aws

사용자 가이드

Amazon Inspector

Amazon Inspector: 사용자 가이드

Table of Contents

Amazon Inspector란 무엇인가?

Amazon Inspector는 소프트웨어 취약성 및 의도하지 않은 네트워크 노출이 있는 AWS 워크로드를 지속적으로 스캔하는 취약성 관리 서비스입니다. Amazon Inspector는 실행 중인 Amazon EC2 인스턴스, Amazon Elastic Container Registry(Amazon ECR)의 컨테이너 이미지 및 AWS Lambda 함수를 자동으로 검색하고 스캔하여 알려진 소프트웨어 취약성 및 의도하지 않은 네트워크 노출이 있는지 확인합니다.

소프트웨어 취약성 또는 네트워크 구성 문제가 발견되면 Amazon Inspector에서 결과를 생성합니다. 결과는 취약성을 설명하고, 영향을 받는 리소스를 식별하며, 취약성의 심각도를 평가하고, 해결 지침을 제공합니다. Amazon Inspector 콘솔을 사용하여 결과를 분석하거나 다른 AWS 서비스를 통해 결과를 확인하고 처리할 수 있습니다. 자세한 내용은 Amazon Inspector의 결과에 대한 이해 섹션을 참조하세요.

주제

Amazon Inspector의 특징

여러 Amazon Inspector 계정을 중앙에서 관리

AWS 환경에 여러 개의 계정이 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 멤버 계정에 대해 집계된 결과 세부 정보 보기, 멤버 계정에 대한 스캔 활성화 또는 비활성화, AWS 조직 내의 스캔한 리소스 검토 등이 포함됩니다.

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 패키지를 설치하거나, 패치를 설치하거나, 리소스에 영향을 미치는 새로운 일반적인 취약성 및 노출(CVE)이 발표되는 경우 등 새로운 취약성을 유발할 수 있는 변경 사항에 대응하여 리소스를 자동

으로 재스캔함으로써 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 정보가 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 NVD(National Vulnerability Database) 기본 점수를 구성하는 보안 지표를 검사하여 컴퓨팅 환경에 따라 조정합니다. 예를 들어 네트워크를 통해 취약성이 악용될 소지가 있지만 인터넷으로 연결되는 오픈 네트워크 경로를 인스턴스에서 사용할 수 없는 경우 이 서비스가 Amazon EC2 인스턴스의 Amazon Inspector 결과 점수를 낮출 수 있습니다. 이 점수는 CVSS 형식이며 NVD에서 제공하는 기본 CVSS(Common Vulnerability Scoring System) 점수를 수정한 것입니다.

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 Amazon EventBridge에 결과 이벤트로 게시합니다. EventBridge는 결과 데이터를 AWS Lambda 함수 및 Amazon Simple

Notification Service(SNS) 주제 등의 대상으로 라우팅할 수 있는 서버리스 이벤트 버스 서비스입니다. EventBridge를 사용하면 기존 보안 및 규정 준수 워크플로우의 일부로 결과를 거의 실시간으로 모니터링하고 처리할 수 있습니다.

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

Amazon Inspector 액세스

Amazon Inspector는 대부분의 AWS 리전에서 사용할 수 있습니다. 현재 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 리전 관리를 참조하세요. 각 리전에서 다음과 같은 방법으로 Amazon Inspector를 사용할 수 있습니다.

AWS Management Console

AWS Management Console은 AWS 리소스를 생성하고 관리하는 데 사용할 수 있는 웹 기반 사용자 인터페이스입니다. 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 시스템 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.

AWS에서는 AWS Command Line Interface(AWS CLI) 및 AWS Tools for PowerShell라는 두 가지 명령줄 도구 세트를 제공합니다. AWS CLI 설치 및 사용에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서를 참조하세요. Tools for PowerShell 설치 및 사용에 대한 자세한 내용은 AWS Tools for PowerShell 사용 설명서를 참조하세요.

AWS SDK

AWS는 Java, Go, Python, C++, .NET을 비롯한 다양한 프로그래밍 언어 및 플랫폼용 라이브러리와 샘플 코드로 구성된 SDK를 제공합니다. SDK를 사용하면 Amazon Inspector 및 다른 AWS 서비스에 프로그래밍 방식으로 편리하게 액세스할 수 있습니다. SDK는 또한 요청에 암호화 방식으로 서명, 오류

관리, 요청 재시도 등의 작업을 자동으로 처리합니다. AWS SDK 사용에 대한 자세한 내용은 AWS에서의 구축을 위한 도구를 참조하세요.

Amazon Inspector REST API

Amazon Inspector REST API는 Amazon Inspector 계정 및 리소스에 대한 포괄적인 프로그래밍 방식의 액세스를 제공합니다. 이 API를 사용하면 HTTPS 요청을 Amazon Inspector로 직접 보낼 수 있습니다. 그러나 AWS 명령줄 도구 및 SDK와 달리 이 API를 사용하려면 애플리케이션에서 요청에 서명하기 위한 해시 생성과 같은 특정한 세부 정보를 처리해야 합니다.

Amazon Inspector 시작하기

이 자습서에서는 Amazon Inspector에 대한 실습 소개를 제공합니다.

1단계에서는 독립 실행형 계정에 대해 또는 다중 계정 환경에서 Amazon Inspector 위임 관리자로서 Amazon Inspector 스캔을 활성화하는 방법을 다룹니다. AWS Organizations

2단계에서는 콘솔에서 Amazon Inspector 결과를 이해하는 방법을 다룹니다.

> **Note**
>
> 이 자습서에서는 현재 상태에서 작업을 완료합니다. AWS 리전다른 리전에서 Amazon Inspector를 설정하려면 해당 리전마다 이러한 단계를 완료해야 합니다.

주제

시작하기 전 준비 사항

Amazon Inspector는 Amazon EC2 인스턴스, Amazon ECR 컨테이너 이미지 AWS Lambda 및 함수를 지속적으로 스캔하여 소프트웨어 취약성 및 의도하지 않은 네트워크 노출을 찾아내는 취약성 관리 서비스입니다.

Amazon Inspector를 활성화하기 전에 다음 사항을 참고하세요.

- Amazon Inspector는 지역 서비스이며, 데이터는 서비스를 사용하는 AWS 리전 곳에 저장됩니다. Amazon Inspector로 모니터링하려는 각 구성 절차에서 이 자습서에서 AWS 리전 완료하는 모든 구성 절차를 반복해야 합니다.
- Amazon Inspector는 Amazon EC2 인스턴스, Amazon ECR 컨테이너 이미지 및 함수 스캔을 활성화할 수 있는 유연성을 제공합니다. AWS Lambda 스캔 유형은 Amazon Inspector 콘솔의 계정 관리 페이지에서 또는 Amazon Inspector API를 사용하여 관리할 수 있습니다.
- Amazon Inspector는 Amazon EC2 Systems Manager(SSM) 에이전트가 설치되어 활성화된 경우에만 EC2 인스턴스에 대해 일반적인 취약성 및 노출(CVE) 데이터를 제공할 수 있습니다. 이 에이전트

는 많은 EC2 인스턴스에 사전 설치되어 있지만 수동으로 활성화해야 할 수도 있습니다. SSM 에이전트 상태에 관계없이 모든 EC2 인스턴스를 스캔하여 네트워크 노출 문제를 찾아냅니다. Amazon EC2 스캔 구성에 대한 자세한 내용은 Amazon EC2 인스턴스 스캔 섹션을 참조하세요. Amazon ECR 및 AWS Lambda 함수 스캔에는 에이전트를 사용할 필요가 없습니다.

- 관리자 권한이 있는 IAM 사용자 ID는 Amazon Inspector를 AWS 계정 활성화할 수 있습니다. 데이터 보호를 위해 자격 증명을 보호하고 AWS IAM Identity Center OR AWS Identity and Access Management (IAM) 를 사용하여 개별 사용자를 설정하는 것이 좋습니다. 이렇게 하면 각 사용자에게 Amazon Inspector를 관리하는 데 필요한 권한만 부여됩니다. Amazon Inspector를 활성화하는 데 필요한 권한에 대한 자세한 내용은 AWS 관리형 정책: AmazonInspector2FullAccess 섹션을 참조하세요.
- 어느 리전에서든 Amazon Inspector를 처음으로 활성화하면 `AWSServiceRoleForAmazonInspector2` 계정에 대해 전역적으로 서비스 연결 역할이 생성됩니다. 이 역할에는 Amazon Inspector를 통해 소프트웨어 패키지 세부 정보를 수집하고 Amazon VPC 구성을 분석하여 취약성 결과를 생성할 수 있는 권한 및 신뢰 정책이 포함됩니다. 자세한 설명은 Amazon Inspector에 서비스 연결 역할 사용 섹션을 참조하세요. 서비스 연결 역할에 대한 자세한 내용은 서비스 연결 역할 사용을 참조하세요.

1단계: Amazon Inspector 활성화

Amazon Inspector를 사용하기 위한 첫 번째 단계는 AWS 계정에서 활성화하는 것입니다. Amazon Inspector 스캔 유형을 활성화하면 Amazon Inspector에서 즉시 적합한 모든 리소스를 검색하고 스캔하기 시작합니다.

중앙 집중식 관리자 계정을 통해 조직 내 여러 계정에 대해 Amazon Inspector를 관리하려면 Amazon Inspector에 대한 위임 관리자를 할당해야 합니다. 사용 환경에 맞게 Amazon Inspector를 활성화하는 방법을 알아보려면 다음 옵션 중 하나를 선택합니다.

Standalone account environment

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 시작하기를 선택합니다.
3. Amazon Inspector 활성화를 선택합니다.

   독립 실행형 계정에서 Amazon Inspector를 활성화하면 기본적으로 모든 스캔 유형이 활성화됩니다. 활성화된 스캔 유형은 Amazon Inspector 콘솔의 계정 관리 페이지에서 관리하거나 Amazon Inspector API를 사용하여 관리할 수 있습니다. Amazon Inspector가 활성화되면 적합한 모든 리소

스가 자동으로 검색되고 스캔됩니다. 다음 스캔 유형 정보를 검토하여 기본적으로 적합한 리소스를 파악하세요.

Amazon EC2 스캔

EC2 인스턴스에 대한 일반적인 취약성 및 노출 (CVE) 데이터를 제공하려면 Amazon Inspector에서 Systems AWS Manager (SSM) 에이전트를 설치하고 활성화해야 합니다. 이 에이전트는 많은 EC2 인스턴스에 사전 설치되어 있지만 수동으로 활성화해야 할 수도 있습니다. SSM 에이전트 상태에 관계없이 모든 EC2 인스턴스를 스캔하여 네트워크 노출 문제를 찾아냅니다. Amazon EC2 스캔 구성에 대한 자세한 내용은 Amazon Inspector로 Amazon EC2 인스턴스 스캔 섹션을 참조하세요.

Amazon ECR 스캔

Amazon ECR 스캔을 활성화하면 Amazon Inspector는 Amazon ECR에서 제공하는 기본 스캔을 수행하도록 구성된 프라이빗 레지스트리 내의 모든 컨테이너 리포지토리를 연속 스캔 기능을 갖춘 고급 스캔으로 변환합니다. 푸시할 때만 스캔하거나 포함 규칙을 통해 일부 리포지토리를 스캔하도록 이 설정을 선택적으로 구성할 수도 있습니다. 지난 30일 이내에 푸시된 모든 이미지는 전체 기간 스캔으로 예약되며, 이 Amazon ECR 스캔 설정은 언제든지 변경할 수 있습니다. Amazon ECR 스캔 구성에 대한 자세한 내용은 Amazon Inspector로 Amazon ECR 컨테이너 이미지 스캔 섹션을 참조하세요.

AWS Lambda 함수 스캔

AWS Lambda 함수 스캔을 활성화하면 Amazon Inspector는 계정에서 Lambda 함수를 발견하고 즉시 취약성 검사를 시작합니다. Amazon Inspector는 새로운 Lambda 함수와 계층이 배포될 때 스캔을 수행하며, 해당 함수와 계층이 업데이트되거나 새로운 일반 취약성 및 노출(CVE)이 발표될 때 재스캔을 수행합니다. Amazon Inspector는 두 가지 수준의 Lambda 함수 스캔을 제공합니다. Amazon Inspector를 처음 활성화하면 기본적으로 함수의 패키지 종속성을 스캔하는 Lambda 표준 스캔이 활성화됩니다. 또한 Lambda 코드 스캔을 활성화하여 함수의 개발자 코드에서 코드 취약성을 검사할 수도 있습니다. Lambda 함수 스캔 구성에 대한 자세한 내용은 Amazon AWS Lambda Inspector를 사용한 스캔 기능 섹션을 참조하세요.

Multi-account environment

> **Important**
>
> 이 단계를 완료하려면 관리하려는 모든 계정과 동일한 조직에 속해 있어야 하며, 조직 내 Amazon Inspector 관리자를 위임할 수 있도록 AWS Organizations 관리 계정에 대한 액세

스 권한이 있어야 합니다. 관리자를 위임하려면 추가 권한이 필요할 수 있습니다. 자세한 설명은 위임 관리자를 지정하는 데 필요한 권한 섹션을 참조하세요.

> **Note**
>
> 여러 리전의 여러 계정에 대해 Amazon Inspector를 프로그래밍 방식으로 활성화하려면 Amazon Inspector에서 개발한 쉘 스크립트를 사용할 수 있습니다. 이 스크립트 사용에 대한 자세한 내용은 inspector2-on을 참조하십시오. enablement-with-cli GitHub

Amazon Inspector 관리자 위임

1. 관리 계정에 로그인합니다. AWS Organizations
2. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
3. 위임된 관리자 창에서 조직의 Amazon Inspector 위임 관리자로 지정하려는 12자리 ID를 입력합니다. AWS 계정 그런 다음 삭제를 선택합니다. 그런 다음 확인 창에서 위임을 선택합니다.

> **Note**
>
> 관리자를 위임하면 계정에 대해 Amazon Inspector가 활성화됩니다.

멤버 계정 추가

위임 관리자는 Organizations 관리 계정과 연결된 모든 멤버에 대한 스캔을 활성화할 수 있습니다. 이 워크플로우는 모든 멤버 계정의 모든 스캔 유형을 활성화합니다. 그러나 멤버가 본인 계정에 대해 Amazon Inspector를 활성화하거나, 위임 관리자가 선택적으로 서비스 스캔을 활성화할 수 있습니다. 자세한 설명은 다중 계정 관리 섹션을 참조하세요.

1. 위임 관리자 계정에 로그인합니다.
2. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
3. 탐색 창에서 계정 관리를 선택합니다. 계정 테이블에 Organizations 관리 계정과 관련된 모든 멤버 계정이 표시됩니다.
4. 계정 관리 페이지에서 상단 배너의 모든 계정에 대한 스캔 활성화를 선택하여 EC2 인스턴스, ECR 컨테이너 이미지 및 조직 내 모든 계정에 대한 AWS Lambda 함수 스캔을 활성화할 수 있

습니다. 또는 계정 테이블에서 멤버로 추가할 계정을 선택할 수도 있습니다. 그런 다음 활성화 메뉴에서 모든 스캔을 선택합니다.

5. (선택 사항) 새 멤버 계정에 대해 Inspector 자동 활성화 기능을 켜고 포함할 스캔 유형을 선택하여 조직에 추가된 새 멤버 계정에 대해 해당 스캔을 활성화합니다.

Amazon Inspector는 현재 EC2 인스턴스, ECR 컨테이너 이미지 및 함수에 대한 스캔을 제공합니다. AWS Lambda Amazon Inspector를 활성화하면 적합한 모든 리소스가 자동으로 검색되고 스캔됩니다. 다음 스캔 유형 정보를 검토하여 기본적으로 적합한 리소스를 파악하세요.

Amazon EC2 스캔

EC2 인스턴스에 대한 CVE 취약성 데이터를 제공하려면 Amazon Inspector에서 AWS Systems Manager (SSM) 에이전트를 설치하고 활성화해야 합니다. 이 에이전트는 많은 EC2 인스턴스에 사전 설치되어 있지만 수동으로 활성화해야 할 수도 있습니다. SSM 에이전트 상태에 관계없이 모든 EC2 인스턴스를 스캔하여 네트워크 노출 문제를 찾아냅니다. Amazon EC2 스캔 구성에 대한 자세한 내용은 Amazon Inspector로 Amazon EC2 인스턴스 스캔 섹션을 참조하세요.

Amazon ECR 스캔

Amazon ECR 스캔을 활성화하면 Amazon Inspector는 Amazon ECR에서 제공하는 기본 스캔을 수행하도록 구성된 프라이빗 레지스트리 내의 모든 컨테이너 리포지토리를 연속 스캔 기능을 갖춘 고급 스캔으로 변환합니다. 푸시할 때만 스캔하거나 포함 규칙을 통해 일부 리포지토리를 스캔하도록 이 설정을 선택적으로 구성할 수도 있습니다. 지난 30일 이내에 푸시된 모든 이미지는 전체 기간 스캔으로 예약됩니다. 이 Amazon ECR 스캔 설정은 위임 관리자가 언제든지 변경할 수 있습니다. Amazon ECR 스캔 구성에 대한 자세한 내용은 Amazon Inspector로 Amazon ECR 컨테이너 이미지 스캔 섹션을 참조하세요.

AWS Lambda 함수 스캔

AWS Lambda 함수 스캔을 활성화하면 Amazon Inspector는 계정에서 Lambda 함수를 발견하고 즉시 취약성 검사를 시작합니다. Amazon Inspector는 새로운 Lambda 함수와 계층이 배포될 때 스캔을 수행하며, 해당 함수와 계층이 업데이트되거나 새로운 일반 취약성 및 노출(CVE)이 발표될 때 재스캔을 수행합니다. Lambda 함수 스캔 구성에 대한 자세한 내용은 Amazon AWS Lambda Inspector를 사용한 스캔 기능 섹션을 참조하세요.

2단계: Amazon Inspector 결과 보기

사용 환경에 대한 결과는 Amazon Inspector 콘솔 또는 API를 통해 확인할 수 있습니다. 모든 조사 결과는 Amazon EventBridge 및 AWS Security Hub (활성화된 경우) 에도 푸시됩니다. 또한 컨테이너 이미지 결과는 Amazon ECR로 푸시됩니다.

Amazon Inspector 콘솔은 결과에 대해 여러 가지 보기 형식을 제공합니다. Amazon Inspector 대시보드에서는 환경에 미치는 위험에 대한 전반적인 개요를 확인할 수 있고, 결과 테이블에서는 특정 결과에 대한 세부 정보를 확인할 수 있습니다.

이 단계에서는 결과 테이블과 결과 대시보드를 사용하여 결과에 대한 세부 정보를 살펴봅니다. Amazon Inspector 대시보드에 대한 자세한 내용은 대시보드 이해 섹션을 참조하세요.

Amazon Inspector 콘솔에서 환경에 대한 조사 결과의 세부 정보를 보려면:

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 탐색 창에서 대시보드를 선택합니다. 대시보드의 링크 중 하나를 선택하면 Amazon Inspector 콘솔에서 해당 항목에 대한 자세한 정보가 있는 페이지로 이동할 수 있습니다.
3. 탐색 창에서 결과를 선택합니다.
4. 기본적으로 모든 결과 탭이 표시됩니다. 이 탭에는 모든 EC2 인스턴스, ECR 컨테이너 이미지, 사용자 환경에 대한 AWS Lambda 함수 검색 결과가 표시됩니다.
5. 결과 목록의 제목 열에서 결과 이름을 선택하여 해당 결과에 대한 세부 정보 창을 엽니다. 모든 결과에는 결과 세부 정보 탭이 있습니다. 다음과 같은 방법으로 결과 세부 정보 탭과 상호 작용할 수 있습니다.

   - 취약성에 대한 자세한 내용을 보려면 취약성 세부 정보 섹션의 링크를 따라 이동하여 해당 취약성에 대한 설명서를 엽니다.
   - 리소스를 더 자세히 조사하려면 영향을 받는 리소스 섹션의 리소스 ID 링크를 따라 이동하여 영향을 받는 리소스의 서비스 콘솔을 엽니다.

   또한 패키지 취약성 유형 결과에는 Inspector 점수 및 취약성 인텔리전스 탭이 있으며, 이 탭에서는 해당 결과에 대한 Amazon Inspector 점수 계산 방법을 설명하고 해당 결과와 관련된 일반적인 취약성 및 악용(CVE)에 대한 정보를 제공합니다. 결과 유형에 대한 자세한 내용은 Amazon Inspector의 결과 유형 섹션을 참조하세요.

Amazon Inspector 대시보드 이해

Amazon Inspector 대시보드는 현재 AWS 리전의 AWS 리소스에 대해 집계된 통계의 스냅샷을 제공합니다. 이러한 통계에는 리소스 적용 범위와 활성 취약성에 대한 주요 지표가 포함됩니다. 또한 대시보드에는 가장 중요한 결과가 있는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스, Amazon Elastic Container Registry(Amazon ECR), 및 AWS Lambda 함수 등 계정에 대해 집계된 결과 데이터 그룹도 표시됩니다. 대시보드 항목에 대한 지원 데이터를 확인하면 더 심층적인 분석을 수행할 수 있습니다.

조직의 Amazon Inspector 위임 관리자 계정인 경우 대시보드에는 자신의 계정을 포함하여 조직 내 모든 계정에 대한 계정 적용 범위, 집계된 통계 및 결과 데이터가 포함됩니다.

대시보드 표시

대시보드에는 환경 적용 범위와 중요 결과에 대한 개요가 표시됩니다.

대시보드를 표시하려면:

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 탐색 창에서 대시보드를 선택합니다.
3. 다음과 같은 방법으로 대시보드와 상호 작용할 수 있습니다.

   - 대시보드는 5분마다 자동으로 새로 고쳐집니다. 하지만 페이지 오른쪽 상단에 있는 새로 고침 아이콘을 선택하여 데이터를 수동으로 새로 고칠 수 있습니다.
   - 대시보드에서 항목의 지원 데이터를 보려면 해당 항목을 선택하십시오.
   - Amazon Inspector 위임 관리자로서 AWS Organizations를 통해 여러 계정을 관리하는 경우 대시보드에는 멤버 계정에 대해 집계된 통계가 표시됩니다. 대시보드를 필터링하고 특정 계정에 대한 데이터만 표시하려면 계정 상자에 계정 ID를 입력합니다.

대시보드 구성 요소 이해 및 데이터 해석

Amazon Inspector 대시보드의 각 섹션은 현재 AWS에서 AWS 리전 리소스의 취약성 상태를 이해하는 데 도움이 되는 주요 지표 또는 활성 결과 데이터를 제공합니다.

환경 적용 범위

환경 적용 범위 섹션은 Amazon Inspector에서 스캔한 리소스에 대한 통계를 제공합니다. 이 섹션에서는 Amazon Inspector에서 스캔한 Amazon EC2 인스턴스, Amazon ECR 이미지 및 AWS Lambda 함수의 개수와 비율을 확인할 수 있습니다. Amazon Inspector 위임 관리자로서 AWS Organizations를 통해 여러 계정을 관리하는 경우 총 조직 계정 수, Amazon Inspector가 활성화된 횟수 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 환경의 Amazon Inspector 적용 범위 평가 섹션을 참조하세요.

적용 범위 그룹을 선택하면 선택한 그룹에 대한 계정 관리 페이지로 이동합니다. 계정 관리 페이지에는 Amazon Inspector가 적용되는 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 Inspector의 결과에 대한 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.

- ECR 컨테이너 이미지 결과
- Amazon EC2 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 결과가 있는 계정 섹션에는 사용 환경에서 가장 중요한 결과가 있는 상위 5개 AWS 계정과 해당 계정에 대한 총 결과 수가 표시됩니다. AWS Organizations에서 다중 계정 스캔을 수행하도록 Amazon Inspector가 구성된 경우 이 섹션은 위임 관리자 계정에서만 볼 수 있습니다. 이 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 결과가 있는 Amazon ECR 리포지토리

가장 중요한 결과가 있는 Elastic Container Registry(ECR) 리포지토리 섹션에는 사용 환경에서 가장 중요한 컨테이너 이미지 결과가 있는 상위 5개 Amazon ECR 리포지토리가 표시됩니다. 이 보기에는 리포지토리 이름, AWS 계정 ID, 리포지토리 생성 날짜, 중요 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. 이 보기에는 이미지 태그 데이터, 리포지토리 이름, 이미지 다이제스트, AWS 계정 ID, 중요 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 결과가 있는 인스턴스 섹션에는 가장 중요한 결과가 있는 상위 5개 Amazon EC2 인스턴스가 표시됩니다. 이 보기에는 인스턴스 식별자, AWS 계정 ID, Amazon Machine Image(AMI) 식별자, 중요 취약성 수, 총 취약성 수가 표시됩니다. 이 보기는 인프라 소유자가 패치 적용이 필요한 인스턴스를 파악하는 데 도움이 됩니다.

인스턴스 ID를 선택하면 영향을 받는 Amazon EC2 인스턴스에 대한 자세한 정보를 볼 수 있습니다.

가장 중요한 결과가 있는 Amazon Machine Image(AMI)

가장 중요한 결과가 있는 Amazon Machine Image(AMI) 섹션에는 사용 환경에서 가장 중요한 결과가 있는 상위 5개 AMI가 표시됩니다. 이 보기에는 AMI 식별자, AWS 계정 ID, 환경에서 실행 중인 영향을 받는 EC2 인스턴스 수, AMI 생성 날짜, AMI의 운영 체제 플랫폼, 중요 취약성 수, 총 취약성 수가 표시됩니다. 이 보기는 인프라 소유자가 재구축이 필요한 AMI를 식별하는 데 도움이 됩니다.

영향을 받는 인스턴스를 선택하면 영향을 받는 AMI에서 실행된 인스턴스에 대한 자세한 정보를 볼 수 있습니다.

가장 중요한 결과가 있는 AWS Lambda 함수

가장 중요한 결과가 있는 AWS Lambda 함수 섹션에는 사용 환경에서 가장 중요한 결과가 있는 상위 5개 Lambda 함수가 표시됩니다. 이 보기에는 Lambda 함수 이름, AWS 계정 ID, 런타임 환경, 중요 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 AWS Lambda 함수에 대한 자세한 정보를 볼 수 있습니다.

Amazon Inspector의 결과에 대한 이해

발견은 AWS 리소스 중 하나에 영향을 미치는 취약성에 대한 자세한 보고서입니다. 발견한 취약점의 이름을 따서 조사 결과를 명명하고 심각도 등급, 영향을 받는 리소스에 대한 정보, 보고된 취약성을 해결하는 방법을 설명하는 세부 정보를 제공합니다.

