결과 보고서 구성 및 내보내기 ... 75
오류 해결 ... 78
다음을 통해 결과에 대한 응답을 자동화합니다. EventBridge ... 78

이벤트 스키마 ... 79
Amazon Inspector 조사 결과를 알려주는 EventBridge 규칙 생성 ... 81
EventBridge Amazon Inspector 다중 계정 환경용 ... 85
대시보드 ... 86
대시보드 보기 ... 86
대시보드 구성 요소 이해 ... 87
취약성 데이터베이스 검색 ... 90
취약성 데이터베이스 검색 ... 90
CVE 세부 정보 이해 ... 91
CVE 세부 정보 ... 91
취약성 인텔리전스 ... 91
참조 ... 91
익스포트 SBOMs ... 92
Amazon Inspector 형식 ... 92
필터: SBOMs ... 97
구성 및 내보내기 SBOMs ... 98
EventBridge 스키마 ... 100
Amazon Inspector용 Amazon EventBridge 기본 스키마 ... 100
Amazon Inspector 결과 이벤트 스키마 예제 ... 101
Amazon Inspector 최초 스캔 완료 이벤트 스키마 예제 ... 113
Amazon Inspector 적용 범위 이벤트 스키마 예제 ... 116
CI/CD 통합 ... 118
플러그인 통합 ... 118
지원되는 CI/CD 솔루션 ... 119
사용자 지정 통합 ... 119
CI/CD 통합을 위한 계정 설정 ... 120
에 가입 AWS 계정 ... 120
관리자 액세스 권한이 있는 사용자 생성 ... 121
CI/CD 통합을 위한 IAM 역할 구성 ... 122
Amazon Inspector SBOM 생성기 ... 123
지원되는 패키지 유형 ... 124
지원되는 컨테이너 이미지 구성 확인 ... 124
설치 Sbomgen ... 124
사용 Sbomgen ... 126
를 사용하여 프라이빗 레지스트리에 인증 Sbomgen ... 130
의 출력 예제 Sbomgen ... 131

이전 버전 134
Amazon Inspector Dockerfile 검사 136
사용 Sbomgen Dockerfile 검사 136
지원되는 Dockerfile 검사 138
사용자 지정 CI/CD 통합 생성 142
단계 1. 구성 AWS 계정 143
단계 2. 설치 Sbomgen 이진수 143
단계 3. 사용 Sbomgen 143
4단계. Amazon Inspector Scan 호출 API 143
(선택 사항) 5단계. 단일 명령SBOM으로 생성 및 스캔 144
API 출력 형식 144
Jenkins 플러그인 152
단계 1. 설정 AWS 계정 153
단계 2. Amazon Inspector Jenkins 플러그인 설치 153
(선택 사항) 3단계. 에 도커 자격 증명 추가 Jenkins 153
(선택 사항) 4단계. AWS 보안 인증 정보 추가 154
5단계. 에서 CSS 지원 추가 Jenkins 스크립트 154
6단계. 빌드에 Amazon Inspector Scan 추가 154
7단계. Amazon Inspector 취약성 보고서 보기 157
문제 해결 158
TeamCity 플러그인 비활성화 159
GitHub actions 161
사용 CodeCatalyst actions 161
CycloneDX네임스페이스 사용 162
`amazon:inspector:sbom_scanner` 네임스페이스 분류법 162
`amazon:inspector:sbom_generator` 네임스페이스 분류법 163
적용 범위 평가 166
계정 수준 적용 범위 평가 167
Amazon EC2 인스턴스의 적용 범위 평가 167
Amazon EC2 인스턴스 상태 값 168
Amazon ECR 리포지토리의 적용 범위 평가 170
Amazon ECR 리포지토리 스캔 상태 값 170
Amazon ECR 컨테이너 이미지의 적용 범위 평가 171
Amazon ECR 컨테이너 이미지 스캔 상태 값 172
AWS Lambda 함수 적용 범위 평가 173
Lambda 함수 스캔 상태 값 174

다중 계정 관리 ... 175
위임된 관리자 계정 및 멤버 계정 이해 ... 175
위임 관리자 작업 ... 175
멤버 계정 작업 ... 176
관리자 계정 지정하기 ... 177
위임 관리자에 대한 중요 고려 사항 ... 177
위임 관리자를 지정하는 데 필요한 권한 ... 178
위임 관리자 지정 ... 179
멤버 계정에 대한 Amazon Inspector 스캔 활성화 ... 180
멤버 계정 연결 해제 ... 183
위임된 관리자 제거 ... 183
사용량 ... 185
사용량 콘솔 사용 ... 185
Amazon Inspector의 사용 비용 계산 방식 이해 ... 186
Amazon Inspector 무료 평가판 정보 ... 187
보안 ... 188
데이터 보호 ... 188
저장 중 암호화 ... 190
전송 중 암호화 ... 193
ID 및 액세스 관리 ... 194
고객 ... 194
ID를 통한 인증 ... 195
정책을 사용한 액세스 관리 ... 198
Amazon Inspector의 작동 방식 IAM ... 200
자격 증명 기반 정책 예시 ... 206
AWS 관리형 정책 ... 210
서비스 링크 역할 사용 ... 221
문제 해결 ... 235
Amazon Inspector 모니터링 ... 237
CloudTrail 로그 ... 237
규정 준수 확인 ... 240
복원력 ... 242
인프라 보안 ... 242
인시던트 대응 ... 242
통합 ... 244
아마존 인스펙터와 아마존 통합 ECR ... 244

Amazon Inspector와 Security Hub 통합 244
아마존 ECR 통합 244
통합 활성화 245
다중 계정 환경과의 통합 사용 245
Security Hub 통합 245
에서 Amazon Inspector 조사 결과 보기 AWS Security Hub 246
Amazon Inspector와 보안 허브 통합 활성화 및 구성 249
통합 결과 흐름 비활성화 250
Security Hub에서 Amazon Inspector의 보안 제어 항목 보기 250
지원되는 운영 체제 및 프로그래밍 언어 251
지원되는 운영 체제 252
지원되는 운영 체제: Amazon EC2 스캔 252
지원되는 운영 체제: Amazon Inspector를 사용한 Amazon ECR 스캔 255
지원되는 운영 체제: CIS 스캔 256
중단된 운영 체제 257
지원되는 프로그래밍 언어 262
지원되는 프로그래밍 언어: Amazon EC2 에이전트리스 스캔 262
지원되는 프로그래밍 언어: Amazon EC2 심층 검사 263
지원되는 프로그래밍 언어: Amazon ECR 스캔 263
지원되는 런타임 264
지원되는 런타임: Amazon Inspector Lambda 표준 스캔 264
지원되는 런타임: Amazon Inspector Lambda 코드 스캔 265
Amazon Inspector 비활성화 267
Amazon Inspector 비활성화 268
할당량 269
리전 및 엔드포인트 270
Amazon Inspector 스캔용 엔드포인트 API 270
리전별 기능 가용성 274
문서 기록 277
AWS 용어집 288
..... cclxxxix

Amazon Inspector란 무엇인가?

Amazon Inspector는 워크로드를 자동으로 발견하고 소프트웨어 취약성과 의도하지 않은 네트워크 노출 여부를 지속적으로 스캔하는 취약성 관리 서비스입니다. Amazon Inspector는 아마존 EC2인스턴스, 아마존의컨테이너 이미지, Lambda 함수를 검색하고 스캔합니다. ECR Amazon Inspector는 소프트웨어 취약성 또는 의도하지 않은 네트워크 노출을 탐지하면 해당 문제에 대한 상세 보고서인 탐지 결과를 생성합니다. Amazon Inspector 콘솔 또는 에서 결과를 관리할 수 있습니다. API

주제

- Amazon Inspector의 특징
- Amazon Inspector 액세스

Amazon Inspector의 특징

여러 Amazon Inspector 계정을 중앙에서 관리

AWS 환경에 계정이 여러 개 있는 경우 AWS Organizations를 사용하여 단일 계정을 통해 환경을 중앙에서 관리할 수 있습니다. 이 방법을 사용하면 특정 계정을 Amazon Inspector의 위임 관리자 계정으로 지정할 수 있습니다.

한 번의 클릭으로 전체 조직에 대해 Amazon Inspector를 활성화할 수 있습니다. 또한 향후 멤버가 조직에 가입할 때마다 서비스를 자동으로 활성화할 수 있습니다. Amazon Inspector 위임 관리자 계정은 조직 멤버에 대한 결과 데이터와 특정 설정을 관리할 수 있습니다. 여기에는 모든 구성원 계정에 대한 집계된 결과 세부 정보 보기, 구성원 계정 검색 활성화 또는 비활성화, 조직 내 스캔한 리소스 검토 등이 포함됩니다. AWS

환경의 취약성과 네트워크 노출 여부를 지속적으로 스캔

Amazon Inspector를 사용하면 평가 스캔을 수동으로 예약하거나 구성할 필요가 없습니다. Amazon Inspector는 적합한 리소스를 자동으로 검색하고 스캔을 시작합니다. Amazon Inspector는 EC2 인스턴스에 새 패키지 설치, 패치 설치, 리소스에 영향을 미치는 새로운 일반적인 취약성 및 노출 (CVE) 이 게시되는 경우 등 새로운 취약성을 유발할 수 있는 변경 사항에 대응하여 리소스를 자동으로 재스캔하여 리소스 수명 주기 전반에 걸쳐 환경을 계속 평가합니다. 기존 보안 스캔 소프트웨어와 달리 Amazon Inspector는 플릿 성능에 미치는 영향을 최소화합니다.

취약성 또는 오픈 네트워크 경로가 식별되면 Amazon Inspector에서는 사용자가 조사할 수 있도록 결과를 생성합니다. 결과에는 취약성, 영향을 받는 리소스 및 해결 권장 사항에 대한 포괄적인 세부 정보

가 포함됩니다. 결과를 적절하게 수정하면 Amazon Inspector에서 자동으로 수정 사항을 탐지하고 결과를 종결합니다.

Amazon Inspector 위험 점수를 사용하여 정확하게 취약성 평가

Amazon Inspector는 스캔을 통해 환경에 대한 정보를 수집하므로 사용 환경에 맞게 특별히 조정된 심각도 점수를 제공합니다. Amazon Inspector는 취약성에 대한 국가 취약성 데이터베이스 (NVD) 기본 점수를 구성하는 보안 지표를 검사하여 컴퓨팅 환경에 따라 조정합니다. 예를 들어 네트워크를 통해 취약성이 악용될 수 있지만 인스턴스에서 인터넷으로 연결되는 개방형 네트워크 경로를 사용할 수 없는 경우, 서비스는 Amazon EC2 인스턴스의 Amazon Inspector 검색 결과 점수를 낮출 수 있습니다. 이 점수는 CVSS 형식에 따른 것으로, 에서 제공하는 기본 일반 취약성 점수 산정 시스템 (CVSS) 점수를 수정한 것입니다. NVD

Amazon Inspector 대시보드를 사용하여 영향력이 큰 결과 식별

Amazon Inspector 대시보드에서는 환경 전반의 결과를 개괄적으로 볼 수 있습니다. 대시보드에서는 결과의 세부 정보에 액세스할 수 있습니다. 대시보드에는 사용 환경의 스캔 적용 범위, 가장 중요한 결과, 가장 많은 결과가 발견된 리소스에 대한 간소화된 정보가 포함되어 있습니다. Amazon Inspector 대시보드의 위험에 기반한 해결 방법 패널에는 가장 많은 수의 인스턴스와 이미지에 영향을 미치는 결과가 표시됩니다. 이 패널에서는 환경에 가장 큰 영향을 미치는 결과를 쉽게 식별하고, 결과 세부 정보를 검토하고, 제안된 솔루션을 검토할 수 있습니다.

사용자 지정 가능한 보기를 사용하여 결과 관리

대시보드 외에도 Amazon Inspector 콘솔에서 결과 보기를 제공합니다. 이 페이지에는 사용 환경에 대한 모든 결과가 나열되고 개별 결과에 대한 세부 정보가 제공됩니다. 범주 또는 취약성 유형별로 그룹화된 결과를 볼 수 있습니다. 각 보기에서는 필터를 사용하여 결과를 추가로 사용자 지정할 수 있습니다. 필터를 사용하여 원치 않는 결과를 보기에서 숨기는 억제 규칙을 생성할 수도 있습니다.

필터와 억제 규칙을 사용하여 모든 검색 결과 또는 사용자 지정된 결과를 보여주는 결과 보고서를 생성할 수 있습니다. 보고서는 CSV 또는 JSON 형식으로 생성할 수 있습니다.

다른 서비스 및 시스템과 함께 결과 모니터링 및 처리

다른 서비스 및 시스템과의 통합을 지원하기 위해 Amazon Inspector는 결과를 검색 EventBridge 이벤트로 Amazon에 게시합니다. EventBridge 는 결과 데이터를 AWS Lambda 함수 및 Amazon Simple Notification Service (AmazonSNS) 주제와 같은 대상으로 라우팅할 수 있는 서버리스 이벤트 버스 서비스입니다. 를 사용하면 기존 보안 및 규정 준수 워크플로의 일부로 거의 실시간으로 결과를 모니터링하고 처리할 수 있습니다. EventBridge

AWS Security Hub를 활성화한 경우 Amazon Inspector는 결과를 Security Hub에도 게시합니다. Security Hub는 AWS 환경 전반의 보안 상태를 포괄적으로 파악하고 보안 업계 표준 및 모범 사례와 비교하여 환경을 점검하는 데 도움이 되는 서비스입니다. Security Hub를 사용하면 AWS의 조직 보안 상태에 대한 광범위한 분석의 일부로 결과를 더 쉽게 모니터링하고 처리할 수 있습니다.

Amazon Inspector 액세스

Amazon Inspector는 대부분 사용할 수 있습니다. AWS 리전현재 Amazon Inspector가 사용 가능한 모든 리전 목록은 Amazon Web Services 일반 참조에서 Amazon Inspector 엔드포인트 및 할당량을 참조하세요. AWS 리전에 대해 자세히 알아보려면 Amazon Web Services 일반 참조에서 AWS 리전관리를 참조하세요. 각 리전에서 다음과 같은 방법으로 Amazon Inspector를 사용할 수 있습니다.

AWS 관리 콘솔

리소스를 AWS Management Console 생성하고 관리하는 데 사용할 수 있는 브라우저 기반 인터페이스입니다. AWS Amazon Inspector 콘솔은 해당 콘솔의 일부로 Amazon Inspector 계정 및 리소스에 대한 액세스를 제공합니다. Amazon Inspector 콘솔에서 Amazon Inspector 작업을 수행할 수 있습니다.

AWS 명령줄 도구

AWS 명령줄 도구를 사용하면 시스템의 명령줄에서 명령을 실행하여 Amazon Inspector 작업을 수행할 수 있습니다. 명령줄을 사용하는 것이 콘솔을 사용하는 것보다 더 빠르고 편리할 수 있습니다. 작업을 수행하는 스크립트를 작성할 때도 명령줄 도구가 유용합니다.

AWS 두 가지 명령줄 도구 세트, 즉 AWS Command Line Interface (AWS CLI) 와 를 AWS Tools for PowerShell제공합니다. 설치 및 사용에 대한 자세한 내용은 AWS 명령줄 인터페이스 사용 설명서를 참조하십시오. AWS CLI도구 설치 및 사용에 대한 자세한 내용은 사용 AWS Tools for PowerShell 설명서를 참조하십시오. PowerShell

AWS SDKs

AWS 는 SDKs Java, Go, Python, C++ 등을 비롯한 다양한 프로그래밍 언어 및 플랫폼에 대한 라이브러리 및 샘플 코드로 구성되어 있습니다. NET. Amazon Inspector 및 기타에 프로그래밍 방식으로 SDKs 편리하게 액세스할 수 있습니다. AWS 서비스 SDK는 요청에 암호화 방식으로 서명, 오류 관리 및 자동으로 요청 재시도와 같은 작업을 포함합니다. 설치 및 사용에 대한 자세한 내용은 구축 도구를 참조하십시오. AWS SDKs AWS

아마존 인스펙터 REST API

Amazon Inspector를 REST API 사용하면 Amazon Inspector 계정 및 리소스에 포괄적이고 프로그래밍 방식으로 액세스할 수 있습니다. 이를 API 통해 Amazon Inspector에 직접 HTTPS 요청을 보낼 수 있습

니다. 그러나 AWS 명령줄 도구와 달리 이 도구를 API 사용하려면 애플리케이션에서 요청에 서명하기 위한 해시 생성과 SDKs 같은 낮은 수준의 세부 정보를 처리해야 합니다.

Amazon Inspector 시작하기

이 섹션에서는 Amazon Inspector를 활성화하기 전에 고려해야 할 정보와 Amazon Inspector를 활성화하고 Amazon Inspector 콘솔 및 Amazon Inspector 를 사용하여 조사 결과를 보는 방법을 설명하는 시작하기 자습서를 제공합니다API. Amazon Inspector

주제

- Amazon Inspector를 활성화하기 전에
- 시작하기 자습서: Amazon Inspector 활성화

Amazon Inspector를 활성화하기 전에

Amazon Inspector 를 활성화하기 전에 다음 사항을 고려하세요.

Amazon Inspector는 리전 서비스입니다.

데이터는 Amazon Inspector를 활성화 AWS 리전 하는 에 저장됩니다. Amazon Inspector 를 사용하려는 모든 AWS 리전 에 대해 시작하기 자습서의 첫 번째 부분에 있는 단계를 반복합니다.

Amazon Inspector는 서비스 연결 역할 AWSServiceRoleForAmazonInspector2 및 AWSServiceRoleForAmazonInspector2Agentless

서비스 연결 역할은 AWS 서비스에 연결된 AWS Identity and Access Management (IAM)의 역할입니다. AWSServiceRoleForAmazonInspector2 Amazon Inspector가 보안 평가를 수행하는 데 AWS 서비스 필요한 에 액세스할 수 AWSServiceRoleForAmazonInspector2Agentless 있도록 허용합니다.

IAM 관리자 권한이 있는 자격 증명은 Amazon Inspector를 활성화할 수 있습니다.

IAM 또는 를 사용하여 사용자를 생성하여 보안 인증을 보호합니다AWS IAM Identity Center. 이렇게 하면 사용자에게 Amazon Inspector 를 관리하는 데 필요한 권한만 부여할 수 있습니다. 자세한 내용은 AWS 관리형 정책: AmazonInspectorFullAccess을 참조하세요.

하이브리드 스캔이 자동으로 활성화됨

하이브리드 스캔에는 에이전트 기반 스캔 및 에이전트리스 스캔이 포함됩니다. 기본적으로 Amazon Inspector는 모든 적격 Amazon EC2 인스턴스에서 이러한 스캔 방법을 사용합니다. 자세한 내용은 Amazon Inspector로 Amazon EC2 인스턴스 스캔을 참조하세요.

Amazon ECR 스캔 및 Lambda 함수 스캔에는 SSM 에이전트가 필요하지 않습니다.

에이전트 기반 스캔은 SSM 에이전트를 사용하여 소프트웨어 인벤토리를 수집합니다. 에이전트리스 스캔은 Amazon EBS 스냅샷을 사용하여 소프트웨어 인버턴토리를 수집합니다.

> **Note**
>
> 기본적으로 SSM 에이전트는 Amazon Machine Images를 기반으로 Amazon EC2 인스턴스에 이미 설치되어 있습니다. 그러나 경우에 따라 SSM 에이전트를 수동으로 활성화해야 할 수도 있습니다. 자세한 내용은 AWS Systems Manager 사용 설명서의 SSM 에이전트 작업을 참조하세요.

월별 비용은 스캔한 워크로드를 기준으로 합니다.

자세한 내용은 Amazon Inspector 요금을 참조하세요.

시작하기 자습서: Amazon Inspector 활성화

이 시작하기 자습서에서는 독립형 계정 환경(회원 계정) 및 다중 계정 환경(위임된 관리자 계정)에 대해 Amazon Inspector를 활성화하는 방법을 설명합니다. Amazon Inspector 를 활성화하면 자동으로 워크로드를 검색하고 소프트웨어 취약성 및 의도하지 않은 네트워크 노출이 있는지 스캔하기 시작합니다. 위임된 관리자 계정 및 멤버 계정에 대한 자세한 내용은 Amazon Inspector의 위임된 관리자 계정 및 멤버 계정 이해를 참조하세요.

Standalone account environment

이 절차에서는 콘솔에서 Amazon Inspector를 활성화하는 방법을 설명합니다. Amazon Inspector 를 프로그래밍 방식으로 활성화하려면 Amazon Inspector 쉘 스크립트를 사용합니다. 자세한 내용은 의 인스펙터2-enablement-with-cli를 참조하세요 GitHub.

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 시작을 선택합니다.
3. Amazon Inspector 활성화를 선택합니다.

독립형 계정에 대해 Amazon Inspector를 활성화하면 기본적으로 모든 스캔 유형이 활성화됩니다. 자세한 내용은 Amazon Inspector 의 자동 스캔 유형을 참조하세요.

Multi-account environment

이 절차에서는 콘솔에서 Amazon Inspector를 활성화하는 방법을 설명합니다. Amazon Inspector를 프로그래밍 방식으로 활성화하려면 Amazon Inspector 쉘 스크립트를 사용합니다. 자세한 내용은 의 인스펙터2-enablement-with-cli를 참조하세요 GitHub.

> **Note**
>
> 이 절차를 완료하려면 AWS Organizations 관리 계정에 액세스할 수 있고 관리하려는 계정과 동일한 조직에 있어야 합니다. 관리자를 위임하려면 권한이 필요할 수 있습니다. 자세한 내용은 위임 관리자를 지정하는 데 필요한 권한 단원을 참조하십시오.

Amazon Inspector 관리자 위임

1. AWS Organizations 관리 계정에 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. 위임된 관리자 계정 ID 에 위임된 관리자로 AWS 계정 지정할 의 12자리 ID를 입력한 다음 위임을 선택합니다.
3. 확인 창에서 위임을 다시 선택합니다.

위임된 관리자에 대해 Amazon Inspector를 활성화하면 기본적으로 모든 스캔 유형이 활성화됩니다. 자세한 내용은 Amazon Inspector 의 자동 스캔 유형을 참조하세요.

Amazon Inspector의 자동 스캔 유형

Amazon Inspector는 리소스에 소프트웨어 취약성과 의도하지 않은 네트워크 노출이 있는지 모니터링하는 특수 설계된 스캔 엔진을 사용합니다. Amazon Inspector가 소프트웨어 취약성 또는 의도하지 않은 네트워크 노출을 감지하면 결과가 생성됩니다. Amazon Inspector를 처음 활성화하면 Amazon 스캔, Amazon EC2 스캔 ECR 및 Lambda 표준 스캔을 포함한 모든 스캔 유형 에 계정이 자동으로 등록됩니다.

> **Note**
>
> Lambda 코드 스캔은 Lambda 함수 스캔의 선택적 계층으로, 언제든지 활성화할 수 있습니다.

주제

- Amazon Inspector 스캔 유형 개요
- 스캔 유형 활성화
- Amazon Inspector를 사용하여 Amazon EC2 인스턴스 스캔
- Amazon Inspector로 Amazon Elastic Container Registry 컨테이너 이미지 스캔
- Amazon Inspector를 사용하여 AWS Lambda 함수 스캔
- Amazon Inspector에서 스캔 유형 비활성화

Amazon Inspector 스캔 유형 개요

Amazon Inspector는 AWS 환경의 특정 리소스 유형에 초점을 맞춘 다양한 스캔 유형을 제공합니다.

Amazon EC2 스캔

Amazon EC2 스캔을 활성화하면 Amazon Inspector가 EC2 인스턴스에서 다음을 스캔합니다.

- CVE(일반적인 취약성 및 노출도)
- 운영 체제 및 프로그래밍 언어 패키지 취약성
- 네트워크 연결성
- 네트워크 노출 문제

Amazon Inspector는 인스턴스에 설치된 SSM 에이전트를 사용하거나 인스턴스의 Amazon EBS 스냅샷을 통해 스캔을 수행합니다. Amazon 에 대한 스캔에 대한 자세한 내용은 섹션을 EC2참조하세요Amazon Inspector를 사용하여 Amazon EC2 인스턴스 스캔.

> **Note**
>
> 기본적으로 Amazon EC2 스캔을 활성화하면 하이브리드 스캔 모드가 자동으로 활성화됩니다. 자세한 내용은 에이전트리스 스캔을 참조하세요.

Amazon ECR 스캔

Amazon ECR 스캔을 활성화하면 Amazon Inspector는 프라이빗 레지스트리의 모든 Basic 스캔 컨테이너 리포지토리를 지속적인 스캔을 통해 향상된 스캔으로 변환합니다. 푸시할 때만 스캔하거나 포함 규칙을 통해 일부 리포지토리를 스캔하도록 이 설정을 선택적으로 구성할 수도 있습니다. 지난 30일 이내에 푸시되거나 지난 90일 이내에 가져온 모든 이미지는 처음에 스캔됩니다. Amazon Inspector는 기본적으로 90일 동안 이미지를 계속 모니터링합니다. 이 설정은 언제든지 변경할 수 있습니다. Amazon 에 대한 스캔에 대한 자세한 내용은 섹션을 ECR참조하세요Amazon Inspector로 Amazon Elastic Container Registry 컨테이너 이미지 스캔.

Lambda 표준 스캔

Lambda 표준 스캔을 활성화하면 Amazon Inspector는 계정에서 Lambda 함수를 검색하고 즉시 취약성 스캔을 시작합니다. Amazon Inspector는 배포 시 새 Lambda 함수 및 계층을 스캔하고 업데이트되거나 새 공통 취약성 및 노출(CVEs)이 게시될 때 다시 스캔합니다. Lambda 함수 스캔에 대한 자세한 내용은 Amazon Inspector를 사용하여 AWS Lambda 함수 스캔 섹션을 참조하세요.

Lambda 표준 스캔 + Lambda 코드 스캔

이 옵션은 Lambda 표준 스캔과 Lambda 코드 스캔을 결합합니다. Lambda 코드 스캔이 활성화되면 Amazon Inspector는 계정에서 Lambda 함수와 계층을 검색하고 애플리케이션 패키지 종속성에서 코드 취약성을 스캔합니다. Lambda 코드 스캔은 Lambda 함수의 사용자 지정 애플리케이션 코드를 스캔하여 코드 취약성을 찾아냅니다. 이 두 스캔 유형은 함께 활성화해야 합니다. 자세한 내용은 Amazon Inspector Lambda 코드 스캔 섹션을 참조하세요.

스캔 유형 활성화

언제든지 Amazon Inspector 스캔 유형을 활성화할 수 있습니다. 스캔 유형을 활성화하면 Amazon Inspector는 스캔 유형에 적합한 리소스를 즉시 스캔하기 시작합니다. 다음은 각 스캔 유형을 간략하게 설명합니다.

Amazon EC2 스캔

이 스캔 유형은 보안 권고에서 수집한 규칙과 메타데이터를 비교하기 전에 EC2 인스턴스에서 메타데이터를 추출합니다. 이 스캔 유형을 활성화하면 Amazon Inspector는 계정의 모든 적격 인스턴스에서 패키지 취약성 및 네트워크 연결성 문제를 스캔합니다.

Amazon ECR 스캔

이 스캔 유형은 Amazon 에서 컨테이너 이미지를 스캔합니다ECR. 이 스캔 유형을 활성화하면 프라이빗 레지스트리의 스캔 구성 설정을 기본 스캔에서 향상된 스캔으로 변경합니다.

Lambda 표준 스캔

Lambda 표준 스캔은 기본 Lambda 스캔 유형입니다. Lambda 표준 스캔을 활성화하면 계정의 모든 Lambda 함수가 지난 90일 동안 호출되거나 업데이트된 경우 코드 취약성에 대해 스캔됩니다.

Lambda 코드 스캔

Lambda 코드 스캔은 Lambda 함수에서 사용자 지정 애플리케이션 코드를 스캔합니다. Lambda 코드 스캔을 활성화하면 계정의 모든 Lambda 함수가 지난 90일 동안 호출되거나 업데이트된 경우 코드 취약성에 대해 스캔됩니다.

> **Note**
>
> Lambda 코드 스캔을 사용하여 Lambda 표준 스캔 또는 Lambda 표준 스캔을 활성화할 수 있습니다.

사용 가능한 스캔 유형에 대한 보다 포괄적인 개요는 Amazon Inspector를 사용한 자동 리소스 스캔을 참조하세요. 이 섹션에서는 Amazon Inspector 에서 스캔 유형을 활성화하는 방법을 설명합니다.

스캔 활성화

AWS 조직에서 Amazon Inspector의 위임된 관리자인 경우 에서 Amazon Inspector Inspector2-enablement-with-cli에서 개발한 쉘 스크립트를 사용하여 여러 리전의 여러 계정에 대해 다양한 Amazon Inspector 스캔 유형을 자동으로 활성화할 수 있습니다 GitHub. 그렇지 않은 경우 콘솔을 통해 다중 계정 환경에 대해 이 절차를 완료하려면 Amazon Inspector 위임 관리자로 로그인한 상태에서 다음 단계를 완료하세요.

Console

스캔을 활성화하려면

1. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 새 스캔 유형을 활성화하려는 리전을 선택합니다.
3. 탐색 창에서 계정 관리를 선택합니다.
4. 계정 관리 페이지에서 스캔 유형을 활성화할 계정을 선택합니다.
5. 활성화를 선택하고 활성화하려는 스캔 유형을 선택합니다.
6. (권장) 해당 스캔 유형을 활성화 AWS 리전 하려는 각 에서 이 단계를 반복합니다.

API

활성화 API 작업을 실행합니다. 요청에서 스캔을 활성화IDs하는 계정과 idempotency 토큰을 제공하고 `EC2`, `ECRLAMBDA`, 또는 중 하나 이상을 제공하여 해당 유형의 스캔을 `LAMBDA_CODE` `resourceTypes` 활성화합니다.

Amazon Inspector를 사용하여 Amazon EC2 인스턴스 스캔

Amazon Inspector Amazon EC2 스캔은 보안 권고에서 수집한 규칙과 메타데이터를 비교하기 전에 EC2 인스턴스에서 메타데이터를 추출합니다. Amazon Inspector는 인스턴스에서 패키지 취약성 및 네트워크 연결성 문제를 스캔하여 결과를 생성합니다. Amazon Inspector는 24시간마다 네트워크 연결성 스캔을 수행하고 EC2 인스턴스와 연결된 스캔 방법에 따라 가변 주기로 취약성 스캔을 패키징합니다.

패키지 취약성 스캔은 에이전트 기반 또는 에이전트 없음 스캔 방법을 사용하여 수행할 수 있습니다. 이러한 두 스캔 방법 모두 Amazon Inspector가 패키지 취약성 스캔을 위해 EC2 인스턴스 인스턴스에서 소프트웨어 인벤토리를 수집하는 방법과 시기를 결정합니다. 에이전트 기반 스캔은 SSM 에이전트를 사용하여 소프트웨어 인벤토리를 수집하고 에이전트리스 스캔은 Amazon EBS 스냅샷에서 를 사용하여 소프트웨어 인벤토리를 수집합니다.

Amazon Inspector는 계정에 대해 활성화하는 스캔 방법을 사용합니다. Amazon Inspector를 처음 활성화하면 두 스캔 방법을 모두 사용하는 하이브리드 스캔에 계정이 자동으로 등록됩니다. 그러나 언제든지 이 설정을 변경할 수 있습니다. 스캔 유형을 활성화하는 방법에 대한 자세한 내용은 스캔 유형 활성화를 참조하세요. 이 섹션에서는 Amazon EC2 스캔에 대한 정보를 제공합니다.

에이전트 기반 스캔

에이전트 기반 스캔은 모든 적격 인스턴스에서 SSM 에이전트를 사용하여 지속적으로 수행됩니다. 에이전트 기반 스캔의 경우 Amazon Inspector는 이러한 SSM 연결을 통해 설치된 연결 및 플러그인을 사용하여 인스턴스에서 소프트웨어 인벤토리를 수집합니다. Amazon Inspector 에이전트 기반 검사는 운영 체제 패키지에 대한 패키지 취약성 스캔뿐 아니라 Linux 기반 Amazon 인스턴스에 대한 Amazon Inspector 심층 검사 EC2 를 통해 Linux 기반 인스턴스의 애플리케이션 프로그래밍 언어 패키지에 대한 패키지 취약성 탐지도 가능합니다.

다음 프로세스는 Amazon Inspector가 를 사용하여 인벤토리SSM를 수집하고 에이전트 기반 스캔을 수행하는 방법을 설명합니다.

1. Amazon Inspector는 계정에 SSM 연결을 생성하여 인스턴스에서 인벤토리를 수집합니다. 일부 인스턴스 유형(Windows 및 Linux)의 경우 이러한 연결이 개별 인스턴스에 플러그인을 설치하여 인벤토리를 수집합니다.
2. Amazon SSM Amazon Inspector는 를 사용하여 인스턴스에서 패키지 인벤토리를 추출합니다.
3. Amazon Inspector는 추출된 인벤토리를 평가하고 탐지된 취약성에 대한 조사 결과를 생성합니다.

적격 인스턴스

Amazon Inspector는 다음 조건을 충족하는 경우 에이전트 기반 방법을 사용하여 인스턴스를 스캔합니다.

- 인스턴스에는 지원되는 OS가 있습니다. 지원되는 OS 목록은 the section called “지원되는 운영 체제: Amazon EC2 스캔”의 에이전트 기반 스캔 지원 열을 참조하세요.
- 인스턴스는 Amazon Inspector 제외 태그의 스캔에서 EC2 제외되지 않습니다.
- 인스턴스가 SSM 관리됩니다. 에이전트를 확인하고 구성하는 방법은 SSM 에이전트 구성 섹션을 참조하세요.

에이전트 기반 스캔 동작

에이전트 기반 스캔 방법을 사용할 때 Amazon Inspector는 다음과 같은 상황에서 EC2 인스턴스의 새 취약성 스캔을 시작합니다.

- 새 EC2 인스턴스를 시작할 때.
- 기존 EC2 인스턴스(Linux 및 Mac)에 새 소프트웨어를 설치하는 경우.

- Amazon Inspector가 데이터베이스에 새로운 공통 취약성 및 노출(CVE) 항목을 추가하는 경우, 이는 EC2 인스턴스(Linux 및 Mac)와 CVE 관련이 있습니다.

Amazon Inspector는 초기 스캔이 완료될 때 EC2 인스턴스의 마지막 스캔 필드를 업데이트합니다. 그런 다음 Amazon Inspector가 SSM 인벤토리를 평가할 때(기본적으로 30분마다) 또는 해당 인스턴스에 CVE 영향을 미치는 새 인스턴스가 Amazon Inspector 데이터베이스에 추가되어 인스턴스가 다시 스캔될 때 마지막으로 스캔된 필드가 업데이트됩니다.

계정 관리 페이지의 EC2 인스턴스 탭에서 또는 ListCoverage 명령.

SSM 에이전트 구성

Amazon Inspector가 에이전트 기반 스캔 방법을 사용하여 Amazon EC2 인스턴스의 소프트웨어 취약성을 감지하려면 인스턴스가 Amazon EC2 Systems Manager()의 관리형 인스턴스여야 합니다SSM. SSM 관리형 인스턴스에는 SSM 에이전트가 설치 및 실행 중이며 인스턴스를 관리할 수 있는 권한이 SSM 있습니다. 를 사용하여 인스턴스SSM를 이미 관리하고 있는 경우 에이전트 기반 스캔에는 다른 단계가 필요하지 않습니다.

SSM 에이전트는 일부 Amazon Machine Images()에서 생성된 EC2 인스턴스에 기본적으로 설치됩니다AMIs. 자세한 내용은 AWS Systems Manager 사용 설명서의 SSM 에이전트 정보를 참조하세요. 하지만 설치된 경우에도 SSM 에이전트를 수동으로 활성화하고 인스턴스를 관리할 수 있는 SSM 권한을 부여해야 할 수 있습니다.

다음 절차에서는 EC2 인스턴스 IAM 프로파일을 사용하여 Amazon 인스턴스를 관리형 인스턴스로 구성하는 방법을 설명합니다. 이 절차에는 AWS Systems Manager 사용 설명서의 자세한 정보로 연결되는 링크도 제공됩니다.

AmazonSSMManagedInstanceCore는 인스턴스 프로파일을 연결할 때 권장되는 정책입니다. 이 정책에는 Amazon Inspector EC2 스캔에 필요한 모든 권한이 있습니다.

> **Note**
>
> SSM 기본 호스트 SSM 관리 구성을 사용하여 EC2 인스턴스 프로파일을 사용하지 않고도 모든 IAM 인스턴스의 관리를 자동화할 수도 있습니다. 자세한 내용은 기본 호스트 관리 구성을 참조하세요.

Amazon EC2 인스턴스에 SSM 대해 구성하려면

1. 운영 체제 공급업체에서 아직 설치하지 않은 경우 SSM 에이전트를 설치합니다. 자세한 내용은 SSM 에이전트 작업을 참조하세요.
2. AWS CLI 를 사용하여 SSM 에이전트가 실행 중인지 확인합니다. 자세한 내용은 SSM 에이전트 상태 확인 및 에이전트 시작을 참조하세요.
3. 에 인스턴스를 관리할 수 SSM 있는 권한을 부여합니다. IAM 인스턴스 프로파일을 생성하고 인스턴스에 연결하여 권한을 부여할 수 있습니다. 를 사용하는 것이 좋습니다. AmazonSSMManagedInstanceCore 이 정책에는 Amazon Inspector가 스캔에 필요한 SSM 배포자, SSM 인벤토리 및 SSM 상태 관리자에 대한 권한이 있으므로 정책입니다. 이러한 권한으로 인스턴스 프로파일을 생성하고 이를 인스턴스에 연결하는 방법에 대한 지침은 Systems Manager에 대한 인스턴스 권한 구성을 참조하세요.
4. (선택 사항) SSM 에이전트에 대한 자동 업데이트를 활성화합니다. 자세한 내용은 SSM 에이전트에 대한 업데이트 자동화를 참조하세요.
5. (선택 사항) Amazon Virtual Private Cloud(AmazonVPC) 엔드포인트를 사용하도록 Systems Manager를 구성합니다. 자세한 내용은 Amazon VPC 엔드포인트 생성 섹션을 참조하세요.

> **Important**
>
> Amazon Inspector에서 소프트웨어 애플리케이션 인벤토리를 수집하려면 계정에 Systems Manager State Manager 연결이 필요합니다. 해당 연결이 없는 경우 Amazon Inspector에서 자동으로 `InspectorInventoryCollection-do-not-delete`라는 연결을 생성합니다. 또한 리소스 데이터 동기화도 필요하며, 아직 없는 경우 Amazon Inspector에서 자동으로 `InspectorResourceDataSync-do-not-delete`라는 동기화를 생성합니다. 자세한 내용은 AWS Systems Manager 사용 설명서에서 인벤토리의 리소스 데이터 동기화 구성을 참조하세요. 각 계정에는 리전당 리소스 데이터 동기화 수를 설정할 수 있습니다. 자세한 내용은 엔드포인트 및 할당량의 최대 리소스 데이터 동기화 수(리전 AWS 계정 당)를 참조하세요. SSM

SSM 스캔을 위해 생성된 리소스

Amazon Inspector는 Amazon EC2 스캔을 실행하기 위해 계정에 많은 SSM 리소스가 필요합니다. Amazon Inspector EC2 스캔을 처음 활성화하면 다음 리소스가 생성됩니다.

> **Note**
>
> 계정에 대해 Amazon Inspector Amazon EC2 스캔이 활성화되어 있는 동안 이러한 SSM 리소스 중 하나라도 삭제되면 Amazon Inspector는 다음 스캔 간격에서 해당 리소스를 다시 생성하려고 시도합니다.

`InspectorInventoryCollection-do-not-delete`

이는 Amazon Inspector가 Amazon EC2 인스턴스에서 소프트웨어 애플리케이션 인벤토리를 수집하는 데 사용하는 Systems Manager State Manager(SSM) 연결입니다. 계정에 이미 에서 인벤토리를 수집하기 위한 SSM 연결이 있는 경우 `InstanceIds*` Amazon Inspector는 자체 인벤토리를 생성하는 대신 해당 를 사용합니다.

`InspectorResourceDataSync-do-not-delete`

이는 Amazon Inspector가 Amazon EC2 인스턴스에서 Amazon Inspector 소유의 Amazon S3 버킷으로 수집된 인벤토리 데이터를 보내는 데 사용하는 리소스 데이터 동기화입니다 Amazon Inspector. 자세한 내용은 AWS Systems Manager 사용 설명서에서 인벤토리의 리소스 데이터 동기화 구성을 참조하세요.

`InspectorDistributor-do-not-delete`

이는 Amazon Inspector가 Windows 인스턴스를 스캔하는 데 사용하는 SSM 연결입니다. 이 연결은 Windows 인스턴스에 Amazon Inspector SSM 플러그인을 설치합니다. 플러그인 파일이 실수로 삭제된 경우 이 연결을 통해 다음 연결 간격에 다시 설치됩니다.

`InvokeInspectorSsmPlugin-do-not-delete`

이는 Amazon Inspector가 Windows 인스턴스를 스캔하는 데 사용하는 SSM 연결입니다. 이 연결을 통해 Amazon Inspector에서 플러그인을 사용하여 스캔을 시작할 수 있으며, 사용자도 이 연결을 통해 Windows 인스턴스 스캔에 대한 사용자 지정 간격을 설정할 수 있습니다. 자세한 내용은 에 대한 사용자 지정 일정 설정 Windows 인스턴스 스캔 단원을 참조하십시오.

`InspectorLinuxDistributor-do-not-delete`

Amazon Inspector가 Amazon EC2 Linux 심층 검사에 사용하는 SSM 연결입니다. 이 연결은 Linux 인스턴스에 Amazon Inspector SSM 플러그인을 설치합니다.

`InvokeInspectorLinuxSsmPlugin-do-not-delete`

이는 Amazon Inspector가 Amazon EC2 Linux 심층 검사에 사용하는 SSM 연결입니다. 이 연결을 통해 Amazon Inspector에서 플러그인을 사용하여 스캔을 시작할 수 있습니다.

> **Note**
>
> Amazon Inspector Amazon EC2 스캔 또는 심층 검사를 비활성화하면 SSM 리소스`InvokeInspectorLinuxSsmPlugin-do-not-delete`가 더 이상 호출되지 않습니다.

에이전트 없는 스캔

Amazon Inspector는 계정이 하이브리드 스캔 모드일 때 적격 인스턴스에 에이전트리스 스캔 방법을 사용합니다. 하이브리드 스캔 모드에는 에이전트 기반 및 에이전트리스 스캔이 포함되며 Amazon EC2 스캔을 활성화하면 자동으로 활성화됩니다.

에이전트리스 스캔의 경우 Amazon Inspector는 EBS 스냅샷을 사용하여 인스턴스에서 소프트웨어 인벤토리를 수집합니다. 에이전트리스 스캔은 인스턴스에서 운영 체제 및 애플리케이션 프로그래밍 언어 패키지 취약성을 스캔합니다.

> **Note**
>
> Linux 인스턴스에서 애플리케이션 프로그래밍 언어 패키지 취약성을 스캔할 때 에이전트 없는 스캔 방법은 사용 가능한 모든 경로를 검사하는 반면 에이전트 기반 스캔은 기본 경로와 사용자가 Linux 기반 Amazon 인스턴스에 대한 Amazon Inspector 심층 검사 EC2 의 일부로 지정한 추가 경로만 검사합니다. 이로 인해 에이전트 기반 방법을 사용하여 스캔했는지 아니면 에이전트 없는 스캔 방법을 사용하여 스캔했는지에 따라 동일한 인스턴스의 조사 결과가 달라질 수 있습니다.

다음 프로세스는 Amazon Inspector가 EBS 스냅샷을 사용하여 인벤토리를 수집하고 에이전트리스 스캔을 수행하는 방법을 설명합니다.

1. Amazon Inspector는 인스턴스에 연결된 모든 볼륨의 EBS 스냅샷을 생성합니다. Amazon Inspector에서 스냅샷을 사용하는 동안에는 스냅샷이 계정에 저장되고 `InspectorScan` 태그 키로 태그가 지정되고 고유한 스캔 ID가 태그 값으로 지정됩니다.
2. Amazon Inspector는 EBS 직접 APIs 를 사용하여 스냅샷에서 데이터를 검색하고 취약성이 있는지 평가합니다. 탐지된 모든 취약성에 대한 조사 결과가 생성됩니다.
3. Amazon Inspector는 계정에서 생성한 EBS 스냅샷을 삭제합니다.

적격 인스턴스

Amazon Inspector는 다음 조건을 충족하는 경우 에이전트 없는 스캔 방법을 사용하여 인스턴스를 스캔합니다.

- 인스턴스에는 지원되는 OS가 있습니다. 자세한 내용은 의 >상담원 기반 스캔 지원 열을 참조하세요the section called “지원되는 운영 체제: Amazon EC2 스캔”.
- 인스턴스의 상태는 `Unmanaged EC2 instance`, `Stale inventory`또는 입니다`No inventory`.
- 인스턴스는 Amazon에서 지원EBS하며 다음 파일 시스템 형식 중 하나가 있습니다.
  - `ext3`
  - `ext4`
  - `xfs`
- 인스턴스는 Amazon 제외 태그를 통한 스캔에서 EC2 제외되지 않습니다.
- 인스턴스에 연결된 볼륨의 수는 8개 미만이며, 합산 크기가 1200GB 이하입니다.

에이전트 없는 스캔 동작

계정이 하이브리드 스캔을 사용하도록 구성된 경우 Amazon Inspector는 24시간마다 적격 인스턴스에 대해 에이전트 없는 스캔을 수행합니다. Amazon Inspector는 SSM 에이전트가 없는 새 인스턴스 또는 상태가 로 변경된 기존 인스턴스를 포함하여 매시간 새로 적격한 인스턴스를 감지하고 스캔합니다`SSM_UNMANAGED`.

Amazon Inspector는 에이전트리스 스캔 후 EC2 인스턴스에서 추출된 스냅샷을 스캔할 때마다 Amazon 인스턴스의 마지막 스캔 필드를 업데이트합니다.

계정 관리 페이지의 EC2 인스턴스 탭에서 또는 ListCoverage 명령.

스캔 모드 관리

EC2 스캔 모드는 Amazon Inspector가 계정에서 스캔을 수행할 때 사용할 EC2 스캔 방법을 결정합니다. 일반 설정 아래의 스캔 설정 페이지에서 계정의 EC2 스캔 모드를 볼 수 있습니다. 독립형 계정 또는 Amazon Inspector 위임 관리자는 스캔 모드를 변경할 수 있습니다. Amazon Inspector 위임 관리자 권한으로 스캔 모드를 설정하면 조직의 모든 멤버 계정에 해당 스캔 모드가 설정됩니다. Amazon Inspector에는 다음과 같은 스캔 모드가 있습니다.

에이전트 기반 스캔 - 이 스캔 모드에서는 Amazon Inspector가 패키지 취약성을 검사할 때 에이전트 기반 스캔 방법만 사용합니다. 이 스캔 모드는 계정의 SSM 관리형 인스턴스만 스캔하지만 새 의 또는 인

스턴스CVE의 변경 사항에 대한 응답으로 연속 스캔을 제공하는 이점이 있습니다. 에이전트 기반 스캔은 Amazon Inspector에 적격 인스턴스에 대한 심층 검사도 제공합니다. 새로 활성화된 계정에서는 이 모드가 기본 스캔 모드입니다.

하이브리드 스캔 - 이 스캔 모드에서 Amazon Inspector는 에이전트 기반과 에이전트 없음 스캔 방법을 조합하여 패키지 취약성을 스캔합니다. SSM 에이전트가 설치 및 구성된 적격 EC2 인스턴스의 경우 Amazon Inspector는 에이전트 기반 메서드를 사용합니다. SSM 관리되지 않는 적격 인스턴스의 경우 Amazon Inspector는 적격 EBS지원 인스턴스에 에이전트리스 메서드를 사용합니다.

스캔 모드를 변경하는 방법

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 EC2 스캔 모드를 변경할 리전을 선택합니다.
3. 측면 탐색 패널의 일반 설정에서 EC2 스캔 설정 을 선택합니다.
4. 스캔 모드에서 편집을 선택합니다.
5. 스캔 모드를 선택한 다음 변경 사항 저장을 선택합니다.

Amazon Inspector 스캔에서 인스턴스 제외

제외할 수 있습니다.Linux 그리고 Windows Amazon Inspector의 인스턴스는 이러한 인스턴스를 `InspectorEc2Exclusion` 키로 태그 지정하여 스캔합니다. 태그 값을 포함하는 것은 선택 사항입니다. 태그 추가에 대한 자세한 내용은 Amazon EC2 리소스 태그 지정을 참조하세요.

Amazon Inspector 스캔에서 제외할 인스턴스에 태그를 지정하면 Amazon Inspector는 인스턴스를 제외됨으로 표시하고 해당 인스턴스에 대한 조사 결과를 생성하지 않습니다. 하지만 Amazon Inspector SSM 플러그인은 계속 호출됩니다. 플러그인이 호출되지 않도록 하려면 인스턴스 메타데이터의 태그에 대한 액세스를 허용해야 합니다.

> **Note**
>
> 제외된 인스턴스에 대해서는 요금이 부과되지 않습니다.

또한 해당 EBS 볼륨을 태그로 암호화하는 데 사용되는 AWS KMS 키에 태그를 지정하여 에이전트리스 스캔에서 암호화된 볼륨을 제외할 수 있습니다`InspectorEc2Exclusion`. 자세한 내용은 키 태그 지정을 참조하세요.

지원되는 운영 체제

Amazon Inspector는 지원되는 Mac, Windows 및 Linux EC2 인스턴스에서 운영 체제 패키지의 취약성을 스캔합니다. Linux 인스턴스의 경우 Amazon Inspector는 Linux 기반 Amazon 인스턴스에 대한 Amazon Inspector 심층 검사 EC2 를 사용하여 애플리케이션 프로그래밍 언어 패키지에 대한 결과를 생성할 수 있습니다. Mac 및 Windows 인스턴스의 경우 운영 체제 패키지만 스캔됩니다.

SSM 에이전트 없이 스캔할 수 있는 운영 체제를 포함하여 지원되는 운영 체제에 대한 자세한 내용은 섹션을 참조하세요지원되는 운영 체제: Amazon EC2 스캔.

Linux 기반 Amazon 인스턴스에 대한 Amazon Inspector 심층 검사 EC2

Amazon Inspector는 Amazon EC2 스캔 범위를 확장하여 심층 검사를 포함합니다. Amazon Inspector는 심층 검사를 통해 Linux 기반 Amazon EC2 인스턴스의 애플리케이션 프로그래밍 언어 패키지에 대한 패키지 취약성을 감지합니다. Amazon Inspector는 프로그래밍 언어 패키지 라이브러리의 기본 경로를 스캔합니다. 하지만 Amazon Inspector가 기본적으로 스캔하는 경로 외에도 사용자 지정 경로를 구성할 수 있습니다.

> **Note**
>
> 기본 호스트 관리 구성 설정에서 심층 검사를 사용할 수 있습니다. 하지만 인스턴스 프로파일을 생성하고 `ssm:PutInventory` 및 `ssm:GetParameter` 권한을 연결해야 합니다.

Amazon Inspector는 Linux 기반 Amazon EC2 인스턴스에 대한 심층 검사 스캔을 수행하기 위해 Amazon Inspector SSM 플러그인으로 수집된 데이터를 사용합니다. Amazon Inspector SSM 플러그인을 관리하고 Linux에 대한 심층 검사를 수행하기 위해 Amazon Inspector는 `InvokeInspectorLinuxSsmPlugin-do-not-delete` 계정에 SSM 연결을 자동으로 생성합니다. Amazon Inspector는 Linux 기반 Amazon EC2 인스턴스에서 6시간마다 업데이트된 애플리케이션 인벤토리를 수집합니다.

> **Note**
>
> 에 대한 심층 검사는 지원되지 않습니다.Windows 또는 Mac 인스턴스.

이 섹션에서는 Amazon Inspector가 스캔할 사용자 지정 경로를 설정하는 방법을 포함하여 Amazon EC2 인스턴스에 대한 Amazon Inspector 심층 검사를 관리하는 방법을 설명합니다.

주제

- 심층 검사 액세스 또는 비활성화
- Linux용 Amazon Inspector SSM 플러그인 정보
- Amazon Inspector 심층 검사를 위한 사용자 지정 경로
- Amazon Inspector 심층 검사를 위한 사용자 지정 일정
- 지원되는 프로그래밍 언어

심층 검사 액세스 또는 비활성화

> **Note**
>
> 2023년 4월 17일 이후에 Amazon Inspector를 활성화하는 계정의 경우 Amazon EC2 스캔의 일부로 심층 검사가 자동으로 활성화됩니다.

심층 검사를 관리하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home에서 Amazon Inspector 콘솔을 엽니다.
2. 탐색 창에서 일반 설정 을 선택한 다음 Amazon EC2 스캔 설정을 선택합니다.
3. Amazon EC2 인스턴스 에 대한 심층 검사에서 조직 또는 자체 계정 에 대한 사용자 지정 경로를 설정할 수 있습니다.

GetEc2DeepInspectionConfiguration 가 있는 단일 계정에 대해 프로그래밍 방식으로 활성화 상태를 확인할 수 있습니다API. 를 사용하여 여러 계정에 대해 프로그래밍 방식으로 활성화 상태를 확인할 수 있습니다. BatchGetMemberEc2DeepInspectionStatus API.

2023년 4월 17일 이전에 Amazon Inspector를 활성화한 경우 콘솔 배너 또는 UpdateEc2DeepInspectionConfiguration API. Amazon Inspector 에서 조직의 위임된 관리자인 경우 BatchUpdateMemberEc2DeepInspectionStatus API 를 사용하여 자신과 멤버 계정에 대한 심층 검사를 활성화합니다.

를 통해 심층 검사를 비활성화할 수 있습니다. UpdateEc2DeepInspectionConfiguration API. 조직의 멤버 계정으로는 심층 검사를 비활성화할 수 없습니다. 대신 를 사용하여 위임된 관리자가 멤버 계정을 비활성화해야 합니다. BatchUpdateMemberEc2DeepInspectionStatus API.

Linux용 Amazon Inspector SSM 플러그인 정보

Amazon Inspector는 Amazon Inspector SSM 플러그인을 사용하여 Linux 인스턴스에 대한 심층 검사를 수행합니다. Amazon Inspector SSM 플러그인은 `/opt/aws/inspector/bin` 디렉터리의 Linux 인스턴스에 자동으로 설치됩니다. 실행 파일의 이름은 `inspectorssmplugin`입니다.

Amazon Inspector는 Systems Manager Distributor를 사용하여 인스턴스에 플러그인을 배포합니다. 심층 검사 스캔을 수행하려면 Systems Manager Distributor 및 Amazon Inspector가 Amazon EC2 인스턴스 운영 체제를 지원해야 합니다. Systems Manager Distributor가 지원하는 운영 체제에 대한 자세한 내용은 AWS Systems Manager 사용 설명서의 지원되는 패키지 플랫폼 및 아키텍처를 참조하세요.

Amazon Inspector는 Amazon Inspector SSM 플러그인의 심층 검사를 위해 수집된 데이터를 관리하기 위해 다음 파일 디렉터리를 생성합니다.

- `/opt/aws/inspector/var/input`
- `/opt/aws/inspector/var/output` - 이 디렉터리의 `packages.txt` 파일은 심층 검사가 발견한 패키지에 대한 전체 경로를 저장합니다. Amazon Inspector가 인스턴스에서 동일한 패키지를 여러 번 감지하면 `packages.txt` 파일에 패키지가 발견된 각 위치가 나열됩니다.

Amazon Inspector는 플러그인에 대한 로그를 `/var/log/amazon/inspector` 디렉터리에 저장합니다.

Amazon Inspector SSM 플러그인 제거

`inspectorssmplugin` 파일이 실수로 삭제된 경우 SSM 연결`InspectorLinuxDistributor-do-not-delete`은 다음 스캔 간격으로 `inspectorssmplugin` 파일을 다시 설치하려고 시도합니다.

Amazon EC2 스캔을 비활성화하면 플러그인이 모든 Linux 호스트에서 자동으로 제거됩니다.

Amazon Inspector 심층 검사를 위한 사용자 지정 경로

Linux Amazon 인스턴스에 대한 심층 검사 중에 Amazon Inspector가 스캔하도록 사용자 지정 경로를 설정할 수 있습니다. EC2 사용자 지정 경로를 설정하면 Amazon Inspector는 해당 디렉터리의 패키지와 해당 디렉터리의 모든 하위 디렉터리를 스캔합니다.

모든 계정은 최대 5개의 사용자 지정 경로를 정의할 수 있습니다. 조직의 위임된 관리자는 10개의 사용자 지정 경로를 정의할 수 있습니다.

Amazon Inspector는 다음 기본 경로 외에도 모든 사용자 지정 경로를 스캔합니다. Amazon Inspector는 모든 계정을 검색합니다.

- `/usr/lib`
- `/usr/lib64`
- `/usr/local/lib`
- `/usr/local/lib64`

> **Note**
> 사용자 지정 경로는 로컬 경로여야 합니다. Amazon Inspector는 Network File System 마운트 또는 Amazon S3 파일 시스템 마운트와 같은 매핑된 네트워크 경로를 스캔하지 않습니다.

사용자 지정 경로 형식 지정

사용자 지정 경로는 256자를 초과할 수 없습니다. 다음은 사용자 지정 경로가 어떻게 보일 수 있는지 보여주는 검사입니다.

예제 경로

```
/home/usr1/project01
```

> **Note**
> 인스턴스당 패키지 제한은 5,000입니다. 최대 패키지 인벤토리 수집 시간은 15분입니다. Amazon Inspector는 이러한 제한을 피하기 위해 사용자 지정 경로를 선택하는 것이 좋습니다.

Amazon Inspector 콘솔 및 Amazon Inspector를 사용하여 사용자 지정 경로 설정 API

다음 절차에서는 Amazon Inspector 콘솔과 Amazon Inspector 를 사용하여 Amazon Inspector 심층 검사를 위한 사용자 지정 경로를 설정하는 방법을 설명합니다API. 사용자 지정 경로를 설정한 후 Amazon Inspector는 다음 심층 검사에 경로를 포함합니다.

Console

1. 위임된 관리자 AWS Management Console 로 에 로그인하고 https://console.aws.amazon.com/inspector/v2/home에서 Amazon Inspector 콘솔을 엽니다.
2. AWS 리전 선택기를 사용하여 Lambda 표준 스캔을 활성화할 리전을 선택합니다.

3. 탐색 창에서 일반 설정 을 선택한 다음 EC2 스캔 설정 을 선택합니다.
4. 자체 계정의 사용자 지정 경로에서 편집을 선택합니다.
5. 경로 텍스트 상자에 사용자 지정 경로를 입력합니다.
6. 저장(Save)을 선택합니다.

API

를 실행합니다. UpdateEc2DeepInspectionConfiguration 명령. `packagePaths`의 경우 스캔할 경로 배열을 지정합니다.

Amazon Inspector 심층 검사를 위한 사용자 지정 일정

기본적으로 Amazon Inspector는 6시간마다 Amazon EC2 인스턴스에서 애플리케이션 인벤토리를 수집합니다. 그러나 다음 명령을 실행하여 Amazon Inspector가 이 작업을 수행하는 빈도를 제어할 수 있습니다.

예제 명령 1: 연결 ID 및 현재 간격을 보기 위한 연결 나열

다음 명령은 연결 의 연결 ID를 보여줍니다`InvokeInspectorLinuxSsmPlugin-do-not-delete`.

```
aws ssm list-associations \
--association-filter-list "key=AssociationName,value=InvokeInspectorLinuxSsmPlugin-do-not-delete" \
--region your-Region
```

예제 명령 2: 새 간격을 포함하도록 연결 업데이트

다음 명령은 연결 에 대한 연결 ID를 사용합니다`InvokeInspectorLinuxSsmPlugin-do-not-delete`. 의 속도를 6시간`schedule-expression`에서 12시간과 같은 새 간격으로 설정할 수 있습니다.

```
aws ssm update-association \
--association-id "your-association-ID" \
--association-name "InvokeInspectorLinuxSsmPlugin-do-not-delete" \
--schedule-expression "rate(6 hours)" \
--region your-Region
```

> **Note**
>
> 사용 사례에 따라 의 속도를 6시간`schedule-expression`에서 30분과 같은 간격으로 설정하면 일일 ssm 인벤토리 제한 을 초과할 수 있습니다. 이로 인해 결과가 지연되고 일부 오류 상태가 있는 Amazon EC2 인스턴스가 발생할 수 있습니다.

지원되는 프로그래밍 언어

Linux 인스턴스의 경우 Amazon Inspector 심층 검사는 애플리케이션 프로그래밍 언어 패키지 및 운영 체제 패키지에 대한 결과를 생성할 수 있습니다.

Mac 및 Windows 인스턴스의 경우 Amazon Inspector 심층 검사는 운영 체제 패키지에 대한 조사 결과만 생성할 수 있습니다.

지원되는 프로그래밍 언어에 대한 자세한 내용은 지원되는 프로그래밍 언어: Amazon EC2 심층 검사를 참조하세요.

스캔 Windows EC2 Amazon Inspector가 있는 인스턴스

Amazon Inspector는 지원되는 모든 를 자동으로 검색합니다.Windows 인스턴스 및 에는 추가 작업 없이 연속 스캔에 포함됩니다. 지원되는 인스턴스에 대한 자세한 내용은 Amazon Inspector에서 지원하는 운영 체제 및 프로그래밍 언어를 참조하세요. Amazon Inspector 실행 Windows 정기적으로 스캔합니다.Windows 인스턴스는 검색 시 스캔된 다음 6시간마다 스캔됩니다. 그러나 첫 번째 스캔 후 기본 스캔 간격을 조정할 수 있습니다.

Amazon EC2 스캔이 활성화되면 Amazon Inspector는 에 대해 다음과 같은 SSM 연결을 생성합니다.Windows 리소스: `InspectorDistributor-do-not-delete`, `InspectorInventoryCollection-do-not-delete`및 `InvokeInspectorSsmPlugin-do-not-delete`. 에 Amazon Inspector SSM 플러그인을 설치하려면 Windows 인스턴스의 경우 `InspectorDistributor-do-not-delete` SSM 연결은 `AWS-ConfigureAWSPackage` SSM 문서와 `AmazonInspector2-InspectorSsmPlugin` SSM 배포자 패키지 를 사용합니다. 자세한 내용은 용 Amazon Inspector SSM 플러그인 정보 Windows 단원을 참조하십시오. 인스턴스 데이터를 수집하고 Amazon Inspector 결과를 생성하기 위해 `InvokeInspectorSsmPlugin-do-not-delete` SSM 연결은 Amazon Inspector SSM 플러그인을 6시간 간격으로 실행합니다. 그러나 cron 또는 rate 표현식 을 사용하여 이 설정을 사용자 지정할 수 있습니다.

> **Note**
>
> Amazon Inspector 단계에서 Open Vulnerability and Assessment Language(OVAL) 정의 파일을 S3 버킷 로 업데이트했습니다`inspector2-oval-prod-`*`your-AWS-Region`*. Amazon S3 버킷에는 스캔에 사용되는 OVAL 정의가 포함되어 있습니다. 이러한 OVAL 정의는 수정할 수 없습니다. 그렇지 않으면 Amazon Inspector가 릴리스CVEs될 때 새 를 스캔하지 않습니다.

에 대한 Amazon Inspector 스캔 요구 사항 Windows instances

를 스캔하려면 Windows 인스턴스에서 Amazon Inspector는 인스턴스가 다음 기준을 충족하도록 요구합니다.

- 인스턴스는 SSM 관리형 인스턴스입니다. 스캔할 인스턴스 설정에 대한 지침은 SSM 에이전트 구성 섹션을 참조하세요.
- 인스턴스 운영 체제는 지원되는 Windows 운영 체제. 지원되는 운영 체제의 전체 목록은 지원되는 운영 체제: Amazon EC2 스캔 섹션을 참조하세요.
- 인스턴스에 Amazon Inspector SSM 플러그인이 설치되어 있습니다. Amazon Inspector는 검색 시 관리형 인스턴스에 대한 Amazon Inspector SSM 플러그인을 자동으로 설치합니다. 플러그인에 대한 자세한 내용은 다음 주제를 참조하세요.

> **Note**
>
> 호스트가 Amazon에서 인터넷 액세스 VPC 없이 실행 중인 경우 Windows 스캔을 수행하려면 호스트가 리전 Amazon S3 엔드포인트에 액세스할 수 있어야 합니다. Amazon S3 Amazon VPC 엔드포인트를 구성하는 방법을 알아보려면 Amazon Virtual Private Cloud 사용 설명서의 게이트웨이 엔드포인트 생성을 참조하세요. Amazon VPC 엔드포인트 정책에서 외부 S3 버킷에 대한 액세스를 제한하는 경우 인스턴스 평가에 사용되는 OVAL정의를 AWS 리전 저장하는 에서 Amazon Inspector가 관리하는 버킷에 대한 액세스를 구체적으로 허용해야 합니다. 이 버킷의 형식은 `inspector2-oval-prod-`*`REGION`*입니다.

용 Amazon Inspector SSM 플러그인 정보 Windows

Amazon Inspector SSM 플러그인은 Amazon Inspector가 Windows 인스턴스. Amazon Inspector SSM 플러그인은 Windows 의 인스턴스`C:\Program Files\Amazon\Inspector`와 실행 가능한 바이너리 파일의 이름은 입니다`InspectorSsmPlugin.exe`.

Amazon Inspector SSM 플러그인이 수집하는 데이터를 저장하기 위해 다음 파일 위치가 생성됩니다.

- `C:\ProgramData\Amazon\Inspector\Input`
- `C:\ProgramData\Amazon\Inspector\Output`
- `C:\ProgramData\Amazon\Inspector\Logs`

기본적으로 Amazon Inspector SSM 플러그인은 정상 우선 순위 미만으로 실행됩니다.