Amazon Inspector는 Amazon EC2 인스턴스, Amazon ECR 리포지토리의 컨테이너 이미지 또는 함수에서 취약성을 탐지할 때마다 탐지 결과를 생성합니다. AWS Lambda Amazon Inspector는 컴퓨팅 환경을 지속적으로 스캔하고 수정하기 전까지 모든 활성 결과를 저장합니다.

검색 결과를 수정하면 검색 결과가 자동으로 닫히고 Amazon Inspector는 7일 후에 검색 결과를 삭제합니다. 리소스를 삭제하면 Amazon Inspector는 30일 후에 해당 리소스와 관련된 모든 검색 결과를 삭제합니다.

Amazon Inspector를 비활성화하면 24시간 후에 조사 결과가 제거됩니다. 계정이 AWS 일시 중지되면 90일 후에 조사 결과가 제거됩니다.

조사 결과는 다음 상태 중 하나로 분류됩니다.

활성

Amazon Inspector는 수정되지 않은 결과를 활성 상태로 식별합니다.

표시되지 않음

Amazon Inspector는 하나 이상의 금지 규칙이 적용되는 결과를 차단된 것으로 식별합니다. 숨겨진 검색 결과 목록에서 숨겨진 검색 결과를 찾을 수 있습니다. 자세한 설명은 억제 규칙을 사용하여 Amazon Inspector 결과를 숨기는 방법 섹션을 참조하세요.

종결됨

취약성을 수정한 후 Amazon Inspector는 이를 자동으로 탐지하고 탐지 결과를 종료됨으로 변경합니다. 종료된 조사 결과는 7일 후에 삭제됩니다.

주제

- Amazon Inspector 결과의 심각도 수준

Amazon Inspector의 결과 유형

Amazon Inspector는 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스, Amazon Elastic Container Registry(Amazon ECR) 리포지토리의 컨테이너 이미지 및 AWS Lambda 함수에 대한 결과를 생성합니다. Amazon Inspector는 다음과 같은 유형의 결과를 생성할 수 있습니다.

패키지 취약성

패키지 취약성 결과는 사용자 AWS 환경에서 일반적인 취약성 및 노출(CVE)에 표시된 소프트웨어 패키지를 식별합니다. 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 노출도에 대한 참조 방법입니다. 자세한 내용은 https://www.cve.org/를 참조하세요.

Linux용 CVE 탐지는 공급업체 보안 권고에 따라 발표된 후 24시간 이내에 Amazon Inspector에 추가됩니다. Windows용 CVE 탐지는 Microsoft에서 발표한 후 48시간 이내에 Amazon Inspector에 추가됩니다. 아마존 인스펙터 취약성 데이터베이스 검색을 사용하여 CVE 탐지가 지원되는지 확인할 수 있습니다.

Amazon Inspector는 EC2 인스턴스, ECR 컨테이너 이미지 및 Lambda 함수에 대한 패키지 취약성 탐지 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. 또한 Amazon CodeGuru와 공동으로 개발한 내부 탐지기를 기반으로 정책 위반 및 취약성을 식별합니다. 가능한 탐지 목록은 CodeGuru 탐지기 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.

코드 취약성과 관련하여 발견된 코드 스니펫은 CodeGuru 서비스에 저장됩니다. 기본적으로 CodeGuru에서 제어하는 AWS 소유 키가 코드를 암호화하는 데 사용되지만, Amazon Inspector API를 통해 고객이 관리하는 자체 키를 암호화에 사용할 수도 있습니다. 자세한 내용은 결과 코드에 대한 저장 중 암호화 섹션을 참조하세요.

네트워크 연결성

네트워크 연결성 결과는 사용자 환경에 Amazon EC2 인스턴스에 대한 오픈 네트워크 경로가 있음을 나타냅니다. 이러한 결과는 인터넷 게이트웨이(Application Load Balancer 또는 Classic Load Balancer 뒤에 있는 인스턴스 포함), VPC 피어링 연결 또는 가상 게이트웨이를 통한 VPN 등의 VPC 엣지에서 TCP 및 UDP 포트에 연결할 수 있는 경우에 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 연결성 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

- 게이트웨이 VPC 엔드포인트
- VPC 피어링 연결
- VPN 연결

Amazon Inspector 조사 결과 찾기 및 보기

이 섹션의 절차에서는 Amazon Inspector 콘솔 및 API를 통해 Amazon Inspector에서 결과를 찾고 확인하는 방법을 설명합니다. 검색 세부 정보는 검색 결과 유형, 취약성 유형, 영향을 받는 리소스에 따라 달라집니다. 자세한 설명은 Amazon Inspector 결과 세부 정보 섹션을 참조하세요.

Console

콘솔에서 조사 결과를 확인하는 방법

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 탐색 창에서 [검색 결과] 를 선택합니다. 모든 결과를 볼 수 있는 결과 화면으로 이동합니다. 검색 결과 테이블에서 제목 열 아래에 있는 검색 결과 이름을 선택하여 검색 결과를 선택할 수 있습니다.
3. (선택 사항) 범주별로 그룹화된 검색 결과를 볼 수도 있습니다. 탐색 창에서 [검색 결과] 를 선택하고 다음 범주 중 하나를 선택합니다.
   - 취약성별
   - 인스턴스별

     > **Note**
     > 인스턴스별로 그룹화된 결과에는 네트워크 가용성에 대한 정보가 포함되지 않습니다.

   - 컨테이너 이미지별
   - 컨테이너 리포지토리별
   - Lambda 함수별

API

ListFindings API 작업을 실행합니다. 요청에서 특정 결과를 반환하도록 `filterCriteria`를 지정할 수 있습니다.

Amazon Inspector 결과 세부 정보

Amazon Inspector 콘솔에서는 결과 각각에 대한 세부 정보를 확인할 수 있습니다. 결과 세부 정보는 결과 유형에 따라 달라집니다.

결과 세부 정보를 확인하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 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 설명은 Amazon Inspector 결과 필터링 섹션을 참조하세요.
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 결과의 심각도 수준 섹션을 참조하세요.

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

모든 검색 결과에는 검색 결과가 식별된 AWS 계정 ID 번호, 심각도, 검색 결과 유형, 검색 결과 생성 날짜, 해당 리소스에 대한 세부 정보가 포함된 영향을 받는 리소스 섹션이 포함됩니다.

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

패키지 취약성

패키지 취약성 결과는 EC2 인스턴스, ECR 컨테이너 이미지 및 Lambda 함수에 제공됩니다. 자세한 내용은 패키지 취약성 섹션을 참조하세요.

패키지 취약성 결과에는 Amazon Inspector 점수 및 취약성 인텔리전스도 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 - 조사 결과와 관련된 EBS 볼륨 ID 및 파티션 번호입니다. 이 필드는 에이전트 없는 스캔을 사용하여 스캔한 EC2 인스턴스의 조사 결과에 있습니다.
- 설치된 버전/수정된 버전 - 현재 설치된 패키지 중에서 취약성이 탐지된 버전 번호입니다. 설치된 버전 번호를 슬래시(/) 뒤의 값과 비교하세요. 두 번째 값은 탐지된 취약성을 수정한 패키지의 버전 번호로, 결과와 관련된 일반적인 취약성 및 노출(CVE) 또는 권고에 따라 제공됩니다. 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 - 결과에서 식별된 CVE와 관련하여 NVD(National Vulnerability Database), REDHAT 또는 다른 OS 공급업체 등 Amazon Inspector 선호 소스로 연결되는 링크를 제공합니다. 또한 결과에 대한 심각도 점수도 확인할 수 있습니다. 심각도 점수에 대한 자세한 내용은 Amazon Inspector 결과의 심각도 수준 섹션을 참조하세요. 점수 벡터를 포함하여 다음과 같은 점수가 포함됩니다.
  - EPSS 점수

- Inspector 점수
- Amazon CVE의 CVSS 3.1
- NVD의 CVSS 3.1
- NVD의 CVSS 2.0(해당하는 경우, 이전 CVE의 경우)

- 관련 취약성 - 결과와 관련된 다른 취약성을 지정합니다. 일반적으로 이들은 동일한 패키지 버전에 영향을 미치는 다른 CVE이거나 공급업체에서 결정한 결과 CVE와 동일한 그룹 내의 다른 CVE입니다.

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 CodeGuru 탐지하는 데 사용되는 탐지기의 이름입니다. 가능한 탐지 목록은 CodeGuru 탐지기 라이브러리를 참조하십시오.
- 검출기 태그 - 검출기와 관련된 CodeGuru 태그는 태그를 CodeGuru 사용하여 탐지를 분류합니다.
- 관련 CWE - 코드 취약성과 관련된 CWE(Common Weakness Enumeration) ID입니다.
- 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 인스턴스에만 사용할 수 있습니다. 자세한 내용은 네트워크 연결성 섹션을 참조하세요. 이 결과 유형에 대한 세부 정보는 다음과 같습니다.

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

Amazon Inspector 점수 및 취약성 인텔리전스

Amazon Inspector 콘솔에서 조사 결과를 선택하면 Inspector 점수 및 취약성 인텔리전스 탭에 패키지 취약성 조사 결과에 대한 점수 세부 정보와 취약성 인텔리전스 세부 정보가 표시됩니다. 이러한 세부 정보는 패키지 취약성 결과에만 제공됩니다.

Amazon Inspector 점수

Amazon Inspector 점수는 Amazon Inspector가 각 EC2 인스턴스 결과에 대해 생성하는 컨텍스트화된 점수입니다. Amazon Inspector 점수는 기본 CVSS v3.1 점수 정보를 스캔 중에 컴퓨팅 환경에서 수집한 정보(예: 네트워크 연결성 결과 및 악용 가능성 데이터)와 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 제공하는 CVSS v3.1 기본 점수입니다. RHEL, Debian 또는 Amazon 공급업체의 기본 점수가 지원되며, 다른 공급업체 또는 공급업체가 점수를 제공하지 않는 경우 Amazon Inspector는 National Vulnerability Database(NVD)의 기본 점수를 사용합니다. Amazon Inspector는 Common Vulnerability Scoring System Version 3.1 Calculator를 사용하여 점수를 계산합니다. 개별 조사 결과의 기본 점수 소스는 취약성 세부 정보 아래의 조사 결과 세부 정보에서 취약성 소스(또는 조사 결과 JSON의 packageVulnerabilityDetails.source)로 표시됩니다.

> **Note**
>
> Ubuntu를 실행하는 Linux 인스턴스에는 Amazon Inspector 점수를 사용할 수 없습니다. Ubuntu의 경우 관련 CVE 심각도와 다를 수 있는 자체 취약성 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 섹션에서는 CVSS 기본 점수 지표와 Inspector 점수를 비교한 표를 볼 수 있습니다. 비교된 지표는 first.org에서 관리하는 CVSS 사양 문서에 정의된 기본 지표입니다. 다음은 기본 지표를 요약한 것입니다.

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 Amazon을 비롯하여 Recorded Future, 사이버 보안 및 인프라 보안국(CISA) 등의 업계 표준 보안 인텔리전스 소스에서 제공하는 CVE 관련 인텔리전스를 요약하여 설명합니다.

> **Note**
>
> CISA, Amazon 또는 Recorded Future에서 제공하는 인텔리전스를 모든 CVE에 사용할 수 있는 것은 아닙니다.

취약성 인텔리전스 세부 정보는 콘솔에서 또는 BatchGetFindingDetails API를 사용하여 볼 수 있습니다. 콘솔에서 다음 세부 정보를 확인할 수 있습니다.

ATT&CK

이 섹션에서는 CVE와 관련된 MITRE 전술, 기법 및 절차(TTP)를 설명합니다. 관련 TTP가 표시되며, 해당하는 TTP가 두 개 이상인 경우 링크를 선택하여 전체 목록을 볼 수 있습니다. 전술 또는 기법을 선택하면 MITRE 웹 사이트에서 해당 전술 또는 기법에 대한 정보가 열립니다.

CISA

이 섹션에서는 취약성과 연관된 관련 날짜를 다룹니다. 사이버 보안 및 인프라 보안국(CISA)에서 적극적인 악용의 증거를 바탕으로, 알려진 악용 취약성 카탈로그에 취약성을 추가한 날짜와 CISA에서 시스템에 패치를 적용할 것으로 예상하는 기한입니다. 이 정보는 CISA에서 제공합니다.

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 결과의 심각도 수준

Amazon Inspector에서 취약성 결과를 생성하면 결과에 심각도가 자동으로 할당됩니다. 결과의 심각도는 결과의 주요 특성을 반영하고 있으므로 결과를 평가하고 우선 순위를 정하는 데 유용합니다. 결과의

심각도가 영향을 받는 리소스가 조직에 미칠 수 있는 중요도나 중요성을 암시하거나 나타내는 것은 아닙니다.

결과의 심각도 등급은 정보, 낮음, 중간, 높음, 중요 수준 중 하나에 해당하는 수치 점수를 기준으로 결정됩니다.

Amazon Inspector에서 심각도를 결정하는 방법은 결과 유형에 따라 다릅니다. Amazon Inspector에서 각 결과 유형의 심각도 등급을 결정하는 방법에 대해 자세히 알아보려면 다음 섹션을 참조하세요.

소프트웨어 패키지 취약성 심각도

Amazon Inspector에서는 소프트웨어 패키지 취약성에 대한 심각도 점수의 기준으로 NVD/CVSS 점수가 사용됩니다. NVD/CVSS 점수는 NVD에서 발표하고 CVSS에서 정의한 취약성 심각도 점수로, 공격 복잡도, 익스플로잇 코드 성숙도, 필요한 권한 등의 보안 지표로 구성됩니다. 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 다음과 같은 심각도 등급에 매핑합니다.

점수	등급
0	Informational
0.1–3.9	Low
4.0–6.9	Medium
7.0–8.9	High
9.0–10.0	Critical

패키지 취약성 결과의 심각도로 분류되지 않음이 할당될 수도 있습니다. 이는 공급업체가 탐지된 취약성에 대해 취약성 점수를 아직 설정하지 않은 경우입니다. 이 경우 결과의 참조 URL을 사용하여 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

- Amazon CVE의 CVSS 3.1
- NVD의 CVSS 3.1
- NVD의 CVSS 2.0(해당하는 경우)

코드 취약성 심각도

코드 취약성 발견의 경우 Amazon Inspector는 탐지 결과를 생성한 CodeGuru Amazon 탐지기에서 정의한 심각도 수준을 사용합니다. CVSS v3 채점 시스템을 사용하여 각 탐지기에 심각도가 할당됩니다. 심각도 CodeGuru 사용에 대한 설명은 가이드의 심각도 정의를 참조하십시오. CodeGuru 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. Open path 등급 열의 값은 가상 게이트웨이, 피어링된 VPC 및 네트워크에서 열린 경로를 나타냅니다. AWS Direct Connect 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	Medium	Informational
Elasticsearch	9300, 9200	NA	Medium	Informational
FTP	21	21	High	Medium
Global catalog LDAP	3268	NA	Medium	Informational
Global catalog LDAP over TLS	3269	NA	Medium	Informational
HTTP	80	80	Low	Informational
HTTPS	443	443	Low	Informational

Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752	Medium	Informational
LDAP	389	389	Medium	Informational
LDAP over TLS	636	NA	Medium	Informational
MongoDB	27017, 27018, 27019, 28017	NA	Medium	Informational
MySQL	3306	NA	Medium	Informational
NetBIOS	137, 139	137, 138	Medium	Informational
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110	Medium	Informational
Oracle	1521, 1630	NA	Medium	Informational
PostgreSQL	5432	NA	Medium	Informational
Print services	515	NA	High	Medium
RDP	3389	3389	Medium	Low
RPC	111, 135, 530	111, 135, 530	Medium	Informational
SMB	445	445	Medium	Informational
SSH	22	22	Medium	Low
SQL Server	1433	1434	Medium	Informational
Syslog	601	514	Medium	Informational
Telnet	23	23	High	Medium
WINS	1512, 42	1512, 42	Medium	Informational

Amazon Inspector에서 결과 관리

Amazon Inspector는 결과를 정렬, 그룹화 및 관리하는 여러 가지 방법을 제공합니다. 이러한 기능을 통해 검색 결과를 환경에 맞게 조정하고, 다양한 관점에서 결과를 집계하고, 특정 환경의 취약성에 초점을 맞출 수 있습니다. AWS

결과는 상태에 따라 활성, 표시되지 않음, 종결됨 등 다양한 보기에 표시됩니다. 기본적으로 각 보기에는 활성 결과만 표시됩니다. 활성 결과는 Amazon Inspector에서 탐지한 잠재적 보안 문제로, 취약성 또는 잠재적 위협을 나타냅니다. 표시되지 않은 결과는 억제 규칙을 사용하여 제외한 활성 결과입니다. Amazon Inspector에서 결과가 해결된 것으로 탐지되면 결과 상태가 자동으로 종결됨으로 설정됩니다. 결과를 수동으로 종결할 수는 없습니다.

환경 전반의 보안 상태를 포괄적으로 보여주는 AWS Security Hub서비스인 에서도 결과를 확인할 수 있습니다. AWS 자세한 설명은 Amazon Inspector와 AWS Security Hub 통합 섹션을 참조하세요. 컨테이너 이미지 검색 결과는 Amazon ECR 콘솔에서도 사용할 수 있으며, AWS Command Line Interface (AWS CLI) 또는 API를 사용하여 모든 리소스의 검색 결과를 볼 수 있습니다.

주제

Amazon Inspector 결과 보기

Amazon Inspector 콘솔은 관련 그룹을 기준으로 탭 보기에 결과를 표시합니다. 각 보기에는 특정 취약성을 분석하고, 가장 취약한 리소스를 식별하고, 취약성이 환경에 미치는 전반적인 영향을 측정하는 데 도움이 되는 정보가 포함되어 있습니다. 결과 탐색 사이드 패널에서 옵션을 선택하여 다른 결과 보기로 이동할 수 있습니다. 또한 각 보기에 필터를 만들어 특정 유형의 결과에 초점을 맞출 수 있습니다. 필터 사용에 대한 자세한 내용은 Amazon Inspector 결과 필터링 섹션을 참조하세요.

다음 매개변수별로 결과를 그룹화할 수 있습니다.

- 취약성별 - 사용자 환경에서 발견된 가장 심각한 취약성이 나열됩니다. 이 보기에서 취약성 제목을 선택하면 추가 정보가 포함된 세부 정보 창이 열립니다.

- 계정별 - 사용자 계정, 계정별 Amazon Inspector 스캔 적용 범위 비율, 계정별 위험 및 높음 심각도 결과의 총 개수가 나열됩니다. 이 그룹은 위임된 관리자만 사용할 수 있습니다.
- 인스턴스별 - 사용자 환경에서 가장 취약한 Amazon EC2 인스턴스가 나열됩니다.
- 컨테이너 이미지별 - 사용자 환경에서 가장 취약한 Amazon ECR 컨테이너 이미지가 나열됩니다.
- 컨테이너 리포지토리별 - 취약성이 가장 많은 리포지토리가 표시됩니다.
- Lambda 함수별 - 취약성이 가장 많은 Lambda 함수가 표시됩니다.
- 모든 결과 - 사용자 환경에 대한 전체 결과 목록이 표시됩니다. 결과 페이지로 이동할 때 기본 보기입니다. 이 보기에서는 활성 결과, 표시되지 않은 결과 및 종결된 결과를 기준으로 필터링할 수 있습니다.

필터를 기준으로 억제 규칙을 만들어 결과 보기에서 결과를 제외할 수 있습니다. 자세한 설명은 억제 규칙을 사용하여 Amazon Inspector 결과를 숨기는 방법 섹션을 참조하세요.

Amazon Inspector 결과 필터링

결과 필터를 사용하면 지정한 기준과 일치하는 결과만 볼 수 있습니다. 필터 기준과 일치하지 않는 결과는 보기에서 제외됩니다. Amazon Inspector 콘솔을 사용하여 결과 필터를 생성할 수 있습니다. 이러한 필터를 사용하여 기존 결과 및 향후 결과를 자동으로 숨기려면 억제 규칙을 사용하여 Amazon Inspector 결과를 숨기는 방법 섹션을 참조하세요.

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 ID 값과 연결하면 됩니다.

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

결과 보기에 필터를 적용하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
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.

억제 규칙을 사용하여 Amazon Inspector 결과를 숨기는 방법

억제 규칙을 사용하여 기준과 일치하는 결과를 제외할 수 있습니다. 예를 들어 취약성 점수가 낮은 모든 발견을 차단하는 규칙을 만들어 가장 중요한 발견에만 집중할 수 있습니다.

> **Note**
>
> 금지 규칙은 검색 결과 목록을 필터링하는 데만 사용되며, 결과에 영향을 주거나 Amazon Inspector가 결과를 생성하는 것을 막지는 않습니다.

Amazon Inspector에서 금지 규칙과 일치하는 검색 결과를 생성하는 경우 검색 결과는 숨김으로 설정됩니다. 금지 규칙과 일치하는 검색 결과는 기본적으로 목록에 표시되지 않습니다.

Amazon Inspector는 숨겨진 검색 결과를 수정될 때까지 저장합니다. Amazon Inspector는 수정된 결과를 탐지합니다. Amazon Inspector는 수정된 검색 결과를 감지하면 검색 결과를 폐쇄로 설정하고 7일 동안 보관합니다.

제외된 검색 결과는 Amazon에 AWS Security Hub EventBridge 이벤트로 게시됩니다. EventBridge 규칙을 사용하여 검색 결과 상태를 변경하여 Security Hub에서 원치 않는 검색 결과를 자동으로 숨길 수 있습니다. 자세한 내용은 에서 자동 억제 규칙을 만드는 방법을 참조하십시오. AWS Security Hub

검색 결과를 닫거나 수정하는 금지 규칙을 만들 수 없습니다. 목록에 표시할 검색 결과를 필터링하는 금지 규칙만 만들 수 있습니다. 숨겨진 결과는 Amazon Inspector 콘솔에서 언제든지 볼 수 있습니다.

> **Note**
>
> 조직의 구성원 계정은 금지 규칙을 만들거나 관리할 수 없습니다.

억제 규칙 생성

억제 규칙을 생성하여 기본적으로 표시되는 결과 목록을 필터링할 수 있습니다. CreateFilterAPI를 사용하고 의 SUPRESS 값으로 지정하여 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. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
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. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
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. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
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. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 탐색 창에서 억제 규칙을 선택합니다.
3. 삭제하려는 억제 규칙 제목 옆의 확인란을 선택합니다.
4. 삭제를 선택한 다음 선택을 확인하여 규칙을 영구 삭제합니다.

Amazon Inspector에서 결과 보고서 내보내기

결과를 Amazon EventBridge 및 AWS Security Hub로 보내는 것 외에도, 선택적으로 결과를 Amazon Simple Storage Service(S3) 버킷에 결과 보고서로 내보낼 수 있습니다. 결과 보고서는 보고서에 포함되도록 선택한 결과의 세부 정보가 들어 있는 CSV 또는 JSON 파일로서, 특정 시점의 결과에 대한 자세한 스냅샷을 제공합니다. 각 결과에 대한 파일에는 영향을 받는 리소스의 Amazon 리소스 이

름(ARN), 결과가 생성된 날짜 및 시간, 관련된 일반적 취약성 및 노출(CVE) ID, 결과의 심각도, 상태, Amazon Inspector 및 CVSS 점수 등의 세부 정보가 포함됩니다.

결과 보고서를 구성할 때는 먼저 보고서에 포함할 결과를 지정해야 합니다. Amazon Inspector는 기본적으로 현재 AWS 리전 리전에서 활성 상태인 모든 결과에 대한 데이터를 제공합니다. 조직의 Amazon Inspector 위임 관리자인 경우 여기에 조직 내 모든 멤버 계정에 대한 결과 데이터가 포함됩니다.

선택적으로 데이터를 필터링하여 보고서를 사용자 지정할 수도 있습니다. 필터를 사용하면 특정 특성이 있는 결과에 대한 데이터를 포함하거나 제외할 수 있습니다. 예를 들어 특정 시간 범위 동안 생성된 모든 중요 결과, 특정 리소스에 대한 모든 활성 결과 또는 특정 유형의 모든 중요 결과 등을 필터링할 수 있습니다. 조직의 Amazon Inspector 관리자인 경우 필터를 사용하여 조직 내 특정 AWS 계정에 대한 결과가 포함된 보고서를 생성할 수 있습니다. 예를 들어 계정의 모든 중요 결과 중 활성 상태이고 수정이 가능한 결과 등이 포함될 수 있습니다. 그런 다음 해결을 위해 보고서를 계정 소유자와 공유할 수 있습니다.

> **Note**
>
> CreateFindingsReport API를 사용하여 결과 보고서를 내보내는 경우 기본적으로 활성 결과만 표시됩니다. 숨겨진 결과나 종결된 결과를 보려면 FindingStatus 필터 기준 값으로 `SUPPRESSED` 또는 `CLOSED`를 지정해야 합니다.

결과 보고서를 내보내면 Amazon Inspector에서는 사용자가 지정한 AWS Key Management Service(AWS KMS) 키로 데이터를 암호화하고 사용자가 지정한 S3 버킷에 보고서를 추가합니다. 암호화 키는 현재 AWS 리전에 있는 고객 관리형 AWS Key Management Service(AWS KMS) 대칭 암호화 키여야 합니다. 또한 키 정책에서 Amazon Inspector가 해당 키를 사용할 수 있도록 허용해야 합니다. S3 버킷도 현재 리전에 있어야 하며, 버킷 정책에서 Amazon Inspector가 버킷에 객체를 추가할 수 있도록 허용해야 합니다.

Amazon Inspector에서 보고서 암호화 및 저장을 완료한 후에는 지정한 S3 버킷에서 보고서를 다운로드하거나 다른 위치로 이동할 수 있습니다. 또는 보고서를 동일한 S3 버킷에 보관하고 이 버킷을 이후에 내보내는 결과 보고서의 리포지토리로 사용할 수도 있습니다.

이 주제에서는 AWS Management Console을 사용하여 결과 보고서를 내보내는 프로세스를 안내합니다. 이 프로세스는 필요한 권한이 있는지 확인하고 필요한 리소스를 구성한 다음 보고서를 구성하고 내보내는 것으로 구성됩니다.

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

작업

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

결과 보고서를 처음 내보낸 후에는 1–3단계를 선택적으로 수행할 수 있습니다. 이는 주로 후속 보고서에 동일한 S3 버킷과 AWS KMS key를 사용할지 여부에 따라 달라집니다.

1–3단계 후에 프로그래밍 방식으로 보고서를 내보내려면 Amazon Inspector API의 결과 CreateFindingsReport 작업을 사용합니다.

1단계: 권한 확인

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 IAM ID에 연결된 IAM 정책을 검토합니다. 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

AWS KMS의 경우 다음 작업을 수행할 수 있는지 확인합니다.