> **Note**
>
> 스캔할 수 있습니다.Windows 기본 호스트 관리 구성 설정이 인 인스턴스. 하지만 인스턴스 프로파일을 생성하고 `ssm:PutInventory` 권한을 연결해야 합니다.

Amazon Inspector SSM 플러그인 제거

`InspectorSsmPlugin.exe` 파일이 실수로 삭제된 경우 `InspectorDistributor-do-not-delete` SSM 연결은 다음에 플러그인을 다시 설치합니다.Windows 스캔 간격. Amazon Inspector SSM 플러그인을 제거하려면 `AmazonInspector2-ConfigureInspectorSsmPlugin` 문서에서 제거 작업을 사용할 수 있습니다.

또한 Amazon Inspector SSM 플러그인은 모든 에서 자동으로 제거됩니다.Windows Amazon EC2 스캔을 비활성화하면 가 호스트됩니다.

> **Note**
>
> Amazon Inspector 를 비활성화하기 전에 SSM 에이전트를 제거하면 Amazon Inspector SSM 플러그인이 Windows 호스트이지만 더 이상 Amazon Inspector SSM 플러그인으로 데이터를 전송하지 않습니다. 자세한 내용은 Amazon Inspector 비활성화 단원을 참조하십시오.

에 대한 사용자 지정 일정 설정 Windows 인스턴스 스캔

사이의 시간을 사용자 지정할 수 있습니다.Windows Amazon EC2 인스턴스는 를 사용하여 `InvokeInspectorSsmPlugin-do-not-delete` 연결에 대한 cron 표현식 또는 속도 표현식을 설정하여 스캔합니다SSM. 자세한 내용은 AWS Systems Manager 사용 설명서의 참조: Systems Manager 의 Cron 및 Rate 표현식을 참조하거나 다음 지침을 따르세요.

다음 코드 예제 중에서 선택하여 에 대한 스캔 주기를 변경합니다.Windows 속도 표현식 또는 cron 표현식을 사용하는 기본 6시간에서 12시간까지의 인스턴스입니다.

다음 예제에서는 라는 연결에 AssociationId 를 사용해야 합니다`InvokeInspectorSsmPlugin-do-not-delete`. 다음 AWS CLI 명령을 실행AssociationId하여 를 검색할 수 있습니다.

```
$ aws ssm list-associations --association-filter-list
 "key=AssociationName,value=InvokeInspectorSsmPlugin-do-not-delete" --region us-east-1
```

> **Note**
>
> AssociationId 는 리전이므로 먼저 각 에 대한 고유 ID를 검색해야 합니다 AWS 리전. 그런 다음 명령을 실행하여 사용자 지정 스캔 일정을 설정하려는 각 리전의 스캔 주기를 변경할 수 있습니다.Windows 인스턴스.

Example rate expression

```
$ aws ssm update-association \
--association-id "YourAssociationId" \
--association-name "InvokeInspectorSsmPlugin-do-not-delete" \
--schedule-expression "rate(12 hours)"
```

Example cron expression

```
$ aws ssm update-association \
--association-id "YourAssociationId" \
--association-name "InvokeInspectorSsmPlugin-do-not-delete" \
--schedule-expression "cron(0 0/12 * * ? *)"
```

Amazon Inspector로 Amazon Elastic Container Registry 컨테이너 이미지 스캔

Amazon Inspector는 Amazon Elastic Container Registry에 저장된 컨테이너 이미지를 스캔하여 소프트웨어 취약성을 확인하여 패키지 취약성 조사 결과를 생성합니다. Amazon ECR 스캔을 활성화할 때 Amazon Inspector를 프라이빗 레지스트리의 기본 스캔 서비스로 설정합니다. 즉, 프라이빗 레지스트리의 스캔 구성 설정을 기본 스캔에서 향상된 스캔으로 변경해야 합니다.

기본 스캔을 사용하면 푸시 시 스캔하거나 수동 스캔을 수행하도록 리포지토리를 구성할 수 있습니다. 향상된 스캔을 통해 레지스트리 수준에서 운영 체제 및 프로그래밍 언어 패키지 취약성을 스캔합니다. 기본 스캔과 향상된 스캔의 차이점을 side-by-side 비교하려면 Amazon Inspector FAQ를 참조하세요.

> **Note**
>
> 기본 스캔은 Amazon 를 통해 제공 및 청구됩니다ECR. 자세한 내용은 Amazon Elastic Container Registry 요금 섹션을 참조하세요. 향상된 스캔은 Amazon Inspector 를 통해 제공 및 청구됩니다. 자세한 내용은 Amazon Inspector 요금을 참조하세요.

Amazon ECR 스캔을 활성화하는 방법에 대한 자세한 내용은 섹션을 참조하세요스캔 유형 활성화. 조사 결과를 보는 방법에 대한 자세한 내용은 섹션을 참조하세요Amazon Inspector에서 결과 관리. 이미지 수준에서 조사 결과를 보는 방법에 대한 자세한 내용은 Amazon Elastic Container Registry 사용 설명서의 이미지 스캔을 참조하세요. AWS Security Hub 및 Amazon EventBridge 와 같은 기본 스캔에 사용할 수 AWS 서비스 없는 의 조사 결과를 관리할 수도 있습니다.

이 섹션에서는 Amazon ECR 스캔에 대한 정보를 제공하고 Amazon ECR리포지토리에 대한 고급 스캔을 구성하는 방법을 설명합니다.

Amazon 스캔을 위한 ECR 스캔 동작

ECR 스캔을 처음 활성화하고 리포지토리가 연속 스캔을 위해 구성된 경우 Amazon Inspector는 30일 이내에 푸시했거나 지난 90일 이내에 가져온 모든 적격 이미지를 감지합니다. 그런 다음 Amazon Inspector는 감지된 이미지를 스캔하고 스캔 상태를 로 설정합니다`active`. Amazon Inspector는 지난 90일(기본값) 또는 구성한 ECR 재스캔 기간 내에 푸시되거나 풀링된 이미지를 계속 모니터링합니다. 자세한 내용은 ECR 재스캔 기간 구성 단원을 참조하십시오.

연속 스캔을 위해 Amazon Inspector는 다음과 같은 상황에서 컨테이너 이미지의 새로운 취약성 스캔을 시작합니다.

- 새 컨테이너 이미지가 푸시될 때마다
- Amazon InspectorCVE가 데이터베이스에 새로운 공통 취약성 및 노출(CVE) 항목을 추가할 때마다 해당 컨테이너 이미지와 관련이 있습니다(연속 스캔만 해당).

푸시 스캔 시 리포지토리를 로 구성하면 이미지를 푸시할 때만 이미지가 스캔됩니다.

계정 관리 페이지의 컨테이너 이미지 탭에서 또는 ListCoverage API. Amazon Inspector는 다음 이벤트에 대한 응답으로 Amazon ECR 이미지의 필드에서 마지막으로 스캔된 를 업데이트합니다.

- Amazon Inspector에서 컨테이너 이미지의 첫 번째 스캔을 완료한 경우
- Amazon Inspector가 컨테이너 이미지에 영향을 미치는 새로운 일반적인 취약성 및 노출(CVE) 항목이 Amazon Inspector 데이터베이스에 추가되어 컨테이너 이미지를 다시 스캔하는 경우.

지원되는 운영 체제 및 미디어 유형

지원되는 운영 체제에 대한 자세한 내용은 지원되는 운영 체제: Amazon Inspector를 사용한 Amazon ECR 스캔 섹션을 참조하세요.

Amazon ECR 리포지토리의 Amazon Inspector 스캔은 다음과 같은 지원되는 미디어 유형을 다룹니다.

- `"application/vnd.docker.distribution.manifest.v1+json"`
- `"application/vnd.docker.distribution.manifest.v1+prettyjws"`
- `"application/vnd.oci.image.manifest.v1+json"`
- `"application/vnd.docker.distribution.manifest.v2+json"`

> **Note**
>
> 스크래치 이미지 및 `"application/vnd.docker.distribution.manifest.list.v2+json"` 이미지는 지원되지 않습니다.

Amazon ECR리포지토리에 대한 향상된 스캔 구성

Amazon ECR 컨테이너 이미지에 대해 Amazon Inspector 스캔을 활성화하면 프라이빗 레지스트리의 스캔 구성 설정을 기본 스캔에서 향상된 스캔으로 변경합니다. 기본 스캔은 오픈 소스 Clair 프로젝트의 Common Vulnerabilities and Exposures 데이터베이스를 사용합니다. 향상된 스캔은 Amazon Inspector와 통합되며 리포지토리에 대한 자동 및 지속적 스캔을 제공합니다.

기본 스캔을 사용하면 푸시 시 스캔하도록 리포지토리를 구성하거나 수동 스캔을 수행합니다. 향상된 스캔을 통해 Amazon Inspector는 컨테이너 이미지에서 운영 체제 및 프로그래밍 언어 패키지 취약성을 스캔합니다. 자세한 내용은 기본 스캔과 향상된 스캔의 차이점을 side-by-side 비교한 Amazon InspectorFAQs를 참조하세요.

의 리포지토리 수준에서 향상된 스캔을 위한 설정을 관리할 수 있습니다ECR. 푸시 또는 연속 스캔에서 선택할 수 있습니다. 푸시 스캔에서는 이미지를 푸시할 때만 스캔합니다. 연속 스캔에는 푸시 스캔 및 자동 재스캔이 포함됩니다. 포함 필터 를 사용하여 두 옵션의 범위를 세분화할 수 있습니다.

> **Note**
>
> Amazon 컨테이너 이미지에 대해 Amazon Inspector 스캔을 활성화하면 연속 스캔이 활성화됩니다. ECR 그러나 이 옵션을 선택 취소하여 콘솔에 스캔 필터를 적용할 수 있습니다.

고급 스캔 설정을 구성하려면

1. 자격 증명을 사용하여 로그인합니다.
2. 에서 Amazon ECR 콘솔을 엽니다https://console.aws.amazon.com/ecr/.
3. AWS 리전 선택기 드롭다운 메뉴에서 스캔 중인 리포지토리 AWS 리전 가 있는 를 선택합니다.
4. 탐색 창에서 프라이빗 레지스트리 를 선택한 다음 설정 을 선택합니다.
5. 스캔 에서 편집 을 선택한 다음 향상된 스캔 을 선택합니다.
6. (선택 사항) 모든 리포지토리를 연속 스캔하도록 선택 취소하여 연속 스캔 및 푸시 스캔 필터를 구성합니다.
7. 선택 사항을 확인한 다음 저장을 선택합니다.

ECR 재스캔 기간 구성

ECR 재스캔 기간 설정은 Amazon Inspector가 리포지토리에서 컨테이너 이미지를 지속적으로 모니터링하는 기간을 결정합니다. 이미지 푸시 날짜 및 이미지 풀 날짜의 재스캔 기간을 구성할 수 있습니다. 조직에 추가된 새 계정을 포함하여 새 계정의 기본 스캔 기간은 90일입니다.

이미지 푸시 날짜 지속 시간

이미지 푸시 날짜 지속 시간은 Amazon Inspector가 가장 최근 풀 날짜 이후 리포지토리로 푸시된 후 이미지를 지속적으로 모니터링하는 기간을 결정합니다. 다음 옵션은 재스캔 기간으로 사용할 수 있습니다.

- 14일
- 30일
- 60일
- 90일(기본값)
- 180일
- 수명

이미지 가져오기 날짜 지속 시간

이미지 풀 날짜 지속 시간은 Amazon Inspector가 가장 최근 풀 날짜 이후에 이미지를 지속적으로 모니터링하는 기간을 결정합니다. 다음 옵션은 재스캔 기간으로 사용할 수 있습니다.

- 14일
- 30일
- 60일
- 90일(기본값)
- 180일

Amazon Inspector는 구성된 푸시 및 풀 날짜 내에 이미지를 푸시하거나 풀링하는 한 이미지를 계속 모니터링하고 다시 스캔합니다. 구성된 푸시 및 풀 날짜 내에 이미지를 푸시하거나 풀링하지 않은 경우 Amazon Inspector는 이미지 모니터링을 중지합니다.

> **Note**
>
> Amazon Inspector가 이미지 모니터링을 중지하면 이미지 스캔 상태 코드가 로 설정`inactive`되고 사유 코드가 로 설정됩니다`expired`. 그런 다음 연결된 모든 이미지 조사 결과를 종료하도록 예약합니다.

환경에 가장 적합하도록 재스캔 기간을 설정합니다. 예를 들어 이미지를 자주 빌드하는 경우 더 짧은 스캔 기간을 선택합니다. 마찬가지로 이미지를 장기간 사용하는 경우 더 긴 스캔 기간을 선택합니다.

위임된 관리자 계정에서 재스캔 기간을 구성하면 Amazon Inspector는 조직의 모든 멤버 계정에 설정을 적용합니다.

ECR 재스캔 기간을 구성하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 탐색 창에서 일반 설정 을 선택한 다음 ECR 스캔 설정 을 선택합니다.
3. ECR 스캔 설정 의 ECR 재스캔 지속 시간 에서 설정하려는 이미지 푸시 날짜 지속 시간 및 이미지 풀 날짜 지속 시간을 선택합니다.
4. 저장(Save)을 선택합니다. 새 설정이 즉시 적용됩니다.

> **Note**
>
> 푸시 날짜 기간을 늘리면 Amazon Inspector는 연속 스캔을 위해 구성된 리포지토리의 모든 활성 스캔 이미지에 변경 사항을 적용합니다. 그러나 비활성 이미지는 새 기간 내에 푸시했더라도 비활성 상태로 유지됩니다.

Amazon Inspector를 사용하여 AWS Lambda 함수 스캔

Amazon Inspector의 AWS Lambda 함수 및 계층 지원은 지속적인 자동 보안 취약성 평가를 제공합니다. Amazon Inspector는 두 가지 유형의 Lambda 함수 스캔을 제공합니다.

Amazon Inspector Lambda 표준 스캔

기본 Lambda 스캔 유형입니다. Lambda 표준 스캔은 Lambda 함수 및 계층 내의 애플리케이션 종속성을 스캔하여 패키지 취약성 을 확인합니다.

Amazon Inspector Lambda 코드 스캔

이 스캔 유형은 Lambda 함수 및 계층의 사용자 지정 애플리케이션 코드에 코드 취약성이 있는지 스캔합니다. Lambda 표준 스캔을 활성화하거나 Lambda 코드 스캔을 사용하여 Lambda 표준 스캔을 활성화할 수 있습니다.

Lambda 함수 스캔을 활성화하면 Amazon Inspector는 계정에 다음과 같은 AWS CloudTrail 서비스 연결 채널을 생성합니다`cloudtrail:DeleteServiceLinkedChannel`. `cloudtrail:CreateServiceLinkedChannel` 및 Amazon Inspector는 이러한 채널을 관리하고 이를 사용하여 스캔할 CloudTrail 이벤트를 모니터링합니다. 이러한 채널을 통해 계정의 CloudTrail 이벤트를 마치 에 추적이 있는 것처럼 볼 수 있습니다 CloudTrail. 에서 자체 추적을 생성 CloudTrail 하여 계정의 이벤트를 관리하는 것이 좋습니다.

Lambda 함수 스캔을 활성화하는 방법에 대한 자세한 내용은 스캔 유형 활성화를 참조하세요. 이 섹션에서는 Lambda 함수 스캔에 대한 정보를 제공합니다.

Lambda 함수 스캔의 스캔 동작

Amazon Inspector는 활성화되면 계정에서 지난 90일 동안 호출되거나 업데이트된 모든 Lambda 함수를 스캔합니다. Amazon Inspector는 다음과 같은 경우에 Lambda 함수의 취약성 스캔을 시작합니다.

- Amazon Inspector에서 기존 Lambda 함수를 발견하는 즉시
- Lambda 서비스에 새 Lambda 함수를 배포하는 경우

- 기존 Lambda 함수 또는 해당 계층의 애플리케이션 코드 또는 종속성에 대한 업데이트를 배포하는 경우
- Amazon Inspector가 데이터베이스에 새로운 공통 취약성 및 노출(CVE) 항목을 추가할 때마다 해당 항목은 함수와 CVE 관련이 있습니다.

Amazon Inspector는 Lambda 함수가 삭제되거나 검사에서 제외될 때까지 전체 기간 동안 각 Lambda 함수를 모니터링합니다.

계정 관리 페이지의 Lambda 함수 탭에서 또는 를 사용하여 Lambda 함수의 취약성을 마지막으로 확인한 시점을 확인할 수 있습니다. ListCoverage API. Amazon Inspector는 다음 이벤트에 대한 응답으로 Lambda 함수의 마지막 스캔 시간 필드를 업데이트합니다.

- Amazon Inspector에서 Lambda 함수의 첫 번째 스캔을 완료한 경우
- Lambda 함수가 업데이트된 경우
- Amazon Inspector가 해당 함수에 영향을 미치는 새 CVE 항목이 Amazon Inspector 데이터베이스에 추가되어 Lambda 함수를 다시 스캔하는 경우.

지원되는 런타임 및 적합한 함수

Amazon Inspector는 Lambda 표준 스캔 및 Lambda 코드 스캔에 대해 다양한 런타임을 지원합니다. 스캔 유형별 지원되는 런타임 목록은 지원되는 런타임: Amazon Inspector Lambda 표준 스캔 및 지원되는 런타임: Amazon Inspector Lambda 코드 스캔 섹션을 참조하세요.

지원되는 런타임이 외에도 Lambda 함수가 Amazon Inspector 스캔 대상이 되려면 다음 기준을 충족해야 합니다.

- 함수가 지난 90일 이내에 호출 또는 업데이트되었습니다.
- 함수가 `$LATEST`로 표시됩니다.
- 함수가 태그 기준 스캔에서 제외되지 않았습니다.

> **Note**
>
> 지난 90일 이내에 호출 또는 수정되지 않은 Lambda 함수는 스캔에서 자동으로 제외됩니다. 자동으로 제외된 함수가 다시 호출되거나 Lambda 함수 코드가 변경되면 Amazon Inspector에서 해당 함수의 스캔을 재개합니다.

Amazon Inspector Lambda 표준 스캔

Amazon Inspector Lambda 표준 스캔은 Lambda 함수 코드 및 계층에 추가하는 애플리케이션 패키지 종속성의 소프트웨어 취약성을 식별합니다. 예를 들어 Lambda 함수에서 사용하는 `python-jwt` 패키지 버전에 알려진 취약성이 있을 경우 Lambda 표준 스캔은 해당 함수에 대한 결과를 생성합니다.

Amazon Inspector에서 Lambda 함수 애플리케이션 패키지 종속성의 취약성을 탐지한 경우 Amazon Inspector는 상세한 패키지 취약성 유형의 결과를 생성합니다.

스캔 유형 활성화에 대한 지침은 스캔 유형 활성화 섹션을 참조하세요.

> **Note**
>
> Lambda 표준 스캔은 Lambda 런타임 환경에 기본적으로 설치된 종속성을 스캔 AWS SDK하지 않습니다. Amazon Inspector는 함수 코드와 함께 업로드되거나 계층에서 상속된 종속성만 스캔합니다.

> **Note**
>
> Amazon Inspector Lambda 표준 스캔을 비활성화하면 Amazon Inspector Lambda 코드 스캔도 비활성화됩니다.

Lambda 표준 스캔에서 함수 제외

Lambda 함수에 태그를 추가할 수 있으므로 Amazon Inspector Lambda 표준 스캔에서 태그를 제외할 수 있습니다. 스캔에서 함수를 제외하면 실행 불가능한 경보를 방지할 수 있습니다. 제외를 위해 함수에 태그를 지정할 때 태그에는 다음과 같은 키-값 페어가 있어야 합니다.

- 키: `InspectorExclusion`
- 값: `LambdaStandardScanning`

이 주제에서는 스캔에서 제외하기 위해 함수에 태그를 지정하는 방법을 설명합니다. Lambda에 태그를 추가하는 방법에 대한 자세한 내용은 Lambda 함수에서 태그 사용을 참조하세요.

스캔에서 함수를 제외하려면

1. 자격 증명을 사용하여 로그인한 다음 에서 Lambda 콘솔을 엽니다https://console.aws.amazon.com/lambda/.
2. 탐색 창에서 함수를 선택합니다.
3. Amazon Inspector Lambda 표준 스캔에서 제외할 함수의 이름을 선택합니다.
4. 구성(Configuration)을 선택한 다음 태그(Tags)를 선택합니다.
5. 태그 관리를 선택한 다음 새 태그 추가를 선택합니다.
   a. 키에 `InspectorExclusion`를 입력합니다.
   b. [값(Value)]에 `LambdaStandardScanning`을 입력합니다.
6. 저장(Save)을 선택합니다.

Amazon Inspector Lambda 코드 스캔

> **Important**
> 이 기능은 Lambda 함수 조각을 캡처하여 탐지된 취약성을 강조 표시합니다. 이러한 조각은 하드코딩된 보안 인증 정보 및 기타 민감한 자료를 표시할 수 있습니다.

이 기능을 사용하여 Amazon Inspector는 Lambda 함수의 애플리케이션 코드를 스캔하여 데이터 유출, 주입 결함, 누락된 암호화 및 약한 암호화를 감지하는 AWS 보안 모범 사례를 기반으로 코드 취약성을 확인합니다. Amazon Inspector는 자동 추론 및 기계 학습을 사용하여 Lambda 함수 애플리케이션 코드를 평가합니다. 또한 Amazon과 협력하여 개발된 내부 탐지기를 사용하여 정책 위반 및 취약성 CodeGuru 을 식별합니다. 자세한 내용은 CodeGuru 감지기 라이브러리를 참조하세요.

Amazon Inspector는 Lambda 함수 애플리케이션 코드에서 취약성을 감지하면 코드 취약성을 생성합니다. 이 결과 유형에는 문제를 보여주는 코드 조각과 코드에서 문제를 찾을 수 있는 위치가 포함됩니다. 또한 문제를 해결하는 방법도 제안합니다. 제안에는 취약한 plug-and-play 코드 라인을 대체하는 데 사용할 수 있는 코드 블록이 포함되어 있습니다. 이러한 코드 수정은 이 조사 결과 유형에 대한 일반 코드 수정 지침에 추가하여 제공됩니다.

코드 수정 제안은 자동화된 추론 및 생성 인공 지능 서비스를 기반으로 합니다. 일부 코드 수정 제안은 의도한 대로 작동하지 않을 수 있습니다. 채택한 코드 수정 제안에 대한 책임은 사용자에게 있습니다. 그러므로 제안된 코드 수정 방법을 사용하기 전에 항상 검토하세요. 코드가 의도한 대로 작동하는지 확인하기 위해 편집해야 할 수 있습니다. 자세한 내용은 책임 있는 AI 정책 섹션을 참조하세요.

Lambda 코드 스캔은 자체적으로 또는 Lambda 표준 스캔과 함께 활성화할 수 있습니다. 자세한 내용은 스캔 유형 활성화를 참조하세요. 이 기능을 AWS 리전 지원하는 기능에 대한 자세한 내용은 섹션을 참조하세요리전별 기능 가용성.

코드 취약성 결과에서 코드 암호화

CodeGuru 는 Lambda 코드 스캔을 사용하여 코드 취약성 조사 결과와 연결된 것으로 감지된 코드 조각을 저장합니다. 기본적으로 는 코드를 암호화하는 데 사용되는 AWS 소유 키를 CodeGuru 제어합니다. 하지만 Amazon Inspector 를 통해 암호화에 자체 고객 관리형 키를 사용할 수 있습니다API. 자세한 내용은 결과 코드에 대한 저장 중 암호화 섹션을 참조하세요.

Lambda 코드 스캔에서 함수 제외

Lambda 함수에 태그를 추가할 수 있으므로 Amazon Inspector Lambda 코드 스캔에서 태그를 제외할 수 있습니다. 스캔에서 함수를 제외하면 실행 불가능한 경보를 방지할 수 있습니다. 제외를 위해 함수에 태그를 지정할 때 태그에는 다음과 같은 키-값 페어가 있어야 합니다.

- 키 - `InspectorCodeExclusion`
- 값 - `LambdaCodeScanning`

이 주제에서는 코드 스캔에서 제외하기 위해 함수에 태그를 지정하는 방법을 설명합니다. Lambda에 태그를 추가하는 방법에 대한 자세한 내용은 Lambda 함수에서 태그 사용을 참조하세요.

코드 스캔에서 함수를 제외하려면

1. 자격 증명을 사용하여 로그인한 다음 에서 Lambda 콘솔을 엽니다https://console.aws.amazon.com/lambda/.
2. 탐색 창에서 함수를 선택합니다.
3. Amazon Inspector Lambda 코드 스캔에서 제외할 함수의 이름을 선택합니다.
4. 구성(Configuration)을 선택한 다음 태그(Tags)를 선택합니다.
5. 태그 관리를 선택한 다음 새 태그 추가를 선택합니다.
   a. 키에 `InspectorExclusion`를 입력합니다.
   b. [값(Value)]에 `LambdaCodeScanning`을 입력합니다.
6. 저장(Save)을 선택합니다.

Amazon Inspector에서 스캔 유형 비활성화

이 섹션에서는 스캔 유형을 비활성화하는 방법을 설명합니다. 스캔 유형을 비활성화하면 스캔 유형이 생성한 결과에 액세스할 수 없습니다. 스캔 유형 를 다시 활성화하면 Amazon Inspector는 적합한 모든 리소스를 스캔하여 새 결과를 생성합니다.

> **Tip**
>
> 조사 결과를 기록으로 유지하려면 조사 결과 보고서로 Amazon Simple Storage Service(Amazon S3) 버킷으로 내보낼 수 있습니다. 자세한 내용은 아마존 인스펙터 조사 결과 보고서 내보내기 단원을 참조하십시오.

스캔 유형을 비활성화하면 스캔 유형을 비활성화한 AWS 계정에서 다음과 같은 변경 사항이 발생할 수 있습니다.

Amazon EC2 스캔

계정에 대한 Amazon Inspector Amazon EC2 스캔을 비활성화하면 다음 SSM 연결이 삭제됩니다.

- `InspectorDistributor-do-not-delete`
- `InspectorInventoryCollection-do-not-delete`
- `InspectorLinuxDistributor-do-not-delete`
- `InvokeInspectorLinuxSsmPlugin-do-not-delete`
- `InvokeInspectorSsmPlugin-do-not-delete`.

또한 이 연결을 통해 설치된 Amazon Inspector SSM 플러그인은 모든 에서 제거됩니다.Windows 호스트. 자세한 내용은 스캔 Windows EC2 인스턴스 단원을 참조하십시오.

Amazon ECR 스캔

계정에 대한 Amazon ECR 스캔을 비활성화하면 Amazon ECR 스캔 유형 계정이 Amazon Inspector를 사용한 향상된 스캔에서 Amazon 를 사용한 기본 스캔으로 변경됩니다ECR.

Lambda 표준 스캔

계정에 대한 Lambda 표준 스캔을 비활성화하면 스캔 유형이 활성화된 경우 Lambda 코드 스캔을 비활성화합니다. Lambda 표준 스캔을 활성화할 때 Amazon Inspector가 생성한 CloudTrail 서비스 연결 채널도 삭제합니다.

스캔 비활성화

계정에 대한 모든 스캔 유형을 비활성화하면 해당 AWS 리전의 해당 계정에 대한 Amazon Inspector가 비활성화됩니다. 자세한 내용은 Amazon Inspector 비활성화 단원을 참조하십시오.

다중 계정 환경에 대해 이 절차를 완료하려면 Amazon Inspector의 위임 관리자로 로그인한 상태에서 다음 단계를 수행하세요.

Console

스캔을 비활성화하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 스캔을 비활성화할 리전을 선택합니다.
3. 탐색 창에서 계정 관리를 선택합니다.
4. 계정 탭을 선택하여 계정의 스캔 상태를 표시합니다.
5. 스캔을 비활성화할 각 계정의 확인란을 선택합니다.
6. 작업을 선택하고 비활성화 옵션에서 비활성화하려는 스캔 유형을 선택합니다.
7. (권장) 해당 스캔 유형을 비활성화 AWS 리전 하려는 각 에서 이 단계를 반복합니다.

API

비활성화 API 작업을 실행합니다. 요청에서 스캔을 비활성화IDs할 계정을 제공하고 , `EC2`, `ECR` `LAMBDA`또는 중 하나 이상을 `resourceTypes` 제공하여 스캔을 비활성화`LAMBDA_CODE`합니다.

Amazon EC2 인스턴스 운영 체제에 대한 Center for Internet Security(CIS) 스캔

Amazon Inspector는 Amazon EC2 인스턴스 운영 체제를 CIS 스캔(CIS스캔)하여 인터넷 보안 센터에서 설정한 모범 사례 권장 사항에 따라 구성했는지 확인합니다. CIS Security Benchmarks는 시스템을 안전하게 구성하기 위한 업계 표준 구성 기준 및 모범 사례를 제공합니다. 계정에 대해 Amazon Inspector EC2 스캔을 활성화한 후 CIS 스캔을 수행하거나 예약할 수 있습니다. Amazon EC2 스캔을 활성화하는 방법에 대한 자세한 내용은 스캔 유형 활성화를 참조하세요.

Amazon Inspector는 EC2 인스턴스 태그와 정의된 CIS 스캔 일정에 따라 대상 Amazon 인스턴스에 대한 스캔을 수행합니다. Amazon Inspector는 각 대상 인스턴스에 대해 일련의 인스턴스 검사를 수행합니다. 각 검사는 시스템 구성이 특정 CIS 벤치마크 권장 사항을 충족하는지 평가합니다. 각 검사에는 해당 플랫폼에 대한 CIS 벤치마크 권장 사항에 해당하는 CIS 검사 ID와 제목이 있습니다. CIS 스캔이 완료되면 결과를 보고 해당 시스템에 대해 통과, 건너뛰기 또는 실패한 인스턴스 확인을 확인할 수 있습니다.

> **Note**
>
> CIS 스캔을 수행하거나 예약하려면 안전한 인터넷 연결이 있어야 합니다. 하지만 프라이빗 인스턴스에서 CIS 스캔을 실행하려면 VPC 엔드포인트를 사용해야 합니다.

주제

- Amazon Inspector 스캔에 대한 Amazon EC2 인스턴스 요구 사항 CIS
- CIS 스캔 실행
- 를 사용하여 Amazon Inspector CIS 스캔을 관리하기 위한 고려 사항 AWS Organizations
- Amazon Inspector가 소유한 Amazon S3 버킷으로 Amazon Inspector CIS 스캔에 사용
- CIS 스캔 구성 생성
- CIS 스캔 결과 보기
- CIS 스캔 구성 편집
- CIS 스캔 결과 다운로드

Amazon Inspector 스캔에 대한 Amazon EC2 인스턴스 요구 사항 CIS

Amazon EC2 인스턴스에서 CIS 스캔을 실행하려면 Amazon EC2 인스턴스가 다음 기준을 충족해야 합니다.

- 인스턴스 운영 체제는 CIS 스캔에 지원되는 운영 체제 중 하나입니다. 자세한 내용은 Amazon Inspector에서 지원하는 운영 체제 및 프로그래밍 언어를 참조하세요.
- 인스턴스는 Amazon EC2 Systems Manager 인스턴스입니다. 자세한 내용은 AWS Systems Manager 사용 설명서의 SSM 에이전트 작업을 참조하세요.
- Amazon Inspector SSM 플러그인은 인스턴스에 설치됩니다. Amazon Inspector는 관리형 인스턴스에 이 플러그인을 자동으로 설치합니다.
- 인스턴스에는 SSM에 인스턴스를 관리할 수 있는 권한과 해당 인스턴스에 대한 CIS 스캔을 실행할 수 있는 Amazon Inspector 권한을 부여하는 인스턴스 프로파일이 있습니다. 이러한 권한을 부여하려면 AmazonSSMManagedInstanceCore 및 AmazonInspector2ManagedCisPolicy 정책을 IAM 역할에 연결합니다. 그런 다음 IAM 역할을 인스턴스에 인스턴스 프로파일로 연결합니다. 인스턴스 프로파일 생성 및 연결에 대한 지침은 Amazon EC2 사용 설명서의 IAM 역할 작업을 참조하세요.

> **Note**
>
> Amazon 인스턴스에서 스캔을 실행하기 전에 Amazon Inspector 심층 검사를 활성화할 필요는 없습니다. CIS EC2 Amazon Inspector 심층 검사를 비활성화하면 Amazon Inspector가 SSM 에이전트를 자동으로 설치하지만 SSM 에이전트는 더 이상 심층 검사를 실행하도록 호출되지 않습니다. 그러나 결과적으로 계정에 `InspectorLinuxDistributor-do-not-delete` 연결이 존재합니다.

프라이빗 Amazon 인스턴스에서 스캔을 실행하기 위한 Amazon Virtual Private Cloud 엔드포인트 요구 사항 CIS EC2

인터넷 연결을 통해 Amazon EC2 인스턴스에서 CIS 스캔을 실행할 수 있습니다. 하지만 프라이빗 Amazon EC2 인스턴스에서 CIS 스캔을 실행하려면 인터넷을 통해 정보를 전송하지 않도록 Amazon VPC 엔드포인트를 생성해야 합니다.

Systems Manager용 Amazon 엔드포인트를 생성할 때 다음 VPC 엔드포인트가 필요합니다.

- `amazonaws.com.region.ec2messages`
- `amazonaws.com.region.inspector2`
- `amazonaws.com.region.s3`
- `amazonaws.com.region.ssm`
- `amazonaws.com.region.ssmmessages`

자세한 내용은 AWS Systems Manager 사용 설명서의 Systems Manager용 Amazon VPC 엔드포인트 생성을 참조하세요.

> **Note**
>
> 현재 일부 AWS 리전 는 `amazonaws.com.region.inspector2` 엔드포인트를 지원하지 않습니다.

CIS 스캔 실행

온디맨드 방식으로 또는 예약된 반복 CIS 스캔으로 스캔을 실행할 수 있습니다. 스캔을 실행하려면 먼저 스캔 구성을 생성합니다.

스캔 구성을 생성할 때 인스턴스를 대상으로 지정하는 데 사용할 태그 키-값 페어를 지정합니다. 조직의 Amazon Inspector 위임 관리자인 경우 스캔 구성에서 여러 계정을 지정할 수 있으며 Amazon Inspector는 각 계정에서 지정된 태그가 있는 인스턴스를 찾습니다. 스캔에 대한 CIS 벤치마크 수준을 선택합니다. 각 벤치마크에 대해 는 환경마다 필요할 수 있는 다양한 보안 수준에 대한 기준을 제공하도록 설계된 수준 1 및 수준 2 프로파일을 CIS 지원합니다.

- 레벨 1 - 모든 시스템에서 구성할 수 있는 필수 기본 보안 설정을 권장합니다. 이러한 설정을 구현하면 서비스가 거의 중단되지 않거나 중단되지 않습니다. 이러한 권장 사항의 목표는 시스템의 진입점 수를 줄여 전반적인 사이버 보안 위험을 줄이는 것입니다.
- 레벨 2 - 고보안 환경에 대한 고급 보안 설정을 권장합니다. 이러한 설정을 구현하려면 비즈니스 영향의 위험을 최소화하기 위한 계획과 조정이 필요합니다. 이러한 권장 사항의 목표는 규정 준수를 달성하는 데 도움이 되는 것입니다.

레벨 2는 레벨 1을 확장합니다. 레벨 2를 선택하면 Amazon Inspector는 레벨 1 및 레벨 2에 권장되는 모든 구성을 확인합니다.

스캔에 대한 파라미터를 정의한 후 구성을 완료한 후 실행되는 일회성 스캔으로 실행할지 아니면 반복 스캔으로 실행할지 선택할 수 있습니다. 반복 스캔은 선택한 시간에 매일, 매주 또는 매월 실행할 수 있습니다.

> **Tip**
>
> 스캔이 실행되는 동안 시스템에 영향을 미칠 가능성이 가장 낮은 날짜와 시간을 선택하는 것이 좋습니다.

를 사용하여 Amazon Inspector CIS 스캔을 관리하기 위한 고려 사항 AWS Organizations

조직에서 CIS 스캔을 실행하면 Amazon Inspector 위임 관리자와 멤버 계정은 CIS 스캔 구성 및 스캔 결과와 다르게 상호 작용합니다.

Amazon Inspector 위임 관리자가 CIS 스캔 구성 및 스캔 결과와 상호 작용하는 방법

위임된 관리자가 모든 계정 또는 특정 멤버 계정에 대해 스캔 구성을 생성하면 조직이 구성을 소유합니다. 조직이 소유한 스캔 구성에는 조직 ID를 소유자로 ARN 지정하는 가 있습니다.

```
arn:aws:inspector2:Region:111122223333:owner/OrganizationId/cis-configuration/scanId
```

위임된 관리자는 다른 계정이 스캔 구성을 생성했더라도 조직이 소유한 스캔 구성을 관리할 수 있습니다.

위임된 관리자는 조직의 모든 계정에 대한 스캔 결과를 볼 수 있습니다.

위임된 관리자가 스캔 구성을 생성하고 대상 계정으로 `SELF`를 지정하는 경우 위임된 관리자가 조직을 떠나더라도 스캔 구성을 소유합니다. 그러나 위임된 관리자는 를 대상으로 하는 스캔 구성의 대상`SELF`을 변경할 수 없습니다.

> **Note**
>
> 위임된 관리자는 조직이 소유한 CIS 스캔 구성에 태그를 추가할 수 없습니다.

Amazon Inspector 멤버 계정이 CIS 스캔 구성 및 스캔 결과와 상호 작용하는 방법

멤버 계정이 CIS 스캔 구성을 생성하면 해당 계정이 구성을 소유합니다. 그러나 위임된 관리자는 구성을 볼 수 있습니다. 멤버 계정이 조직을 떠날 경우 위임된 관리자는 구성을 볼 수 없습니다.

> **Note**
> 위임된 관리자는 멤버 계정이 생성하는 스캔 구성을 편집할 수 없습니다.

멤버 계정, 를 대상으로 `SELF` 하는 위임된 관리자 및 독립 실행형 계정은 자신이 생성한 모든 스캔 구성을 나타냅니다. 이러한 스캔 구성에는 계정 ID를 소유자로 ARN 표시하는 이 있습니다.

```
arn:aws:inspector2:Region:111122223333:owner/111122223333/cis-configuration/scanId
```

멤버 계정은 위임된 관리자가 예약한 스캔의 스캔 결과를 포함하여 계정의 CIS 스캔 결과를 볼 수 있습니다.

Amazon Inspector가 소유한 Amazon S3 버킷으로 Amazon Inspector CIS 스캔에 사용

Open Vulnerability and Assessment Language(OVAL)는 컴퓨터 시스템의 기계 상태를 평가하고 보고하는 방법을 표준화하는 정보 보안 작업입니다. 다음 표에는 CIS 스캔에 사용되는 OVAL 정의가 있는 Amazon Inspector 소유 Amazon S3 버킷이 모두 나열되어 있습니다. CIS 스캔에 필요한 Amazon Inspector 스테이지 OVAL 정의 파일입니다. 필요한 VPCs 경우 Amazon Inspector 소유 Amazon S3 버킷을 허용 목록에 추가해야 합니다.

> **Note**
> 다음 각 Amazon Inspector 소유 Amazon S3 버킷에 대한 세부 정보는 변경될 수 없습니다. 그러나 새로 지원되는 를 반영하도록 테이블을 업데이트할 수 있습니다 AWS 리전. Amazon Inspector 소유 Amazon S3 버킷은 다른 Amazon S3 작업이나 자체 Amazon S3 버킷에 사용할 수 없습니다.

CIS 버킷	AWS 리전
`cis-datasets-prod-arn-5908f6f`	유럽(스톡홀름)

CIS 버킷	AWS 리전
cis-datasets-prod-bah-8f88801	중동(바레인)
cis-datasets-prod-bjs-0f40506	중국(베이징)
cis-datasets-prod-bom-435a167	아시아 태평양(뭄바이)
cis-datasets-prod-cdg-f3a9c58	유럽(파리)
cis-datasets-prod-cgk-09eb12f	아시아 태평양(자카르타)
cis-datasets-prod-cmh-63030b9	미국 동부(오하이오)
cis-datasets-prod-cpt-02c5c6f	아프리카(케이프타운)
cis-datasets-prod-dub-984936f	유럽(아일랜드)
cis-datasets-prod-fra-6eb96eb	유럽(프랑크푸르트)
cis-datasets-prod-gru-de69f99	남아메리카(상파울루)
cis-datasets-prod-hkg-8e30800	아시아 태평양(홍콩)
cis-datasets-prod-iad-8438411	미국 동부(버지니아 북부)
cis-datasets-prod-icn-f4eff1c	아시아 태평양(서울)
cis-datasets-prod-kix-5743b21	아시아 태평양(오사카)
cis-datasets-prod-lhr-8b1fbd0	유럽(런던)
cis-datasets-prod-mxp-7b1bbce	유럽(밀라노)
cis-datasets-prod-nrt-464f684	아시아 태평양(도쿄)
cis-datasets-prod-osu-5bead6f	AWS GovCloud (미국 동부)
cis-datasets-prod-pdt-adadf9c	AWS GovCloud (미국 서부)
cis-datasets-prod-pdx-acfb052	미국 서부(오레곤)

CIS 버킷	AWS 리전
`cis-datasets-prod-sfo-1515ba8`	미국 서부(캘리포니아 북부)
`cis-datasets-prod-sin-309725b`	아시아 태평양(싱가포르)
`cis-datasets-prod-syd-f349107`	아시아 태평양(시드니)
`cis-datasets-prod-yul-5e0c95e`	캐나다(중부)
`cis-datasets-prod-zhy-5a8eacb`	중국(닝샤)
`cis-datasets-prod-zrh-67e0e3d`	유럽(취리히)

CIS 스캔 구성 생성

이 주제에서는 CIS 스캔 구성을 생성하는 방법을 설명합니다.

CIS 스캔을 실행하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. AWS 리전 드롭다운을 사용하여 CIS 스캔을 실행할 AWS 리전 을 선택합니다.
3. 탐색 창에서 온디맨드 스캔을 선택한 다음 CIS 스캔 을 선택합니다.
4. 새 스캔 생성을 선택합니다.
5. 스캔 구성 이름 에 스캔 구성 이름 을 입력합니다.
6. 대상 리소스 태그 에 스캔하려는 인스턴스의 키와 해당 값을 입력합니다. 각 키에 대해 최대 5개의 서로 다른 값과 스캔에 포함할 총 25개의 태그를 지정할 수 있습니다.
7. CIS 벤치마크 수준 의 경우 기본 보안 구성의 경우 수준 1을 선택하고 고급 보안 구성의 경우 수준 2를 선택할 수 있습니다.
8. 대상 계정 에서 CIS 스캔에 포함할 계정을 지정합니다. 자세한 내용은 를 사용하여 Amazon Inspector CIS 스캔을 관리하기 위한 고려 사항 AWS Organizations 단원을 참조하십시오.

   계정이 위임된 관리자 계정인 경우 모든 계정 또는 계정 지정을 선택할 수 있습니다. 모든 계정 옵션은 조직의 모든 계정을 대상으로 합니다. 계정 지정은 조직의 개별 계정만 대상으로 합니다. 이 옵션을 선택하면 계정 번호를 쉼표로 구분하여 두 개 이상의 계정을 지정할 수 있습니다. 계정 ID `SELF` 대신 를 입력하여 계정에 대한 스캔 구성을 생성할 수도 있습니다.

계정이 조직의 독립 실행형 계정 또는 멤버 계정인 경우 자체를 선택하여 계정에 대한 스캔 구성을 생성할 수 있습니다.

9. 일정 에서 스캔 구성 생성을 마치는 즉시 실행되는 일회용 스캔 또는 지정한 시점에 실행되는 반복 스캔 을 선택합니다.
10. 선택 사항을 확인한 다음 생성을 선택합니다.

CIS 스캔 결과 보기

Amazon Inspector는 고유한 스캔 ID로 스캔 결과를 실행하고 수집하는 모든 스캔 구성에 대한 스캔 작업을 생성합니다. CIS 스캔 결과는 90일 동안 사용할 수 있습니다. 검사 또는 스캔된 리소스로 CIS 스캔 결과를 볼 수 있습니다.

- 검사별로 집계된 스캔 결과 - 스캔 중에 수행된 각 개별 검사별로 스캔 결과를 그룹화합니다. 각 검사에 대해 실패한 리소스, 건너뛴 리소스 또는 전달된 리소스 수에 대한 보고서를 가져옵니다.
- 스캔된 리소스별로 집계된 스캔 결과 - 스캔 중에 스캔 대상인 각 스캔된 리소스별로 스캔 결과를 그룹화합니다. 각 리소스에 대해 리소스가 실패, 건너뛰기 또는 전달된 검사 수에 대한 보고서를 가져옵니다.

이 주제에서는 CIS 스캔 결과를 보는 방법을 설명합니다.

스캔 결과를 보려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. AWS 리전 드롭다운을 사용하여 CIS 스캔 구성을 생성한 AWS 리전 를 선택합니다.
3. 탐색 창에서 온디맨드 스캔을 선택한 다음 CIS 스캔 을 선택합니다.
4. 스캔 결과 탭을 선택합니다.
5. 예약 기준 열에서 보려는 스캔 일정 ID를 선택합니다. 또는 보려는 스캔 일정 ID가 있는 행을 선택한 다음 세부 정보 보기 를 선택합니다.
6. 검사를 선택하여 수행된 각 검사를 보거나 스캔된 리소스를 선택하여 스캔 중에 대상으로 지정된 각 스캔된 리소스를 봅니다.

예약된 CIS 스캔에 대한 세부 정보를 볼 수도 있습니다.

예약된 CIS 스캔에 대한 세부 정보를 보려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. AWS 리전 드롭다운을 사용하여 CIS 스캔 구성을 생성한 AWS 리전 를 선택합니다.
3. 탐색 창에서 온디맨드 스캔을 선택한 다음 CIS 스캔을 선택합니다.
4. 예약됨 탭을 선택합니다.
5. 스캔 구성 이름 열에서 보려는 스캔 구성의 이름을 선택합니다. 또는 보려는 스캔 구성이 있는 행을 선택한 다음 세부 정보 보기 를 선택합니다.

CIS 스캔 구성 편집

이 주제에서는 CIS 스캔 구성을 편집하는 방법을 설명합니다.

CIS 스캔 구성을 편집하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. AWS 리전 드롭다운을 사용하여 CIS 스캔 구성을 생성한 AWS 리전 를 선택합니다.
3. 탐색 창에서 온디맨드 스캔을 선택한 다음 CIS 스캔 을 선택합니다.
4. 예약됨 탭을 선택합니다.
5. 편집하려는 스캔 구성이 있는 행을 선택한 다음 편집을 선택합니다.

CIS 스캔 결과 다운로드

Amazon Inspector 콘솔 또는 를 사용하여 CIS 스캔CSV의 PDF 또는 를 다운로드할 수 있습니다API.

> **Note**
>
> 05/03/2024 이후에 수집된 CIS 스캔에 대한 CIS 스캔 결과 CSV 파일만 다운로드할 수 있습니다.

이 주제에서는 Amazon Inspector 콘솔을 사용하여 CIS 스캔을 다운로드하는 방법을 설명합니다.

콘솔에서 CIS 스캔 결과를 다운로드하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. AWS 리전 드롭다운을 사용하여 CIS 스캔 구성을 생성한 AWS 리전 를 선택합니다.
3. 탐색 창에서 온디맨드 스캔을 선택한 다음 CIS 스캔 을 선택합니다.
4. 스캔 결과 탭을 선택합니다.
5. 예약자 열에서 보려는 스캔 일정 ID를 선택합니다. 또는 보려는 스캔 일정 ID가 있는 행을 선택한 다음 세부 정보 보기 를 선택합니다.
6. 다운로드를 선택한 다음 PDF 또는 를 선택합니다CSV. 계정이 위임된 관리자 계정인 경우 계정 선택을 선택하여 특정 멤버 계정에 대한 결과를 다운로드할 수 있습니다.

Amazon Inspector 조사 결과 이해

Amazon Inspector는 Amazon EC2 인스턴스의 취약성, Amazon 의 컨테이너 이미지 ECR또는 AWS Lambda 함수를 감지할 때 결과를 생성합니다. 조사 결과는 AWS 리소스 중 하나에 영향을 미치는 취약성에 대한 자세한 보고서입니다.

조사 결과는 취약성의 이름을 따서 명명되며 심각도 등급, 영향을 받는 AWS 리소스에 대한 정보, 탐지된 취약성을 해결하는 방법을 설명하는 세부 정보를 제공합니다. Amazon Inspector는 모든 활성 조사 결과를 수정할 때까지 저장합니다.

조사 결과를 수정하면 Amazon Inspector가 자동으로 조사 결과를 닫습니다. 30일 후 Amazon Inspector는 결과를 자동으로 삭제합니다.

> **Note**
>
> 취약성을 유발한 문제가 다시 발생하는 경우 Amazon Inspector는 조사 결과 종결 후 7일 이내에 종결된 조사 결과를 다시 엽니다.

Amazon Inspector 를 비활성화하면 24시간 후에 조사 결과가 제거됩니다. 리소스가 종료되면 리소스와 관련된 모든 결과는 7일 후에 제거됩니다. 가 계정을 AWS 일시 중지하면 90일 후에 조사 결과가 제거됩니다. 중지된 인스턴스에 대한 결과는 활성 상태로 유지됩니다.

조사 결과 상태

Amazon Inspector는 결과를 다음 상태로 분류합니다.

활성

Amazon Inspector는 수정되지 않은 결과를 활성 로 분류합니다.

표시되지 않음

Amazon Inspector는 하나 이상의 억제 규칙이 적용되는 결과를 억제됨으로 분류합니다.

종결됨

조사 결과가 수정되면 Amazon Inspector는 조사 결과를 종료됨 으로 분류합니다.

주제

- Amazon Inspector 조사 결과 유형
- Amazon Inspector 조사 결과 보기
- Amazon Inspector 조사 결과에 대한 세부 정보 보기
- Amazon Inspector 점수 보기 및 취약성 인텔리전스 세부 정보 이해
- Amazon Inspector 조사 결과의 심각도 수준 이해

Amazon Inspector 조사 결과 유형

이 섹션에서는 Amazon Inspector 의 다양한 결과 유형을 설명합니다.

주제

- 패키지 취약성
- 코드 취약성
- 네트워크 연결성

패키지 취약성

패키지 취약성 조사 결과는 일반적인 취약성 및 노출()에 노출된 AWS 환경의 소프트웨어 패키지를 식별합니다CVEs. 공격자는 이러한 패치되지 않은 취약성을 악용하여 데이터의 기밀성, 무결성 또는 가용성을 손상시키거나 다른 시스템에 액세스할 수 있습니다. CVE 시스템은 공개적으로 알려진 정보 보안 취약성 및 노출에 대한 참조 방법입니다. 자세한 내용은 https://www.cve.org/ 참조하세요.

Amazon Inspector는 EC2 인스턴스, ECR 컨테이너 이미지 및 Lambda 함수에 대한 패키지 취약성 결과를 생성할 수 있습니다. 패키지 취약성 결과에는 이 결과 유형에만 적용되는 추가 세부 정보가 있는데, 바로 Inspector 점수 및 취약성 인텔리전스입니다.

코드 취약성

코드 취약성 결과는 공격자가 악용할 수 있는 코드 라인을 식별합니다. 코드 취약성에는 주입 결함, 데이터 유출, 취약한 암호화, 코드의 암호화 누락 등이 있습니다.

Amazon Inspector는 애플리케이션 코드의 전반적인 보안 규정 준수를 분석하는 자동 추론 및 기계 학습을 사용하여 Lambda 함수 애플리케이션 코드를 평가합니다. Amazon 와 공동으로 개발된 내부 탐지기를 기반으로 정책 위반 및 취약성을 식별합니다 CodeGuru. 가능한 감지 목록은 CodeGuru 감지기 라이브러리를 참조하세요.

> **Important**
>
> Amazon Inspector 코드 스캔은 코드 스니펫을 캡처하여 탐지된 취약성을 강조 표시합니다. 이러한 스니펫에는 하드코딩된 보안 인증 또는 기타 민감한 자료가 일반 텍스트로 표시될 수 있습니다.

Amazon Inspector Lambda 코드 스캔이 활성화된 경우 Amazon Inspector는 Lambda 함수에 대한 코드 취약성 결과를 생성할 수 있습니다. Amazon Inspector

코드 취약성과 관련하여 감지된 코드 조각은 CodeGuru 서비스에 저장됩니다. 기본적으로 에서 제어하는 AWS 소유 키 CodeGuru 는 코드를 암호화하는 데 사용되지만 Amazon Inspector 를 통해 자체 고객 관리형 키를 사용하여 암호화할 수 있습니다API. 자세한 정보는 결과 코드에 대한 저장 중 암호화 섹션을 참조하세요.

네트워크 연결성

네트워크 연결성 조사 결과에 따르면 환경에 Amazon EC2 인스턴스에 대한 개방형 네트워크 경로가 있는 것으로 나타납니다. 이러한 결과는 인터넷 게이트웨이(Application Load Balancer 또는 Classic Load Balancer 뒤에 있는 인스턴스 포함), VPC 피어링 연결 또는 가상 게이트웨이를 VPN 통한 와 같은 VPC 엣지에서 TCP 및 UDP 포트에 연결할 수 있을 때 나타납니다. 이러한 결과는 잘못 관리된 보안 그룹, 액세스 제어 목록 또는 인터넷 게이트웨이와 같이 지나치게 허용적인 네트워크 구성이나 잠재적으로 악의적인 액세스를 허용할 수 있는 네트워크 구성을 강조합니다.

Amazon Inspector는 Amazon EC2 인스턴스에 대한 네트워크 연결 가능성 조사 결과만 생성합니다. Amazon Inspector는 24시간마다 네트워크 연결성 결과에 대한 스캔을 수행합니다.

Amazon Inspector는 네트워크 경로를 스캔할 때 다음 구성을 평가합니다.

- Amazon EC2 인스턴스
- AWS Lambda 함수
- Application Load Balancers
- Direct Connect
- Elastic Load Balancer
- 탄력적 네트워크 인터페이스
- 인터넷 게이트웨이

- 네트워크 액세스 제어 목록
- 라우팅 테이블
- 보안 그룹
- 서브넷
- 가상 프라이빗 클라우드
- 가상 프라이빗 게이트웨이
- VPC 엔드포인트
- VPC 게이트웨이 엔드포인트
- VPC 피어링 연결
- VPN 연결

Amazon Inspector 조사 결과 보기

Amazon Inspector 콘솔 및 Amazon Inspector `ListFindings` 에서 Amazon Inspector 조사 결과를 볼 수 있습니다API. Amazon Inspector 콘솔의 Amazon Inspector 대시보드와 조사 결과 화면에서 조사 결과를 볼 수 있습니다. AWS Security Hub 및 Amazon Elastic Container Registry(AmazonECR)에서도 조사 결과를 볼 수 있습니다. 기본적으로 Amazon Inspector 대시보드 및 조사 결과 화면에는 활성 조사 결과가 표시됩니다. 범주별로 조사 결과를 볼 수도 있습니다. 이 섹션의 절차에서는 Amazon Inspector 콘솔 및 Amazon Inspector 에서 조사 결과를 보는 방법을 설명합니다API.

Console

Amazon Inspector 조사 결과를 보려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. (선택 사항) 탐색 창에서 대시보드 를 선택합니다. 대시보드는 환경에 대한 적용 범위와 중요한 조사 결과에 대한 개요만 보여줍니다.
3. (선택 사항) 탐색 창에서 결과 를 선택합니다. 조사 결과 화면에는 상태 및 필터 기준을 기준으로 조사 결과를 필터링할 수 있는 테이블의 모든 활성 조사 결과가 표시됩니다. 뷰에서 조사 결과를 제외하는 금지 규칙을 생성할 수도 있습니다. 조사 결과의 이름을 선택하여 조사 결과의 세부 정보를 볼 수 있습니다.
4. (선택 사항) 탐색 창에서 다음 옵션 중 하나를 선택하여 범주별로 조사 결과를 확인합니다.
   - 취약성별 - 가장 중요한 취약성을 표시합니다.

- 계정별 - 모든 계정과 스캔 범위 및 중요하고 심각도가 높은 등급 의 총 조사 결과 수를 표시합니다.

> **Note**
> 이 범주는 위임된 관리자만 사용할 수 있습니다.

- 인스턴스별 - 가장 취약한 Amazon EC2 인스턴스를 표시합니다.

> **Note**
> 이 범주에 그룹화된 조사 결과에는 네트워크 가용성에 대한 정보가 포함되지 않습니다.

- 컨테이너 이미지별 - 가장 취약한 Amazon ECR 컨테이너 이미지를 표시합니다.
- 컨테이너 리포지토리별 - 가장 취약한 리포지토리를 표시합니다.
- Lambda 함수별 - 가장 취약한 Lambda 함수를 표시합니다.

API

Amazon Inspector 조사 결과를 보려면

- ListFindings API 작업을 실행합니다. 요청에서 특정 조사 결과를 반환filterCriteria하도록 를 지정합니다.

Amazon Inspector 조사 결과에 대한 세부 정보 보기

이 섹션의 절차에서는 Amazon Inspector 결과에 대한 세부 정보를 보는 방법을 설명합니다.

결과 세부 정보를 확인하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home에서 Amazon Inspector 콘솔을 엽니다.
2. 결과를 조사 확인할 리전을 선택합니다.
3. 탐색 창에서 결과를 선택하여 결과 목록을 표시합니다.
4. (선택 사항) 필터 막대를 사용하여 특정 결과를 선택합니다. 자세한 내용은 Amazon Inspector 조사 결과 필터링 단원을 참조하십시오.

5. 조사 결과를 선택하여 세부 정보 패널을 확인합니다.

조사 결과 세부 정보 패널에는 조사 결과의 기본 식별 기능이 포함되어 있습니다. 이 기능은 결과 제목과 식별된 취약성에 대한 기본 설명, 제안된 해결 방법, 심각도 점수입니다. 점수에 대한 자세한 내용은 Amazon Inspector 조사 결과의 심각도 수준 이해 섹션을 참조하세요.

결과에 제공되는 세부 정보는 결과 유형 및 영향을 받는 리소스에 따라 다릅니다.

모든 결과에는 조사 결과가 식별된 AWS 계정 ID 번호, 심각도, 조사 결과 유형, 조사 결과가 생성된 날짜, 해당 리소스에 대한 세부 정보가 포함된 리소스 영향 섹션이 포함됩니다.

결과 유형에 따라 결과에 제공되는 해결 방법 및 취약성 인텔리전스 정보가 결정됩니다. 결과 유형에 따라 다양한 결과 세부 정보를 사용할 수 있습니다.

패키지 취약성

패키지 취약성 조사 결과는 EC2 인스턴스, ECR 컨테이너 이미지 및 Lambda 함수에 사용할 수 있습니다. 자세한 내용은 패키지 취약성 섹션을 참조하세요.

패키지 취약성 결과에는 Amazon Inspector 점수 보기 및 취약성 인텔리전스 세부 정보 이해도 포함됩니다.

이 결과 유형에 대한 세부 정보는 다음과 같습니다.

- 수정 버전 있음 – 영향을 받는 패키지의 최신 버전에서 취약성이 수정되었는지 여부를 나타냅니다. 다음 값 중 하나를 사용합니다.
  - `YES` - 영향을 받는 모든 패키지에 수정 버전이 있습니다.
  - `NO` - 영향을 받는 패키지에 수정 버전이 없습니다.
  - `PARTIAL` - 영향을 받는 패키지 중 하나 이상(전부는 아님)에 수정 버전이 있습니다.
- 공격 가능 - 취약성에 알려진 악용 사례가 있음을 나타냅니다.
  - `YES` - 환경에서 발견된 취약성에 알려진 악용 사례가 있습니다. Amazon Inspector에서는 환경 내에서 발생하는 악용 사례를 파악할 수 없습니다.
  - `NO` - 해당 취약성에 알려진 악용 사례가 없습니다.
- 영향을 받는 패키지 - 결과에서 취약한 것으로 식별된 각 패키지와 각 패키지의 세부 정보가 나열됩니다.
- Filepath - 결과와 연결된 EBS 볼륨 ID 및 파티션 번호입니다. 이 필드는 를 사용하여 스캔한 EC2 인스턴스의 결과에 있습니다에이전트 없는 스캔.

- 설치된 버전/수정된 버전 - 현재 설치된 패키지 중에서 취약성이 탐지된 버전 번호입니다. 설치된 버전 번호를 슬래시(/) 뒤의 값과 비교하세요. 두 번째 값은 공통 취약성 및 노출(CVEs) 또는 결과와 관련된 권고에서 제공하는 대로 감지된 취약성을 수정하는 패키지의 버전 번호입니다. 여러 버전에서 취약성이 수정된 경우 이 필드에는 수정 사항이 포함된 최신 버전이 나열됩니다. 수정 사항이 없는 경우 이 값은 `None available`입니다.

  > **Note**
  >
  > Amazon Inspector에서 이 필드를 결과에 포함시키기 전에 결과가 발견된 경우 이 필드의 값은 비어 있습니다. 하지만 수정 사항이 있을 수 있습니다.

- 패키지 관리자 - 이 패키지를 구성하는 데 사용되는 패키지 관리자입니다.
- 해결 - 업데이트된 패키지 또는 프로그래밍 라이브러리를 통해 수정 사항이 제공되는 경우 이 섹션에는 업데이트를 위해 실행할 수 있는 명령이 포함됩니다. 제공된 명령을 복사하여 사용 환경에서 실행할 수 있습니다.

  > **Note**
  >
  > 해결 명령은 공급업체 데이터 피드에서 제공되며 시스템 구성에 따라 달라질 수 있습니다. 자세한 지침은 결과 참조 또는 운영 체제 설명서를 참조하세요.

- 취약성 세부 정보 - National Vulnerability Database(NVD), REDHAT또는 다른 OS 공급업체와 같이 조사 결과에서 CVE 식별된 에 대한 Amazon Inspector 기본 설정 소스에 대한 링크를 제공합니다. 또한 결과에 대한 심각도 점수도 확인할 수 있습니다. 심각도 점수에 대한 자세한 내용은 Amazon Inspector 조사 결과의 심각도 수준 이해 섹션을 참조하세요. 점수 벡터를 포함하여 다음과 같은 점수가 포함됩니다.
  - Exploit Prediction Scoring System(EPSS) 점수
  - Inspector 점수
  - CVSS Amazon의 3.1 CVE
  - CVSS 3.1 시작 NVD
  - CVSS 2.0NVD(해당하는 경우, 이전 의 경우CVEs)
- 관련 취약성 - 결과와 관련된 다른 취약성을 지정합니다. 일반적으로 공급업체의 결정에 CVE따라 CVEs 동일한 패키지 버전 또는 결과와 동일한 그룹 CVEs 내의 다른 버전입니다.

코드 취약성

코드 취약성 결과는 Lambda 함수에만 사용할 수 있습니다. 자세한 내용은 코드 취약성 섹션을 참조하세요. 이 결과 유형에 대한 세부 정보는 다음과 같습니다.

- 수정 버전 있음 - 코드 취약성의 경우 이 값은 항상 YES입니다.
- 탐지기 이름 - 코드 취약성을 탐지하는 데 사용되는 CodeGuru 탐지기의 이름입니다. 가능한 감지 목록은 CodeGuru 감지기 라이브러리를 참조하세요.
- 탐지기 태그 - 탐지기와 연결된 태그는 CodeGuru 태그를 CodeGuru 사용하여 탐지를 분류합니다.
- 관련 CWE - 코드 취약성과 관련된 IDs 일반적인 약점 열거(CWE)입니다.
- 파일 경로 - 코드 취약성이 있는 파일 위치입니다.
- 취약성 위치 - Lambda 코드 스캔 코드 취약성의 경우 이 필드에는 Amazon Inspector에서 취약성을 발견한 정확한 코드 줄이 표시됩니다.
- 제안된 해결 방법 - 결과를 해결하기 위한 코드 편집 방법을 제안합니다.

네트워크 연결성

네트워크 연결성 조사 결과는 EC2 인스턴스에서만 사용할 수 있습니다. 자세한 내용은 네트워크 연결성 섹션을 참조하세요. 이 결과 유형에 대한 세부 정보는 다음과 같습니다.

- 오픈 포트 범위 - EC2 인스턴스에 액세스할 수 있는 포트 범위입니다.
- 개방형 네트워크 경로 - EC2 인스턴스에 대한 개방형 액세스 경로를 표시합니다. 경로에서 항목을 선택하면 자세한 내용을 확인할 수 있습니다.
- 해결 - 오픈 네트워크 경로를 닫는 방법을 권장합니다.

Amazon Inspector 점수 보기 및 취약성 인텔리전스 세부 정보 이해

Amazon Inspector는 Amazon Elastic Compute Cloud(AmazonEC2) 인스턴스 결과에 대한 점수를 생성합니다. Amazon Inspector 콘솔에서 Amazon Inspector 점수 및 취약성 인텔리전스 세부 정보를 볼 수 있습니다. Amazon Inspector 점수는 공통 취약성 점수 체계 의 지표와 비교할 수 있는 세부 정보를 제공합니다. 이러한 세부 정보는 패키지 취약성 조사 결과에만 사용할 수 있습니다. 이 섹션에서는 Amazon Inspector 점수를 해석하고 취약성 인텔리전스 세부 정보를 이해하는 방법을 설명합니다.

Amazon Inspector 점수

Amazon Inspector 점수는 Amazon Inspector가 각 EC2 인스턴스 결과에 대해 생성하는 컨텍스트화된 점수입니다. Amazon Inspector 점수는 기본 CVSS v3.1 점수 정보와 네트워크 도달 가능성 결과 및 이

용 가능성 데이터와 같이 스캔 중에 컴퓨팅 환경에서 수집한 정보를 상호 연관시켜 결정됩니다. 예를 들어 네트워크를 통해 취약성이 악용될 소지가 있지만 취약한 인스턴스에 대한 오픈 네트워크 경로를 인터넷에서 사용할 수 없다고 Amazon Inspector에서 판단하는 경우, Amazon Inspector의 결과 점수는 기본 점수보다 낮을 수 있습니다.