- kms:GetKeyPolicy
- kms:PutKeyPolicy

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

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

- kms:DescribeKey
- kms:ListAliases

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

보고서 암호화를 위한 KMS 키를 새로 생성하려는 경우 kms:CreateKey 작업을 수행할 수 있어야 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. 이 버킷은 내 계정의 기존 버킷일 수도 있고, 다른 AWS 계정의 소유지만 내가 액세스할 수 있는 기존 버킷일 수도 있습니다. 보고서를 새 버킷에 저장하려면 진행하기 전에 버킷을 생성합니다.

S3 버킷은 내보낼 결과 데이터와 동일한 AWS 리전에 있어야 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. 버킷 URI도 확보합니다. 보고서를 내보낼 때 이 URI를 입력해야 합니다.

버킷 정책을 업데이트하려면

1. Amazon S3 콘솔(https://console.aws.amazon.com/s3)을 엽니다.
2. 왼쪽 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 정책에는 JSON 형식이 사용됩니다. 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*은 버킷 이름입니다.
   - *111122223333*은 AWS 계정의 계정 ID입니다.
   - *Region*은 Amazon Inspector를 사용 중이며 Amazon Inspector가 버킷에 보고서를 추가할 수 있도록 허용하려는 AWS 리전입니다. 예를 들어 미국 동부(버지니아 북부) 리전의 경우 `us-east-1`입니다.

> **Note**
>
> 수동으로 활성화된 AWS 리전에서 Amazon Inspector를 사용하는 경우, `Service` 필드 값에 적절한 리전 코드도 추가합니다. 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.

예제 명령문에는 다음과 같은 두 개의 IAM 전역 조건 키를 사용하는 조건이 정의되어 있습니다.

- AWS:sourceAccount - 이 조건을 사용하면 Amazon Inspector에서 사용자의 계정에 대해서만 버킷에 보고서를 추가하고, 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. 예를 들면 다음과 같습니다.

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

- AWS:sourceARN - 이 조건은 버킷에 추가되는 객체의 소스에 따라 버킷에 대한 액세스를 제한하고, 다른 AWS 서비스가 버킷에 객체를 추가하지 못하도록 합니다. 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(ARN)을 추가합니다. 예를 들면 다음과 같습니다.

  ```
  "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 key 구성

권한을 확인하고 S3 버킷을 구성한 후에는 Amazon Inspector에서 결과 보고서를 암호화하는 데 사용할 AWS KMS key를 결정해야 합니다. 이 키는 고객 관리형 대칭 암호화 KMS 키여야 합니다. 또한 보고서를 저장하도록 구성한 S3 버킷과 동일한 AWS 리전에 있어야 합니다.

이 키는 내 계정의 기존 KMS 키이거나 다른 계정에서 소유하고 있는 기존 KMS 키일 수 있습니다. 새 KMS 키를 사용하려면 진행하기 전에 키를 생성합니다. 다른 계정에서 소유하고 있는 기존 키를 사용하려면 키의 Amazon 리소스 이름(ARN)을 확보합니다. Amazon Inspector에서 보고서를 내보낼 때 이 ARN을 입력해야 합니다. KMS 키 설정 생성 및 검토에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서에서 키 관리를 참조하세요.

사용하려는 KMS 키를 결정한 후에는 Amazon Inspector에 키 사용 권한을 부여합니다. 그렇지 않으면 Amazon Inspector에서 보고서를 암호화하고 내보낼 수 없습니다. Amazon Inspector에 키 사용 권한을 부여하려면 키에 대한 키 정책을 업데이트합니다. 키 정책 및 KMS 키 액세스 관리에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서에서 AWS KMS의 키 정책을 참조하세요.

키 정책을 업데이트하려면

> **Note**
>
> 다음 절차는 Amazon Inspector에서 사용할 수 있도록 기존 키를 업데이트하는 절차입니다. 기존 키가 없는 경우 키 생성 지침은 https://docs.aws.amazon.com/kms/latest/developerguide/create-keys.html을 참조하세요.

1. AWS KMS 콘솔(https://console.aws.amazon.com/kms)을 엽니다.
2. AWS 리전을 변경하려면 페이지의 오른쪽 상단 모서리에 있는 리전 선택기를 사용합니다.
3. 탐색 창에서 고객 관리형 키를 선택합니다.

4. 보고서를 암호화하는 데 사용할 KMS 키를 선택합니다. 이 키는 대칭 암호화(SYMMETRIC_DEFAULT) 키여야 합니다.
5. 키 정책 탭에서 편집을 선택합니다. 키 정책에서 편집 버튼이 보이지 않으면 먼저 정책 보기로 전환을 선택해야 합니다.
6. 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/*"
        }
    }
}
```

7. AWS KMS 콘솔의 키 정책 편집기에서 위의 명령문을 키 정책에 붙여넣어 정책에 추가합니다.

   명령문을 추가할 때 구문이 올바른지 확인합니다. 키 정책에는 JSON 형식이 사용됩니다. 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.

   - *111122223333*은 AWS 계정의 계정 ID입니다.
   - *Region*은 Amazon Inspector가 키를 사용하여 보고서를 암호화하도록 허용할 AWS 리전입니다. 예를 들어 미국 동부(버지니아 북부) 리전의 경우 `us-east-1`입니다.

> **Note**
>
> 수동으로 활성화된 AWS 리전에서 Amazon Inspector를 사용하는 경우, `Service` 필드 값에 적절한 리전 코드도 추가합니다. 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 전역 조건 키를 사용합니다.

- aws:SourceAccount - 이 조건을 사용하면 Amazon Inspector에서 사용자 계정에 대해서만 지정된 작업을 수행할 수 있습니다. 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. 예를 들면 다음과 같습니다.

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

- aws:SourceArn - 이 조건은 다른 AWS 서비스에서 지정된 작업을 수행하지 못하도록 합니다. 또한 사용자 계정에 대해 다른 작업을 수행하는 동안 Amazon Inspector가 키를 사용하지 못하도록 합니다. 즉, 객체가 결과 보고서인 경우에만, 그리고 해당 보고서가 조건에 지정된 계정과 리전에서 생성된 경우에만 Amazon Inspector가 해당 키로 S3 객체를 암호화할 수 있도록 허용합니다.

  Amazon Inspector가 추가 계정에 대해 지정된 작업을 수행할 수 있도록 하려면 이 조건에 각 추가 계정의 ARN을 추가합니다. 예를 들면 다음과 같습니다.

  ```
  "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 AWS KMS와의 트랜잭션 중에 Amazon Inspector가 혼동되는 대리자로 사용되는 것을 방지하는 데 도움이 됩니다. 권장하지는 않지만 명령문에서 이러한 조건을 제거할 수 있습니다.

9. 키 정책 업데이트가 완료되면 변경 사항 저장을 선택합니다.

4단계: 결과 보고서 구성 및 내보내기

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

결과 보고서를 구성하고 내보내려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
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.

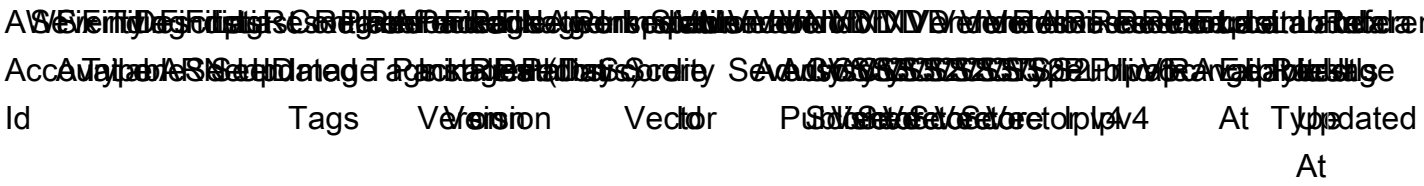
   > **Note**
   >
   > 필터 기준을 추가하는 것이 좋습니다. 추가하지 않으면 현재 AWS 리전에서 활성 상태인 모든 결과에 대한 데이터가 보고서에 포함됩니다. 조직의 Amazon Inspector 관리자인 경우 여기에 조직 내 모든 멤버 계정에 대한 결과 데이터가 포함됩니다.
   > 보고서에 전체 또는 여러 결과의 데이터가 포함되어 있는 경우 보고서를 생성하고 내보내는 데 시간이 오래 걸릴 수 있으며, 한 번에 하나의 보고서만 내보낼 수 있습니다.

4. 결과 내보내기를 선택합니다.
5. 내보내기 설정 섹션의 내보내기 파일 유형에서 보고서의 파일 형식을 지정합니다.

   - 데이터가 포함된 JavaScript Object Notation(.json) 파일을 생성하려면 JSON을 선택합니다.

     JSON 옵션을 선택하면 보고서에 각 결과에 대한 모든 필드가 포함됩니다. 가능한 JSON 필드 목록은 Amazon Inspector API 참조에서 결과 데이터 유형을 참조하세요.
   - 데이터가 포함된 쉼표로 구분된 값 (.csv) 파일을 생성하려면 CSV를 선택합니다.

CSV 옵션을 선택하면 각 결과에 대한 일부 필드, 즉 결과의 주요 속성을 보고하는 약 45개 필드만 보고서에 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 대한 참조 URL을 캡처하는 필드도 포함됩니다. 다음은 결과 보고서의 CSV 헤더 샘플입니다.



6. 내보내기 위치의 S3 URI에서 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**
     >
     > 또한 보고서에 대해 Amazon S3 경로 접두사를 지정하려면 슬래시(/)와 접두사를 S3 URI 상자의 값에 추가합니다. 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 URI 상자의 값에 **/111122223333**을 추가합니다.
     > 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 버킷 URI를 입력합니다. 예를 들어 **s3://DOC-EXAMPLE_BUCKET**과 같이 입력합니다. 여기서 DOC-EXAMPLE_BUCKET은 버킷의 이름입니다. 버킷 소유자가 버킷 속성에서 이 정보를 찾을 수 있습니다.

7. KMS 키에서 보고서를 암호화하는 데 사용할 AWS KMS key를 지정합니다.

   - 내 계정의 키를 사용하려면 목록에서 키를 선택합니다. 목록에 계정의 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 안내서에서 키 ID 및 ARN 찾기를 참조하세요.

8. 내보내기를 선택합니다.

Amazon Inspector에서 결과 보고서를 생성하고 지정한 KMS 키로 암호화한 다음 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 현재 AWS 리전에 있고 버킷 정책에 따라 Amazon Inspector가 버킷에 객체를 추가할 수 있는지 확인합니다. 또한 현재 리전에서 AWS KMS key가 활성화되어 있고 키 정책에서 Amazon Inspector가 키를 사용하도록 허용하는지 확인합니다.

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.

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

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

Amazon EventBridge를 사용하여 Amazon Inspector 결과에 대한 사용자 지정 응답 생성

Amazon Inspector는 새로 생성된 결과, 새로 집계된 결과, 결과 상태 변경이 발생할 경우 Amazon EventBridge 이벤트를 생성합니다. `updatedAt` 및 `lastObservedAt` 필드가 변경된 경우를 제외한 다른 변경 사항이 있을 경우 새 이벤트가 발행됩니다. 즉, 리소스를 다시 시작하거나 리소스와 관련된 태그를 변경하는 등의 조치를 취하면 결과에 대한 새 이벤트가 생성됩니다. 하지만 `id` 필드의 결과 ID는 동일하게 유지됩니다. 이벤트는 최선의 작업을 기반으로 발생됩니다.

> **Note**
>
> Amazon Inspector 위임 관리자 계정인 경우 EventBridge는 이벤트가 발생한 멤버 계정과 함께 사용자 계정에 이벤트를 발행합니다.

Amazon Inspector에서 EventBridge 이벤트를 사용하면 작업을 자동화하여 Amazon Inspector 결과로 나타난 보안 문제에 대응할 수 있습니다.

Amazon Inspector는 동일한 리전에 있는 기본 이벤트 버스로 이벤트를 보냅니다. 즉, Amazon Inspector를 실행 중인 리전마다 이벤트 규칙을 구성해야 해당 리전의 이벤트를 볼 수 있습니다.

EventBridge 이벤트에 따라 Amazon Inspector 결과에 대한 알림을 받으려면 EventBridge 규칙과 Amazon Inspector의 대상을 생성해야 합니다. 이 규칙을 사용하면 EventBridge를 통해 Amazon Inspector에서 생성한 결과에 대한 알림을 규칙에 지정된 대상에 보낼 수 있습니다. 자세한 내용은 Amazon EventBridge 사용 설명서에서 Amazon EventBridge 규칙을 참조하세요.

이벤트 스키마

다음은 EC2 결과 이벤트에 대한 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 결과를 알리는 EventBridge 규칙 생성

Amazon Inspector 결과의 가시성을 높이기 위해 EventBridge를 사용하여 자동 결과 알림을 메시징 허브로 보내도록 설정할 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.

1단계. Amazon SNS 주제 및 엔드포인트 설정

자동 알림을 설정하려면 먼저 Amazon Simple Notification Service에서 주제를 설정하고 엔드포인트를 추가해야 합니다. 자세한 내용은 SNS 설명서를 참조하세요.

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

Email setup

SNS 주제 생성

1. Amazon SNS 콘솔(https://console.aws.amazon.com/sns/v3/home)에 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 생성

1. Amazon SNS 콘솔(https://console.aws.amazon.com/sns/v3/home)에 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.

AWS Chatbot 클라이언트 구성

1. AWS Chatbot 콘솔(https://console.aws.amazon.com/chatbot/)로 이동합니다.
2. 구성된 클라이언트 창에서 새 클라이언트 구성을 선택합니다.
3. Slack을 선택한 다음 구성을 선택하여 확인합니다.

> **Note**
>
> Slack을 선택할 때는 허용을 선택하여 AWS Chatbot의 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 Management Console의 AWS Chatbot 창에서 Slack에서 복사한 채널 ID를 프라이빗 채널 ID 필드에 붙여넣습니다.
   e. 아직 역할이 없는 경우 권한에서 템플릿을 사용하여 IAM 역할을 생성하도록 선택합니다.
   f. 정책 템플릿에서 알림 권한을 선택합니다. 이는 AWS Chatbot에 대한 IAM 정책 템플릿입니다. 이 정책은 CloudWatch 경보, 이벤트, 로그, Amazon SNS 주제에 필요한 읽기 및 나열 권한을 제공합니다.
   g. 채널 가드레일 정책으로 AmazonInspector2ReadOnlyAccess를 선택합니다.
   h. 이전에 SNS 주제를 생성할 때 사용한 리전을 선택한 다음 생성한 Amazon SNS 주제를 선택하여 Slack 채널에 알림을 보냅니다.

5. 구성을 선택합니다.

Amazon Chime setup

SNS 주제 생성

1. Amazon SNS 콘솔(https://console.aws.amazon.com/sns/v3/home)에 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.

AWS Chatbot 클라이언트 구성

1. AWS Chatbot 콘솔(https://console.aws.amazon.com/chatbot/)로 이동합니다.
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. URL 복사를 선택하여 웹후크 URL을 클립보드에 복사합니다.
6. AWS Management Console의 AWS Chatbot 창에서 웹후크 URL 필드로 복사한 URL을 붙여넣습니다.
7. 아직 역할이 없는 경우 권한에서 템플릿을 사용하여 IAM 역할을 생성하도록 선택합니다.
8. 정책 템플릿에서 알림 권한을 선택합니다. 이는 AWS Chatbot에 대한 IAM 정책 템플릿입니다. CloudWatch 경보, 이벤트, 로그, Amazon SNS 주제에 필요한 읽기 및 나열 권한을 제공합니다.
9. 이전에 SNS 주제를 생성할 때 사용한 리전을 선택한 다음 생성한 Amazon SNS 주제를 선택하여 Amazon Chime 룸에 알림을 보냅니다.
10. 구성을 선택합니다.

2단계. Amazon Inspector 결과에 대한 EventBridge 규칙 생성

1. Amazon EventBridge 콘솔(https://console.aws.amazon.com/events/)을 엽니다.
2. 탐색 창에서 규칙을 선택한 다음 규칙 생성을 선택합니다.
3. 규칙의 이름과 설명(선택 사항)을 입력합니다.
4. 이벤트 패턴이 있는 규칙을 선택한 후다음을 선택합니다.
5. 이벤트 패턴 창에서 사용자 지정 패턴(JSON 편집기)을 선택합니다.
6. 다음 JSON을 편집기에 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.

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

Amazon Inspector 다중 계정 환경용 EventBridge

Amazon Inspector 위임 관리자인 경우 멤버 계정의 해당 결과에 따라 EventBridge 규칙이 계정에 표시됩니다. 이전 섹션에 설명된 대로, 관리자 계정의 EventBridge를 통해 결과 알림을 설정하면 다중 계정

에 대한 알림을 받게 됩니다. 즉, 내 계정에서 생성된 결과와 이벤트 외에도 멤버 계정에서 생성된 결과와 이벤트에 대한 알림을 받게 됩니다.

결과의 JSON 세부 정보에 있는 `accountId`를 사용하여 Amazon Inspector 결과가 발생한 멤버 계정을 식별할 수 있습니다.

Amazon Inspector를 사용하여 SBOM 내보내기

Amazon Inspector 콘솔 또는 API를 사용하여 리소스에 대한 Software Bill of Materials(SBOM)를 생성할 수 있습니다. SBOM은 코드베이스의 모든 오픈 소스 및 타사 소프트웨어 구성 요소의 중첩된 인벤토리입니다. Amazon Inspector는 사용 환경의 개별 리소스에 대한 SBOM을 제공합니다. Amazon Inspector에서 내보낸 SBOM을 사용하면 가장 일반적으로 사용되는 패키지, 조직 전체의 관련 취약성 등 소프트웨어 공급에 대한 정보를 파악하는 데 도움이 됩니다.

Amazon Inspector에서 적극적으로 모니터링하고 있는 모든 지원 리소스의 SBOM을 내보낼 수 있습니다. 리소스 상태는 AWS 환경의 Amazon Inspector 적용 범위 평가를 통해 검토할 수 있습니다.

> **Note**
>
> Amazon Inspector는 Windows EC2 인스턴스에 대한 SBOM 내보내기를 지원하지 않습니다.

Amazon Inspector 형식

Amazon Inspector는 CycloneDX 1.4 및 SPDX 2.3 호환 형식으로 SBOM 내보내기를 지원합니다. Amazon Inspector는 선택한 Amazon S3 버킷에 SBOM을 JSON 파일로 내보냅니다.

> **Note**
>
> Amazon Inspector에서 내보내는 SPDX 형식은 SPDX 2.3을 사용하는 시스템과 호환되지만, Creative Commons Zero(CC0) 필드가 포함되어 있지 않습니다. 이 필드가 포함되어 있으면 사용자가 자료를 재배포하거나 편집할 수 있기 때문입니다.

Amazon Inspector의 CyclonedX 1.4 SBOM 형식 예제

```
                {
"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"
        }
      ]
    }
  ]
}
```

Amazon Inspector의 SPDX 2.3 SBOM 형식 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"
}
```

SBOM용 필터

SBOM을 내보낼 때 필터를 포함시켜 리소스의 특정 하위 집합에 대한 보고서를 생성할 수 있습니다. 필터를 제공하지 않으면 지원되는 모든 활성 리소스의 SBOM을 내보냅니다. 또한 위임 관리자인 경우 여기에 모든 멤버를 위한 리소스도 포함됩니다. 다음과 같은 필터를 사용할 수 있습니다.

- AccountID — 이 필터는 특정 계정 ID와 연결된 리소스의 SBOM을 내보내는 데 사용할 수 있습니다.
- EC2 인스턴스 태그 — 이 필터는 특정 태그가 있는 EC2 인스턴스의 SBOM을 내보내는 데 사용할 수 있습니다.
- 함수 이름 — 이 필터는 특정 Lambda 함수의 SBOM을 내보내는 데 사용할 수 있습니다.
- 이미지 태그 — 이 필터는 특정 태그가 있는 컨테이너 이미지의 SBOM을 내보내는 데 사용할 수 있습니다.
- Lambda 함수 태그 — 이 필터는 특정 태그가 있는 Lambda 함수의 SBOM을 내보내는 데 사용할 수 있습니다.
- 리소스 유형 — 이 필터는 리소스 유형(EC2/ECR/Lambda)을 필터링하는 데 사용할 수 있습니다.

- 리소스 ID — 이 필터는 특정 리소스의 SBOM을 내보내는 데 사용할 수 있습니다.
- 리포지토리 이름 — 이 필터는 특정 리포지토리에 있는 컨테이너 이미지의 SBOM을 생성하는 데 사용할 수 있습니다.

SBOM 구성 및 내보내기

SBOM을 내보내려면 먼저 Amazon Inspector에서 사용할 수 있는 Amazon S3 버킷과 AWS KMS 키를 구성해야 합니다. 필터를 사용하여 리소스의 특정 하위 집합에 대한 SBOM을 내보낼 수 있습니다. AWS Organization 내 여러 계정의 SBOM을 내보내려면 Amazon Inspector 위임 관리자로 로그인한 상태에서 다음 단계를 수행합니다.

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 Inspector에서 보고서를 암호화하는 데 사용할 수 있도록 허용하는 정책으로 구성된 AWS KMS 키. 정책 구성에 대한 자세한 내용은 Configure an AWS KMS key for export를 참조하세요.

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

원하는 액세스 방법을 선택하여 SBOM을 내보냅니다.

Console

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단에 있는 AWS 리전 선택기를 사용하여 SBOM을 내보낼 리소스가 있는 리전을 선택합니다.
3. 탐색 창에서 SBOM 내보내기를 선택합니다.
4. (선택 사항) SBOM 내보내기 페이지에서 필터 추가 메뉴를 사용하여 보고서를 생성할 리소스의 하위 집합을 선택합니다. 필터를 제공하지 않으면 Amazon Inspector에서 모든 활성 리소스

에 대한 보고서를 내보냅니다. 위임 관리자인 경우 여기에 조직의 모든 활성 리소스가 포함됩니다.

5. 내보내기 설정에서 SBOM에 사용할 형식을 선택합니다.
6. Amazon S3 URI를 입력하거나 Amazon S3 찾아보기를 선택하여 SBOM을 저장할 Amazon S3 위치를 선택합니다.
7. Amazon Inspector에서 보고서를 암호화하는 데 사용하도록 구성된 AWS KMS 키를 입력합니다.

API

- 리소스의 SBOM을 프로그래밍 방식으로 내보내려면 Amazon Inspector API의 CreatesBomExport 작업을 사용합니다.

  요청에서 `reportFormat` 파라미터를 사용하여 SBOM 출력 형식을 지정하고 `CYCLONEDX_1_4` 또는 `SPDX_2_3`을 선택합니다. `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

- AWS Command Line Interface를 사용하여 리소스의 sBOM을 내보내려면 다음 명령을 실행합니다.

  ```
  aws inspector2 create-sbom-export --report-format
  FORMAT --s3-destination bucketName=DOC-EXAMPLE-
  BUCKET1,keyPrefix=PREFIX,kmsKeyArn=arn:aws:kms:Region:111122223333:key/123
  ```

  요청에서 *FORMAT*을 `CYCLONEDX_1_4` 또는 `SPDX_2_3` 중에서 원하는 형식으로 바꾸세요. 그런 다음 s3 대상의 *user input placeholders*를 내보낼 대상 S3 버킷의 이름, S3의 출력에 사용할 접두사, 보고서를 암호화하는 데 사용하는 KMS 키의 ARN으로 바꾸세요.

아마존 인스펙터 취약성 데이터베이스 검색

Amazon Inspector 취약성 데이터베이스에서 취약성 및 노출 (CVE) 을 검색할 수 있습니다. Amazon Inspector는 취약성 데이터베이스의 정보를 사용하여 CVE ID와 관련된 세부 정보를 생성합니다. CVE 세부 정보 페이지에서 이러한 세부 정보에 액세스할 수 있습니다.

이 주제에서는 CVE ID를 사용하여 Amazon Inspector 취약성 데이터베이스를 검색하고 CVE 세부 정보 페이지를 해석하는 방법을 설명합니다. 조사 결과에 대한 자세한 내용은 을 참조하십시오. Amazon Inspector 결과 세부 정보

> **Note**
>
> Amazon Inspector는 데이터베이스의 다른 소프트웨어 취약성을 추적하고 탐지 결과를 생성합니다. 하지만 Amazon Inspector는 CVE 세부 정보 페이지의 탐지 플랫폼 섹션에 플랫폼이 나열된 CVE만 지원합니다. 현재 CVE 검색은 지원되지 않습니다. Microsoft Windows

취약성 데이터베이스 검색

이 섹션에서는 콘솔과 Amazon Inspector API를 사용하여 취약성 데이터베이스를 검색하는 방법을 설명합니다.

> **Note**
>
> 취약성 데이터베이스를 검색하려면 AWS 리전 먼저 현재 Amazon Inspector를 활성화해야 합니다.

Console

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/)을 엽니다.
2. 탐색 창에서 취약성 데이터베이스 검색을 선택합니다.
3. 검색 창에 CVE ID를 입력하고 검색을 선택합니다.

API

Amazon Inspector SearchVulnerabilitiesAPI를 실행하고 다음 형식으로 단일 CVE ID를 제공하십시오. `filterCriteria CVE-<year>-<ID>`

CVE 세부 정보 이해

이 섹션에서는 CVE 세부 정보 페이지를 해석하는 방법을 설명합니다.

CVE 세부 정보

CVE 세부 정보 섹션에는 다음 정보가 포함됩니다.

- CVE 설명 및 ID
- CVE 심각도
- 일반 취약성 평가 시스템 (CVSS) 및 악용 예측 평가 시스템 (EPSS) 점수
- 탐지 플랫폼

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

- 일반적인 약점 열거 (CWE)
- 공급업체 생성 및 업데이트 날짜

취약성 인텔리전스

취약성 인텔리전스 섹션에서는 악용 대상 및 마지막으로 알려진 공개 악용 날짜와 같은 위협 인텔리전스 데이터를 제공합니다.

또한 사이버 보안 및 인프라 보안국 (CISA) 의 데이터도 제공합니다. 여기에는 수정 조치, 알려진 악용 취약성 카탈로그에 CVE가 추가된 날짜, CISA가 연방 기관이 CVE를 수정할 것으로 예상되는 날짜 등이 포함됩니다.

참조

참고 문헌 섹션은 CVE에 대한 자세한 정보를 제공하는 리소스 링크를 제공합니다.

Amazon Inspector 이벤트에 대한 Amazon EventBridge 이벤트 스키마

다른 애플리케이션, 서비스 및 시스템(예: 모니터링 또는 이벤트 관리 시스템)과의 통합을 지원하기 위해 Amazon Inspector에서는 결과를 자동으로 Amazon EventBridge에 이벤트로 게시합니다. EventBridge는 애플리케이션 및 기타 AWS 서비스의 실시간 데이터 스트림을 AWS Lambda 함수, Amazon Simple Notification Service 주제, Amazon Kinesis Data Streams 등의 대상에 전달하는 서버리스 이벤트 버스 서비스입니다. EventBridge 및 EventBridge 이벤트에 대해 자세히 알아보려면 Amazon EventBridge 사용 설명서https://docs.aws.amazon.com/eventbridge/latest/userguide/eb-what-is.html를 참조하세요.

Amazon Inspector에서는 결과, 리소스 적용 범위 변경, 개별 리소스 최초 스캔에 대한 이벤트를 게시합니다. 각 이벤트는 AWS 이벤트에 대한 EventBridge 스키마를 준수하는 JSON 객체입니다. 데이터가 EventBridge 이벤트로 구조화되기 때문에 다른 애플리케이션, 서비스 및 도구를 사용하여 결과 및 지원되는 Amazon Inspector 이벤트를 더 쉽게 모니터링하고 처리 및 조치할 수 있습니다.

주제

Amazon Inspector의 Amazon EventBridge 기본 스키마

다음은 Amazon Inspector의 EventBridge 이벤트에 대한 기본 스키마 예제입니다. 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 이벤트의 스키마 예제입니다. 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 Amazon EventBridge를 사용하여 Amazon Inspector 결과에 대한 사용자 지정 응답 생성을 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": "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 EC2 network reachability finding

```
{
    "version": "0",
    "id": "d0384f63-1621-1b75-d014-a5e45628ef3e",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T09:17:57Z",
    "region": "us-east-1",
```

```
    "resources": ["i-0a96278c2206a8e4b"],
    "detail": {
        "awsAccountId": "111122223333",
        "description": "On the instance i-0a96278c2206a8e4b, the port range
 22-22 is reachable from the InternetGateway igw-72069c09 from an attached ENI
 eni-0976efe678170408f.",
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 20, 2023, 9:17:57 AM",
        "lastObservedAt": "Jan 20, 2023, 9:17:57 AM",
        "networkReachabilityDetails": {
            "networkPath": {
                "steps": [{
                    "componentId": "igw-72069c09",
                    "componentType": "AWS::EC2::InternetGateway"
                }, {
                    "componentId": "acl-91d74eec",
                    "componentType": "AWS::EC2::NetworkAcl"
                }, {
                    "componentId": "sg-0aaed0af450bd0165",
                    "componentType": "AWS::EC2::SecurityGroup"
                }, {
                    "componentId": "eni-0976efe678170408f",
                    "componentType": "AWS::EC2::NetworkInterface"
                }, {
                    "componentId": "i-0a96278c2206a8e4b",
                    "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::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-0b5eea76982371e91",
                    "ipV4Addresses": ["3.89.90.19", "172.31.93.57"],
                    "ipV6Addresses": [],
                    "keyName": "example-inspector-test",
                    "launchedAt": "Jan 19, 2023, 7:25:02 PM",
                    "platform": "AMAZON_LINUX_2",
                    "subnetId": "subnet-8213f2a3",
                    "type": "t2.micro",
                    "vpcId": "vpc-ab6650d1"
                }
            },
            "id": "i-0a96278c2206a8e4b",
            "partition": "aws",
            "region": "us-east-1",
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "Port 22 is reachable from an Internet Gateway",
        "type": "NETWORK_REACHABILITY",
        "updatedAt": "Jan 20, 2023, 9:17:57 AM"
    }
}
```

Amazon ECR package vulnerability finding

```
{
    "version": "0",
    "id": "5b52952e-26df-3a51-6d14-4dbe737e58ec",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T21:59:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ecr:us-east-1:111122223333:repository/inspector2/
sha256:98f0304b3a3b7c12ce641177a99d1f3be56f532473a528fda38d53d519cafb13"
    ],
    "detail": {
```

```
        "awsAccountId": "111122223333",
        "description": "libcurl would reuse a previously created connection even
 when a TLS or SSHrelated option had been changed that should have prohibited
 reuse.libcurl keeps previously used connections in a connection pool for
 subsequenttransfers to reuse if one of them matches the setup. However, several TLS
 andSSH settings were left out from the configuration match checks, making themmatch
 too easily.",
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 9:59:00 PM",
        "fixAvailable": "YES",
        "inspectorScore": 7.5,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "adjustments": [],
                "cvssSource": "NVD",
                "score": 7.5,
                "scoreSource": "NVD",
                "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Jan 19, 2023, 9:59:00 PM",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 5,
                    "scoringVector": "AV:N/AC:L/Au:N/C:N/I:P/A:N",
                    "source": "NVD",
                    "version": "2.0"
                },
                {
                    "baseScore": 7.5,
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:N",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://hackerone.com/reports/1555796",
                "https://security.gentoo.org/glsa/202212-01",
                "https://lists.debian.org/debian-lts-announce/2022/08/
msg00017.html",
```

```
                "https://www.debian.org/security/2022/dsa-5197"
            ],
            "relatedVulnerabilities": [],
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2022-27782",
            "vendorCreatedAt": "Jun 2, 2022, 2:15:00 PM",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Jan 5, 2023, 5:51:00 PM",
            "vulnerabilityId": "CVE-2022-27782",
            "vulnerablePackages": [
                {
                    "arch": "X86_64",
                    "epoch": 0,
                    "fixedInVersion": "0:7.61.1-22.el8_6.3",
                    "name": "libcurl",
                    "packageManager": "OS",
                    "release": "22.el8",
                    "remediation": "yum update libcurl",
                    "sourceLayerHash":
"sha256:38a980f2cc8accf69c23deae6743d42a87eb34a54f02396f3fcfd7c2d06e2c5b",
                    "version": "7.61.1"
                },
                {
                    "arch": "X86_64",
                    "epoch": 0,
                    "fixedInVersion": "0:7.61.1-22.el8_6.3",
                    "name": "curl",
                    "packageManager": "OS",
                    "release": "22.el8",
                    "remediation": "yum update curl",
                    "sourceLayerHash":
"sha256:38a980f2cc8accf69c23deae6743d42a87eb34a54f02396f3fcfd7c2d06e2c5b",
                    "version": "7.61.1"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
```

```
                    "awsEcrContainerImage": {
                        "architecture": "amd64",
                        "imageHash":
 "sha256:98f0304b3a3b7c12ce641177a99d1f3be56f532473a528fda38d53d519cafb13",
                        "imageTags": [
                            "o3"
                        ],
                        "platform": "ORACLE_LINUX_8",
                        "pushedAt": "Jan 19, 2023, 7:38:39 PM",
                        "registry": "111122223333",
                        "repositoryName": "inspector2"
                    }
                },
                "id": "arn:aws:ecr:us-east-1:111122223333:repository/inspector2/
sha256:98f0304b3a3b7c12ce641177a99d1f3be56f532473a528fda38d53d519cafb13",
                "partition": "aws",
                "region": "us-east-1",
                "type": "AWS_ECR_CONTAINER_IMAGE"
            }
        ],
        "severity": "HIGH",
        "status": "ACTIVE",
        "title": "CVE-2022-27782 - libcurl, curl",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 9:59:00 PM"
    }
}
```

Lambda package vulnerability finding

```
{
    "version": "0",
    "id": "040bb590-3a12-353f-ecb1-05e54b0fbea7",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T19:20:25Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:lambda:us-east-1:111122223333:function:ExampleFunction:$LATEST"
    ],
```

```
    "detail": {
        "awsAccountId": "111122223333",
        "description": "Those using Woodstox to parse XML data may be vulnerable to
 Denial of Service attacks (DOS) if DTD support is enabled. If the parser is running
 on user supplied input, an attacker may supply content that causes the parser to
 crash by stackoverflow. This effect may support a denial of service attack.",
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/
FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 7:20:25 PM",
        "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:N/I:N/A:H",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Jan 19, 2023, 7:20:25 PM",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 7.5,
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:N/A:H",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://bugs.chromium.org/p/oss-fuzz/issues/detail?id=47434"
            ],
            "relatedVulnerabilities": [],
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2022-40152",
            "vendorCreatedAt": "Sep 16, 2022, 10:15:00 AM",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Nov 25, 2022, 11:15:00 AM",
            "vulnerabilityId": "CVE-2022-40152",
            "vulnerablePackages": [
                {
                    "epoch": 0,
```

```
                    "filePath": "lib/woodstox-core-6.2.7.jar",
                    "fixedInVersion": "6.4.0",
                    "name": "com.fasterxml.woodstox:woodstox-core",
                    "packageManager": "JAR",
                    "remediation": "Update woodstox-core to 6.4.0",
                    "version": "6.2.7"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
                            "X86_64"
                        ],
                        "codeSha256": "+EwrOrht2um4fdVCD73gj
+O7HJIAUvUxi8AD0eKHSkc=",
                        "executionRoleArn": "arn:aws:iam::111122223333:role/
ExampleFunction-ExecutionRole",
                        "functionName": "Example-function",
                        "lastModifiedAt": "Nov 7, 2022, 8:29:27 PM",
                        "packageType": "ZIP",
                        "runtime": "JAVA_11",
                        "version": "$LATEST"
                    }
                },
                "id": "arn:aws:lambda:us-
east-1:111122223333:function:ExampleFunction:$LATEST",
                "partition": "aws",
                "region": "us-east-1",
                "tags": {
                    "TargetAlias": "DeploymentStack",
                    "SoftwareType": "Infrastructure"
                },
                "type": "AWS_LAMBDA_FUNCTION"
            }
        ],
        "severity": "HIGH",
```

```
        "status": "ACTIVE",
        "title": "CVE-2022-40152 - com.fasterxml.woodstox:woodstox-core",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 7:20:25 PM"
    }
}
```

Lambda code vulnerability finding

```
{
   "version":"0",
   "id":"9df01cb1-df24-bc46-5650-085a4087e7aa",
   "detail-type":"Inspector2 Finding",
   "source":"aws.inspector2",
   "account":"111122223333",
   "time":"2023-12-07T22:14:45Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:lambda:us-east-1:111122223333:function:code-finding:$LATEST"
   ],
   "detail":{
      "awsAccountId":"111122223333",
      "codeVulnerabilityDetails":{
         "detectorId":"python/lambda-override-reserved@v1.0",
         "detectorName":"Override of reserved variable names in a Lambda function",
         "detectorTags":[
            "availability",
            "aws-python-sdk",
            "aws-lambda",
            "data-integrity",
            "maintainability",
            "security",
            "security-context",
            "python"
         ],
         "filePath":{
            "endLine":6,
            "fileName":"lambda_function.py",
            "filePath":"lambda_function.py",
            "startLine":6
         },
```

```
         "ruleId":"Rule-434311"
      },
      "description":"Overriding environment variables that are reserved by AWS
 Lambda might lead to unexpected behavior or failure of the Lambda function.",
      "findingArn":"arn:aws:inspector2:us-east-1:111122223333:finding/FINDING_ID",
      "firstObservedAt":"Aug 8, 2023, 7:33:58 PM",
      "lastObservedAt":"Dec 7, 2023, 10:14:45 PM",
      "remediation":{
         "recommendation":{
            "text":"Your code attempts to override an environment variable that is
 reserved by the Lambda runtime environment. This can lead to unexpected behavior
 and might break the execution of your Lambda function.\n\n[Learn more](https://
docs.aws.amazon.com/lambda/latest/dg/configuration-envvars.html#configuration-
envvars-runtime)"
         }
      },
      "resources":[
         {
            "details":{
               "awsLambdaFunction":{
                  "architectures":[
                     "X86_64"
                  ],
                  "codeSha256":"2mtfH+CgubesG6NYpb2zEqBja5WN6FfbH4AAYDuF8RE=",
                  "executionRoleArn":"arn:aws:iam::193043430472:role/service-role/
code-finding-role-7jgg3wan",
                  "functionName":"code-finding",
                  "lastModifiedAt":"Dec 7, 2023, 10:12:48 PM",
                  "packageType":"ZIP",
                  "runtime":"PYTHON_3_7",
                  "version":"$LATEST"
               }
            },
            "id":"arn:aws:lambda:us-east-1:193043430472:function:code-finding:
$LATEST",
            "partition":"aws",
            "region":"us-east-1",
            "type":"AWS_LAMBDA_FUNCTION"
         }
      ],
      "severity":"HIGH",
      "status":"ACTIVE",
      "title":"Overriding environment variables that are reserved by AWS Lambda
 might lead to unexpected behavior.",
```

```
      "type":"CODE_VULNERABILITY",
      "updatedAt":"Dec 7, 2023, 10:14:45 PM"
   }
}
```

> **Note**
>
> detail 값은 단일 결과에 대한 JSON 세부 정보를 객체로 반환하며, 배열 내의 여러 결과를 지원하는 전체 결과 응답 구문은 반환하지 않습니다.

Amazon Inspector 최초 스캔 완료 이벤트 스키마 예제

다음은 최초 스캔 완료에 대한 Amazon Inspector 이벤트의 EventBridge 이벤트 스키마 예제입니다. 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 이벤트의 EventBridge 이벤트 스키마 예제입니다. 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 컨테이너 이미지 스캔을 CI/CD 파이프라인에 직접 통합하여 소프트웨어 취약성을 스캔하고 빌드가 끝날 때 보고서를 제공할 수 있습니다. Amazon Inspector에서 생성한 취약성 보고서를 통해 배포 전에 위험을 조사하고 해결할 수 있습니다.

Amazon Inspector CI/CD 통합은 Amazon Inspector SBOM 생성기와 Amazon Inspector 스캔 API의 조합을 활용하여 컨테이너 이미지에 대한 취약성 보고서를 생성합니다. Amazon Inspector SBOM 생성기는 제공된 컨테이너 이미지로부터 소프트웨어 재료 명세서(SBOM)를 생성합니다. 그런 다음, Amazon Inspector 스캔 API가 해당 SBOM을 스캔하고 발견된 취약성에 대한 세부 정보가 포함된 보고서를 생성합니다.

개별 CI/CD 솔루션용으로 개발되고 해당 마켓플레이스에서 사용할 수 있는 Amazon Inspector 플러그인을 통해 Amazon Inspector와 CI/CD를 통합하거나 사용자 지정 스캔 통합을 생성할 수 있습니다.

주제

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 단계에서는 제공한 이미지에서 Amazon Inspector SBOM 생성기를 실행한 다음, 생성된 SBOM에서 Amazon Inspector 스캔 API를 실행합니다.

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

1. Amazon Inspector 스캔 API에 대한 액세스를 AWS 계정 허용하도록 구성합니다. 지침은 Amazon Inspector CI/CD 통합을 사용하기 위한 AWS 계정 설정 섹션을 참조하십시오.

2. 마켓플레이스에서 Amazon Inspector 플러그인을 설치합니다.
3. Amazon Inspector SBOM 생성기 바이너리를 설치하고 구성합니다. 지침은 Amazon Inspector SBOM 생성기 섹션을 참조하십시오.
4. Amazon Inspector 스캔을 CI/CD 파이프라인의 빌드 단계로 추가하고 스캔을 구성합니다.
5. 빌드를 실행하면 플러그인이 컨테이너 이미지를 입력으로 받은 다음 이미지에서 Amazon Inspector SBOM 생성기를 실행하여 CycloneDX와 호환되는 SBOM을 생성합니다.
6. 그러면 플러그인은 생성된 SBOM을 Amazon Inspector 스캔 API 엔드포인트로 전송합니다. Amazon Inspector 스캔 API 엔드포인트는 각 SBOM 구성 요소의 취약성을 평가합니다.
7. Amazon Inspector 스캔 API 응답은 CSV, SBOM, JSON 및 HTML 형식의 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

사용자 지정 통합

Amazon Inspector에서 CI/CD 솔루션용 플러그인을 제공하지 않는 경우, Amazon Inspector SBOM 생성기와 Amazon Inspector 스캔 API를 함께 사용하여 사용자 지정 CI/CD 통합을 생성할 수 있습니다. 또한 사용자 지정 통합을 사용하여 Amazon Inspector SBOM 생성기를 통해 제공되는 옵션을 사용하여 스캔을 미세 조정할 수 있습니다.

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

1. Amazon Inspector 스캔 API에 대한 액세스를 AWS 계정 허용하도록 구성합니다. 지침은 Amazon Inspector CI/CD 통합을 사용하기 위한 AWS 계정 설정 섹션을 참조하십시오.
2. Amazon Inspector SBOM 생성기 바이너리를 설치하고 구성합니다. 지침은 Amazon Inspector SBOM 생성기 섹션을 참조하십시오.
3. Amazon Inspector SBOM 생성기를 사용하여 컨테이너 이미지에 대해 CycloneDX와 호환되는 SBOM을 생성합니다.

4. 생성된 SBOM에서 Amazon Inspector 스캔 API를 사용하여 취약성 보고서를 생성합니다.

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

Amazon Inspector CI/CD 통합을 사용하기 위한 AWS 계정 설정

Amazon Inspector AWS 계정 CI/CD 통합을 사용하려면 에 가입해야 합니다. 파이프라인에 Amazon Inspector Scan API에 대한 액세스 권한을 부여하는 IAM 역할이 AWS 계정 있어야 합니다.

다음 주제의 작업을 완료하여 등록하고, 관리자 사용자를 생성하고 AWS 계정, CI/CD 통합을 위한 IAM 역할을 구성하십시오.

> **Note**
>
> 이미 가입한 AWS 계정경우 으로 건너뛰어도 됩니다. CI/CD 통합을 위한 IAM 역할 구성

주제

- 가입해 보세요. AWS 계정
- 관리자 사용자 생성
- CI/CD 통합을 위한 IAM 역할 구성

가입해 보세요. AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

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

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

   에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자는 계정의 모든 AWS 서비스 및 리소스에 액세스하는 권한이 있습니다. 보안 모범 사례는 관리 사용자에게 관리자 액세

스 권한을 할당하고, 루트 사용자만 루트 사용자 액세스 권한이 필요한 태스크를 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 이동하고 내 계정을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요 AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 암호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면 AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.
2. 루트 사용자에 대해 다중 인증(MFA)을 활성화합니다.

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

관리 사용자 생성

1. IAM Identity Center를 활성화합니다.

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

   를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

관리 사용자로 로그인

- IAM 자격 증명 센터 사용자로 로그인하려면 IAM 자격 증명 센터 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인을 참조하십시오.AWS 로그인

CI/CD 통합을 위한 IAM 역할 구성

Amazon Inspector 스캔을 CI/CD 파이프라인에 통합하려면 소프트웨어 재료 명세서(SBOM)를 스캔하는 Amazon Inspector 스캔 API에 대한 액세스를 허용하는 IAM 정책을 생성해야 합니다. 그런 다음 Amazon Inspector 스캔 API를 실행하기 위해 계정에서 위임할 수 있는 IAM 역할에 해당 정책을 연결할 수 있습니다.

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/iam/ 에서 IAM 콘솔을 엽니다.
2. IAM 콘솔의 탐색 창에서 정책을 선택한 후 정책 생성을 선택합니다.
3. 정책 편집기에서 JSON을 선택한 후 다음 문을 붙여 넣습니다.

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

4. 다음을 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 생성을 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. 다음을 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

Amazon Inspector SBOM 생성기(Sbomgen)는 컨테이너 이미지에 대한 소프트웨어 재료 명세서(SBOM 를 생성하는 바이너리 도구입니다. SBOM은 시스템에 설치된 소프트웨어의 수집된 인벤토리입니다.

Sbomgen은 설치된 패키지에 대한 정보가 들어 있는 것으로 알려진 파일을 스캔하는 방식으로 작동합니다. 이러한 파일 중 하나가 발견되면 이 도구는 패키지 이름, 버전 및 기타 메타데이터를 추출합니다. 그러면 이 패키지 메타데이터가 CycloneDX SBOM으로 변환됩니다.

Sbomgen은 CycloneDX SBOM을 파일 또는 STDOUT에 제공하는 독립 실행형 도구로 사용할 수 있습니다. 또한 배포 파이프라인의 일부로 컨테이너 이미지를 자동으로 스캔하는 Amazon Inspector CI/CD 통합의 일부로도 사용됩니다. 자세한 내용은 Amazon Inspector 스캔을 CI/CD 파이프라인에 통합 섹션을 참조하세요.

지원되는 패키지 및 이미지 형식

현재 Sbomgen은 다음 패키지 유형에 대한 인벤토리를 수집할 수 있습니다.

- Alpine APK
- Debian / Ubuntu DPKG
- Red Hat RPM

- `go.mod` 및 `go mod cache`를 통한 Go 패키지
- `pom.properties`를 통한 Java 패키지
- `node_modules` 내부의 `package.json` 파일을 통한 Node.js 패키지
- Nuget 파일을 통한 C# 패키지(`.deps.json`, `csproj`, `Packages.config`, packages.lock.json)
- `installed.json` 및 `composer.lock`을 통한 PHP
- `requirements.txt`, `Pipfile.lock`, `poetry.lock`, 및 `egg/wheel` 파일을 통한 Python 패키지
- `Gemfile.lock`, `.gemspec` 및 전 세계에 설치된 젬을 통한 Ruby 패키지
- `Cargo.lock` 및 `Cargo.toml`를 통한 Rust 패키지

Sbomgen는 이미지에 다음과 같은 컨테이너 이미지 매니페스트 형식을 지원합니다.

- OCI 이미지 매니페스트
- Docker 이미지 매니페스트 버전 2, 스키마 2
- Docker 이미지 매니페스트 버전 2, 스키마 1
- Docker 이미지 매니페스트 버전 1

> **Important**
>
> 컨테이너 이미지 크기가 5GB를 초과하거나, 계층이 60개 이상이거나, 설치된 패키지가 2,000개가 넘는 경우 Sbomgen은 컨테이너 이미지를 스캔할 수 없습니다.

Amazon Inspector SBOM 생성기 설치(Sbomgen)

Sbomgen은 Linux 운영 체제에서만 사용할 수 있습니다. 컨테이너 이미지를 분석하는 데 사용하는 경우 Docker, Podman, 또는 containerd와 같은 컨테이너 서비스가 설치되어 있어야 합니다.

최상의 성능을 위해 다음과 같은 최소 하드웨어 사양을 갖춘 시스템에서 바이너리를 실행하는 것이 좋습니다.

- 4x 코어 CPU
- 8GB RAM

Sbomgen을(를) 설치하려면

1. 아키텍처에 맞는 올바른 URL에서 Sbomgen zip 파일을 다운로드하세요.

   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
2. 다음 명령을 사용하여 다운로드를 압축 해제합니다.

   ```
   unzip inspector-sbomgen.zip
   ```
3. 아카이브에서 다음 파일을 확인하세요.

   - `inspector-sbomgen` - SBOM을 생성하기 위해 실행할 바이너리입니다.
   - `README.txt` - Sbomgen 사용 설명서입니다.
   - `LICENSE.txt` - 이 파일에는 Sbomgen에 대한 소프트웨어 라이선스가 들어 있습니다.
   - `licenses` - 이 폴더에는 Sbomgen에서 사용하는 서드 파티 패키지의 라이선스 정보가 들어 있습니다.
   - `checksums.txt` - 이 파일은 Sbomgen 바이너리의 해시를 제공합니다.
   - `sbom.json` - Sbomgen 바이너리의 CycloneDX SBOM입니다.
4. (선택 사항) 다음 명령을 사용하여 바이너리의 신뢰성 및 무결성을 확인합니다.

   ```
   sha256sum < inspector-sbomgen
   ```

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

   ```
   chmod +x inspector-sbomgen
   ```
6. 다음 명령을 실행하여 Sbomgen이 성공적으로 설치되었는지 확인합니다.

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

   다음과 유사한 출력 화면이 표시되어야 합니다.

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

Sbomgen 사용하기

Sbomgen을 사용하여 컨테이너 이미지에 대한 SBOM을 생성할 수 있습니다.

특정 파일을 제외하거나 도구에서 스캔할 패키지를 정의하는 등의 옵션을 통해 SBOM 생성 결과를 사용자 지정할 수도 있습니다. 이러한 사용 사례의 예 등을 보려면 다음 명령을 실행하세요.

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

컨테이너 이미지에 대한 SBOM을 생성하고 결과를 파일로 출력하는 방법

이 예에서는 *image:tag*를 이미지의 ID로 바꾸고, *output_path.json*을 출력을 저장할 경로로 바꾸세요.

```
./inspector-sbomgen container --image image:tag -o output_path.json
```

Sbomgen을 사용하여 프라이빗 레지스트리에 인증

프라이빗 레지스트리 인증 보안 인증 정보를 제공하여 프라이빗 레지스트리에 호스팅된 컨테이너에서 SBOM을 생성할 수 있습니다. 여러 가지 방법으로 보안 인증 정보를 제공할 수 있습니다. 예를 들어 캐시된 보안 인증 정보, 대화형 방법 또는 Sbomgen 실행 전에 보안 인증 정보를 환경 변수로 제공하는 비대화형 방법을 통해 제공할 수 있습니다.

캐시된 보안 인증 정보를 사용하여 인증(권장)

1. 에이전트에서 사용할 수 있는 경우 Sbomgen은 캐시된 보안 인증 정보를 사용하려고 합니다. 이 방법을 사용하려면 먼저 컨테이너 레지스트리에 인증해야 합니다. 예를 들어 Docker를 사용하는 경우 Docker `login` 명령을 사용하여 레지스트리에 인증할 수 있습니다.

   ```
   docker login
   ```

2. 그러면 프라이빗 레지스트리 인증에 성공한 후 해당 레지스트리의 컨테이너 이미지에서 Sbomgen을 사용할 수 있습니다. 다음 예를 사용하려면 *image:tag*를 스캔할 이미지의 이름으로 바꾸세요.

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

대화형 방법을 사용하여 인증

- 이 방법에서는 사용자 이름을 파라미터로 제공하면 Sbomgen에서 필요할 때 안전한 암호 입력을 요구하는 메시지가 표시됩니다. 다음 예를 사용하려면 *image:tag*를 스캔할 이미지의 이름으로 바꾸고, *your_username*을 해당 이미지에 액세스할 수 사용자 이름으로 바꾸세요.

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

비대화형 방법을 사용하여 인증

- 이 방법을 사용하려면 현재 사용자만 읽을 수 있는.txt 파일에 암호 또는 레지스트리 토큰을 저장해야 합니다. 텍스트 파일에는 한 줄에 암호 또는 토큰만 포함해야 합니다. 다음 예를 사용하려면 *your_username*을 사용자 이름으로 바꾸고, *password.txt*를 암호 또는 토큰이 들어 있는 파일로 바꾸고, *image:tag*를 스캔할 이미지의 이름으로 바꾸세요.