결과의 기본 점수는 공급업체가 제공한 CVSS v3.1 기본 점수입니다. RHEL, Debian 또는 Amazon 공급업체 기본 점수는 다른 공급업체 또는 공급업체가 Amazon Inspector가 국가 취약성 데이터베이스()의 기본 점수를 사용하지 않는 경우에 지원됩니다NVD. Amazon Inspector는 Common Vulnerability Scoring System Version 3.1 Calculator를 사용하여 점수를 계산합니다. 취약성 세부 정보 의 조사 결과 세부 정보에서 개별 조사 결과의 기본 점수 소스를 취약성 소스(또는 packageVulnerabilityDetails.source 결과에서 JSON)

> **Note**
>
> Ubuntu를 실행하는 Linux 인스턴스에는 Amazon Inspector 점수를 사용할 수 없습니다. 이는 Ubuntu가 관련 심각도와 다를 수 있는 자체 취약성 CVE 심각도를 정의하기 때문입니다.

Amazon Inspector 점수 세부 정보

조사 결과의 세부 정보 페이지를 열면 Inspector 점수 및 취약성 인텔리전스 탭을 선택할 수 있습니다. 이 패널에는 기본 점수와 Inspector 점수 간의 차이가 표시됩니다. 이 섹션에서는 Amazon Inspector에서 Amazon Inspector 점수와 소프트웨어 패키지의 공급업체 점수를 조합하여 심각도 등급을 할당하는 방식에 대해 설명합니다. 점수가 서로 다른 경우 이 패널에는 다른 이유에 대한 설명이 표시됩니다.

CVSS 점수 지표 섹션에서 CVSS 기본 점수 지표와 Inspector 점수 간의 비교가 포함된 테이블을 볼 수 있습니다. 비교 지표는 에서 유지 관리하는 CVSS 사양 문서에 정의된 기본 지표입니다.first.org. 다음은 기본 지표의 요약입니다.

공격 벡터

취약성이 악용될 수 있는 컨텍스트입니다. Amazon Inspector 조사 결과의 경우 네트워크, 인접 네트워크 또는 로컬일 수 있습니다.

공격 복잡도

공격자가 취약성을 악용할 때 직면하게 될 난이도를 나타냅니다. 낮은 점수는 공격자가 취약성을 악용하기 위해 충족해야 할 추가 조건이 거의 또는 전혀 없다는 것을 의미합니다. 높은 점수는 공격자가 이 취약성을 이용해 공격에 성공하려면 상당한 노력을 투자해야 한다는 것을 의미합니다.

필수 권한

공격자가 취약성을 악용하는 데 필요한 권한 수준을 나타냅니다.

사용자 상호 작용

이 지표는 이 취약성을 이용한 공격이 성공하려면 공격자가 아닌 다른 사람이 필요한지 여부를 나타냅니다.

범위

한 취약한 구성 요소의 취약성이 취약한 구성 요소의 보안 범위를 벗어난 구성 요소의 리소스에 영향을 미치는지 여부를 나타냅니다. 이 값이 변경되지 않음인 경우 영향을 받은 리소스와 영향을 받는 리소스가 동일합니다. 이 값이 변경됨인 경우 취약한 구성 요소를 악용하여 여러 보안 기관에서 관리하는 리소스에 영향을 미칠 수 있습니다.

기밀성

취약성이 악용될 때 리소스 내 데이터의 기밀성에 미치는 영향 수준을 측정합니다. 점수 범위는 기밀성이 손실되지 않는 없음부터 리소스 내의 모든 정보가 공개되거나 암호 또는 암호화 키 등의 기밀 정보가 공개될 수 있는 높음까지 다양합니다.

무결성

취약성이 악용될 경우 영향을 받는 리소스 내의 데이터 무결성에 미치는 영향 수준을 측정합니다. 공격자가 영향을 받는 리소스 내에서 파일을 수정하면 무결성이 훼손됩니다. 점수 범위는 취약성을 악용하더라도 공격자가 정보를 수정할 수 없는 없음부터 취약성을 악용할 경우 공격자가 일부 또는 모든 파일을 수정할 수 있거나 수정할 수 있는 파일이 심각한 결과를 초래할 수 있는 높음까지 다양합니다.

가용성

취약성이 악용될 때 영향을 받는 리소스의 가용성에 미치는 영향 수준을 측정합니다. 점수 범위는 취약성이 가용성에 전혀 영향을 미치지 않는 없음부터 악용될 경우 공격자가 리소스의 가용성을 완전히 거부하거나 서비스를 사용할 수 없게 만들 수 있는 높음까지 다양합니다.

취약성 인텔리전스

이 섹션에서는 AmazonCVE의 에 대한 사용 가능한 인텔리전스와 Recorded Future, Cybersecurity and Infrastructure Security Agency()와 같은 업계 표준 보안 인텔리전스 소스를 요약합니다CISA.

> **Note**
>
> CISA, Amazon 또는 Recorded Future의 인텔을 모든 에서 사용할 수 있는 것은 아닙니다 CVEs.

콘솔에서 또는 를 사용하여 취약성 인텔리전스 세부 정보를 볼 수 있습니다. BatchGetFindingDetails API. 콘솔에서 다음 세부 정보를 확인할 수 있습니다.

ATT&CK

이 섹션에서는 와 연결된 MITRE 전술, 기법 및 절차(TTPs)를 보여줍니다CVE. 연결된 이 TTPs 표시됩니다. 해당하는 항목이 두 개 이상인 경우 링크를 선택하여 전체 목록을 볼 TTPs 수 있습니다. 전술 또는 기법을 선택하면 MITRE 웹 사이트에서 이에 대한 정보가 열립니다.

CISA

이 섹션에서는 취약성과 연관된 관련 날짜를 다룹니다. Cybersecurity and Infrastructure Security Agency(CISA)가 적극적인 악용의 증거를 기반으로 알려진 악용 취약성 카탈로그에 취약성을 추가한 날짜 및 기한은 시스템이 패치될 것으로 CISA 예상합니다. 이 정보는 에서 제공됩니다CISA.

알려진 멀웨어

이 섹션에는 이 취약성을 악용하는 알려진 악용 키트와 도구가 나열되어 있습니다.

증거

이 섹션에는 해당 취약성과 관련된 가장 중요한 보안 이벤트가 요약되어 있습니다. 중요도 수준이 같은 이벤트가 3개 이상인 경우 가장 최근의 상위 3개 이벤트가 표시됩니다.

최근 보고 시간

이 섹션에는 해당 취약성에 대해 마지막으로 알려진 공개 악용 날짜가 표시됩니다.

Amazon Inspector 조사 결과의 심각도 수준 이해

Amazon Inspector가 결과를 생성하면 결과에 심각도 등급을 할당합니다. 심각도 등급은 조사 결과를 평가하고 우선순위를 정하는 데 도움이 됩니다. 결과의 심각도 등급은 숫자 점수 및 수준, 즉 정보 , 낮음 , 중간 , 높음 및 중요 에 해당합니다. Amazon Inspector는 결과 유형 에 따라 결과의 심각도 등급을 결정합니다. 이 섹션에서는 Amazon Inspector가 각 결과 유형에 대한 심각도 등급을 결정하는 방법을 설명합니다.

소프트웨어 패키지 취약성 심각도

Amazon Inspector는 NVD/CVSS 점수를 소프트웨어 패키지 취약성의 심각도 점수 기준으로 사용합니다. NVD/CVSS 점수는 에서 게시NVD하고 에서 정의한 취약성 심각도 점수입니다CVSS. NVD/CVSS 점수는 공격 복잡성, 익스플로잇 코드 성숙도 및 필요한 권한과 같은 보안 지표의 구성입니다. Amazon Inspector는 취약성의 심각도를 반영하여 1부터 10까지의 숫자로 점수를 산출합니다. 이 점수는 시간이 지나도 일정하게 유지되는 취약성의 본질적인 특성에 따라 취약성의 심각도를 반영하기 때문에 Amazon Inspector에서는 이를 기본 점수로 분류합니다. 또한 이 점수는 배포된 여러 환경 전반에서 합리적인 최악의 영향을 가정한 것입니다. CVSS v3 표준은 CVSS 점수를 다음 심각도 등급으로 매핑합니다.

점수	등급
0	정보
0.1~3.9	낮음
4.0~6.9	중간
7.0~8.9	높음
9.0~10.0	심각

패키지 취약성 결과의 심각도로 분류되지 않음이 할당될 수도 있습니다. 이는 공급업체가 탐지된 취약성에 대해 취약성 점수를 아직 설정하지 않은 경우입니다. 이 경우 결과에 URLs 대한 참조를 사용하여 해당 취약성을 조사하고 그에 따라 대응하는 것이 좋습니다.

패키지 취약성 결과에는 다음과 같은 점수와 관련 점수 벡터가 결과 세부 정보의 일부로 포함됩니다.

- EPSS 점수
- Inspector 점수
- CVSS Amazon의 3.1 CVE
- CVSS 3.1 시작 NVD
- CVSS 2.0 시작NVD(해당하는 경우)

코드 취약성 심각도

코드 취약성 조사 결과의 경우 Amazon Inspector는 조사 결과를 생성한 Amazon CodeGuru 탐지기가 정의한 심각도 수준을 사용합니다. 각 탐지기에는 CVSS v3 채점 시스템을 사용하여 심각도가 할당됩니다. 심각도 CodeGuru 사용에 대한 설명은 CodeGuru 가이드의 심각도 정의를 참조하세요. 심각도별 탐지기 목록을 보려면 아래의 지원되는 프로그래밍 언어 중에서 선택하세요.

- 심각도별 Python 탐지기
- 심각도별 Java 탐지기

네트워크 연결성 심각도

Amazon Inspector는 노출된 서비스, 포트 및 프로토콜과 개방 경로 유형을 기반으로 네트워크 연결성 취약성의 심각도를 결정합니다. 다음 표에는 이러한 심각도 등급이 정의되어 있습니다. 오픈 경로 등급 열의 값은 가상 게이트웨이, 피어링된 VPCs및 AWS Direct Connect 네트워크의 오픈 경로를 나타냅니다. 기타 모든 노출된 서비스, 포트 및 프로토콜에는 정보 심각도 등급이 할당됩니다.

서비스	TCP 포트	UDP 포트	인터넷 경로 등급	개방 경로 등급
DHCP	67, 68, 546, 547	67, 68, 546, 547	중간	정보
Elasticsearch	9300, 9200	NA	중간	정보
FTP	21	21	높음	중간
글로벌 카탈로그 LDAP	3268	NA	중간	정보
LDAP 의 글로벌 카탈로그 TLS	3269	NA	중간	정보
HTTP	80	80	낮음	정보
HTTPS	443	443	낮음	정보
Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752	중간	정보
LDAP	389	389	중간	정보

LDAP 초과 TLS	636	NA	중간	정보
MongoDB	27017, 27018, 27019, 28017	NA	중간	정보
내SQL	3306	NA	중간	정보
순BIOS	137, 139	137, 138	중간	정보
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110	중간	정보
Oracle	1521, 1630	NA	중간	정보
PostgreSQL	5432	NA	중간	정보
인쇄 서비스	515	NA	높음	중간
RDP	3389	3389	중간	낮음
RPC	111, 135, 530	111, 135, 530	중간	정보
SMB	445	445	중간	정보
SSH	22	22	중간	낮음
SQL 서버	1433	1434	중간	정보
Syslog	601	514	중간	정보
Telnet	23	23	높음	중간
WINS	1512, 42	1512, 42	중간	정보

Amazon Inspector에서 결과 관리

Amazon Inspector를 사용하면 다양한 방식으로 결과를 관리할 수 있습니다. 상태를 기준으로 결과를 필터링할 수 있습니다. 필터 기준에 따라 결과를 검색할 수 있습니다. 검색 결과 목록에서 검색 결과를 제외하는 억제 규칙을 만들 수 있습니다. 또한 AWS Security Hub EventBridge, Amazon 및 아마존 심플 스토리지 서비스 (Amazon S3) 로 조사 결과를 내보낼 수 있습니다.

주제

- Amazon Inspector 조사 결과 필터링
- Amazon Inspector 조사 결과 숨김
- 아마존 인스펙터 조사 결과 보고서 내보내기
- Amazon을 사용하여 Amazon Inspector의 조사 결과에 대한 사용자 지정 응답 생성 EventBridge

Amazon Inspector 조사 결과 필터링

필터 기준을 사용하여 Amazon Inspector 결과를 필터링할 수 있습니다. 검색 결과가 필터 기준과 일치하지 않는 경우 Amazon Inspector는 해당 결과를 보기에서 제외합니다. 이 섹션에서는 필터 기준을 사용하여 Amazon Inspector 결과를 필터링하는 방법을 설명합니다.

Amazon Inspector 콘솔에서 필터 생성

각 결과 보기에서 필터 기능을 사용하여 특정한 특성을 가진 결과를 찾을 수 있습니다. 다른 탭 보기로 이동하면 필터는 제거됩니다.

필터는 필터 기준으로 구성되고, 필터 기준은 필터 값과 쌍을 이루는 하나의 필터 속성으로 구성됩니다. 필터 기준과 일치하지 않는 결과는 결과 목록에서 제외됩니다. 예를 들어 관리자 계정과 관련된 모든 결과를 보려면 계정 ID 속성을 선택하고 이를 12자리 AWS AWS 계정 ID의 값과 연결하면 됩니다.

모든 결과에 적용되는 필터 기준이 있는 반면, 특정 리소스 유형이나 결과 유형에만 적용되는 필터 기준도 있습니다.

결과 보기에 필터를 적용하려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창에서 결과를 선택합니다. 기본 보기에는 활성 상태인 모든 결과가 표시됩니다.

3. 결과를 기준별로 필터링하려면 필터 추가 막대를 선택하여 해당 보기에 적용할 수 있는 모든 필터 기준 목록을 표시합니다. 보기마다 다른 필터 기준을 사용할 수 있습니다.
4. 목록에서 필터 기준을 선택합니다.
5. 기준 입력 창에 원하는 필터 값을 입력하여 해당 기준을 정의합니다.
6. 적용을 선택하여 해당 필터 기준을 현재 결과에 적용합니다. 필터 입력 막대를 다시 선택하여 다른 필터 기준을 계속 추가할 수 있습니다.
7. (선택 사항) 표시되지 않은 결과나 종결된 결과를 보려면 필터 막대에서 활성을 선택한 다음 표시되지 않음 또는 종결됨을 선택합니다. 활성 결과, 표시되지 않은 결과 및 종결된 결과를 동일한 보기에서 보려면 모두 보기를 선택합니다.

Amazon Inspector 조사 결과 숨김

억제 규칙을 만들어 기준과 일치하는 결과를 숨길 수 있습니다. 예를 들어, 심각도 등급을 기준으로 검색 결과를 숨기는 금지 규칙을 만들 수 있습니다. Amazon Inspector에서 금지 규칙과 일치하는 검색 결과를 생성하면 Amazon Inspector는 검색 결과를 표시하지 않고 보이지 않게 숨깁니다. Amazon Inspector는 숨겨진 검색 결과를 수정될 때까지 저장합니다. 숨겨진 검색 결과가 수정되면 Amazon Inspector는 해당 검색 결과를 종료합니다. 콘솔에서 숨겨진 결과를 볼 수 있습니다.

억제 규칙을 만들어 가장 중요한 발견의 우선 순위를 정합니다. 억제 규칙은 검색 결과를 보이지 않게 숨길 뿐이므로 검색 결과에 영향을 주지 않습니다. 검색 결과를 닫거나 수정하는 억제 규칙을 만들 수는 없습니다. Amazon EventBridge 규칙을 사용하여 원치 않는 결과를 억제할 수도 있습니다. AWS Security Hub 이 섹션의 절차는 금지 규칙을 만들고, 보고, 편집하고, 삭제하는 방법을 설명합니다.

> **Note**
> 조직의 위임된 관리자만 금지 규칙을 만들고 관리할 수 있습니다.

억제 규칙 생성

억제 규칙을 생성하여 기본적으로 표시되는 결과 목록을 필터링할 수 있습니다. 를 사용하고 의 `SUPRESS` 값으로 CreateFilterAPI지정하여 프로그래밍 방식으로 금지 규칙을 만들 수 있습니다. `action`

> **Note**
>
> 독립형 계정과 Amazon Inspector 위임 관리자만 억제 규칙을 생성하고 관리할 수 있습니다. 조직의 멤버는 탐색 창에서 억제 규칙 옵션을 볼 수 없습니다.

억제 규칙을 생성하려면(콘솔)

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창에서 억제 규칙을 선택합니다. 그런 다음 규칙 생성을 선택합니다.
3. 각 기준에 대해 다음을 수행합니다.
   - 필터 막대를 선택하여 억제 규칙에 추가할 수 있는 필터 기준 목록을 표시합니다.
   - 억제 규칙의 필터 기준을 선택합니다.
4. 기준 추가를 마쳤으면 규칙 이름과 설명(선택 사항)을 입력합니다.
5. 규칙 저장을 선택합니다. Amazon Inspector에서 새 억제 규칙을 즉시 적용하고 기준과 일치하는 결과는 숨깁니다.

숨겨진 결과 보기

기본적으로 Amazon Inspector는 숨겨진 결과를 Amazon Inspector 콘솔에 표시하지 않습니다. 하지만 특정 규칙에 따라 숨겨진 결과를 볼 수는 있습니다.

숨겨진 결과를 보려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창에서 억제 규칙을 선택합니다.
3. 억제 규칙 목록에서 규칙 제목을 선택합니다.

억제 규칙 변경

억제 규칙은 언제든지 변경할 수 있습니다.

억제 규칙을 수정하려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창에서 억제 규칙을 선택합니다.
3. 수정할 억제 규칙의 제목을 선택합니다.
4. 원하는 대로 변경한 다음 저장을 선택하여 규칙을 업데이트합니다.

억제 규칙 삭제

억제 규칙을 삭제할 수 있습니다. 억제 규칙을 삭제하면 Amazon Inspector에서 규칙 기준을 충족하고 다른 규칙에 의해 차단되지 않는 새로운 결과와 기존 결과의 억제가 중단됩니다.

억제 규칙을 삭제하면 해당 규칙의 기준을 충족하는 새로운 결과와 기존 결과가 활성 상태가 됩니다. 즉, Amazon Inspector 콘솔에 기본적으로 표시됩니다. 또한 Amazon Inspector는 이러한 결과를 AWS Security Hub 및 EventBridge Amazon에 이벤트로 게시합니다.

억제 규칙을 삭제하려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창에서 억제 규칙을 선택합니다.
3. 삭제하려는 억제 규칙 제목 옆의 확인란을 선택합니다.
4. 삭제를 선택한 다음 선택을 확인하여 규칙을 영구 삭제합니다.

아마존 인스펙터 조사 결과 보고서 내보내기

조사 결과 보고서는 조사 결과에 대한 상세한 스냅샷을 제공하는 CSV 또는 JSON 파일입니다. 결과 보고서를 아마존 EventBridge 및 아마존 심플 스토리지 서비스 (Amazon S3) 로 AWS Security Hub내보낼 수 있습니다. 결과 보고서를 구성할 때 보고서에 포함할 결과를 지정합니다. 기본적으로 검색 결과 보고서에는 모든 활성 검색 결과에 대한 데이터가 포함됩니다. 조직의 위임 관리자인 경우 조사 결과 보고서에는 조직의 모든 구성원 계정에 대한 데이터가 포함됩니다. 결과 보고서를 사용자 지정하려면 필터를 만들어 적용하세요.

조사 결과 보고서를 내보내는 경우 Amazon Inspector는 검색 결과 데이터를 지정된 데이터로 암호화합니다. AWS KMS key Amazon Inspector는 검색 결과 데이터를 암호화한 후 사용자가 지정한 Amazon S3 버킷에 검색 결과 보고서를 저장합니다. AWS KMS 키는 Amazon S3 AWS 리전 버킷과 동

일하게 사용해야 합니다. AWS KMS 키 정책은 Amazon Inspector가 이를 사용할 수 있도록 허용해야 하며, Amazon S3 버킷 정책은 Amazon Inspector가 객체를 추가할 수 있도록 허용해야 합니다. 결과 보고서를 내보낸 후 Amazon S3 버킷에서 다운로드하거나 새 위치로 전송할 수 있습니다. Amazon S3 버킷을 내보낸 다른 결과 보고서의 리포지토리로 사용할 수도 있습니다.

이 섹션에서는 Amazon Inspector 콘솔에서 결과 보고서를 내보내는 방법을 설명합니다. 다음 작업을 수행하려면 권한을 확인하고, Amazon S3 버킷을 구성하고, 구성하고 AWS KMS key, 결과 보고서를 구성하고 내보내야 합니다.

> **Note**
>
> Amazon CreateFindingsReportAPIInspector를 사용하여 조사 결과 보고서를 내보내는 경우 활성 조사 결과만 볼 수 있습니다. 숨겨지거나 종료된 결과를 보려면 필터 기준의 `CLOSED` 일부로 `SUPPRESSED` 또는 를 지정해야 합니다.

Tasks

- 1단계: 권한 확인
- 2단계: S3 버킷 구성
- 3단계: AWS KMS key구성
- 4단계: 결과 보고서 구성 및 내보내기
- 내보내기 오류 해결

1단계: 권한 확인

> **Note**
>
> 검색 결과 보고서를 처음으로 내보낸 후 1~3단계는 선택 사항입니다. 다음 단계는 동일한 Amazon S3 버킷을 사용할지 여부와 다른 내보낸 결과 보고서를 사용할지 여부에 따라 결정됩니다. AWS KMS key 1~3단계를 완료한 후 프로그래밍 방식으로 결과 보고서를 내보내려면 Amazon Inspector의 CreateFindingsReport작업을 사용하십시오. API

Amazon Inspector에서 결과 보고서를 내보내기 전에 결과 보고서를 내보내고 보고서를 암호화 및 저장하기 위한 리소스를 구성하는 데 필요한 권한이 있는지 확인합니다. 권한을 확인하려면 AWS Identity and Access Management (IAM) 를 사용하여 자격 증명에 연결된 IAM 정책을 검토하십시오.

IAM 그런 다음 해당 정책의 정보를 다음 작업 목록과 비교하여 결과 보고서 내보내기를 위해 사용자가 수행할 수 있어야 하는 작업을 확인합니다.

Amazon Inspector

Amazon Inspector의 경우 다음 작업을 수행할 수 있는지 확인합니다.

- `inspector2:ListFindings`
- `inspector2:CreateFindingsReport`

이러한 작업을 통해 계정의 결과 데이터를 검색하고 해당 데이터를 결과 보고서로 내보낼 수 있습니다.

대용량 보고서를 프로그래밍 방식으로 내보내려는 경우 `inspector2:GetFindingsReportStatus`(보고서 상태 확인) 및 `inspector2:CancelFindingsReport`(진행 중인 내보내기 취소) 작업을 수행할 수 있는 권한이 있는지도 확인할 수 있습니다.

AWS KMS

의 경우 AWS KMS, 다음 작업을 수행할 수 있는지 확인하십시오.

- `kms:GetKeyPolicy`
- `kms:PutKeyPolicy`

이러한 작업을 통해 Amazon Inspector에서 보고서를 암호화하는 데 사용할 AWS KMS key 에 대한 키 정책을 검색하고 업데이트할 수 있습니다.

Amazon Inspector 콘솔을 사용하여 보고서를 내보내려면 다음 AWS KMS 작업을 수행할 수 있는지도 확인하십시오.

- `kms:DescribeKey`
- `kms:ListAliases`

이러한 작업을 통해 계정의 AWS KMS keys 에 대한 정보를 검색하고 표시할 수 있습니다. 그런 다음 이러한 키 중 하나를 선택하여 보고서를 암호화할 수 있습니다.

보고서 암호화를 위한 새 KMS 키를 생성하려는 경우에도 `kms:CreateKey` 작업 수행을 허용해야 합니다.

Amazon S3

Amazon S3의 경우 다음 작업을 수행할 수 있는지 확인합니다.

- `s3:CreateBucket`
- `s3:DeleteObject`
- `s3:PutBucketAcl`
- `s3:PutBucketPolicy`
- `s3:PutBucketPublicAccessBlock`
- `s3:PutObject`
- `s3:PutObjectAcl`

이러한 작업을 통해 Amazon Inspector에서 보고서를 저장할 S3 버킷을 생성하고 구성할 수 있습니다. 또한 버킷에서 객체를 추가하고 삭제할 수도 있습니다.

Amazon Inspector 콘솔을 사용하여 보고서를 내보내려는 경우 `s3:ListAllMyBuckets` 및 `s3:GetBucketLocation` 작업을 수행할 수 있는지도 확인합니다. 이러한 작업을 통해 계정의 S3 버킷에 대한 정보를 검색하고 표시할 수 있습니다. 그런 다음 이러한 버킷 중 하나를 선택하여 보고서를 저장할 수 있습니다.

필요한 작업을 하나 이상 수행할 수 없는 경우 다음 단계로 진행하기 전에 AWS 관리자에게 도움을 요청하세요.

2단계: S3 버킷 구성

권한을 확인했으면 결과 보고서를 저장할 S3 버킷을 구성할 준비가 된 것입니다. 자신의 계정을 위한 기존 버킷일 수도 있고, 다른 사람이 소유하고 AWS 계정 있고 액세스가 허용된 기존 버킷일 수도 있습니다. 보고서를 새 버킷에 저장하려면 진행하기 전에 버킷을 생성합니다.

S3 버킷은 내보내려는 결과 데이터와 AWS 리전 같아야 합니다. 예를 들어, Amazon Inspector를 미국 동부(버지니아 북부) 리전에서 사용 중이고 해당 리전에 대한 결과 데이터를 내보내려면 버킷도 미국 동부(버지니아 북부) 리전에 있어야 합니다.

또한 버킷 정책에서 Amazon Inspector가 버킷에 객체를 추가할 수 있도록 허용해야 합니다. 이 주제에서는 버킷 정책을 업데이트하는 방법을 설명하고 정책에 추가할 명령문의 예를 제공합니다. 버킷 정책 추가 및 업데이트에 대한 자세한 내용은 Amazon Simple Storage Service 사용 설명서에서 버킷 정책 사용을 참조하세요.

다른 계정이 소유하고 있는 S3 버킷에 보고서를 저장하려면 버킷 소유자와 협력하여 버킷 정책을 업데이트합니다. 또한 해당 URI 버킷에 사용할 것도 구하십시오. 보고서를 내보낼 URI 때 이를 입력해야 합니다.

버킷 정책을 업데이트하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/s3 에서 Amazon S3 콘솔을 엽니다.
2. 탐색 창에서 버킷을 선택합니다.
3. 결과 보고서를 저장할 S3 버킷을 선택합니다.
4. 권한 탭을 선택합니다.
5. 버킷 정책 섹션에서 편집을 선택합니다.
6. 다음 예제 명령문을 클립보드로 복사합니다.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "allow-inspector",
   "Effect": "Allow",
   "Principal": {
    "Service": "inspector2.amazonaws.com"
   },
   "Action": [
    "s3:PutObject",
    "s3:PutObjectAcl",
    "s3:AbortMultipartUpload"
   ],
   "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
   "Condition": {
    "StringEquals": {
     "aws:SourceAccount": "111122223333"
    },
    "ArnLike": {
     "aws:SourceArn": "arn:aws:inspector2:Region:111122223333:report/*"
    }
   }
  }
 ]
}
```

7. Amazon S3 콘솔의 버킷 정책 편집기에서 위의 명령문을 정책에 붙여넣어 정책에 추가합니다.

명령문을 추가할 때 구문이 올바른지 확인합니다. 버킷 정책은 JSON 형식을 사용합니다. 즉, 정책에 명령문을 추가하는 위치에 따라 명령문 앞이나 뒤에 쉼표를 추가해야 합니다. 명령문을 마지막 명령문으로 추가하는 경우 위 명령문의 닫는 괄호 뒤에 쉼표를 추가합니다. 명령문을 첫 번째 명령문으로 추가하거나 기존 두 명령문 사이에 추가하는 경우 명령문의 닫는 괄호 뒤에 쉼표를 추가합니다.

8. 사용 환경에 적합한 값으로 명령문을 업데이트합니다.

   - *DOC-EXAMPLE-BUCKET* 버킷 이름입니다.
   - *111122223333* 는 사용자의 계정 AWS 계정 ID입니다.
   - *Region* Amazon Inspector를 사용하고 있으며 Amazon Inspector가 버킷에 보고서를 추가할 수 있도록 허용하려는 곳입니다. AWS 리전 예를 들어 미국 동부(버지니아 북부) 리전의 경우 `us-east-1`입니다.

> **Note**
>
> 수동으로 활성화된 상태에서 AWS 리전 Amazon Inspector를 사용하는 경우 필드 값에 적절한 지역 코드도 추가하십시오. `Service` 이 필드는 Amazon Inspector 서비스 주체를 지정합니다.
> 예를 들어 리전 코드가 `me-south-1`인 중동(바레인) 리전에서 Amazon Inspector를 사용하는 경우, 명령문에서 `inspector2.amazonaws.com`을 `inspector2.me-south-1.amazonaws.com`으로 바꿉니다.

참고로, 예제 명령문은 두 개의 IAM 글로벌 조건 키를 사용하는 조건을 정의합니다.

- aws: SourceAccount — 이 조건을 사용하면 Amazon Inspector에서 사용자 계정에 대한 보고서만 버킷에 추가할 수 있습니다. Amazon Inspector가 다른 계정에 대해 버킷에 보고서를 추가하지 못하도록 합니다. 더 구체적으로, 이 조건은 `aws:SourceArn` 조건에 지정된 리소스 및 작업에 대해 버킷을 사용할 수 있는 계정을 지정합니다.

  버킷의 추가 계정에 대한 보고서를 저장하려면 각 추가 계정의 계정 ID를 이 조건에 추가합니다. 예:

  ```
  "aws:SourceAccount": [111122223333,444455556666,123456789012]
  ```

- aws: SourceArn — 이 조건은 버킷에 추가되는 객체의 소스에 따라 버킷에 대한 액세스를 제한합니다. 다른 AWS 서비스 사람이 버킷에 객체를 추가하는 것을 방지합니다. 또한 사용자 계정에 대해 다른 작업을 수행하는 동안 Amazon Inspector가 버킷에 객체를 추가하지 못하도록 합니다. 더 구체적으로, 이 조건은 객체가 결과 보고서인 경우에만, 그리고 해당 보고서가 조건에 지정된 계정과 리전에서 생성된 경우에만 Amazon Inspector가 버킷에 객체를 추가할 수 있도록 허용합니다.

  Amazon Inspector가 추가 계정에 대해 지정된 작업을 수행할 수 있도록 하려면 이 조건에 각 추가 계정의 Amazon 리소스 이름 (ARNs) 을 추가하십시오. 예:

  ```
  "aws:SourceArn": [
      "arn:aws:inspector2:Region:111122223333:report/*",
      "arn:aws:inspector2:Region:444455556666:report/*",
      "arn:aws:inspector2:Region:123456789012:report/*"
  ]
  ```

  `aws:SourceAccount` 및 `aws:SourceArn` 조건에 지정된 계정이 일치해야 합니다.

두 조건 모두 Amazon S3와의 트랜잭션 중에 Amazon Inspector가 혼동되는 대리자로 사용되는 것을 방지하는 데 도움이 됩니다. 권장하지는 않지만 버킷 정책에서 이러한 조건을 제거할 수 있습니다.

9. 버킷 정책 업데이트가 완료되면 변경 사항 저장을 선택합니다.

3단계: AWS KMS key구성

권한을 확인하고 S3 버킷을 구성한 후에는 Amazon Inspector에서 결과 보고서를 암호화하는 데 사용할 AWS KMS key 를 결정해야 합니다. 키는 고객이 관리하는 대칭 암호화 KMS 키여야 합니다. 또한 키는 보고서를 저장하도록 구성한 S3 AWS 리전 버킷과 동일해야 합니다.

키는 사용자 계정의 기존 KMS 키이거나 다른 계정이 소유한 기존 KMS 키일 수 있습니다. 새 KMS 키를 사용하려면 진행하기 전에 키를 생성해야 합니다. 다른 계정이 소유한 기존 키를 사용하려면 해당 키의 Amazon 리소스 이름 (ARN) 을 확보하십시오. Amazon Inspector에서 보고서를 내보낼 ARN 때 이 정보를 입력해야 합니다. KMS키 설정 생성 및 검토에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 키 관리를 참조하십시오.

사용할 KMS 키를 결정한 후 Amazon Inspector에 키 사용 권한을 부여하십시오. 그렇지 않으면 Amazon Inspector에서 보고서를 암호화하고 내보낼 수 없습니다. Amazon Inspector에 키 사용 권한

을 부여하려면 키에 대한 키 정책을 업데이트합니다. 키 정책 및 키 액세스 관리에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 키 정책을 참조하십시오. KMS AWS KMS

> **Note**
>
> 다음 절차는 Amazon Inspector에서 사용할 수 있도록 기존 키를 업데이트하는 절차입니다. 기존 키가 없는 경우 AWS Key Management Service 개발자 안내서에서 키 만들기를 참조하십시오.

키 정책을 업데이트하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/kms 에서 AWS KMS 콘솔을 엽니다.
2. 탐색 창에서 고객 관리형 키를 선택합니다.
3. 보고서를 암호화하는 데 사용할 KMS 키를 선택합니다. 키는 대칭 암호화 (SYMMETRIC_DEFAULT) 키여야 합니다.
4. 키 정책 탭에서 편집을 선택합니다. 키 정책에서 편집 버튼이 보이지 않으면 먼저 정책 보기로 전환을 선택해야 합니다.
5. 다음 예제 명령문을 클립보드로 복사합니다.

```
{
    "Sid": "Allow Amazon Inspector to use the key",
    "Effect": "Allow",
    "Principal": {
        "Service": "inspector2.amazonaws.com"
    },
    "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey*"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "111122223333"
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:inspector2:Region:111122223333:report/*"
        }
    }
```

```
}
```

6. AWS KMS 콘솔의 키 정책 편집기에서 위의 설명을 키 정책에 붙여넣어 정책에 추가합니다.

   명령문을 추가할 때 구문이 올바른지 확인합니다. 키 정책은 JSON 형식을 사용합니다. 즉, 정책에 명령문을 추가하는 위치에 따라 명령문 앞이나 뒤에 쉼표를 추가해야 합니다. 명령문을 마지막 명령문으로 추가하는 경우 위 명령문의 닫는 괄호 뒤에 쉼표를 추가합니다. 명령문을 첫 번째 명령문으로 추가하거나 기존 두 명령문 사이에 추가하는 경우 명령문의 닫는 괄호 뒤에 쉼표를 추가합니다.

7. 사용 환경에 적합한 값으로 명령문을 업데이트합니다.

   - *111122223333* 는 귀하의 계정 AWS 계정 ID입니다.
   - *Region* Amazon Inspector가 키를 사용하여 보고서를 암호화하도록 허용하려는 곳입니다. AWS 리전 예를 들어 미국 동부(버지니아 북부) 리전의 경우 `us-east-1`입니다.

   > **Note**
   > 수동으로 활성화된 상태에서 AWS 리전 Amazon Inspector를 사용하는 경우 필드 값에 적절한 지역 코드도 추가하십시오. `Service` 예를 들어 중동(바레인) 리전에서 Amazon Inspector를 사용할 경우 `inspector2.amazonaws.com`을 `inspector2.me-south-1.amazonaws.com`으로 바꿉니다.

   이전 단계의 버킷 정책 예제 명령문과 마찬가지로 이 예제의 `Condition` 필드는 두 개의 IAM 글로벌 조건 키를 사용합니다.

   - aws: SourceAccount — 이 조건은 Amazon Inspector가 사용자 계정에 대해서만 지정된 작업을 수행할 수 있도록 허용합니다. 더 구체적으로, 이 조건은 `aws:SourceArn` 조건에 지정된 리소스 및 작업에 대해 지정된 작업을 수행할 수 있는 계정을 결정합니다.

     Amazon Inspector가 추가 계정에 대해 지정된 작업을 수행할 수 있도록 하려면 이 조건에 각 추가 계정의 계정 ID를 추가합니다. 예:

     ```
     "aws:SourceAccount": [111122223333,444455556666,123456789012]
     ```

   - aws: SourceArn — 이 조건은 다른 AWS 서비스 사람이 지정된 작업을 수행하는 것을 방지합니다. 또한 사용자 계정에 대해 다른 작업을 수행하는 동안 Amazon Inspector가 키를 사용하지 못하도록 합니다. 즉, 객체가 결과 보고서인 경우에만, 그리고 해당 보고서가 조건에 지정된 계정

과 리전에서 생성된 경우에만 Amazon Inspector가 해당 키로 S3 객체를 암호화할 수 있도록 허용합니다.

Amazon Inspector가 추가 계정에 대해 지정된 작업을 수행할 수 있도록 하려면 각 추가 계정을 이 조건에 추가하십시오ARNs. 예:

```
"aws:SourceArn": [
    "arn:aws:inspector2:us-east-1:111122223333:report/*",
    "arn:aws:inspector2:us-east-1:444455556666:report/*",
    "arn:aws:inspector2:us-east-1:123456789012:report/*"
]
```

`aws:SourceAccount` 및 `aws:SourceArn` 조건에 지정된 계정이 일치해야 합니다.

이러한 조건은 Amazon Inspector가 거래 중에 혼동을 일으키는 대리인으로 사용되는 것을 방지하는 데 도움이 됩니다. AWS KMS권장하지는 않지만 명령문에서 이러한 조건을 제거할 수 있습니다.

8. 키 정책 업데이트가 완료되면 변경 사항 저장을 선택합니다.

4단계: 결과 보고서 구성 및 내보내기

> **Note**
> 조사 결과 보고서는 한 번에 하나만 내보낼 수 있습니다. 현재 내보내기가 진행 중인 경우 내보내기가 완료될 때까지 기다려야 다른 검색 결과 보고서를 내보낼 수 있습니다.

권한을 확인하고 결과 보고서를 암호화하고 저장하도록 리소스를 구성했으면 보고서를 구성하고 내보낼 준비가 된 것입니다.

결과 보고서를 구성하고 내보내려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창의 결과에서 모든 결과를 선택합니다.

3. (선택 사항) 결과 테이블 위의 필터 막대를 사용하여 보고서에 포함할 결과를 지정하는 필터 기준을 추가합니다. 기준을 추가하면 Amazon Inspector에서 기준과 일치하는 결과만 포함하도록 테이블을 업데이트합니다. 이 테이블은 보고서에 포함될 데이터의 미리 보기를 제공합니다.

> **Note**
>
> 필터 기준을 추가하는 것이 좋습니다. 그렇지 않으면 보고서에는 활성 상태인 현재 AWS 리전 조사 결과의 모든 데이터가 포함됩니다. 조직의 Amazon Inspector 관리자인 경우 여기에 조직 내 모든 멤버 계정에 대한 결과 데이터가 포함됩니다.
> 보고서에 전체 또는 여러 결과의 데이터가 포함되어 있는 경우 보고서를 생성하고 내보내는 데 시간이 오래 걸릴 수 있으며, 한 번에 하나의 보고서만 내보낼 수 있습니다.

4. 결과 내보내기를 선택합니다.
5. 내보내기 설정 섹션의 내보내기 파일 유형에서 보고서의 파일 형식을 지정합니다.
   - 데이터가 포함된 JavaScript 개체 표기법 (.json) 파일을 만들려면 을 선택합니다. JSON

     JSON옵션을 선택하면 보고서에는 각 결과에 대한 모든 필드가 포함됩니다. 가능한 JSON 필드 목록은 Amazon API Inspector의 데이터 유형 찾기 참조를 참조하십시오.
   - 데이터를 포함하는 쉼표로 구분된 값 (.csv) 파일을 생성하려면 을 선택합니다. CSV

     CSV옵션을 선택하면 보고서에는 각 검색 결과에 대한 필드의 하위 집합, 즉 검색 결과의 주요 속성을 보고하는 약 45개의 필드만 포함됩니다. 이 필드에는 결과 유형, 제목, 심각도, 상태, 설명, 처음 발견 날짜, 최종 발견 날짜, 수정 버전 있음, AWS 계정 ID, 리소스 ID, 리소스 태그, 해결 등이 있습니다. 여기에는 점수 세부 정보 및 각 결과에 URLs 대한 참조를 캡처하는 필드 외에도 포함됩니다. 다음은 조사 결과 보고서의 CSV 헤더 샘플입니다.

AWS유Title설찾처마최리컨리플리영패버패파네연이인인상취공공공공NVDVDCVSSSS2리리리리리포익마람Lambda
계각정형 명기음지근소테전랫소향키전키일트링제스스태약급급급급점벡점벡급급급급소미소소소소트스지다패최조
정도가찾 ARN막업스이 폼스을지에지경위(옵Security상업업업업수터수터업업업업스 스스스스범플막레키종URL
ID 능기 으데IDL너 태받설서개로크 Hub터시ID체체체체 체체체체유 퍼프IpV6로악이지업
로이 이 그는치수선 경 가스스기 심자권 CVSS점라 잇용어유데
본트 미 패버정 로 와코코 각문고 점벡점벡 릭이 사행 형이
지 키전됨 통어어 도 게 수터수터 IPv4 용위 트
태 지 합 벡 시 IPv4 가 날
그 되 터 능 짜
었

습니다

6. 내보내기 위치에서 S3의 URI 경우 보고서를 저장할 S3 버킷을 지정합니다.

   - 계정이 소유한 버킷에 보고서를 저장하려면 S3 찾아보기를 선택합니다. Amazon Inspector에서 계정의 S3 버킷 테이블을 표시합니다. 원하는 버킷의 행을 선택한 다음 선택을 선택합니다.

     > **Tip**
     >
     > 보고서에 Amazon S3 경로 접두사도 지정하려면 S3 상자의 값에 슬래시 (/) 와 접두사를 추가합니다. URI 그러면 Amazon Inspector가 보고서를 버킷에 추가할 때 해당 접두사가 포함되고, Amazon S3는 접두사로 지정된 경로를 생성합니다.
     > 예를 들어 AWS 계정 ID를 접두사로 사용하고 계정 ID가 111122223333인 경우 S3 상자의 값에 추가하십시오. **`/111122223333`** URI
     > 접두사는 S3 버킷 내에서 디렉터리 경로와 비슷합니다. 유사한 파일을 파일 시스템의 폴더에 함께 저장하는 것처럼 유사한 객체를 버킷에 그룹화할 수 있습니다. 자세한 내용은 Amazon Simple Storage Service 사용 설명서에서 Amazon S3 콘솔에서 폴더를 사용하여 객체 구성을 참조하세요.

   - 다른 계정이 소유한 버킷에 보고서를 저장하려면 해당 버킷에 를 입력합니다. URI 예를 들어, 여기서 - _는 버킷 이름입니다. ***`s3://DOC-EXAMPLE_BUCKET`*** DOC EXAMPLE BUCKET 버킷 소유자가 버킷 속성에서 이 정보를 찾을 수 있습니다.

7. KMS키에는 보고서를 AWS KMS key 암호화하는 데 사용할 이름을 지정합니다.

   - 내 계정의 키를 사용하려면 목록에서 키를 선택합니다. 목록에는 계정에 대한 고객 관리형 대칭 암호화 KMS 키가 표시됩니다.
   - 다른 계정이 소유한 키를 사용하려면 키의 Amazon 리소스 이름 (ARN) 을 입력합니다. 키 소유자가 키 속성에서 이 정보를 찾을 수 있습니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 키 ID 및 키 ARN 찾기를 참조하십시오.

8. 내보내기를 선택합니다.

Amazon Inspector는 검색 결과 보고서를 생성하고, 지정된 KMS 키로 암호화하고, 지정한 S3 버킷에 추가합니다. 보고서에 포함되도록 선택한 결과 수에 따라 이 프로세스는 몇 분 또는 몇 시간이 걸릴 수 있습니다. 내보내기가 완료되면 Amazon Inspector에서 결과 보고서를 성공적으로 내보냈다는 메시지를 표시합니다. 선택적으로 메시지에서 보고서 보기를 선택하여 Amazon S3의 보고서로 이동합니다.

보고서는 한 번에 하나만 내보낼 수 있습니다. 내보내기가 현재 진행 중이라면 내보내기가 완료될 때까지 기다린 후 다른 보고서를 내보냅니다.

내보내기 오류 해결

결과 보고서를 내보내려고 할 때 오류가 발생하면 Amazon Inspector에서 오류를 설명하는 메시지를 표시합니다. 이 주제에 설명된 정보를 참고하여 오류의 가능한 원인과 해결 방법을 파악할 수 있습니다.

예를 들어, S3 버킷이 최신 AWS 리전 상태이고 버킷 정책이 Amazon Inspector에서 버킷에 객체를 추가할 수 있도록 허용하는지 확인합니다. 또한 현재 지역에서 AWS KMS key 가 활성화되어 있는지 확인하고 키 정책이 Amazon Inspector에서 키를 사용하도록 허용하는지 확인하십시오.

오류를 해결한 후 보고서를 다시 내보냅니다.

보고서가 여러 개일 수 없음 오류

보고서를 생성하려고 하는데 Amazon Inspector에서 이미 보고서를 생성하고 있는 경우 원인: 진행 중인 보고서가 여러 개일 수 없음이라는 오류 메시지가 나타납니다. Amazon Inspector에서는 계정에 대해 한 번에 하나의 보고서만 생성할 수 있기 때문에 이 오류가 발생합니다.

이 오류를 해결하려면 다른 보고서가 완료될 때까지 기다리거나 새 보고서를 요청하기 전에 보고서를 취소하면 됩니다.

GetFindingsReportStatus작업을 사용하여 보고서 상태를 확인할 수 있습니다. 이 작업은 현재 생성 중인 보고서의 보고서 ID를 반환합니다.

필요한 경우 작업에서 제공한 보고서 ID를 사용하여 `GetFindingsReportStatus` 작업을 사용하여 현재 진행 중인 내보내기를 취소할 수 있습니다. CancelFindingsReport

Amazon을 사용하여 Amazon Inspector의 조사 결과에 대한 사용자 지정 응답 생성 EventBridge

Amazon Inspector는 새로 생성된 검색 결과 및 집계된 결과에 EventBridge 대해 Amazon에서 이벤트를 생성합니다. Amazon Inspector는 검색 결과 상태의 변경 사항에 대한 이벤트도 생성합니다. 즉, Amazon Inspector는 사용자가 리소스를 다시 시작하거나 리소스와 관련된 태그를 변경하는 등의 작업을 수행할 때 검색 결과에 대한 새 이벤트를 생성합니다. Amazon Inspector에서 업데이트된 결과에 대해 새 이벤트를 생성해도 검색 결과는 `id` 동일하게 유지됩니다.

> **Note**
>
> Amazon Inspector의 위임 관리자 계정인 경우 이벤트가 발생한 사용자 계정 및 멤버 계정에 이벤트를 EventBridge 게시합니다.

Amazon Inspector에서 EventBridge 이벤트를 사용하면 작업을 자동화하여 조사 결과로 드러난 보안 문제에 대응하는 데 도움이 될 수 있습니다. EventBridge 이벤트를 기반으로 Amazon Inspector 조사 결과에 대한 알림을 받으려면 EventBridge 규칙을 생성하고 Amazon Inspector의 대상을 지정해야 합니다. EventBridge 규칙은 Amazon Inspector 조사 결과에 대한 알림을 보낼 수 EventBridge 있도록 허용하며, 대상은 알림을 보낼 위치를 지정합니다.

Amazon Inspector는 현재 Amazon Inspector를 사용하고 AWS 리전 있는 곳의 기본 이벤트 버스로 이벤트를 내보냅니다. 즉, Amazon Inspector를 활성화하고 이벤트를 수신하도록 Amazon Inspector를 구성한 각 AWS 리전 위치에 대해 이벤트 규칙을 구성해야 합니다. EventBridge Amazon Inspector는 최선을 다해 이벤트를 내보냅니다.

이 섹션에서는 이벤트 스키마의 예를 제공하고 규칙을 생성하는 방법을 설명합니다. EventBridge

이벤트 스키마

다음은 EC2 찾기 이벤트에 사용되는 Amazon Inspector 이벤트 형식의 예입니다. 다른 결과 유형 및 이벤트 유형의 스키마 예는 EventBridge 스키마를 참조하세요.

```
{
    "version": "0",
    "id": "66a7a279-5f92-971c-6d3e-c92da0950992",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T22:46:15Z",
    "region": "us-east-1",
    "resources": ["i-0c2a343f1948d5205"],
    "detail": {
        "awsAccountId": "111122223333",
        "description": "\n It was discovered that the sound subsystem in the Linux
 kernel contained a\n race condition in some situations. A local attacker could use
 this to cause\n a denial of service (system crash).",
        "exploitAvailable": "YES",
        "exploitabilityDetails": {
```

```
            "lastKnownExploitAt": "Oct 24, 2022, 11:08:59 PM"
        },
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "fixAvailable": "YES",
        "lastObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "packageVulnerabilityDetails": {
            "cvss": [{
                "baseScore": 4.7,
                "scoringVector": "CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:H",
                "source": "NVD",
                "version": "3.1"
            }],
            "referenceUrls": ["https://lore.kernel.org/all/
CAFcO6XN7JDM4xSXGhtusQfS2mSBcx50VJKwQpCq=WeLt57aaZA@mail.gmail.com/", "https://
ubuntu.com/security/notices/USN-5792-1", "https://ubuntu.com/security/notices/
USN-5791-2", "https://ubuntu.com/security/notices/USN-5791-1", "https://ubuntu.com/
security/notices/USN-5793-2", "https://git.kernel.org/pub/scm/linux/kernel/git/
torvalds/linux.git/commit/?id=8423f0b6d513b259fdab9c9bf4aaa6188d054c2d", "https://
ubuntu.com/security/notices/USN-5793-1", "https://ubuntu.com/security/notices/
USN-5792-2", "https://ubuntu.com/security/notices/USN-5791-3", "https://ubuntu.com/
security/notices/USN-5793-4", "https://ubuntu.com/security/notices/USN-5793-3",
 "https://git.kernel.org/linus/8423f0b6d513b259fdab9c9bf4aaa6188d054c2d(6.0-rc5)",
 "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-3303"],
            "relatedVulnerabilities": [],
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2022/
CVE-2022-3303.html",
            "vendorCreatedAt": "Sep 27, 2022, 11:15:00 PM",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2022-3303",
            "vulnerablePackages": [{
                "arch": "X86_64",
                "epoch": 0,
                "fixedInVersion": "0:5.15.0.1027.31~20.04.16",
                "name": "linux-image-aws",
                "packageManager": "OS",
                "remediation": "apt update && apt install --only-upgrade linux-image-
aws",
                "version": "5.15.0.1026.30~20.04.16"
            }]
        },
        "remediation": {
            "recommendation": {
```

```
                "text": "None Provided"
            }
        },
        "resources": [{
            "details": {
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-0b7ff1a8d69f1bb35",
                    "ipV4Addresses": ["172.31.85.212", "44.203.45.27"],
                    "ipV6Addresses": [],
                    "launchedAt": "Jan 19, 2023, 7:53:14 PM",
                    "platform": "UBUNTU_20_04",
                    "subnetId": "subnet-8213f2a3",
                    "type": "t2.micro",
                    "vpcId": "vpc-ab6650d1"
                }
            },
            "id": "i-0c2a343f1948d5205",
            "partition": "aws",
            "region": "us-east-1",
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2022-3303 - linux-image-aws",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 10:46:15 PM"
    }
}
```


Amazon Inspector 조사 결과를 알려주는 EventBridge 규칙 생성

Amazon Inspector 검색 결과의 가시성을 높이기 EventBridge 위해 메시징 허브로 전송되는 자동 검색 알림을 설정하는 데 사용할 수 있습니다. 이 주제에서는 `CRITICAL` 및 `HIGH` 심각도 결과에 대한 알림을 이메일, Slack 또는 Amazon Chime으로 보내는 방법에 대해 설명합니다. Amazon Simple Notification Service 주제를 설정한 다음 해당 주제를 EventBridge 이벤트 규칙에 연결하는 방법을 알아봅니다.

단계 1. Amazon SNS 주제 및 엔드포인트 설정

자동 알림을 설정하려면 먼저 Amazon Simple Notification Service에서 주제를 설정하고 엔드포인트를 추가해야 합니다. 자세한 내용은 SNS가이드를 참조하십시오.

이 절차는 Amazon Inspector 결과 데이터를 보낼 위치를 설정합니다. 이벤트 규칙을 생성하는 동안 또는 생성한 후에 EventBridge 이벤트 규칙에 SNS 주제를 추가할 수 있습니다.

Email setup

SNS 주제 만들기

1. https://console.aws.amazon.com/sns/v3/home에서 아마존 SNS 콘솔에 로그인합니다.
2. 탐색 창에서 주제를 선택한 다음 주제 생성을 선택합니다.
3. 주제 생성 섹션에서 표준을 선택합니다. 주제 이름을 입력합니다(예: **`Inspector_to_Email`**). 기타 세부 정보는 선택 사항입니다.
4. 주제 생성을 선택합니다. 그러면 새 주제에 대한 세부 정보가 포함된 새 패널이 열립니다.
5. 구독 섹션에서 구독 생성을 선택합니다.
6. a. 프로토콜 메뉴에서 이메일을 선택합니다.
   b. 엔드포인트 필드에 알림을 받을 이메일 주소를 입력합니다.

   > **Note**
   > 구독을 생성한 후 이메일 클라이언트를 통해 구독을 확인해야 합니다.

   c. 구독 생성을 선택합니다.
7. 받은 편지함에서 구독 메시지를 확인하고 구독 확인을 선택합니다.

Slack setup

SNS 주제 만들기

1. https://console.aws.amazon.com/sns/v3/home에서 아마존 SNS 콘솔에 로그인합니다.
2. 탐색 창에서 주제를 선택한 다음 주제 생성을 선택합니다.
3. 주제 생성 섹션에서 표준을 선택합니다. 주제 이름을 입력합니다(예: **`Inspector_to_Slack`**). 기타 세부 정보는 선택 사항입니다. 주제 생성을 선택하여 엔드포인트 생성을 완료합니다.

클라이언트 구성 AWS Chatbot

1. 에서 AWS Chatbot 콘솔로 이동합니다https://console.aws.amazon.com/chatbot/.
2. 구성된 클라이언트 창에서 새 클라이언트 구성을 선택합니다.
3. Slack을 선택한 다음 구성을 선택하여 확인합니다.

> **Note**
>
> Slack을 선택할 때는 허용을 선택하여 AWS Chatbot 의 채널 액세스 권한을 확인해야 합니다.

4. 새 채널 구성을 선택하여 구성 세부 정보 창을 엽니다.
   a. 채널 이름을 입력합니다.
   b. Slack 채널에서 사용할 채널을 선택합니다.
   c. Slack에서 채널 이름을 마우스 오른쪽 버튼으로 클릭하고 링크 복사를 선택하여 프라이빗 채널의 채널 ID를 복사합니다.
   d. AWS Chatbot 창에서 Slack에서 복사한 채널 ID를 비공개 채널 ID 필드에 붙여넣습니다. AWS Management Console
   e. 아직 IAM 역할이 없는 경우 권한에서 템플릿을 사용하여 역할을 생성하도록 선택합니다.
   f. 정책 템플릿에서 알림 권한을 선택합니다. 에 대한 IAM 정책 AWS Chatbot템플릿입니다. 이 정책은 CloudWatch 경보, 이벤트, 로그, Amazon SNS 주제에 필요한 읽기 및 목록 권한을 제공합니다.
   g. 채널 가드레일 정책의 경우 2를 선택하십시오. AmazonInspector ReadOnlyAccess
   h. 이전에 주제를 생성한 지역을 선택한 다음 생성한 Amazon SNS SNS 주제를 선택하여 Slack 채널에 알림을 전송합니다.
5. 구성을 선택합니다.

Amazon Chime setup

SNS 주제 만들기

1. https://console.aws.amazon.com/sns/v3/home에서 아마존 SNS 콘솔에 로그인합니다.
2. 탐색 창에서 주제를 선택한 다음 주제 생성을 선택합니다.

3. 주제 생성 섹션에서 표준을 선택합니다. 주제 이름을 입력합니다(예: **`Inspector_to_Chime`**). 기타 세부 정보는 선택 사항입니다. 주제 생성을 선택하여 완료합니다.

클라이언트 구성 AWS Chatbot

1. 에서 AWS Chatbot 콘솔로 이동합니다https://console.aws.amazon.com/chatbot/.
2. 구성된 클라이언트 패널에서 새 클라이언트 구성을 선택합니다.
3. Chime을 선택한 다음 구성을 선택하여 확인합니다.
4. 구성 세부 정보 창에서 채널 이름을 입력합니다.
5. Amazon Chime에서 원하는 채팅룸을 엽니다.
   a. 오른쪽 상단 모서리에 있는 기어 모양 아이콘을 선택하고 Manage webhooks(Webhook 관리)를 선택합니다.
   b. 복사를 URL 선택하여 웹후크를 URL 클립보드에 복사합니다.
6. AWS Chatbot 창에서 복사한 내용을 Webhook 필드에 URL 붙여넣습니다. AWS Management Console URL
7. 아직 IAM 역할이 없는 경우 권한에서 템플릿을 사용하여 역할을 생성하도록 선택합니다.
8. 정책 템플릿에서 알림 권한을 선택합니다. 에 대한 IAM 정책 AWS Chatbot템플릿입니다. CloudWatch 알람, 이벤트, 로그, Amazon SNS 주제에 필요한 읽기 및 목록 권한을 제공합니다.
9. 이전에 주제를 생성한 지역을 선택한 다음 생성한 Amazon SNS 주제를 선택하여 Amazon SNS Chime 룸에 알림을 전송합니다.
10. 구성을 선택합니다.

단계 2. Amazon EventBridge Inspector 조사 결과에 대한 규칙 생성

1. 자격 증명을 사용하여 로그인합니다.
2. 에서 Amazon EventBridge 콘솔을 엽니다 https://console.aws.amazon.com/events/.
3. 탐색 창에서 규칙을 선택한 다음 규칙 생성을 선택합니다.
4. 규칙의 이름과 설명(선택 사항)을 입력합니다.
5. 이벤트 패턴이 있는 규칙을 선택한 후다음을 선택합니다.
6. 이벤트 패턴 창에서 사용자 지정 패턴 (JSON편집기) 을 선택합니다.

7. 다음을 편집기에 JSON 붙여넣습니다.

```
{
  "source": ["aws.inspector2"],
  "detail-type": ["Inspector2 Finding"],
  "detail": {
    "severity": ["HIGH", "CRITICAL"],
    "status": ["ACTIVE"]
  }
}
```

> **Note**
>
> 이 패턴은 Amazon Inspector에서 탐지한 모든 활성 `CRITICAL` 또는 `HIGH` 심각도 결과에 대해 알림을 보냅니다.

이벤트 패턴 입력을 완료하면 다음을 선택합니다.

8. 대상 선택 페이지에서 AWS 서비스를 선택합니다. 그런 다음 대상 유형 선택에서 SNS주제를 선택합니다.
9. 주제의 경우 1단계에서 만든 SNS 주제의 이름을 선택합니다. 다음을 선택합니다.
10. 필요한 경우 선택적 태그를 추가하고 다음을 선택합니다.
11. 규칙을 검토한 다음 규칙 생성을 선택합니다.

EventBridge Amazon Inspector 다중 계정 환경용

Amazon Inspector의 위임을 받은 관리자인 경우 멤버 계정에서 발생한 해당 결과에 따라 계정에 EventBridge 규칙이 표시됩니다. 이전 섹션에 설명된 대로 관리자 계정을 통해 EventBridge 조사 결과 알림을 설정하면 여러 계정에 대한 알림을 받게 됩니다. 즉, 내 계정에서 생성된 결과와 이벤트 외에도 멤버 계정에서 생성된 결과와 이벤트에 대한 알림을 받게 됩니다.

조사 결과 JSON 세부 정보를 사용하여 Amazon Inspector 검색 결과가 나온 회원 계정을 식별할 수 있습니다. `accountId`

Amazon Inspector에서 대시보드 사용하기

대시보드는 Amazon Inspector가 스캔하는 리소스에 대한 집계된 통계의 스냅샷을 제공합니다. 대시보드를 사용하여 환경 적용 범위와 중요한 결과에 대해 알아보십시오.

> **Note**
>
> 사용자 계정이 조직의 위임된 관리자 계정인 경우 대시보드에는 사용자 계정 및 조직의 다른 모든 계정에 대한 정보가 표시됩니다.

이 섹션에서는 대시보드를 보는 방법과 대시보드를 구성하는 구성 요소를 이해하는 방법을 설명합니다.

주제

- 대시보드 보기
- 대시보드 구성 요소 이해 및 데이터 해석

대시보드 보기

대시보드에는 사용자 환경의 적용 범위와 중요한 결과에 대한 개요가 표시됩니다.

대시보드를 보려면:

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 탐색 창에서 대시보드를 선택합니다.
   a. 대시보드는 5분마다 데이터를 자동으로 새로 고칩니다. 페이지 오른쪽 상단의 새로 고침 아이콘을 선택하여 데이터를 수동으로 새로 고칠 수 있습니다.
   b. 항목을 선택하면 해당 항목에 대한 지원 데이터를 볼 수 있습니다.
   c. 조직의 위임된 관리자 계정인 경우 계정 필드에 구성원 계정 ID를 입력하여 구성원 계정에 대한 집계 통계를 볼 수 있습니다.

대시보드 구성 요소 이해 및 데이터 해석

대시보드의 각 섹션은 주요 지표 및 조사 결과 데이터에 대한 통찰력을 제공하므로 현재 AWS 리소스의 취약성 상태를 이해할 수 있습니다. AWS 리전

환경 적용 범위

환경 적용 범위 섹션은 Amazon Inspector에서 스캔한 리소스에 대한 통계를 제공합니다. 이 섹션에서는 Amazon Inspector에서 스캔한 Amazon EC2 인스턴스, 아마존 ECR 이미지 및 AWS Lambda 함수의 수와 비율을 확인할 수 있습니다. Amazon Inspector의 위임 관리자를 통해 AWS Organizations 여러 계정을 관리하는 경우 총 조직 계정 수, Amazon Inspector가 활성화된 계정 수 및 해당 조직의 적용 범위 비율도 확인할 수 있습니다. 또한 이 섹션에서는 Amazon Inspector가 적용되지 않는 리소스를 확인할 수도 있습니다. 이러한 리소스에는 조직을 위험에 빠뜨리는 데 악용될 수 있는 취약성이 포함되어 있을 수 있습니다. 자세한 내용은 AWS 환경에 대한 Amazon Inspector 적용 범위 평가를 참조하세요.

적용 범위 그룹을 선택하면 선택한 그룹에 대한 계정 관리 페이지로 이동합니다. 계정 관리 페이지에는 Amazon Inspector에서 다루는 계정, Amazon EC2 인스턴스 및 Amazon ECR 리포지토리에 대한 세부 정보가 표시됩니다.

사용 가능한 적용 범위 그룹은 다음과 같습니다.

- 계정
- 인스턴스
- 컨테이너 리포지토리
- 컨테이너 이미지
- Lambda

중요 결과

중요 결과 섹션에서는 환경의 중요 취약점 수와 환경 내 모든 결과의 총 개수를 확인할 수 있습니다. 이 섹션에서는 리소스 및 평가 유형별로 개수가 표시됩니다. 중요 결과 및 Amazon Inspector의 중요도 결정 방식에 대한 자세한 내용은 Amazon Inspector 조사 결과 이해 섹션을 참조하세요.

중요 결과 그룹을 선택하면 모든 결과 페이지로 이동하며 선택한 그룹과 일치하는 모든 중요 결과를 표시하도록 필터가 자동으로 적용됩니다.

사용 가능한 중요 결과 그룹은 다음과 같습니다.

- ECR컨테이너 이미지, 결과

- 아마존 EC2 조사 결과
- 네트워크 연결성 결과
- AWS Lambda 함수 결과

위험에 기반한 해결 방법

위험에 기반한 해결 방법 섹션에는 사용 환경에서 가장 많은 리소스에 영향을 미치는 중요한 취약점이 있는 상위 5개 소프트웨어 패키지가 표시됩니다. 이러한 패키지를 해결하면 환경에 대한 심각한 위험 수를 크게 줄일 수 있습니다. 소프트웨어 패키지 이름을 선택하면 관련 취약성 세부 정보와 영향을 받는 리소스를 확인할 수 있습니다.

가장 중요한 결과가 있는 계정

가장 중요한 검색 결과가 있는 계정 섹션에는 사용자 환경에서 가장 중요한 검색 결과가 있는 상위 5개 AWS 계정과 해당 계정에 대한 검색 결과의 총 수가 표시됩니다. 이 섹션은 Amazon Inspector가 다중 계정 스캔을 사용하도록 구성된 경우 위임된 관리자 계정에서만 볼 수 있습니다. AWS Organizations이 보기는 위임 관리자가 조직 내에서 가장 위험할 수 있는 계정을 파악하는 데 도움이 됩니다.

계정 ID를 선택하면 영향을 받는 멤버 계정에 대한 자세한 정보를 볼 수 있습니다.

가장 중요한 발견이 있는 Amazon ECR 리포지토리

Elastic Container Registry (ECR) 가장 중요한 검색 결과가 있는 리포지토리 섹션에는 사용자 환경에서 가장 중요한 컨테이너 이미지 검색 결과가 있는 상위 5개의 Amazon ECR 리포지토리가 표시됩니다. 뷰에는 리포지토리 이름, AWS 계정 식별자, 리포지토리 생성 날짜, 심각한 취약성 수, 총 취약성 수가 표시됩니다. 이 보기는 가장 위험할 수 있는 리포지토리를 식별하는 데 도움이 됩니다.

리포지토리 이름을 선택하면 영향을 받는 리포지토리에 대한 자세한 정보를 볼 수 있습니다.

가장 중요한 결과가 있는 컨테이너 이미지

가장 중요한 결과가 있는 컨테이너 이미지 섹션에는 사용 환경에서 가장 중요한 결과가 있는 상위 5개 컨테이너 이미지가 표시됩니다. 뷰에는 이미지 태그 데이터, 리포지토리 이름, 이미지 다이제스트, AWS 계정 식별자, 심각한 취약성 수, 총 취약성 수가 표시됩니다. 이 보기는 애플리케이션 소유자가 재구축하여 재실행해야 하는 컨테이너 이미지를 파악하는 데 도움이 됩니다.

컨테이너 이미지를 선택하면 영향을 받는 컨테이너 이미지에 대한 자세한 정보를 볼 수 있습니다.

가장 중요한 결과가 있는 인스턴스

가장 중요한 검색 결과가 있는 인스턴스 섹션에는 가장 중요한 검색 결과가 있는 상위 5개의 Amazon EC2 인스턴스가 표시됩니다. 이 뷰에는 인스턴스 식별자, AWS 계정 식별자, Amazon

Machine Image (AMI) 식별자, 심각한 취약성 수, 총 취약성 수가 표시됩니다. 이 보기는 인프라 소유자가 패치 적용이 필요한 인스턴스를 파악하는 데 도움이 됩니다.

영향을 받는 Amazon 인스턴스에 대한 자세한 정보를 보려면 EC2 인스턴스 ID를 선택합니다.

가장 중요한 발견이 포함된 Amazon 머신 이미지 (AMI)

Amazon Machine Images (AMIs) — 가장 중요한 검색 결과가 포함된 Amazon Machine Images () 섹션에는 사용자 AMIs 환경에서 가장 중요한 검색 결과가 있는 상위 5개 항목이 표시됩니다. 뷰에는 AMI 식별자, AWS 계정 식별자, 환경에서 실행되는 영향을 받는 EC2 인스턴스 수, AMI 생성 날짜, 운영 체제 플랫폼AMI, 심각한 취약성 수, 총 취약성 수가 표시됩니다. 이 뷰를 통해 인프라 소유자는 재구축이 AMIs 필요할 수 있는 항목을 식별할 수 있습니다.

영향을 받는 인스턴스를 선택하면 영향을 받는 인스턴스에서 시작된 인스턴스에 대한 자세한 정보를 볼 수 있습니다. AMI

AWS Lambda 가장 중요한 결과가 있는 함수

가장 중요한 결과가 있는AWS Lambda 함수 섹션에는 사용 환경에서 가장 중요한 결과가 있는 상위 5개 Lambda 함수가 표시됩니다. 이 뷰에는 Lambda 함수 이름 AWS , 계정 식별자, 런타임 환경, 심각한 취약성 수, 취약성이 높은 항목 수, 총 취약성 수가 표시됩니다. 이 보기는 인프라 소유자가 해결이 필요한 Lambda 함수를 파악하는 데 도움이 됩니다.

함수 이름을 선택하면 영향을 받는 함수에 대한 자세한 정보를 볼 수 있습니다. AWS Lambda

Amazon Inspector 취약성 데이터베이스 검색

Amazon Inspector 취약성 데이터베이스에서 일반적인 취약성 및 노출()을 검색할 수 있습니다CVE. Amazon Inspector는 취약성 데이터베이스의 정보를 사용하여 CVE ID와 관련된 세부 정보를 생성합니다. 세부 정보 화면에서 이러한 CVE 세부 정보를 볼 수 있습니다. Amazon Inspector는 취약성 데이터베이스에서 소프트웨어 취약성에 대한 결과를 추적하고 생성합니다. Amazon Inspector는 CVE 세부 정보 화면의 탐지 CVEs 플랫폼 섹션에 나열된 플랫폼만 지원합니다. 이 섹션에서는 CVE ID를 사용하여 Amazon Inspector 실행 가능성 데이터베이스를 검색하는 방법을 설명합니다.

> **Note**
>
> 현재 CVE 검색은 지원되지 않습니다.Microsoft Windows.

취약성 데이터베이스 검색

이 섹션에서는 콘솔 및 Amazon Inspector 에서 취약성 데이터베이스를 검색하는 방법을 설명합니다 API.

> **Note**
>
> 취약성 데이터베이스를 검색 AWS 리전 하려면 먼저 현재 에서 Amazon Inspector를 활성화해야 합니다.

Console

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home에서 Amazon Inspector 콘솔을 엽니다.
2. 탐색 창에서 취약성 데이터베이스 검색 을 선택합니다.
3. 검색 창에 CVE ID를 입력하고 검색을 선택합니다.

API

Amazon Inspector SearchVulnerabilities 를 실행API하고 `filterCriteria`와 같은 형식으로 단일 CVE ID를 제공합니다`CVE-<year>-<ID>`.

CVE 세부 정보 이해

이 섹션에서는 CVE 세부 정보 페이지를 인터페팅하는 방법을 설명합니다.

CVE 세부 정보

CVE 세부 정보 섹션에는 다음 정보가 포함되어 있습니다.

- CVE 설명 및 ID
- CVE 심각도
- 공통 취약성 점수 체계(CVSS) 및 익스플로잇 예측 점수 체계(EPSS) 점수
- 탐지 플랫폼

> **Note**
>
> 이 필드가 비어 있는 경우 Amazon Inspector는 CVE ID에 대한 감지를 지원하지 않습니다.

- 일반적인 약점 열거(CWE)
- 공급업체 생성 및 업데이트 날짜

취약성 인텔리전스

취약성 인텔리전스 섹션은 익스플로잇 대상 및 마지막으로 알려진 퍼블릭 익스플로잇 날짜와 같은 위협 인텔리전스 데이터를 제공합니다.

또한 사이버 보안 및 인프라 보안 기관(CISA)의 데이터를 제공합니다. 여기에는 수정 작업, 알려진 악용 취약성 카탈로그에 이 CVE 추가된 날짜, 연방 기관이 를 수정할 것으로 CISA 예상되는 날짜 시간이 포함됩니다CVE.

참조

참조 섹션에서는 에 대한 자세한 내용을 확인할 수 있는 리소스 링크를 제공합니다CVE.

아마존 SBOMs 인스펙터로 내보내기

소프트웨어 BOM (SBOM) 은 코드베이스에 있는 모든 오픈 소스 및 타사 소프트웨어 구성 요소의 중첩된 인벤토리입니다. Amazon Inspector는 사용자 SBOMs 환경의 개별 리소스를 제공합니다. Amazon Inspector 콘솔 또는 Amazon Inspector를 사용하여 리소스를 API SBOMs 생성할 수 있습니다. Amazon Inspector가 지원하고 모니터링하는 모든 리소스를 내보낼 SBOMs 수 있습니다. 내보낸 정보는 소프트웨어 공급에 대한 정보를 SBOMs 제공합니다. AWS 환경의 적용 범위를 평가하여 리소스 상태를 검토할 수 있습니다. 이 섹션에서는 구성 및 내보내기 SBOMs 방법을 설명합니다.

> **Note**
>
> 현재 Amazon Inspector는 윈도우 아마존 SBOMs 인스턴스용 익스포트를 지원하지 않습니다. EC2

Amazon Inspector 형식

Amazon Inspector는 CyclonedX SBOMs 1.4 및 2.3 호환 포맷으로 익스포트를 지원합니다. SPDX Amazon Inspector는 선택한 Amazon S3 버킷에 SBOMs `JSON` 파일로 내보냅니다.

> **Note**
>
> SPDXAmazon Inspector의 형식 내보내기는 SPDX2.3을 사용하는 시스템과 호환되지만 크리에이티브 커먼즈 제로 (CC0) 필드는 포함하지 않습니다. 이 필드가 포함되어 있으면 사용자가 자료를 재배포하거나 편집할 수 있기 때문입니다.

아마존 인스펙터의 CyclonedX 1.4 SBOM 포맷 예시

```
                    {
  "bomFormat": "CycloneDX",
  "specVersion": "1.4",
  "version": 1,
  "metadata": {
    "timestamp": "2023-06-02T01:17:46Z",
    "component": null,
    "properties": [
      {
```

```
        "name": "imageId",
        "value":
 "sha256:c8ee97f7052776ef223080741f61fcdf6a3a9107810ea9649f904aa4269fdac6"
      },
      {
        "name": "architecture",
        "value": "arm64"
      },
      {
        "name": "accountId",
        "value": "111122223333"
      },
      {
        "name": "resourceType",
        "value": "AWS_ECR_CONTAINER_IMAGE"
      }
    ]
  },
  "components": [
    {
      "type": "library",
      "name": "pip",
      "purl": "pkg:pypi/pip@22.0.4?path=usr/local/lib/python3.8/site-packages/
pip-22.0.4.dist-info/METADATA",
      "bom-ref": "98dc550d1e9a0b24161daaa0d535c699"
    },
    {
      "type": "application",
      "name": "libss2",
      "purl": "pkg:dpkg/libss2@1.44.5-1+deb10u3?
arch=ARM64&epoch=0&upstream=libss2-1.44.5-1+deb10u3.src.dpkg",
      "bom-ref": "2f4d199d4ef9e2ae639b4f8d04a813a2"
    },
    {
      "type": "application",
      "name": "liblz4-1",
      "purl": "pkg:dpkg/liblz4-1@1.8.3-1+deb10u1?
arch=ARM64&epoch=0&upstream=liblz4-1-1.8.3-1+deb10u1.src.dpkg",
      "bom-ref": "9a6be8907ead891b070e60f5a7b7aa9a"
    },
    {
      "type": "application",
      "name": "mawk",
```

```
      "purl": "pkg:dpkg/mawk@1.3.3-17+b3?
arch=ARM64&epoch=0&upstream=mawk-1.3.3-17+b3.src.dpkg",
      "bom-ref": "c2015852a729f97fde924e62a16f78a5"
    },
    {
      "type": "application",
      "name": "libgmp10",
      "purl": "pkg:dpkg/libgmp10@6.1.2+dfsg-4+deb10u1?
arch=ARM64&epoch=2&upstream=libgmp10-6.1.2+dfsg-4+deb10u1.src.dpkg",
      "bom-ref": "52907290f5beef00dff8da77901b1085"
    },
    {
      "type": "application",
      "name": "ncurses-bin",
      "purl": "pkg:dpkg/ncurses-bin@6.1+20181013-2+deb10u3?
arch=ARM64&epoch=0&upstream=ncurses-bin-6.1+20181013-2+deb10u3.src.dpkg",
      "bom-ref": "cd20cfb9ebeeadba3809764376f43bce"
    }
  ],
  "vulnerabilities": [
    {
      "id": "CVE-2022-40897",
      "affects": [
        {
          "ref": "a74a4862cc654a2520ec56da0c81cdb3"
        },
        {
          "ref": "0119eb286405d780dc437e7dbf2f9d9d"
        }
      ]
    }
  ]
}
```


아마존 인스펙터의 SPDX 2.3 SBOM 포맷 예제


```
{
 "name": "409870544328/EC2/i-022fba820db137c64/ami-074ea14c08effb2d8",
 "spdxVersion": "SPDX-2.3",
 "creationInfo": {
  "created": "2023-06-02T21:19:22Z",
```

```
  "creators": [
   "Organization: 409870544328",
   "Tool: Amazon Inspector SBOM Generator"
  ]
 },
 "documentNamespace": "EC2://i-022fba820db137c64/AMAZON_LINUX_2/null/x86_64",
 "comment": "",
 "packages": [{
   "name": "elfutils-libelf",
   "versionInfo": "0.176-2.amzn2",
   "downloadLocation": "NOASSERTION",
   "sourceInfo": "/var/lib/rpm/Packages",
   "filesAnalyzed": false,
   "externalRefs": [{
    "referenceCategory": "PACKAGE-MANAGER",
    "referenceType": "purl",
    "referenceLocator": "pkg:rpm/elfutils-libelf@0.176-2.amzn2?
arch=X86_64&epoch=0&upstream=elfutils-libelf-0.176-2.amzn2.src.rpm"
   }],
   "SPDXID": "SPDXRef-Package-rpm-elfutils-libelf-ddf56a513c0e76ab2ae3246d9a91c463"
  },
  {
   "name": "libcurl",
   "versionInfo": "7.79.1-1.amzn2.0.1",
   "downloadLocation": "NOASSERTION",
   "sourceInfo": "/var/lib/rpm/Packages",
   "filesAnalyzed": false,
   "externalRefs": [{
     "referenceCategory": "PACKAGE-MANAGER",
     "referenceType": "purl",
     "referenceLocator": "pkg:rpm/libcurl@7.79.1-1.amzn2.0.1?
arch=X86_64&epoch=0&upstream=libcurl-7.79.1-1.amzn2.0.1.src.rpm"
    },
    {
     "referenceCategory": "SECURITY",
     "referenceType": "vulnerability",
     "referenceLocator": "CVE-2022-32205"
    }
   ],
   "SPDXID": "SPDXRef-Package-rpm-libcurl-710fb33829bc5106559bcd380cddb7d5"
  },
  {
   "name": "hunspell-en-US",
   "versionInfo": "0.20121024-6.amzn2.0.1",
```

```
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
     "referenceCategory": "PACKAGE-MANAGER",
     "referenceType": "purl",
     "referenceLocator": "pkg:rpm/hunspell-en-US@0.20121024-6.amzn2.0.1?
arch=NOARCH&epoch=0&upstream=hunspell-en-US-0.20121024-6.amzn2.0.1.src.rpm"
    }],
    "SPDXID": "SPDXRef-Package-rpm-hunspell-en-US-de19ae0883973d6cea5e7e079d544fe5"
   },
   {
    "name": "grub2-tools-minimal",
    "versionInfo": "2.06-2.amzn2.0.6",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
      "referenceCategory": "PACKAGE-MANAGER",
      "referenceType": "purl",
      "referenceLocator": "pkg:rpm/grub2-tools-minimal@2.06-2.amzn2.0.6?
arch=X86_64&epoch=1&upstream=grub2-tools-minimal-2.06-2.amzn2.0.6.src.rpm"
     },
     {
      "referenceCategory": "SECURITY",
      "referenceType": "vulnerability",
      "referenceLocator": "CVE-2021-3981"
     }
    ],
    "SPDXID": "SPDXRef-Package-rpm-grub2-tools-minimal-c56b7ea76e5a28ab8f232ef6d7564636"
   },
   {
    "name": "unixODBC-devel",
    "versionInfo": "2.3.1-14.amzn2",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
     "referenceCategory": "PACKAGE-MANAGER",
     "referenceType": "purl",
     "referenceLocator": "pkg:rpm/unixODBC-devel@2.3.1-14.amzn2?
arch=X86_64&epoch=0&upstream=unixODBC-devel-2.3.1-14.amzn2.src.rpm"
    }],
    "SPDXID": "SPDXRef-Package-rpm-unixODBC-devel-1bb35add92978df021a13fc9f81237d2"
```

```
  }
 ],
 "relationships": [{
   "spdxElementId": "SPDXRef-DOCUMENT",
   "relatedSpdxElement": "SPDXRef-Package-rpm-elfutils-libelf-
ddf56a513c0e76ab2ae3246d9a91c463",
   "relationshipType": "DESCRIBES"
  },
  {
   "spdxElementId": "SPDXRef-DOCUMENT",
   "relatedSpdxElement": "SPDXRef-Package-rpm-yajl-8476ce2db98b28cfab2b4484f84f1903",
   "relationshipType": "DESCRIBES"
  },
  {
   "spdxElementId": "SPDXRef-DOCUMENT",
   "relatedSpdxElement": "SPDXRef-Package-rpm-unixODBC-
devel-1bb35add92978df021a13fc9f81237d2",
   "relationshipType": "DESCRIBES"
  }
 ],
 "SPDXID": "SPDXRef-DOCUMENT"
}
```


필터: SBOMs

SBOMs내보낼 때 필터를 포함하여 리소스의 특정 하위 집합에 대한 보고서를 만들 수 있습니다. 필터를 제공하지 않으면 SBOMs 지원되는 모든 활성 리소스가 내보내집니다. 또한 위임 관리자인 경우 여기에 모든 멤버를 위한 리소스도 포함됩니다. 다음과 같은 필터를 사용할 수 있습니다.

- AccountID — 이 필터는 특정 계정 ID와 관련된 모든 리소스를 SBOMs 내보내는 데 사용할 수 있습니다.
- EC2인스턴스 태그 - 이 필터는 특정 태그가 있는 EC2 인스턴스를 내보내는 SBOMs 데 사용할 수 있습니다.
- 함수 이름 - 이 필터는 특정 Lambda 함수를 내보내는 SBOMs 데 사용할 수 있습니다.
- 이미지 태그 - 이 필터는 특정 태그가 있는 컨테이너 이미지를 내보내는 SBOMs 데 사용할 수 있습니다.
- Lambda 함수 태그 — 이 필터는 특정 태그가 있는 Lambda 함수를 내보내는 SBOMs 데 사용할 수 있습니다.

- 리소스 유형 - 이 필터는 리소스 유형:EC2/ECR/Lambda를 필터링하는 데 사용할 수 있습니다.
- 리소스 ID - 이 필터는 특정 리소스에 SBOM 대한 를 내보내는 데 사용할 수 있습니다.
- 리포지토리 이름 - 이 필터는 특정 리포지토리의 컨테이너 이미지를 생성하는 SBOMs 데 사용할 수 있습니다.

구성 및 내보내기 SBOMs

SBOMs내보내려면 먼저 Amazon S3 버킷과 Amazon Inspector에서 사용할 수 있는 AWS KMS 키를 구성해야 합니다. 필터를 사용하여 리소스의 특정 하위 SBOMs 집합을 내보낼 수 있습니다. AWS 조직의 여러 계정에 SBOMs 대해 내보내려면 Amazon Inspector의 위임 관리자로 로그인한 상태에서 다음 단계를 수행하십시오.

사전 조건

- Amazon Inspector에서 적극적으로 모니터링하고 있는 지원 리소스
- Amazon Inspector에서 객체를 추가할 수 있도록 허용하는 정책으로 구성된 Amazon S3 버킷. 정책 구성에 대한 자세한 내용은 내보내기 권한 구성을 참조하세요.
- Amazon Inspector가 보고서를 암호화하는 데 사용할 수 있도록 허용하는 정책으로 구성된 AWS KMS 키입니다. 정책 구성에 대한 자세한 내용은 내보내기를 위한 AWS KMS 키 구성을 참조하십시오.

> **Note**
>
> 이전에 Amazon S3 버킷과 검색 결과 내보내기용 AWS KMS 키를 구성한 경우 내보내기에 동일한 버킷과 키를 사용할 수 있습니다. SBOM

원하는 액세스 방법을 선택하여 내보내십시오SBOM.

Console

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 내보내려는 리소스가 있는 지역을 선택합니다. SBOM
3. 탐색 창에서 내보내기를 선택합니다. SBOMs

4. (선택 사항) 내보내기 SBOMs 페이지에서 필터 추가 메뉴를 사용하여 보고서를 생성할 리소스의 하위 집합을 선택합니다. 필터를 제공하지 않으면 Amazon Inspector에서 모든 활성 리소스에 대한 보고서를 내보냅니다. 위임 관리자인 경우 여기에 조직의 모든 활성 리소스가 포함됩니다.
5. 내보내기 설정에서 원하는 형식을 선택합니다. SBOM
6. Amazon S3를 URI 입력하거나 Amazon S3 찾아보기를 선택하여 저장할 Amazon S3 위치를 선택합니다SBOM.
7. Amazon Inspector에서 보고서를 암호화하는 데 사용하도록 구성된 AWS KMS 키를 입력합니다.

API

- 리소스를 프로그래밍 방식으로 SBOMs 내보내려면 Amazon API Inspector의 CreateSbomExport작업을 사용하십시오.

  요청에서 `reportFormat` 파라미터를 사용하여 SBOM 출력 형식을 지정하고 또는 를 선택합니다`CYCLONEDX_1_4`. `SPDX_2_3` `s3Destination` 파라미터는 필수이며, Amazon Inspector에서 쓰기를 허용하는 정책으로 구성된 S3 버킷을 지정해야 합니다. 선택적으로 `resourceFilterCriteria` 파라미터를 사용하여 보고서의 범위를 특정 리소스로 제한할 수 있습니다.

AWS CLI

- 리소스를 SBOMs 익스포트하려면 다음 명령어를 AWS Command Line Interface 실행하세요.

  ```
  aws inspector2 create-sbom-export --report-format
  FORMAT --s3-destination bucketName=amzn-s3-demo-
  bucket1,keyPrefix=PREFIX,kmsKeyArn=arn:aws:kms:Region:111122223333:key/123
  ```

  요청에서 다음을 대체하십시오.*FORMAT* 원하는 형식으로, `CYCLONEDX_1_4` 또는`SPDX_2_3`. 그런 다음 교체하십시오.*### ## ######* 내보낼 S3 버킷의 이름, S3의 출력에 사용할 접두사, 보고서를 암호화하는 데 사용하는 KMS 키가 포함된 s3 대상의 경우 ARN

Amazon Inspector EventBridge 이벤트에 대한 Amazon 이벤트 스키마 Amazon Inspector

Amazon EventBridge은 애플리케이션 및 기타 의 실시간 데이터 스트림 AWS 서비스 을 Amazon Kinesis Data Streams의 AWS Lambda 함수, Amazon Simple Notification Service 주제 및 데이터 스트림과 같은 대상에 제공합니다. 다른 애플리케이션, 서비스 및 시스템과의 통합을 지원하기 위해 Amazon Inspector는 결과를 이벤트 EventBridge 로 https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-events.html에 자동으로 게시합니다. Amazon Inspector를 사용하여 조사 결과, 적용 범위 및 스캔에 대한 이벤트를 게시할 수 있습니다. 이 섹션에서는 EventBridge 이벤트에 대한 예제 스키마를 제공합니다.

주제

- Amazon Inspector용 Amazon EventBridge 기본 스키마
- Amazon Inspector 결과 이벤트 스키마 예제
- Amazon Inspector 최초 스캔 완료 이벤트 스키마 예제
- Amazon Inspector 적용 범위 이벤트 스키마 예제

Amazon Inspector용 Amazon EventBridge 기본 스키마

다음은 Amazon Inspector 에 대한 EventBridge 이벤트의 기본 스키마의 예입니다. 이벤트 세부 정보는 이벤트 유형에 따라 다릅니다.

```
{
    "version": "0",
    "id": "Event ID",
    "detail-type": "Inspector2 *event type*",
    "source": "aws.inspector2",
    "account": "AWS ## ID (string)",
    "time": "event timestamp (string)",
    "region": "AWS ## (string)",
    "resources": [
        *IDs or ARNs of the resources involved in the event*
    ],
    "detail": {
        *Details of an Amazon Inspector event type*
    }
```

```
}
```

Amazon Inspector 결과 이벤트 스키마 예제

다음은 Amazon Inspector 결과에 대한 EventBridge 이벤트 스키마의 예입니다. 결과 이벤트는 Amazon Inspector가 리소스 중 하나에서 소프트웨어 취약성 또는 네트워크 문제를 식별할 경우에 생성됩니다. 이 유형의 이벤트에 대한 대응으로 알림을 생성하는 방법에 대한 지침은 Amazon을 사용하여 Amazon Inspector의 조사 결과에 대한 사용자 지정 응답 생성 EventBridge을 참조하세요.

다음은 결과 이벤트를 식별하는 필드입니다.

- `detail-type` 필드가 `Inspector2 Finding`으로 설정되어 있습니다.
- `detail` 객체가 결과를 설명합니다.

아래 옵션 중 하나를 선택하면 다양한 리소스에 대한 결과 이벤트 스키마와 결과 유형을 확인할 수 있습니다.

Amazon EC2 package vulnerability finding

```
{
    "version": "0",
    "id": "4d621919-f1f4-4201-a0e2-37e4e330ff51",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-09-04T17:00:36Z",
    "region": "eu-central-1",
    "resources": [
        "i-12345678901234567"
    ],
    "detail": {
        "awsAccountId": "123456789012",
        "description": "In snapd versions prior to 2.62, snapd failed to properly
 check the destination of symbolic links when extracting a snap. The snap format
 is a squashfs file-system image and so can contain symbolic links and other file
 types. Various file entries within the snap squashfs image (such as icons and
 desktop files etc) are directly read by snapd when it is extracted. An attacker who
 could convince a user to install a malicious snap which contained symbolic links
 at these paths could then cause snapd to write out the contents of the symbolic
```

```
 link destination into a world-readable directory. This in-turn could allow an
 unprivileged user to gain access to privileged information.",
        "epss": {
            "score": 0.00043
        },
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Wed Sep 04 16:59:44.356 UTC 2024",
        "fixAvailable": "YES",
        "inspectorScore": 4.8,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "adjustments": [],
                "cvssSource": "UBUNTU_CVE",
                "score": 4.8,
                "scoreSource": "UBUNTU_CVE",
                "scoringVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:L",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Wed Sep 04 16:59:44.476 UTC 2024",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 4.8,
                    "scoringVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:L",
                    "source": "UBUNTU_CVE",
                    "version": "3.1"
                },
                {
                    "baseScore": 7.3,
                    "scoringVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:H/I:H/A:H",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://www.cve.org/CVERecord?id=CVE-2024-29069",
                "https://ubuntu.com/security/notices/USN-6940-1"
            ],
            "relatedVulnerabilities": [
                "USN-6940-1"
            ],
```

```
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2024/
CVE-2024-29069.html",
            "vendorCreatedAt": "Thu Jul 25 20:15:00.000 UTC 2024",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2024-29069",
            "vulnerablePackages": [
                {
                    "arch": "ALL",
                    "epoch": 0,
                    "fixedInVersion": "0:2.63+22.04ubuntu0.1",
                    "name": "snapd",
                    "packageManager": "OS",
                    "remediation": "apt-get update && apt-get upgrade",
                    "version": "2.63"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsEc2Instance": {
                        "iamInstanceProfileArn":
 "arn:aws:iam::123456789012:instance-profile/AmazonSSMRoleForInstancesQuickSetup",
                        "imageId": "ami-02ff980600c693b38",
                        "ipV4Addresses": [
                            "1.23.456.789",
                            "123.45.67.890"
                        ],
                        "ipV6Addresses": [],
                        "launchedAt": "Wed Sep 04 16:57:40.000 UTC 2024",
                        "platform": "UBUNTU_22_04",
                        "subnetId": "subnet-12345678",
                        "type": "t2.small",
                        "vpcId": "vpc-12345678"
                    }
                },
                "id": "i-12345678901234567",
                "partition": "aws",
```

```
                "region": "eu-central-1",
                "type": "AWS_EC2_INSTANCE"
            }
        ],
        "severity": "MEDIUM",
        "status": "CLOSED",
        "title": "CVE-2024-29069 - snapd",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 17:00:36.951 UTC 2024"
    }
}
```

Amazon EC2 network reachability finding

```
{
    "version": "0",
    "id": "9eb1603b-4263-19ec-8be2-33184694cb92",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-09-05T13:06:56Z",
    "region": "eu-central-1",
    "resources": ["i-12345678901234567"],
    "detail": {
        "awsAccountId": "123456789012",
        "description": "On the instance i-12345678901234567, the port range
 22-22 is reachable from the InternetGateway igw-261bab4d from an attached ENI
 eni-094ad651219472857.",
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Thu Sep 05 13:06:56.334 UTC 2024",
        "lastObservedAt": "Thu Sep 05 13:06:56.334 UTC 2024",
        "networkReachabilityDetails": {
            "networkPath": {
                "steps": [{
                    "componentId": "igw-261bab4d",
                    "componentType": "AWS::EC2::InternetGateway"
                }, {
                    "componentId": "acl-171b527d",
                    "componentType": "AWS::EC2::NetworkAcl"
                }, {
```

```
                    "componentId": "sg-0d34debf87410f2d9",
                    "componentType": "AWS::EC2::SecurityGroup"
                }, {
                    "componentId": "eni-094ad651219472857",
                    "componentType": "AWS::EC2::NetworkInterface"
                }, {
                    "componentId": "i-12345678901234567",
                    "componentType": "AWS::EC2::Instance"
                }]
            },
            "openPortRange": {
                "begin": 22,
                "end": 22
            },
            "protocol": "TCP"
        },
        "remediation": {
            "recommendation": {
                "text": "You can restrict access to your instance by modifying the
 Security Groups or ACLs in the network path."
            }
        },
        "resources": [{
            "details": {
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::123456789012:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-02ff980600c693b38",
                    "ipV4Addresses": ["1.23.456.789", "123.45.67.890"],
                    "ipV6Addresses": [],
                    "launchedAt": "Wed Sep 04 17:41:24.000 UTC 2024",
                    "platform": "UBUNTU_22_04",
                    "subnetId": "subnet-12345678",
                    "type": "t2.small",
                    "vpcId": "vpc-12345678"
                }
            },
            "id": "i-12345678901234567",
            "partition": "aws",
            "region": "eu-central-1",
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
```

```
        "title": "Port 22 is reachable from an Internet Gateway - TCP",
        "type": "NETWORK_REACHABILITY",
        "updatedAt": "Thu Sep 05 13:06:56.334 UTC 2024"
    }
}
```

Amazon ECR package vulnerability finding

```
{
    "version": "0",
    "id": "5325facf-a1aa-7d97-6bce-25fde6f6d2fc",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-09-04T16:55:38Z",
    "region": "eu-central-1",
    "resources": [
        "arn:aws:ecr:eu-central-1:123456789012:repository/inspector2/
sha256:84f507df33c6864d49c296fb734192696e4cb6f78166ac51ac8b9b118181085d"
    ],
    "detail": {
        "awsAccountId": "123456789012",
        "description": "Possible denial of service in X.509 name checks",
        "epss": {
            "score": 0.00045
        },
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Wed Sep 04 16:55:38.411 UTC 2024",
        "fixAvailable": "YES",
        "lastObservedAt": "Wed Sep 04 16:55:38.411 UTC 2024",
        "packageVulnerabilityDetails": {
            "cvss": [],
            "referenceUrls": [
                "https://www.cve.org/CVERecord?id=CVE-2024-6119",
                "https://ubuntu.com/security/notices/USN-6986-1"
            ],
            "relatedVulnerabilities": [
                "USN-6986-1"
            ],
```

```
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2024/
CVE-2024-6119.html",
            "vendorCreatedAt": "Tue Sep 03 00:00:00.000 UTC 2024",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2024-6119",
            "vulnerablePackages": [
                {
                    "arch": "ARM64",
                    "epoch": 0,
                    "fixedInVersion": "0:3.0.13-0ubuntu3.4",
                    "name": "libssl3t64",
                    "packageManager": "OS",
                    "release": "0ubuntu3.2",
                    "remediation": "apt-get update && apt-get upgrade",
                    "sourceLayerHash":
 "sha256:1567e7ea90b67fc95ccdeeec39bdc3045098dee7e0c604975b957a9f8c0e9616",
                    "version": "3.0.13"
                },
                {
                    "arch": "ARM64",
                    "epoch": 0,
                    "fixedInVersion": "0:3.0.13-0ubuntu3.4",
                    "name": "openssl",
                    "packageManager": "OS",
                    "release": "0ubuntu3.2",
                    "remediation": "apt-get update && apt-get upgrade",
                    "sourceLayerHash":
 "sha256:1567e7ea90b67fc95ccdeeec39bdc3045098dee7e0c604975b957a9f8c0e9616",
                    "version": "3.0.13"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsEcrContainerImage": {
                        "architecture": "arm64",
```

```
                        "imageHash":
 "sha256:84f507df33c6864d49c296fb734192696e4cb6f78166ac51ac8b9b118181085d",
                        "imageTags": [
                            "ubuntu_latest"
                        ],
                        "platform": "UBUNTU_24_04",
                        "pushedAt": "Wed Sep 04 16:55:28.000 UTC 2024",
                        "registry": "123456789012",
                        "repositoryName": "inspector2"
                    }
                },
                "id": "arn:aws:ecr:eu-central-1:123456789012:repository/inspector2/
sha256:84f507df33c6864d49c296fb734192696e4cb6f78166ac51ac8b9b118181085d",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_ECR_CONTAINER_IMAGE"
            }
        ],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2024-6119 - libssl3t64, openssl",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 16:55:38.411 UTC 2024"
    }
}
```

Lambda package vulnerability finding

```
{
    "version": "0",
    "id": "9eadd71a-e49c-9864-6ba9-2a5d3f83c88f",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-09-04T16:50:37Z",
    "region": "eu-central-1",
    "resources": [
        "arn:aws:lambda:eu-central-1:123456789012:function:VulnerableFunction:
$LATEST"
    ],
    "detail": {
```

```
        "awsAccountId": "123456789012",
        "description": "Flask is a lightweight WSGI web application framework. When
 all of the following conditions are met, a response containing data intended for
 one client may be cached and subsequently sent by the proxy to other clients. If
 the proxy also caches `Set-Cookie` headers, it may send one client's `session`
 cookie to other clients. The severity depends on the application's use of the
 session and the proxy's behavior regarding cookies. The risk depends on all these
 conditions being met.\n\n1. The application must be hosted behind a caching proxy
 that does not strip cookies or ignore responses with cookies. 2. The application
 sets `session.permanent = True` 3. The application does not access or modify the
 session at any point during a request. 4. `SESSION_REFRESH_EACH_REQUEST` enabled
 (the default). 5. The application does not set a `Cache-Control` header to indicate
 that a page is private or should not be cached.\n\nThis happens because vulnerable
 versions of Flask only set the `Vary: Cookie` header when the session is ac",
        "epss": {
            "score": 0.00208
        },
        "exploitAvailable": "YES",
        "exploitabilityDetails": {
            "lastKnownExploitAt": "Sat Aug 31 00:04:50.000 UTC 2024"
        },
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Wed Sep 04 16:50:37.627 UTC 2024",
        "fixAvailable": "YES",
        "inspectorScore": 7.5,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "cvssSource": "NVD",
                "score": 7.5,
                "scoreSource": "NVD",
                "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Wed Sep 04 16:50:37.627 UTC 2024",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 7.5,
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N",
                    "source": "NVD",
                    "version": "3.1"
                }
```

```
            ],
            "referenceUrls": [
                "https://www.debian.org/security/2023/dsa-5442",
                "https://lists.debian.org/debian-lts-announce/2023/08/msg00024.html"
            ],
            "relatedVulnerabilities": [],
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2023-30861",
            "vendorCreatedAt": "Tue May 02 18:15:52.000 UTC 2023",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Sun Aug 20 21:15:09.000 UTC 2023",
            "vulnerabilityId": "CVE-2023-30861",
            "vulnerablePackages": [
                {
                    "epoch": 0,
                    "filePath": "requirements.txt",
                    "fixedInVersion": "2.3.2",
                    "name": "flask",
                    "packageManager": "PIP",
                    "version": "2.0.0"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
                            "X86_64"
                        ],
                        "codeSha256": "O7jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
                        "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
                        "functionName": "VulnerableFunction",
                        "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
                        "packageType": "ZIP",
                        "runtime": "PYTHON_3_11",
                        "version": "$LATEST"
```

```
                    }
                },
                "id": "arn:aws:lambda:eu-
central-1:123456789012:function:VulnerableFunction:$LATEST",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_LAMBDA_FUNCTION"
            }
        ],
        "severity": "HIGH",
        "status": "ACTIVE",
        "title": "CVE-2023-30861 - flask",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 16:50:37.627 UTC 2024"
    }
}
```

Lambda code vulnerability finding

```
{
    "version": "0",
    "id": "e764f7be-f931-ff1b-204b-8cab2d91724b",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-09-04T16:51:01Z",
    "region": "eu-central-1",
    "resources": [
        "arn:aws:lambda:eu-central-1:123456789012:function:VulnerableFunction:
$LATEST"
    ],
    "detail": {
        "awsAccountId": "123456789012",
        "codeVulnerabilityDetails": {
            "cwes": [
                "CWE-798"
            ],
            "detectorId": "python/hardcoded-credentials@v1.0",
            "detectorName": "Hardcoded credentials",
            "detectorTags": [
                "secrets",
```

```
                "security",
                "owasp-top10",
                "top25-cwes",
                "cwe-798",
                "Python"
            ],
            "filePath": {
                "endLine": 6,
                "fileName": "lambda_function.py",
                "filePath": "lambda_function.py",
                "startLine": 6
            },
            "ruleId": "python-detect-hardcoded-aws-credentials"
        },
        "description": "Access credentials, such as passwords and access keys,
 should not be hardcoded in source code. Hardcoding credentials may cause leaks even
 after removing them. This is because version control systems might retain older
 versions of the code. Credentials should be stored securely and obtained from the
 runtime environment.",
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Wed Sep 04 16:51:01.869 UTC 2024",
        "lastObservedAt": "Wed Sep 04 16:51:01.869 UTC 2024",
        "remediation": {
            "recommendation": {
                "text": "Your code uses hardcoded AWS credentials which might
 allow unauthorized users access to your AWS account. These attacks can occur
 a long time after the credentials are removed from the code. We recommend that
 you set AWS credentials with environment variables or an AWS profile instead.
 You should consider deleting the affected account or rotating the secret key
 and then monitoring Amazon CloudWatch for unexpected activity.\n[https://
boto3.amazonaws.com/v1/documentation/api/latest/guide/credentials.html](https://
boto3.amazonaws.com/v1/documentation/api/latest/guide/credentials.html)"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
                            "X86_64"
                        ],
                        "codeSha256": "O7jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
```

```
                        "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
                        "functionName": "VulnerableFunction",
                        "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
                        "packageType": "ZIP",
                        "runtime": "PYTHON_3_11",
                        "version": "$LATEST"
                    }
                },
                "id": "arn:aws:lambda:eu-
central-1:123456789012:function:VulnerableFunction:$LATEST",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_LAMBDA_FUNCTION"
            }
        ],
        "severity": "CRITICAL",
        "status": "ACTIVE",
        "title": "CWE-798 - Hardcoded credentials",
        "type": "CODE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 16:51:01.869 UTC 2024"
    }
}
```

> **Note**
>
> 세부 정보 값은 단일 결과의 JSON 세부 정보를 객체로 반환합니다. 배열 내의 여러 결과를 지원하는 전체 결과 응답 구문은 반환하지 않습니다.

Amazon Inspector 최초 스캔 완료 이벤트 스키마 예제

다음은 초기 스캔을 완료하기 위한 Amazon Inspector 이벤트의 이벤트 EventBridge 스키마의 예입니다. 이 이벤트는 Amazon Inspector에서 리소스 중 하나에 대한 최초 스캔을 완료할 경우에 생성됩니다.

다음은 최초 스캔 완료 이벤트를 식별하는 필드입니다.

- `detail-type` 필드가 `Inspector2 Scan`으로 설정되어 있습니다.

- `detail` 객체에는 해당 심각도 범주(예: `CRITICAL`, `HIGH`, `MEDIUM`)에 있는 결과 수를 자세히 설명하는 `finding-severity-counts` 객체가 포함되어 있습니다.

아래 옵션 중 하나를 선택하면 여러 최초 스캔 이벤트 스키마를 리소스 유형별로 확인할 수 있습니다.

Amazon EC2 instance initial scan

```
{
    "version": "0",
    "id": "28a46762-6ac8-6cc4-4f55-bc9ab99af928",
    "detail-type": "Inspector2 Scan",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T22:52:35Z",
    "region": "us-east-1",
    "resources": [
        "i-087d63509b8c97098"
    ],
    "detail": {
        "scan-status": "INITIAL_SCAN_COMPLETE",
        "finding-severity-counts": {
            "CRITICAL": 0,
            "HIGH": 0,
            "MEDIUM": 0,
            "TOTAL": 0
        },
        "instance-id": "i-087d63509b8c97098",
        "version": "1.0"
    }
}
```

Amazon ECR image initial scan

```
{
    "version": "0",
    "id": "fdaa751a-984c-a709-44f9-9a9da9cd3606",
    "detail-type": "Inspector2 Scan",
    "source": "aws.inspector2",
    "account": "111122223333",
```

```
    "time": "2023-01-20T23:15:18Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ecr:us-east-1:111122223333:repository/inspector2"
    ],
    "detail": {
        "scan-status": "INITIAL_SCAN_COMPLETE",
        "repository-name": "arn:aws:ecr:us-east-1:111122223333:repository/
inspector2",
        "finding-severity-counts": {
            "CRITICAL": 0,
            "HIGH": 0,
            "MEDIUM": 0,
            "TOTAL": 0
        },
        "image-digest":
 "sha256:965fbcae990b0467ed5657caceaec165018ef44a4d2d46c7cdea80a9dff0d1ea",
        "image-tags": [
            "ubuntu22"
        ],
        "version": "1.0"
    }
}
```

Lambda function initial scan

```
{
  "version": "0",
  "id": "4f290a7c-361b-c442-03c8-a629f6f20d6c",
  "detail-type": "Inspector2 Scan",
  "source": "aws.inspector2",
  "account": "111122223333",
  "time": "2023-02-23T18:06:03Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:lambda:us-west-2:111122223333:function:lambda-example:$LATEST"
  ],
  "detail": {
    "scan-status": "INITIAL_SCAN_COMPLETE",
    "finding-severity-counts": {
```

```
        "CRITICAL": 0,
        "HIGH": 0,
        "MEDIUM": 0,
        "TOTAL": 0
      },
      "version": "1.0"
    }
  }
```

Amazon Inspector 적용 범위 이벤트 스키마 예제

다음은 적용 범위에 대한 Amazon Inspector EventBridge 이벤트의 이벤트 스키마의 예입니다. 이 이벤트는 리소스에 대한 Amazon Inspector 스캔 적용 범위가 변경될 때 생성됩니다. 다음은 적용 범위 이벤트를 식별하는 필드입니다.

- `detail-type` 필드가 `Inspector2 Coverage`으로 설정되어 있습니다.
- `detail` 객체에는 리소스의 새 스캔 상태를 나타내는 `scanStatus` 객체가 포함되어 있습니다.

```
{
    "version": "0",
    "id": "000adda5-0fbf-913e-bc0e-10f0376412aa",
    "detail-type": "Inspector2 Coverage",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T22:51:39Z",
    "region": "us-east-1",
    "resources": [
        "i-087d63509b8c97098"
    ],
    "detail": {
        "scanStatus": {
            "reason": "UNMANAGED_EC2_INSTANCE",
            "statusCodeValue": "INACTIVE"
        },
        "scanType": "PACKAGE",
        "eventTimestamp": "2023-01-20T22:51:35.665501Z",
        "version": "1.0"
```

```
    }
}
```

Amazon Inspector 스캔을 CI/CD 파이프라인에 통합

Amazon Inspector CI/CD 통합은 Amazon Inspector SBOM Generator 및 Amazon Inspector ScanAPI을 사용하여 컨테이너 이미지에 대한 취약성 보고서를 생성합니다. Amazon Inspector SBOM Generator는 SBOM아카이브, 컨테이너 이미지, 디렉터리, 로컬 시스템 및 컴파일된 Go 그리고 Rust 바이너리. Amazon Inspector Scan은 를 API 스캔SBOM하여 탐지된 취약성에 대한 세부 정보가 포함된 보고서를 생성합니다. Amazon Inspector 컨테이너 이미지 스캔을 CI/CD 파이프라인과 통합하여 소프트웨어 취약성을 스캔하고 취약성 보고서를 생성하여 배포 전에 위험을 조사하고 해결할 수 있습니다. CI/CD 통합을 설정하려면 플러그인을 사용하거나 Amazon Inspector SBOM Generator 및 Amazon Inspector Scan을 사용하여 사용자 지정 CI/CD 통합을 생성할 수 있습니다API.

주제

- 플러그인 통합
- 사용자 지정 통합
- Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정
- Amazon Inspector SBOM 생성기
- Amazon Inspector Dockerfile 검사
- Amazon Inspector Scan과 사용자 지정 CI/CD 파이프라인 통합 생성
- Amazon Inspector 사용 Jenkins 플러그인 비활성화
- Amazon Inspector 사용 TeamCity 플러그인 비활성화
- 에서 Amazon Inspector 사용 GitHub actions
- 사용 CodeCatalyst Amazon Inspector를 사용한 작업
- Amazon Inspector에서 CycloneDX 네임스페이스 사용

플러그인 통합

Amazon Inspector는 지원되는 CI/CD 솔루션을 위한 플러그인을 제공합니다. 각 마켓플레이스에서 이러한 플러그인을 설치한 다음 이를 사용하여 Amazon Inspector 스캔을 파이프라인의 빌드 단계로 추가할 수 있습니다. 플러그인 빌드 단계는 사용자가 제공한 이미지에서 Amazon Inspector SBOM 생성기를 실행한 다음 생성된 API에서 Amazon Inspector 스캔을 실행합니다SBOM.

다음은 Amazon Inspector CI/CD 통합이 플러그인을 통해 작동하는 방식에 대한 개요입니다.

1. Amazon Inspector Scan에 대한 액세스를 허용 AWS 계정 하도록 를 구성합니다API. 지침은 Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정 단원을 참조하십시오.
2. 마켓플레이스에서 Amazon Inspector 플러그인을 설치합니다.
3. Amazon Inspector SBOM Generator 바이너리를 설치하고 구성합니다. 지침은 Amazon Inspector SBOM 생성기 단원을 참조하십시오.
4. Amazon Inspector 스캔을 CI/CD 파이프라인의 빌드 단계로 추가하고 스캔을 구성합니다.
5. 빌드를 실행하면 플러그인이 컨테이너 이미지를 입력으로 가져온 다음 이미지에서 Amazon Inspector SBOM Generator를 실행하여 CycloneDX 호환됩니다SBOM.
6. 여기에서 플러그인은 생성된 를 Amazon Inspector Scan API 엔드포인트로 전송SBOM하여 각 SBOM 구성 요소의 취약성을 평가합니다.
7. Amazon Inspector Scan API 응답은 CSV, SBOM JSON및 HTML 형식으로 취약성 보고서로 변환됩니다. 보고서에는 Amazon Inspector에서 발견한 모든 취약성에 대한 세부 정보가 포함되어 있습니다.

지원되는 CI/CD 솔루션

Amazon Inspector는 현재 다음과 같은 CI/CD 솔루션을 지원합니다. 플러그인을 사용하여 CI/CD 통합을 설정하는 방법에 대한 전체 지침을 보려면 CI/CD 솔루션용 플러그인을 선택하세요.

- Jenkins 플러그인
- TeamCity 플러그인 비활성화
- GitHub 작업

사용자 지정 통합

Amazon Inspector가 CI/CD 솔루션용 플러그인을 제공하지 않는 경우 Amazon Inspector SBOM Generator와 Amazon Inspector Amazon Inspector Scan을 조합하여 사용자 지정 CI/CD 통합을 생성할 수 있습니다API. 사용자 지정 통합을 사용하여 Amazon Inspector SBOM Generator를 통해 사용할 수 있는 옵션을 사용하여 스캔을 미세 조정할 수도 있습니다.

다음은 사용자 지정 Amazon Inspector CI/CD 통합이 작동하는 방식에 대한 개요입니다.

1. Amazon Inspector Scan에 대한 액세스를 허용 AWS 계정 하도록 를 구성합니다API. 지침은 Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정 단원을 참조하십시오.

2. Amazon Inspector SBOM Generator 바이너리를 설치하고 구성합니다. 지침은 Amazon Inspector SBOM 생성기 단원을 참조하십시오.
3. Amazon Inspector SBOM Generator를 사용하여 CycloneDX 컨테이너 이미지SBOM와 호환됩니다.
4. 생성된 API 에서 Amazon Inspector ScanSBOM을 사용하여 취약성 보고서를 생성합니다.

사용자 지정 통합을 설정하는 방법에 대한 지침은 Amazon Inspector Scan과 사용자 지정 CI/CD 파이프라인 통합 생성 섹션을 참조하세요.

Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정

Amazon Inspector CI/CD 통합을 사용하려면 에 가입해야 합니다 AWS 계정. 에는 Amazon Inspector Scan 에 대한 CI/CD pipleline 액세스 권한을 부여하는 IAM 역할이 있어야 AWS 계정 합니다API. 다음 주제의 작업을 완료하여 에 가입하고 AWS 계정, 관리자 사용자를 생성하고, CI/CD 통합을 위한 IAM 역할을 구성합니다.

> **Note**
> 이미 에 가입한 경우 로 건너뛸 AWS 계정수 있습니다CI/CD 통합을 위한 IAM 역할 구성.

주제

- 에 가입 AWS 계정
- 관리자 액세스 권한이 있는 사용자 생성
- CI/CD 통합을 위한 IAM 역할 구성

에 가입 AWS 계정

가 없는 경우 다음 단계를 AWS 계정완료하여 를 생성합니다.

에 가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/가입 을 엽니다.
2. 온라인 지시 사항을 따릅니다.

   등록 절차 중 전화를 받고 전화 키패드로 확인 코드를 입력하는 과정이 있습니다.

에 가입하면 AWS 계정AWS 계정 루트 사용자가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스에 액세스할 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 는 가입 프로세스가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/로 이동하여 내 계정 을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

에 가입한 후 일상적인 작업에 루트 사용자를 사용하지 않도록 AWS 계정를 보호하고, 를 AWS 계정 루트 사용자활성화하고 AWS IAM Identity Center, 관리 사용자를 생성합니다.

보안 AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 소유자AWS Management Console로 에 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면 AWS 로그인 User Guide의 루트 사용자로 로그인을 참조하십시오.
2. 루트 사용자에 대해 다중 인증(MFA)을 켭니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자(콘솔)에 대한 가상 MFA 디바이스 활성화를 참조하세요.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

   를 자격 증명 소스 IAM Identity Center 디렉터리 로 사용하는 방법에 대한 자습서는 AWS IAM Identity Center 사용 설명서의 기본값으로 사용자 액세스 구성을 IAM Identity Center 디렉터리 참조하세요.

관리 액세스 권한이 있는 사용자로 로그인

- IAM Identity Center 사용자로 로그인하려면 IAM Identity Center 사용자를 생성할 때 이메일 주소로 전송URL된 로그인을 사용합니다.

  IAM Identity Center 사용자를 사용하여 로그인하는 방법에 대한 자세한 내용은 AWS 로그인 사용 설명서의 AWS 액세스 포털에 로그인을 참조하세요.

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

CI/CD 통합을 위한 IAM 역할 구성

Amazon Inspector 스캔을 CI/CD 파이프라인에 통합하려면 소프트웨어 자재 명세서()를 API 스캔하는 Amazon Inspector 스캔에 대한 액세스를 허용하는 IAM 정책을 생성해야 합니다SBOMs. 그런 다음 Amazon Inspector 스캔 을 실행하기 위해 계정이 수임할 수 있는 IAM 역할에 해당 정책을 연결할 수 있습니다API.

1. 에 로그인 AWS Management Console 하고 에서 IAM 콘솔을 엽니다https://console.aws.amazon.com/iam/.
2. IAM 콘솔의 탐색 창에서 정책을 선택한 다음 정책 생성을 선택합니다.
3. 정책 편집기에서 다음 문을 선택하여 JSON 붙여넣습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "inspector-scan:ScanSbom",
            "Resource": "*"
        }
```

```
    ]
}
```

4. Next(다음)를 선택합니다.
5. 정책 이름(예: `InspectorCICDscan-policy`)과 선택 사항인 설명을 입력한 다음 정책 생성을 선택합니다. 이 정책은 다음 단계에서 생성할 역할에 연결됩니다.
6. IAM 콘솔의 탐색 창에서 역할을 선택한 다음 새 역할 생성 을 선택합니다.
7. 신뢰할 수 있는 엔터티 유형에서 사용자 지정 신뢰 정책을 선택한 후 다음 정책을 입력합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{ACCOUNT_ID}:root"
            },
            "Action": "sts:AssumeRole",
            "Condition": {}
        }
    ]
}
```

8. Next(다음)를 선택합니다.
9. 권한 추가에서 앞서 생성한 정책을 검색하여 선택하고 다음을 선택합니다.
10. 역할 이름(예: `InspectorCICDscan-role`)과 선택 사항인 설명을 입력한 다음 `Create Role`을 선택합니다.

Amazon Inspector SBOM 생성기

Software Bill of Materials(SBOM)는 소프트웨어 구성 요소를 구축하는 데 필요한 구성 요소, 라이브러리 및 모듈의 공식 구조화된 목록입니다. Amazon Inspector SBOM 생성기(Sbomgen)는 아카이브, 컨테이너 이미지, 디렉터리, 로컬 시스템 및 컴파일을 SBOM 위한 를 생성하는 도구입니다.Go 그리고 Rust 바이너리.Sbomgen 는 설치된 패키지에 대한 정보가 포함된 파일을 검색합니다. 일시 Sbomgen 는 관련 파일을 찾고 패키지 이름, 버전 및 기타 메타데이터를 추출합니다.Sbomgen 그런 다음 패키지 메타데이터를 로 변환합니다.CycloneDX SBOM. 다음을 사용할 수 있음: Sbomgen 를 생성하여 CycloneDX SBOM 취약성 감지를 SBOMs 위해 또는 의 파일로 Amazon Inspector에 STDOUT 전송합

니다. 를 사용할 수도 있습니다.Sbomgen 배포 파이프라인의 일부로 컨테이너 이미지를 자동으로 스캔하는 CI/CD 통합의 일부입니다.

지원되는 패키지 유형

Sbomgen 는 다음 패키지 유형에 대한 인벤토리를 수집합니다.

- Alpine APK
- Debian/Ubuntu DPKG
- Red Hat RPM
- C#
- Go
- Java
- Node.js
- PHP
- Python
- Ruby
- Rust

지원되는 컨테이너 이미지 구성 확인

Sbomgen 는 독립 실행형 Dockerfiles를 스캔하고 보안 문제에 대한 기존 이미지에서 기록을 빌드할 수 있습니다. 자세한 내용은 Amazon Inspector Dockerfile 검사 단원을 참조하세요.

설치 Sbomgen

Sbomgen 는 Linux 운영 체제에서만 사용할 수 있습니다.

...이(가) 있어야 합니다 Docker 원하는 경우 설치됨 Sbomgen 로컬에서 캐시된 이미지를 분석합니다.Docker 는 원격 컨테이너 레지스트리에서 호스팅되는 `.tar` 파일 또는 이미지로 내보낸 이미지를 분석할 필요가 없습니다.

Amazon Inspector는 실행을 권장합니다.Sbomgen 최소한 다음 하드웨어 사양이 있는 시스템에서

- 4x 코어 CPU
- 8GB RAM

설치 Sbomgen

1. 최신 다운로드 Sbomgen 아키텍처에 URL 맞는 의 zip 파일:

   Linux AMD64: https://amazon-inspector-sbomgen.s3.amazonaws.com/latest/linux/amd64/inspector-sbomgen.zip

   Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/latest/linux/arm64/inspector-sbomgen.zip

   또는 Amazon Inspector SBOM Generator zip 파일 의 이전 버전을 다운로드할 수 있습니다.
2. 다음 명령을 사용하여 다운로드를 압축 해제합니다.

   ```
   unzip inspector-sbomgen.zip
   ```
3. 추출된 디렉터리에서 다음 파일을 확인합니다.

   - `inspector-sbomgen` - 를 생성하기 위해 실행할 도구입니다SBOMs.
   - `README.txt` - 를 사용하기 위한 설명서입니다.Sbomgen.
   - `LICENSE.txt` - 이 파일에는 에 대한 소프트웨어 라이선스가 포함되어 있습니다.Sbomgen.
   - `licenses` - 이 폴더에는 에서 사용하는 타사 패키지에 대한 라이선스 정보가 포함되어 있습니다.Sbomgen.
   - `checksums.txt` - 이 파일은 해시를 제공합니다.Sbomgen 도구.
   - `sbom.json` - 이것은 CycloneDX SBOM 용 Sbomgen 도구.
   - `WhatsNew.txt` - 이 파일에는 요약된 변경 로그가 포함되어 있으므로 Sbomgen 버전은 빠릅니다.
4. (선택 사항) 다음 명령을 사용하여 도구의 신뢰성과 무결성을 확인합니다.

   ```
   sha256sum < inspector-sbomgen
   ```

   - 결과를 `checksums.txt` 파일 콘텐츠와 비교하세요.
5. 다음 명령을 사용하여 도구에 실행 가능 권한을 부여합니다.

   ```
   chmod +x inspector-sbomgen
   ```
6. 확인 Sbomgen 는 다음 명령을 사용하여 성공적으로 설치됩니다.

   ```
   ./inspector-sbomgen --version
   ```

   다음과 유사한 출력 화면이 표시되어야 합니다.

```
Version: 1.X.X
```

사용 Sbomgen

이 섹션에서는 다양한 사용 방법을 설명합니다.Sbomgen. 사용 방법에 대해 자세히 알아볼 수 있습니다.Sbomgen 기본 제공 예제를 통해 다음 예제를 보려면 `list-examples` 명령을 실행합니다.

```
./inspector-sbomgen list-examples
```

컨테이너 이미지에 SBOM 대한 생성 및 결과 출력

다음을 사용할 수 있음: Sbomgen 컨테이너 이미지에 SBOMs 대해 를 생성하고 결과를 파일로 출력합니다. 이 기능은 `container` 하위 명령을 사용하여 활성화할 수 있습니다.

명령 예:

다음 코드 조각에서 를 교체할 수 있습니다.*###:##* 이미지의 ID와 *output_path.json* 저장하려는 출력 경로가 있습니다.

```
# generate SBOM for container image
./inspector-sbomgen container --image image:tag -o output_path.json
```

> **Note**
>
> 스캔 시간과 성능은 이미지 크기와 계층 수가 얼마나 적은지에 따라 달라집니다. 작은 이미지는 개선될 뿐만 아니라 Sbomgen 잠재적 공격 표면을 줄일 수 있습니다. 작은 이미지는 이미지 빌드, 다운로드 및 업로드 시간도 개선합니다.

사용 시 Sbomgen ScanSbom를 사용하면 Amazon Inspector ScanAPI은 2,000개 이상의 패키지를 포함하는 SBOMs를 처리하지 않습니다. 이 시나리오에서 Amazon Inspector Scan은 HTTP 400 응답을 API 반환합니다.

이미지에 대용량 미디어 파일 또는 디렉터리가 포함된 경우 에서 제외하는 것이 좋습니다.Sbomgen `--skip-files` 인수를 사용합니다.

디렉터리 및 아카이브SBOM에서 생성

다음을 사용할 수 있음: Sbomgen 디렉터리 및 아카이브SBOMs에서 를 생성합니다. 이 기능은 `directory` 또는 `archive` 하위 명령을 사용하여 활성화할 수 있습니다. Amazon Inspector는 다운로드한 git 리포지토리와 같은 프로젝트 폴더SBOM에서 를 생성하려는 경우 이 기능을 사용하는 것이 좋습니다.

예제 명령 1

다음 조각은 디렉터리 파일SBOM에서 를 생성하는 하위 명령을 보여줍니다.

```
# generate SBOM from directory
./inspector-sbomgen directory --path /path/to/dir -o /tmp/sbom.json
```

예제 명령 2

다음 조각은 아카이브 파일SBOM에서 를 생성하는 하위 명령을 보여줍니다. 지원되는 유일한 아카이브 형식은 `.zip`, 및 `.tar`입니다`.tar.gz`.

```
# generate SBOM from archive file (tar, tar.gz, and zip formats only)
./inspector-sbomgen archive --path testData.zip -o /tmp/sbom.json
```

SBOM 에서 생성 Go 또는 Rust 컴파일된 바이너리

다음을 사용할 수 있음: Sbomgen 컴파일된 SBOMs에서 생성 Go 그리고 Rust 바이너리. `binary` 하위 명령을 통해 이 기능을 활성화할 수 있습니다.

```
./inspector-sbomgen binary --path /path/to/your/binary
```

취약성 식별SBOM을 위해 Amazon Inspector로 전송

를 생성하는 것 외에도 Amazon Inspector Scan 에서 단일 명령으로 스캔SBOM할 를 보낼 SBOM수 있습니다API. Amazon Inspector는 결과를 로 반환하기 전에 취약성에 SBOM 대한 의 내용을 평가합니다.Sbomgen. 입력에 따라 조사 결과를 파일에 표시하거나 쓸 수 있습니다.

> **Note**
>
> 이 기능을 사용하려면 `InspectorScan:ScanSbom`에 대한 읽기 권한이 AWS 계정 있는 활성가 있어야 합니다.

이 기능을 활성화하려면 `--scan-sbom` 인수를 에 전달합니다.Sbomgen CLI. `--scan-sbom` 인수를 다음 중 하나에 전달할 수도 있습니다.Sbomgen 하위 명령: `archive`, `binary`, `container`, `directory`, `localhost`.

> **Note**
>
> Amazon Inspector Scan은 2,000개 이상의 패키지SBOMs로 처리되지 API 않습니다. 이 시나리오에서 Amazon Inspector Scan은 HTTP 400 응답을 API 반환합니다.

다음 AWS CLI 인수를 사용하여 AWS 프로필 또는 IAM 역할을 통해 Amazon Inspector에 인증할 수 있습니다.

```
--aws-profile profile
--aws-region region
--aws-iam-role-arn role_arn
```

다음 환경 변수를 에 제공하여 Amazon Inspector에 인증할 수도 있습니다.Sbomgen.

```
AWS_ACCESS_KEY_ID=$access_key \
AWS_SECRET_ACCESS_KEY=$secret_key \
AWS_DEFAULT_REGION=$region \
./inspector-sbomgen arguments
```

응답 형식을 지정하려면 `--scan-sbom-output-format cyclonedx` 인수 또는 `--scan-sbom-output-format inspector` 인수를 사용합니다.

예제 명령 1

이 명령은 SBOM 최신 Alpine Linux 릴리스하고, 를 스캔하고SBOM, 취약성 결과를 JSON 파일에 기록합니다.

```
./inspector-sbomgen container --image alpine:latest \
                          --scan-sbom \
                          --aws-profile your_profile \
                          --aws-region your_region \
```

```
                            --scan-sbom-output-format cyclonedx \
                            --outfile /tmp/inspector_scan.json
```

예제 명령 2

이 명령은 AWS 자격 증명을 환경 변수로 사용하여 Amazon Inspector에 인증합니다.

```
AWS_ACCESS_KEY_ID=$your_access_key \
AWS_SECRET_ACCESS_KEY=$your_secret_key \
AWS_DEFAULT_REGION=$your_region \
./inspector-sbomgen container --image alpine:latest \
                            -o /tmp/sbom.json \
                            --scan-sbom \
                            --scan-sbom-output-format inspector
```

예제 명령 3

이 명령은 IAM 역할에 ARN 대해 를 사용하여 Amazon Inspector에 인증합니다.

```
./inspector-sbomgen container --image alpine:latest \
                            --scan-sbom \
                            --aws-profile your_profile \
                            --aws-region your_region \
                            --outfile /tmp/inspector_scan.json
                            --aws-iam-role-arn arn:aws:iam::123456789012:role/your_role
```

특정 파일을 제외하도록 스캔 사용자 지정

컨테이너 이미지를 분석하고 처리할 때 Sbomgen 는 해당 컨테이너 이미지의 모든 파일 크기를 스캔합니다. 스캔을 사용자 지정하여 특정 파일을 제외하거나 특정 패키지를 대상으로 지정할 수 있습니다.

디스크 소비, RAM 소비, 경과된 런타임을 줄이고 제공된 임계값을 초과하는 파일을 건너뛰려면 `container` 하위 명령과 함께 `--max-file-size` 인수를 사용합니다.

```
./inspector-sbomgen container --image alpine:latest \
--outfile /tmp/sbom.json \
--max-file-size 300000000
```

진행률 표시기 비활성화

Sbomgen 는 CI/CD 환경에서 과도한 슬래시 문자가 발생할 수 있는 회전 진행률 표시기를 표시합니다.

```
INFO[2024-02-01 14:58:46]coreV1.go:53: analyzing artifact
|
\
/
|
\
/
INFO[2024-02-01 14:58:46]coreV1.go:62: executing post-processors
```

`--disable-progress-bar` 주장을 사용하여 진행률 표시기를 비활성화할 수 있습니다.