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

Sbomgen의 예시 출력

다음은 Sbomgen을 사용하여 인벤토리에 추가된 컨테이너 이미지에 대한 SBOM의 예입니다.

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 Inspector 스캔을 사용하여 사용자 지정 CI/CD 파이프라인 통합 생성

CI/CD 마켓플레이스에서 Amazon Inspector CI/CD 플러그인을 사용할 수 있는 경우 해당 플러그인을 사용하는 것이 좋습니다. 사용 가능한 플러그인 목록은 지원되는 CI/CD 솔루션 섹션을 참조하세요.

Amazon Inspector에서 CI/CD 솔루션용 플러그인을 제공하지 않는 경우, Amazon Inspector SBOM 생성기와 Amazon Inspector 스캔 API를 함께 사용하여 사용자 지정 CI/CD 통합을 생성할 수 있습니다. 또한 사용자 지정 통합을 사용하여 Amazon Inspector SBOM 생성기에서 제공되는 옵션을 통해 스캔을 미세 조정할 수 있습니다.

사용자 지정 통합을 설정하는 방법

1. Amazon Inspector 스캔 API에 대한 액세스를 AWS 계정 허용하도록 구성하십시오. 지침은 Amazon Inspector CI/CD 통합을 사용하기 위한 AWS 계정 설정 섹션을 참조하십시오.
2. Amazon Inspector SBOM 생성기 바이너리를 설치하고 구성합니다. 지침은 Amazon Inspector SBOM 생성기 설치(Sbomgen) 섹션을 참조하십시오.
3. SBOM 생성기를 사용하여 스캔하려는 컨테이너 이미지에 대한 SBOM 파일을 생성합니다. 다음 예시를 사용하려면 *image:id*를 스캔할 이미지의 이름으로 바꾸고 *sbom_path.json*을 SBOM 출력을 저장할 위치로 바꾸세요.

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

4. `inspector-scan` API를 직접 호출하여 생성된 SBOM을 스캔하고 취약성 보고서를 제공합니다. 다음 예시를 사용하려면 *sbom_path.json*을 유효한 CycloneDX 호환 SBOM 파일의 파일 경로로 바꾸세요. 그런 다음 *AWS ## ENDPOINT#* 현재 인증된 API 엔드포인트로 교체하고 *###* 해당 지역으로 대체하십시오. 리전과 엔드포인트 전체 목록은 Amazon Inspector 스캔 API용 엔드포인트 섹션을 참조하세요.

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

API 출력 형식

Amazon Inspector 스캔 API는 CycloneDX 1.5 형식 또는 Amazon Inspector 조사 결과 JSON 으로 취약성 보고서를 출력할 수 있습니다. `--output-format` 플래그를 사용하여 기본값을 변경할 수 있습니다.

CycloneDX 1.5 형식 출력의 예


```
{
  "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 Jenkins 플러그인 사용

Jenkins플러그인은 Amazon Inspector SBOM Generator 바이너리와 Amazon Inspector Scan API를 활용하여 빌드가 끝날 때 상세한 보고서를 생성하므로 배포 전에 위험을 조사하고 해결할 수 있습니다.

Amazon Inspector는 CVE를 기반으로 컨테이너 이미지를 스캔하여 운영 체제 및 프로그래밍 언어 패키지 취약성을 찾아내는 취약성 관리 서비스입니다.

Amazon Inspector Jenkins 플러그인을 사용하여 파이프라인에 Amazon Inspector 취약성 스캔을 추가할 수 있습니다. Jenkins

> **Note**
>
> Amazon Inspector 취약성 스캔은 탐지된 취약성의 수와 심각도에 따라 파이프라인 실행을 통과 또는 실패하도록 구성할 수 있습니다.

마켓플레이스 https://plugins.jenkins.io/ /에서 최신 버전의 Jenkins 플러그인을 볼 수 있습니다. Jenkins amazon-inspector-image-scanner

다음 단계는 Amazon Inspector Jenkins 플러그인을 설정하는 방법을 설명합니다.

> **Important**
>
> 다음 단계를 완료하기 전에 Jenkins를 버전 2.387.3 이상으로 업그레이드해야 플러그인을 실행할 수 있습니다.

단계 1. 설정 AWS 계정

Amazon Inspector 스캔 API에 대한 액세스를 허용하는 IAM AWS 계정 역할로 구성하십시오. 지침은 Amazon Inspector CI/CD 통합을 사용하기 위한 AWS 계정 설정 섹션을 참조하십시오.

단계 2. 아마존 인스펙터 젠킨스 플러그인 설치

다음 절차는 대시보드에서 Amazon Inspector Jenkins 플러그인을 설치하는 방법을 설명합니다. Jenkins

1. Jenkins 대시보드에서 Jenkins 관리를 선택한 다음 플러그인 관리를 선택합니다.
2. [사용 가능] 을 선택합니다.
3. 사용 가능 탭에서 Amazon Inspector 스캔을 검색한 다음 플러그인을 설치합니다.

(선택 사항) 3단계. 에 docker 자격 증명 추가 Jenkins

> **Note**
>
> docker 이미지가 개인 저장소에 있는 경우에만 docker 자격 증명을 추가하십시오. 그렇지 않은 경우 이 단계를 건너뜁니다.

다음 절차는 Jenkins 대시보드에서 docker 자격 증명을 추가하는 방법을 설명합니다. Jenkins

1. Jenkins 대시보드에서 Jenkins 관리, 자격 증명, 시스템을 차례로 선택합니다.
2. 글로벌 자격 증명을 선택한 다음 자격 증명 추가를 선택합니다.
3. 종류에서 비밀번호가 있는 사용자 이름을 선택합니다.
4. 범위에서 글로벌 (Jenkins, 노드, 항목, 모든 하위 항목 등) 을 선택합니다.
5. 세부 정보를 입력한 다음 확인을 선택합니다.

(선택 사항) 4단계. AWS 자격 증명 추가

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

다음 절차는 대시보드에서 AWS 자격 증명을 추가하는 방법을 설명합니다. Jenkins

1. Jenkins 대시보드에서 Jenkins 관리, 자격 증명, 시스템을 차례로 선택합니다.

2. 글로벌 자격 증명을 선택한 다음 자격 증명 추가를 선택합니다.
3. 종류에서 AWS 자격 증명을 선택합니다.
4. 액세스 키 ID 및 보안 액세스 키를 포함한 세부 정보를 입력한 다음 OK를 선택합니다.

5단계. Jenkins스크립트에 CSS 지원 추가

다음 절차는 Jenkikns 스크립트에 CSS 지원을 추가하는 방법을 설명합니다.

1. Jenkins를 다시 시작하세요.
2. 대시보드에서 Jenkins 관리, 노드, 내장 노드, 스크립트 콘솔을 차례로 선택합니다.
3. 텍스트 상자에 줄을 `System.setProperty("hudson.model.DirectoryBrowserSupport.CSP", "")` 추가한 다음 [Run] 을 선택합니다.

6단계. 빌드에 아마존 인스펙터 스캔 추가

프로젝트에 빌드 단계를 추가하거나 Jenkins 선언적 파이프라인을 사용하여 Amazon Inspector 스캔을 빌드에 추가할 수 있습니다.

Amazon Inspector 프로젝트에 빌드 단계를 추가하여 빌드를 스캔합니다.

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

   a. (옵션 1) Inspector-sbomgen의 최신 버전을 다운로드하려면 자동을 선택합니다. 이 방법을 선택하는 경우 플러그인을 실행하는 시스템에 맞는 CPU 아키텍처를 선택해야 합니다.

   b. (옵션 2) Amazon Inspector SBOM Generator 바이너리를 스캔용으로 설정하려면 수동을 선택합니다. 이 방법을 선택하는 경우 이전에 다운로드한 버전의 inspector-sbomgen에 대한 전체 경로를 제공해야 합니다.

   자세한 내용은 Amazon Inspector SBOM Generator의 Installing Amazon Inspector SBOM Generator (Sbomge)를 참조하세요.
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 Docker 이름 지정 규칙을 따라야 합니다. 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 보안 인증 정보의 경우 Docker 사용자 이름을 선택합니다. 컨테이너 이미지가 프라이빗 리포지토리에 있는 경우에만 이 작업을 수행하세요.

d. (선택 사항) 지원되는 다음과 같은 인증 방법을 제공할 수 있습니다. AWS

   i. (선택 사항) IAM 역할의 경우 역할 ARN (arn:aws:iam: ::role/) 을 제공하십시오. *AccountNumberRoleName*

   ii. (선택 사항) AWS 자격 증명의 경우 Id를 선택하여 IAM 사용자를 기반으로 인증합니다.

   iii. (선택 사항) AWS 프로필 이름에는 프로필 이름을 사용하여 인증할 프로필 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. 저장을 선택합니다.

선언적 파이프라인을 사용하여 빌드에 Amazon Inspector Scan을 추가합니다. Jenkins

Jenkins 선언적 파이프라인을 사용하여 Amazon Inspector Scan을 빌드에 자동 또는 수동으로 추가할 수 있습니다.

sBOMgen 선언적 파이프라인을 자동으로 다운로드하려면

- Amazon Inspector Scan을 빌드에 추가하려면 다음 예제 구문을 사용하십시오. Amazon Inspector SBOM 제너레이터 다운로드의 선호하는 OS 아키텍처에 따라 SBOMGEN_SOURCE를 LinuxAMD64 또는 *LinuxARM64#* 대체하십시오. *IMAGE_PATH#* 이미지 경로 (예: *alpine:latest*) 로, IAM_ROLE은 1단계에서 구성한 IAM 역할의 *ARN##*, 프라이빗 리포지토리를 사용하는 경우 *ID#* 자격 증명 ID로 바꾸십시오. Docker 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 선언적 파이프라인을 수동으로 다운로드하려면

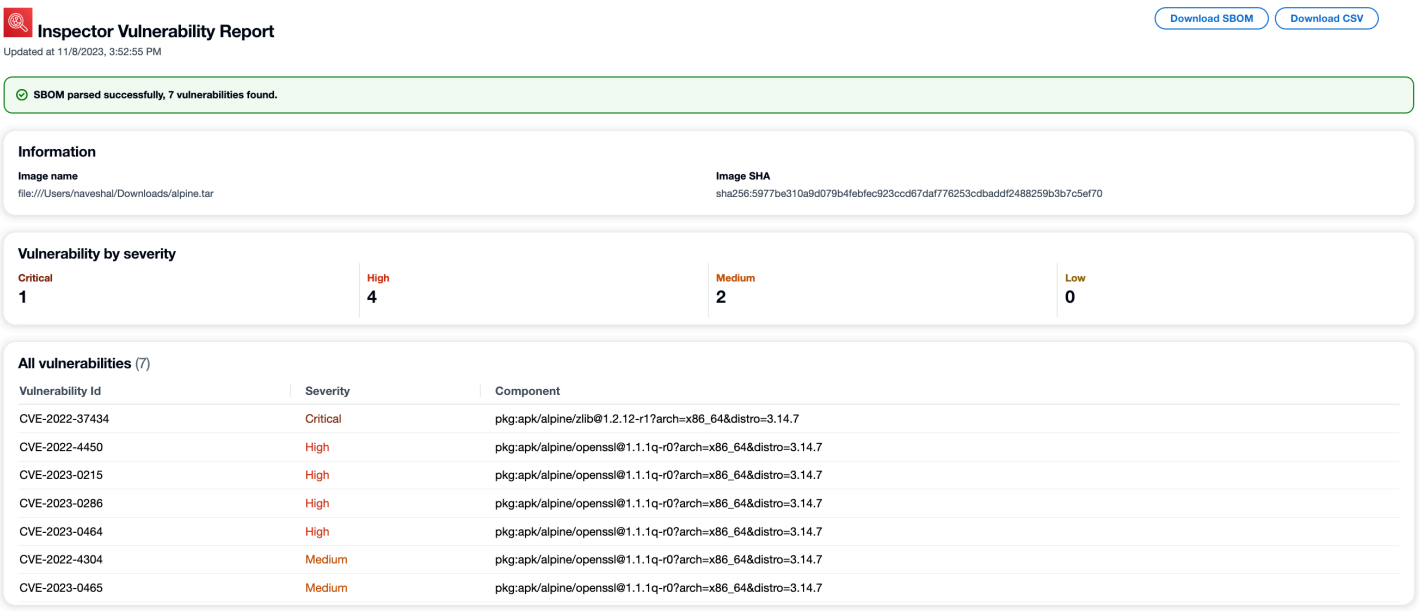
- Amazon Inspector Scan을 빌드에 추가하려면 다음 예제 구문을 사용하십시오. *SBOMGEN_PATH # 3#### ### Amazon Inspector SBOM #### ###, IMAGE_PATH# ### ## (#: alpine:latest) #, IAM_ROLE# 1#### ### IAM ### ARN##, ## ###### #### ## ID# ## ## ID# #####.* Docker 선택적으로 취약성 임계값을 활성화하고 각 심각도에 대한 값을 지정할 수 있습니다.

> **Note**
>
> 젠킨스 Sbomgen 디렉터리에 넣고 플러그인의 젠킨스 디렉터리 경로를 입력합니다 (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. HTML을 선택하면 보고서의 JSON SBOM 또는 CSV 버전을 다운로드할 수 있습니다. 다음은 HTML 보고서의 예를 보여줍니다.


Inspector Vulnerability Report

Updated at 11/8/2023, 3:52:55 PM

Download SBOM Download CSV

SBOM parsed successfully, 7 vulnerabilities found.

Information

Image name
file:///Users/naveshal/Downloads/alpine.tar

Image SHA
sha256:5977be310a9d079b4febfec923ccd67daf776253cdbaddf2488259b3b7c5ef70

Vulnerability by severity

Critical	High	Medium	Low
1	4	2	0

All vulnerabilities (7)

Vulnerability Id	Severity	Component
CVE-2022-37434	Critical	pkg:apk/alpine/zlib@1.2.12-r1?arch=x86_64&distro=3.14.7
CVE-2022-4450	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0215	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0286	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0464	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2022-4304	Medium	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0465	Medium	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7

문제 해결

다음은 Amazon Inspector Scan 플러그인을 사용할 때 발생할 수 있는 일반적인 오류입니다. Jenkins

자격 증명을 로드하지 못했거나 sts 예외 오류가 발생했습니다.

오류:

```
InstanceProfileCredentialsProvider(): Failed to load credentials or sts exception.
```

해결:

계정을 `aws_access_key_id` `aws_secret_access_key` 구매하세요. AWS `aws_access_key_id` 및 `aws_secret_access_key`를 `~/.aws/credentials`에 설정합니다.

인스펙터-스봄겐 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 해결하십시오.

1. Jenkins디렉터리에 올바른 OS 아키텍처 인스펙터-SBOMgen을 배치하십시오. 자세한 내용은 Amazon Inspector SBOM 생성기를 참조하십시오.
2. 다음 명령을 사용하여 바이너리에 실행 권한을 부여합니다. `chmod +x inspector-sbomgen`
3. 플러그인에 올바른 Jenkins 컴퓨터 경로 (예/`opt/folder/arm64/inspector-sbomgen`:) 를 제공하십시오.
4. 구성을 저장하고 Jenkins 작업을 실행합니다.

Amazon Inspector TeamCity 플러그인 사용

Amazon Inspector TeamCity 플러그인을 사용하면 TeamCity 파이프라인에 Amazon Inspector 취약성 스캔을 추가할 수 있습니다. 이 플러그인은 Amazon Inspector SBOM 생성기 바이너리와 Amazon Inspector 스캔 API를 활용하여 빌드가 끝날 때 상세한 보고서를 생성하므로 배포 전에 위험을 조사하고 해결할 수 있습니다. 또한 탐지된 취약성의 수와 심각도에 따라 파이프라인 실행에 성공하거나 실패하도록 스캔을 구성할 수 있습니다.

Amazon Inspector는 CVE를 기반으로 컨테이너 이미지를 스캔하여 운영 체제 및 프로그래밍 언어 패키지 취약성을 모두 AWS 찾아내는 취약성 관리 서비스입니다. Amazon Inspector CI/CD 통합에 대한 자세한 내용은 Amazon Inspector 스캔을 CI/CD 파이프라인에 통합 섹션을 참조하세요.

Amazon Inspector 플러그인이 지원하는 패키지 및 컨테이너 이미지 형식 목록은 지원되는 패키지 및 이미지 형식 섹션을 참조하세요.

마켓플레이스의 https://plugins.jetbrains.com/plugin/23236 -에서 최신 버전의 플러그인을 볼 수 있습니다. TeamCity amazon-inspector-scanner 또는 이 문서의 각 섹션에 있는 단계에 따라 Amazon Inspector TeamCity 플러그인을 설정하세요.

1. 설정하세요 AWS 계정.
   - Amazon Inspector 스캔 API에 대한 액세스를 허용하는 IAM AWS 계정 역할로 구성하십시오. 지침은 Amazon Inspector CI/CD 통합을 사용하기 위한 AWS 계정 설정 섹션을 참조하십시오.
2. Amazon Inspector TeamCity 플러그인을 설치합니다.
   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 생성기 바이너리를 설치합니다. 지침은 Amazon Inspector SBOM 생성기 설치(Sbomgen) 섹션을 참조하십시오.

4. Amazon Inspector 스캔 빌드 단계를 프로젝트에 추가합니다.

   a. 구성 페이지에서 빌드 단계까지 아래로 스크롤하고 빌드 단계 추가를 선택한 다음 Amazon Inspector Scan을 선택합니다.

   b. 다음 세부 정보를 입력하여 Amazon Inspector 스캔 빌드 단계를 구성합니다.

      - 단계 이름을 추가합니다.
      - 두 가지 Amazon Inspector SBOM 생성기 설치 방법 (자동 또는 수동) 중에서 선택하십시오.
        - 시스템 및 CPU 아키텍처를 기반으로 최신 버전의 Amazon Inspector SBOM 생성기를 자동으로 다운로드합니다.
        - 매뉴얼을 사용하려면 이전에 다운로드한 Amazon Inspector SBOM Generator 버전의 전체 경로를 제공해야 합니다.

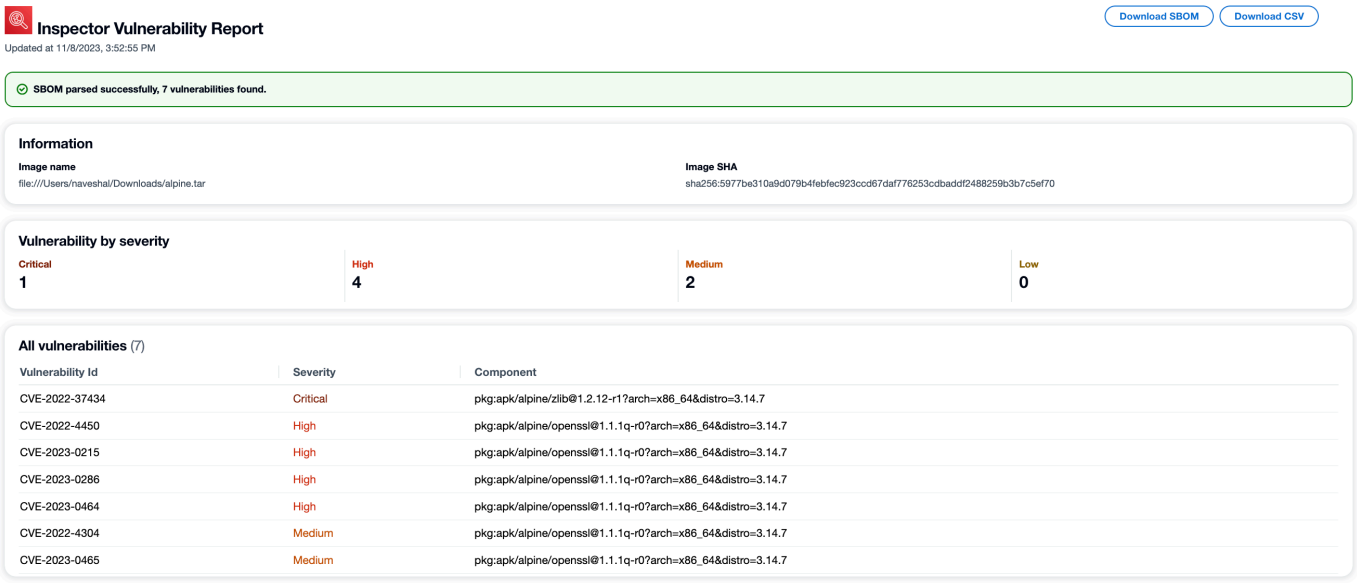
        자세한 내용은 Amazon Inspector SBOM 생성기에 Amazon Inspector SBOM 생성기 (Sbomgen) 설치를 참조하십시오.
      - 이미지 ID를 입력합니다. 이미지는 로컬, 원격 또는 아카이브 위치에 있을 수 있습니다. 이미지 이름은 Docker 이름 지정 규칙을 따라야 합니다. 내보낸 이미지를 분석하는 경우 예상 tar 파일의 경로를 제공하세요. 다음 이미지를 예시 이미지 ID 경로로 참조하세요.
        - 로컬 또는 원격 컨테이너의 경우: `NAME[:TAG|@DIGEST]`
        - tar 파일의 경우: `/path/to/image.tar`
      - IAM 역할의 경우 1단계에서 구성한 역할의 ARN을 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 인증을 위해 액세스 키 ID와 비밀 키를 입력합니다. AWS AWS AWS 자격 증명을 기반으로 인증하려는 경우에만 이 작업을 수행하십시오.
      - (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. HTML을 선택하면 보고서의 JSON SBOM 또는 CSV 버전을 다운로드할 수 있습니다. 다음은 HTML 보고서의 예입니다.



Inspector Vulnerability Report

Updated at 11/8/2023, 3:52:55 PM

Download SBOM Download CSV

SBOM parsed successfully, 7 vulnerabilities found.

Information

Image name	Image SHA
file:///Users/naveshal/Downloads/alpine.tar	sha256:5977be310a9d079b4febfec923ccd67daf776253cdbaddf2488259b3b7c5ef70

Vulnerability by severity

Critical	High	Medium	Low
1	4	2	0

All vulnerabilities (7)

Vulnerability Id	Severity	Component
CVE-2022-37434	Critical	pkg:apk/alpine/zlib@1.2.12-r1?arch=x86_64&distro=3.14.7
CVE-2022-4450	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0215	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0286	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0464	High	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2022-4304	Medium	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7
CVE-2023-0465	Medium	pkg:apk/alpine/openssl@1.1.1q-r0?arch=x86_64&distro=3.14.7

Amazon Inspector CycloneDX 네임스페이스

Amazon Inspector에는 Amazon Inspector SBOM 생성기 및 Amazon Inspector 스캔 API에서 생성한 SBOM에 사용할 CycloneDX 네임스페이스와 속성 이름이 예약되어 있습니다. 이 페이지는 Amazon Inspector 도구를 사용하여 생성한 CycloneDX SBOM의 구성 요소에 추가할 수 있는 모든 사용자 지정 키/값 속성이 문서화됩니다. CycloneDX 속성 분류법에 대한 자세한 내용은 공식 설명서를 참조하세요.

`amazon:inspector:sbom_scanner` 네임스페이스 분류법

`amazon:inspector:sbom_scanner` 네임스페이스는 Amazon Inspector 스캔 API에서 사용됩니다. 여기에는 다음과 같은 속성이 있습니다.

속성	설명
`amazon:inspector:sbom_scanner:critical_vulnerabilities`	SBOM에서 발견된 심각한 심각도 취약성의 총 개수.
`amazon:inspector:sbom_scanner:high_vulnerabilities`	SBOM에서 발견된 높은 심각도 취약성의 총 개수.

속성	설명
amazon:inspector:sbom_scanner:medium_vulnerabilities	SBOM에서 발견된 중간 심각도 취약성의 총 개수.
amazon:inspector:sbom_scanner:low_vulnerabilities	SBOM에서 발견된 낮은 심각도 취약성의 총 개수.
amazon:inspector:sbom_scanner:info	특정 구성 요소에 대한 스캔 컨텍스트를 제공합니다(예: '구성 요소 검사됨: 발견된 취약성 없음').
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_scanner:fixed_version: *component_bom_ref*	지정된 취약성에 대해 표시된 구성 요소의 수정된 버전을 제공합니다.
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:cisa_kev_date_added	취약성이 CISA의 알려진 악용된 취약성 카탈로그에 추가된 시기를 나타냅니다.
amazon:inspector:sbom_scanner:cisa_kev_date_due	CISA의 알려진 악용된 취약성 카탈로그에 따라 취약성 수정이 마감되는 시기를 나타냅니다.
amazon:inspector:sbom_scanner:path	주제 패키지 정보를 생성한 파일의 경로.

amazon:inspector:sbom_generator 네임스페이스 분류법

amazon:inspector:sbom_generator 네임스페이스는 Amazon Inspector SBOM 생성기에서 사용됩니다. 여기에는 다음과 같은 속성이 있습니다.

속성	설명
amazon:inspector:sbom_generator:os_hostname	인벤토리 대상 시스템의 호스트 이름.
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:cpu_architecture	인벤토리 대상 시스템의 CPU 아키텍처(예: x86_64).
amazon:inspector:sbom_generator:image_id	컨테이너 이미지 구성 파일의 해시(이미지 ID라고도 함).
amazon:inspector:sbom_generator:layer_diff_id	압축되지 않은 컨테이너 이미지 계층의 해시.
amazon:inspector:sbom_generator:source_file_scanner	패키지 정보가 들어 있는 파일을 찾은 스캐너(예: /var/lib/dpkg/status).
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.
amazon:inspector:sbom_generator:is_duplicate_package	두 개 이상의 파일 스캐너에서 주제 패키지를 찾았음을 나타냅니다.
amazon:inspector:sbom_generator:go_toolchain	Go 실행 파일을 생성하는 데 사용된 Go 컴파일러 또는 툴체인 버전을 나타냅니다.

속성	설명
amazon:inspector:sbom_generator:expires_before	이 날짜 전의 인증서는 유효하지 않습니다.
amazon:inspector:sbom_generator:expires_after	이 날짜 후의 인증서는 유효하지 않습니다.
amazon:inspector:sbom_generator:is_expired	SSL 인증서가 만료되었는지를 나타내는 부울 값입니다.

Amazon Inspector를 사용한 자동 리소스 스캔

Amazon EC2용 Amazon Inspector 에이전트 없는 스캔은 미리 보기로 출시되었습니다. 에이전트 없는 Amazon EC2 스캔 기능 사용에는 AWS 서비스 약관 섹션 2('베타 및 미리 보기')가 적용됩니다.

Amazon Inspector는 특별히 제작된 자체 스캔 엔진을 사용합니다. 이 엔진은 리소스를 모니터링하여 워크로드 손상, 리소스 악용 또는 데이터에 대한 무단 액세스를 초래할 수 있는 소프트웨어 취약성 또는 오픈 네트워크 경로를 찾아냅니다. Amazon Inspector에서 취약성을 탐지하면 결과가 생성됩니다. 결과에는 취약성을 해결하는 데 도움이 되는 탐지와 관련된 세부 정보가 포함되어 있습니다. Amazon Inspector 콘솔 또는 Amazon Inspector API를 사용하여 결과를 검토할 수 있습니다. 자세한 설명은 Amazon Inspector에서 결과 관리 섹션을 참조하세요.

Amazon Inspector가 활성화된 경우 적합한 모든 리소스가 자동으로 검색되고 해당 리소스에 대한 연속 스캔이 시작됩니다. Amazon Inspector는 소프트웨어 취약성과 의도하지 않은 네트워크 노출이 있는지 스캔합니다. 또한 Amazon Inspector는 새 애플리케이션 또는 패치 설치와 같은 이벤트에 대한 응답으로 스캔을 실행합니다.

Amazon Inspector를 처음 활성화하면 계정이 모든 스캔 유형에 자동으로 등록됩니다. 다음 주제에서는 Amazon Inspector에서 제공하는 스캔 유형에 대한 구체적인 내용을 다룹니다. Amazon Inspector는 취약성의 영향을 받는 리소스 유형을 기준으로 스캔 유형을 분류합니다. 다음 주제에서는 Amazon Inspector에서 스캔하는 리소스, 해당 리소스에 대한 새 스캔을 시작하는 대상, 각 리소스 유형에 대한 스캔을 구성하는 방법을 다룹니다.

주제

- Amazon Inspector 스캔 유형 개요
- 스캔 유형 활성화
- Amazon Inspector로 Amazon EC2 인스턴스 스캔
- Amazon Inspector로 Amazon ECR 컨테이너 이미지 스캔
- Amazon AWS Lambda Inspector를 사용한 스캔 기능
- 스캔 유형 비활성화

Amazon Inspector를 처음 활성화하면 계정이 Amazon EC2 스캔, Amazon ECR 스캔, Lambda 표준 스캔 등의 스캔 유형에 자동으로 등록됩니다. Lambda 코드 스캔은 Lambda 함수 스캔의 선택적 계층으로, 언제든지 활성화할 수 있습니다.

Amazon Inspector 스캔 유형 개요

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

Amazon EC2 스캔

Amazon EC2 스캔을 활성화하면 Amazon Inspector에서 Amazon EC2 인스턴스를 스캔하여 운영 체제 패키지와 프로그래밍 언어 패키지 취약성, 네트워크 연결성을 검사합니다. Amazon Inspector는 EC2 인스턴스에서 일반적인 취약성 및 노출(CVE)과 네트워크 노출 문제를 스캔합니다. Amazon Inspector는 인스턴스에 설치된 SSM 에이전트를 사용하거나 인스턴스의 Amazon EBS 스냅샷을 통해 스캔을 수행합니다. Amazon EC2 스캔에 대한 자세한 내용은 Amazon Inspector로 Amazon EC2 인스턴스 스캔 섹션을 참조하세요.

Amazon ECR 스캔

Amazon ECR 스캔을 활성화하면 Amazon Inspector는 프라이빗 레지스트리의 모든 기본 스캔 컨테이너 리포지토리를 연속 스캔이 가능한 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 이미지가 처음에 스캔됩니다. Amazon Inspector는 기본적으로 90일 동안 이미지를 계속 모니터링합니다. 이 설정은 언제든지 변경할 수 있습니다. Amazon ECR 스캔에 대한 자세한 내용은 Amazon Inspector로 Amazon ECR 컨테이너 이미지 스캔 섹션을 참조하세요.

Lambda 표준 스캔

Lambda 표준 스캔을 활성화하면 Amazon Inspector는 계정에서 Lambda 함수를 검색하고 즉시 취약성 스캔을 시작합니다. Amazon Inspector는 새로운 Lambda 함수와 계층이 배포될 때 스캔을 수행하며, 해당 함수와 계층이 업데이트되거나 새로운 일반 취약성 및 노출(CVE)이 발표될 때 재스캔을 수행합니다. Lambda 함수 스캔에 대한 자세한 내용은 Amazon AWS Lambda Inspector를 사용한 스캔 기능 섹션을 참조하세요.

Lambda 표준 스캔 + Lambda 코드 스캔

이 옵션은 Lambda 표준 스캔과 Lambda 코드 스캔을 결합한 것입니다. 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

새 Amazon Inspector 스캔 유형은 언제든지 활성화할 수 있습니다. 스캔 유형을 활성화하면 Amazon Inspector에서 해당 스캔 유형에 적합한 리소스를 즉시 스캔하기 시작합니다. 사용 가능한 스캔 유형에 대한 개요는 Amazon Inspector 스캔 유형 개요 섹션을 참조하세요. 각 스캔 유형을 처음 활성화하면 다음과 같은 과정이 진행됩니다.

- Amazon EC2 스캔 - 계정에 대해 Amazon Inspector Amazon EC2 스캔을 활성화하면 Amazon Inspector에서 계정의 적합한 모든 인스턴스를 스캔하여 패키지 취약성 및 네트워크 연결성 문제를 찾아냅니다. Amazon Inspector SSM 플러그인은 모든 SSM 관리 호스트에 설치됩니다. Windows 자세한 설명은 Windows 인스턴스 스캔 섹션을 참조하세요. 또한 Amazon Inspector는 계정에 다음과 같은 SSM 연결을 생성합니다.
  - `InspectorDistributor-do-not-delete`
  - `InspectorInventoryCollection-do-not-delete`
  - `InspectorLinuxDistributor-do-not-delete`
  - `InvokeInspectorLinuxSsmPlugin-do-not-delete`
  - `InvokeInspectorSsmPlugin-do-not-delete`.
- Amazon ECR 스캔 - 계정에 대해 Amazon ECR 컨테이너 이미지 스캔을 활성화하면 해당 계정의 프라이빗 리포지토리에 대한 Amazon ECR 스캔 유형이 Amazon ECR을 사용하는 기본 스캔에서 Amazon Inspector를 사용하는 고급 스캔으로 변경됩니다. 그런 다음 지난 30일 이내에 푸시되거나 지난 90일 이내에 가져온 모든 적격 Amazon ECR 컨테이너 이미지에 대해 패키지 취약성이 있는지 스캔합니다. 또한 이미지 푸시 및 풀 날짜의 Amazon ECR 재스캔 기간은 90일로 설정되어 있습니다.
- Lambda 표준 스캔 - 계정에서 Lambda 표준 스캔을 활성화하면 지난 90일 동안 호출되거나 업데이트된 계정의 모든 Lambda 함수를 스캔하여 패키지 취약성이 있는지 검사합니다. 또한 CloudTrail 서비스 연결 채널이 계정에 생성됩니다.
- Lambda 표준 스캔 + Lambda 코드 스캔 - 이 Lambda 함수 스캔 유형은 함께 활성화됩니다. 계정에서 Lambda 코드 스캔을 활성화하면 지난 90일 동안 호출되거나 업데이트된 계정의 모든 Lambda 함수를 스캔하여 코드 취약성이 있는지 검사합니다.

스캔 활성화

조직의 Amazon Inspector의 위임 관리자인 경우 Amazon Inspector inspector2-에서 AWS 개발한 셸 스크립트를 사용하여 여러 지역의 여러 계정에 대해 다양한 Amazon Inspector 스캔 유형을 자동으로 활성화할 수 있습니다. enablement-with-cli GitHub 그렇지 않은 경우 콘솔을 통해 다중 계정 환경에 대

해 이 절차를 완료하려면 Amazon Inspector 위임 관리자로 로그인한 상태에서 다음 단계를 완료하세요.

Console

스캔을 활성화하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 새 스캔 유형을 활성화하려는 지역을 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 활성화하려는 각 AWS 리전 항목에서 이 단계를 반복합니다.

API

Enable API 작업을 실행합니다. 요청에 스캔을 활성화할 계정 ID를 제공하고, 멱등성 토큰 및 `resourceTypes`로 `EC2`, `ECR`, `LAMBDA`, `LAMBDA_CODE` 중 하나 이상을 제공하면 해당 유형의 스캔이 활성화됩니다.

Amazon Inspector로 Amazon EC2 인스턴스 스캔

Amazon EC2용 Amazon Inspector 에이전트 없는 스캔은 미리 보기로 출시되었습니다. 에이전트 없는 Amazon EC2 스캔 기능 사용에는 AWS 서비스 약관 섹션 2('베타 및 미리 보기')가 적용됩니다.

Amazon Inspector EC2 스캔은 EC2 인스턴스에서 메타데이터를 추출한 다음, 이 메타데이터를 보안 권고에서 수집한 규칙과 비교하여 조사 결과를 생성합니다. Amazon Inspector는 인스턴스에서 패키지 취약성과 네트워크 연결 문제를 스캔합니다. 이러한 문제에 대해 생성되는 결과 유형에 대한 자세한 내용은 Amazon Inspector의 결과 유형 섹션을 참조하세요.

Amazon Inspector는 24시간마다 한 번씩 네트워크 연결성 스캔을 수행하는 반면, 패키지 취약성 스캔은 인스턴스와 관련된 스캔 방법에 따라 다양한 빈도로 수행됩니다.

스캔 방법

패키지 취약성 스캔은 에이전트 기반 또는 에이전트 없음 스캔 방법을 사용하여 수행할 수 있습니다. 이러한 스캔 방법은 Amazon Inspector가 패키지 취약성 스캔을 위해 EC2 인스턴스에서 소프트웨어 인벤토리를 수집하는 방법과 시기를 결정합니다. '에이전트 기반' 방법은 SSM 에이전트를 사용하여 소프트웨어 인벤토리를 수집하는 반면, '에이전트 없음' 방법은 에이전트 대신 Amazon EBS 스냅샷을 사용합니다.

Amazon Inspector에서 사용하는 스캔 방법은 계정의 스캔 모드 설정에 따라 달라집니다. 자세한 내용은 스캔 모드 관리 섹션을 참조하세요.

Amazon EC2 스캔을 활성화하려면 스캔 유형 활성화 섹션을 참조하세요.

에이전트 기반 스캔

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

다음 프로세스는 Amazon Inspector에서 SSM을 사용하여 인벤토리를 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 Inspector는 SSM을 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 “Amazon EC2 스캔을 지원하는 운영 체제”의 에이전트 기반 스캔 지원 열을 참조하세요.
- Amazon Inspector EC2 제외 태그로 인해 스캔에서 인스턴스가 제외되지 않습니다.

- 인스턴스가 SSM 관리형입니다. 에이전트를 확인하고 구성하는 방법은 SSM 에이전트 구성 섹션을 참조하세요.

에이전트 기반 스캔 동작

에이전트 기반 스캔 방법을 사용할 때 Amazon Inspector는 다음과 같은 경우에 EC2 인스턴스에 대한 새로운 취약성 스캔을 시작합니다.

- 새 EC2 인스턴스를 시작하는 경우
- 기존 EC2 인스턴스에 새 소프트웨어를 설치하는 경우(Linux 및 Mac)
- Amazon Inspector가 새로운 일반적인 취약성 및 노출(CVE) 항목을 데이터베이스에 추가하고, 해당 CVE가 EC2 인스턴스와 관련이 있는 경우(Linux 및 Mac)

Amazon Inspector는 초기 스캔이 완료되면 EC2 인스턴스의 마지막 스캔 필드를 업데이트합니다. 이후 Amazon Inspector가 SSM 인벤토리를 평가할 때(기본적으로 30분마다) 또는 해당 인스턴스에 영향을 미치는 새 CVE가 Amazon Inspector 데이터베이스에 추가되어 인스턴스를 다시 스캔할 때 마지막 스캔 필드가 업데이트됩니다.

EC2 인스턴스의 취약성을 마지막으로 스캔한 시기는 계정 관리 페이지의 인스턴스 탭에서 또는 ListCoverage 명령을 사용하여 확인할 수 있습니다.

SSM 에이전트 구성

Amazon Inspector에서 에이전트 기반 스캔 방법을 사용하여 Amazon EC2 인스턴스의 소프트웨어 취약성을 탐지하려면 해당 인스턴스가 Amazon EC2 Systems Manager(SSM)의 관리형 인스턴스여야 합니다. SSM 관리형 인스턴스에는 SSM 에이전트가 설치되어 실행 중이며 SSM에는 인스턴스 관리 권한이 있습니다. 이미 SSM을 사용하여 인스턴스를 관리하고 있는 경우 에이전트 기반 스캔을 위해 다른 단계가 필요하지 않습니다.

SSM 에이전트는 일부 Amazon Machine Images(AMI)에서 생성된 EC2 인스턴스에 기본적으로 설치됩니다. 자세한 내용은 AWS Systems Manager 사용 설명서에서 SSM 에이전트 정보를 참조하세요. 하지만 설치되어 있더라도 SSM 에이전트를 수동으로 활성화하고 SSM에 인스턴스 관리 권한을 부여해야 할 수 있습니다.

다음 절차에서는 IAM 인스턴스 프로파일을 사용하여 Amazon EC2 인스턴스를 관리형 인스턴스로 구성하는 방법에 대해 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 포함되어 있습니다.

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

Amazon EC2 인스턴스에 대해 SSM을 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 확인할 수 있습니다. 자세한 내용은 SSM 에이전트 상태 확인 및 에이전트 시작을 참조하세요.
3. SSM에 인스턴스 관리 권한을 부여합니다. IAM 인스턴스 프로파일을 생성한 후 이를 인스턴스에 연결하여 권한을 부여할 수 있습니다. Amazon Inspector에서 스캔하는 데 필요한 SSM Distributor, SSM Inventory 및 SSM State Manager에 대한 권한이 포함되어 있는 AmazonSSMManagedInstanceCore 정책을 사용하는 것이 좋습니다. 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 에이전트 업데이트 자동화를 참조하세요.
5. (선택 사항) Amazon Virtual Private Cloud(Amazon VPC) 엔드포인트를 사용하도록 Systems Manager를 구성합니다. 자세한 내용은 Amazon VPC 엔드포인트 생성을 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 SSM

엔드포인트의 최대 리소스 데이터 동기화 수 (AWS 계정 지역당) 를 참조하십시오. 이 최대값에 도달한 경우 리소스 데이터 동기화를 삭제해야 합니다(리소스 데이터 동기화 관리 참조).

스캔을 위해 생성된 SSM 리소스

Amazon Inspector에서 Amazon EC2 스캔을 실행하려면 계정에 여러 SSM 리소스가 필요합니다. Amazon Inspector EC2 스캔을 처음 활성화하면 다음과 같은 리소스가 생성됩니다.

> **Note**
>
> Amazon Inspector Amazon EC2 스캐닝이 사용자 계정에 대해 활성화되어 있는 동안 이러한 SSM 리소스가 하나라도 삭제되면 Amazon Inspector는 다음 스캔 간격으로 해당 리소스를 다시 생성하려고 시도합니다.

`InspectorInventoryCollection-do-not-delete`

Amazon Inspector가 Amazon EC2 인스턴스에서 소프트웨어 애플리케이션 인벤토리를 수집할 때 사용하는 Systems Manager State Manager(SSM) 연결입니다. `InstanceIds*`에서 인벤토리를 수집하기 위한 SSM 연결이 계정에 이미 있는 경우 Amazon Inspector에서 자체적으로 생성하는 대신 해당 SSM 연결이 사용됩니다.

`InspectorResourceDataSync-do-not-delete`

Amazon Inspector가 Amazon EC2 인스턴스에서 수집된 인벤토리 데이터를 Amazon Inspector가 소유한 Amazon S3 버킷으로 보낼 때 사용하는 리소스 데이터 동기화입니다. 자세한 내용은 AWS Systems Manager 사용 설명서에서 인벤토리의 리소스 데이터 동기화 구성을 참조하세요.

`InspectorDistributor-do-not-delete`

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

`InvokeInspectorSsmPlugin-do-not-delete`

Amazon Inspector에서 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 설명은 Windows 인스턴스 스캔을 위한 사용자 지정 일정 설정 섹션을 참조하세요.

`InspectorLinuxDistributor-do-not-delete`

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

`InvokeInspectorLinuxSsmPlugin-do-not-delete`

이는 Amazon Inspector가 아마존 EC2 리눅스 심층 검사를 위해 사용하는 SSM 연결입니다. 이 연결을 통해 Amazon Inspector에서 플러그인을 사용하여 스캔을 시작할 수 있습니다.

> **Note**
>
> Amazon Inspector Amazon EC2 스캐닝 또는 딥 인스펙션을 비활성화하면 모든 SSM 리소스가 해당 Linux 호스트에서 자동으로 제거됩니다.

에이전트 없는 스캔

Amazon Inspector는 계정이 하이브리드 스캔 모드(에이전트 기반 스캔과 에이전트 없는 스캔 모두 포함)인 경우 적격 인스턴스에서 에이전트 없는 스캔 방법을 사용합니다. 에이전트 없는 스캔의 경우 Amazon Inspector는 EBS 스냅샷을 사용하여 인스턴스에서 소프트웨어 인벤토리를 수집합니다. 에이전트 없는 스캔 방법을 사용하여 스캔한 인스턴스는 운영 체제 패키지와 애플리케이션 프로그래밍 언어 패키지 취약성이 모두 스캔됩니다.

> **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 Amazon EC2 Linux 인스턴스에 대한 Amazon Inspector 심층 검사의 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 Inspector에서 EBS 스냅샷을 사용하여 인벤토리를 수집하고 에이전트 없는 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 다이렉트 API를 사용하여 스냅샷에서 데이터를 검색하고 취약성이 있는지 평가합니다. 탐지된 모든 취약성에 대한 조사 결과가 생성됩니다.
3. Amazon Inspector는 계정에 생성한 EBS 스냅샷을 삭제합니다.

적격 인스턴스

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

- 인스턴스에는 지원되는 OS가 있습니다. 지원되는 OS 목록은 the section called “Amazon EC2 스캔을 지원하는 운영 체제”의 에이전트 기반 스캔 지원 열을 참조하세요.
- Amazon Inspector EC2 제외 태그로 인해 스캔에서 인스턴스가 제외되지 않습니다.