```
./inspector-sbomgen container --image alpine:latest \
--outfile /tmp/sbom.json \
--disable-progress-bar
```

를 사용하여 프라이빗 레지스트리에 인증 Sbomgen

프라이빗 레지스트리 인증 자격 증명을 제공하여 프라이빗 레지스트리에서 호스팅되는 컨테이너 SBOMs에서 를 생성할 수 있습니다. 다음 방법을 통해 이러한 자격 증명을 제공할 수 있습니다.

캐시된 보안 인증 정보를 사용하여 인증(권장)

이 방법의 경우 컨테이너 레지스트리에 인증합니다. 예를 들어 를 사용하는 경우 Docker에서 를 사용하여 컨테이너 레지스트리에 인증할 수 있습니다.Docker 로깅 명령: `docker login`.

1. 컨테이너 레지스트리에 인증합니다. 예를 들어 를 사용하는 경우 Docker에서 를 사용하여 레지스트리에 인증할 수 있습니다.Docker `login` 명령:
2. 컨테이너 레지스트리에 인증한 후 를 사용합니다.Sbomgen 레지스트리에 있는 컨테이너 이미지에서. 다음 예제를 사용하려면 를 바꿉니다.*###:##* 스캔할 이미지의 이름:

```
./inspector-sbomgen container --image image:tag
```

비활성 메서드를 사용하여 인증

이 메서드의 경우 사용자 이름을 파라미터로 제공하고 Sbomgen 는 필요한 경우 보안 암호 입력을 요청합니다.

다음 예제를 사용하려면 를 바꿉니다.*###:##* 스캔하려는 이미지의 이름과 *### ##* 이미지에 액세스할 수 있는 사용자 이름:

```
./inspector-sbomgen container --image image:tag --username your_username
```

대화형이 아닌 메서드를 사용하여 인증

이 방법의 경우 암호 또는 레지스트리 토큰을 `.txt` 파일에 저장합니다.

> **Note**
> 현재 사용자는 이 파일만 읽을 수 있어야 합니다. 파일에는 한 줄에 암호 또는 토큰도 포함되어야 합니다.

다음 예제를 사용하려면 를 바꿉니다.*### ##* 사용자 이름을 사용하여 *password.txt* 한 줄에 암호 또는 토큰이 포함된 `.txt` 파일 *###:##* 스캔할 이미지의 이름:

```
INSPECTOR_SBOMGEN_USERNAME=your_username \
INSPECTOR_SBOMGEN_PASSWORD=`cat password.txt` \
./inspector-sbomgen container --image image:tag
```

의 출력 예제 Sbomgen

다음은 를 사용하여 인벤토리된 컨테이너 이미지에 SBOM 대한 의 예입니다.Sbomgen.

컨테이너 이미지 SBOM

```
{
  "bomFormat": "CycloneDX",
  "specVersion": "1.5",
  "serialNumber": "urn:uuid:828875ef-8c32-4777-b688-0af96f3cf619",
  "version": 1,
  "metadata": {
    "timestamp": "2023-11-17T21:36:38Z",
    "tools": [
      {
        "vendor": "Amazon Web Services, Inc. (AWS)",
        "name": "Amazon Inspector SBOM Generator",
        "version": "1.0.0",
        "hashes": [
          {
            "alg": "SHA-256",
```

```
            "content":
"10ab669cfc99774786301a745165b5957c92ed9562d19972fbf344d4393b5eb1"
          }
        ]
      }
    ],
    "component": {
      "bom-ref": "comp-1",
      "type": "container",
      "name": "fedora:latest",
      "properties": [
        {
          "name": "amazon:inspector:sbom_generator:image_id",
          "value":
"sha256:c81c8ae4dda7dedc0711daefe4076d33a88a69a28c398688090c1141eff17e50"
        },
        {
          "name": "amazon:inspector:sbom_generator:layer_diff_id",
          "value":
"sha256:eddd0d48c295dc168d0710f70364581bd84b1dda6bb386c4a4de0b61de2f2119"
        }
      ]
    }
  },
  "components": [
    {
      "bom-ref": "comp-2",
      "type": "library",
      "name": "dnf",
      "version": "4.18.0",
      "purl": "pkg:pypi/dnf@4.18.0",
      "properties": [
        {
          "name": "amazon:inspector:sbom_generator:source_file_scanner",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_package_collector",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_path",
          "value": "/usr/lib/python3.12/site-packages/dnf-4.18.0.dist-info/METADATA"
        },
```

```
        {
          "name": "amazon:inspector:sbom_generator:is_duplicate_package",
          "value": "true"
        },
        {
          "name": "amazon:inspector:sbom_generator:duplicate_purl",
          "value": "pkg:rpm/fedora/python3-dnf@4.18.0-2.fc39?
arch=noarch&distro=39&epoch=0"
        }
      ]
    },
    {
      "bom-ref": "comp-3",
      "type": "library",
      "name": "libcomps",
      "version": "0.1.20",
      "purl": "pkg:pypi/libcomps@0.1.20",
      "properties": [
        {
          "name": "amazon:inspector:sbom_generator:source_file_scanner",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_package_collector",
          "value": "python-pkg"
        },
        {
          "name": "amazon:inspector:sbom_generator:source_path",
          "value": "/usr/lib64/python3.12/site-packages/libcomps-0.1.20-py3.12.egg-
info/PKG-INFO"
        },
        {
          "name": "amazon:inspector:sbom_generator:is_duplicate_package",
          "value": "true"
        },
        {
          "name": "amazon:inspector:sbom_generator:duplicate_purl",
          "value": "pkg:rpm/fedora/python3-libcomps@0.1.20-1.fc39?
arch=x86_64&distro=39&epoch=0"
        }
      ]
    }
  ]
```

```
}
```

Amazon SBOM 인스펙터 제너레이터의 이전 버전

이 주제에서는 Amazon Inspector SBOM Generator의 이전 버전에 대한 링크를 제공합니다. Sbomgen 설치에 대한 자세한 내용은 Sbomgen 설치를 참조하세요.

Sbomgen1.4.0

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.4.0/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.4.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.3.2

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.2/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.2/linux/arm64/inspector-sbomgen.zip

Sbomgen1.3.1

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.1/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.3.0

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.0/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.2.1

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.1/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.2.0

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.0/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.1.1

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.1/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.1.0

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.0/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.0.0

- 리눅스AMD64: linux/amd64/inspector-sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.0.0/
- 리눅스ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.0.0/linux/arm64/inspector-sbomgen.zip

Amazon Inspector Dockerfile 검사

이 섹션에서는 Amazon Inspector SBOM Generator를 사용하여 스캔하는 방법을 설명합니다.Dockerfiles 그리고 Docker 컨테이너 이미지에 보안 취약성이 발생하는 잘못된 구성이 있습니다.

주제

- 사용 Sbomgen Dockerfile 검사
- 지원되는 Dockerfile 검사

사용 Sbomgen Dockerfile 검사

Dockerfile 검사는 이름이 `Dockerfile` 또는 인 파일이 검색`*.Dockerfile`되고 Docker 이미지가 스캔될 때 자동으로 수행됩니다.

`--skip-scanners dockerfile` 인수를 사용하여 Dockerfile 검사를 비활성화할 수 있습니다. 또한 Dockerfile 검사를 OS 또는 타사 패키지와 같은 사용 가능한 모든 스캐너와 결합할 수 있습니다.

도커 검사 명령의 예

다음 예제 명령은 도커파일 및 도커 컨테이너 이미지와 OS 및 타사 패키지SBOMs에 대해 를 생성하는 방법을 보여줍니다.

```
# generate SBOM only containing Docker checks for Dockerfiles in a local directory
./inspector-sbomgen directory --path ./project/ --scanners dockerfile

# generate SBOM for container image will by default include Dockerfile checks
./inspector-sbomgen container --image image:tag

# generate SBOM only containing Docker checks for specific Dockerfiles and Alpine,
 Debian, and Rhel OS packages in a local directory
/inspector-sbomgen directory --path ./project/ --scanners dockerfile,dpkg,alpine-
apk,rhel-rpm

# generate SBOM only containing Docker checks for specific Dockerfiles in a local
 directory
./inspector-sbomgen directory --path ./project/ --skip-scanners dockerfile
```

예제 파일 구성 요소

다음은 파일 구성 요소에 대한 Dockerfile 결과의 예입니다.

```
{
      "bom-ref": "comp-2",
      "name": "dockerfile:data/docker/Dockerfile",
      "properties": [
        {
          "name": "amazon:inspector:sbom_scanner:dockerfile_finding:IN-DOCKER-001",
          "value": "affected_lines:27-27"
        }
      ],
      "type": "file"
    },
```

취약성 응답 구성 요소 예

다음은 취약성 응답 구성 요소에 대한 Dockerfile 조사 결과의 예입니다.

```
{
      "advisories": [
        {
          "url": "https://docs.docker.com/develop/develop-images/instructions/"
        }
      ],
      "affects": [
        {
          "ref": "comp-2"
        }
      ],
      "analysis": {
        "state": "in_triage"
      },
      "bom-ref": "vuln-13",
      "created": "2024-03-27T14:36:39Z",
      "description": "apt-get layer caching: Using apt-get update alone in a RUN
 statement causes caching issues and subsequent apt-get install instructions to fail.",
      "id": "IN-DOCKER-001",
      "ratings": [
        {
          "method": "other",
          "severity": "info",
          "source": {
            "name": "AMAZON_INSPECTOR",
            "url": "https://aws.amazon.com/inspector/"
          }
```

```
        }
      ],
      "source": {
        "name": "AMAZON_INSPECTOR",
        "url": "https://aws.amazon.com/inspector/"
      },
      "updated": "2024-03-27T14:36:39Z"
    },
```

> **Note**
>
> 호출하는 경우 Sbomgen `--scan-sbom` 플래그가 없으면 원시 Dockerfile 조사 결과만 볼 수 있습니다.

지원되는 Dockerfile 검사

Sbomgen Dockerfile 검사는 다음에 대해 지원됩니다.

- Sudo 바이너리 패키지
- Debian APT 유틸리티
- 하드코딩된 보안 암호
- 루트 컨테이너
- 런타임 약화 명령 플래그
- 런타임 약화 환경 변수

이러한 각 Dockerfile 검사에는 해당 심각도 등급이 있으며, 이는 다음 주제 상단에 나와 있습니다.

> **Note**
>
> 다음 주제에 설명된 권장 사항은 업계 모범 사례를 기반으로 합니다.

Sudo 바이너리 패키지

> **Note**
> 이 확인의 심각도 등급은 정보 입니다.

Sudo 바이너리 패키지는 예측할 수 없고 TTY 신호 전달 동작이 있으므로 설치하거나 사용하지 않는 것이 좋습니다. 자세한 내용은 Docker Docs 웹 사이트의 사용자를 참조하세요. 사용 사례에 Sudo 바이너리 패키지와 유사한 기능이 필요한 경우 Gosu 를 사용하는 것이 좋습니다.

Debian APT 유틸리티

> **Note**
> 이 검사의 심각도 등급은 높음 입니다.

다음은 를 사용하는 모범 사례입니다.Debian APT 유틸리티.

캐싱 문제를 방지하기 위해 단일 **Run** 문에서 **`apt-get`** 명령 결합

Docker 컨테이너 내부의 단일 RUN 문에서 `apt-get` 명령을 결합하는 것이 좋습니다. 자체`apt-get update`적으로 사용하면 캐싱 문제와 후속 `apt-get install` 실패 지침이 발생합니다. 자세한 내용은 Docker Docs 웹 사이트의 apt-get을 참조하세요.

> **Note**
> 설명된 캐싱 동작은 Docker Docker 컨테이너 소프트웨어가 오래된 경우 컨테이너.

상호 작용하지 않는 방식으로 APT 명령줄 유틸리티 사용

APT 명령줄 유틸리티를 대화형으로 사용하는 것이 좋습니다. APT 명령줄 유틸리티는 최종 사용자 도구로 설계되었으며 버전 간에 동작이 변경됩니다. 자세한 내용은 Debian 웹 사이트의 스크립트 사용량 및 다른 APT 도구와의 차이점을 참조하세요.

하드 코딩된 보안 암호

> **Note**
> 이 검사의 심각도 등급은 중요 입니다.

Dockerfile의 기밀 정보는 하드 코딩된 보안 암호로 간주됩니다. 다음 하드 코딩된 보안 암호는 를 통해 식별할 수 있습니다.Sbomgen Docker 파일 확인:

- AWS 액세스 키 IDs - `AKIAIOSFODNN7EXAMPLE`
- DockerHub 개인 액세스 토큰 - `dckr_pat_thisisa27charexample1234567`
- GitHub 개인 액세스 토큰 - `ghp_examplev61wY7Pj1YnotrealUoY123456789`
- GitLab 개인 액세스 토큰 - `glpat-12345example12345678`

루트 컨테이너

> **Note**
> 이 확인의 심각도 마커는 정보 입니다.

루트 권한 없이 Docker 컨테이너를 실행하는 것이 좋습니다. 루트 권한 없이 실행할 수 없는 컨테이너화된 워크로드의 경우 권한이 가장 적은 원칙을 사용하여 애플리케이션을 구축하는 것이 좋습니다. 자세한 내용은 Docker Docs 웹 사이트의 사용자를 참조하세요.

런타임 약화 환경 변수

> **Note**
> 이 검사의 심각도 등급은 높음 입니다.

여러 명령줄 유틸리티 또는 프로그래밍 언어 런타임은 보안 기본값을 우회하도록 지원하므로 안전하지 않은 방법을 통해 실행할 수 있습니다.

NODE_TLS_REJECT_UNAUTHORIZED=0

일시 Node.js 프로세스가 로 `NODE_TLS_REJECT_UNAUTHORIZED` 설정된 상태로 실행0되고 TLS 인증서 검증이 비활성화됩니다. 자세한 내용은 Node.js 웹 사이트의 NODE_TLS_REJECT_UNAUTHORIZED=0을 참조하세요.

GIT_SSL_NO_VERIFY=*

git 명령줄 프로세스가 `GIT_SSL_NO_VERIFY` 세트로 실행되면 Git은 TLS 인증서 확인을 건너뜁니다. 자세한 내용은 Git 웹 사이트의 환경 변수를 참조하세요.

PIP_TRUSTED_HOST=*

일시 Python pip 명령줄 프로세스는 `PIP_TRUSTED_HOST` 세트로 실행되며, Pip는 지정된 도메인에서 TLS 인증서 확인을 건너뜁니다. 자세한 내용은 Pip 웹 사이트의 --trusted-host를 참조하세요.

NPM_CONFIG_STRICT_SSL=거짓

일시 Node.js npm 명령줄 프로세스는 false로 `NPM_CONFIG_STRICT_SSL` 설정하여 실행되며, 노드 패키지 관리자(npm) 유틸리티는 TLS 인증서를 검증하지 않고 NPM 레지스트리에 연결됩니다. 자세한 내용은 npm Docs 웹 사이트의 strict-ssl을 참조하세요.

런타임 약화 명령 플래그

> **Note**
>
> 이 검사의 심각도 등급은 높음 입니다.

런타임 약화 환경 변수와 마찬가지로 여러 명령줄 유틸리티 또는 프로그래밍 언어 런타임은 보안 기본값을 우회하여 안전하지 않은 방법을 통해 실행할 수 있도록 지원합니다.

`npm --strict-ssl=false`

Node.js npm 명령줄 프로세스가 `--strict-ssl=false` 플래그로 실행되면 노드 패키지 관리자(npm) 유틸리티는 TLS 인증서를 검증하지 않고 NPM 레지스트리에 연결됩니다. 자세한 내용은 npm Docs 웹 사이트의 strict-ssl을 참조하세요.

`apk --allow-untrusted`

가 Alpine Package Keeper 유틸리티는 `--allow-untrusted` 플래그와 함께 실행apk되며 서명이 없거나 신뢰할 수 없는 패키지를 설치합니다. 자세한 내용은 Apline 웹 사이트의 다음 리포지토리를 참조하세요.

`apt-get --allow-unauthenticated`

`--allow-unauthenticated` 플래그로 Debian `apt-get` 패키지 유틸리티를 실행할 때 는 패키지 유효성을 확인하지 `apt-get` 않습니다. 자세한 내용은 Debian 웹 사이트의 APT-Get(8)을 참조하세요.

`pip --trusted-host`

가 Python pip 유틸리티는 `--trusted-host` 플래그로 실행되며 지정된 호스트 이름은 TLS 인증서 검증을 우회합니다. 자세한 내용은 Pip 웹 사이트의 --trusted-host를 참조하세요.

`rpm --nodigest, --nosignature, --noverify, --nofiledigest`

`--nodigest`, `--nosignature--noverify`, 및 `--nofiledigest` 플래그로 RPM기반 패키지 관리자를 `rpm` 실행하면 RPM 패키지 설치 시 패키지 관리자가 패키지 헤더, 서명 또는 파일을 검증하지 않습니다. 자세한 내용은 RPM 웹 사이트의 다음 RPM 설명서 페이지를 참조하세요.

`yum-config-manager --setopt=sslverify false`

`--setopt=sslverify` 플래그가 false로 설정된 상태에서 RPM기반 패키지 관리자가 `yum-config-manager` 실행되면 YUM 패키지 관리자가 TLS 인증서를 검증하지 않습니다. 자세한 내용은 Man7 웹 사이트의 다음 YUM 설명서 페이지를 참조하세요.

`yum --nogpgcheck`

RPM기반 패키지 관리자가 `--nogpgcheck` 플래그로 `yum` 실행되면 YUM 패키지 관리자는 패키지의 GPG 서명 확인을 건너뜁니다. 자세한 내용은 Man7 웹 사이트의 yum(8)을 참조하세요.

`curl --insecure, curl -k`

`curl` 가 `--insecure` 또는 `-k` 플래그로 실행되면 TLS 인증서 검증이 비활성화됩니다. 기본적으로 `curl`에서 수행하는 모든 보안 연결은 전송이 발생하기 전에 안전한 것으로 확인됩니다. 이 옵션을 사용하면 확인 단계를 `curl` 건너뛰고 확인하지 않고 진행합니다. 자세한 내용은 Curl 웹 사이트의 다음 Curl 설명서 페이지를 참조하세요.

`wget --no-check-certificate`

`wget` 가 `--no-check-certificate` 플래그로 실행되면 TLS 인증서 검증이 비활성화됩니다. 자세한 내용은 GNU 웹 사이트의 다음 Wget 설명서 페이지를 참조하세요.

Amazon Inspector Scan과 사용자 지정 CI/CD 파이프라인 통합 생성

CI/CD 솔루션에 Amazon Inspector CI/CD 플러그인을 사용할 수 있는 경우 Amazon Inspector CI/CD 플러그인을 사용하는 것이 좋습니다. CI/CD 솔루션에 Amazon Inspector CI/CD 플러그인을 사용할 수

없는 경우 Amazon Inspector SBOM Generator와 Amazon Inspector Scan을 조합하여 사용자 지정 CI/CD 통합을 API 생성할 수 있습니다. 다음 단계에서는 Amazon Inspector Scan과 사용자 지정 CI/CD 파이프라인 통합을 생성하는 방법을 설명합니다.

> **Tip**
>
> Amazon Inspector SBOM 생성기(Sbomgen)를 선택하여 단일 명령 SBOM에서 를 생성하고 스캔하려면 3단계와 4단계를 건너뜁니다.

단계 1. 구성 AWS 계정

Amazon Inspector Scan에 대한 액세스를 AWS 계정 제공하는 를 구성합니다API. 자세한 내용은 Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정 단원을 참조하십시오.

단계 2. 설치 Sbomgen 이진수

설치 및 구성 Sbomgen 바이너리. 자세한 내용은 설치를 참조하세요.Sbomgen.

단계 3. 사용 Sbomgen

를 사용합니다.Sbomgen 를 사용하여 스캔하려는 컨테이너 이미지의 SBOM 파일을 생성합니다.

다음 예제를 사용할 수 있습니다. Replace *###:id* 스캔할 이미지의 이름이 표시됩니다. Replace *sbom_path.json* SBOM 출력을 저장하려는 위치가 표시됩니다.

예

```
./inspector-sbomgen container --image image:id -o sbom_path.json
```

4단계. Amazon Inspector Scan 호출 API

를 호출`inspector-scan`API하여 생성된 를 스캔SBOM하고 취약성 보고서를 제공합니다.

다음 예제를 사용할 수 있습니다. Replace *sbom_path.json* 유효한 CycloneDX 호환 SBOM 파일의 위치를 사용합니다. Replace *ENDPOINT* 현재 인증된 AWS 리전 에 대한 API 엔드포인트를 사용합니다. Replace *REGION* 해당 리전과 함께.

예

```
aws inspector-scan scan-sbom --sbom file://sbom_path.json --endpoint
ENDPOINT-URL --region REGION
```

AWS 리전 및 엔드포인트의 전체 목록은 리전 및 엔드포인트 섹션을 참조하세요.

(선택 사항) 5단계. 단일 명령SBOM으로 생성 및 스캔

> **Note**
>
> 3단계와 4단계를 건너뛴 경우에만 이 단계를 완료합니다.

`--scan-bom` 플래그를 사용하여 단일 명령SBOM으로 를 생성하고 스캔합니다.

다음 예제를 사용할 수 있습니다. Replace *`###:id`* 스캔하려는 이미지의 이름이 표시됩니다. Replace *`profile`* 해당 프로필과 함께. Replace *`REGION`* 해당 리전과 함께. Replace *`/tmp/scan.json`* tmp 디렉터리에 scan.json 파일의 위치가 있습니다.

예

```
./inspector-sbomgen container --image image:id --scan-sbom --aws-profile
profile --aws-region REGION -o /tmp/scan.json
```

AWS 리전 및 엔드포인트의 전체 목록은 리전 및 엔드포인트 를 참조하세요.

API 출력 형식

Amazon Inspector Scan은 에서 취약성 보고서를 출력할 API 수 있습니다.CycloneDX 1.5 형식 또는 Amazon Inspector 결과JSON. `--output-format` 플래그를 사용하여 기본값을 변경할 수 있습니다.

의 예 CycloneDX 1.5 형식 출력

```
{
  "status": "SBOM parsed successfully, 1 vulnerabilities found",
  "sbom": {
    "bomFormat": "CycloneDX",
    "specVersion": "1.5",
    "serialNumber": "urn:uuid:0077b45b-ff1e-4dbb-8950-ded11d8242b1",
    "metadata": {
      "properties": [
        {
          "name": "amazon:inspector:sbom_scanner:critical_vulnerabilities",
          "value": "1"
        },
        {
          "name": "amazon:inspector:sbom_scanner:high_vulnerabilities",
```

```
          "value": "0"
        },
        {
          "name": "amazon:inspector:sbom_scanner:medium_vulnerabilities",
          "value": "0"
        },
        {
          "name": "amazon:inspector:sbom_scanner:low_vulnerabilities",
          "value": "0"
        }
      ],
      "tools": [
        {
          "name": "CycloneDX SBOM API",
          "vendor": "Amazon Inspector",
          "version": "empty:083c9b00:083c9b00:083c9b00"
        }
      ],
      "timestamp": "2023-06-28T14:15:53.760Z"
    },
    "components": [
      {
        "bom-ref": "comp-1",
        "type": "library",
        "name": "log4j-core",
        "purl": "pkg:maven/org.apache.logging.log4j/log4j-core@2.12.1",
        "properties": [
          {
            "name": "amazon:inspector:sbom_scanner:path",
            "value": "/home/dev/foo.jar"
          }
        ]
      }
    ],
    "vulnerabilities": [
      {
        "bom-ref": "vuln-1",
        "id": "CVE-2021-44228",
        "source": {
          "name": "NVD",
          "url": "https://nvd.nist.gov/vuln/detail/CVE-2021-44228"
        },
        "references": [
          {
```

```
            "id": "SNYK-JAVA-ORGAPACHELOGGINGLOG4J-2314720",
            "source": {
              "name": "SNYK",
              "url": "https://security.snyk.io/vuln/SNYK-JAVA-
ORGAPACHELOGGINGLOG4J-2314720"
            }
          },
          {
            "id": "GHSA-jfh8-c2jp-5v3q",
            "source": {
              "name": "GITHUB",
              "url": "https://github.com/advisories/GHSA-jfh8-c2jp-5v3q"
            }
          }
        ],
        "ratings": [
          {
            "source": {
              "name": "NVD",
              "url": "https://www.first.org/cvss/v3-1/"
            },
            "score": 10.0,
            "severity": "critical",
            "method": "CVSSv31",
            "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
          },
          {
            "source": {
              "name": "NVD",
              "url": "https://www.first.org/cvss/v2/"
            },
            "score": 9.3,
            "severity": "critical",
            "method": "CVSSv2",
            "vector": "AC:M/Au:N/C:C/I:C/A:C"
          },
          {
            "source": {
              "name": "EPSS",
              "url": "https://www.first.org/epss/"
            },
            "score": 0.97565,
            "severity": "none",
            "method": "other",
```

```
            "vector": "model:v2023.03.01,date:2023-06-27T00:00:00+0000"
          },
          {
            "source": {
              "name": "SNYK",
              "url": "https://security.snyk.io/vuln/SNYK-JAVA-
ORGAPACHELOGGINGLOG4J-2314720"
            },
            "score": 10.0,
            "severity": "critical",
            "method": "CVSSv31",
            "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:H"
          },
          {
            "source": {
              "name": "GITHUB",
              "url": "https://github.com/advisories/GHSA-jfh8-c2jp-5v3q"
            },
            "score": 10.0,
            "severity": "critical",
            "method": "CVSSv31",
            "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
          }
        ],
        "cwes": [
          400,
          20,
          502
        ],
        "description": "Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security
 releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages,
 and parameters do not protect against attacker controlled LDAP and other JNDI related
 endpoints. An attacker who can control log messages or log message parameters can
 execute arbitrary code loaded from LDAP servers when message lookup substitution is
 enabled. From log4j 2.15.0, this behavior has been disabled by default. From version
 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely
 removed. Note that this vulnerability is specific to log4j-core and does not affect
 log4net, log4cxx, or other Apache Logging Services projects.",
        "advisories": [
          {
            "url": "https://www.intel.com/content/www/us/en/security-center/advisory/
intel-sa-00646.html"
          },
          {
```

```
            "url": "https://support.apple.com/kb/HT213189"
          },
          {
            "url": "https://msrc-blog.microsoft.com/2021/12/11/microsofts-response-to-
cve-2021-44228-apache-log4j2/"
          },
          {
            "url": "https://logging.apache.org/log4j/2.x/security.html"
          },
          {
            "url": "https://www.debian.org/security/2021/dsa-5020"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf"
          },
          {
            "url": "https://www.oracle.com/security-alerts/alert-cve-2021-44228.html"
          },
          {
            "url": "https://www.oracle.com/security-alerts/cpujan2022.html"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf"
          },
          {
            "url": "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/M5CSVUNV4HWZZXGOKNSK6L7RPM7BOKIB/"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf"
          },
          {
            "url": "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/VU57UJDCFIASIO35GC55JMKSRXJMCDFM/"
          },
          {
            "url": "https://www.oracle.com/security-alerts/cpuapr2022.html"
          },
          {
            "url": "https://twitter.com/kurtseifried/status/1469345530182455296"
          },
```

```
          {
            "url": "https://tools.cisco.com/security/center/content/
CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd"
          },
          {
            "url": "https://lists.debian.org/debian-lts-announce/2021/12/msg00007.html"
          },
          {
            "url": "https://www.kb.cert.org/vuls/id/930724"
          }
        ],
        "created": "2021-12-10T10:15:00Z",
        "updated": "2023-04-03T20:15:00Z",
        "affects": [
          {
            "ref": "comp-1"
          }
        ],
        "properties": [
          {
            "name": "amazon:inspector:sbom_scanner:exploit_available",
            "value": "true"
          },
          {
            "name": "amazon:inspector:sbom_scanner:exploit_last_seen_in_public",
            "value": "2023-03-06T00:00:00Z"
          },
          {
            "name": "amazon:inspector:sbom_scanner:cisa_kev_date_added",
            "value": "2021-12-10T00:00:00Z"
          },
          {
            "name": "amazon:inspector:sbom_scanner:cisa_kev_date_due",
            "value": "2021-12-24T00:00:00Z"
          },
          {
            "name": "amazon:inspector:sbom_scanner:fixed_version:comp-1",
            "value": "2.15.0"
          }
        ]
      }
    ]
  }
```

```
}
```

Inspector 형식 출력 예

```
                        {
  "status": "SBOM parsed successfully, 1 vulnerability found",
  "inspector": {
    "messages": [
      {
        "name": "foo",
        "purl": "pkg:maven/foo@1.0.0", // Will not exist in output if missing in sbom
        "info": "Component skipped: no rules found."
      }
    ],
    "vulnerability_count": {
      "critical": 1,
      "high": 0,
      "medium": 0,
      "low": 0
    },
    "vulnerabilities": [
      {
        "id": "CVE-2021-44228",
        "severity": "critical",
        "source": "https://nvd.nist.gov/vuln/detail/CVE-2021-44228",
        "related": [
          "SNYK-JAVA-ORGAPACHELOGGINGLOG4J-2314720",
          "GHSA-jfh8-c2jp-5v3q"
        ],
        "description": "Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security
 releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages,
 and parameters do not protect against attacker controlled LDAP and other JNDI related
 endpoints. An attacker who can control log messages or log message parameters can
 execute arbitrary code loaded from LDAP servers when message lookup substitution is
 enabled. From log4j 2.15.0, this behavior has been disabled by default. From version
 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely
 removed. Note that this vulnerability is specific to log4j-core and does not affect
 log4net, log4cxx, or other Apache Logging Services projects.",
        "references": [
          "https://www.intel.com/content/www/us/en/security-center/advisory/intel-
sa-00646.html",
          "https://support.apple.com/kb/HT213189",
```

```
          "https://msrc-blog.microsoft.com/2021/12/11/microsofts-response-to-
cve-2021-44228-apache-log4j2/",
          "https://logging.apache.org/log4j/2.x/security.html",
          "https://www.debian.org/security/2021/dsa-5020",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf",
          "https://www.oracle.com/security-alerts/alert-cve-2021-44228.html",
          "https://www.oracle.com/security-alerts/cpujan2022.html",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf",
          "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/M5CSVUNV4HWZZXGOKNSK6L7RPM7BOKIB/",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf",
          "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf",
          "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/VU57UJDCFIASIO35GC55JMKSRXJMCDFM/",
          "https://www.oracle.com/security-alerts/cpuapr2022.html",
          "https://twitter.com/kurtseifried/status/1469345530182455296",
          "https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-
sa-apache-log4j-qRuKNEbd",
          "https://lists.debian.org/debian-lts-announce/2021/12/msg00007.html",
          "https://www.kb.cert.org/vuls/id/930724"
        ],
        "created": "2021-12-10T10:15:00Z",
        "updated": "2023-04-03T20:15:00Z",
        "properties": {
          "cisa_kev_date_added": "2021-12-10T00:00:00Z",
          "cisa_kev_date_due": "2021-12-24T00:00:00Z",
          "cwes": [
            400,
            20,
            502
          ],
          "cvss": [
            {
              "source": "NVD",
              "severity": "critical",
              "cvss3_base_score": 10.0,
              "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H",
              "cvss2_base_score": 9.3,
              "cvss2_base_vector": "AC:M/Au:N/C:C/I:C/A:C"
            },
            {
              "source": "SNYK",
              "severity": "critical",
              "cvss3_base_score": 10.0,
```

```
              "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H/E:H"
            },
            {
              "source": "GITHUB",
              "severity": "critical",
              "cvss3_base_score": 10.0,
              "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
            }
          ],
          "epss": 0.97565,
          "exploit_available": true,
          "exploit_last_seen_in_public": "2023-03-06T00:00:00Z"
        },
        "affects": [
          {
            "installed_version": "pkg:maven/org.apache.logging.log4j/log4j-
core@2.12.1",
            "fixed_version": "2.15.0",
            "path": "/home/dev/foo.jar"
          }
        ]
      }
    ]
  }
}
```


Amazon Inspector 사용 Jenkins 플러그인 비활성화

는 Jenkins 플러그인은 Amazon Inspector SBOM Generator 바이너리와 Amazon Inspector ScanAPI을 활용하여 빌드가 끝날 때 세부 보고서를 생성하므로 배포 전에 위험을 조사하고 해결할 수 있습니다. Amazon Inspector 사용 Jenkins 플러그인, Amazon Inspector 취약성 스캔을 에 추가할 수 있습니다.Jenkins 파이프라인. Amazon Inspector 취약성 스캔은 탐지된 취약성의 수와 심각도에 따라 파이프라인 실행을 통과하거나 실패하도록 구성할 수 있습니다. 의 최신 버전을 볼 수 있습니다.Jenkins 플러그인 Jenkins https://plugins.jenkins.io/amazon-inspector-image-scanner 마켓플레이스. 다음 단계에서는 Amazon Inspector를 설정하는 방법을 설명합니다. Jenkins 플러그인과 함께 사용할 수 있습니다.

> **Important**
> 다음 단계를 완료하기 전에 플러그인을 실행하려면 Jenkins를 버전 2.387.3 이상으로 업그레이드해야 합니다.

단계 1. 설정 AWS 계정

Amazon Inspector Scan에 대한 액세스를 허용하는 IAM 역할 AWS 계정 로 를 구성합니다API. 지침은 Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정 단원을 참조하십시오.

단계 2. Amazon Inspector Jenkins 플러그인 설치

다음 절차에서는 에서 Amazon Inspector Jenkins 플러그인을 설치하는 방법을 설명합니다.Jenkins 대시보드.

1. Jenkins 대시보드에서 Jenkins 관리를 선택한 다음 플러그인 관리 를 선택합니다.
2. 사용 가능 을 선택합니다.
3. 사용 가능 탭에서 Amazon Inspector Scans 를 검색한 다음 플러그인을 설치합니다.

(선택 사항) 3단계. 에 도커 자격 증명 추가 Jenkins

> **Note**
> 도커 이미지가 프라이빗 리포지토리에 있는 경우에만 도커 자격 증명을 추가합니다. 그렇지 않은 경우 이 단계를 건너뜁니다.

다음 절차에서는 에 도커 자격 증명을 추가하는 방법을 설명합니다.Jenkins 에서 Jenkins 대시보드.

1. Jenkins 대시보드에서 Jenkins 관리 , 자격 증명 , 시스템 을 차례로 선택합니다.
2. 글로벌 보안 인증 정보를 선택한 다음 보안 인증 정보 추가를 선택합니다.
3. Kind 에서 암호가 인 사용자 이름을 선택합니다.
4. 범위 에서 전역(젠킨스, 노드, 항목, 모든 하위 항목 등)을 선택합니다.
5. 세부 정보를 입력한 다음 확인을 선택합니다.

(선택 사항) 4단계. AWS 보안 인증 정보 추가

> **Note**
>
> IAM 사용자를 기반으로 인증하려는 경우에만 AWS 자격 증명을 추가합니다. 그렇지 않은 경우 이 단계를 건너뜁니다.

다음 절차에서는 에서 AWS 자격 증명을 추가하는 방법을 설명합니다.Jenkins 대시보드.

1. Jenkins 대시보드에서 Jenkins 관리 , 보안 인증 , 시스템 을 차례로 선택합니다.
2. 글로벌 보안 인증 정보를 선택한 다음 보안 인증 정보 추가를 선택합니다.
3. Kind 에서 AWS 자격 증명 을 선택합니다.
4. 액세스 키 ID 및 보안 액세스 키 등 세부 정보를 입력한 다음 확인을 선택합니다.

5단계. 에서 CSS 지원 추가 Jenkins 스크립트

다음 절차에서는 에서 CSS 지원을 추가하는 방법을 설명합니다.Jenkins 스크립트.

1. Jenkins를 다시 시작하세요.
2. 대시보드에서 Jenkins 관리 , 노드 , 내장 노드 를 선택한 다음 스크립트 콘솔 을 선택합니다.
3. 텍스트 상자에 줄을 추가`System.setProperty("hudson.model.DirectoryBrowserSupport.CSP", "")`한 다음 실행을 선택합니다.

6단계. 빌드에 Amazon Inspector Scan 추가

프로젝트의 Amazon Inspector 빌드 단계를 추가하거나 Jenkins 선언 파이프라인.

프로젝트에 빌드 단계를 추가하여 빌드에 Amazon Inspector 스캔

1. 구성 페이지에서 빌드 단계까지 아래로 스크롤하고 빌드 단계 추가를 선택합니다. 그런 다음 Amazon Inspector Scan 을 선택합니다.
2. 두 가지 nspector-sbomgen 설치 방법(자동 또는 수동) 중에서 선택합니다.

    a. (옵션 1) 자동을 선택하여 최신 버전의 Inspector-sbomgen을 다운로드합니다. 이 방법을 선택하는 경우 플러그인을 실행하는 시스템과 일치하는 CPU 아키텍처를 선택해야 합니다.

b. (옵션 2) 스캔을 위해 Amazon Inspector SBOM Generator 바이너리를 설정하려면 수동을 선택합니다. 이 방법을 선택하는 경우 이전에 다운로드한 Inspector-sbomgen 버전의 전체 경로를 제공해야 합니다.

자세한 내용은 Amazon Inspector SBOM 생성기에서 Amazon Inspector 생성기(Sbomgen) 설치를 참조하세요. Amazon Inspector SBOM

3. 다음을 완료하여 Amazon Inspector 스캔 빌드 단계 구성을 완료합니다.

   a. 이미지 ID를 입력합니다. 이미지는 로컬, 원격 또는 아카이브 위치에 있을 수 있습니다. 이미지 이름은 를 따라야 합니다.Docker 명명 규칙. 내보낸 이미지를 분석하는 경우 예상 tar 파일의 경로를 제공하세요. 다음 이미지를 예시 이미지 ID 경로로 참조하세요.

      i. 로컬 또는 원격 컨테이너의 경우: `NAME[:TAG|@DIGEST]`

      ii. tar 파일의 경우: `/path/to/image.tar`

   b. 스캔 요청을 보낼 AWS 리전을 선택합니다.

   c. (선택 사항) Docker 자격 증명 에서 Docker 사용자 이름. 컨테이너 이미지가 프라이빗 리포지토리에 있는 경우에만 이 작업을 수행하세요.

   d. (선택 사항) 지원되는 AWS 인증 방법은 다음과 같습니다.

      i. (선택 사항) IAM 역할 에 역할ARN(arn:aws:iam::*AccountNumber*:role/*RoleName*).

      ii. (선택 사항) AWS 자격 증명 에서 ID를 선택하여 IAM 사용자를 기반으로 인증합니다.

      iii. (선택 사항) AWS 프로파일 이름 에 프로파일 이름을 사용하여 인증할 프로파일의 이름을 입력합니다.

   e. (선택 사항) 심각도별 취약성 임계값을 지정합니다. 스캔 중에 지정한 수를 초과하면 이미지 빌드가 실패합니다. 값이 모두 `0`이면 취약성이 발견되더라도 빌드가 성공합니다.

4. 저장(Save)을 선택합니다.

를 사용하여 빌드에 Amazon Inspector Scan 추가 Jenkins 선언 파이프라인

Jenkins 선언 파이프라인을 사용하여 빌드에 Amazon Inspector Scan을 자동으로 또는 수동으로 추가할 수 있습니다.

SBOMGen 선언 파이프라인을 자동으로 다운로드하려면

- 빌드에 Amazon Inspector Scan을 추가하려면 다음 예제 구문을 사용합니다. Amazon Inspector SBOM Generator 다운로드에서 선호하는 OS 아키텍처에 따라 교체 *SBOMGEN_SOURCE*

linuxAmd64 또는 linuxArm64. Replace *IMAGE_PATH* 이미지 경로(예: *alpine:latest*), *IAM_ROLE* 1단계에서 구성한 IAM 역할ARN의 *ID* 와 함께 Docker 프라이빗 리포지토리를 사용하는 경우 보안 인증 ID입니다. 선택적으로 취약성 임계값을 활성화하고 각 심각도에 대한 값을 지정할 수 있습니다.

```
pipeline {
   agent any
   stages {
       stage('amazon-inspector-image-scanner') {
           steps {
               script {
               step([
               $class:
 'com.amazon.inspector.jenkins.amazoninspectorbuildstep.AmazonInspectorBuilder',
               sbomgenSource: 'SBOMGEN_SOURCE', // this can be linuxAmd64 or linuxArm64
               archivePath: 'IMAGE_PATH',
               awsRegion: 'REGION',
               iamRole: 'IAM ROLE',
               credentialId: 'Id', // provide empty string if image not in private
 repositories
               awsCredentialId: ''AWS ID;',
               awsProfileName: 'Profile Name',
               isThresholdEnabled: false,
               countCritical: 0,
               countHigh: 0,
               countLow: 10,
               countMedium: 5,
              ])
           }
        }
      }
   }
 }
```

SBOMGen 선언 파이프라인을 수동으로 다운로드하려면

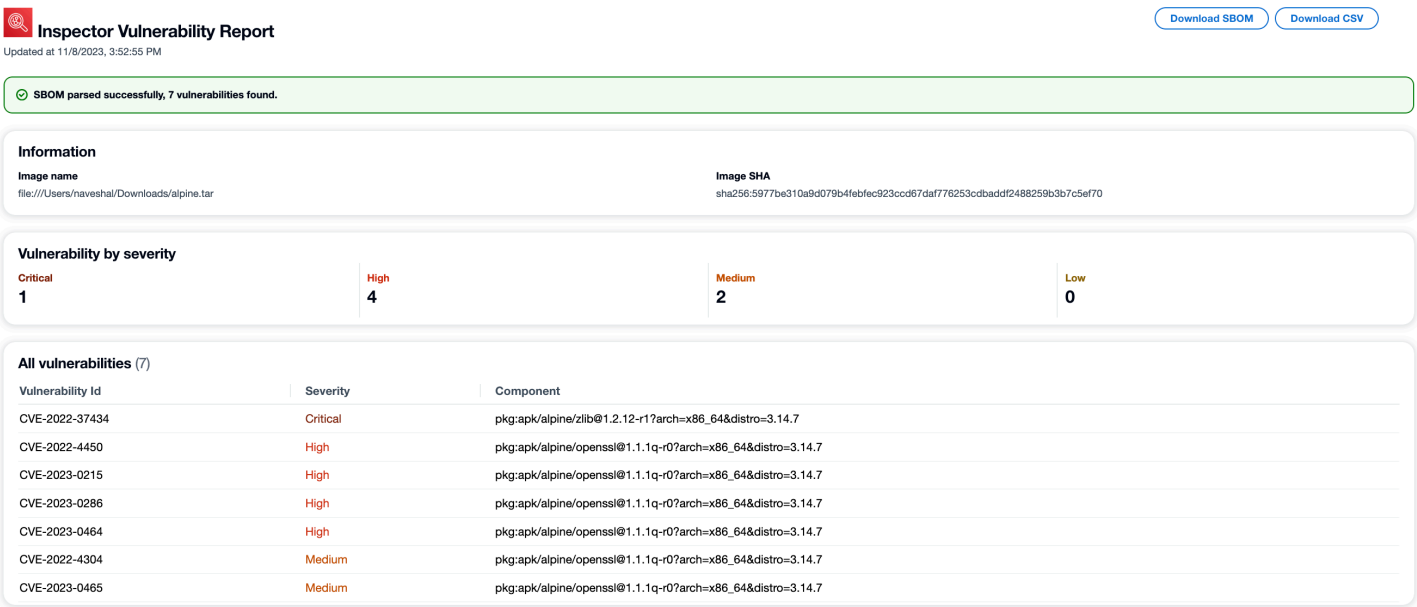
- 빌드에 Amazon Inspector Scan을 추가하려면 다음 예제 구문을 사용합니다. Replace *SBOMGEN_PATH* 3단계에서 설치한 Amazon Inspector SBOM Generator의 경로를 사용하여 *IMAGE_PATH* 이미지 경로(예: *alpine:latest*), *IAM_ROLE* 1단계에서 구성한 IAM 역할ARN의 *ID* 와 함께 Docker 프라이빗 리포지토리를 사용하는 경우 보안 인증 ID입니다. 선택적으로 취약성 임계값을 활성화하고 각 심각도에 대한 값을 지정할 수 있습니다.

> **Note**
>
> Place Sbomgen Jenkins 디렉터리에서, 플러그인의 Jenkins 디렉터리에 대한 경로를 제공합니다(예: */opt/folder/arm64/inspector-sbomgen*).

```
pipeline {
    agent any
    stages {
        stage('amazon-inspector-image-scanner') {
            steps {
                script {
                step([
                $class:
 'com.amazon.inspector.jenkins.amazoninspectorbuildstep.AmazonInspectorBuilder',
                sbomgenPath: 'SBOMGEN_PATH',
                archivePath: 'IMAGE_PATH',
                awsRegion: 'REGION',
                iamRole: 'IAM ROLE',
                awsCredentialId: ''AWS ID;',
                credentialId: 'Id;', // provide empty string if image not in private
 repositories
                awsProfileName: 'Profile Name',
                isThresholdEnabled: false,
                countCritical: 0,
                countHigh: 0,
                countLow: 10,
                countMedium: 5,
               ])
            }
         }
       }
    }
 }
```

7단계. Amazon Inspector 취약성 보고서 보기

1. 프로젝트의 새 빌드를 완료합니다.
2. 빌드가 완료되면 결과에서 출력 형식을 선택합니다. 를 선택하면 JSON SBOM 또는 CSV 버전의 보고서를 다운로드할 수 있는 옵션이 HTML있습니다. 다음은 HTML 보고서의 예입니다.



문제 해결

다음은 Amazon Inspector Scan 플러그인을 사용할 때 발생할 수 있는 일반적인 오류입니다.Jenkins.

자격 증명 또는 sts 예외 오류 로드 실패

오류:

```
InstanceProfileCredentialsProvider(): Failed to load credentials or sts
exception.
```

소울레이션

AWS 계정에 `aws_secret_access_key` 대해 `aws_access_key_id` 및 를 가져옵니다. `aws_access_key_id` 및 `aws_secret_access_key`를 `~/.aws/credentials`에 설정합니다.

Inspector-sbomgen 경로 오류

오류:

```
Exception:com.amazon.inspector.jenkins.amazoninspectorbuildstep.exception.Sbomge
There was an issue running inspector-sbomgen, is /opt/inspector/inspector-
sbomgen the correct path?
```

해결 방법:

다음 절차를 완료하여 문제를 해결합니다.

1. 에 올바른 OS 아키텍처 Inspector-sbomgen 배치 Jenkins 디렉터리 자세한 내용은 Amazon Inspector SBOM Generator 를 참조하세요.
2. 다음 명령을 사용하여 바이너리에 실행 가능 권한을 부여합니다`chmod +x inspector-sbomgen`.
3. 올바른 입력 Jenkins 와 같은 플러그인의 기계 경로입니다`/opt/folder/arm64/inspector-sbomgen`.
4. 구성 저장 및 실행 Jenkins 작업.

Amazon Inspector 사용 TeamCity 플러그인 비활성화

Amazon Inspector TeamCity 플러그인은 Amazon Inspector SBOM Generator 바이너리와 Amazon Inspector ScanAPI을 활용하여 빌드가 끝날 때 세부 보고서를 생성하므로 배포 전에 위험을 조사하고 해결할 수 있습니다. Amazon Inspector 사용 TeamCity 플러그인, Amazon Inspector 취약성 스캔을 에 추가할 수 있습니다.TeamCity 파이프라인. Amazon Inspector 취약성 스캔은 탐지된 취약성의 수와 심각도에 따라 파이프라인 실행을 통과하거나 실패하도록 구성할 수 있습니다. 최신 버전의 Amazon Inspector를 볼 수 있습니다.TeamCity 플러그인 TeamCity https://plugins.jetbrains.com/plugin/23236-amazon-inspector-scanner의 마켓플레이스. Amazon Inspector Scan을 CI/CD 파이프라인에 통합하는 방법에 대한 자세한 내용은 Amazon Inspector 스캔을 CI/CD 파이프라인에 통합을 참조하세요. Amazon Inspector에서 지원하는 운영 체제 및 프로그래밍 언어 목록은 지원되는 운영 체제 및 프로그래밍 언어 를 참조하세요. 다음 단계에서는 Amazon Inspector를 설정하는 방법을 설명합니다. TeamCity 플러그인과 함께 사용할 수 있습니다.

1. 를 설정합니다 AWS 계정.
   - Amazon Inspector Scan에 대한 액세스를 허용하는 IAM 역할 AWS 계정 로 를 구성합니다API. 지침은 Amazon Inspector CI/CD 통합을 사용하도록 AWS 계정 설정 단원을 참조하십시오.
2. Amazon Inspector 설치 TeamCity 플러그인.
   a. 대시보드에서 관리 > 플러그인으로 이동합니다.
   b. Amazon Inspector 스캔을 검색합니다.
   c. 플러그인을 설치합니다.
3. Amazon Inspector SBOM 생성기를 설치합니다.
   - Teamcity 서버 디렉터리에 Amazon Inspector SBOM Generator 바이너리를 설치합니다. 지침은 설치 Sbomgen 단원을 참조하십시오.
4. Amazon Inspector 스캔 빌드 단계를 프로젝트에 추가합니다.

a. 구성 페이지에서 아래로 스크롤하여 빌드 단계 로 이동하고 빌드 단계 추가 를 선택한 다음 Amazon Inspector 스캔 을 선택합니다.

b. 다음 세부 정보를 입력하여 Amazon Inspector 스캔 빌드 단계를 구성합니다.

- 단계 이름 을 추가합니다.
- 두 Amazon Inspector SBOM Generator 설치 방법, 즉 자동 또는 수동 중에서 선택합니다.
  - 자동은 시스템 및 CPU 아키텍처를 기반으로 최신 버전의 Amazon Inspector SBOM Generator를 다운로드합니다.
  - 수동을 사용하려면 이전에 다운로드한 버전의 Amazon Inspector SBOM Generator에 대한 전체 경로를 제공해야 합니다.

  자세한 내용은 Amazon Inspector SBOM Generator에 Amazon Inspector Generator(Sbomgen) 설치를 참조하세요. Amazon Inspector SBOM

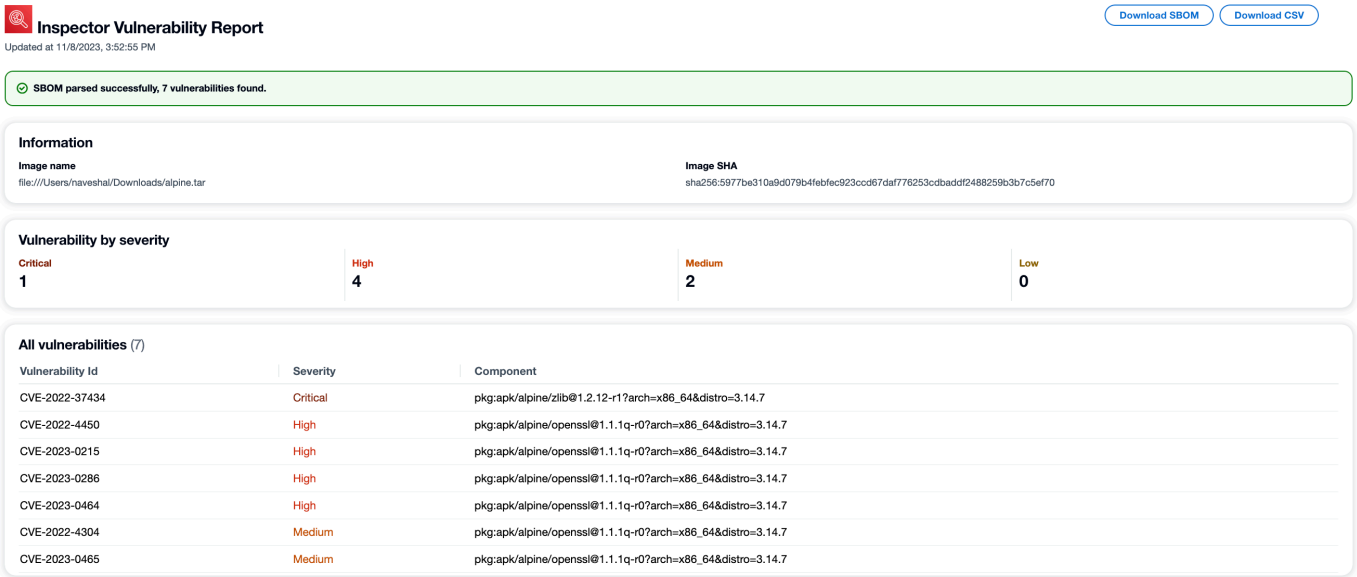
- 이미지 ID를 입력합니다. 이미지는 로컬, 원격 또는 아카이브 위치에 있을 수 있습니다. 이미지 이름은 Docker 명명 규칙. 내보낸 이미지를 분석하는 경우 예상 tar 파일의 경로를 제공하세요. 다음 이미지를 예시 이미지 ID 경로로 참조하세요.
  - 로컬 또는 원격 컨테이너의 경우: `NAME[:TAG|@DIGEST]`
  - tar 파일의 경우: `/path/to/image.tar`
- IAM 역할에 ARN 1단계에서 구성한 역할의 를 입력합니다.
- 스캔 요청을 보낼 AWS 리전을 선택합니다.
- (선택 사항) Docker 인증의 경우 Docker 사용자 이름과 Docker 암호를 입력합니다. 컨테이너 이미지가 프라이빗 리포지토리에 있는 경우에만 이 작업을 수행하세요.
- (선택 사항) AWS 인증 에 AWS 액세스 키 ID와 AWS 보안 키를 입력합니다. 자격 AWS 증명을 기반으로 인증하려는 경우에만 이 작업을 수행합니다.
- (선택 사항) 심각도별 취약성 임계값을 지정합니다. 스캔 중에 지정한 수를 초과하면 이미지 빌드가 실패합니다. 값이 모두 `0`이면 발견되는 취약성 수와 상관없이 빌드가 성공합니다.

c. 저장을 선택합니다.

5. Amazon Inspector 취약성 보고서를 확인합니다.

a. 프로젝트의 새 빌드를 완료합니다.

b. 빌드가 완료되면 결과에서 출력 형식을 선택합니다. 를 선택하면 JSON SBOM 또는 CSV 버전의 보고서를 다운로드할 수 있는 옵션이 HTML 있습니다. 다음은 HTML 보고서의 예입니다.



에서 Amazon Inspector 사용 GitHub actions

Amazon Inspector를 GitHub actions 에 Amazon Inspector 취약성 스캔을 추가하려면 GitHub 워크플로. 이렇게 하면 Amazon Inspector SBOM Generator 및 Amazon Inspector ScanAPI을 활용하여 빌드가 끝날 때 세부 보고서를 생성하므로 배포 전에 위험을 조사하고 해결할 수 있습니다. Amazon Inspector 취약성 스캔은 탐지된 취약성의 수와 심각도에 따라 워크플로를 통과하거나 실패하도록 구성할 수 있습니다. 에서 Amazon Inspector 작업의 최신 버전을 볼 수 있습니다. GitHub 웹 사이트 . Amazon Inspector Scan을 CI/CD 파이프라인에 통합하는 방법에 대한 자세한 내용은 Amazon Inspector 스캔을 CI/CD 파이프라인에 통합을 참조하세요. Amazon Inspector에서 지원하는 운영 체제 및 프로그래밍 언어 목록은 지원되는 운영 체제 및 프로그래밍 언어 를 참조하세요.

사용 CodeCatalyst Amazon Inspector를 사용한 작업

Amazon Inspector와 Amazon CodeCatalyst을 함께 사용하여 CodeCatalyst 워크플로에 Amazon Inspector 취약성 스캔을 추가할 수 있습니다. 이렇게 하면 Amazon Inspector SBOM Generator 및 Amazon Inspector ScanAPI을 활용하여 빌드가 끝날 때 세부 보고서를 생성하므로 배포 전에 위험을 조사하고 해결할 수 있습니다. Amazon Inspector 취약성 스캔은 탐지된 취약성의 수와 심각도에 따라 워크플로를 통과하거나 실패하도록 구성할 수 있습니다. Amazon Inspector Scan을 CI/CD 파이프라인에 통합하는 방법에 대한 자세한 내용은 Amazon Inspector 스캔을 CI/CD 파이프라인에 통합을 참조하세요. Amazon Inspector에서 지원하는 운영 체제 및 프로그래밍 언어 목록은 지원되는 운영 체제 및 프로그래밍 언어 를 참조하세요.

Amazon Inspector에서 CycloneDX 네임스페이스 사용

Amazon Inspector는 함께 사용할 수 있는 CycloneDX 네임스페이스 및 속성 이름을 제공합니다. SBOMs 이 섹션에서는 구성 요소에 추가할 수 있는 모든 사용자 지정 키/값 속성에 대해 설명합니다. CycloneDX SBOMs 자세한 내용은 웹 사이트의 CyclonedX 속성 분류를 참조하십시오. GitHub

`amazon:inspector:sbom_scanner` 네임스페이스 분류법

Amazon Inspector API Scan은 `amazon:inspector:sbom_scanner` 네임스페이스를 사용하며 다음과 같은 속성을 갖습니다.

속성	설명
`amazon:inspector:sbom_scanner:cisa_kev_date_added`	CISA알려진 악용 취약성 카탈로그에 취약성이 추가된 시기를 나타냅니다.
`amazon:inspector:sbom_scanner:cisa_kev_date_due`	CISA알려진 악용 취약성 카탈로그에 따라 취약성 수정 시기가 언제인지를 나타냅니다.
`amazon:inspector:sbom_scanner:critical_vulnerabilities`	에서 발견된 심각한 심각도 취약성의 총 개수 SBOM
`amazon:inspector:sbom_scanner:exploit_available`	해당 취약성에 대한 악용이 가능한지를 나타냅니다.
`amazon:inspector:sbom_scanner:exploit_last_seen_in_public`	해당 취약성에 대한 악용이 공개적으로 마지막으로 발견된 시기를 나타냅니다.
`amazon:inspector:sbom_scanner:fixed_version:` *`component_bom_ref`*	지정된 취약성에 대해 표시된 구성 요소의 수정된 버전을 제공합니다.
`amazon:inspector:sbom_scanner:high_vulnerabilities`	에서 발견된 심각도가 높은 취약성의 총 개수 SBOM
`amazon:inspector:sbom_scanner:info`	특정 구성 요소에 대한 스캔 컨텍스트를 제공합니다(예: '구성 요소 검사됨: 발견된 취약성 없음').

속성	설명
amazon:inspector:sbom_scanner:is_malicious	Open이 영향을 받는 구성 요소를 SSF 악성으로 식별하는지 여부를 나타냅니다.
amazon:inspector:sbom_scanner:low_vulnerabilities	에서 발견된 심각도가 낮은 취약성의 총 개수 SBOM
amazon:inspector:sbom_scanner:medium_vulnerabilities	에서 발견된 중간 심각도 취약성의 총 개수 SBOM
amazon:inspector:sbom_scanner:path	주제 패키지 정보를 제공하는 파일의 경로입니다.
amazon:inspector:sbom_scanner:priority	특정 취약성 해결을 위한 권장 우선 순위. 값은 내림차순으로 ""“," IMMEDIATE “," URGENT “,"MODERATESTANDARD”""""입니다.
amazon:inspector:sbom_scanner:priority_intelligence	특정 취약점의 우선 순위를 결정하는 데 사용되는 인텔리전스 품질. 값에는 "" 또는 VERIFIED UNVERIFIED ""“이 포함됩니다.
amazon:inspector:sbom_scanner:warning	특정 구성 요소가 스캔되지 않은 이유에 대한 컨텍스트를 제공합니다(예: '구성 요소 건너뜀: purl 제공 안 됨').

amazon:inspector:sbom_generator 네임스페이스 분류법

Amazon Inspector SBOM Generator는 amazon:inspector:sbom_generator 네임스페이스를 사용하며 다음과 같은 속성을 가지고 있습니다.

속성	설명
amazon:inspector:sbom_generator:cpu_architecture	인벤토리 대상 시스템의 CPU 아키텍처 (예: x86_64)

속성	설명
amazon:inspector:sbom_generator:dockerfile_finding: *inspector_finding_id*	Amazon Inspector가 구성 요소에서 찾은 결과가 검사와 관련이 Dockerfile 있음을 나타냅니다.
amazon:inspector:sbom_generator:expires_after	SSL인증서가 유효하지 않은 날짜.
amazon:inspector:sbom_generator:expires_before	SSL인증서가 유효하기 전 날짜입니다.
amazon:inspector:sbom_generator:go_toolchain	Go 실행 파일을 생성하는 데 사용된 Go 컴파일러 또는 툴체인 버전을 나타냅니다.
amazon:inspector:sbom_generator:image_id	컨테이너 이미지 구성 파일의 해시(이미지 ID라고도 함).
amazon:inspector:sbom_generator:is_duplicate_package	두 개 이상의 파일 스캐너에서 주제 패키지를 찾았음을 나타냅니다.
amazon:inspector:sbom_generator:is_expired	SSL인증서가 만료되었는지 여부를 나타내는 부울 값입니다.
amazon:inspector:sbom_generator:kernel_name	인벤토리 대상 시스템의 커널 이름.
amazon:inspector:sbom_generator:kernel_version	인벤토리 대상 시스템의 커널 버전.
amazon:inspector:sbom_generator:layer_diff_id	압축되지 않은 컨테이너 이미지 계층의 해시.
amazon:inspector:sbom_generator:os_hostname	인벤토리 대상 시스템의 호스트 이름.
amazon:inspector:sbom_generator:source_file_scanner	패키지 정보가 들어 있는 파일을 찾은 스캐너(예: /var/lib/dpkg/status).

속성	설명
amazon:inspector:sbom_generator:source_package_collector	특정 파일에서 패키지 이름과 버전을 추출한 컬렉터.
amazon:inspector:sbom_generator:source_path	주제 패키지 정보가 추출된 파일의 경로.

AWS 환경에 대한 Amazon Inspector 적용 범위 평가

Amazon Inspector 콘솔의 계정 관리 화면에서 AWS 환경의 Amazon Inspector 범위를 평가할 수 있습니다. 이 화면에는 계정 및 리소스에 대한 Amazon Inspector 스캔 상태에 대한 세부 정보와 통계가 표시됩니다.

> **Note**
> 조직의 위임된 관리자인 경우 조직의 모든 계정에 대한 세부 정보와 통계를 볼 수 있습니다.

다음 절차에서는 Amazon Inspector 환경의 적용 범위를 평가하는 방법을 설명합니다.

AWS 환경의 Amazon Inspector 적용 범위를 평가하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
2. 탐색 창에서 계정 관리 를 선택합니다.
3. 적용 범위를 검토하려면 다음 탭 중 하나를 선택합니다.
   - 계정을 선택하여 계정 수준 적용 범위를 검토합니다.
   - 인스턴스를 선택하여 Amazon Elastic Compute Cloud(AmazonEC2) 인스턴스에 대한 적용 범위를 검토합니다.
   - 컨테이너 리포지토리를 선택하여 Amazon Elastic Container Registry(AmazonECR) 리포지토리의 적용 범위를 검토합니다.
   - 컨테이너 이미지를 선택하여 Amazon ECR 컨테이너 이미지에 대한 적용 범위를 검토합니다.
   - Lambda 함수를 선택하여 Lambda 함수에 대한 적용 범위를 검토합니다.

다음 주제에서는 이러한 각 탭이 제공하는 정보를 설명합니다.

주제

- 계정 수준 적용 범위 평가
- Amazon EC2 인스턴스의 적용 범위 평가
- Amazon ECR 리포지토리의 적용 범위 평가
- Amazon ECR 컨테이너 이미지의 적용 범위 평가

- AWS Lambda 함수 범위 평가

계정 수준 적용 범위 평가

계정이 조직의 일부가 아니거나 조직의 위임된 Amazon Inspector 관리자 계정이 아닌 경우 계정 탭에는 사용자 계정에 대한 정보와 계정의 리소스 스캔 상태가 제공됩니다. 이 탭에서는 계정의 모든 리소스 또는 특정 유형의 리소스에 대해서만 스캔을 활성화하거나 비활성화할 수 있습니다. 자세한 내용은 Amazon Inspector의 자동 스캔 유형 단원을 참조하십시오.

계정이 조직의 위임된 Amazon Inspector 관리자 계정인 경우, 계정 탭에는 조직의 계정에 대한 자동 활성화 설정이 제공되며 조직의 모든 계정이 나열됩니다. 각 계정에 대해 목록에는 해당 계정에 대해 Amazon Inspector가 활성화되어 있는지 여부와 활성화된 경우 계정에 활성화된 리소스 스캔 유형이 표시됩니다. 위임된 관리자는 이 탭에서 조직의 자동 활성화 설정을 변경할 수 있습니다. 개별 멤버 계정에 대해 특정 유형의 리소스 검색을 활성화하거나 비활성화할 수도 있습니다. 자세한 내용은 멤버 계정에 대한 Amazon Inspector 스캔 활성화 단원을 참조하십시오.

Amazon EC2 인스턴스의 적용 범위 평가

인스턴스 탭에는 환경 AWS 의 Amazon EC2 인스턴스가 표시됩니다. 목록은 다음 탭에서 그룹으로 구성되어 있습니다.

- 모두 - 사용자 환경에 있는 모든 인스턴스가 표시됩니다. 상태 열은 인스턴스의 현재 스캔 상태를 나타냅니다.
- 스캔 – Amazon Inspector가 사용자 환경에서 능동적으로 모니터링 및 스캔하고 있는 모든 인스턴스가 표시됩니다.
- 스캔하지 않음 – Amazon Inspector가 사용자 환경에서 모니터링 및 스캔하지 않는 모든 인스턴스가 표시됩니다. 이유 열은 Amazon Inspector가 인스턴스를 모니터링 및 스캔하지 않는 이유를 나타냅니다.

  EC2 인스턴스는 여러 가지 이유로 스캔하지 않음 탭에 표시될 수 있습니다. Amazon Inspector는 AWS Systems Manager (SSM) 및 SSM 에이전트를 사용하여 EC2 인스턴스의 취약성을 자동으로 모니터링하고 스캔합니다. 인스턴스에 SSM 에이전트가 실행되지 않거나, Systems Manager를 지원하는 AWS Identity and Access Management (IAM) 역할이 없거나, 지원되는 운영 체제 또는 아키텍처를 실행하지 않는 경우 Amazon Inspector는 인스턴스를 모니터링하고 스캔할 수 없습니다. 자세한 내용은 Amazon EC2 인스턴스 스캔 단원을 참조하십시오.

각 탭에서 계정 열은 인스턴스를 소유 AWS 계정 한 를 지정합니다.

EC2 인스턴스 태그 - 이 열에는 인스턴스와 연결된 태그가 표시되며, 이 열을 사용하여 인스턴스가 태그로 스캔에서 제외되었는지 확인할 수 있습니다.

운영 체제 - 이 열에는 운영 체제 유형(`WINDOWS`, `MAC`, `LINUX` 또는 `UNKNOWN`)이 표시됩니다.

를 사용하여 모니터링됨 - 이 열은 Amazon Inspector가 이 인스턴스에서 에이전트 기반 또는 에이전트리스 스캔 방법을 사용하고 있는지 여부를 보여줍니다.

마지막 스캔 - 이 열에는 Amazon Inspector가 해당 리소스의 취약성을 마지막으로 검사한 시간이 표시됩니다. Amazon Inspector에서 스캔을 수행하는 빈도는 인스턴스를 스캔하는 데 사용하는 스캔 방법에 따라 다릅니다.

EC2 인스턴스에 대한 추가 세부 정보를 검토하려면 EC2 인스턴스 열에서 링크를 선택합니다. 그러면 Amazon Inspector에서 인스턴스에 대한 세부 정보와 해당 인스턴스에 대한 현재 결과를 표시합니다. 결과에 대한 세부 정보를 검토하려면 제목 열의 링크를 선택하세요. 이러한 세부 정보에 대한 자세한 내용은 Amazon Inspector 조사 결과에 대한 세부 정보 보기 섹션을 참조하세요.

Amazon EC2 인스턴스의 상태 값 스캔

Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스의 경우 가능한 상태 값은 다음과 같습니다.

- 능동 모니터링 – Amazon Inspector가 인스턴스를 지속적으로 모니터링하고 스캔합니다.
- 에이전트리스 인스턴스 스토리지 한도 초과 - Amazon Inspector는 인스턴스에 연결된 모든 볼륨의 결합 크기가 1200GB를 초과하거나 인스턴스에 8개 이상의 볼륨이 연결된 경우 이 상태를 사용합니다.
- 에이전트리스 인스턴스 수집 시간 제한 초과 - Amazon Inspector가 인스턴스에서 에이전트리스 스캔을 실행하려고 시도하는 동안 시간 초과됩니다.
- EC2 인스턴스 중지됨 - 인스턴스가 중지된 상태이므로 Amazon Inspector가 인스턴스에 대한 스캔을 일시 중지했습니다. 기존 스캔 결과는 인스턴스가 종료될 때까지 보존됩니다. 인스턴스가 다시 시작되면 Amazon Inspector가 자동으로 인스턴스 스캔을 재개합니다.
- 내부 오류 – Amazon Inspector에서 인스턴스 스캔을 시도할 때 내부 오류가 발생했습니다. Amazon Inspector에서 자동으로 오류를 해결하고 가능한 한 빨리 스캔을 재개합니다.
- 인벤토리 없음 – Amazon Inspector에서 인스턴스를 스캔할 소프트웨어 애플리케이션 인벤토리를 찾지 못했습니다. 인스턴스의 Amazon Inspector 연결이 삭제되었거나 실행에 실패했을 수 있습니다.

  이 문제를 해결하려면 AWS Systems Manager 를 사용하여 `InspectorInventoryCollection-do-not-delete` 연결이 존재하고 연결 상태가 성공적인지 확인합니다. 또한 AWS Systems

Manager Fleet Manager를 사용하여 인스턴스에 대한 소프트웨어 애플리케이션 인벤토리를 확인하세요.

- 비활성화 보류 중 – Amazon Inspector가 인스턴스 스캔을 중단했습니다. 인스턴스는 비활성화되어 정리 작업이 완료될 때까지 대기 중입니다.
- 첫 번째 스캔 보류 중 – Amazon Inspector에서 첫 번째 스캔을 위해 인스턴스를 대기열에 추가했습니다.
- 리소스 종료됨 – 인스턴스가 종료되었습니다. Amazon Inspector가 현재 인스턴스에 대한 기존 스캔 결과 및 적용 범위 데이터를 정리하고 있습니다.
- 기한 경과 인벤토리 – Amazon Inspector에서 인스턴스에 대해 지난 7일 이내에 캡처한 업데이트된 소프트웨어 애플리케이션 인벤토리를 수집하지 못했습니다.

  이 문제를 해결하려면 AWS Systems Manager 를 사용하여 인스턴스에 필요한 Amazon Inspector 연결이 존재하고 실행 중인지 확인합니다. 또한 AWS Systems Manager Fleet Manager를 사용하여 인스턴스에 대한 소프트웨어 애플리케이션 인벤토리를 확인하세요.
- 관리되지 않는 EC2 인스턴스 - Amazon Inspector는 인스턴스를 모니터링하거나 스캔하지 않습니다. AWS Systems Manager에서 인스턴스를 관리하지 않습니다.

  이 문제를 해결하려면 AWSSupport-TroubleshootManagedInstance runbook AWS Systems Manager Automation에서 제공합니다. 인스턴스를 관리 AWS Systems Manager 하도록 를 구성하면 Amazon Inspector가 인스턴스를 자동으로 지속적으로 모니터링하고 스캔하기 시작합니다.
- 지원되지 않는 OS – Amazon Inspector에서 인스턴스를 모니터링하거나 스캔하지 않습니다. 인스턴스가 Amazon Inspector에서 지원하지 않는 운영 체제 또는 아키텍처를 사용합니다. Amazon Inspector에서 지원하는 운영 체제 목록은 ??? 섹션을 참조하세요.
- 부분 오류가 있는 활성 모니터링 - 이 상태는 EC2 스캔이 활성 상태이지만 와 관련된 오류가 있음을 의미합니다Linux 기반 Amazon 인스턴스에 대한 Amazon Inspector 심층 검사 EC2 . 가능한 심층 검사 오류는 다음과 같습니다.
  - 심층 검사 패키지 수집 한도 초과 - 인스턴스가 Amazon Inspector 심층 검사에 대한 5000 패키지 한도를 초과했습니다. 이 인스턴스에 대한 심층 검사를 재개하려면 계정과 연결된 사용자 지정 경로를 조정해 볼 수 있습니다.
  - 심층 검사 일일 ssm 인벤토리 제한 초과 - SSM 에이전트가 Amazon Inspector에 인벤토리를 보낼 수 없습니다. 이 인스턴스에 대해 인스턴스당 수집된 인벤토리 데이터의 SSM 할당량에 이미 도달했기 때문입니다. 자세한 내용은 Amazon EC2 Systems Manager 엔드포인트 및 할당량 섹션을 참조하세요.
  - 심층 검사 수집 시간 제한 초과 - 패키지 수집 시간이 최대 임계값인 15분을 초과했기 때문에 Amazon Inspector가 패키지 인벤토리를 추출하지 못했습니다.

- 심층 검사에는 인벤토리가 없습니다. Amazon Inspector SSM 플러그인은 아직 이 인스턴스에 대한 패키지 인벤토리를 수집할 수 없습니다. 이는 일반적으로 보류 중인 스캔의 결과이지만 이 상태가 6시간 후에도 지속되면 Amazon EC2 Systems Manager를 사용하여 필요한 Amazon Inspector 연결이 존재하고 인스턴스에 대해 실행 중인지 확인합니다.

EC2 인스턴스에 대한 스캔 설정 구성에 대한 자세한 내용은 섹션을 참조하세요Amazon EC2 인스턴스 스캔.

Amazon ECR 리포지토리의 적용 범위 평가

리포지토리 탭에는 환경 AWS 의 Amazon 리ECR포지토리가 표시됩니다. 목록은 다음 탭에서 그룹으로 구성되어 있습니다.

- 모두 - 사용자 환경에 있는 모든 리포지토리가 표시됩니다. 상태 열은 리포지토리의 현재 스캔 상태를 나타냅니다.
- 활성화됨 – Amazon Inspector가 사용자 환경에서 모니터링 및 스캔하도록 구성된 모든 리포지토리가 표시됩니다. 상태 열은 리포지토리의 현재 스캔 상태를 나타냅니다.
- 활성화 안됨 – Amazon Inspector가 사용자 환경에서 모니터링 및 스캔하지 않는 모든 리포지토리가 표시됩니다. 이유 열은 Amazon Inspector가 리포지토리를 모니터링 및 스캔하지 않는 이유를 나타냅니다.

각 탭에서 계정 열은 리포지토리를 소유 AWS 계정 한 를 지정합니다.

리포지토리에 대한 추가 세부 정보를 검토하려면 리포지토리 이름을 선택하세요. 그러면 Amazon Inspector가 리포지토리에 있는 컨테이너 이미지 목록과 각 이미지에 대한 세부 정보를 표시합니다. 세부 정보에는 이미지 태그, 이미지 다이제스트 및 스캔 상태가 포함됩니다. 또한 이미지의 중요 결과 수와 같은 주요 결과 통계도 포함됩니다. 결과 통계에 대한 지원 데이터를 드릴다운하여 검토하려면 이미지의 이미지 태그를 선택하세요.

Amazon ECR리포지토리의 상태 값 스캔

Amazon Elastic Container Registry(AmazonECR) 리포지토리의 경우 가능한 상태 값은 다음과 같습니다.

- 활성화됨(연속) - 리포지토리의 경우 Amazon Inspector는 이 리포지토리의 이미지를 지속적으로 모니터링합니다. 리포지토리의 고급 스캔 설정이 연속 스캔으로 설정되어 있습니다. Amazon Inspector는 처음에 새 이미지를 푸시할 때 스캔하고 해당 이미지와 CVE 관련된 새 이미지가 게시되

면 이미지를 다시 스캔합니다. Amazon Inspector는 구성한 ECR 스캔 기간 동안 이 리포지토리의 이미지를 계속 모니터링합니다.

- 활성화됨(푸시 시) - Amazon Inspector는 새 이미지를 푸시할 때 리포지토리의 개별 컨테이너 이미지를 자동으로 스캔합니다. 리포지토리에 대해 향상된 스캔이 활성화되고 푸시 시 스캔으로 설정됩니다.
- 액세스 거부됨 – Amazon Inspector에서 리포지토리 또는 리포지토리에 있는 컨테이너 이미지에 액세스할 수 없습니다.

  이 문제를 해결하려면 리포지토리에 대한 AWS Identity and Access Management (IAM) 정책에서 Amazon Inspector가 리포지토리에 액세스할 수 있도록 허용해야 합니다.
- 비활성화됨(수동) – Amazon Inspector에서 리포지토리에 있는 컨테이너 이미지를 모니터링하거나 스캔하지 않습니다. 리포지토리의 Amazon ECR 스캔 설정은 기본 수동 스캔으로 설정됩니다.

  Amazon Inspector로 리포지토리 이미지 스캔을 시작하려면 리포지토리의 스캔 설정을 고급 스캔으로 변경한 다음 이미지를 지속적으로 스캔할지 아니면 새 이미지가 푸시될 때만 스캔할지 여부를 선택하세요.
- 활성화됨(푸시 시) - Amazon Inspector는 새 이미지를 푸시할 때 리포지토리의 개별 컨테이너 이미지를 자동으로 스캔합니다. 리포지토리의 고급 스캔 설정이 푸시할 때 스캔으로 설정되어 있습니다.
- 내부 오류 - Amazon Inspector가 리포지토리를 스캔하려고 할 때 내부 오류가 발생했습니다. Amazon Inspector에서 자동으로 오류를 해결하고 가능한 한 빨리 스캔을 재개합니다.

리포지토리에 대한 스캔 설정 구성에 대한 자세한 내용은 을 참조하세요Amazon ECR 컨테이너 이미지 스캔.

Amazon ECR 컨테이너 이미지의 적용 범위 평가

이미지 탭에는 환경 AWS 의 Amazon ECR 컨테이너 이미지가 표시됩니다. 목록은 다음 탭에서 그룹으로 구성되어 있습니다.

- 모두 – 사용자 환경에 있는 모든 컨테이너 이미지가 표시됩니다. 상태 열은 이미지의 현재 스캔 상태를 나타냅니다.
- 스캔 – Amazon Inspector가 사용자 환경에서 모니터링 및 스캔하도록 구성된 모든 컨테이너 이미지가 표시됩니다. 상태 열은 이미지의 현재 스캔 상태를 나타냅니다.
- 스캔하지 않음 – Amazon Inspector가 사용자 환경에서 모니터링 및 스캔하지 않는 모든 컨테이너 이미지가 표시됩니다. 이유 열은 Amazon Inspector가 이미지를 모니터링 및 스캔하지 않는 이유를 나타냅니다.

여러 가지 이유로 컨테이너 이미지가 활성화되지 않음 탭에 표시될 수 있습니다. 이미지는 Amazon Inspector가 스캔하는 리포지토리가 활성화되지 않은 리포지토리에 저장되거나 Amazon ECR 필터링 규칙으로 인해 해당 리포지토리가 스캔되지 않을 수 있습니다. 또는 ECR 재스캔 기간 에 대해 구성된 일 수 내에 이미지를 푸시하거나 가져오지 않았습니다. 자세한 내용은 ECR 재스캔 기간 구성 단원을 참조하십시오.

각 탭의 리포지토리 이름 열은 컨테이너 이미지를 저장하는 리포지토리의 이름을 지정합니다. 계정 열은 리포지토리를 소유 AWS 계정 하는 를 지정합니다. 마지막 스캔 열에는 Amazon Inspector가 해당 리소스의 취약성을 마지막으로 검사한 시간이 표시됩니다. 여기에는 메타데이터를 찾기 위한 업데이트가 있는 경우, 리소스의 애플리케이션 인벤토리가 업데이트되는 경우 또는 새 에 대한 응답으로 재스캔이 수행되는 경우가 포함될 수 있습니다CVE. 자세한 내용은 Amazon 스캔을 위한 ECR 스캔 동작 단원을 참조하십시오.

컨테이너 이미지에 대한 추가 세부 정보를 검토하려면 ECR 컨테이너 이미지 열에서 링크를 선택합니다. 그러면 Amazon Inspector에서 이미지에 대한 세부 정보와 이미지에 대한 현재 결과를 표시합니다. 결과에 대한 세부 정보를 검토하려면 제목 열의 링크를 선택하세요. 이러한 세부 정보에 대한 자세한 내용은 Amazon Inspector 조사 결과에 대한 세부 정보 보기 섹션을 참조하세요.

Amazon ECR 컨테이너 이미지의 상태 값 스캔

Amazon Elastic Container Registry 컨테이너 이미지의 경우 가능한 상태 값은 다음과 같습니다.

- 능동적 모니터링(연속적) - Amazon Inspector는 지속적으로 모니터링하고 있으며 관련 이미지가 새로 게시될 때마다 이미지 및 새 스캔CVE이 수행됩니다. 이미지를 밀거나 당길 때마다 이미지의 Amazon ECR 재스캔 기간이 새로 고쳐집니다. 이미지를 저장하는 리포지토리에 대해 고급 스캔이 활성화되고 리포지토리의 고급 스캔 설정이 연속 스캔으로 설정되어 있습니다.
- 활성화됨(푸시 시) - Amazon Inspector는 새 이미지를 누를 때마다 이미지를 자동으로 스캔합니다. 이미지를 저장하는 리포지토리에 대해 고급 스캔이 활성화되고 리포지토리의 고급 스캔 설정이 푸시할 때 스캔으로 설정되어 있습니다.
- 내부 오류 - Amazon Inspector가 컨테이너 이미지를 스캔하려고 할 때 내부 오류가 발생했습니다. Amazon Inspector에서 자동으로 오류를 해결하고 가능한 한 빨리 스캔을 재개합니다.
- 초기 스캔 보류 중 - Amazon Inspector가 초기 스캔을 위해 이미지를 대기열에 추가했습니다.
- 스캔 자격 만료됨(연속) - Amazon Inspector가 이미지에 대한 스캔을 일시 중지했습니다. 리포지토리 이미지의 자동 재스캔 기간 내에 이미지가 업데이트되지 않았습니다. 이미지를 밀거나 당겨 스캔을 재개할 수 있습니다.

- 스캔 적합성 만료됨(푸시 시) - Amazon Inspector가 이미지에 대한 스캔을 일시 중지했습니다. 리포지토리 이미지의 자동 재스캔 기간 내에 이미지가 업데이트되지 않았습니다. 이미지를 눌러 스캔을 재개할 수 있습니다.
- 스캔 빈도 설명서(수동) - Amazon Inspector는 Amazon ECR 컨테이너 이미지를 스캔하지 않습니다. 이미지를 저장하는 리포지토리의 Amazon ECR 스캔 설정은 기본 수동 스캔으로 설정됩니다. Amazon Inspector로 리포지토리 이미지 자동 스캔을 시작하려면 리포지토리 설정을 고급 스캔으로 변경한 다음 이미지를 지속적으로 스캔할지 아니면 새 이미지가 푸시될 때만 스캔할지 여부를 선택하세요.
- 지원되지 않는 OS - Amazon Inspector는 이미지를 모니터링하거나 스캔하지 않습니다. 이미지가 Amazon Inspector에서 지원하지 않는 운영 체제를 기반으로 하거나 Amazon Inspector에서 지원하지 않는 미디어 유형을 사용합니다.

  Amazon Inspector에서 지원하는 운영 체제 목록은 ??? 섹션을 참조하세요. Amazon Inspector에서 지원하는 미디어 유형 목록은 지원되는 미디어 유형을 참조하세요.

리포지토리 및 이미지의 스캔 설정 구성에 대한 자세한 내용은 Amazon ECR 컨테이너 이미지 스캔 섹션을 참조하세요.

AWS Lambda 함수 범위 평가

Lambda 탭에는 AWS 환경의 Lambda 함수가 표시됩니다. 이 페이지에는 두 개의 테이블이 있는데, 하나는 Lambda 표준 스캔에 대한 함수 적용 범위 세부 정보를 보여주고 다른 하나는 Lambda 코드 스캔에 대한 함수 적용 범위 세부 정보를 보여줍니다. 다음 탭을 기준으로 함수를 그룹화할 수 있습니다.

- 모두 - 사용자 환경에 있는 모든 Lambda 함수가 표시됩니다. 상태 열은 Lambda 함수의 현재 스캔 상태를 나타냅니다.
- 스캔 – Amazon Inspector가 스캔하도록 구성된 Lambda 함수가 표시됩니다. 상태 열은 각 Lambda 함수의 현재 스캔 상태를 나타냅니다.
- 스캔하지 않음 - Amazon Inspector가 스캔하도록 구성되지 않은 Lambda 함수가 표시됩니다. 이유 열은 Amazon Inspector가 함수를 모니터링 및 스캔하지 않는 이유를 나타냅니다.

  여러 가지 이유로 Lambda 함수가 스캔하지 않음 탭에 표시될 수 있습니다. Lambda 함수가 Amazon Inspector에 추가되지 않은 계정에 속해 있거나 필터링 규칙으로 인해 이 함수가 스캔되지 않을 수 있습니다. 자세한 내용은 Lambda 함수 스캔 단원을 참조하십시오.

각 탭의 함수 이름 열은 Lambda 함수의 이름을 지정합니다. 계정 열은 함수를 소유 AWS 계정 하는 를 지정합니다. 런타임은 함수의 런타임을 지정합니다. 상태 열은 각 Lambda 함수의 현재 스캔 상태를 나타냅니다. 리소스 태그에는 함수에 적용된 태그가 표시됩니다. 마지막 스캔 열에는 Amazon Inspector가 해당 리소스의 취약성을 마지막으로 검사한 시간이 표시됩니다. 여기에는 메타데이터를 찾기 위한 업데이트가 있는 경우, 리소스의 애플리케이션 인벤토리가 업데이트되는 경우 또는 새 에 대한 응답으로 재스캔이 수행되는 경우가 포함될 수 있습니다CVE. 자세한 내용은 Lambda 함수 스캔의 스캔 동작 단원을 참조하십시오.

AWS Lambda 함수의 상태 값 스캔

Lambda 함수의 경우 가능한 상태 값은 다음과 같습니다.

- 능동 모니터링 – Amazon Inspector에서 Lambda 함수를 지속적으로 모니터링하고 스캔합니다. 연속 스캔에는 새 함수가 리포지토리로 푸시될 때의 초기 스캔과 함수가 업데이트되거나 새 공통 취약성 및 노출(CVEs)이 릴리스될 때의 자동 재스캔이 포함됩니다.
- 태그를 기준으로 제외됨 – 해당 함수는 태그 기준 스캔에서 제외되었으므로 Amazon Inspector에서 스캔하지 않습니다.
- 스캔 자격 만료됨 – 해당 함수가 마지막으로 호출되거나 업데이트된 지 90일 이상 지났기 때문에 Amazon Inspector에서 이 함수를 모니터링하지 않습니다.
- 내부 오류 – Amazon Inspector에서 함수 스캔을 시도할 때 내부 오류가 발생했습니다. Amazon Inspector에서 자동으로 오류를 해결하고 가능한 한 빨리 스캔을 재개합니다.
- 첫 번째 스캔 보류 중 – Amazon Inspector에서 첫 번째 스캔을 위해 함수를 대기열에 추가했습니다.
- 지원되지 않음 – Lambda 함수에 지원되지 않는 런타임이 있습니다.

를 사용하여 Amazon Inspector에서 여러 계정 관리 AWS Organizations

Amazon Inspector를 사용하여 와 연결된 여러 계정을 관리할 수 있습니다AWS Organizations. 여러 Amazon Inspector 계정을 관리하기 위해 Organizations 관리 계정은 계정을 Amazon Inspector 의 위임된 관리자 계정으로 지정합니다. 위임된 관리자 계정은 조직의 Amazon Inspector를 관리하고 조직을 대신하여 작업을 수행할 수 있는 특별한 권한을 부여합니다. 이러한 작업에는 멤버 계정에 대한 스캔 활성화 또는 비활성화, 금지 규칙 생성 및 관리, 전체 조직에 대한 집계 결과 보기가 포함됩니다.

> **Note**
>
> 여러 의 여러 계정에 대해 프로그래밍 방식으로 Amazon Inspector를 활성화하려면 Amazon Inspector가 개발한 쉘 스크립트를 사용할 AWS 리전수 있습니다. 자세한 내용은 GitHub 웹 사이트의 Inspector2-enablement-with-cli를 참조하세요.

주제

- Amazon Inspector의 위임된 관리자 계정 및 멤버 계정 이해
- Amazon Inspector의 위임된 관리자 계정 지정

Amazon Inspector의 위임된 관리자 계정 및 멤버 계정 이해

다중 계정 환경에서 Amazon Inspector를 사용하는 경우 위임된 관리자 계정은 특정 메타데이터에 액세스할 수 있습니다. 메타데이터에는 Amazon EC2, Amazon ECR, Lambda에 대한 표준 스캔과 Lambda 코드 스캔이 포함됩니다. 또한 멤버 계정에 대한 보안 검색 결과도 포함되어 있습니다. 이 섹션에서는 위임된 관리자 계정이 수행할 수 있는 작업과 멤버 계정이 수행할 수 있는 작업에 대한 정보를 제공합니다.

위임 관리자 작업

일반적으로 위임된 관리자가 계정에 설정을 적용하면 해당 설정은 조직의 다른 모든 계정에 적용됩니다. 또한 위임 관리자는 자신의 계정 및 연결된 멤버의 정보를 보고 검색할 수 있습니다. Amazon Inspector 위임 관리자 계정은 다음 작업을 수행할 수 있습니다.

- Amazon Inspector 활성화 및 비활성화를 포함하여 관련 계정의 Amazon Inspector 상태를 확인하고 관리합니다.

- 조직 내 모든 멤버 계정의 스캔 유형을 활성화하거나 비활성화합니다.
- 조직 전체에서 집계된 결과 데이터와 조직 내 모든 멤버 계정에 대한 결과 세부 정보를 확인합니다.
- 조직 내 모든 계정의 결과에 적용될 억제 규칙을 생성하고 관리합니다.
- 조직의 모든 구성원에 대해 Amazon ECR 고급 스캔을 활성화합니다.
- 전체 조직의 리소스 적용 범위를 확인합니다.
- 조직의 모든 멤버 계정에 대한 ECR 컨테이너 이미지의 자동 재스캔 기간을 정의합니다. 위임 관리자의 스캔 기간 설정이 이전에 멤버 계정이 설정한 모든 설정보다 우선합니다. 조직의 모든 계정은 위임된 관리자의 Amazon ECR 자동 재스캔 기간을 공유합니다. 개별 계정에 대해 다른 재스캔 기간을 설정할 수 없습니다.
- 조직의 모든 계정에서 EC2 사용할 Amazon Inspector 심층 검사를 위한 사용자 지정 경로 5개를 지정합니다. 이는 위임 관리자가 개별 계정에 대해 설정할 수 있는 5개의 사용자 지정 경로에 추가하여 지정하는 것입니다. 심층 검사 사용자 지정 경로 구성에 대한 자세한 내용은 섹션을 참조하세요Amazon Inspector 심층 검사를 위한 사용자 지정 경로.
- 멤버 계정에 대한 Amazon Inspector 심층 검사를 활성화 및 비활성화합니다.
- 조직의 멤버 계정에 대해 내보SBOMs냅니다.
- 조직의 모든 멤버 계정에 대해 Amazon EC2 스캔 모드를 설정합니다. 자세한 내용은 스캔 모드 관리 단원을 참조하십시오.
- 멤버 계정에서 생성한 CIS 스캔 구성을 제외한 조직의 모든 계정에 대한 스캔 구성을 생성하고 관리합니다.

> **Note**
>
> 멤버 계정이 조직을 떠날 경우 위임된 관리자는 더 이상 해당 계정에 예약된 스캔 구성을 볼 수 없습니다.

- 조직의 모든 계정에 대한 CIS 스캔 결과를 봅니다.

멤버 계정 작업

멤버 계정은 Amazon Inspector 에서 계정에 대한 정보를 보고 검색할 수 있으며, 계정 설정은 위임된 관리자가 관리합니다. 조직 내 멤버 계정은 Amazon Inspector에서 다음 작업을 수행할 수 있습니다.

- 본인 계정에 대해 Amazon Inspector를 활성화합니다.
- 본인 계정에 대한 리소스 적용 범위를 확인합니다.
- 본인 계정에 대한 결과 세부 정보를 확인합니다.

- 자체 계정에 대한 ECR 컨테이너 이미지 자동 재스캔 기간 설정을 봅니다.
- 개별 계정에 EC2 사용할 에 대한 Amazon Inspector 심층 검사를 위한 5개의 사용자 지정 경로를 지정합니다. 이러한 경로는 위임된 관리자가 조직에 대해 지정한 사용자 지정 경로 외에도 스캔됩니다. 심층 검사 경로 구성에 대한 자세한 내용은 섹션을 참조하세요Amazon Inspector 심층 검사를 위한 사용자 지정 경로.
- Amazon Inspector 심층 검사를 위해 위임된 관리자가 설정한 사용자 지정 경로를 확인합니다.
- 계정과 연결된 리소스에 대해 내보SBOMs냅니다.
- 자신의 계정의 스캔 모드를 확인합니다.
- 계정에 대한 CIS 스캔 구성을 생성하고 관리합니다.
- 위임된 관리자가 예약한 리소스를 포함하여 계정의 리소스에 대한 CIS 스캔 결과를 확인합니다.

> **Note**
>
> Amazon Inspector는 활성화 후 위임 관리자 계정을 통해서만 비활성화할 수 있습니다.

Amazon Inspector의 위임된 관리자 계정 지정

위임된 관리자 계정은 AWS 서비스의 모든 멤버 계정을 관리하도록 지정하는 계정입니다. Amazon Inspector의 위임된 관리자 계정은 특정 메타데이터에 액세스할 수 있습니다. 자세한 내용은 위임된 관리자 계정 및 멤버 계정 Amazon Inspector 이해를 참조하세요. 이 섹션에서는 Amazon Inspector 에 대해 위임된 관리자를 지정하는 방법을 설명합니다.

위임 관리자에 대한 중요 고려 사항

다음은 Amazon Inspector에서 위임 관리자의 운영 방식을 정의하는 요소입니다.

위임된 관리자는 최대 10,000명의 멤버를 관리할 수 있습니다.

> 각 Amazon Inspector 위임 관리자는 10,000개의 멤버 계정 할당량을 갖습니다. 그러나 조직에 10,000개 이상의 계정이 포함될 수 있습니다. 멤버 계정이 10,000개를 초과하면 Amazon CloudWatch Personal Health Dashboard를 통해 알림과 위임된 관리자 계정으로의 이메일을 받게 됩니다.

위임 관리자는 리전별로 결정됩니다.

와 달리 AWS Organizations Amazon Inspector는 리전 서비스입니다. 즉, Amazon Inspector를 사용 AWS 리전 하려는 각 에서 위임된 관리자를 지정하고, 멤버 계정을 추가하고, 스캔 유형을 활성화해야 합니다.

조직의 위임 관리자는 한 명입니다.

Amazon Inspector 위임 관리자는 조직당 한 명만 있을 수 있습니다. 한 리전에서 계정을 위임된 관리자로 지정한 경우 해당 계정은 다른 모든 리전에서 위임된 관리자여야 합니다.

위임 관리자를 변경해도 멤버 계정의 Amazon Inspector는 비활성화되지 않습니다.

위임된 관리자를 제거해도 Amazon Inspector는 해당 계정에서 비활성화되지 않으며 스캔 설정에 영향을 주지 않습니다.

AWS 조직에 모든 기능이 활성화되어 있어야 합니다.

의 기본 설정입니다 AWS Organizations. 활성화되지 않은 경우 조직의 모든 기능 활성화를 참조하세요.

위임 관리자를 지정하는 데 필요한 권한

Amazon Inspector를 활성화하고 Amazon Inspector 위임 관리자를 지정할 수 있는 권한이 있어야 합니다. IAM 정책 끝에 다음 문을 추가하여 이러한 권한을 부여합니다. 자세한 내용은 IAM 정책 관리를 참조하세요.

```
{
    "Sid": "PermissionsForInspectorAdmin",
    "Effect": "Allow",
    "Action": [
        "inspector2:EnableDelegatedAdminAccount",
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganizationalUnit",
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization"
    ],
    "Resource": "*"
```

```
}
```

AWS 조직의 위임된 관리자 지정

다음 절차에서는 AWS 조직의 위임된 관리자를 지정하는 방법을 보여줍니다. 이 지정이 완료되면 Organizations 관리 계정과 선택한 위임 관리자 계정 모두에 대해 Amazon Inspector가 활성화됩니다.

> **Note**
>
> Organizations 관리 계정만 위임 관리자를 지정할 수 있습니다.

Amazon Inspector를 처음 활성화하면 계정에 `AWSServiceRoleForAmazonInspector` 대한 서비스 연결 역할(SLR)이 생성됩니다. Amazon Inspector에서 서비스 연결 역할을 사용하는 방법에 대한 자세한 내용은 Amazon Inspector에 서비스 연결 역할 사용 섹션을 참조하세요. 일반적으로 서비스 연결 역할에 대한 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 사용을 참조하세요.

Amazon Inspector 위임 관리자를 지정하려면

Console

콘솔에서 위임 관리자 지정

1. AWS Organizations 관리 계정을 AWS Management Console 사용하여 에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
3. AWS 리전 선택기를 사용하여 위임된 관리자를 지정할 리전을 지정합니다.
4. 일반 설정 을 선택합니다.
5. 위임된 관리자 타일에서 위임된 관리자로 지정 AWS 계정 하려는 의 계정 ID를 입력한 다음 관리 위임을 선택합니다.
6. (선택 사항) 각 에 대해 이전 단계를 반복합니다 AWS 리전.

API

를 사용하여 위임된 관리자 지정 API

- AWS 계정 Organizations 관리 계정의 보안 인증 정보를 사용하여 EnableDelegatedAdminAccount API 작업을 실행합니다. AWS Command Line Interface 를 사용하여 다음 CLI 명령을 실행하여 이 작업을 수행할 수도 있습니다`aws inspector2`

```
enable-delegated-admin-account --delegated-admin-account-id
11111111111.
```

> **Note**
>
> Amazon Inspector 위임 관리자를 만들 계정의 계정 ID를 지정해야 합니다.

위임된 관리자를 지정한 후에는 관리 계정을 사용하여 AWS Organizations 위임된 관리자 계정을 변경하거나 제거해야 합니다.

멤버 계정에 대한 Amazon Inspector 스캔 활성화

조직의 위임된 관리자인 경우 조직의 멤버 계정에 대해 Amazon EC2 및 Amazon ECR 스캔을 활성화할 수 있습니다. 멤버 계정에 대한 스캔을 활성화하면 Amazon Inspector가 해당 계정에 대해 자동으로 활성화되고 계정이 위임된 관리자 계정과 연결됩니다. Amazon Inspector 스캔 유형에 대한 자세한 내용은 섹션을 참조하세요Amazon Inspector의 자동 스캔 유형. 이 섹션에서는 멤버 계정에 대한 스캔을 활성화하는 방법을 설명합니다.

멤버 계정 스캔 활성화

멤버 계정 스캔을 다양한 방식으로 활성화할 수 있습니다. 다음 절차에서는 위임된 관리자로서 모든 멤버 계정 및 특정 멤버 계정에 대한 스캔을 활성화하는 방법과 멤버 계정으로 스캔을 활성화하는 방법을 설명합니다.

모든 멤버 계정에 대한 스캔을 자동으로 활성화하려면

1. 위임된 관리자 계정 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 리전 선택기를 사용하여 모든 멤버 계정에 대해 스캔을 활성화할 AWS 리전 를 선택합니다.
3. 탐색 창에서 계정 관리 를 선택합니다. 계정 탭에는 AWS Organizations 관리 계정과 연결된 모든 멤버 계정이 표시됩니다.
4. 조직 에서 계정 번호 옆의 상자를 선택합니다. 그런 다음 활성화를 선택하여 멤버 계정에 적용할 스캔 옵션을 선택합니다. 다음 스캔 유형을 선택할 수 있습니다.

   - Amazon EC2 스캔
   - Amazon ECR 스캔
   - Lambda 표준 스캔

- Lambda 코드 스캔

- 원하는 스캔 유형을 선택한 후 저장을 선택합니다.

> **Note**
> 여러 페이지의 계정이 있는 경우 각 페이지에서 이 단계를 반복해야 합니다. 기어 아이콘을 선택하여 각 페이지에 표시되는 계정 수를 변경할 수 있습니다.

5. 새 멤버 계정에 대해 검사기 자동 활성화 설정을 켜고 조직에 추가된 새 멤버 계정에 적용할 스캔 옵션을 선택합니다. 다음 스캔 유형을 선택할 수 있습니다.

   - Amazon EC2 스캔
   - Amazon ECR 스캔
   - Lambda 표준 스캔
   - Lambda 코드 스캔

   - 원하는 스캔 유형을 선택한 후 활성화를 선택합니다.

   > **Note**
   > 새 멤버 계정에 대해 자동으로 검사기 활성화 설정을 사용하면 조직의 이후 모든 멤버에 대해 Amazon Inspector가 활성화됩니다.
   > 멤버 계정 수가 5,000개보다 많으면 이 설정이 자동으로 해제됩니다. 총 멤버 계정 수가 5,000개 미만으로 감소하면 설정이 자동으로 다시 활성화됩니다.

6. (권장) 멤버 계정 스캔을 활성화 AWS 리전 하려는 각 에서 이러한 각 단계를 반복합니다.

특정 멤버 계정에 대한 스캔을 활성화하려면

1. 위임된 관리자 계정 보안 인증 정보를 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 리전 선택기를 사용하여 모든 멤버 계정에 대해 스캔을 활성화할 AWS 리전 를 선택합니다.
3. 탐색 창에서 계정 관리 를 선택합니다. 계정 탭에는 AWS Organizations 관리 계정과 연결된 모든 멤버 계정이 표시됩니다.

4. 조직 에서 스캔을 활성화하려는 각 멤버 계정 번호 옆의 상자를 선택합니다. 그런 다음 활성화를 선택하여 멤버 계정에 적용할 스캔 옵션을 선택합니다. 다음 스캔 유형을 선택할 수 있습니다.

   - Amazon EC2 스캔
   - Amazon ECR 스캔
   - Lambda 표준 스캔
   - Lambda 코드 스캔

   - 원하는 스캔 유형을 선택한 후 저장을 선택합니다.

   > **Note**
   > 여러 페이지의 계정이 있는 경우 각 페이지에서 이 단계를 반복해야 합니다. 기어 아이콘을 선택하여 각 페이지에 표시되는 계정 수를 변경할 수 있습니다.

5. (권장) 특정 멤버에 대해 스캔을 활성화 AWS 리전 하려는 각 에서 이러한 각 단계를 반복합니다.

멤버 계정으로 스캔을 활성화하려면

1. 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 리전 선택기를 사용하여 모든 멤버 계정에 대해 스캔을 활성화할 AWS 리전 를 선택합니다.
3. 탐색 창에서 계정 관리 를 선택합니다. 계정 탭에는 AWS Organizations 관리 계정과 연결된 모든 멤버 계정이 표시됩니다.
4. 조직 에서 계정 번호 옆의 상자를 선택합니다. 그런 다음 활성화를 선택하여 적용할 스캔 옵션을 선택합니다. 다음 스캔 유형을 선택할 수 있습니다.

   - Amazon EC2 스캔
   - Amazon ECR 스캔
   - Lambda 표준 스캔
   - Lambda 코드 스캔

   - 원하는 스캔 유형을 선택한 후 저장을 선택합니다.

5. (권장) 멤버 계정에 대한 스캔을 활성화하려는 각 리전에서 이 단계를 반복합니다.

> **Note**
>
> AWS Organizations 관리 계정에 Amazon Inspector 에 대한 위임된 관리자 계정이 있는 경우 멤버 계정으로 계정을 활성화하여 스캔 세부 정보를 볼 수 있습니다.

Amazon Inspector에서 멤버 계정 연결 해제

위임된 관리자로서 계정에서 멤버 계정의 연결을 해제해야 할 수 있습니다. 멤버 계정의 연결을 해제하면 Amazon Inspector가 계정에서 여전히 활성화되고 계정이 독립 실행형 계정이 됩니다. 또한 계정에 대해 Amazon Inspector를 더 이상 관리할 수 있는 권한이 없습니다. 그러나 이전에 연결 해제된 멤버 계정을 언제든지 계정과 연결할 수 있습니다. 이 섹션에서는 멤버 계정을 위임된 관리자로 연결 해제하는 방법을 설명합니다.

Console

콘솔을 사용하여 멤버 계정의 연결을 해제하는 방법

1. 위임된 관리자 계정 자격 증명을 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/home에서 Amazon Inspector 콘솔을 엽니다.
2. 리전 선택기를 사용하여 멤버 계정의 연결을 해제할 AWS 리전 를 선택합니다.
3. 탐색 창에서 계정 관리 를 선택합니다.
4. 조직 에서 연결을 해제하려는 각 계정 번호 옆의 상자를 선택합니다.
5. 작업 메뉴를 선택한 다음 계정 연결 해제를 선택합니다.

API

를 사용하여 멤버 계정 연결을 해제하려면 API

DisassociateMember API 작업을 실행합니다. 요청에 연결 해제IDs하려는 계정을 입력합니다.

Amazon Inspector에서 위임된 관리자 제거

Amazon Inspector 위임 관리자 계정을 제거해야 할 수 있습니다. AWS Organizations 관리 계정에서 이 작업을 수행할 수 있습니다. Amazon Inspector 위임 관리자 계정을 제거해도 Amazon Inspector는 계정과 모든 멤버 계정에서 여전히 활성화됩니다. 위임된 관리자 계정과 모든 멤버 계정은 독립 실행형

계정이 되고 원래 스캔 설정을 유지합니다. 이 섹션에서는 위임된 관리자 계정을 제거하는 방법을 설명합니다.

Amazon Inspector 위임 관리자 제거

다음 절차에서는 Amazon Inspector 위임 관리자를 제거하는 방법과 위임 관리자 계정에서 멤버 계정을 연결하는 방법을 설명합니다.

Amazon Inspector 위임 관리자를 할당하는 방법에 대한 자세한 내용은 Amazon Inspector에 대한 위임 관리자 계정 지정을 참조하세요.

> **Note**
>
> Amazon Inspector 위임 관리자를 할당한 후 Amazon Inspector 위임 관리자는 멤버 계정을 수동으로 연결해야 합니다.

위임 관리자를 제거하려면

1. AWS Organizations 관리 계정을 AWS Management Console 사용하여 에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 엽니다.
3. 리전 선택기를 사용하여 위임된 관리자를 제거할 AWS 리전 를 선택합니다.
4. 탐색 창에서 일반 설정 을 선택합니다.
5. 위임된 관리자 에서 제거를 선택한 다음 작업을 확인합니다.

멤버를 새 위임 관리자와 연결하려면

1. 위임된 관리자 계정 보안 인증 정보를 사용하여 로그인한 다음 https://console.aws.amazon.com/inspector/v2/홈 에서 Amazon Inspector 콘솔을 엽니다.
2. 리전 선택기를 사용하여 멤버를 연결할 AWS 리전 을 선택합니다.
3. 탐색 창에서 계정 관리 를 선택합니다.
4. 조직 에서 계정 번호 옆의 상자를 선택합니다.
5. 작업을 선택하고 구성원 추가를 선택합니다.

Amazon Inspector에서 사용량 및 비용 모니터링

Amazon Inspector 콘솔을 사용하여 환경에 대한 월간 Amazon Inspector 비용을 API 예상할 수 있습니다. 다중 계정 환경의 Amazon Inspector 관리자인 경우 환경의 총 비용과 모든 구성원 계정의 비용 지표를 볼 수 있습니다. 이 섹션에서는 사용 통계에 액세스하고 사용 비용을 계산하는 방법을 설명합니다.

사용량 콘솔 사용

콘솔에서 Amazon Inspector의 사용량과 예상 비용을 평가할 수 있습니다.

사용량 통계에 액세스하려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 비용을 모니터링하려는 지역을 선택합니다.
3. 탐색 창에서 사용량을 선택합니다.

계정별 탭의 계정 사용량에는 30일 기간을 기준으로 예상되는 총 비용이 표시됩니다. 예상 비용 열 아래의 테이블에서 값을 선택하면 해당 계정의 스캔 유형별 사용 내역을 확인할 수 있습니다. 이 세부 정보 창에서는 해당 계정에 대해 무료 평가판이 활성화된 스캔 유형도 확인할 수 있습니다.

조직의 위임 관리자인 경우 조직 내 각 계정에 대한 행이 테이블에 표시됩니다. 조직의 계정 연결이 해제된 경우 콘솔에 예상 비용이 -로 표시됩니다.

스캔 유형별 탭에서는 현재 30일 동안의 실제 사용량 내역을 스캔 유형별로 확인할 수 있습니다. 이는 계정별 탭에서 예상 비용을 계산하는 데 사용되는 정보입니다.

조직의 위임 관리자인 경우 조직 내 각 계정에 대한 사용량을 확인할 수 있습니다.

이 탭에서 다음 창 중 하나를 확장하여 사용량 통계를 확인할 수 있습니다.

아마존 EC2 스캔

Amazon Inspector 사용 콘솔은 에이전트 기반 검사 및 에이전트 없는 검사에 대한 다음 지표를 추적합니다.

- 인스턴스 (평균) — Amazon Inspector는 적용 시간을 사용하여 인스턴스 스캔에 필요한 평균 리소스 EC2 수를 계산합니다. 평균은 총 적용 범위 시간을 720시간(30일 기간의 시간 수)으로 나눈 값입니다.
- 적용 시간 — Amazon EC2 스캔의 경우 Amazon Amazon Inspector가 계정 내 각 EC2 인스턴스에 대해 활성 적용 범위를 제공한 지난 30일 동안의 총 시간입니다. 예를 들어 적용 시간은 Amazon Inspector에서 EC2 인스턴스를 발견한 시점부터 종료 또는 중지되거나 태그별 검사에서 제외될 때까지의 시간입니다. (중지된 인스턴스를 다시 시작하거나 제외 태그를 제거하면 Amazon Inspector는 적용을 재개하고 해당 인스턴스에 대한 적용 시간은 계속 누적됩니다.)

CIS인스턴스 스캔 — 계정 내 인스턴스에 대해 수행된 총 CIS 스캔 수입니다.

아마존 ECR 스캔

최초 스캔 - 지난 30일 동안 계정의 이미지를 처음 스캔한 총 횟수입니다.

재스캔 - 지난 30일 동안 계정의 이미지를 재스캔한 총 횟수입니다. 재스캔은 Amazon Inspector에서 이전에 스캔한 ECR 이미지에 대해 수행한 모든 스캔입니다. 연속 스캔을 위해 ECR 리포지토리를 구성한 경우 Amazon Inspector가 데이터베이스에 새로운 일반적인 취약성 및 노출 CVE () 을 추가하면 자동으로 재검색이 수행됩니다.

Lambda 스캔

Amazon Inspector 사용 콘솔은 Lambda 표준 스캔 및 Lambda 코드 스캔에 대한 다음 지표를 추적합니다.

- Lambda 함수 수 (Avg) — Amazon Inspector는 커버리지 시간을 사용하여 Lambda 함수 스캔을 위한 평균 함수 수를 계산합니다. 평균은 총 적용 범위 시간을 720시간(30일 기간의 시간 수)으로 나눈 값입니다.
- 적용 범위 시간 - Lambda 함수 스캔의 경우, 지난 30일 동안 Amazon Inspector가 계정 내 각 Lambda 함수에 대해 활성 적용 범위를 제공한 총 시간입니다. AWS Lambda 함수의 경우 적용 범위 시간은 Amazon Inspector에서 함수를 발견한 시점부터 함수가 삭제되거나 검사에서 제외되는 시점까지 계산됩니다. 제외된 함수가 다시 포함되면 해당 함수에 대한 적용 범위 시간이 계속 누적됩니다.

Amazon Inspector의 사용 비용 계산 방식 이해

Amazon Inspector에서 제공하는 비용은 실제 비용이 아니라 추정치이므로 콘솔의 비용과 다를 수 있습니다. AWS Billing

사용량 페이지에서 Amazon Inspector가 비용을 계산하는 방법에 대해서는 다음 사항을 참고하세요.

- 사용 비용은 현재 리전에만 적용됩니다. 스캔 유형별 가격은 AWS 지역에 따라 다릅니다. 지역별 정확한 가격을 검토하려면 Amazon Inspector의 가격을 참조하십시오.
- 모든 사용량 예상치는 으로 가장 가까운 금액(미국 달러 기준)으로 반올림됩니다.
- 할인은 예상 비용에 포함되지 않습니다.
- 예상 비용은 스캔 유형별 30일 사용 기간 동안의 총 비용을 나타냅니다. 계정 사용 기간이 30일 미만인 경우, Amazon Inspector는 현재 적용되는 리소스가 남은 30일 동안 계속 적용될 것으로 간주하여 30일 이후의 비용을 예상합니다.
- 스캔 유형별 비용은 다음을 기준으로 계산됩니다.
  - EC2스캔: 비용은 지난 30일 동안 Amazon Inspector에서 처리한 평균 EC2 인스턴스 수를 반영합니다.
  - ECR컨테이너 스캔: 비용은 초기 이미지 스캔 횟수+지난 30일 동안의 이미지 재스캔 횟수의 합계를 반영합니다.
  - Lambda 표준 스캔: 비용에는 지난 30일 동안 Amazon Inspector에서 적용한 평균 Lambda 함수 수가 반영됩니다.
  - Lambda 코드 스캔: 비용에는 지난 30일 동안 Amazon Inspector에서 적용한 평균 Lambda 함수 수가 반영됩니다.

Amazon Inspector 무료 평가판 정보

Amazon Inspector에서는 각 스캔 유형에 무료 트레일이 있습니다. 스캔 유형을 활성화하면 해당 스캔 유형에 대해 15일 무료 평가판이 자동으로 등록됩니다. 무료 평가판이 시작되면 스캔 유형을 비활성화하더라도 15일 후에 자동으로 만료됩니다.

> **Note**
>
> 무료 평가판은 스캔에는 적용되지 CIS 않습니다.

Amazon Inspector의 보안

의 클라우드 보안 AWS 이 최우선 순위입니다. AWS 고객은 가장 보안에 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 이점을 누릴 수 있습니다.

보안은 AWS 와 사용자 간의 공동 책임입니다. 공동 책임 모델은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드 보안 - 에서 AWS 서비스를 실행하는 인프라를 보호할 AWS 책임이 있습니다 AWS 클라우드. AWS 또한 는 안전하게 사용할 수 있는 서비스를 제공합니다. 타사 감사자는 AWS 규정 준수 프로그램 규정 준수 일환으로 보안의 효과를 정기적으로 테스트하고 확인합니다. Amazon Inspector 에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 AWS 규정 준수 프로그램 서비스 범위 내 규정 .
- 클라우드의 보안 - 사용자의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 귀하는 귀사의 데이터의 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon Inspector를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목적에 맞게 Amazon Inspector를 구성하는 방법을 보여줍니다. 또한 Amazon Inspector 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법도 알아봅니다.

주제

- Amazon Inspector의 데이터 보호
- Amazon Inspector용 Identity and Access Management
- Amazon Inspector 모니터링
- Amazon Inspector의 규정 준수 검증
- Amazon Inspector의 복원성
- Amazon Inspector의 인프라 보안
- Amazon Inspector의 인시던트 대응

Amazon Inspector의 데이터 보호

AWS 공동 책임 모델 Amazon Inspector 의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든 를 실행하는 글로벌 인프라를 보호할 책임이 있습니다 AWS 클라우드. 사용자는 인프라에서 호스

팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스 의 보안 구성과 관리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시 섹션을 FAQ참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 AWS 책임 공유 모델 및 GDPR 블로그 게시물을 참조하세요.

데이터 보호를 위해 자격 증명을 보호하고 AWS 계정 AWS IAM Identity Center 또는 AWS Identity and Access Management ()를 사용하여 개별 사용자를 설정하는 것이 좋습니다IAM. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 다단계 인증(MFA)을 사용합니다.
- SSL/TLS를 사용하여 AWS 리소스와 통신합니다. TLS 1.2가 필요하며 TLS 1.3을 권장합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다 AWS CloudTrail. CloudTrail 추적을 사용하여 AWS 활동을 캡처하는 방법에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 CloudTrail 추적 작업을 참조하세요.
- AWS 암호화 솔루션과 내의 모든 기본 보안 제어를 사용합니다 AWS 서비스.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.
- 명령줄 인터페이스 또는 FIPS 를 AWS 통해 액세스할 때 140-3 검증 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 API사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 연방 정보 처리 표준(FIPS) 140-3을 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon Inspector 또는 콘솔, 또는 를 사용하는 기타 AWS 서비스 에서 작업하는 경우가 포함됩니다API AWS CLI AWS SDKs. 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL를 제공하는 경우 해당 서버에 대한 요청을 검증URL하기 위해 에 보안 인증 정보를 포함하지 않는 것이 좋습니다.

주제

- 저장 중 암호화
- 전송 중 암호화

저장 중 암호화

기본적으로 Amazon Inspector는 AWS 암호화 솔루션을 사용하여 저장 데이터를 저장합니다. Amazon Inspector는 다음과 같은 데이터를 암호화합니다.

- 로 수집된 리소스 인벤토리입니다 AWS Systems Manager.
- Amazon Elastic Container Registry 이미지에서 구문 분석된 리소스 인벤토리
- 에서 AWS 소유한 암호화 키를 사용하여 보안 조사 결과 생성 AWS Key Management Service

AWS 소유한 키는 관리, 사용 또는 볼 수 없습니다. 그러나 데이터를 암호화하는 키를 보호하기 위해 조치를 취하거나 프로그램을 변경할 필요는 없습니다. 자세한 내용은 AWS 소유 키 섹션을 참조하세요.

Amazon Inspector 를 비활성화하면 수집된 인벤토리 및 보안 조사 결과와 같이 저장하거나 유지 관리하는 모든 리소스가 영구적으로 삭제됩니다.

결과 코드에 대한 저장 중 암호화

Amazon Inspector Lambda 코드 스캔을 위해 Amazon Inspector는 와 협력하여 코드에 취약성 CodeGuru 이 있는지 스캔합니다. 취약성이 감지되면 취약성이 포함된 코드 조각을 CodeGuru 추출하고 Amazon Inspector가 액세스를 요청할 때까지 해당 코드를 저장합니다. 기본적으로 는 AWS 소유 키를 CodeGuru 사용하여 추출된 코드를 암호화하지만, Amazon Inspector가 암호화에 자체 고객 관리형 AWS KMS 키를 사용하도록 구성할 수 있습니다.

다음 워크플로우는 Amazon Inspector에서 사용자가 구성한 키를 사용하여 코드를 암호화하는 방법을 설명합니다.

1. Amazon Inspector 를 사용하여 Amazon Inspector에 AWS KMS 키를 제공합니다UpdateEncryptionKeyAPI.
2. Amazon Inspector는 AWS KMS 키에 대한 정보를 에 전달합니다 CodeGuru. 는 나중에 사용할 수 있도록 정보를 CodeGuru 저장합니다.
3. CodeGuru 는 Amazon Inspector 에서 구성한 키에 AWS KMS 대한 권한 부여를 에 요청합니다. Amazon Inspector
4. CodeGuru 는 키에서 암호화된 데이터 AWS KMS 키를 생성하고 저장합니다. 이 데이터 키는 에 저장된 코드 데이터를 암호화하는 데 사용됩니다 CodeGuru.
5. Amazon Inspector가 코드 스캔에서 데이터를 요청할 때마다 권한 부여를 CodeGuru 사용하여 암호화된 데이터 키를 복호화한 다음 해당 키를 사용하여 데이터를 복호화하여 검색할 수 있습니다.

Lambda 코드 스캔을 비활성화하면 권한 부여가 CodeGuru 사용 중지되고 연결된 데이터 키가 삭제됩니다.

고객 관리형 키를 사용한 코드 암호화에 대한 권한

암호화를 사용하려면 AWS KMS 작업에 대한 액세스를 허용하는 정책뿐만 아니라 Amazon Inspector에 조건 키를 통해 이러한 작업을 사용할 수 있는 CodeGuru 권한을 부여하는 문이 있어야 합니다.

계정의 암호화 키를 설정, 업데이트 또는 재설정하는 경우 Amazon Inspector 관리자 정책(예: AWS 관리형 정책: AmazonInspector2FullAccess)을 사용해야 합니다. 또한 암호화를 위해 선택한 키에 대한 결과 또는 데이터에서 코드 스니펫을 검색해야 하는 읽기 전용 사용자에게 다음 권한을 부여해야 합니다.

KMS의 경우 정책을 통해 다음 작업을 수행할 수 있어야 합니다.

- `kms:CreateGrant`
- `kms:Decrypt`
- `kms:DescribeKey`
- `kms:GenerateDataKeyWithoutPlainText`
- `kms:Encrypt`
- `kms:RetireGrant`

정책에 올바른 AWS KMS 권한이 있는지 확인한 후에는 Amazon Inspector를 허용하고 키를 암호화 CodeGuru 에 사용할 수 있는 문을 첨부해야 합니다. 다음 정책 설명을 첨부합니다.

> **Note**
>
> 리전을 Amazon Inspector Lambda 코드 스캔이 활성화된 AWS 리전으로 바꿉니다.

```
{
            "Sid": "allow CodeGuru Security to request a grant for a AWS KMS key",
         "Effect": "Allow",
         "Action": "kms:CreateGrant",
         "Resource": "*",
         "Condition": {
          "ForAllValues:StringEquals": {
```

```
            "kms:GrantOperations": [
             "GenerateDataKey",
             "GenerateDataKeyWithoutPlaintext",
             "Encrypt",
             "Decrypt",
             "RetireGrant",
             "DescribeKey"
            ]
           },
           "StringEquals": {
            "kms:ViaService": [
             "codeguru-security.Region.amazonaws.com"
            ]
           }
          }
         },
         {
          "Sid": "allow Amazon Inspector and CodeGuru Security to use your AWS KMS key",
          "Effect": "Allow",
          "Action": [
           "kms:Encrypt",
           "kms:Decrypt",
           "kms:RetireGrant",
           "kms:DescribeKey",
           "kms:GenerateDataKeyWithoutPlaintext"
          ],
          "Resource": "*",
          "Condition": {
           "StringEquals": {
            "kms:ViaService": [
             "inspector2.Region.amazonaws.com",
             "codeguru-security.Region.amazonaws.com"
            ]
           }
          }
         }
```

> **Note**
>
> 명령문을 추가할 때 구문이 올바른지 확인합니다. 정책은 JSON 형식을 사용합니다. 즉, 정책에 명령문을 추가하는 위치에 따라 명령문 앞이나 뒤에 쉼표를 추가해야 합니다. 명령문을 마지막 명령문으로 추가하는 경우 위 명령문의 닫는 괄호 뒤에 쉼표를 추가합니다. 명령문을 첫

번째 명령문으로 추가하거나 기존 두 명령문 사이에 추가하는 경우 명령문의 닫는 괄호 뒤에 쉼표를 추가합니다.

고객 관리형 키를 사용하여 암호화 구성

고객 관리형 키를 사용하여 계정 암호화를 구성하려면 고객 관리형 키를 사용한 코드 암호화에 대한 권한에 설명된 권한을 가진 Amazon Inspector 관리자여야 합니다. 또한 조사 결과와 동일한 AWS 리전에 있는 AWS KMS 키 또는 다중 리전 키 가 필요합니다. 계정의 기존 대칭 키를 사용하거나 AWS 관리 콘솔 또는 AWS KMS 를 사용하여 대칭 고객 관리형 키를 생성할 수 있습니다APIs. 자세한 내용은 AWS KMS 사용 설명서의 대칭 암호화 AWS KMS 키 생성을 참조하세요.

Amazon Inspector를 사용하여 암호화 API 구성

암호화를 위한 키를 설정하려면 Amazon Inspector 관리자로 로그인한 API 상태에서 Amazon Inspector의 UpdateEncryptionKey 작업을 설정합니다. API 요청에서 `kmsKeyId` 필드를 사용하여 사용하려는 AWS KMS 키ARN의 를 지정합니다. `scanType`에 `CODE`를 입력하고 `resourceType`에 `AWS_LAMBDA_FUNCTION`을 입력합니다.

UpdateEncryptionKey API 를 사용하여 Amazon Inspector가 암호화에 사용하는 AWS KMS 키를 확인할 수 있습니다.

> **Note**
>
> 고객 관리형 키를 설정하지 않은 `GetEncryptionKey` 상태에서 를 사용하려고 하면 작업이 `ResourceNotFoundException` 오류를 반환합니다. 즉, AWS 소유 키가 암호화에 사용되고 있음을 의미합니다.

또는 키를 삭제하거나 Amazon Inspector에 대한 액세스를 거부하는 정책을 변경하는 경우 코드 취약성 결과에 액세스할 CodeGuru 수 없으며 계정에 대한 Lambda 코드 스캔이 실패합니다.

`ResetEncryptionKey` 를 사용하여 AWS 소유 키를 사용하여 Amazon Inspector 조사 결과의 일부로 추출된 코드를 암호화할 수 있습니다.

전송 중 암호화

AWS 는 AWS 내부 시스템과 다른 AWS 서비스 간에 전송 중인 모든 데이터를 암호화합니다. 는 평가를 위해 Transport Layer Security(TLS) 보호 채널을 AWS 통해 고객 소유 EC2 인스턴스에서 원격

측정 데이터를 AWS Systems Manager 수집합니다. Security Hub로 전송되는 Amazon ECR 및 AWS Lambda 함수 스캔 결과는 TLS보호 채널을 사용하여 암호화됩니다. 자세한 내용은 Systems Manager의 데이터 보호를 참조하여 SSM 가 전송 중인 데이터를 암호화하는 방법을 알아보세요.

Amazon Inspector용 Identity and Access Management

AWS Identity and Access Management (IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어하는 데 도움이 AWS 서비스 되는 입니다. IAM 관리자는 Amazon Inspector 리소스를 사용할 수 있는 인증(로그인) 및 권한 부여(권한 보유) 대상을 제어합니다. IAM 는 추가 비용 없이 사용할 수 AWS 서비스 있는 입니다.

주제

- 고객
- ID를 통한 인증
- 정책을 사용한 액세스 관리
- Amazon Inspector의 작동 방식 IAM
- Amazon Inspector의 자격 증명 기반 정책 예
- AWS Amazon Inspector 관리형 정책
- Amazon Inspector에 서비스 연결 역할 사용
- Amazon Inspector 자격 증명 및 액세스 문제 해결

고객

사용 방법 AWS Identity and Access Management (IAM)은 Amazon Inspector 에서 수행하는 작업에 따라 다릅니다.

서비스 사용자 – Amazon Inspector 서비스를 사용하여 작업을 수행하는 경우 필요한 보안 인증 정보와 권한을 관리자가 제공합니다. 더 많은 Amazon SNS 기능을 사용하여 작업을 수행한다면 추가 권한이 필요할 수 있습니다. 액세스 권한 관리 방식을 이해하면 적절한 권한을 관리자에게 요청할 수 있습니다. Amazon Inspector의 기능에 액세스할 수 없다면 Amazon Inspector 자격 증명 및 액세스 문제 해결 섹션을 참조하세요.

서비스 관리자 – 회사에서 Amazon Inspector 리소스를 책임지고 있다면 Amazon Inspector에 대한 완전한 액세스 권한이 있을 것입니다. 서비스 관리자는 서비스 사용자가 액세스해야 하는 Amazon

Inspector 기능과 리소스를 결정합니다. 그런 다음 IAM 관리자에게 요청을 제출하여 서비스 사용자의 권한을 변경해야 합니다. 이 페이지의 정보를 검토하여 의 기본 개념을 이해합니다IAM. 회사에서 Amazon Inspector IAM 를 사용하는 방법에 대한 자세한 내용은 섹션을 참조하세요Amazon Inspector의 작동 방식 IAM.

IAM 관리자 - IAM 관리자인 경우 Amazon Inspector에 대한 액세스를 관리하기 위한 정책을 작성하는 방법에 대한 세부 정보를 알고 싶을 수 있습니다. 에서 사용할 수 있는 Amazon Inspector 자격 증명 기반 정책 예제를 보려면 섹션을 IAM참조하세요Amazon Inspector의 자격 증명 기반 정책 예.

ID를 통한 인증

인증은 자격 증명 AWS 으로 에 로그인하는 방법입니다. 로 AWS 계정 루트 사용자, IAM 사용자로 또는 IAM 역할을 수임하여 인증(에 로그인 AWS)되어야 합니다.

자격 증명 소스를 통해 제공된 자격 증명을 사용하여 에 페더레이션 자격 증명 AWS 으로 로그인할 수 있습니다. AWS IAM Identity Center (IAM Identity Center) 사용자, 회사의 Single Sign-On 인증 및 Google 또는 Facebook 자격 증명은 페더레이션 자격 증명의 예입니다. 페더레이션 자격 증명으로 로그인하면 관리자가 이전에 IAM 역할을 사용하여 자격 증명 페더레이션을 설정했습니다. 페더레이션을 사용하여 AWS 에 액세스하면 간접적으로 역할을 수임하게 됩니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. 에 로그인하는 방법에 대한 자세한 내용은 AWS 로그인 사용 설명서의 에 로그인하는 방법을 AWS 계정 AWS참조하세요.

AWS 프로그래밍 방식으로 에 액세스하는 경우 는 소프트웨어 개발 키트(SDK)와 명령줄 인터페이스(CLI)를 AWS 제공하여 자격 증명을 사용하여 요청에 암호화 방식으로 서명합니다. AWS 도구를 사용하지 않는 경우 직접 요청에 서명해야 합니다. 권장 방법을 사용하여 직접 요청에 서명하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 요청 서명을 참조하세요 AWS API.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, 다중 인증(MFA)을 사용하여 계정의 보안을 강화하는 것이 AWS 좋습니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서의 다중 인증 및 사용 설명서의 다중 인증 사용(MFA) AWS을 참조하세요IAM.

AWS 계정 루트 사용자

를 생성하면 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 하나의 로그인 자격 증명으로 AWS 계정시작합니다. 이 자격 증명을 AWS 계정 루트 사용자라고 하며 계정을 생성하는 데 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 일상적인 작업에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는

작업을 수행하는 데 사용합니다. 루트 사용자로 로그인해야 하는 작업의 전체 목록은 IAM 사용 설명서의 루트 사용자 보안 인증이 필요한 작업을 참조하세요.

페더레이션 자격 증명

가장 좋은 방법은 관리자 액세스가 필요한 사용자를 포함한 인간 사용자가 ID 공급자와의 페더레이션을 사용하여 임시 자격 증명을 사용하여 AWS 서비스 에 액세스하도록 요구하는 것입니다.

페더레이션 자격 증명은 엔터프라이즈 사용자 디렉터리, 웹 자격 증명 공급자, , AWS Directory Service Identity Center 디렉터리 또는 자격 증명 소스를 통해 제공된 자격 증명을 사용하여 AWS 서비스 에 액세스하는 모든 사용자의 사용자입니다. 페더레이션 ID가 에 액세스하면 역할을 AWS 계정수임하고 역할은 임시 자격 증명을 제공합니다.

중앙 집중식 액세스 관리를 위해 AWS IAM Identity Center(을)를 사용하는 것이 좋습니다. IAM Identity Center에서 사용자 및 그룹을 생성하거나 및 애플리케이션에서 사용할 수 있도록 자체 자격 증명 소스의 사용자 AWS 계정 및 그룹 집합에 연결하고 동기화할 수 있습니다. IAM Identity Center에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 IAM Identity Center란 무엇입니까?를 참조하세요.

IAM 사용자 및 그룹

IAM 사용자는 한 사람 또는 애플리케이션에 대한 특정 권한이 AWS 계정 있는 내 자격 증명입니다. 가능한 경우 암호 및 액세스 키와 같은 장기 보안 인증 정보가 있는 IAM 사용자를 생성하는 대신 임시 보안 인증 정보를 사용하는 것이 좋습니다. 그러나 IAM 사용자와 장기 보안 인증이 필요한 특정 사용 사례가 있는 경우 액세스 키를 교체하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 장기 보안 인증이 필요한 사용 사례에 대한 액세스 키 정기적으로 교체를 참조하세요.

IAM 그룹은 IAM 사용자 컬렉션을 지정하는 자격 증명입니다. 사용자는 그룹으로 로그인할 수 없습니다. 그룹을 사용하여 여러 사용자의 권한을 한 번에 지정할 수 있습니다. 그룹을 사용하면 대규모 사용자 집합의 권한을 더 쉽게 관리할 수 있습니다. 예를 들어 라는 이름의 그룹을 지정IAMAdmins하고 해당 그룹에 IAM 리소스를 관리할 수 있는 권한을 부여할 수 있습니다.

사용자는 역할과 다릅니다. 사용자는 한 사람 또는 애플리케이션과 고유하게 연결되지만, 역할은 해당 역할이 필요한 사람이라면 누구나 수임할 수 있습니다. 사용자는 영구적인 장기 보안 인증 정보를 가지고 있지만, 역할은 임시 보안 인증만 제공합니다. 자세한 내용은 IAM 사용 설명서의 (역할 대신) IAM 사용자를 생성할 시기를 참조하세요.

IAM 역할

IAM 역할은 특정 권한이 AWS 계정 있는 내 자격 증명입니다. IAM 사용자와 비슷하지만 특정 사람과는 연결되지 않습니다. IAM 역할을 전환 AWS Management Console 하여 에서 역할을 일시적으로 수임할 수 있습니다. https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_use_switch-role-

console.html 또는 AWS API 작업을 호출 AWS CLI 하거나 사용자 지정 를 사용하여 역할을 수임할 수 있습니다URL. 역할 사용 방법에 대한 자세한 내용은 IAM 사용 설명서의 IAM 역할 사용을 참조하세요.

IAM 임시 자격 증명이 있는 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 페더레이션 ID에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 페더레이션 ID가 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 역할에 대한 자세한 내용은 IAM 사용 설명서의 타사 자격 증명 공급자에 대한 역할 생성을 참조하세요. IAM Identity Center를 사용하는 경우 권한 세트를 구성합니다. 인증 후 자격 증명이 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 세트를 의 역할과 상호 연관시킵니다IAM. 권한 세트에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 권한 세트를 참조하세요.
- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할은 특정 작업에 대해 다른 권한을 일시적으로 맡을 IAM 역할을 수 있습니다.
- 교차 계정 액세스 - IAM 역할을 사용하여 다른 계정의 누군가(신뢰할 수 있는 보안 주체)가 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 계정 간 액세스를 부여하는 기본적인 방법입니다. 그러나 일부 에서는 정책을 리소스에 직접 연결할 AWS 서비스수 있습니다(역할을 프록시로 사용하는 대신). 크로스 계정 액세스에 대한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용 설명서의 에서 크로스 계정 리소스 액세스를 IAM 참조하세요.
- 교차 서비스 액세스 - 일부 는 다른 에서 기능을 AWS 서비스 사용합니다 AWS 서비스. 예를 들어 서비스에서 호출할 때 해당 서비스가 Amazon에서 애플리케이션을 실행EC2하거나 Amazon S3에 객체를 저장하는 것이 일반적입니다. 서비스는직접적으로 호출하는 보안 주체의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 작업을 수행할 수 있습니다.
  - 전달 액세스 세션(FAS) - IAM 사용자 또는 역할을 사용하여 에서 작업을 수행하면 보안 주체로 AWS간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS 는 를 호출하는 보안 주체의 권한을 다운스트림 서비스에 AWS 서비스 대한 요청과 AWS 서비스함께 사용합니다. FAS 요청은 서비스가 다른 AWS 서비스 또는 리소스와의 상호 작용을 완료해야 하는 요청을 수신할 때만 수행됩니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 액세스 세션 전달을 참조하세요.
  - 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하기 위해 수임하는 IAM 역할입니다. IAM 관리자는 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다IAM. 자세한 내용은 IAM 사용 설명서의 에 권한을 위임할 역할 생성을 AWS 서비스 참조하세요.
  - 서비스 연결 역할 - 서비스 연결 역할은 에 연결된 서비스 역할의 한 유형입니다 AWS 서비스. 서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 에 표시 AWS 계정 되며 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할에 대한 권한을 볼 수 있지만 편집할 수는 없습니다.

- Amazon에서 실행되는 애플리케이션 EC2 - IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS CLI 또는 AWS API 요청을 수행하는 애플리케이션의 임시 보안 인증을 관리할 수 있습니다. 이는 EC2 인스턴스 내에 액세스 키를 저장하는 것보다 좋습니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행 중인 프로그램이 임시 보안 인증을 가져올 수 있습니다. 자세한 내용은 IAM 사용 설명서EC2의 IAM 역할 사용을 참조하세요.

IAM 역할 또는 IAM 사용자를 사용할지 여부를 알아보려면 IAM 사용 설명서의 IAM 역할 생성 시기(사용자 대신)를 참조하세요.

정책을 사용한 액세스 관리

정책을 AWS 생성하고 AWS 자격 증명 또는 리소스에 연결하여 의 액세스를 제어합니다. 정책은 자격 증명 또는 리소스와 연결된 AWS 경우 권한을 정의하는 의 객체입니다. 는 보안 주체(사용자, 루트 사용자 또는 역할 세션)가 요청할 때 이러한 정책을 AWS 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는 지를 결정합니다. 대부분의 정책은 에 JSON 문서 AWS 로 저장됩니다. JSON 정책 문서의 구조 및 내용에 대한 자세한 내용은 IAM 사용 설명서의 JSON 정책 개요를 참조하세요.

관리자는 정책을 사용하여 AWS JSON 대상에 액세스할 수 있는 사용자를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

기본적으로, 사용자와 역할에는 어떠한 권한도 없습니다. 사용자에게 필요한 리소스에 대한 작업을 수행할 수 있는 권한을 부여하기 위해 IAM 관리자는 IAM 정책을 생성할 수 있습니다. 그런 다음 관리자는 IAM 정책을 역할에 추가하고 사용자는 역할을 수임할 수 있습니다.

IAM 정책은 작업을 수행하는 데 사용하는 방법에 관계없이 작업에 대한 권한을 정의합니다. 예를 들어, `iam:GetRole` 작업을 허용하는 정책이 있다고 가정합니다. 해당 정책을 사용하는 사용자는 AWS Management Console, AWS CLI또는 에서 역할 정보를 가져올 수 있습니다 AWS API.

보안 인증 기반 정책

자격 증명 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 자격 증명에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

보안 인증 기반 정책은 인라인 정책 또는 관리형 정책으로 한층 더 분류할 수 있습니다. 인라인 정책은 단일 사용자, 그룹 또는 역할에 직접 포함됩니다. 관리형 정책은 의 여러 사용자, 그룹 및 역할에 연결할

수 있는 독립 실행형 정책입니다 AWS 계정. 관리형 정책에는 AWS 관리형 정책 및 고객 관리형 정책이 포함됩니다. 관리형 정책 또는 인라인 정책 중에서 선택하는 방법을 알아보려면 IAM 사용 설명서의 관리형 정책 및 인라인 정책 선택을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 문서입니다. 리소스 기반 정책의 예로는 IAM 역할 신뢰 정책 및 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는 가 포함될 수 있습니다 AWS 서비스.

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 정책IAM에서는 의 AWS 관리형 정책을 사용할 수 없습니다.

액세스 제어 목록(ACLs)

액세스 제어 목록(ACLs)은 리소스에 액세스할 수 있는 권한이 있는 보안 주체(계정 멤버, 사용자 또는 역할)를 제어합니다. ACLs 는 리소스 기반 정책과 유사하지만 JSON 정책 문서 형식을 사용하지는 않습니다.

Amazon S3 AWS WAF및 AmazonVPC은 를 지원하는 서비스의 예입니다ACLs. 에 대한 자세한 내용은 Amazon Simple Storage Service 개발자 안내서의 액세스 제어 목록(ACL) 개요를 ACLs참조하세요.

기타 정책 타입

AWS 는 덜 일반적인 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 유형에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 - 권한 경계는 자격 증명 기반 정책이 IAM엔터티(IAM 사용자 또는 역할)에 부여할 수 있는 최대 권한을 설정하는 고급 기능입니다. 개체에 대한 권한 경계를 설정할 수 있습니다. 그 결과로 얻는 권한은 객체의 자격 증명 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 내용은 IAM 사용 설명서의 IAM엔터티에 대한 권한 경계를 참조하세요.
- 서비스 제어 정책(SCPs) - 의 조직 또는 조직 단위(OU)에 대한 최대 권한을 지정하는 SCPs JSON 정책입니다 AWS Organizations. AWS Organizations 는 비즈니스가 소유 AWS 계정 한 여러 을 그룹화하고 중앙에서 관리하기 위한 서비스입니다. 조직의 모든 기능을 활성화하면 서비스 제어 정책(SCPs)을 모든 계정에 적용할 수 있습니다. 는 각 를 포함하여 멤버 계정의 엔터티에 대한 권한을

SCP 제한합니다 AWS 계정 루트 사용자. 조직 및 에 대한 자세한 내용은 AWS Organizations 사용 설명서의 서비스 제어 정책을 SCPs참조하세요.

- 세션 정책 – 세션 정책은 역할 또는 페더레이션 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할의 보안 인증 기반 정책의 교차와 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 내용은 IAM 사용 설명서의 세션 정책을 참조하세요.

여러 정책 유형

여러 정책 유형이 요청에 적용되는 경우, 결과 권한은 이해하기가 더 복잡합니다. AWS 에서 여러 정책 유형이 관련될 때 요청을 허용할지 여부를 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하세요.

Amazon Inspector의 작동 방식 IAM

IAM 를 사용하여 Amazon Inspector 에 대한 액세스를 관리하기 전에 Amazon Inspector 에서 사용할 수 있는 IAM 기능에 대해 알아봅니다.

IAM Amazon Inspector와 함께 사용할 수 있는 기능

IAM 기능	Amazon Inspector 지원
ID 기반 정책	예
리소스 기반 정책	아니요
정책 작업	예
정책 리소스	예
정책 조건 키(서비스별)	예
ACLs	아니요
ABAC (정책의 태그)	부분
임시 보안 인증	예
보안 주체 권한	예

IAM 기능	Amazon Inspector 지원
서비스 역할	아니요
서비스 링크 역할	예

Amazon Inspector 및 기타 가 대부분의 IAM 기능을 어떻게 AWS 서비스 사용하는지 자세히 알아보려면 IAM 사용 설명서의 에서 AWS 서비스 를 IAM 참조하십시오.

Amazon Inspector의 자격 증명 기반 정책

ID 기반 정책 지원: 예

자격 증명 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 자격 증명에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

IAM 자격 증명 기반 정책을 사용하면 허용되거나 거부된 작업 및 리소스와 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. 보안 인증 기반 정책에서는 보안 주체가 연결된 사용자 또는 역할에 적용되므로 보안 주체를 지정할 수 없습니다. JSON 정책에서 사용할 수 있는 모든 요소에 대해 알아보려면 IAM 사용 설명서의 IAM JSON 정책 요소 참조를 참조하세요.

Amazon Inspector의 자격 증명 기반 정책 예

Amazon Inspector 자격 증명 기반 정책 예제를 보려면 Amazon Inspector의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon Inspector 내의 리소스 기반 정책

리소스 기반 정책 지원: 아니요

리소스 기반 정책은 리소스에 연결하는 JSON 정책 문서입니다. 리소스 기반 정책의 예로는 IAM 역할 신뢰 정책 및 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는 가 포함될 수 있습니다 AWS 서비스.

교차 계정 액세스를 활성화하려면 리소스 기반 정책의 보안 주체로 전체 계정 또는 다른 계정의 IAM엔터티를 지정할 수 있습니다. 리소스 기반 정책에 크로스 계정 보안 주체를 추가하는 것은 트러스트 관계 설정의 절반밖에 되지 않는다는 것을 유념하십시오. 보안 주체와 리소스가 다른 에 있는 경우 신뢰할 수 AWS 계정있는 계정의 IAM 관리자는 보안 주체 엔터티(사용자 또는 역할)에게 리소스에 액세스할 수 있는 권한도 부여해야 합니다. 엔터티에 ID 기반 정책을 연결하여 권한을 부여합니다. 하지만 리소스 기반 정책이 동일 계정의 보안 주체에 액세스를 부여하는 경우, 추가 자격 증명 기반 정책이 필요하지 않습니다. 자세한 내용은 IAM 사용 설명서의 에서 크로스 계정 리소스 액세스를 IAM 참조하세요.

Amazon Inspector에 대한 정책 작업

정책 작업 지원: 예

관리자는 정책을 사용하여 AWS JSON 대상에 액세스할 수 있는 사용자를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

JSON 정책의 `Action` 요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 정책 작업은 일반적으로 연결된 AWS API 작업과 이름이 동일합니다. 일치하는 API 작업이 없는 권한 전용 작업과 같은 몇 가지 예외가 있습니다. 정책에서 여러 작업이 필요한 몇 가지 작업도 있습니다. 이러한 추가 작업을 일컬어 종속 작업이라고 합니다.

연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하십시오.

Amazon Inspector 작업 목록을 보려면 서비스 승인 참조에서 Amazon Inspector에서 정의한 작업을 참조하세요.

Amazon Inspector의 정책 작업은 작업 앞에 다음 접두사를 사용합니다.

```
inspector2
```

단일 문에서 여러 작업을 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": [
      "inspector2:action1",
      "inspector2:action2"
         ]
```

Amazon Inspector 자격 증명 기반 정책 예제를 보려면 Amazon Inspector의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon Inspector에 대한 정책 리소스

정책 리소스 지원: 예

관리자는 정책을 사용하여 AWS JSON 대상에 액세스할 수 있는 사용자를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Resource` JSON 정책 요소는 작업이 적용되는 객체를 지정합니다. 문장에는 `Resource`또는 `NotResource`요소가 반드시 추가되어야 합니다. 가장 좋은 방법은 Amazon 리소스 이름(ARN)을 사용하여 리소스를 지정하는 것입니다. 리소스 수준 권한이라고 하는 특정 리소스 유형을 지원하는 작업에 대해 이 태스크를 수행할 수 있습니다.

작업 나열과 같이 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": "*"
```

Amazon Inspector 리소스 유형 및 해당 의 목록을 보려면 서비스 승인 참조의 Amazon Inspector에서 정의한 리소스를 ARNs참조하세요. 각 리소스ARN의 를 지정할 수 있는 작업을 알아보려면 Amazon Inspector에서 정의한 작업을 참조하세요.

Amazon Inspector 자격 증명 기반 정책 예제를 보려면 Amazon Inspector의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon Inspector에 사용되는 정책 조건 키

서비스별 정책 조건 키 지원: 예

관리자는 정책을 사용하여 AWS JSON 대상에 액세스할 수 있는 사용자를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Condition` 요소(또는 `Condition` 블록)를 사용하면 정책이 발효되는 조건을 지정할 수 있습니다. `Condition` 요소는 옵션입니다. 같거나 작음과 같은 조건 연산자를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다.

한 문에서 여러 `Condition` 요소를 지정하거나 단일 `Condition` 요소에서 여러 키를 지정하는 경우, AWS 는 논리적 `AND` 태스크를 사용하여 평가합니다. 단일 조건 키에 여러 값을 지정하는 경우 는 논리

적 OR 작업을 사용하여 조건을 AWS 평가합니다. 명문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어 리소스에 IAM 사용자 IAM 이름으로 태그가 지정된 경우에만 사용자에게 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 IAM 정책 요소: 변수 및 태그를 참조하세요.

AWS 는 전역 조건 키 및 서비스별 조건 키를 지원합니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 AWS 전역 조건 컨텍스트 키를 참조하세요.

Amazon Inspector 조건 키 목록을 보려면 서비스 승인 참조에서 Amazon Inspector에 사용되는 조건 키를 참조하세요. 조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 Amazon Inspector에서 정의한 작업을 참조하세요.

Amazon Inspector 자격 증명 기반 정책 예제를 보려면 Amazon Inspector의 자격 증명 기반 정책 예 섹션을 참조하세요.

ACLs Amazon Inspector의

지원 ACLs: 아니요

액세스 제어 목록(ACLs)은 리소스에 액세스할 수 있는 권한이 있는 보안 주체(계정 멤버, 사용자 또는 역할)를 제어합니다. ACLs 는 리소스 기반 정책과 유사하지만 JSON 정책 문서 형식을 사용하지는 않습니다.

ABAC Amazon Inspector 사용

지원ABAC(정책의 태그): 부분

속성 기반 액세스 제어(ABAC)는 속성을 기반으로 권한을 정의하는 권한 부여 전략입니다. 에서는 AWS이러한 속성을 태그 라고 합니다. IAM 엔터티(사용자 또는 역할)와 많은 AWS 리소스에 태그를 연결할 수 있습니다. 엔터티 및 리소스에 태그를 지정하는 것은 의 첫 번째 단계입니다ABAC. 그런 다음 보안 주체의 태그가 액세스하려는 리소스의 태그와 일치할 때 작업을 허용하는 ABAC 정책을 설계합니다.

ABAC 는 빠르게 성장하는 환경에서 유용하며 정책 관리가 번거로워지는 상황에 도움이 됩니다.

태그에 근거하여 액세스를 제어하려면 `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 조건 요소에 태그 정보를 제공합니다.

서비스가 모든 리소스 유형에 대해 세 가지 조건 키를 모두 지원하는 경우, 값은 서비스에 대해 예입니다. 서비스가 일부 리소스 유형에 대해서만 세 가지 조건 키를 모두 지원하는 경우, 값은 부분적입니다.

에 대한 자세한 내용은 IAM 사용 설명서의 란 무엇입니까ABAC?를 ABAC참조하세요. 설정 단계가 포함된 자습서를 보려면 IAM 사용 설명서의 속성 기반 액세스 제어 사용(ABAC)을 ABAC참조하세요.

Amazon Inspector에서 임시 보안 인증 정보 사용

임시 자격 증명 지원: 예

임시 자격 증명을 사용하여 로그인할 때 작동하지 AWS 서비스 않는 경우도 있습니다. 임시 자격 증명으로 AWS 서비스 작업하는 것을 비롯한 자세한 내용은 IAM 사용 설명서의 AWS 서비스 에서 작업하는 IAM 섹션을 참조하세요.

사용자 이름 및 암호를 제외한 방법을 AWS Management Console 사용하여 에 로그인하는 경우 임시 자격 증명을 사용합니다. 예를 들어 회사의 Single Sign-On(SSO) 링크를 AWS 사용하여 에 액세스하면 해당 프로세스가 임시 자격 증명을 자동으로 생성합니다. 또한 콘솔에 사용자로 로그인한 다음 역할을 전환할 때 임시 보안 인증을 자동으로 생성합니다. 역할 전환에 대한 자세한 내용은 IAM 사용 설명서의 역할(콘솔)로 전환을 참조하세요.

AWS CLI 또는 를 사용하여 임시 자격 증명을 수동으로 생성할 수 있습니다 AWS API. 그런 다음 이러한 임시 자격 증명을 사용하여 장기 액세스 키를 사용하는 대신 임시 자격 증명을 동적으로 생성하는 AWS. AWS recommends에 액세스할 수 있습니다. 자세한 내용은 의 임시 보안 자격 증명을 IAM참조하세요.

Amazon Inspector에 대한 교차 서비스 보안 주체 권한

전달 액세스 세션 지원(FAS): 예

IAM 사용자 또는 역할을 사용하여 에서 작업을 수행하는 경우 보안 주체로 AWS간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS 는 를 호출하는 보안 주체의 권한을 다운스트림 서비스에 AWS 서비스 대한 요청과 AWS 서비스함께 사용합니다. FAS 요청은 서비스가 다른 AWS 서비스 또는 리소스와의 상호 작용을 완료해야 하는 요청을 수신할 때만 수행됩니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 액세스 세션 전달을 참조하세요.

Amazon Inspector의 서비스 역할

서비스 역할 지원: 아니요

서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하기 위해 수임하는 IAM 역할입니다. IAM 관리자는 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다IAM. 자세한 내용은 IAM 사용 설명서의 에 권한을 위임할 역할 생성을 AWS 서비스 참조하세요.

> **Warning**
> 서비스 역할에 대한 권한을 변경하면 Amazon Inspector 기능이 중단될 수 있습니다. Amazon Inspector에서 관련 지침을 제공하는 경우에만 서비스 역할을 편집합니다.

Amazon Inspector의 서비스 연결 역할

서비스 링크 역할 지원: 예

서비스 연결 역할은 에 연결된 서비스 역할의 한 유형입니다 AWS 서비스. 서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 에 표시 AWS 계정 되며 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할에 대한 권한을 볼 수 있지만 편집할 수는 없습니다.

서비스 연결 역할 생성 또는 관리에 대한 자세한 내용은 AWS 서비스 에서 작동하는 섹션을 참조하세요IAM. 서비스 연결 역할 열에서 Yes(이)가 포함된 서비스를 테이블에서 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 Yes(네) 링크를 선택합니다.

Amazon Inspector의 자격 증명 기반 정책 예

기본적으로 사용자 및 역할은 Amazon Inspector 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management Console, AWS Command Line Interface (AWS CLI) 또는 를 사용하여 작업을 수행할 수 없습니다 AWS API. 사용자에게 필요한 리소스에 대한 작업을 수행할 수 있는 권한을 부여하기 위해 IAM 관리자는 IAM 정책을 생성할 수 있습니다. 그런 다음 관리자는 IAM 정책을 역할에 추가하고 사용자는 역할을 수임할 수 있습니다.

이러한 예제 정책 문서를 사용하여 IAM 자격 증명 기반 JSON 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

각 리소스 유형에 대한 형식 등 Amazon Inspector 정의한 작업 및 리소스 유형에 ARNs 대한 자세한 내용은 서비스 승인 참조의 Amazon Inspector에 대한 작업, 리소스 및 조건 키를 참조하세요.

주제

- 정책 모범 사례
- Amazon Inspector 콘솔 사용
- 사용자가 자신의 고유한 권한을 볼 수 있도록 허용
- 모든 Amazon Inspector 리소스에 대한 읽기 전용 액세스 허용
- 모든 Amazon Inspector 리소스에 대한 전체 액세스 허용

정책 모범 사례

자격 증명 기반 정책에 따라 계정에서 사용자가 Amazon Inspector 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부가 결정됩니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따릅니다.

- AWS 관리형 정책을 시작하고 최소 권한 권한으로 전환 - 사용자 및 워크로드에 권한 부여를 시작하려면 많은 일반적인 사용 사례에 대한 권한을 부여하는 AWS 관리형 정책을 사용합니다. 에서 사용할 수 있습니다 AWS 계정. 사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 추가로 줄이는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 관리AWS 형 정책 또는 AWS 작업 함수에 대한 관리형 정책을 참조하세요.
- 최소 권한 적용 - IAM 정책으로 권한을 설정할 때 작업을 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. 를 사용하여 권한을 적용하는 IAM 방법에 대한 자세한 내용은 IAM 사용 설명서의 에서 정책 및 권한을 IAM 참조하세요.
- IAM 정책의 조건을 사용하여 액세스 추가 제한 - 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어 정책 조건을 작성하여 를 사용하여 모든 요청을 전송하도록 지정할 수 있습니다SSL. AWS 서비스와 같은 특정 를 통해 서비스 작업을 사용하는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다 AWS CloudFormation. 자세한 내용은 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.
- IAM Access Analyzer를 사용하여 IAM 정책을 검증하여 안전하고 기능적인 권한을 보장합니다. IAM Access Analyzer는 정책이 정책 언어(JSON) 및 IAM 모범 사례를 준수하도록 새 정책 및 기존 IAM 정책을 검증합니다. IAM Access Analyzer는 안전하고 기능적인 정책을 작성하는 데 도움이 되는 100개 이상의 정책 확인 및 실행 가능한 권장 사항을 제공합니다. 자세한 내용은 IAM 사용 설명서의 IAM Access Analyzer 정책 검증을 참조하세요.
- 다중 인증 필요(MFA) - 에 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 MFA 위해 를 AWS 계정켭니다. API 작업을 호출할 MFA 때 를 요구하려면 정책에 MFA 조건을 추가합니다. 자세한 내용은 IAM 사용 설명서의 MFA-보호된 API 액세스 구성을 참조하세요.

의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 의 보안 모범 사례를 IAM IAM참조하세요.

Amazon Inspector 콘솔 사용

Amazon Inspector 콘솔에 액세스하려면 최소한의 권한이 있어야 합니다. 이러한 권한은 AWS 계정에서 Amazon Inspector 리소스에 대한 세부 정보를 나열하고 볼 수 있도록 허용해야 합니다. 최소 필수 권한보다 더 제한적인 자격 증명 기반 정책을 만들면 콘솔이 해당 정책에 연결된 엔터티(사용자 또는 역할)에 대해 의도대로 작동하지 않습니다.

AWS CLI 또는 에만 전화를 거는 사용자에 대해 최소 콘솔 권한을 허용할 필요는 없습니다 AWS API. 대신 수행하려는 API 작업과 일치하는 작업에만 액세스할 수 있도록 허용합니다.

사용자와 역할이 여전히 Amazon Inspector 콘솔을 사용할 수 있도록 하려면 Amazon Inspector *ConsoleAccess* 또는 *ReadOnly* AWS 관리형 정책도 엔터티에 연결합니다. 자세한 내용은 IAM 사용 설명서의 사용자에게 권한 추가를 참조하세요.

사용자가 자신의 고유한 권한을 볼 수 있도록 허용

이 예제에서는 IAM 사용자가 사용자 ID에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 AWS CLI 를 사용하여 프로그래밍 방식으로 이 작업을 완료할 수 있는 권한이 포함되어 있습니다 AWS API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

모든 Amazon Inspector 리소스에 대한 읽기 전용 액세스 허용

이 예제에서는 모든 Amazon Inspector 리소스에 대한 읽기 전용 액세스를 허용하는 정책을 보여줍니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "inspector2:Describe*",
                "inspector2:Get*",
                "inspector2:BatchGet*",
                "inspector2:List*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "organizations:ListDelegatedAdministrators",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization"
            ],
            "Resource": "*"
        }
    ]
}
```

모든 Amazon Inspector 리소스에 대한 전체 액세스 허용

이 예제에서는 모든 Amazon Inspector 리소스에 대한 전체 액세스를 허용하는 정책을 보여줍니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Action": "inspector2:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "inspector2.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "organizations:EnableAWSServiceAccess",
                "organizations:RegisterDelegatedAdministrator",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS Amazon Inspector 관리형 정책

AWS 관리형 정책은 에서 생성 및 관리하는 독립 실행형 정책입니다 AWS. AWS 관리형 정책은 사용자, 그룹 및 역할에 권한 할당을 시작할 수 있도록 많은 일반적인 사용 사례에 대한 권한을 제공하도록 설계되었습니다.

AWS 관리형 정책은 모든 AWS 고객이 사용할 수 있으므로 특정 사용 사례에 대해 최소 권한 권한을 부여하지 않을 수 있다는 점에 유의하세요. 사용 사례에 고유한 고객 관리형 정책을 정의하여 권한을 줄이는 것이 좋습니다.

AWS 관리형 정책에 정의된 권한은 변경할 수 없습니다. 가 AWS 관리형 정책에 정의된 권한을 AWS 업데이트하면 정책이 연결된 모든 보안 주체 자격 증명(사용자, 그룹 및 역할)에 영향을 미칩니다. AWS 는 새 AWS 서비스 가 시작되거나 기존 서비스에 새 API 작업을 사용할 수 있게 되면 AWS 관리형 정책을 업데이트할 가능성이 높습니다.

자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책을 참조하세요.

AWS 관리형 정책: AmazonInspector2FullAccess

`AmazonInspector2FullAccess` 정책을 IAM 자격 증명에 연결할 수 있습니다.

이 정책은 Amazon Inspector에 대한 전체 액세스를 허용하는 관리 권한을 부여합니다.

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `inspector2` - Amazon Inspector 기능에 대한 전체 액세스를 허용합니다.
- `iam` - Amazon Inspector가 서비스 연결 역할을 생성할 수 있도록 허용`AWSServiceRoleForAmazonInspector2`하고 `AWSServiceRoleForAmazonInspector2Agentless`Amazon Inspector`AWSServiceRoleForAmazonInspector2`가 Amazon EC2 인스턴스, Amazon ECR 리포지토리 및 컨테이너 이미지에 대한 검색 정보와 같은 작업을 수행하는 데 필요합니다. Amazon Inspector 또한 Amazon Inspector가 VPC 네트워크를 분석하고 조직과 연결된 계정을 설명하는 데도 필요합니다. `AWSServiceRoleForAmazonInspector2Agentless`는 Amazon Inspector가 Amazon EC2 인스턴스 및 Amazon EBS 스냅샷에 대한 검색 정보와 같은 작업을 수행하는 데 필요합니다. 또한 AWS KMS 키로 암호화된 Amazon EBS 스냅샷을 복호화해야 합니다. 자세한 내용은 Amazon Inspector에 서비스 연결 역할 사용 단원을 참조하십시오.
- `organizations` - 관리자가 Amazon Inspector를 AWS Organizations의 조직에 사용할 수 있도록 허용합니다. 에서 Amazon Inspector에 대한 신뢰할 수 있는 액세스를 활성화하면 위임된 관리자 계정의 AWS Organizations구성원이 설정을 관리하고 조직 전체에서 결과를 볼 수 있습니다.

- `codeguru-security` - 관리자가 Amazon Inspector를 사용하여 정보 코드 조각을 검색하고 CodeGuru Security가 저장하는 코드에 대한 암호화 설정을 변경할 수 있습니다. 자세한 내용은 결과 코드에 대한 저장 중 암호화 단원을 참조하십시오.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "AllowFullAccessToInspectorApis",
   "Effect": "Allow",
   "Action": "inspector2:*",
   "Resource": "*"
  },
  {
   "Sid": "AllowAccessToCodeGuruApis",
   "Effect": "Allow",
   "Action": [
    "codeguru-security:BatchGetFindings",
    "codeguru-security:GetAccountConfiguration"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowAccessToCreateSlr",
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": [
      "agentless.inspector2.amazonaws.com",
      "inspector2.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AllowAccessToOrganizationApis",
   "Effect": "Allow",
   "Action": [
    "organizations:EnableAWSServiceAccess",
    "organizations:RegisterDelegatedAdministrator",
```

```
    "organizations:ListDelegatedAdministrators",
    "organizations:ListAWSServiceAccessForOrganization",
    "organizations:DescribeOrganizationalUnit",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS 관리형 정책: AmazonInspector2ReadOnlyAccess

`AmazonInspector2ReadOnlyAccess` 정책을 IAM 자격 증명에 연결할 수 있습니다.

이 정책은 Amazon Inspector에 대한 읽기 전용 액세스를 허용하는 권한을 부여합니다.

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `inspector2` – Amazon Inspector 기능에 대한 읽기 전용 액세스를 허용합니다.
- `organizations` - 의 조직에 대한 Amazon Inspector 적용 범위에 대한 세부 정보를 볼 수 AWS Organizations 있습니다.
- `codeguru-security` - CodeGuru 보안에서 코드 조각을 검색할 수 있습니다. 또한 CodeGuru Security에 저장된 코드의 암호화 설정을 볼 수 있습니다.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "organizations:ListDelegatedAdministrators",
    "organizations:ListAWSServiceAccessForOrganization",
    "organizations:DescribeOrganizationalUnit",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization",
```

```
    "inspector2:BatchGet*",
    "inspector2:List*",
    "inspector2:Describe*",
    "inspector2:Get*",
    "inspector2:Search*",
    "codeguru-security:BatchGetFindings",
    "codeguru-security:GetAccountConfiguration"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS 관리형 정책: AmazonInspector2ManagedCisPolicy

`AmazonInspector2ManagedCisPolicy` 정책을 IAM엔터티에 연결할 수 있습니다. 이 정책은 Amazon EC2 인스턴스에 인스턴스 CIS 스캔을 실행할 수 있는 권한을 부여하는 역할에 연결되어야 합니다. IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS CLI 또는 AWS API 요청을 하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. 이는 EC2 인스턴스 내에 액세스 키를 저장하는 것보다 좋습니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로필에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행 중인 프로그램이 임시 자격 증명을 가져올 수 있습니다. 자세한 내용은 IAM 사용 설명서EC2의 IAM 역할 사용을 참조하세요.

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `inspector2` - CIS 스캔을 실행하는 데 사용되는 작업에 대한 액세스를 허용합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
            "inspector2:StartCisSession",
            "inspector2:StopCisSession",
            "inspector2:SendCisSessionTelemetry",
            "inspector2:SendCisSessionHealth"
            ],
```

```
            "Resource": "*",
        }
    ]
}
```

AWS 관리형 정책: AmazonInspector2ServiceRolePolicy

`AmazonInspector2ServiceRolePolicy` 정책을 IAM엔터티에 연결할 수 없습니다. 이 정책을 서비스 연결 역할에 연결하면 Amazon Inspector가 사용자를 대신하여 작업을 수행할 수 있습니다. 자세한 내용은 Amazon Inspector에 서비스 연결 역할 사용 단원을 참조하십시오.

AWS 관리형 정책: AmazonInspector2AgentlessServiceRolePolicy

`AmazonInspector2AgentlessServiceRolePolicy` 정책을 IAM엔터티에 연결할 수 없습니다. 이 정책을 서비스 연결 역할에 연결하면 Amazon Inspector가 사용자를 대신하여 작업을 수행할 수 있습니다. 자세한 내용은 Amazon Inspector에 서비스 연결 역할 사용 단원을 참조하십시오.

AWS 관리형 정책에 대한 Amazon Inspector 업데이트

이 서비스가 이러한 변경 사항을 추적하기 시작한 이후 Amazon Inspector의 AWS 관리형 정책 업데이트에 대한 세부 정보를 확인합니다. 이 페이지의 변경 사항에 대한 자동 알림을 받으려면 Amazon Inspector 문서 기록 페이지에서 RSS 피드를 구독하세요.

변경 사항	설명	날짜
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 에서 함수 태그를 반환할 수 있는 새 권한을 추가했습니다 AWS Lambda.	2024년 7월 31일
AmazonInspector2FullAccess - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 서비스 연결 역할을 생성할 수 있는 권한을 추가했습니다`AWSServiceRoleForAmazonInspector2Agentless` . 이	2024년 4월 24일

변경 사항	설명	날짜
	를 통해 사용자는 Amazon Inspector 를 활성화할 때 에이전트 기반 스캔 및 에이전트리스 스캔을 수행할 수 있습니다.	
AmazonInspector2ManagedCisPolicy - 새 정책	Amazon Inspector는 인스턴스에 대한 CIS 스캔을 허용하기 위해 인스턴스 프로파일의 일부로 사용할 수 있는 새로운 관리형 정책을 추가했습니다.	2024년 1월 23일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 대상 인스턴스에 대한 CIS 스캔을 시작할 수 있는 새 권한을 추가했습니다.	2024년 1월 23일
AmazonInspector2AgentlessServiceRolePolicy - 새 정책	Amazon Inspector는 에이전트 없이 EC2 인스턴스를 스캔할 수 있도록 새 서비스 연결 역할 정책을 추가했습니다.	2023년 11월 27일
AmazonInspector2ReadOnlyAccess - 기존 정책 업데이트	Amazon Inspector에서 읽기 전용 사용자가 패키지 취약성 결과에 대한 취약성 인텔리전스 세부 정보를 검색할 수 있는 새로운 권한을 추가했습니다.	2023년 9월 22일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 Elastic Load Balancing 대상 그룹의 일부인 Amazon EC2 인스턴스의 네트워크 구성을 스캔할 수 있는 새로운 권한을 추가했습니다.	2023년 8월 31일

변경 사항	설명	날짜
AmazonInspector2ReadOnlyAccess - 기존 정책 업데이트	Amazon Inspector는 읽기 전용 사용자가 리소스에 대한 소프트웨어 재료 명세서(SBOM)를 내보낼 수 있는 새 권한을 추가했습니다.	2023년 6월 29일
AmazonInspector2ReadOnlyAccess - 기존 정책 업데이트	Amazon Inspector에서 읽기 전용 사용자가 자신의 계정에 대한 Lambda 코드 스캔 결과의 암호화 설정 세부 정보를 검색할 수 있는 새로운 권한을 추가했습니다.	2023년 6월 13일
AmazonInspector2FullAccess - 기존 정책 업데이트	Amazon Inspector는 사용자가 Lambda 코드 스캔의 조사 결과에서 코드를 암호화하도록 고객 관리형 KMS 키를 구성할 수 있는 새로운 권한을 추가했습니다.	2023년 6월 13일
AmazonInspector2ReadOnlyAccess - 기존 정책 업데이트	Amazon Inspector에서 읽기 전용 사용자가 자신의 계정에 대한 Lambda 코드 스캔 상태 및 결과에 대한 세부 정보를 검색할 수 있는 새로운 권한을 추가했습니다.	2023년 5월 2일

변경 사항	설명	날짜
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Lambda 스캔을 활성화할 때 Amazon Inspector가 계정에서 AWS CloudTrail 서비스 연결 채널을 생성할 수 있는 새로운 권한을 추가했습니다. 이렇게 하면 Amazon Inspector가 계정의 CloudTrail 이벤트를 모니터링할 수 있습니다.	2023년 4월 30일
AmazonInspector2FullAccess - 기존 정책 업데이트	Amazon Inspector에서 사용자가 Lambda 코드 스캔의 코드 취약성 결과에 대한 세부 정보를 검색할 수 있는 새로운 권한을 추가했습니다.	2023년 4월 21일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 고객이 Amazon EC2 심층 검사를 위해 정의한 사용자 지정 경로에 대한 정보를 Amazon EC2 Systems Manager로 전송할 수 있는 새로운 권한을 추가했습니다.	2023년 4월 17일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Lambda 스캔을 활성화할 때 Amazon Inspector가 계정에서 AWS CloudTrail 서비스 연결 채널을 생성할 수 있는 새로운 권한을 추가했습니다. 이렇게 하면 Amazon Inspector가 계정의 CloudTrail 이벤트를 모니터링할 수 있습니다.	2023년 4월 30일

변경 사항	설명	날짜
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 AWS Lambda 함수의 개발자 코드 스캔을 요청하고 Amazon CodeGuru Security로부터 스캔 데이터를 수신할 수 있는 새로운 권한을 추가했습니다. 또한 Amazon Inspector는 IAM 정책을 검토할 수 있는 권한을 추가했습니다. Amazon Inspector는 이 정보를 사용하여 Lambda 함수의 코드 취약성을 스캔합니다.	2023년 2월 28일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 AWS Lambda 함수가 마지막으로 호출된 시기에 CloudWatch 대한 정보를 검색할 수 있는 새 문을 추가했습니다. Amazon Inspector는 이 정보를 사용하여 사용자 환경에서 지난 90일 동안 활성화된 Lambda 함수에 초점을 맞추어 스캔을 수행합니다.	2023년 2월 20일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 각 AWS Lambda 함수와 연결된 각 계층 버전을 포함하여 함수에 대한 정보를 검색할 수 있는 새 문을 추가했습니다. Amazon Inspector는 이 정보를 사용하여 Lambda 함수의 보안 취약성을 스캔합니다.	2022년 11월 28일

변경 사항	설명	날짜
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 SSM 연결 실행을 설명할 수 있도록 새 작업을 추가했습니다. 또한 Amazon Inspector는 Amazon Inspector가 `AmazonInspector2` 소유 SSM 문서와의 SSM 연결을 생성, 업데이트, 삭제 및 시작할 수 있도록 추가 리소스 범위 조정을 추가했습니다.	2022년 8월 31일
AmazonInspector2ServiceRolePolicy 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 다른 AWS 파티션의 소프트웨어 인벤토리를 수집할 수 있도록 정책의 리소스 범위 조정을 업데이트했습니다.	2022년 8월 12일
AmazonInspector2ServiceRolePolicy - 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 SSM 연결을 생성, 삭제 및 업데이트할 수 있도록 작업의 리소스 범위를 재구성했습니다.	2022년 8월 10일
AmazonInspector2ReadOnlyAccess - 새 정책	Amazon Inspector 기능에 대한 읽기 전용 액세스를 허용하는 새로운 정책이 추가되었습니다.	2022년 1월 21일
AmazonInspector2FullAccess - 새 정책	Amazon Inspector 기능에 대한 전체 액세스를 허용하는 새로운 정책이 추가되었습니다.	2021년 11월 29일

변경 사항	설명	날짜
AmazonInspector2ServiceRolePolicy - 새 정책	Amazon Inspector에서 사용자를 대신하여 다른 서비스의 작업을 수행할 수 있도록 허용하는 새로운 정책이 추가되었습니다.	2021년 11월 29일
Amazon Inspector에서 변경 사항 추적 시작	Amazon Inspector는 AWS 관리형 정책에 대한 변경 사항을 추적하기 시작했습니다.	2021년 11월 29일

Amazon Inspector에 서비스 연결 역할 사용

Amazon Inspector는 라는 AWS Identity and Access Management (IAM) 서비스 연결 역할을 사용합니다`AWSServiceRoleForAmazonInspector2`. 이 서비스 연결 역할은 Amazon Inspector 에 직접 연결된 IAM 역할입니다. Amazon Inspector에서 사전 정의하며 Amazon Inspector가 AWS 서비스 사용자를 대신하여 다른 를 호출하는 데 필요한 모든 권한을 포함합니다.

서비스 연결 역할을 사용하면 필요한 권한을 수동으로 추가할 필요가 없으므로 Amazon Inspector를 더 쉽게 설정할 수 있습니다. Amazon Inspector는 서비스 연결 역할의 권한을 정의하며, 다르게 정의되지 않는 한 Amazon Inspector만 해당 역할을 맡을 수 있습니다. 정의된 권한에는 신뢰 정책 및 권한 정책이 포함되며 해당 권한 정책은 다른 IAM엔터티에 연결할 수 없습니다.

IAM 엔터티(예: 그룹 또는 역할)가 서비스 연결 역할을 생성, 편집 또는 삭제할 수 있도록 권한을 구성해야 합니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 권한을 참조하세요. 먼저 관련 리소스를 삭제해야만 서비스 연결 역할을 삭제할 수 있습니다. 이렇게 하면 리소스 액세스 권한을 실수로 삭제할 수 없기 때문에 Amazon Inspector 리소스가 보호됩니다.

서비스 연결 역할을 지원하는 다른 서비스에 대한 자세한 내용은 AWS 에서 작업하는 서비스를 IAM 참조하고 서비스 연결 역할 열에서 예인 서비스를 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 검토하려면 예 링크를 선택합니다.

Amazon Inspector의 서비스 연결 역할 권한

Amazon Inspector는 `AWSServiceRoleForAmazonInspector2`라는 서비스 연결 역할을 사용합니다. 이 서비스 연결 역할은 `inspector2.amazonaws.com` 서비스에 해당 역할을 맡깁니다.

역할에 대한 권한 정책(`AmazonInspector2ServiceRolePolicy`)을 통해 Amazon Inspector에서는 다음과 같은 작업을 수행할 수 있습니다.

- Amazon Elastic Compute Cloud(AmazonEC2) 작업을 사용하여 인스턴스 및 네트워크 경로에 대한 정보를 검색합니다.
- AWS Systems Manager 작업을 사용하여 Amazon EC2 인스턴스에서 인벤토리를 검색하고 사용자 지정 경로에서 타사 패키지에 대한 정보를 검색합니다.
- 작업을 사용하여 대상 인스턴스에 AWS Systems Manager `SendCommand` 대한 CIS 스캔을 호출합니다.
- Amazon Elastic Container Registry 작업을 사용하여 컨테이너 이미지에 대한 정보를 검색합니다.
- AWS Lambda 작업을 사용하여 Lambda 함수에 대한 정보를 검색합니다.
- AWS Organizations 작업을 사용하여 연결된 계정을 설명합니다.
- CloudWatch 작업을 사용하여 Lambda 함수가 마지막으로 호출된 시간에 대한 정보를 검색합니다.
- 선택 IAM 작업을 사용하여 Lambda 코드에 보안 취약성을 생성할 수 있는 IAM 정책에 대한 정보를 검색합니다.
- CodeGuru 보안 작업을 사용하여 Lambda 함수에서 코드 스캔을 수행합니다. Amazon Inspector는 다음 CodeGuru 보안 작업을 사용합니다.
  - codeguru-security:CreateScan – CodeGuru Security 스캔을 생성할 수 있는 권한을 부여합니다.
  - codeguru-security:GetScan – CodeGuru Security 스캔 메타데이터를 검색할 수 있는 권한을 부여합니다.
  - codeguru-security:ListFindings – CodeGuru Security에서 생성한 조사 결과를 검색할 수 있는 권한을 부여합니다.
  - codeguru-security:DeleteScansByCategory – CodeGuru Security가 Amazon Inspector 시작한 스캔을 삭제할 수 있는 권한을 부여합니다.
  - codeguru-security:BatchGetFindings – CodeGuru Security에서 생성한 특정 조사 결과의 배치를 검색할 수 있는 권한을 부여합니다.
- Elastic Load Balancing 작업을 선택하여 Elastic Load Balancing 대상 그룹의 일부인 EC2 인스턴스의 네트워크 스캔을 미리 구성합니다.

역할은 다음과 같은 권한 정책으로 구성됩니다.

```
    {
"Version": "2012-10-17",
```

```
"Statement": [
 {
  "Sid": "TirosPolicy",
  "Effect": "Allow",
  "Action": [
   "directconnect:DescribeConnections",
   "directconnect:DescribeDirectConnectGatewayAssociations",
   "directconnect:DescribeDirectConnectGatewayAttachments",
   "directconnect:DescribeDirectConnectGateways",
   "directconnect:DescribeVirtualGateways",
   "directconnect:DescribeVirtualInterfaces",
   "ec2:DescribeAvailabilityZones",
   "ec2:DescribeCustomerGateways",
   "ec2:DescribeInstances",
   "ec2:DescribeInternetGateways",
   "ec2:DescribeManagedPrefixLists",
   "ec2:DescribeNatGateways",
   "ec2:DescribeNetworkAcls",
   "ec2:DescribeNetworkInterfaces",
   "ec2:DescribePrefixLists",
   "ec2:DescribeRegions",
   "ec2:DescribeRouteTables",
   "ec2:DescribeSecurityGroups",
   "ec2:DescribeSubnets",
   "ec2:DescribeTransitGatewayAttachments",
   "ec2:DescribeTransitGatewayConnects",
   "ec2:DescribeTransitGatewayPeeringAttachments",
   "ec2:DescribeTransitGatewayRouteTables",
   "ec2:DescribeTransitGatewayVpcAttachments",
   "ec2:DescribeTransitGateways",
   "ec2:DescribeVpcEndpointServiceConfigurations",
   "ec2:DescribeVpcEndpoints",
   "ec2:DescribeVpcPeeringConnections",
   "ec2:DescribeVpcs",
   "ec2:DescribeVpnConnections",
   "ec2:DescribeVpnGateways",
   "ec2:GetManagedPrefixListEntries",
   "ec2:GetTransitGatewayRouteTablePropagations",
   "ec2:SearchTransitGatewayRoutes",
   "elasticloadbalancing:DescribeListeners",
   "elasticloadbalancing:DescribeLoadBalancerAttributes",
   "elasticloadbalancing:DescribeLoadBalancers",
   "elasticloadbalancing:DescribeRules",
   "elasticloadbalancing:DescribeTags",
```

```
    "elasticloadbalancing:DescribeTargetGroups",
    "elasticloadbalancing:DescribeTargetGroupAttributes",
    "elasticloadbalancing:DescribeTargetHealth",
    "network-firewall:DescribeFirewall",
    "network-firewall:DescribeFirewallPolicy",
    "network-firewall:DescribeResourcePolicy",
    "network-firewall:DescribeRuleGroup",
    "network-firewall:ListFirewallPolicies",
    "network-firewall:ListFirewalls",
    "network-firewall:ListRuleGroups",
    "tiros:CreateQuery",
    "tiros:GetQueryAnswer"
   ],
   "Resource": [
    "*"
   ]
  },
  {
   "Sid": "PackageVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "ecr:BatchGetImage",
    "ecr:BatchGetRepositoryScanningConfiguration",
    "ecr:DescribeImages",
    "ecr:DescribeRegistry",
    "ecr:DescribeRepositories",
    "ecr:GetAuthorizationToken",
    "ecr:GetDownloadUrlForLayer",
    "ecr:GetRegistryScanningConfiguration",
    "ecr:ListImages",
    "ecr:PutRegistryScanningConfiguration",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization",
    "organizations:ListAccounts",
    "ssm:DescribeAssociation",
    "ssm:DescribeAssociationExecutions",
    "ssm:DescribeInstanceInformation",
    "ssm:ListAssociations",
    "ssm:ListResourceDataSync"
   ],
   "Resource": "*"
  },
  {
   "Sid": "LambdaPackageVulnerabilityScanning",
```

```
    "Effect": "Allow",
    "Action": [
     "lambda:ListFunctions",
     "lambda:GetFunction",
     "lambda:GetLayerVersion",
     "lambda:ListTags",
     "cloudwatch:GetMetricData"
    ],
    "Resource": "*"
   },
   {
    "Sid": "GatherInventory",
    "Effect": "Allow",
    "Action": [
     "ssm:CreateAssociation",
     "ssm:StartAssociationsOnce",
     "ssm:DeleteAssociation",
     "ssm:UpdateAssociation"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:instance/*",
     "arn:aws:ssm:*:*:document/AmazonInspector2-*",
     "arn:aws:ssm:*:*:document/AWS-GatherSoftwareInventory",
     "arn:aws:ssm:*:*:managed-instance/*",
     "arn:aws:ssm:*:*:association/*"
    ]
   },
   {
    "Sid": "DataSyncCleanup",
    "Effect": "Allow",
    "Action": [
     "ssm:CreateResourceDataSync",
     "ssm:DeleteResourceDataSync"
    ],
    "Resource": [
     "arn:aws:ssm:*:*:resource-data-sync/InspectorResourceDataSync-do-not-delete"
    ]
   },
   {
    "Sid": "ManagedRules",
    "Effect": "Allow",
    "Action": [
     "events:PutRule",
     "events:DeleteRule",
```

```
    "events:DescribeRule",
    "events:ListTargetsByRule",
    "events:PutTargets",
    "events:RemoveTargets"
   ],
   "Resource": [
    "arn:aws:events:*:*:rule/DO-NOT-DELETE-AmazonInspector*ManagedRule"
   ]
  },
  {
   "Sid": "LambdaCodeVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "codeguru-security:CreateScan",
    "codeguru-security:GetAccountConfiguration",
    "codeguru-security:GetFindings",
    "codeguru-security:GetScan",
    "codeguru-security:ListFindings",
    "codeguru-security:BatchGetFindings",
    "codeguru-security:DeleteScansByCategory"
   ],
   "Resource": [
    "*"
   ]
  },
  {
   "Sid": "CodeGuruCodeVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "iam:GetRole",
    "iam:GetRolePolicy",
    "iam:GetPolicy",
    "iam:GetPolicyVersion",
    "iam:ListAttachedRolePolicies",
    "iam:ListPolicies",
    "iam:ListPolicyVersions",
    "iam:ListRolePolicies",
    "lambda:ListVersionsByFunction"
   ],
   "Resource": [
    "*"
   ],
   "Condition": {
    "ForAnyValue:StringEquals": {
```

```
      "aws:CalledVia": [
       "codeguru-security.amazonaws.com"
      ]
     }
    }
   },
   {
    "Sid": "Ec2DeepInspection",
    "Effect": "Allow",
    "Action": [
     "ssm:PutParameter",
     "ssm:GetParameters",
     "ssm:DeleteParameter"
    ],
    "Resource": [
     "arn:aws:ssm:*:*:parameter/inspector-aws/service/inspector-linux-application-paths"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowManagementOfServiceLinkedChannel",
    "Effect": "Allow",
    "Action": [
     "cloudtrail:CreateServiceLinkedChannel",
     "cloudtrail:DeleteServiceLinkedChannel"
    ],
    "Resource": [
     "arn:aws:cloudtrail:*:*:channel/aws-service-channel/inspector2/*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowListServiceLinkedChannels",
    "Effect": "Allow",
    "Action": [
     "cloudtrail:ListServiceLinkedChannels"
```

```
    ],
    "Resource": [
     "*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowToRunInvokeCisSpecificDocuments",
    "Effect": "Allow",
    "Action": [
     "ssm:SendCommand",
     "ssm:GetCommandInvocation"
    ],
    "Resource": [
     "arn:aws:ssm:*:*:document/AmazonInspector2-InvokeInspectorSsmPluginCIS"
    ]
   },
   {
    "Sid": "AllowToRunCisCommandsToSpecificResources",
    "Effect": "Allow",
    "Action": [
     "ssm:SendCommand"
    ],
    "Resource": [
     "arn:aws:ec2:*:*:instance/*"
    ],
    "Condition": {
     "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   },
   {
    "Sid": "AllowToPutCloudwatchMetricData",
    "Effect": "Allow",
    "Action": [
     "cloudwatch:PutMetricData"
    ],
    "Resource": [
     "*"
```

```
    ],
    "Condition": {
     "StringEquals": {
      "cloudwatch:namespace": "AWS/Inspector2"
     }
    }
   }
  ]
}
```

Amazon Inspector의 서비스 연결 역할 생성

서비스 링크 역할은 수동으로 생성할 필요가 없습니다. AWS Management Console, AWS CLI또는 에서 Amazon Inspector를 활성화하면 AWS API Amazon Inspector가 서비스 연결 역할을 생성합니다.

Amazon Inspector의 서비스 연결 역할 편집

Amazon Inspector에서는 `AWSServiceRoleForAmazonInspector2` 서비스 연결 역할을 편집하도록 허용하지 않습니다. 서비스 연결 역할을 생성한 후에는 다양한 개체가 역할을 참조할 수 있으므로 역할 이름을 변경할 수 없습니다. 그러나 를 사용하여 역할에 대한 설명을 편집할 수 있습니다IAM. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 편집을 참조하세요.

Amazon Inspector의 서비스 연결 역할 삭제

Amazon Inspector를 더 이상 사용하지 않을 경우에는 `AWSServiceRoleForAmazonInspector2` 서비스 연결 역할을 삭제하는 것이 좋습니다. 역할을 삭제하려면 먼저 해당 역할이 활성화된 각 에서 Amazon Inspector를 비활성화 AWS 리전 해야 합니다. Amazon Inspector를 비활성화해도 역할은 삭제되지 않습니다. 따라서 Amazon Inspector를 다시 활성화하면 기존 역할을 사용할 수 있습니다. 이렇게 하면 적극적으로 모니터링되거나 유지 관리되지 않는 미사용 개체를 방지할 수 있습니다. 단, 서비스 링크 역할에 대한 리소스를 먼저 정리해야 수동으로 삭제할 수 있습니다.

이 서비스 연결 역할을 삭제한 다음 다시 생성해야 하는 경우 동일한 프로세스를 사용하여 계정에서 역할을 다시 생성할 수 있습니다. Amazon Inspector를 활성화하면 Amazon Inspector에서 서비스 연결 역할을 자동으로 다시 생성합니다.

> **Note**
>
> 리소스를 삭제하려고 할 때 Amazon Inspector 서비스에서 해당 역할을 사용 중이면 삭제가 실패할 수 있습니다. 이 경우 몇 분 정도 기다렸다가 작업을 다시 시도하세요.

IAM 콘솔, AWS CLI또는 AWS API 를 사용하여 `AWSServiceRoleForAmazonInspector2` 서비스 연결 역할을 삭제할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 삭제를 참조하세요.

Amazon Inspector 에이전트 없는 스캔을 위한 서비스 연결 역할 권한

Amazon Inspector 에이전트 없는 스캔은 `AWSServiceRoleForAmazonInspector2Agentless`라는 서비스 연결 역할을 사용합니다. 이렇게 SLR 하면 Amazon Inspector가 계정에서 Amazon EBS 볼륨 스냅샷을 생성한 다음 해당 스냅샷에서 데이터에 액세스할 수 있습니다. 이 서비스 연결 역할은 `agentless.inspector2.amazonaws.com` 서비스에 해당 역할을 맡깁니다.

> **Important**
>
> 이 서비스 연결 역할의 문은 Amazon Inspector가 `InspectorEc2Exclusion` 태그를 사용하여 스캔에서 제외한 모든 EC2 인스턴스에 대해 에이전트리스 스캔을 수행하지 못하도록 합니다. 또한 문은 암호화에 사용된 KMS 키에 `InspectorEc2Exclusion` 태그가 있는 경우 Amazon Inspector가 볼륨에서 암호화된 데이터에 액세스하지 못하도록 합니다. 자세한 내용은 Amazon Inspector 스캔에서 인스턴스 제외 단원을 참조하십시오.

역할에 대한 권한 정책(`AmazonInspector2AgentlessServiceRolePolicy`)을 통해 Amazon Inspector에서는 다음과 같은 작업을 수행할 수 있습니다.

- Amazon Elastic Compute Cloud(AmazonEC2) 작업을 사용하여 EC2 인스턴스, 볼륨 및 스냅샷에 대한 정보를 검색합니다.
  - Amazon EC2 태그 지정 작업을 사용하여 태그 키로 스캔에 대한 스냅샷에 `InspectorScan` 태그를 지정합니다.
  - Amazon EC2 스냅샷 작업을 사용하여 스냅샷을 생성하고 태그 `InspectorScan` 키로 태그를 지정한 다음 `InspectorScan` 태그 키로 태그가 지정된 Amazon EBS 볼륨의 스냅샷을 삭제합니다.
- Amazon EBS 작업을 사용하여 `InspectorScan` 태그 키로 태그가 지정된 스냅샷에서 정보를 검색합니다.
- 선택 AWS KMS 복호화 작업을 사용하여 AWS KMS 고객 관리형 키로 암호화된 스냅샷을 복호화합니다. Amazon Inspector는 스냅샷을 암호화하는 데 사용되는 KMS 키에 태그가 지정되어 있는 경우 스냅샷을 복호화하지 않습니다`InspectorEc2Exclusion`.

역할은 다음과 같은 권한 정책으로 구성됩니다.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Sid": "InstanceIdentification",
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeInstances",
    "ec2:DescribeVolumes",
    "ec2:DescribeSnapshots"
   ],
   "Resource": "*"
  },
  {
   "Sid": "GetSnapshotData",
   "Effect": "Allow",
   "Action": [
    "ebs:ListSnapshotBlocks",
    "ebs:GetSnapshotBlock"
   ],
   "Resource": "arn:aws:ec2:*:*:snapshot/*",
   "Condition": {
    "StringLike": {
     "aws:ResourceTag/InspectorScan": "*"
    }
   }
  },
  {
   "Sid": "CreateSnapshotsAnyInstanceOrVolume",
   "Effect": "Allow",
   "Action": "ec2:CreateSnapshots",
   "Resource": [
    "arn:aws:ec2:*:*:instance/*",
    "arn:aws:ec2:*:*:volume/*"
   ]
  },
  {
   "Sid": "DenyCreateSnapshotsOnExcludedInstances",
   "Effect": "Deny",
   "Action": "ec2:CreateSnapshots",
   "Resource": "arn:aws:ec2:*:*:instance/*",
   "Condition": {
```

```
    "StringEquals": {
     "ec2:ResourceTag/InspectorEc2Exclusion": "true"
    }
   }
  },
  {
   "Sid": "CreateSnapshotsOnAnySnapshotOnlyWithTag",
   "Effect": "Allow",
   "Action": "ec2:CreateSnapshots",
   "Resource": "arn:aws:ec2:*:*:snapshot/*",
   "Condition": {
    "Null": {
     "aws:TagKeys": "false"
    },
    "ForAllValues:StringEquals": {
     "aws:TagKeys": "InspectorScan"
    }
   }
  },
  {
   "Sid": "CreateOnlyInspectorScanTagOnlyUsingCreateSnapshots",
   "Effect": "Allow",
   "Action": "ec2:CreateTags",
   "Resource": "arn:aws:ec2:*:*:snapshot/*",
   "Condition": {
    "StringLike": {
     "ec2:CreateAction": "CreateSnapshots"
    },
    "Null": {
     "aws:TagKeys": "false"
    },
    "ForAllValues:StringEquals": {
     "aws:TagKeys": "InspectorScan"
    }
   }
  },
  {
   "Sid": "DeleteOnlySnapshotsTaggedForScanning",
   "Effect": "Allow",
   "Action": "ec2:DeleteSnapshot",
   "Resource": "arn:aws:ec2:*:*:snapshot/*",
   "Condition": {
    "StringLike": {
     "ec2:ResourceTag/InspectorScan": "*"
```

```
      }
     }
    },
    {
     "Sid": "DenyKmsDecryptForExcludedKeys",
     "Effect": "Deny",
     "Action": "kms:Decrypt",
     "Resource": "arn:aws:kms:*:*:key/*",
     "Condition": {
      "StringEquals": {
       "aws:ResourceTag/InspectorEc2Exclusion": "true"
      }
     }
    },
    {
     "Sid": "DecryptSnapshotBlocksVolContext",
     "Effect": "Allow",
     "Action": "kms:Decrypt",
     "Resource": "arn:aws:kms:*:*:key/*",
     "Condition": {
      "StringEquals": {
       "aws:ResourceAccount": "${aws:PrincipalAccount}"
      },
      "StringLike": {
       "kms:ViaService": "ec2.*.amazonaws.com",
       "kms:EncryptionContext:aws:ebs:id": "vol-*"
      }
     }
    },
    {
     "Sid": "DecryptSnapshotBlocksSnapContext",
     "Effect": "Allow",
     "Action": "kms:Decrypt",
     "Resource": "arn:aws:kms:*:*:key/*",
     "Condition": {
      "StringEquals": {
       "aws:ResourceAccount": "${aws:PrincipalAccount}"
      },
      "StringLike": {
       "kms:ViaService": "ec2.*.amazonaws.com",
       "kms:EncryptionContext:aws:ebs:id": "snap-*"
      }
     }
    },
```

```
  {
   "Sid": "DescribeKeysForEbsOperations",
   "Effect": "Allow",
   "Action": "kms:DescribeKey",
   "Resource": "arn:aws:kms:*:*:key/*",
   "Condition": {
    "StringEquals": {
     "aws:ResourceAccount": "${aws:PrincipalAccount}"
    },
    "StringLike": {
     "kms:ViaService": "ec2.*.amazonaws.com"
    }
   }
  },
  {
   "Sid": "ListKeyResourceTags",
   "Effect": "Allow",
   "Action": "kms:ListResourceTags",
   "Resource": "arn:aws:kms:*:*:key/*"
  }
 ]
}
```

에이전트 없는 스캔을 위한 서비스 연결 역할 생성

서비스 링크 역할은 수동으로 생성할 필요가 없습니다. AWS Management Console, AWS CLI또는 에서 Amazon Inspector를 활성화하면 AWS API Amazon Inspector가 서비스 연결 역할을 생성합니다.

에이전트 없는 스캔을 위한 서비스 연결 역할 편집

Amazon Inspector에서는 `AWSServiceRoleForAmazonInspector2Agentless` 서비스 연결 역할을 편집하도록 허용하지 않습니다. 서비스 연결 역할을 생성한 후에는 다양한 개체가 역할을 참조할 수 있으므로 역할 이름을 변경할 수 없습니다. 그러나 를 사용하여 역할에 대한 설명을 편집할 수 있습니다IAM. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 편집을 참조하세요.

에이전트 없는 스캔을 위한 서비스 연결 역할 삭제

서비스 연결 역할이 필요한 기능 또는 서비스가 더 이상 필요 없는 경우에는 해당 역할을 삭제하는 것이 좋습니다. 따라서 적극적으로 모니터링하거나 유지하지 않는 미사용 엔티티가 없도록 합니다.

> **Important**
>
> `AWSServiceRoleForAmazonInspector2Agentless` 역할을 삭제하려면 에이전트 없는 스캔을 사용할 수 있는 모든 리전에서 스캔 모드를 에이전트 기반으로 설정해야 합니다.

를 사용하여 서비스 연결 역할을 수동으로 삭제하려면 IAM

IAM 콘솔, AWS CLI또는 AWS API 를 사용하여 AWSServiceRoleForAmazonInspector2Agentless 서비스 연결 역할을 삭제합니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 삭제를 참조하세요.

Amazon Inspector 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 Amazon Inspector 및 에서 작업할 때 발생할 수 있는 일반적인 문제를 진단하고 해결할 수 있습니다IAM.

주제

- Amazon Inspector에서 작업을 수행할 권한이 없음
- iam을 수행할 권한이 없습니다.PassRole
- 내 외부의 사람이 내 Amazon Inspector 리소스 AWS 계정 에 액세스하도록 허용하고 싶습니다.

Amazon Inspector에서 작업을 수행할 권한이 없음

작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 작업을 수행할 수 있도록 정책을 업데이트해야 합니다.