- 인스턴스의 상태는, 또는 입니다. `Unmanaged EC2 instance` `Stale inventory` `No inventory`
- 인스턴스는 EBS 기반 인스턴스이며 다음 파일 시스템 형식 중 하나를 사용합니다.
  - `ext3`
  - `ext4`
  - `xfs`

에이전트 없는 스캔 동작

계정이 하이브리드 스캔을 사용하도록 구성된 경우 Amazon Inspector는 24시간마다 적격 인스턴스에 대해 에이전트 없는 스캔을 수행합니다. Amazon Inspector는 새로운 적격 인스턴스를 매시간 탐지하고 스캔하며, 여기에는 SSM 에이전트가 없는 새 인스턴스 또는 상태가 `SSM_UNMANAGED`로 변경된 기존 인스턴스가 포함됩니다.

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

EC2 인스턴스의 취약성을 마지막으로 스캔한 시기는 계정 관리 페이지의 인스턴스 탭에서 또는 ListCoverage 명령을 사용하여 확인할 수 있습니다.

스캔 모드 관리

EC2 스캔 모드는 Amazon Inspector가 계정에서 EC2 스캔을 수행할 때 사용할 스캔 방법을 결정합니다. 일반 설정의 EC2 스캔 설정 페이지에서 계정의 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 새로운 CVE 또는 인스턴스 변경에 대한 응답으로 연속 스캔을 제공한다는 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 방법을 사용합니다. SSM 관리형이 아닌 적격 인스턴스의 경우 Amazon Inspector는 EBS 지원 적격 인스턴스에 에이전트 없는 스캔 방법을 사용합니다.

스캔 모드를 변경하는 방법

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 EC2 스캔 모드를 변경할 지역을 선택합니다.
3. 측면 탐색 패널의 일반 설정에서 EC2 스캔 설정을 선택합니다.
4. 스캔 모드에서 편집을 선택합니다.
5. 스캔 모드를 선택한 다음 변경 사항 저장을 선택합니다.

Amazon Inspector 스캔에서 인스턴스 제외

특정 인스턴스에 태그를 지정하여 Amazon Inspector 스캔에서 제외할 수 있습니다. 인스턴스를 스캔에서 제외하면 조치할 수 없는 알림을 방지하는 데 도움이 됩니다. 제외된 인스턴스에 대해서는 요금이 부과되지 않습니다.

EC2 인스턴스를 스캔에서 제외하려면 다음 키를 사용하여 해당 인스턴스에 태그를 지정하세요.

- `InspectorEc2Exclusion`

값은 선택 사항입니다.

태그 추가에 대한 자세한 내용은 Amazon EC2 리소스 태깅을 참조하세요.

또한 볼륨을 암호화하는 데 사용된 AWS KMS 키에 태그를 지정하여 에이전트 없는 스캔에서 암호화된 EBS 볼륨을 제외할 수 있습니다. `InspectorEc2Exclusion` 자세한 내용은 키 태그 지정을 참조하세요.

지원되는 운영 체제

Amazon Inspector는 지원되는 Mac, Windows, Linux EC2 인스턴스를 스캔하여 운영 체제 패키지의 취약성을 검사합니다. Linux 인스턴스의 경우 Amazon Inspector는 Amazon EC2 Linux 인스턴스에 대한 Amazon Inspector 심층 검사를 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 비롯하여 지원되는 운영 체제에 대한 자세한 내용은 Amazon EC2 스캔을 지원하는 운영 체제 섹션을 참조하세요.

Amazon EC2 Linux 인스턴스에 대한 Amazon Inspector 심층 검사

Amazon Inspector는 심층 검사를 포함하도록 Amazon EC2 스캔 범위를 확장했습니다. Amazon Inspector는 심층 검사를 통해 Linux 기반 Amazon EC2 인스턴스의 애플리케이션 프로그래밍 언어 패키지의 패키지 취약성을 탐지합니다.

Amazon Inspector는 프로그래밍 언어 패키지 라이브러리의 기본 경로를 스캔합니다. 기본 경로 외에 사용자 지정 경로를 구성할 수도 있습니다. 자세한 설명은 Amazon Inspector 심층 검사를 위한 사용자 지정 경로 섹션을 참조하세요.

Amazon Inspector는 Amazon Inspector SSM 플러그인으로 수집한 데이터를 사용하여 심층 검사 스캔을 수행합니다. 플러그인을 관리하고 Linux에 대한 심층 검사를 수행하기 위해 Amazon Inspector는 사용자 계정에 다음과 같은 SSM 연결을 `InvokeInspectorLinuxSsmPlugin-do-not-delete` 자동으로 생성합니다. 이는 Amazon Inspector가 심층 검사를 활성화할 때 발생합니다.

Amazon Inspector는 6시간마다 심층 검사를 위해 인스턴스로부터 업데이트된 애플리케이션 인벤토리를 수집합니다.

심층 검사를 위해 Amazon Inspector에서 지원하는 프로그래밍 언어 목록은 지원되는 프로그래밍 언어: Amazon EC2 심층 검사 섹션을 참조하세요.

> **Note**
>
> Windows 또는 Mac 인스턴스의 경우 심층 검사가 지원되지 않습니다.

심층 검사 활성화 또는 비활성화

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

계정에 대한 심층 검사가 활성화되어 있는지는 Amazon Inspector 콘솔의 계정 관리 페이지에 있는 Amazon EC2 스캔 열에서 확인할 수 있습니다. 심층 검사가 활성화되어 있지 않은 경우 이 열에 활성화됨(심층 검사가 비활성화됨)이라고 표시됩니다. 활성화 상태를 프로그래밍 방식으로 확인하려면 GetEc2DeepInspectionConfiguration API를 사용하세요. 여러 계정의 경우 BatchGetMemberEc2DeepInspectionStatus API를 사용하세요.

2023년 4월 17일 이전에 Amazon Inspector를 활성화한 경우 콘솔 배너 또는 UpdateEc2DeepInspectionConfiguration API를 통해 심층 검사를 활성화할 수 있습니다. Amazon Inspector에서 조직의 위임 관리자인 경우 BatchUpdateMemberEc2DeepInspectionStatus API를 사용하여 자신과 멤버 계정에 대해 심층 검사를 활성화할 수 있습니다.

UpdateEc2DeepInspectionConfiguration API를 통해 심층 검사를 비활성화할 수 있습니다. 조직의 멤버 계정으로는 심층 검사를 비활성화할 수 없습니다. 대신 위임 관리자가 BatchUpdateMemberEc2DeepInspectionStatus API를 사용하여 멤버 계정을 비활성화해야 합니다.

Linux용 Amazon Inspector SSM 플러그인 정보

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

> **Note**
>
> Amazon Inspector는 Systems Manager Distributor를 사용하여 Amazon EC2 인스턴스에 플러그인을 배포합니다. Systems Manager Distributor는 Systems Manager 설명서에서 지원되는 패키지 플랫폼 및 아키텍처로 나열된 운영 체제를 지원합니다. Amazon Inspector에서 심층 검사 스캔을 수행하려면 Systems Manager Distributor 및 Amazon Inspector에서 Amazon EC2 인스턴스의 운영 체제를 지원해야 합니다.

Amazon Inspector SSM 플러그인에서 수집한 심층 검사용 데이터를 관리하기 위해 Amazon Inspector는 다음과 같은 파일 디렉터리를 생성합니다.

- `/opt/aws/inspector/var/input`
- `/opt/aws/inspector/var/output`
  - 이 디렉터리의 `packages.txt`에는 심층 검사로 발견된 패키지의 전체 경로가 저장됩니다. Amazon Inspector가 인스턴스에서 동일한 패키지를 여러 번 탐지한 경우 이 파일에는 해당 패키지가 발견된 위치가 각각 나열됩니다.

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

Amazon Inspector SSM 플러그인 제거

`inspectorssmplugin` 파일이 실수로 삭제된 경우 `InspectorLinuxDistributor-do-not-delete` SSM 연결을 통해 다음 스캔 간격에 플러그인이 재설치됩니다.

Amazon EC2 스캐닝을 비활성화하면 플러그인이 모든 Linux 호스트에서 자동으로 제거됩니다.

Amazon Inspector 심층 검사를 위한 사용자 지정 경로

Amazon Inspector가 Linux Amazon EC2 인스턴스에 대한 심층 검사를 수행할 때 검색하도록 사용자 지정 경로를 구성할 수 있습니다. 사용자 지정 경로를 추가하면 Amazon Inspector는 해당 디렉터리와 그 안에 있는 모든 하위 디렉터리의 패키지를 검색합니다.

모든 계정은 개별 계정당 최대 5개의 사용자 지정 경로를 정의할 수 있습니다. 조직의 위임 관리자인 경우 조직 전체에 적용될 경로 5개를 추가로 정의할 수 있습니다. 이로써 조직의 계정당 스캔할 수 있는 사용자 지정 경로는 총 10개까지 늘어납니다.

Amazon Inspector는 모든 계정을 대상으로 스캔하는 다음 기본 경로 외에 모든 사용자 지정 경로를 스캔합니다.

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

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

사용자 지정 경로의 형식

사용자 지정 경로 형식의 예는 `/home/usr1/project01`입니다.

사용자 지정 경로는 256자를 초과할 수 없습니다.

인스턴스당 패키지 수량은 5,000개로 제한되며, 패키지 인벤토리 수집 시간은 15분으로 제한됩니다. 이러한 제한을 피하려면 사용자 지정 경로를 선택하는 것이 좋습니다.

콘솔에서 사용자 지정 경로 설정

Console

Amazon Inspector 위임 관리자로 로그인한 후 다음 단계에 따라 조직의 사용자 지정 경로를 추가합니다.

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 Lambda 표준 스캔을 활성화하려는 지역을 선택합니다.
3. 측면 탐색 패널의 일반 설정에서 EC2 스캔 설정을 선택합니다.
4. 자체 계정의 사용자 지정 경로에서 편집을 선택하여 개별 계정의 경로를 추가합니다. 위임 관리자인 경우 조직의 사용자 지정 경로 창에서 편집을 선택하여 조직 내 모든 계정에 대한 사용자 지정 경로를 추가할 수 있습니다.
5. 텍스트 상자에 사용자 지정 경로를 입력합니다.
6. 저장을 선택하여 사용자 지정 경로를 저장합니다. Amazon Inspector는 다음 심층 검사에 이러한 경로를 포함할 예정입니다.

API

UpdateEc2DeepInspectionConfiguration 명령을 실행합니다. `packagePaths`의 경우 스캔할 경로 배열을 지정합니다.

지원되는 프로그래밍 언어

Linux 인스턴스의 경우 Amazon Inspector 심층 검사를 통해 운영 체제 패키지의 취약성 외에도 애플리케이션 프로그래밍 언어 패키지에 대한 조사 결과를 얻을 수 있습니다. Mac 및 Windows 인스턴스의 경우 운영 체제 패키지만 스캔됩니다.

지원되는 프로그래밍 언어에 대한 자세한 내용은 Amazon Inspector 심층 검사를 지원하는 프로그래밍 언어 섹션을 참조하세요.

Amazon Inspector로 Windows EC2 인스턴스 스캔

> **Note**
>
> 2022년 8월 31일, Amazon Inspector의 Amazon EC2 스캔 적용 범위가 확대되어 Windows를 실행 중인 EC2 인스턴스가 포함되었습니다.

Amazon Inspector는 지원되는 모든 Windows 인스턴스를 자동으로 검색하여 추가 조치 없이 연속 스캔에 포함시킵니다. 지원되는 인스턴스에 대한 자세한 내용은 Amazon EC2 스캔을 지원하는 운영 체제 섹션을 참조하세요.

Linux 기반 인스턴스 스캔과 달리, Amazon Inspector는 정기적으로 Windows 스캔을 실행합니다. Windows 인스턴스는 처음 검색 시 스캔된 후 6시간마다 스캔됩니다. 하지만 기본 6시간 스캔 간격은 조정할 수 있습니다. 자세한 설명은 Windows 인스턴스 스캔을 위한 사용자 지정 일정 설정 섹션을 참조하세요. 다음은 Amazon Inspector에서 Windows 인스턴스를 스캔하는 방법에 대한 간략한 설명입니다.

1. Amazon EC2 스캔이 활성화되면 Amazon Inspector에서 Windows 리소스에 대한 새 SSM 연결 `InspectorDistributor-do-not-delete`, `InspectorInventoryCollection-do-not-delete`, `InvokeInspectorSsmPlugin-do-not-delete`를 생성합니다.
2. `InspectorDistributor-do-not-delete`SSM 연결은 `AWS-ConfigureAWSPackage` SSM 문서와 `AmazonInspector2-InspectorSsmPlugin` SSM 배포자 패키지를 사용하여 인스턴스에 Amazon Inspector SSM 플러그인을 설치합니다. Windows 자세한 내용은 에 대한 Amazon Inspector SSM 플러그인에 대한 정보 Windows를 참조하세요.
3. `InvokeInspectorSsmPlugin-do-not-delete`SSM 연결은 Amazon Inspector SSM 플러그인을 정기적으로 실행하여 인스턴스 데이터를 수집하고 Amazon Inspector 검색 결과를 생성합니다. 기본적으로 6시간 간격으로 실행됩니다. 하지만 SSM을 사용하여 연결에 대한 cron 표현식 또

는 rate 표현식을 설정하여 이 간격을 사용자 지정할 수 있습니다. 자세한 내용은 AWS Systems Manager 사용 설명서의 참조: Systems Manager용 Cron 및 Rate 표현식을 참조하세요.

> **Note**
>
> Amazon Inspector는 업데이트된 OVAL(Open Vulnerability and Assessment Language) 정의 파일을 S3 버킷 `inspector2-oval-prod-REGION`에 스테이징합니다. 이 S3 버킷에는 스캔에 사용되는 OVAL 정의가 포함되어 있으므로 수정하면 안됩니다. 이 설정을 변경하면 Amazon Inspector에서 새로운 CVE가 발표될 때 이를 스캔하지 못하게 됩니다.

Windows 인스턴스에 대한 Amazon Inspector 스캔 요구 사항

Amazon Inspector에서 Windows 인스턴스를 스캔하려면 인스턴스가 다음 기준을 충족해야 합니다.

- 인스턴스가 SSM 관리형 인스턴스입니다. 스캔할 인스턴스 설정에 대한 지침은 SSM 에이전트 구성 섹션을 참조하세요.
- 인스턴스 운영 체제가 지원되는 Windows 운영 체제 중 하나입니다. 지원되는 운영 체제의 전체 목록은 Amazon EC2 스캔을 지원하는 운영 체제 섹션을 참조하세요.
- 인스턴스에는 Amazon Inspector SSM 플러그인이 설치되어 있습니다. Amazon Inspector는 검색 시 관리형 인스턴스를 위한 Amazon Inspector SSM 플러그인을 자동으로 설치합니다. 플러그인에 대한 자세한 내용은 다음 주제를 참조하세요.

> **Note**
>
> 호스트가 외부 인터넷에 액세스할 수 없는 Amazon VPC에서 실행되는 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 설명서에서 게이트웨이 엔드포인트 생성을 참조하세요. Amazon VPC 엔드포인트 정책이 외부 S3 버킷에 대한 액세스를 제한하는 경우, 인스턴스를 평가하는 데 사용되는 OVAL 정의를 저장하는 Amazon Inspector에서 유지 관리하는 AWS 리전 버킷에 대한 액세스를 특별히 허용해야 합니다. 이 버킷의 형식은 `inspector2-oval-prod-REGION`입니다.

에 대한 Amazon Inspector SSM 플러그인에 대한 정보 Windows

Amazon Inspector에서 인스턴스를 스캔하려면 Amazon Inspector SSM 플러그인이 필요합니다. Windows Amazon Inspector SSM 플러그인은 의 Windows `C:\Program Files\Amazon\Inspector` 인스턴스에 자동으로 설치되며 실행 가능한 바이너리 파일의 이름이 지정됩니다. `InspectorSsmPlugin.exe`

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

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

> **Note**
>
> 기본적으로 Amazon Inspector SSM 플러그인은 보통 우선순위보다 낮은 우선 순위로 실행됩니다.

Amazon Inspector SSM 플러그인 제거

`InspectorSsmPlugin.exe` 파일이 실수로 삭제된 경우 `InspectorDistributor-do-not-delete` SSM 연결을 통해 다음 Windows 스캔 간격에 플러그인이 재설치됩니다. Amazon Inspector SSM 플러그인을 제거하려는 경우 문서에서 제거 작업을 사용할 수 있습니다. `AmazonInspector2-ConfigureInspectorSsmPlugin`

또한 Amazon EC2 스캐닝을 비활성화하면 Amazon Inspector SSM 플러그인이 모든 Windows 호스트에서 자동으로 제거됩니다.

> **Note**
>
> Amazon Inspector를 비활성화하기 전에 SSM 에이전트를 제거하면 Amazon Inspector SSM 플러그인은 Windows 호스트에 남아 있지만 더 이상 Amazon Inspector SSM 플러그인으로 데이터를 전송하지 않습니다. 자세한 설명은 Amazon Inspector 비활성화 섹션을 참조하세요.

Windows 인스턴스 스캔을 위한 사용자 지정 일정 설정

SSM을 통해 `InvokeInspectorSsmPlugin-do-not-delete` 연결에 대한 cron 표현식 또는 rate 표현식을 설정하여 Windows Amazon EC2 인스턴스의 스캔 간격을 사용자 지정할 수 있습니다. 자세한 내용은 AWS Systems Manager 사용 설명서의 참조: Systems Manager의 Cron 및 Rate 표현식을 참조하거나 다음 지침을 따르세요.

다음 코드 예제 중 하나를 선택하여 rate 표현식 또는 cron 표현식을 사용하여 Windows 인스턴스의 스캔 주기를 기본 6시간에서 12시간으로 변경할 수 있습니다.

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

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

> **Note**
>
> AssociationId는 지역적이므로 먼저 각 ID의 고유 ID를 검색해야 AWS 리전합니다. 그런 다음 명령을 실행하여 Windows 인스턴스에 대한 사용자 지정 스캔 일정을 설정하려는 각 리전의 스캔 주기를 변경할 수 있습니다.

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 ECR 컨테이너 이미지 스캔

Amazon Inspector는 Amazon ECR에 저장된 컨테이너 이미지에 소프트웨어 취약성이 있는지 스캔하여 패키지 취약성 결과를 생성합니다. 이러한 문제에 대해 생성되는 결과 유형에 대한 자세한 내용은 Amazon Inspector의 결과 유형 섹션을 참조하세요.

Amazon ECR에 대한 Amazon Inspector 스캔을 활성화할 때 프라이빗 레지스트리의 기본 스캔 서비스로 Amazon Inspector를 설정하세요. 그러면 기본 스캔(Amazon ECR에서 무료로 제공함)이 고급 스캔(Amazon Inspector를 통해 제공되고 요금이 청구됨)으로 대체됩니다.

Amazon Inspector에서 제공하는 고급 스캔 기능을 사용하면 레지스트리 수준에서 운영 체제 및 프로그래밍 언어 패키지 모두에 대한 취약성 스캔의 이점을 누릴 수 있습니다. 이미지의 각 계층에 대해 이미지 수준에서 고급 스캔을 사용하여 발견된 결과는 Amazon ECR 콘솔에서 검토할 수 있습니다. 또한 Amazon을 비롯한 AWS Security Hub 기본 스캔 결과에 사용할 수 없는 다른 서비스에서도 이러한 결과를 검토하고 사용할 수 EventBridge 있습니다. Amazon Inspector 콘솔 https://console.aws.amazon.com/inspector/v2/home 에서 스캔을 통해 발견된 결과를 볼 수 있습니다. 결과 작업에 대한 자세한 내용은 Amazon Inspector에서 결과 관리 섹션을 참조하세요.

Amazon ECR 스캔 활성화에 대한 지침은 스캔 유형 활성화 섹션을 참조하세요.

Amazon ECR 스캔의 스캔 동작

ECR 스캔을 처음 활성화하고 리포지토리가 연속 스캔을 위해 구성되면 Amazon Inspector는 30일 이내에 푸시했거나 지난 90일 이내에 가져온 모든 적격 이미지를 탐지합니다. 그런 다음 Amazon Inspector는 탐지된 이미지를 스캔하고 스캔 상태를 로 설정합니다. `active` Amazon Inspector는 최근 90일 이내 (기본값) 또는 사용자가 구성한 ECR 재스캔 기간 내에 푸시 또는 풀링된 이미지를 계속 모니터링합니다. 자세한 설명은 ECR 재스캔 기간 구성 섹션을 참조하세요.

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

- 새 컨테이너 이미지가 푸시될 때마다
- Amazon Inspector가 새로운 일반적인 취약성 및 노출(CVE) 항목을 데이터베이스에 추가하고, 해당 CVE가 해당 컨테이너 이미지와 관련이 있을 때마다(연속 스캔만 해당)

온푸시 스캔을 사용하도록 리포지토리를 구성하면 이미지를 푸시할 때만 이미지가 스캔됩니다.

컨테이너 이미지의 취약성을 마지막으로 검사한 시기는 계정 관리 페이지의 컨테이너 이미지 탭에서 또는 ListCoverage API를 사용하여 확인할 수 있습니다. Amazon Inspector는 다음 이벤트에 대한 응답으로 Amazon ECR 이미지의 마지막 스캔 시간 필드를 업데이트합니다.

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

지원되는 운영 체제 및 미디어 유형

지원되는 운영 체제에 대한 자세한 내용은 Amazon ECR 스캔을 지원하는 운영 체제 섹션을 참조하세요.

Amazon Inspector의 Amazon ECR 리포지토리 스캔에서 지원되는 미디어 유형은 다음과 같습니다.

- `"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 `DockerV2ListMediaType` 이미지는 지원되지 않습니다.

Amazon ECR 리포지토리에 대한 고급 스캔 구성

Amazon ECR 컨테이너 이미지에 대한 Amazon Inspector 스캔을 활성화하면 프라이빗 레지스트리의 스캔 구성 설정이 변경됩니다. 해당 레지스트리의 스캔 유형이 기본 스캔에서 Amazon Inspector에서 제공하는 고급 스캔으로 변경됩니다. 자세한 내용은 Amazon ECR 사용 설명서에서 이미지 스캔을 참조하세요.

고급 스캔 설정은 ECR의 리포지토리 수준에서 관리할 수 있습니다. 리포지토리에 대해 연속 스캔 또는 푸시 시 스캔을 선택할 수 있습니다. 연속 스캔에는 푸시 시 스캔 및 자동 재스캔이 포함됩니다. 푸시 시 스캔은 이미지를 처음 푸시할 때만 스캔합니다. 두 옵션 모두 포함 필터를 통해 스캔 범위를 조정할 수 있습니다. 기본적으로 고급 스캔을 처음 활성화하면 모든 리포지토리를 지속적으로 스캔하도록 설정됩니다.

고급 스캔 설정을 구성하려면

1. Amazon ECR 콘솔(https://console.aws.amazon.com/ecr/)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기에서 스캔 중인 리포지토리가 있는 지역을 선택합니다.
3. 탐색 창에서 프라이빗 레지스트리, 스캔을 차례로 선택합니다.
4. 스캔 유형에서 고급 스캔이 선택되어 있는지 확인합니다. 선택되어 있지 않은 경우 고급 스캔을 선택합니다.

   기본적으로 모든 리포지토리를 지속적으로 스캔 옵션이 선택되어 있으며, 이 경우 모든 리포지토리에 대해 Amazon Inspector의 전체 스캔 적용 범위가 활성화됩니다.
5. 연속 스캔 또는 푸시 시 스캔을 수행할 리포지토리를 필터링하려면 모든 리포지토리를 지속적으로 스캔을 선택 취소합니다.

고급 스캔 구성에 대한 자세한 내용은 Amazon ECR 사용 설명서에서 고급 스캔 사용을 참조하세요.

ECR 재스캔 기간 구성

ECR 재스캔 기간 설정에 따라 Amazon Inspector가 리포지토리의 컨테이너 이미지를 지속적으로 모니터링하는 기간이 결정됩니다. 이미지 푸시 날짜 및 이미지 풀 날짜의 재스캔 기간을 구성할 수 있습니다. 조직에 추가된 새 계정을 포함하여 새 계정의 기본 검색 기간은 90일입니다.

이미지 푸시 날짜 기간

이미지 푸시 날짜 기간은 Amazon Inspector에서 이미지를 최신 풀 날짜 이후에 리포지토리로 푸시한 후 이미지를 지속적으로 모니터링하는 기간을 결정합니다. 재스캔 기간으로 사용할 수 있는 옵션은 다음과 같습니다.

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

이미지 가져오기 날짜 기간

이미지 풀 날짜 기간에 따라 Amazon Inspector가 최근 풀 날짜 이후에 이미지를 지속적으로 모니터링하는 기간이 결정됩니다. 재스캔 기간으로 사용할 수 있는 옵션은 다음과 같습니다.

- 14일
- 30일
- 60일
- 90일 (기본값)
- 180일

Amazon Inspector는 구성된 푸시 및 풀 날짜 내에 푸시 또는 풀링된 이미지를 계속 모니터링하고 다시 스캔합니다. 구성된 푸시 및 풀 날짜 내에 이미지가 푸시되거나 풀링되지 않은 경우 Amazon Inspector는 이미지 모니터링을 중단합니다.

> **Note**
>
> Amazon Inspector는 이미지 모니터링을 중지하면 이미지 스캔 상태 코드를 로 `inactive` 설정하고 사유 코드를 로 설정합니다. `expired` 그런 다음 모든 관련 이미지 검색 결과를 닫도록 스케줄링합니다.

환경에 가장 적합하도록 재스캔 시간을 설정하십시오. 예를 들어 이미지를 자주 만드는 경우 스캔 시간을 짧게 선택하십시오. 마찬가지로 이미지를 장기간 사용하는 경우 스캔 시간을 더 길게 선택하십시오.

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

ECR 재스캔 기간을 구성하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 탐색 창에서 일반 설정을 선택한 다음 ECR 스캔 설정을 선택합니다.
3. ECR 스캔 설정의 ECR 재스캔 기간에서 설정하려는 이미지 푸시 날짜 기간과 이미지 가져오기 날짜 기간을 선택합니다.
4. 저장을 선택합니다. 새 설정이 즉시 적용됩니다.

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

Amazon AWS Lambda Inspector를 사용한 스캔 기능

Amazon Inspector 함수 지원은 Lambda AWS Lambda 함수 및 계층에 대한 지속적이고 자동화된 보안 취약성 평가를 제공합니다. Amazon Inspector는 Lambda에 대해 두 가지 유형의 스캔을 제공합니다. 이러한 스캔 유형은 서로 다른 유형의 취약성을 찾아냅니다.

Amazon Inspector Lambda 표준 스캔

기본 Lambda 스캔 유형입니다. Lambda 표준 스캔은 Lambda 함수 및 해당 계층 내의 애플리케이션 종속성을 스캔하여 패키지 취약성을 찾아냅니다. 자세한 설명은 Lambda 표준 스캔 섹션을 참조하세요.

Amazon Inspector Lambda 코드 스캔

이 스캔 유형은 함수 및 계층의 사용자 지정 애플리케이션 코드를 스캔하여 코드 취약성을 찾아냅니다. Lambda 표준 스캔을 활성화하거나, Lambda 코드 스캔과 함께 Lambda 표준 스캔을 활성화할 수 있습니다. 자세한 설명은 Amazon Inspector Lambda 코드 스캔 섹션을 참조하세요.

Lambda 스캔을 활성화하면 Amazon Inspector는 계정에 AWS CloudTrail 다음과 같은 서비스 연결 채널을 생성합니다.

- `cloudtrail:CreateServiceLinkedChannel`
- `cloudtrail:DeleteServiceLinkedChannel`

Amazon Inspector는 이러한 채널을 관리하고 이를 사용하여 스캔을 위한 CloudTrail 이벤트를 모니터링합니다. 서비스 연결 채널에 대한 자세한 내용은 CLI를 CloudTrail 사용한 서비스 연결 채널 보기를 참조하십시오. AWS

> **Note**
>
> Amazon Inspector에서 만든 서비스 연결 채널을 사용하면 계정의 이벤트를 CloudTrail 추적한 것처럼 CloudTrail 볼 수 있지만 계정의 이벤트를 관리하려면 CloudTrail 직접 만드는 것이 좋습니다.

Lambda 함수 스캔 활성화에 대한 지침은 스캔 유형 활성화 섹션을 참조하세요.

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 새로운 일반적인 취약성 및 노출(CVE) 항목을 데이터베이스에 추가하고, 해당 CVE가 함수와 관련이 있는 경우

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

Lambda 함수의 취약성을 마지막으로 확인한 시기는 계정 관리 페이지의 Lambda 함수 탭에서 또는 ListCoverage API를 사용하여 확인할 수 있습니다. 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
- 함수에 영향을 미치는 새 CVE 항목이 Amazon Inspector 데이터베이스에 추가되어 Amazon Inspector에서 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 SDK 종속성을 AWS 스캔하지 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

특정 함수에 태그를 지정하여 Amazon Inspector Lambda 표준 스캔에서 제외할 수 있습니다. 함수를 스캔에서 제외하면 조치할 수 없는 알림을 방지하는 데 도움이 됩니다.

Lambda 함수를 Lambda 표준 스캔에서 제외하려면 다음 키-값 쌍을 사용하여 함수에 태그를 지정합니다.

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

Lambda 표준 스캔에서 함수를 제외하려면

1. https://console.aws.amazon.com/lambda/에서 Lambda 콘솔을 엽니다.
2. 함수를 선택합니다.
3. 함수 테이블에서 Amazon Inspector Lambda 표준 스캔에서 제외하려는 함수의 이름을 선택합니다.
4. 구성을 선택하고 메뉴에서 태그를 선택합니다.
5. 태그 관리, 새 태그 추가를 차례로 선택합니다.
6. 키 필드에 `InspectorExclusion`을 입력한 다음 값 필드에 `LambdaStandardScanning`을 입력합니다.