다음 예제 오류는 `mateojackson` IAM 사용자가 콘솔을 사용하여 가상 *`my-example-widget`* 리소스에 대한 세부 정보를 보려고 하지만 가상 `inspector2:`*`GetWidget`* 권한이 없는 경우에 발생합니다.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 inspector2:GetWidget on resource: my-example-widget
```

이 경우 `inspector2:`*`GetWidget`*작업을 사용하여 *`my-example-widget`*리소스에 액세스할 수 있도록 `mateojackson`사용자 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

iam을 수행할 권한이 없습니다.PassRole

`iam:PassRole` 작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 Amazon Inspector에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

일부는 새 서비스 역할 또는 서비스 연결 역할을 생성하는 대신 기존 역할을 해당 서비스에 전달할 수 있도록 AWS 서비스 허용합니다. 이렇게 하려면 사용자가 서비스에 역할을 전달할 수 있는 권한을 가지고 있어야 합니다.

다음 예제 오류는 라는 IAM 사용자가 콘솔을 사용하여 Amazon Inspector 에서 작업을 수행하려고 `marymajor` 할 때 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 수 있는 권한을 가지고 있지 않습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

이 경우, Mary가 `iam:PassRole` 작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

내 외부의 사람이 내 Amazon Inspector 리소스 AWS 계정 에 액세스하도록 허용하고 싶습니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스할 때 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스 제어 목록(ACLs)을 지원하는 서비스의 경우 이러한 정책을 사용하여 사용자에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세히 알아보려면 다음을 참조하세요.

- Amazon Inspector에서 이러한 기능을 지원하는지 여부를 알아보려면 Amazon Inspector의 작동 방식 IAM 섹션을 참조하세요.
- 소유 AWS 계정 한 의 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 소유 AWS 계정 한 다른 의 IAM 사용자에게 액세스 권한 제공을 참조하세요.
- 타사에 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사 AWS 계정 소유에 대한 액세스 권한 제공을 AWS 계정참조하세요.
- 자격 증명 페더레이션을 통해 액세스를 제공하는 방법을 알아보려면 IAM 사용 설명서의 외부 인증 사용자(자격 증명 페더레이션)에 대한 액세스 제공을 참조하세요.

- 교차 계정 액세스를 위한 역할 및 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 에서 교차 계정 리소스 액세스를 IAM 참조하세요.

Amazon Inspector 모니터링

모니터링은 Amazon Inspector 및 기타 AWS 솔루션의 가용성, 안정성 및 성능을 유지하는 데 중요한 부분입니다. 는 Amazon Inspector 모니터링하고, 발생하는 문제를 보고하고, 이러한 문제를 해결하기 위한 조치를 취하는 도구를 AWS 제공합니다.

- Amazon EventBridge은 이벤트를 사용하여 애플리케이션 구성 요소를 함께 연결하여 확장 가능한 이벤트 기반 애플리케이션을 더 쉽게 구축할 수 있는 AWS 서비스입니다. EventBridge 는 애플리케이션, S oftware-as-a-Service(SaaS) 애플리케이션, AWS 서비스 및 경로의 실시간 데이터 스트림을 제공하므로 서비스에서 발생하는 이벤트를 모니터링하고 이벤트 기반 아키텍처를 구축할 수 있습니다.
- AWS CloudTrail 는 에 의해 또는 를 대신하여 수행된 API 호출 및 관련 이벤트를 캡처하는 AWS 서비스입니다 AWS 계정. CloudTrail 는 사용자가 지정한 Amazon S3 버킷에 로그 파일을 전송하므로 라는 사용자 및 계정 AWS, 호출이 수행된 원본 IP 주소, 호출이 발생한 시기를 식별할 수 있습니다.

를 사용하여 Amazon Inspector API 호출 로깅 AWS CloudTrail

Amazon Inspector는 Amazon Inspector . CloudTrail Capture AWS 서비스에서 IAM 사용자 또는 역할이 수행한 작업의 레코드를 제공하는 AWS CloudTrail서비스인 와 통합되어 Amazon Inspector 에 대한 모든 API 호출을 이벤트로 캡처합니다. 캡처된 호출에는 Amazon Inspector 콘솔의 호출과 Amazon Inspector API 작업에 대한 호출이 포함됩니다. 추적을 생성하는 경우 Amazon Inspector 에 대한 CloudTrail 이벤트를 포함하여 Amazon S3 버킷으로 이벤트를 지속적으로 전송할 수 있습니다. Amazon S3 추적을 구성하지 않은 경우에도 콘솔의 이벤트 기록 에서 CloudTrail 최신 이벤트를 볼 수 있습니다. 에서 수집한 정보를 사용하여 다음을 확인할 CloudTrail수 있습니다.

- Amazon Inspector에 보낸 요청
- 요청이 발생한 IP 주소
- 요청한 사람
- 요청이 발생한 시간

에 대한 자세한 내용은 AWS CloudTrail 사용 설명서 섹션을 CloudTrail참조하세요.

의 Amazon Inspector 정보 CloudTrail

CloudTrail 는 계정을 생성할 AWS 계정 때 에서 활성화됩니다. Amazon Inspector 에서 활동이 발생하면 해당 활동은 CloudTrail 이벤트 기록 의 다른 AWS 서비스 이벤트와 함께 이벤트에 기록됩니다. 에서 최근 이벤트를 보고 검색하고 다운로드할 수 있습니다 AWS 계정. 자세한 내용은 이벤트 기록을 사용하여 CloudTrail 이벤트 보기를 참조하세요.

Amazon Inspector 에 대한 이벤트를 AWS 계정포함하여 에서 이벤트에 대한 지속적인 레코드를 생성하려면 추적을 생성합니다. 추적을 사용하면 CloudTrail 가 Amazon S3 버킷에 로그 파일을 전달할 수 있습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정된 S3 버킷으로 로그 파일을 전송합니다. 또한 CloudTrail 로그에서 수집된 이벤트 데이터를 추가로 분석하고 이에 따라 작업 AWS 서비스 하도록 다른 를 구성할 수 있습니다. 자세한 정보는 다음 주제를 참조하세요.

- 추적 생성 개요
- CloudTrail 지원되는 서비스 및 통합
- 에 대한 Amazon SNS 알림 구성 CloudTrail
- 여러 계정에서 CloudTrail 로그 파일 수신
- 여러 리전에서 CloudTrail 로그 파일 수신

모든 Amazon Inspector 작업은 에 의해 기록됩니다 CloudTrail. Amazon Inspector가 수행할 수 있는 모든 작업은 Amazon Inspector API 참조 에 문서화되어 있습니다. 예를 들어 `CreateFindingsReport`, `ListCoverage`및 `UpdateOrganizationConfiguration` 작업에 대한 호출은 CloudTrail 로그 파일에 항목을 생성합니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 보안 인증 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 루트 사용자 또는 IAM 사용자 자격 증명으로 요청이 이루어졌는지 여부.
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청했는지 여부
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail userIdentity 요소를 참조하세요.

Amazon Inspector 로그 파일 항목 이해

추적은 사용자가 지정한 Amazon S3 버킷에 로그 파일로 이벤트를 전달할 수 있도록 하는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함됩니다. 이벤트는 모든 소스로부터 단일 요청을 나타냅니다. 이벤트에는 요청된 작업, 작업 날짜 및 시간, 요청 파라미터 등에 대한 정보가 포함되어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 순서가 지정된 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

의 Amazon Inspector 스캔 정보 CloudTrail

Amazon Inspector Scan은 와 통합됩니다 CloudTrail. 모든 Amazon Inspector 스캔 API 작업은 관리 이벤트로 기록됩니다. Amazon Inspector가 에 로그하는 Amazon Inspector Scan API 작업 목록은 Amazon Inspector 참조의 Amazon Inspector Scan을 CloudTrail참조하세요. Amazon Inspector API

다음 예제에서는 `ScanSbom` 작업을 보여주는 CloudTrail 로그 항목을 보여줍니다.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROA123456789EXAMPLE:akua_mansa",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/akua_mansa",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROA123456789EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-10-17T15:22:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-10-17T16:02:34Z",
    "eventSource": "gamma-inspector-scan.amazonaws.com",
```

```
    "eventName": "ScanSbom",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-sdk-java/2.20.162 Mac_OS_X/13.5.2 OpenJDK_64-
Bit_Server_VM/17.0.8+7-LTS Java/17.0.8 vendor/Amazon.com_Inc. io/sync http/
UrlConnection cfg/retry-mode/legacy",
    "requestParameters": {
        "sbom": {
            "specVersion": "1.5",
            "metadata": {
                "component": {
                    "name": "debian",
                    "type": "operating-system",
                    "version": "9"
                }
            },
            "components": [
                {
                    "name": "packageOne",
                    "purl": "pkg:deb/debian/packageOne@1.0.0?arch=x86_64&distro=9",
                    "type": "application"
                }
            ],
            "bomFormat": "CycloneDX"
        }
    },
    "responseElements": null,
    "requestID": "f041a27f-f33e-4f70-b09b-5fbc5927282a",
    "eventID": "abc8d1e4-d214-4f07-bc56-8a31be6e36fe",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Amazon Inspector의 규정 준수 검증

AWS 서비스 가 특정 규정 준수 프로그램의 범위 내에 있는지 알아보려면 AWS 서비스 규정 준수 프로그램 범위 참조하고 관심 있는 규정 준수 프로그램을 선택합니다. 일반 정보는 AWS 규정 준수 프로그램 참조하세요.

를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다 AWS Artifact. 자세한 내용은 의 보고서 다운로드 AWS Artifact.

사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률 및 규정에 따라 AWS 서비스 결정됩니다. 는 규정 준수를 지원하는 다음 리소스를 AWS 제공합니다.

- 보안 및 규정 준수 빠른 시작 안내서 - 이 배포 안내서에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수에 중점을 둔 에 기준 환경을 배포 AWS 하기 위한 단계를 제공합니다.
- Amazon Web Services의 HIPAA 보안 및 규정 준수를 위한 아키텍처 - 이 백서에서는 기업이 HIPAA 를 사용하여 적격 애플리케이션을 AWS 생성하는 방법을 설명합니다.

> **Note**
>
> 모두 HIPAA 적합한 AWS 서비스 것은 아닙니다. 자세한 내용은 HIPAA 적격 서비스 참조를 참조하세요.

- AWS 규정 준수 리소스 - 이 워크북 및 가이드 모음은 해당 산업 및 위치에 적용될 수 있습니다.
- AWS 고객 규정 준수 가이드 - 규정 준수의 관점에서 공동 책임 모델을 이해합니다. 이 가이드는 여러 프레임워크(미국 국립표준기술연구소(), 결제카드 산업보안표준위원회(NIST), PCI국제표준화기구(ISO) 포함)의 보안 제어에 대한 지침을 보호하고 AWS 서비스 매핑하기 위한 모범 사례를 요약합니다.
- AWS Config 개발자 안내서의 규칙을 사용하여 리소스 평가 - 이 AWS Config 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub - AWS 서비스 에서 보안 상태를 포괄적으로 볼 수 있습니다 AWS. Security Hub 는 보안 제어를 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 Security Hub 제어 참조를 참조하세요.
- Amazon GuardDuty - 의심스러운 활동 및 악의적인 활동이 있는지 환경을 모니터링하여 AWS 계정, 워크로드, 컨테이너 및 데이터에 대한 잠재적 위협을 AWS 서비스 탐지합니다. 는 특정 규정 준수 프레임워크에서 요구하는 침입 탐지 요구 사항을 충족DSS하여 PCI 와 같은 다양한 규정 준수 요구 사항을 해결하는 데 도움이 될 GuardDuty 수 있습니다.
- AWS Audit Manager - 이를 AWS 서비스 통해 AWS 사용량을 지속적으로 감사하여 규정 및 업계 표준 준수 및 위험을 관리하는 방법을 간소화할 수 있습니다.

Amazon Inspector의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 기반으로 구축됩니다. 는 지연 시간이 짧고 처리량이 높으며 중복성이 높은 네트워킹에 연결된 물리적으로 분리되고 격리된 여러 가용 영역을 AWS 리전 제공합니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 극복 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

Amazon Inspector의 인프라 보안

관리형 서비스인 Amazon Inspector는 AWS 글로벌 네트워크 보안으로 보호됩니다. AWS 보안 서비스 및 가 인프라를 AWS 보호하는 방법에 대한 자세한 내용은 AWS 클라우드 보안 섹션을 참조하세요. 인프라 보안 모범 사례를 사용하여 AWS 환경을 설계하려면 Security Pillar AWS Well-Architected Framework의 인프라 보호를 참조하세요.

AWS 게시된 API 호출을 사용하여 네트워크를 통해 Amazon Inspector에 액세스합니다. 고객은 다음을 지원해야 합니다.

- 전송 계층 보안(TLS). TLS 1.2가 필요하며 TLS 1.3을 권장합니다.
- DHE (Ephemeral Diffie-HellmanPFS) 또는 (Elliptic Curve Ephemeral Diffie-Hellman)과 같은 완벽한 순방향 보안ECDHE()이 포함된 Cipher 제품군입니다. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 액세스 키 ID와 IAM 보안 주체와 연결된 보안 액세스 키를 사용하여 요청에 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 인증을 생성하여 요청에 서명할 수 있습니다.

Amazon Inspector의 인시던트 대응

AWS에서는 보안을 가장 중요하게 생각합니다. “클라우드 보안”의 AWS 공동 책임 모델에서 언급한 바와 같이 AWS 는 AWS 클라우드의 모든 서비스를 실행하는 인프라를 보호할 책임이 있습니다. AWS 는 Amazon Inspector 서비스와 관련된 모든 인시던트 대응도 책임집니다.

AWS 고객은 AWS 클라우드에서 보안을 유지할 책임이 있습니다. 즉, 액세스한 모든 AWS 도구와 기능을 포함하여 구현하려는 보안을 제어합니다. 또한 공동 책임 모델의 측면에서 인시던트 대응에 대한 책임이 있음을 의미합니다.

AWS 클라우드에서 실행되는 애플리케이션의 모든 목표를 충족하는 보안 기준을 설정하면 대응할 수 있는 편차를 감지할 수 있습니다. 인시던트 대응은 복잡한 주제이므로 다음 리소스를 검토하여 인시던트 대응의 영향과 선택한 항목이 기업 목표에 미치는 영향을 더 잘 이해하세요. AWS 보안 인시던트 대응 가이드 , AWS 보안 모범 사례 및 AWS 클라우드 채택 프레임워크: 보안 관점 .

Amazon Inspector 통합

Amazon Inspector는 다른 서비스와 통합됩니다. AWS 이러한 서비스는 Amazon Inspector에서 데이터를 수집할 수 있으므로 결과를 다양한 방식으로 볼 수 있습니다. 자세히 알아보려면 다음 통합 옵션을 검토하십시오.

아마존 인스펙터와 아마존 통합 ECR

Amazon Elastic 컨테이너 레지스트리 (AmazonECR) 는 프라이빗 레지스트리를 지원하는 AWS관리형 컨테이너 이미지 레지스트리입니다. Amazon ECR 사설 레지스트리는 가용성과 확장성이 뛰어난 아키텍처에서 컨테이너 이미지를 호스팅합니다. Amazon Inspector를 사용하여 Amazon ECR 리포지토리에 있는 컨테이너 이미지를 스캔하여 취약한 운영 체제 패키지 및 프로그래밍 언어 패키지를 찾을 수 있습니다. 자세한 내용은 아마존 인스펙터와 아마존 엘라스틱 컨테이너 레지스트리 (아마존) 통합 ECR 단원을 참조하십시오.

아마존 인스펙터와 통합 AWS Security Hub

AWS Security Hub보안 상태를 포괄적으로 파악하고 보안 업계 표준 AWS 및 모범 사례와 비교하여 환경을 확인할 수 있도록 도와줍니다. Security Hub는 AWS 계정, 서비스 및 지원되는 제품으로부터 보안 데이터를 수집합니다. Security Hub를 사용하여 Amazon Inspector 조사 결과 데이터를 수집하고 모든 AWS 통합 서비스 AWS 및 파트너 네트워크 제품의 결과를 위한 중앙 위치를 생성할 수 있습니다. 자세한 내용은 아마존 인스펙터와 통합 AWS Security Hub 단원을 참조하십시오.

아마존 인스펙터와 아마존 엘라스틱 컨테이너 레지스트리 (아마존) 통합 ECR

Amazon Elastic 컨테이너 레지스트리는 Docker와 OCI 이미지 및 AWS 아티팩트를 지원하는 완전 관리형 컨테이너 레지스트리입니다. ECRAmazon을 사용하는 경우 컨테이너 레지스트리의 향상된 스캔을 활성화할 수 있습니다. 향상된 스캔을 활성화하면 Amazon Inspector가 컨테이너 이미지를 자동으로 탐지하고 스캔하여 취약한 운영 체제 및 프로그래밍 언어 패키지를 찾아냅니다. 이 통합을 통해 Amazon Inspector에서 컨테이너 이미지에 대한 결과를 확인하고 Amazon 콘솔에서 스캔 빈도와 범위를 관리할 수 있습니다. ECR 자세한 내용은 Amazon Inspector를 사용하여 아마존 ECR 컨테이너 이미지 스캔을 참조하십시오.

통합 활성화

Amazon Inspector 콘솔을 통해 Amazon Inspector 스캔을 API 활성화하거나, Amazon 콘솔 또는 Amazon Inspector를 통한 향상된 스캔 기능을 사용하도록 리포지토리를 구성하여 통합을 활성화할 수 있습니다. ECR API

Amazon Inspector를 통한 통합 활성화에 대한 자세한 내용은 Amazon Inspector의 자동 스캔 유형 섹션을 참조하세요.

Amazon에서 향상된 스캔을 활성화하고 구성하는 방법에 대한 자세한 내용은 Amazon ECR ECR 사용 설명서의 향상된 스캔을 참조하십시오.

다중 계정 환경과의 통합 사용

다중 계정 환경의 회원인 경우 ECR Amazon을 통해 향상된 스캔을 활성화할 수 있습니다. 하지만 활성화한 후에는 Amazon Inspector 위임 관리자만 비활성화할 수 있습니다. 비활성화되면 기본 스캔으로 돌아갑니다. 자세한 내용은 Amazon Inspector 비활성화 단원을 참조하십시오.

아마존 인스펙터와 통합 AWS Security Hub

AWS Security Hub 보안 상태를 포괄적으로 파악하고 보안 업계 표준 AWS 및 모범 사례와 비교하여 환경을 점검할 수 있도록 도와줍니다. Security Hub는 AWS 계정, 서비스 및 지원되는 제품으로부터 보안 데이터를 수집합니다. Security Hub에서 제공하는 정보를 사용하여 보안 동향을 분석하고 가장 우선순위가 높은 보안 문제를 식별할 수 있습니다. 통합을 활성화하면 Amazon Inspector의 결과를 Security Hub로 전송할 수 있으며, Security Hub는 이러한 결과를 보안 상태 분석에 포함시킬 수 있습니다.

Security Hub는 보안 문제를 결과로 추적합니다. 이러한 결과 중 일부는 다른 AWS 서비스 또는 타사 제품에서 감지한 문제로 인해 발생할 수 있습니다. Security Hub는 일련의 규칙을 사용하여 보안 문제를 탐지하고 결과를 생성합니다. Security Hub는 검색 결과를 관리하는 데 도움이 되는 도구를 제공합니다. Security Hub는 Amazon Inspector에서 조사 결과가 종료되면 Amazon Inspector 조사 결과를 보관합니다. 또한 조사 결과 기록 및 세부 정보를 볼 수 있을 뿐만 아니라 조사 결과에 대한 조사 상태를 추적할 수 있습니다.

Security Hub 조사 결과는 AWS 보안 검색 결과 JSON 형식 (ASFF) 이라는 표준 형식을 사용합니다. ASFF여기에는 문제의 원인, 영향을 받는 리소스, 조사 결과의 현재 상태에 대한 세부 정보가 포함됩니다.

주제

- 에서 Amazon Inspector 조사 결과 보기 AWS Security Hub
- Amazon Inspector와 보안 허브 통합 활성화 및 구성
- 통합 결과 흐름 비활성화
- Security Hub에서 Amazon Inspector의 보안 제어 항목 보기

에서 Amazon Inspector 조사 결과 보기 AWS Security Hub

Security Hub에서 아마존 인스펙터 클래식 및 아마존 인스펙터 조사 결과를 확인할 수 있습니다.

> **Note**
>
> Amazon Inspector 검색 결과만 기준으로 `"aws/inspector/ProductVersion": "2"` 필터링하려면 필터 막대에 추가하십시오. 이 필터는 Security Hub 대시보드에서 Amazon Inspector Classic 검색 결과를 제외합니다.

Amazon Inspector 결과 예제

```
{
  "SchemaVersion": "2018-10-08",
  "Id": "arn:aws:inspector2:us-east-1:123456789012:finding/FINDING_ID",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/inspector",
  "ProductName": "Inspector",
  "CompanyName": "Amazon",
  "Region": "us-east-1",
  "GeneratorId": "AWSInspector",
  "AwsAccountId": "123456789012",
  "Types": [
    "Software and Configuration Checks/Vulnerabilities/CVE"
  ],
  "FirstObservedAt": "2023-01-31T20:25:38Z",
  "LastObservedAt": "2023-05-04T18:18:43Z",
  "CreatedAt": "2023-01-31T20:25:38Z",
  "UpdatedAt": "2023-05-04T18:18:43Z",
  "Severity": {
    "Label": "HIGH",
    "Normalized": 70
  },
  "Title": "CVE-2022-34918 - kernel",
```

```
  "Description": "An issue was discovered in the Linux kernel through 5.18.9. A type
 confusion bug in nft_set_elem_init (leading to a buffer overflow) could be used by a
 local attacker to escalate privileges, a different vulnerability than CVE-2022-32250.
 (The attacker can obtain root access, but must start with an unprivileged user
 namespace to obtain CAP_NET_ADMIN access.) This can be fixed in nft_setelem_parse_data
 in net/netfilter/nf_tables_api.c.",
  "Remediation": {
    "Recommendation": {
      "Text": "Remediation is available. Please refer to the Fixed version in the
 vulnerability details section above. For detailed remediation guidance for each of the
 affected packages, refer to the vulnerabilities section of the detailed finding JSON."
    }
  },
  "ProductFields": {
    "aws/inspector/FindingStatus": "ACTIVE",
    "aws/inspector/inspectorScore": "7.8",
    "aws/inspector/resources/1/resourceDetails/awsEc2InstanceDetails/platform":
 "AMAZON_LINUX_2",
    "aws/inspector/ProductVersion": "2",
    "aws/inspector/instanceId": "i-0f1ed287081bdf0fb",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/inspector/
arn:aws:inspector2:us-east-1:123456789012:finding/FINDING_ID",
    "aws/securityhub/ProductName": "Inspector",
    "aws/securityhub/CompanyName": "Amazon"
  },
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "arn:aws:ec2:us-east-1:123456789012:i-0f1ed287081bdf0fb",
      "Partition": "aws",
      "Region": "us-east-1",
      "Tags": {
        "Patch Group": "SSM",
        "Name": "High-SEv-Test"
      },
      "Details": {
        "AwsEc2Instance": {
          "Type": "t2.micro",
          "ImageId": "ami-0cff7528ff583bf9a",
          "IpV4Addresses": [
            "52.87.229.97",
            "172.31.57.162"
          ],
          "KeyName": "ACloudGuru",
```

```
          "IamInstanceProfileArn": "arn:aws:iam::123456789012:instance-profile/
AmazonSSMRoleForInstancesQuickSetup",
          "VpcId": "vpc-a0c2d7c7",
          "SubnetId": "subnet-9c934cb1",
          "LaunchedAt": "2022-07-26T21:49:46Z"
        }
      }
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "Vulnerabilities": [
    {
      "Id": "CVE-2022-34918",
      "VulnerablePackages": [
        {
          "Name": "kernel",
          "Version": "5.10.118",
          "Epoch": "0",
          "Release": "111.515.amzn2",
          "Architecture": "X86_64",
          "PackageManager": "OS",
          "FixedInVersion": "0:5.10.130-118.517.amzn2",
          "Remediation": "yum update kernel"
        }
      ],
      "Cvss": [
        {
          "Version": "2.0",
          "BaseScore": 7.2,
          "BaseVector": "AV:L/AC:L/Au:N/C:C/I:C/A:C",
          "Source": "NVD"
        },
        {
          "Version": "3.1",
          "BaseScore": 7.8,
          "BaseVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H",
          "Source": "NVD"
        },
        {
          "Version": "3.1",
```

```
          "BaseScore": 7.8,
          "BaseVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H",
          "Source": "NVD",
          "Adjustments": []
        }
      ],
      "Vendor": {
        "Name": "NVD",
        "Url": "https://nvd.nist.gov/vuln/detail/CVE-2022-34918",
        "VendorSeverity": "HIGH",
        "VendorCreatedAt": "2022-07-04T21:15:00Z",
        "VendorUpdatedAt": "2022-10-26T17:05:00Z"
      },
      "ReferenceUrls": [
        "https://git.kernel.org/pub/scm/linux/kernel/git/netdev/net.git/commit/?
id=7e6bc1f6cabcd30aba0b11219d8e01b952eacbb6",
        "https://lore.kernel.org/netfilter-devel/cd9428b6-7ffb-dd22-d949-
d86f4869f452@randorisec.fr/T/",
        "https://www.debian.org/security/2022/dsa-5191"
      ],
      "FixAvailable": "YES"
    }
  ],
  "FindingProviderFields": {
    "Severity": {
      "Label": "HIGH"
    },
    "Types": [
      "Software and Configuration Checks/Vulnerabilities/CVE"
    ]
  },
  "ProcessedAt": "2023-05-05T20:28:38.822Z"
}
```


Amazon Inspector와 보안 허브 통합 활성화 및 구성

보안 허브를 활성화하여 Amazon Inspector 통합을 AWS Security Hub 활성화할 수 있습니다. 보안 허브를 활성화하면 Amazon Inspector AWS Security Hub 통합이 자동으로 활성화되고 Amazon Inspector는 보안 검색 결과 형식 () 을 사용하여AWS 모든 탐지 결과를 보안 허브로 보내기 시작합니다. ASFF

통합 결과 흐름 비활성화

Amazon Inspector가 검색 결과를 Security Hub로 보내는 것을 중지하려면 Security Hub 콘솔을 사용하거나... API AWS CLI

Security Hub에서 Amazon Inspector의 보안 제어 항목 보기

Security Hub는 지원되는 제품 AWS 및 타사 제품의 결과를 분석하고 규칙에 대해 자동화되고 지속적인 보안 검사를 실행하여 자체 결과를 생성합니다. 규칙은 표준 요구 사항이 충족되는지 여부를 판단하는 데 도움이 되는 보안 제어로 표현됩니다.

Amazon Inspector는 보안 제어를 사용하여 Amazon Inspector 기능이 활성화되었는지 또는 활성화되어야 하는지를 확인합니다. 이러한 기능은 다음과 같습니다.

- 아마존 EC2 스캔
- 아마존 ECR 스캔
- Lambda 표준 스캔
- Lambda 코드 스캔

자세한 내용은 AWS Security Hub 사용 설명서의 Amazon Inspector 컨트롤을 참조하십시오.

Amazon Inspector에서 지원되는 운영 체제 및 프로그래밍 언어

Amazon Inspector는 다음에 설치된 소프트웨어 애플리케이션을 스캔할 수 있습니다.

- Amazon Elastic Compute Cloud(AmazonEC2) 인스턴스
- Amazon Elastic Container Registry(AmazonECR) 리포지토리에 저장된 컨테이너 이미지

> **Note**
>
> ECR 컨테이너 이미지의 경우 Amazon Inspector는 운영 체제 및 프로그래밍 언어 패키지 취약성을 스캔할 수 있습니다.

- AWS Lambda 함수

> **Note**
>
> Lambda 함수의 경우 Amazon Inspector는 코드 취약성을 스캔할 수 있습니다.

Amazon Inspector가 리소스를 스캔하면 Amazon Inspector는 50개 이상의 데이터 피드를 소싱하여 일반적인 취약성 및 노출에 대한 조사 결과를 생성합니다(CVEs). 이러한 소스의 예로는 공급업체 보안 권고 데이터 피드 및 위협 인텔리전스 피드와 국가 취약성 데이터베이스(NVD) 및 가 있습니다MITRE. Amazon Inspector는 소스 피드의 취약성 데이터를 하루에 한 번 이상 업데이트합니다.

Amazon Inspector가 리소스를 스캔하려면 리소스가 지원되는 운영 체제를 실행하거나 지원되는 프로그래밍 언어를 사용해야 합니다. 이 섹션의 주제에서는 Amazon Inspector가 다양한 리소스 및 스캔 유형에 대해 지원하는 운영 체제, 프로그래밍 언어 및 런타임을 나열합니다. 또한 중단된 운영 체제도 나열됩니다.

> **Note**
>
> 공급업체가 지원을 중단한 운영 체제에 대해서는 Amazon Inspector에서 제한된 지원만 제공할 수 있습니다.

주제

- 지원되는 운영 체제
- 중단된 운영 체제
- 지원되는 프로그래밍 언어
- 지원되는 런타임

지원되는 운영 체제

이 섹션에서는 Amazon Inspector가 지원하는 운영 체제를 나열합니다.

지원되는 운영 체제: Amazon EC2 스캔

다음 표에는 Amazon Inspector가 Amazon EC2 인스턴스 스캔을 위해 지원하는 운영 체제가 나열되어 있습니다. 각 운영 체제에 대한 공급업체 보안 권고 사항과 에이전트 기반 스캔 및 에이전트리스 스캔을 지원하는 운영 체제를 지정합니다.

> **Note**
>
> 에이전트 기반 스캔 방법을 사용하는 경우 모든 적격 인스턴스에 대해 연속 스캔을 수행하도록 SSM 에이전트를 구성합니다. Amazon Inspector는 3.2.2086.0보다 큰 버전의 SSM 에이전트를 구성할 것을 권장합니다. 자세한 내용은 Amazon EC2 Systems Manager 사용 설명서의 SSM 에이전트 작업을 참조하세요.

Linux 운영 체제 감지는 기본 패키지 관리자 리포지토리(rpm 및 dpkg)에서만 지원되며 타사 애플리케이션, 확장 지원 리포지토리(BYOS RHEL, PAYG RHEL, RHEL의 경우 SAP), 선택적 리포지토리(애플리케이션 스트림)는 포함되지 않습니다.

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
AlmaLinux	8	ALSA	예	예
AlmaLinux	9	ALSA	예	예
Amazon Linux(AL2)	AL2	ALAS	예	예

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
Amazon Linux 2023(AL2023)	AL2023	ALAS	예	예
Bottlerocket	1.7.0 이상	GHSA, CVE	아니요	예
Debian Server(Bullseye)	11	DSA	예	예
Debian Server(Bookworm)	12	DSA	예	예
Fedora	39	CVE	예	예
Fedora	40	CVE	예	예
Leap 열기SUSE	15.5	CVE	예	예
Oracle Linux(Oracle)	7	ELSA	예	예
Oracle Linux(Oracle)	8	ELSA	예	예
Oracle Linux(Oracle)	9	ELSA	예	예
Red Hat Enterprise Linux(RHEL)	8	RHSA	예	예
Red Hat Enterprise Linux(RHEL)	9	RHSA	예	예
Rocky Linux	8	RLSA	예	예

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
Rocky Linux	9	RLSA	예	예
SUSE Linux 엔터프라이즈 서버 (SLES)	12.5	SUSE CVE	예	예
SUSE Linux 엔터프라이즈 서버 (SLES)	15.5	SUSE CVE	예	예
Ubuntu(Xenial)	16.04(ESM)	USN, Ubuntu Pro	예	예
Ubuntu(Bionic)	18.04(ESM)	USN, Ubuntu Pro	예	예
Ubuntu(Focal)	20.04(LTS)	USN	예	예
Ubuntu(Jammy)	22.04(LTS)	USN	예	예
Ubuntu (Noble Numbat)	24.04	USN	예	예
Windows Server	2016	MSKB	아니요	예
Windows Server	2019	MSKB	아니요	예
Windows Server	2022	MSKB	아니요	예
macOS(Mojave)	10.14	APPLE-SA	아니요	예
macOS(Catalina)	10.15	APPLE-SA	아니요	예
macOS(Big Sur)	11	APPLE-SA	아니요	예
macOS(Monterey)	12	APPLE-SA	아니요	예

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
macOS(벤투라)	13	APPLE-SA	아니요	예

지원되는 운영 체제: Amazon Inspector를 사용한 Amazon ECR 스캔

다음 표에는 Amazon Inspector가 Amazon ECR리포지토리의 컨테이너 이미지 스캔에 지원하는 운영 체제가 나열되어 있습니다. 또한 각 운영 체제에 대한 공급업체 보안 권고를 지정합니다.

운영 체제	버전	공급업체 보안 권고
Alpine Linux (Alpine)	3.17	Alpine SecDB
Alpine Linux (Alpine)	3.18	Alpine SecDB
Alpine Linux (Alpine)	3.19	Alpine SecDB
Alpine Linux (Alpine)	3.20	Alpine SecDB
AlmaLinux	8	ALSA
AlmaLinux	9	ALSA
Amazon Linux (AL2)	AL2	ALAS
Amazon Linux 2023 (AL2023)	AL2023	ALAS
Debian Server (Bullseye)	11	DSA
Debian Server (Bookworm)	12	DSA
Fedora	39	CVE
Fedora	40	CVE
OpenSUSE Leap	15.5	CVE
Oracle Linux (Oracle)	7	ELSA

운영 체제	버전	공급업체 보안 권고
Oracle Linux (Oracle)	8	ELSA
Oracle Linux (Oracle)	9	ELSA
Photon OS	4	PHSA
Photon OS	5	PHSA
Red Hat Enterprise Linux (RHEL)	8	RHSA
Red Hat Enterprise Linux (RHEL)	9	RHSA
Rocky Linux	8	RLSA
Rocky Linux	9	RLSA
SUSE Linux Enterprise Server (SLES)	12.5	SUSE CVE
SUSE Linux Enterprise Server (SLES)	15.5	SUSE CVE
Ubuntu (Xenial)	16.04 (ESM)	USN, Ubuntu Pro
Ubuntu (Bionic)	18.04 (ESM)	USN, Ubuntu Pro
Ubuntu (Focal)	20.04 (LTS)	USN
Ubuntu (Jammy)	22.04 (LTS)	USN
Ubuntu (Noble Numbat)	24.04	USN

지원되는 운영 체제: CIS 스캔

다음 표에는 Amazon Inspector가 CIS 스캔에 지원하는 운영 체제가 나열되어 있습니다. 또한 각 운영 체제의 CIS 벤치마크 버전도 지정합니다.

운영 체제	버전	CIS 벤치마크 버전
Amazon Linux 2	AL2	2.0.0
Amazon Linux 2023	AL2023	1.0.0
Red Hat Enterprise Linux(RHEL)	8	3.0.0
Red Hat Enterprise Linux(RHEL)	9	1.0.0
Rocky Linux	8	2.0.0
Rocky Linux	9	1.0.0
Ubuntu(Bonic)	18.04(LTS)	2.1.0
Ubuntu(Focal)	20.04(LTS)	2.0.1
Ubuntu(Jammy)	22.04(LTS)	1.0.0
Windows Server	2016	3.0.0
Windows Server	2019	2.0.0
Windows Server	2022	2.0.0

중단된 운영 체제

다음 표에는 중단된 운영 체제와 중단된 시간이 나와 있습니다.

Amazon Inspector가 다음과 같이 중단된 운영 체제에 대한 완전한 지원을 제공하지는 않지만 Amazon Inspector는 해당 운영 체제를 실행하는 Amazon EC2 인스턴스 및 Amazon ECR 컨테이너 이미지를 계속 스캔합니다. 보안 모범 사례로 중단된 운영 체제의 지원되는 버전으로 이동하는 것이 좋습니다. Amazon Inspector가 중단된 운영 체제에 대해 생성하는 조사 결과는 정보 제공 목적으로만 사용해야 합니다.

공급업체 정책에 따라 다음 운영 체제는 더 이상 패치 업데이트를 수신하지 않습니다. 중단된 운영 체제에 대해 새 보안 권고가 릴리스되지 않을 수 있습니다. 공급업체는 표준 지원이 종료된 운영 체제에

대한 피드에서 기존 보안 권고 및 탐지를 제거할 수 있습니다. 따라서 Amazon Inspector는 알려진 에 대한 결과 생성을 중지할 수 있습니다CVEs.

중단된 운영 체제: Amazon EC2 스캔

운영 체제	버전	중단됨
Amazon Linux(AL1)	2012	2021년 12월 31일
CentOS Linux(CentOS)	7	2024년 6월 30일
CentOS Linux(CentOS)	8	2021년 12월 31일
Debian Server(Stretch)	9	2022년 6월 30일
Debian Server(Buster)	10	2024년 6월 30일
Fedora	35	2022년 12월 13일
Fedora	36	2023년 5월 16일
Fedora	37	2023년 12월 15일
Fedora	38	2024년 5월 21일
Leap 열기SUSE	15.2	2021년 12월 1일
OpenSUSE Leap	15.3	2022년 12월 1일
Leap 열기SUSE	15.4	2023년 12월 7일
Oracle Linux(Oracle)	6	2021년 3월 1일
Red Hat Enterprise Linux(RHEL)	7	2024년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	12	2016년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	12.1	2017년 5월 31일

운영 체제	버전	중단됨
SUSE Linux 엔터프라이즈 서버(SLES)	12.2	2018년 3월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	12.3	2019년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	12.4	2020년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	15	2019년 12월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.1	2021년 1월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.2	2021년 12월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.3	2022년 12월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.4	2023년 12월 31일
Ubuntu(Trusty)	14.04(ESM)	2024년 4월 1일
Ubuntu(Groovy)	20.10	2021년 7월 22일
Ubuntu(Hirsute)	2004년 21월	2022년 1월 20일
Ubuntu(Impish)	21.10	2022년 7월 31일
Ubuntu (Kinetic)	22.10	July 20, 2023
Ubuntu (Lunar Lobster)	23.04	January 25, 2024
Ubuntu(Mantic Minotaur)	23.10	2024년 7월 11일
Windows Server	2012	2023년 10월 10일

운영 체제	버전	중단됨
Windows Server	2012 R2	2023년 10월 10일

중단된 운영 체제: Amazon ECR 스캔

운영 체제	버전	중단됨
Alpine Linux(Alpine)	3.12	2022년 5월 1일
Alpine Linux(Alpine)	3.13	2022년 11월 1일
Alpine Linux (Alpine)	3.14	May 1, 2023
Alpine Linux (Alpine)	3.15	November 1, 2023
Alpine Linux (Alpine)	3.16	May 23, 2024
Amazon Linux(AL1)	2012	2021년 12월 31일
CentOS Linux(CentOS)	7	2024년 6월 30일
CentOS Linux(CentOS)	8	2021년 12월 31일
Debian Server(Stretch)	9	2022년 6월 30일
Debian Server(Buster)	10	2024년 6월 30일
Fedora	35	2022년 12월 13일
Fedora	36	2023년 5월 16일
Fedora	37	2023년 12월 15일
Fedora	38	2024년 5월 21일
OpenSUSE Leap	15.2	2021년 12월 1일
OpenSUSE Leap	15.3	2022년 12월 1일

운영 체제	버전	중단됨
OpenSUSE Leap	15.4	December 7, 2023
Oracle Linux(Oracle)	6	2021년 3월 1일
Photon OS	3	2024년 3월 1일
Red Hat Enterprise Linux(RHEL)	7	2024년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	12	2016년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	12.1	2017년 5월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	12.2	2018년 3월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	12.3	2019년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	12.4	2020년 6월 30일
SUSE Linux 엔터프라이즈 서버(SLES)	15	2019년 12월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.1	2021년 1월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.2	2021년 12월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.3	2022년 12월 31일
SUSE Linux 엔터프라이즈 서버(SLES)	15.4	2023년 12월 31일

운영 체제	버전	중단됨
Ubuntu(Trusty)	14.04(ESM)	2024년 4월 1일
Ubuntu(Groovy)	20.10	2021년 7월 22일
Ubuntu(Hirsute)	2004년 21월	2022년 1월 20일
Ubuntu(Impish)	21.10	2022년 7월 31일
Ubuntu (Kinetic)	22.10	July 20, 2023
Ubuntu (Lunar Lobster)	23.04	January 25, 2024
Ubuntu(Mantic Minotaur)	23.10	2024년 7월 11일

지원되는 프로그래밍 언어

이 섹션에서는 Amazon Inspector가 지원하는 프로그래밍 언어 목록을 제공합니다.

지원되는 프로그래밍 언어: Amazon EC2 에이전트리스 스캔

Amazon Inspector는 현재 적격 Amazon EC2 인스턴스에서 에이전트리스 스캔을 수행할 때 다음 프로그래밍 언어를 지원합니다. 자세한 내용은 에이전트리스 스캔을 참조하세요.

- C#
- Go
- Java
- JavaScript
- PHP
- Python
- Ruby
- Rust

지원되는 프로그래밍 언어: Amazon EC2 심층 검사

Amazon Inspector는 현재 Amazon EC2 Linux 인스턴스에서 심층 검사 스캔을 수행할 때 다음 프로그래밍 언어를 지원합니다. 자세한 내용은 Linux 기반 Amazon 인스턴스용 Amazon Inspector 딥 인스페이션을 EC2 참조하세요.

- Java (.ear, .jar, .par 및 .war 아카이브 형식)
- JavaScript
- Python

Amazon Inspector는 Systems Manager Distributor를 사용하여 Amazon EC2 인스턴스에 대한 심층 검사를 위해 플러그인을 배포합니다.

> **Note**
>
> Bottlerocket 운영 체제에서는 심층 검사가 지원되지 않습니다.

심층 검사 스캔을 수행하려면 Systems Manager Distributor 및 Amazon Inspector가 Amazon EC2 인스턴스 운영 체제를 지원해야 합니다. Systems Manager Distributor에서 지원되는 운영 체제에 대한 자세한 내용은 Systems Manager 사용 설명서의 지원되는 패키지 플랫폼 및 아키텍처를 참조하세요.

지원되는 프로그래밍 언어: Amazon ECR 스캔

Amazon Inspector는 현재 Amazon 리ECR포지토리에서 컨테이너 이미지를 스캔할 때 다음 프로그래밍 언어를 지원합니다.

- C#
- Go
- Java
- JavaScript
- PHP
- Python
- Ruby
- Rust

지원되는 런타임

이 섹션에서는 Amazon Inspector가 지원하는 런타임을 나열합니다.

지원되는 런타임: Amazon Inspector Lambda 표준 스캔

Amazon Inspector Lambda 표준 스캔은 현재 타사 소프트웨어 패키지의 취약성에 대해 Lambda 함수를 스캔할 때 사용할 수 있는 프로그래밍 언어에 대해 다음과 같은 런타임을 지원합니다.

- Go
  - go1.x
- Java
  - java8
  - java8.al2
  - java11
  - java17
  - java21
- .NET
  - .NET 6
  - .NET 8
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
  - python3.12

- Ruby
  - ruby2.7
  - ruby3.2
  - ruby3.3
- Custom runtimes
  - AL2
  - AL2023

지원되는 런타임: Amazon Inspector Lambda 코드 스캔

Amazon Inspector Lambda 코드 스캔은 현재 Lambda 함수에서 코드의 취약성을 스캔할 때 사용할 수 있는 프로그래밍 언어에 대해 다음과 같은 런타임을 지원합니다.

- Java
  - java8
  - java8.al2
  - java11
  - java17
- .NET
  - .NET 6
  - .NET 8
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10

  - python3.11
  - python3.12
- Ruby
  - ruby2.7
  - ruby3.2
  - ruby3.3

Amazon Inspector 비활성화

아마존 인스펙터 콘솔에서 또는 아마존 인스펙터를 사용하여 아마존 인스펙터를 비활성화할 수 있습니다. API 계정의 모든 스캔 유형을 비활성화하면 해당 계정에 대해 Amazon Inspector가 자동으로 비활성화됩니다.

계정에 대해 Amazon Inspector를 비활성화하면 해당 계정에 대한 모든 스캔 유형이 비활성화됩니다. 또한 계정에 대한 모든 Amazon Inspector 스캔 설정 (필터, 금지 규칙 및 결과 포함) 이 삭제됩니다.

Amazon Inspector 아마존 스캐닝을 비활성화하면 EC2 Amazon Inspector는 다음과 같은 연결을 삭제합니다. SSM

- `InspectorDistributor-do-not-delete`
- `InspectorInventoryCollection-do-not-delete`
- `InvokeInspectorSsmPlugin-do-not-delete`. 또한 이 연결을 통해 설치된 Amazon Inspector SSM 플러그인이 모든 호스트에서 제거됩니다. Windows 자세한 내용은 스캔 Windows EC2 인스턴스 단원을 참조하십시오.

> **Note**
>
> Amazon Inspector를 비활성화하면 더 이상 서비스 요금이 발생하지 않습니다. 하지만 언제든지 Amazon Inspector를 다시 활성화할 수 있습니다.

다양한 리소스의 스캔 유형을 비활성화하는 방법에 대한 자세한 내용은 스캔 유형 비활성화를 참조하십시오.

사전 조건

계정 유형에 따라 다음 사항을 고려하십시오.

- 계정이 독립형 Amazon Inspector 계정인 경우 언제든지 Amazon Inspector를 비활성화할 수 있습니다.
- 다중 계정 환경의 멤버 계정인 경우 Amazon Inspector를 비활성화할 수 없습니다. Amazon Inspector를 비활성화하려면 조직의 위임 관리자에게 문의해야 합니다.
- 조직의 위임 관리자인 경우 Amazon Inspector를 비활성화하기 전에 모든 멤버 계정을 연결 해제해야 합니다.

> **Note**
> 위임 관리자로 Amazon Inspector를 비활성화하면 조직의 자동 활성화 기능이 비활성화됩니다.

Amazon Inspector 비활성화

> **Note**
> Amazon Inspector를 비활성화하기 전에 결과를 내보내는 것을 고려해 보십시오.

Console

Amazon Inspector를 비활성화하려면

1. 자격 증명을 사용하여 로그인한 다음 v2/home에서 Amazon Inspector 콘솔을 엽니다. https://console.aws.amazon.com/inspector/
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 Amazon Inspector를 비활성화하려는 지역을 선택합니다.
3. 탐색 창에서 일반 설정을 선택합니다.
4. Inspector 비활성화를 선택합니다.
5. 확인 메시지가 표시되면 텍스트 상자에 deactivate를 입력한 다음 Inspector 비활성화를 선택합니다.
6. (권장) Amazon Inspector를 비활성화할 각 리전에 대해 이 단계를 반복합니다.

API

비활성화 작업을 실행합니다. API 요청 시 IDs 비활성화하려는 계정을 입력하고 스캔을 모두 `resourceTypes` 비활성화할 것인지 입력합니다. 그러면 계정이 비활성화됩니다. `EC2, ECR, LAMBDA`

아마존 인스펙터 쿼터

이 섹션에는 Amazon Inspector 할당량이 나열되어 있습니다. AWS 리전

Resource	기본값	설명
멤버 계정	10,000개	Amazon Inspector 위임 관리자 계정과 연결된 최대 멤버 계정 수입니다. 한도는 할당량을 기준으로 합니다. AWS Organizations
억제 규칙	500	지역별 AWS 계정당 저장된 금지 규칙의 최대 수입니다. 할당량 증가를 요청할 수 없습니다.
Amazon EC2 네트워크 조사 결과	10,000개	AWS 계정당 Amazon EC2 네트워크 검색 결과의 최대 수입니다. 할당량 증가를 요청할 수 없습니다.
CIS스캔 구성	500	CIS스캔 구성의 최대 수입니다. 할당량 증가를 요청할 수 없습니다.

Amazon Inspector Classic과 관련된 할당량 목록은 의 Amazon Inspector Classic 서비스 할당량을 참조하십시오. AWS 일반 참조 관련 할당량 목록은 의 서비스 할당량을 참조하십시오. AWS OrganizationsAWS OrganizationsAWS 일반 참조

리전 및 엔드포인트

이 주제에는 Amazon Inspector API Scan의 엔드포인트를 보여 주고 Amazon Inspector 기능을 AWS 리전 지원하는 표가 포함되어 있습니다.

Amazon Inspector를 사용할 수 AWS 리전 있는 위치를 보려면 의 Amazon Inspector 엔드포인트를 참조하십시오. Amazon Web Services 일반 참조

Amazon Inspector 스캔용 엔드포인트 API

다음 표에는 Amazon API Inspector Scan을 호출할 때 사용할 수 있는 리전 엔드포인트가 나와 있습니다. 를 사용할 때는 엔드포인트를 제공해야 하며, 엔드포인트는 현재 인증된 AWS 지역에 해당하는 지역입니다. API

Amazon Inspector 스캔 엔드포인트의 명명 규칙은 `inspector-scan.`*`region`*`.amazonaws.com`입니다. 예를 들어, 인증을 받은 경우 `us-west-2` 엔드포인트를 사용하여 를 `inspector-scan.us-west-2.amazonaws.com` 호출할 수 있습니다. `inspector-scan` API

리전 이름	지역	엔드포인트	프로토콜
미국 동부(오하이오)	us-east-2	inspector-scan.us-east-2.amazonaws.com inspector-scan-fips.us-east-2.amazonaws.com	HTTPS
미국 동부(버지니아 북부)	us-east-1	inspector-scan.us-east-1.amazonaws.com inspector-scan-fips.us-east-1.amazonaws.com	HTTPS

리전 이름	지역	엔드포인트	프로토콜
미국 서부(캘리포니아 북부)	us-west-1	inspector-scan.us-west-1.amazonaws.com inspector-scan-fips.us-west-1.amazonaws.com	HTTPS
미국 서부(오레곤)	us-west-2	inspector-scan.us-west-2.amazonaws.com inspector-scan-fips.us-west-2.amazonaws.com	HTTPS
아프리카(케이프타운)	af-south-1	inspector-scan.af-south-1.amazonaws.com	HTTPS
아시아 태평양(홍콩)	ap-east-1	inspector-scan.ap-east-1.amazonaws.com	HTTPS
아시아 태평양(자카르타)	ap-southeast-3	inspector-scan.ap-southeast-3.amazonaws.com	HTTPS
아시아 태평양(뭄바이)	ap-south-1	inspector-scan.ap-south-1.amazonaws.com	HTTPS
아시아 태평양(오사카)	ap-northeast-3	inspector-scan.ap-northeast-3.amazonaws.com	HTTPS

리전 이름	지역	엔드포인트	프로토콜
아시아 태평양(서울)	ap-northeast-2	inspector-scan.ap-northeast-2.amazonaws.com	HTTPS
아시아 태평양(싱가포르)	ap-southeast-1	inspector-scan.ap-southeast-1.amazonaws.com	HTTPS
아시아 태평양(시드니)	ap-southeast-2	inspector-scan.ap-southeast-2.amazonaws.com	HTTPS
아시아 태평양(도쿄)	ap-northeast-1	inspector-scan.ap-northeast-1.amazonaws.com	HTTPS
캐나다(중부)	ca-central-1	inspector-scan.ca-central-1.amazonaws.com	HTTPS
유럽(프랑크푸르트)	eu-central-1	inspector-scan.eu-central-1.amazonaws.com	HTTPS
유럽(아일랜드)	eu-west-1	inspector-scan.eu-west-1.amazonaws.com	HTTPS
유럽(런던)	eu-west-2	inspector-scan.eu-west-2.amazonaws.com	HTTPS
유럽(밀라노)	eu-south-1	inspector-scan.eu-south-1.amazonaws.com	HTTPS

리전 이름	지역	엔드포인트	프로토콜
유럽(파리)	eu-west-3	inspector-scan.eu-west-3.amazonaws.com	HTTPS
유럽(스톡홀름)	eu-north-1	inspector-scan.eu-north-1.amazonaws.com	HTTPS
유럽(취리히)	eu-central-2	inspector-scan.eu-central-2.amazonaws.com	HTTPS
중동(바레인)	me-south-1	inspector-scan.me-south-1.amazonaws.com	HTTPS
남아메리카(상파울루)	sa-east-1	inspector-scan.sa-east-1.amazonaws.com	HTTPS
AWS GovCloud (미국 동부)	us-gov-east-1	인스펙터 스캔. us-gov-east-1.amazonaws.com inspector-scan-fips.us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud (미국 서부)	us-gov-west-1	인스펙터 스캔. us-gov-west-1.amazonaws.com inspector-scan-fips.us-gov-west-1.amazonaws.com	HTTPS

리전별 기능 가용성

이 섹션에서는 AWS 리전별 Amazon Inspector 기능의 사용 가능 여부에 대해 설명합니다.

Amazon 지역을 위한 에이전트 없는 EC2 스캔 EC2

다음 표는 Amazon의 에이전트 없는 EC2 스캐닝이 현재 제공되는 AWS 리전 지역을 보여줍니다.

리전 이름	리전 코드
미국 동부(버지니아 북부)	us-east-1
미국 동부(오하이오)	us-east-2
미국 서부(캘리포니아 북부)	us-west-1
미국 서부(오레곤)	us-west-2
아프리카(케이프타운)	af-south-1
아시아 태평양(홍콩)	ap-east-1
아시아 태평양(도쿄)	ap-northeast-1
아시아 태평양(서울)	ap-northeast-2
아시아 태평양(오사카)	ap-northeast-3
아시아 태평양(뭄바이)	ap-south-1
아시아 태평양(싱가포르)	ap-southeast-1
아시아 태평양(시드니)	ap-southeast-2
아시아 태평양(자카르타)	ap-southeast-3
캐나다(중부)	ca-central-1
유럽(스톡홀름)	eu-north-1
유럽(프랑크푸르트)	eu-central-1

리전 이름	리전 코드
유럽(취리히)	eu-central-2
유럽(아일랜드)	eu-west-1
유럽(런던)	eu-west-2
유럽(파리)	eu-west-3
유럽(밀라노)	eu-south-1
중동(바레인)	me-south-1
남아메리카(상파울루)	sa-east-1
AWS GovCloud (미국 동부)	us-gov-east-1
AWS GovCloud (미국 서부)	us-gov-west-1

Lambda 코드 스캔 리전

다음 표는 현재 Lambda 코드 스캔을 사용할 수 AWS 리전 있는 지역을 보여줍니다.

리전 이름	리전 코드
미국 동부(버지니아 북부)	us-east-1
미국 서부(오레곤)	us-west-2
미국 동부(오하이오)	us-east-2
아시아 태평양(시드니)	ap-southeast-2
아시아 태평양(도쿄)	ap-northeast-1
유럽(프랑크푸르트)	eu-central-1
유럽(아일랜드)	eu-west-1
유럽(런던)	eu-west-2

리전 이름	리전 코드
유럽(스톡홀름)	eu-north-1
아시아 태평양(싱가포르)	ap-southeast-1

> **Important**
>
> Lambda 코드 스캔을 사용할 수 없는 AWS 리전 곳에서 Amazon Inspector Enable을 사용하여 Lambda 코드 스캔을 API활성화하려고 하면 다음과 같은 액세스 거부 오류가 발생합니다.
>
> ```
> An error occurred (AccessDeniedException) when calling the Enable operation:
>  Lambda code scanning is not supported in unsupported-AWS ##
> ```

AWS GovCloud (US) 리전

자세한 내용은 AWS GovCloud (US) 사용 설명서에서 Amazon Inspector 섹션을 참조하세요.

문서 기록

다음 표에는 2021년 11월에 시작되는 Amazon Inspector 사용 설명서의 각 릴리스에서 변경된 주요 내용이 설명되어 있습니다. RSS피드를 구독하면 설명서 업데이트에 대한 알림을 받을 수 있습니다.

변경 사항	설명	날짜
업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 명령문을 추가했습니다. 새 명령문을 사용하면 Amazon Inspector에서 함수 태그를 반환할 수 있습니다. AWS Lambda	2024년 7월 31일
업데이트된 기능	Amazon Inspector는 새로운 보안 제어를 출시합니다. 자세한 내용은 AWS Security Hub 사용 설명서의 Amazon Inspector 컨트롤을 참조하십시오.	2024년 7월 11일
업데이트된 기능	Amazon Inspector SBOM Generator는 이제 Dockerfile 및 Docker 컨테이너 이미지를 스캔하여 보안 취약성을 유발할 수 있는 잘못된 구성이 있는지 확인합니다. 자세한 내용은 Amazon Inspector 도커파일 검사를 참조하십시오.	2024년 6월 10일
업데이트된 기능	Amazon Inspector는 CodeCatalyst 작업을 지원하도록 CI/CD 통합 기능을 업데이트하므로 워크플로에 Amazon Inspector 취약성 스캔을 추가할 수 있습니다. CodeCatalyst	2024년 6월 7일

	자세한 내용은 작업 사용을 참조하십시오. CodeCatalyst	
업데이트된 기능	Amazon Inspector에는 CIS 스캔 결과 CSV 파일을 다운로드하는 옵션이 포함되어 있습니다. 자세한 내용은 Amazon EC2 인스턴스용 인터넷 보안 센터 (CIS) 검사의 CIS검사 결과 보기 및 다운로드를 참조하십시오.	2024년 5월 3일
업데이트된 기능	Amazon Inspector는 CI/CD 통합 기능을 GitHub Actions 지원하도록 업데이트하므로 워크플로에 Amazon Inspector 취약성 스캔을 추가할 수 있습니다. GitHub 자세한 내용은 Amazon Inspector와 함께 사용하기 항목을 참조하십시오. GitHub Actions	2024년 4월 29일
업데이트된 기능	Amazon Inspector는 관리형 정책을 `AmazonInspector2FullAccess` 업데이트하여 서비스 연결 역할을 생성합니다. `AWSServiceRoleForAmazonInspector2Agentless` 이를 통해 사용자는 Amazon Inspector를 활성화하면 에이전트 기반 스캔 및 에이전트 없는 스캔을 수행할 수 있습니다.	2024년 4월 24일

업데이트된 기능	Amazon Inspector는 마감된 조사 결과에 대한 보존 기간을 30일에서 7일로 업데이트합니다. 자세한 내용은 Amazon Inspector의 결과 이해를 참조하십시오.	2024년 2월 12일
업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 명령문을 추가했습니다. 새 명령문을 사용하면 Amazon Inspector에서 인스턴스 CIS 스캔을 시작할 수 있습니다.	2024년 1월 23일
새 정책	Amazon Inspector는 인스턴스 프로필의 일부로 사용하여 인스턴스 CIS 스캔을 허용할 수 있는 새 AmazonInspector2ManagedCisPolicy정책인 정책을 추가했습니다.	2024년 1월 23일
새 기능	Amazon Inspector는 이제 컨테이너 이미지를 가져올 때 해당 이미지의 ECR재스캔 시간을 새로 고칩니다. 푸시 또는 풀 날짜를 기준으로 재스캔 기간을 변경하려면 재스캔 기간 구성을 참조하십시오. ECR	2024년 1월 23일
새 기능	Amazon Inspector는 이제 인스턴스에서 인터넷 보안 센터 (CIS) 스캔을 실행할 수 있습니다. EC2 자세한 내용은 Amazon Inspector CIS 스캔을 참조하십시오.	2024년 1월 23일

새 기능	Amazon Inspector는 이제 CI/CD 파이프라인에서 컨테이너 이미지를 스캔할 수 있습니다. 자세한 내용은 CI/CD integration with Amazon Inspector를 참조하세요.	2023년 11월 30일
새 정책	Amazon Inspector는 Amazon Inspector가 에이전트 없는 스캔을 위해 인스턴스에서 EBS 아마존 스냅샷을 스캔할 EC2 수 있도록 허용하는 새 정책을 추가했습니다. 정책에 대한 자세한 내용은 에이전트 없는 스캔을 참조하세요.	2023년 11월 27일
새 기능	Amazon Inspector는 이제 에이전트 없는 스캔을 통해 SSM 에이전트 없이 지원되는 Linux Amazon EC2 인스턴스를 스캔할 수 있도록 지원합니다. 자세한 내용은 에이전트 없는 스캔을 참조하세요.	2023년 11월 27일
새로 지원되는 리소스	Amazon Inspector는 이제 macOS 아마존 인스턴스 스캔을 지원합니다. EC2 지원되는 운영 체제: 지원되는 macOS EC2 버전에 대한 Amazon 스캔을 참조하십시오.	2023년 10월 5일
새로운 리전	이제 아시아 태평양(자카르타), 아프리카(케이프타운), 아시아 태평양(오사카), 유럽(취리히)에서 Amazon Inspector를 사용할 수 있습니다.	2023년 9월 29일

새 기능	이제 제외 태그를 사용하여 Amazon Inspector 스캔에서 EC2 인스턴스를 제외할 수 있습니다.	2023년 9월 14일
새 기능	Amazon Inspector는 Amazon Inspector가 Elastic Load Balancing 대상 그룹에 속하는 EC2 아마존 인스턴스의 네트워크 구성을 스캔할 수 있도록 하는 새로운 권한을 추가했습니다.	2023년 8월 31일
새 기능	Amazon Inspector에서 이제 패키지 취약성 발견에 대한 취약성 인텔리전스 세부 정보를 제공합니다.	2023년 7월 31일
업데이트된 기능	Amazon Inspector는 읽기 전용 사용자가 자신의 리소스에 대한 소프트웨어 BOM (SBOM)을 내보낼 수 있는 새로운 권한을 추가했습니다.	2023년 6월 29일
새 기능	이제 Amazon Inspector에서 스캔 중인 리소스를 SBOM 익스포트할 수 있습니다.	2023년 6월 13일
새 기능	Lambda 코드 스캔이 이제 완전히 일반 공개되었습니다. Lambda 코드 스캔 결과에서 식별된 코드를 암호화할 수 있는 새로운 기능이 추가되었습니다. 또한 Lambda 코드 스캐닝은 이제 코드에 대한 권장 수정 재작성을 제공합니다.	2023년 6월 13일

업데이트된 기능	Amazon Inspector에서 AmazonInspector2ReadOnlyAccess 정책에 새 명령문을 추가했습니다. 새 명령문은 읽기 전용 사용자가 자신의 계정에 대한 Lambda 코드 스캔 상태 및 결과에 대한 세부 정보를 검색할 수 있도록 허용합니다.	2023년 5월 2일
새 기능	Amazon Inspector에는 Amazon Inspector가 특정 문제를 다루는지 확인할 수 있는 취약성 데이터베이스 검색 기능이 추가되었습니다. CVE	2023년 5월 1일
업데이트된 기능	Amazon Inspector는 Lambda 스캔을 활성화할 때 Amazon Inspector가 사용자 계정에 서비스 연결 채널을 AWS CloudTrail 생성할 수 있도록 허용하는 새로운 권한을 AmazonInspector2ServiceRolePolicy정책에 추가했습니다. 이렇게 하면 Amazon Inspector에서 사용자 계정의 CloudTrail 이벤트를 모니터링할 수 있습니다.	2023년 4월 30일
업데이트된 기능	Amazon Inspector에서 AmazonInspector2FullAccess 정책에 새 명령문을 추가했습니다. 이 명령문은 사용자가 Lambda 코드 스캔에서 코드 취약성 결과에 대한 세부 정보를 검색할 수 있도록 허용합니다.	2023년 4월 17일

업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 명령문을 추가했습니다. 새 명령문을 통해 Amazon Inspector는 아마존 EC2 심층 검사를 위해 정의한 사용자 지정 경로에 대한 정보를 Amazon EC2 Systems Manager에 보낼 수 있습니다.	2023년 4월 17일
새 기능	Amazon Inspector는 Amazon Inspector 심층 검사 형태로 Linux EC2 인스턴스에 대한 지원을 추가하여 애플리케이션 프로그래밍 언어 패키지의 패키지 취약성이 있는지 인스턴스를 검사합니다.	2023년 4월 17일
업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 명령문을 추가했습니다. 새 명령문을 통해 Amazon Inspector는 AWS Lambda 함수의 개발자 코드 스캔을 요청하고 Amazon Security로부터 스캔 데이터를 수신할 수 있습니다. CodeGuru 또한 Amazon Inspector는 정책을 IAM 검토할 수 있는 권한을 추가했습니다. Amazon Inspector는 이 정보를 사용하여 Lambda 함수의 코드 취약성을 스캔합니다.	2023년 2월 28일

새 기능	Amazon Inspector는 Lambda 코드 스캔이라는 Lambda 함수에 대한 추가 지원을 제공하는데, 이 기능은 Lambda 함수의 개발자 코드에서 보안 취약성을 스캔합니다.	2023년 2월 28일
업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 명령문을 추가했습니다. 새 명령문을 사용하면 Amazon Inspector에서 AWS Lambda 함수가 마지막으로 호출된 시기에 CloudWatch 대한 정보를 검색할 수 있습니다. 이 정보를 사용하여 사용자 환경에서 지난 90일 동안 활성 상태였던 Lambda 함수를 집중 검사합니다.	2023년 2월 20일
업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 명령문을 추가했습니다. 새 명령문은 Amazon Inspector에서 AWS Lambda 함수에 대한 정보를 검색할 수 있도록 허용합니다. Amazon Inspector는 이 정보를 사용하여 Lambda 함수의 보안 취약성을 스캔합니다.	2022년 11월 28일
새 기능	Amazon Inspector는 스캔 AWS Lambda 기능에 대한 지원을 추가합니다.	2022년 11월 28일

업데이트 내용	Amazon Inspector에서 Amazon Simple Storage Service(S3) 버킷으로 결과 보고서를 내보내는 절차, 정책 예제 및 팁이 추가되었습니다.	2022년 10월 14일
새 콘텐츠	Amazon Inspector 콘솔을 사용하여 AWS 사용자 환경의 Amazon Inspector 적용 범위를 평가하는 방법에 대한 정보가 추가되었습니다. 이 정보에는 환경의 개별 리소스에 대한 상태 값 설명이 포함됩니다.	2022년 10월 7일
새 기능	Amazon Inspector에서 이제 패키지 취약성을 해결하는 방법에 대한 추가 세부 정보를 제공합니다. 결과 세부 정보에 새로운 필드가 추가되었습니다. 새로운 필드는 패키지 업데이트를 통해 수정이 가능한지 여부에 대한 컨텍스트를 제공합니다. 수정이 가능한 경우 결과의 제안된 해결 방법 섹션에 수정을 실행할 수 있는 명령이 표시됩니다.	2022년 9월 2일

업데이트된 기능	Amazon Inspector에서 AmazonInspector2ServiceRolePolicy 정책에 새 작업을 추가했습니다. 새 작업을 통해 Amazon Inspector는 SSM 연결 실행을 설명할 수 있습니다. Amazon Inspector는 또한 Amazon Inspector에서 소유 문서를 생성, 업데이트, 삭제 및 연결할 수 있도록 추가 리소스 범위 지정 기능을 추가했습니다. SSM `AmazonInspector2` SSM	2022년 8월 31일
새 기능	Amazon Inspector에서 이제 Windows 인스턴스에 대한 스캔을 지원합니다. Amazon Inspector는 이제 지원되는 Windows 운영 체제를 실행하는 SSM 관리형 인스턴스를 스캔할 수 있습니다. Windows호스트 스캔은 Amazon Inspector 플러그인에 의해 수행되며, 이 플러그인은 Amazon SSM Inspector에서 자동으로 생성된 SSM 새로운 연결을 통해 설치 및 호출됩니다.	2022년 8월 31일
업데이트된 기능	Amazon Inspector는 Amazon Inspector가 다른 파티션의 소프트웨어 인벤토리를 수집할 수 있도록 AmazonInspector2ServiceRolePolicy정책의 리소스 범위를 업데이트했습니다. AWS	2022년 8월 12일

업데이트된 기능	AmazonInspector2ServiceRolePolicy정책에서 Amazon Inspector는 Amazon Inspector가 연결을 생성, 삭제 및 업데이트할 수 있도록 작업의 리소스 범위를 재구성했습니다. SSM	2022년 8월 10일
새 기능	Amazon Inspector는 이제 ECR 자동 재스캔 기간 설정 변경을 지원합니다. Amazon ECR 자동 재스캔 기간 설정에 따라 Amazon Inspector가 리포지토리로 푸시된 이미지를 지속적으로 모니터링하는 기간이 결정됩니다. 이미지가 스캔 기간보다 오래된 경우 Amazon Inspector에서 더 이상 이미지를 스캔하지 않고 이미지에 대한 기존 결과를 모두 종결합니다. 모든 새 계정의 자동 재스캔 기간은 수명으로 ECR 자동 설정됩니다. 이전에 만든 계정의 ECR 자동 재스캔 기간은 30일이었지만 이제는 30일, 180일 또는 평생 스캔 기간 중에서 선택할 수 있습니다.	2022년 6월 25일
새로운 기능	Amazon Inspector는 Amazon Inspector 기능에 대한 읽기 전용 액세스를 허용하는 새로운 AWS 관리형 AmazonInspector2ReadOnlyAccess정책인 정책을 추가했습니다.	2022년 1월 21일
정식 출시	이는 Amazon Inspector 사용 설명서의 최초 공개 릴리스입니다.	2021년 11월 29일

AWS 용어집

최신 AWS 용어는 참조의 AWS 용어집을 참조하십시오.AWS 용어집

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.