7. 저장을 선택하여 태그를 추가하고 Amazon Inspector Lambda 표준 스캔에서 함수를 제외합니다.

Lambda에 태그를 추가하는 방법에 대한 자세한 내용은 Lambda 함수에서 태그 사용을 참조하세요.

Amazon Inspector Lambda 코드 스캔

> **Important**
>
> 코드 스캔은 Lambda 함수에서 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 스캔은 Lambda 함수 내의 사용자 지정 애플리케이션 코드를 스캔하여 보안 모범 사례를 기반으로 코드 취약성을 찾아냅니다. AWS Lambda 코드 스캔으로 코드에서 주입 결함, 데이터 유출, 취약한 암호화 또는 누락된 암호화를 탐지할 수 있습니다. 사용 가능한 리전에 대한 자세한 내용은 리전별 기능 가용성 섹션을 참조하세요.

Lambda 표준 스캔은 함수에 사용된 애플리케이션 패키지 종속성을 평가하여 일반적인 취약성 및 노출(CVE)을 찾아내는 기능입니다. Lambda 코드 스캔은 Lambda 표준 스캔과 함께 활성화할 수 있습니다.

Amazon Inspector는 애플리케이션 코드의 전반적인 보안 규정 준수를 분석하는 자동 추론 및 기계 학습을 사용하여 Lambda 함수 애플리케이션 코드를 평가합니다. Amazon과 공동으로 개발한 내부 탐지기를 기반으로 정책 위반 및 취약성을 식별합니다. CodeGuru 가능한 탐지 목록은 탐지기 라이브러리를 참조하십시오. CodeGuru

Amazon Inspector에서 Lambda 함수 애플리케이션 코드의 취약성을 탐지한 경우 Amazon Inspector는 상세한 코드 취약성 유형의 결과를 생성합니다. 이 조사 결과 유형에는 코드에서 문제의 정확한 위치, 문제를 보여주는 코드 스니펫, 제안된 해결 방법이 포함되어 있습니다. 제안된 해결 방법에는 취약한 plug-and-play 코드 라인을 대체하는 데 사용할 수 있는 코드 블록이 포함됩니다. 이러한 제안된 코드 수정은 해당 조사 결과에 대한 일반 코드 수정 지침과 함께 제공됩니다.

> **Important**
>
> 코드 수정 제안은 자동화된 추론 및 생성형 AI 서비스를 기반으로 하므로 의도한 대로 작동하지 않을 수 있습니다. 제안된 코드 수정을 사용할 경우 이에 따른 책임은 사용자에게 있습니다. 그러므로 제안된 코드 수정 방법을 사용하기 전에 항상 검토하세요. 코드가 의도한 대로 작동하도록 제안된 코드 수정 방법을 편집해야 할 수도 있습니다. 책임감 있는 AI 정책을 참조하세요.

코드 취약성 결과에서 코드 암호화

Lambda 코드 스캔을 사용하여 발견된 코드 취약성과 관련하여 탐지된 코드 스니펫은 서비스에 저장됩니다. CodeGuru 기본적으로 에서 제어하는 AWS 소유 키가 코드를 암호화하는 데 사용되지만 Amazon Inspector API를 통해 암호화에 자체 고객 관리 키를 사용할 수 있습니다. CodeGuru 자세한 내용은 결과 코드에 대한 저장 중 암호화 섹션을 참조하세요.

Lambda 코드 스캔은 Lambda 표준 스캔과 함께 활성화할 수 있습니다. 스캔 유형 활성화에 대한 지침은 스캔 유형 활성화 섹션을 참조하세요.

Lambda 코드 스캔에서 함수 제외

특정 함수에 태그를 지정하여 Amazon Inspector Lambda 코드 스캔에서 제외할 수 있습니다. 함수를 스캔에서 제외하면 조치할 수 없는 알림을 방지하는 데 도움이 됩니다.

Lambda 함수를 Amazon Inspector, Lambda 코드 스캔에서 제외하려면 다음 키-값 쌍을 사용하여 함수에 태그를 지정합니다.

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

Lambda 코드 스캔에서 함수를 제외하려면

1. Lambda 콘솔(https://console.aws.amazon.com/lambda)에 로그인합니다.
2. 함수를 선택합니다.
3. 함수 테이블에서 Amazon Inspector Lambda 코드 스캔에서 제외하려는 함수의 이름을 선택합니다.
4. 구성을 선택하고 메뉴에서 태그를 선택합니다.
5. 태그 관리, 새 태그 추가를 차례로 선택합니다.
6. 키 필드에 `InspectorCodeExclusion`을 입력한 다음 값 필드에 `LambdaCodeScanning`을 입력합니다.
7. 저장을 선택하여 태그를 추가하고 Amazon Inspector Lambda 코드 스캔에서 함수를 제외합니다.

Lambda에 태그를 추가하는 방법에 대한 자세한 내용은 Lambda 함수에서 태그 사용을 참조하세요.

스캔 유형 비활성화

새 Amazon Inspector 스캔 유형은 언제든지 비활성화할 수 있습니다. 스캔 유형을 비활성화하면 해당 스캔 유형으로 생성된 기존 결과에 액세스할 수 없게 됩니다. 스캔 유형을 다시 활성화하면 적합한 리소스가 스캔되고 Amazon Inspector에서 새로운 결과를 생성합니다. 결과 데이터를 보관해 두려면 비활성화하기 전에 결과를 내보내면 됩니다. 자세한 설명은 Amazon Inspector에서 결과 보고서 내보내기 섹션을 참조하세요.

스캔 유형을 비활성화하면 비활성화되는 스캔 유형에 따라 해당 AWS 계정에서 특정 변경이 발생할 수 있습니다. 이러한 스캔 유형을 비활성화하면 다음 사항이 변경됩니다.

- Amazon EC2 스캔 - 계정에 대해 Amazon Inspector Amazon EC2 스캔을 비활성화하면 Amazon Inspector에서 사용하는 다음과 같은 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 플러그인이 모든 호스트에서 제거됩니다. Windows 자세한 설명은 Windows 인스턴스 스캔 섹션을 참조하세요.
- Amazon ECR 스캔 - 계정에 대해 Amazon ECR 컨테이너 이미지 스캔을 비활성화하면 해당 계정의 Amazon ECR 스캔 유형이 Amazon Inspector를 사용하는 고급 스캔에서 Amazon ECR을 사용하는 기본 스캔으로 변경됩니다.
- Lambda 표준 스캔 - 계정에서 Lambda 표준 스캔을 비활성화하면 코드 스캔이 활성화되어 있는 경우 Lambda 코드 스캔도 비활성화됩니다. 또한 스캔이 활성화되었을 때 생성된 CloudTrail 서비스 연결 채널이 삭제됩니다.

스캔 비활성화

계정에 대한 모든 스캔 유형을 비활성화하면 해당 AWS 리전의 해당 계정에 대한 Amazon Inspector가 비활성화됩니다. 자세한 설명은 Amazon Inspector 비활성화 섹션을 참조하세요.

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

Console

스캔을 비활성화하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 스캔을 비활성화하려는 지역을 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 AWS 리전 스캔 유형을 비활성화하려는 각 항목에서 이 단계를 반복합니다.

API

Disable API 작업을 실행합니다. 요청에 스캔을 비활성화할 계정 ID를 제공하고, `resourceTypes`로 `EC2`, `ECR`, `LAMBDA`, `LAMBDA_CODE` 중 하나 이상을 제공하면 스캔이 비활성화됩니다.

EC2 인스턴스에 대한 CIS (인터넷 보안 센터) 스캔

계정에 대한 Amazon Inspector EC2 스캐닝을 활성화하면 Amazon Inspector가 CIS 스캔을 수행하거나 예약할 수 있습니다. Amazon Inspector CIS는 Amazon EC2 인스턴스의 운영 체제를 벤치마킹하여 해당 운영 체제가 인터넷 보안 센터에서 수립한 모범 사례 권장 사항에 따라 구성되었는지 확인합니다. CIS 보안 벤치마크 프로그램은 시스템을 안전하게 구성하기 위한 업계 표준 구성 기준과 모범 사례를 제공합니다. 자세한 내용은 CIS 벤치마크란 무엇입니까? 를 참조하십시오.

Amazon Inspector는 스캔 구성에서 정의한 인스턴스 태그와 스캔 일정을 기반으로 대상 Amazon EC2 인스턴스에서 CIS 스캔을 수행합니다. Amazon Inspector는 각 대상 인스턴스에 대해 인스턴스에 대해 일련의 검사를 수행합니다. 각 검사는 시스템 구성이 특정 CIS 벤치마크 권장 사항을 충족하는지 여부를 평가합니다. 모든 검사에는 해당 플랫폼에 대한 CIS 벤치마크 권장 사항과 직접 관련된 CIS 검사 ID와 제목이 있습니다. 검사가 완료되면 결과를 보고 해당 시스템에서 인스턴스가 어떤 검사를 통과, 실패 또는 건너뛰었는지 확인할 수 있습니다.

Amazon Inspector CIS 스캔을 위한 EC2 인스턴스 요구 사항

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

- 인스턴스 운영 체제는 CIS 스캔을 지원하는 운영 체제 중 하나입니다. 지원되는 운영 체제의 전체 목록은 지원되는 운영 체제: CIS 스캐닝 섹션을 참조하세요.
- 인스턴스는 Amazon EC2 Systems Manager (SSM) 관리형 인스턴스입니다. 자세한 내용은 SSM 에이전트 작업을 참조하세요.
- 인스턴스에는 Amazon Inspector SSM 플러그인이 설치되어 있습니다. Amazon Inspector는 SSM 관리형 인스턴스용 이 플러그인을 자동으로 설치합니다.
- 인스턴스에는 SSM이 인스턴스를 관리할 수 있는 권한을 부여하는 인스턴스 프로필과 Amazon Inspector에서 해당 인스턴스에 대해 CIS 스캔을 실행할 수 있는 권한이 부여됩니다. 이러한 권한을 부여하려면 AmazonInspector2FullAccess, AmazonSSM ManagedInstanceCore 및 AmazonInspector2 ManagedCispolicy 정책을 IAM 역할에 연결하고 해당 역할을 인스턴스에 인스턴스 프로필로 연결합니다. 인스턴스 프로필 생성 및 연결에 대한 지침은 Amazon EC2 사용 설명서의 IAM 역할 사용을 참조하십시오.

> **Note**
>
> 인스턴스에서 CIS 스캔을 실행할 때 더 이상 Amazon Inspector 심층 검사를 활성화할 필요가 없습니다. 심층 검사를 비활성화해도 Amazon Inspector는 계속해서 SSM 에이전트를 설치하지만 더 이상 심층 검사를 실행하기 위해 플러그인을 호출하지 않습니다. 즉, 계정에 다음과 같은 연결이 나타납니다. `InspectorLinuxDistributor-do-not-delete`

CIS 스캔 실행

필요에 따라 CIS 스캔을 한 번 실행하거나 예약된 반복 스캔으로 실행할 수 있습니다. 검사를 실행하려면 먼저 검사 구성을 만들어야 합니다.

스캔 구성을 생성할 때 대상 인스턴스에 사용할 태그 키-값 쌍을 지정합니다. Amazon Inspector의 위임 조직 관리자인 경우 스캔 구성에서 여러 계정을 지정할 수 있으며, 그러면 Amazon Inspector는 각 계정에서 지정된 태그가 있는 인스턴스를 찾습니다. 스캔에 사용할 CIS 벤치마크 수준을 선택합니다. 각 벤치마크에 대해 CIS는 다양한 환경에 필요할 수 있는 다양한 보안 수준에 대한 기준을 제공하도록 설계된 레벨 1 및 레벨 2 프로파일을 지원합니다.

- 레벨 1 — 모든 시스템에서 구성할 수 있는 필수 기본 보안 설정을 권장합니다. 이러한 설정을 구현하면 서비스 중단이 거의 또는 전혀 발생하지 않을 것입니다. 이러한 권장 사항의 목표는 시스템의 진입점 수를 줄여 전반적인 사이버 보안 위험을 줄이는 것입니다.
- 레벨 2 — 보안이 엄격한 환경을 위한 고급 보안 설정을 권장합니다. 이러한 설정을 구현하려면 비즈니스에 미치는 영향을 최소화할 수 있는 계획과 조정이 필요합니다. 이러한 권장 사항의 목표는 규정 준수를 지원하는 것입니다.

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

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

> **Tip**
>
> 스캔이 실행되는 동안 시스템에 미치는 영향이 가장 적은 요일과 시간을 선택하는 것이 좋습니다.

CIS 스캔 구성을 만들려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 CIS 스캔을 실행할 AWS 리전 위치를 선택합니다.
3. 탐색 패널의 온디맨드 검사에서 CIS 검사를 선택합니다.
4. 새 스캔 만들기를 선택합니다.
   a. 스캔 구성 이름을 입력합니다.
   b. 대상 리소스에 스캔하려는 인스턴스에 있는 태그의 키와 해당 값을 입력합니다. 스캔에 포함할 총 25개의 태그를 지정할 수 있으며, 각 키에 대해 최대 5개의 다른 값을 지정할 수 있습니다.
   c. CIS 벤치마크 수준을 선택합니다. 기본 보안 구성의 경우 수준 1을 선택하고 고급 보안 구성의 경우 수준 2를 선택할 수 있습니다.
5. 대상 계정의 경우 검사에 포함할 계정을 지정합니다. 독립 실행형 계정 또는 조직 구성원은 셀프를 선택하여 자신의 계정에 대한 스캔 구성을 만들 수 있습니다. Amazon Inspector의 위임 관리자는 모든 계정을 선택하여 조직 내 모든 계정을 대상으로 지정하거나, 계정 지정을 선택하고 대상으로 지정할 구성원 계정의 하위 집합을 지정할 수 있습니다. 위임된 관리자는 계정 ID SELF 대신 입력하여 자신의 계정에 대한 스캔 구성을 생성할 수 있습니다. 자세한 내용은 조직에서 Amazon Inspector CIS 스캔을 관리하기 위한 고려 사항 AWS 섹션을 참조하세요.
6. 스캔 일정을 선택합니다. 검사 구성 만들기를 완료하는 즉시 실행되는 1회 검사와 삭제될 때까지 선택한 예약된 시간에 실행되는 반복 검사 중에서 선택하십시오.
7. [Create] 를 선택하여 스캔 구성 만들기를 완료합니다.

CIS 스캔 구성 보기 및 편집

언제든지 이전에 예약된 검사를 보거나 편집할 수 있습니다.

CIS 스캔 구성을 보거나 편집하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 CIS 스캔 구성을 만든 AWS 리전 위치를 선택합니다.
3. 탐색 패널의 온디맨드 검사에서 CIS 검사를 선택합니다.
4. 예약된 검사 구성을 보려면 예약됨을 선택합니다.

5. 스캔 구성 이름 열에서 항목을 선택하여 해당 스캔 구성의 세부 정보를 엽니다.
6. (선택 사항) 이 스캔의 매개 변수를 변경하려면 편집을 선택합니다.

CIS 스캔 결과 보기

Amazon Inspector는 스캔 구성이 실행될 때마다 스캔 작업을 생성하고 고유한 스캔 ID로 스캔 결과를 수집합니다.

스캔 결과는 스캔 완료 후 90일 동안 사용할 수 있습니다. 검사 또는 대상 자원별로 집계된 스캔 결과를 볼 수 있습니다.

검사별로 집계된 스캔 결과

스캔 결과는 스캔 중에 수행된 각 개별 검사별로 그룹화됩니다. 각 검사에 대해 통과, 실패 또는 건너뛰는 리소스의 수에 대한 보고서를 받게 됩니다.

리소스별로 집계된 스캔 결과

검색 결과는 검색 구성이 대상으로 지정한 각 리소스별로 그룹화됩니다. 각 리소스에 대해 해당 리소스의 검사 통과, 실패 또는 건너뛰기에 대한 보고서가 제공됩니다.

스캔 결과를 보려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 스캔 결과를 보려는 AWS 리전 위치를 선택합니다.
3. 탐색 패널의 온디맨드 검사에서 CIS 검사를 선택합니다.
4. 스캔 ID 열에서 결과를 보려는 스캔의 ID를 선택합니다.
5. 스캔 결과를 보는 방법을 선택합니다.
   - 검사 탭을 선택하면 검사별로 집계된 검사 결과를 볼 수 있습니다.
     - 나열된 검사의 경우 Resource status 열에서 합격, 건너뛰기 또는 실패 중에서 숫자를 선택하여 해당 상태 및 해당 검사를 기준으로 필터링된 리소스 보기를 엽니다.
   - 리소스별로 집계된 스캔 결과를 보려면 스캔한 리소스 탭을 선택합니다.
     - 리소스를 선택하면 리소스가 통과, 실패 또는 건너뛰었는지 여부를 나열하는 세부 정보 패널이 열립니다.
6. (선택 사항) 두 보기 중 하나에서 필터 막대를 사용하여 결과를 구체화하십시오.

콘솔 또는 API를 사용하여 CIS 스캔 결과를 다운로드할 수 있습니다.

검사 결과를 다운로드하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 스캔 결과를 보려는 AWS 리전 위치를 선택합니다.
3. 탐색 패널의 온디맨드 검사에서 CIS 검사를 선택합니다.
4. 스캔 ID 열에서 결과를 보려는 스캔의 ID를 선택합니다.
5. 다운로드를 선택합니다. 위임 관리자인 경우 특정 구성원 계정의 결과를 다운로드하도록 선택할 수 있습니다.

조직에서 Amazon Inspector CIS 스캔을 관리하기 위한 고려 사항 AWS

조직 내에서 CIS 스캔을 실행할 때 멤버 계정과 Amazon Inspector의 위임 관리자는 서로 다른 방식으로 CIS 스캔 구성 및 스캔 결과와 상호 작용합니다.

위임된 관리자가 모든 계정에 대한 CIS 스캔 구성 또는 구성원 계정 ID 목록을 생성하면 해당 조직이 해당 스캔 구성을 소유합니다. 현재 위임된 관리자는 다른 계정으로 만든 경우에도 조직 소유의 스캔 구성을 관리할 수 있습니다. 조직이 소유한 CIS 스캔 구성에는 다음과 같은 패턴에 따라 조직 ID를 소유자로 나열하는 ARN이 있습니다. `arn:aws:inspector2:`*`Region`*`:`*`111122223333`*`:owner/`*`OrganizationId`*`/cis-configuration/`*`scanId` 계정 ID는 조직 관리 계정의 ID가 됩니다.

> **Important**
>
> 조직 소유의 CIS 스캔 구성에는 태그를 추가할 수 없습니다.

위임된 관리자가 스캔 구성을 만들고 대상 계정으로 SELF 지정하면 위임된 관리자가 해당 스캔 구성을 소유하게 됩니다. 조직을 떠나도 해당 스캔 구성을 관리할 수 있습니다.

> **Note**
>
> 위임된 관리자는 대상으로 하는 검사 구성의 대상을 변경할 수 없습니다. SELF

구성원 계정, 독립 실행형 계정 또는 대상으로 위임된 SELF 관리자가 만든 스캔 구성은 해당 구성을 만든 계정이 소유합니다. 이러한 CIS 스캔 구성에는 패턴에 따라 해당 계정을 소유자로 나열하는 ARN이 있습니다. `arn:aws:inspector2:Region:111122223333:owner/111122223333/cis-configuration/scanId` 계정 ID는 스캔을 만든 계정이 됩니다.

조직의 구성원 계정은 자신의 계정에 대한 스캔 구성을 만들 수 있습니다. 위임된 관리자는 구성원이 만든 스캔 구성을 볼 수 있지만 편집하거나 삭제할 수는 없습니다. 구성원 계정이 조직을 떠나는 경우 위임된 관리자는 해당 계정으로 만든 스캔 구성을 더 이상 볼 수 없습니다.

위임된 관리자는 구성원이 예약한 계정을 포함하여 기관 내 모든 계정의 스캔 결과를 볼 수 있습니다. 구성원 계정은 위임된 관리자가 예약한 리소스를 포함하여 자신의 계정에 있는 리소스에 대한 모든 CIS 스캔 결과를 볼 수 있습니다.

아마존 인스펙터 소유의 아마존 S3 버킷은 아마존 인스펙터 CIS 스캔에 사용됩니다.

Amazon Inspector는 CIS 스캔에 필요한 업데이트된 공개 취약성 및 평가 언어 (OVAL) 정의 파일을 스테이징합니다. 다음 표에는 Amazon Inspector 소유의 모든 Amazon S3 버킷이 나열되어 있으며, CIS 스캔에서 지원되는 항목에 따라 사용하는 OVAL 정의가 나와 있습니다. AWS 리전필요한 경우 버킷을 VPC에서 허용 목록에 추가해야 합니다.

> **Note**
>
> 다음 Amazon Inspector 소유 Amazon S3 버킷 각각에 대한 세부 정보는 변경될 수 없습니다. 하지만 새로 지원되는 내용을 반영하도록 목록이 업데이트될 수 있습니다. AWS 리전이러한 버킷을 다른 Amazon S3 작업이나 자체 Amazon S3 버킷에 사용할 수 없습니다.

CIS 버킷	AWS 리전
`cis-datasets-prod-arn-5908f6f`	유럽(스톡홀름)
`cis-datasets-prod-bah-8f88801`	중동(바레인)
`cis-datasets-prod-bjs-0f40506`	중국(베이징)
`cis-datasets-prod-bom-435a167`	아시아 태평양(뭄바이)

CIS 버킷	AWS 리전
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(오리건)
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 버킷	AWS 리전
cis-datasets-prod-yul-5e0c95e	캐나다(중부)
cis-datasets-prod-zhy-5a8eacb	중국(닝샤)
cis-datasets-prod-zrh-67e0e3d	유럽(취리히)

AWS 환경의 Amazon Inspector 적용 범위 평가

Amazon Inspector 콘솔의 계정 관리 페이지는 AWS 환경의 Amazon Inspector 적용 범위를 평가하고 해석하는 데 도움이 되도록 계정 및 리소스에 대한 Amazon Inspector 스캔 상태에 대한 통계 및 세부 정보를 제공합니다. 이 페이지에서는 리소스에 대한 집계된 통계 및 기타 데이터를 검토할 수 있습니다. 또한 개별 리소스의 Amazon Inspector 적용 범위를 심층적으로 분석하고 특정 리소스에 대한 결과를 자세히 검토할 수 있습니다. 조직의 Amazon Inspector 위임 관리자인 경우 데이터에는 조직의 모든 계정에 대한 통계 및 세부 정보가 포함됩니다.

AWS 환경의 Amazon Inspector 적용 범위를 평가하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 탐색 창에서 계정 관리를 선택합니다.
3. 계정 관리 페이지에서 5가지 적용 범위 보기 중 하나의 탭을 선택하세요.

   - 계정 - 계정 수준 적용 범위입니다.
   - 인스턴스 - Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스의 적용 범위입니다.
   - 리포지토리 - Amazon Elastic Container Registry(Amazon ECR) 리포지토리의 적용 범위입니다.
   - 이미지 - Amazon ECR 컨테이너 이미지의 적용 범위입니다.
   - Lambda - Lambda 함수의 적용 범위입니다.

이 섹션의 주제에서는 개별 리소스의 스캔 상태를 포함하여 각 탭이 제공하는 정보에 대해 설명합니다.

주제

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 설명은 Amazon Inspector를 사용한 자동 리소스 스캔 섹션을 참조하세요.

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 설명은 멤버 계정에 대한 Amazon Inspector 스캔 활성화 섹션을 참조하세요.

Amazon EC2 인스턴스의 적용 범위 평가

인스턴스 탭에는 AWS 환경의 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.

  여러 가지 이유로 EC2 인스턴스가 스캔하지 않음 탭에 표시될 수 있습니다. Amazon Inspector는 AWS Systems Manager(SSM) 및 SSM 에이전트를 사용하여 EC2 인스턴스의 취약성을 자동으로 모니터링하고 스캔합니다. 인스턴스에서 SSM 에이전트가 실행되지 않거나, Systems Manager를 지원하는 AWS Identity and Access Management(IAM) 역할이 없거나, 지원되는 운영 체제 또는 아키텍처가 실행되고 있지 않은 경우, Amazon Inspector는 인스턴스를 모니터링하고 스캔할 수 없습니다. 자세한 설명은 Amazon EC2 인스턴스 스캔 섹션을 참조하세요.

각 탭의 계정 열은 인스턴스를 소유하고 있는 AWS 계정을 지정합니다.

EC2 인스턴스 태그 - 이 열에는 인스턴스와 연결된 태그가 표시되며, 이를 통해 해당 인스턴스가 태그로 인해 스캔에서 제외되었는지 확인할 수 있습니다.

운영 체제 - 이 열에는 운영 체제 유형(`WINDOWS`, `MAC`, `LINUX` 또는 `UNKNOWN`)이 표시됩니다.

사용된 모니터링 - 이 열에는 Amazon Inspector가 이 인스턴스에서 에이전트 기반 스캔 방법을 사용하는지 아니면 에이전트 없는 스캔 방법을 사용하는지가 표시됩니다.

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 열의 링크를 선택하세요. 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 결과 세부 정보 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.
- EC2 인스턴스가 중지됨 – 인스턴스가 중지 상태이기 때문에 Amazon Inspector에서 인스턴스 검색을 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 Manager를 사용하여 `InspectorInventoryCollection-do-not-delete` 연결이 존재하고 해당 연결 상태가 성공적인지 확인하세요. 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 Manager를 사용하여 필요한 해당 인스턴스에 필요한 Amazon Inspector 연결이 존재하고 실행 중인지 확인하세요. 또한 AWS Systems Manager Fleet Manager를 사용하여 인스턴스에 대한 소프트웨어 애플리케이션 인벤토리를 확인하세요.
- 비관리형 EC2 인스턴스 – Amazon Inspector에서 인스턴스를 모니터링하거나 스캔하지 않습니다. AWS Systems Manager에서 인스턴스를 관리하지 않습니다.

  이 문제를 해결하려면 AWS Systems Manager Automation에서 제공하는 AWSSupport-TroubleshootManagedInstance runbook을 사용할 수 있습니다. 인스턴스를 관리하도록 AWS Systems Manager를 구성하면 Amazon Inspector에서 자동으로 인스턴스를 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 Amazon EC2 스캔을 지원하는 운영 체제 섹션을 참조하세요.
- 능동적으로 모니터링, 부분적 오류 발생) - 이 상태는 EC2 스캔이 활성화되어 있지만 Amazon EC2 Linux 인스턴스에 대한 Amazon Inspector 심층 검사와 관련된 오류가 있음을 의미합니다. 가능한 심층 검사 오류는 다음과 같습니다.
  - 심층 검사 패키지 수집 한도 초과 — 인스턴스가 Amazon Inspector 심층 검사의 패키지 한도 5000개를 초과했습니다. 이 인스턴스에 대한 심층 검사를 재개하려면 계정과 관련된 사용자 지정 경로를 조정해 볼 수 있습니다.
  - 일일 ssm 인벤토리 심층 검사 한도 초과 — 이 인스턴스에서 인스턴스당 매일 수집되는 인벤토리 데이터에 대한 SSM 할당량에 이미 도달했기 때문에 SSM 에이전트가 Amazon Inspector에 인벤토리를 보낼 수 없었습니다. 자세한 내용은 Amazon EC2 Systems Manager 엔드포인트 및 할당량을 참조하세요.
  - 심층 검사 수집 시간 제한 초과 — 패키지 수거 시간이 최대 임계값인 15분을 초과하여 Amazon Inspector에서 패키지 인벤토리를 추출하지 못했습니다.
  - 심층 검사에 인벤토리가 없음 – Amazon Inspector SSM 플러그인에서 이 인스턴스의 패키지 인벤토리를 아직 수집하지 못했습니다. 이는 일반적으로 보류 중인 스캔의 결과이지만, 6시간 후에도 이 상태가 지속되면 Amazon EC2 Systems Manager를 사용하여 해당 인스턴스에 필요한 Amazon Inspector 연결이 존재하고 실행 중인지 확인하세요.

EC2 인스턴스의 스캔 설정 구성에 대한 자세한 내용은 Amazon EC2 인스턴스 스캔 섹션을 참조하세요.

Amazon ECR 리포지토리의 적용 범위 평가

리포지토리 탭에는 AWS 환경의 Amazon ECR 리포지토리가 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 탭의 계정 열은 리포지토리를 소유하고 있는 AWS 계정을 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 ECR 리포지토리의 상태 값 스캔

Amazon Elastic 컨테이너 레지스트리 (Amazon ECR) 리포지토리의 경우 가능한 상태 값은 다음과 같습니다.

- 활성화 (연속) — 리포지토리의 경우 Amazon Inspector는 이 리포지토리의 이미지를 지속적으로 모니터링합니다. 리포지토리의 고급 스캔 설정이 연속 스캔으로 설정되어 있습니다. Amazon Inspector는 처음에 새 이미지가 푸시되면 해당 이미지를 스캔하고 해당 이미지와 관련된 새 CVE가 게시되면 이미지를 다시 스캔합니다. Amazon Inspector는 사용자가 구성한 ECR 스캔 기간 동안 이 리포지토리의 이미지를 계속 모니터링합니다.
- 활성화 (푸시 시) — 새 이미지가 푸시되면 Amazon Inspector가 리포지토리의 개별 컨테이너 이미지를 자동으로 스캔합니다. 리포지토리에 대해 향상된 스캔 기능이 활성화되고 푸시 시 스캔하도록 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 Inspector의 리포지토리 액세스를 허용해야 합니다.
- 비활성화됨(수동) – Amazon Inspector에서 리포지토리에 있는 컨테이너 이미지를 모니터링하거나 스캔하지 않습니다. 리포지토리의 Amazon ECR 스캔 설정이 기본(수동 스캔)으로 설정되어 있습니다.

  Amazon Inspector로 리포지토리 이미지 스캔을 시작하려면 리포지토리의 스캔 설정을 고급 스캔으로 변경한 다음 이미지를 지속적으로 스캔할지 아니면 새 이미지가 푸시될 때만 스캔할지 여부를 선택하세요.
- 활성화 (푸시 시) — 새 이미지가 푸시되면 Amazon Inspector가 리포지토리의 개별 컨테이너 이미지를 자동으로 스캔합니다. 리포지토리의 고급 스캔 설정이 푸시할 때 스캔으로 설정되어 있습니다.
- 내부 오류 — Amazon Inspector에서 리포지토리 스캔을 시도할 때 내부 오류가 발생했습니다. Amazon Inspector에서 자동으로 오류를 해결하고 가능한 한 빨리 스캔을 재개합니다.

Amazon ECR 컨테이너 이미지 스캔리포지토리의 스캔 설정 구성에 대한 자세한 내용은 여기를 참조하십시오.

Amazon ECR 컨테이너 이미지의 적용 범위 평가

이미지 탭에는 AWS 환경의 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. 이미지가 Amazon Inspector 스캔이 활성화되지 않은 리포지토리에 저장되어 있거나 Amazon ECR 필터링 규칙으로 인해 해당 리포지토리가 스캔되지 않을 수 있습니다. 또는 ECR 재스캔 기간으로 구성한 일수 내에 이

미지를 푸시하거나 가져오지 않은 경우도 있습니다. 자세한 설명은 ECR 재스캔 기간 구성 섹션을 참조하세요.

각 탭의 리포지토리 이름 열은 컨테이너 이미지를 저장하는 리포지토리의 이름을 지정합니다. 계정 열은 리포지토리를 소유하고 있는 AWS 계정을 지정합니다. 마지막 스캔 열에는 Amazon Inspector가 해당 리소스의 취약성을 마지막으로 검사한 시간이 표시됩니다. 여기에는 조사 결과 메타데이터에 대한 업데이트가 있을 때, 리소스의 애플리케이션 인벤토리에 대한 업데이트가 있을 때, 새로운 CVE에 대한 대응으로 재스캔이 진행될 때 수행되는 검사가 포함될 수 있습니다. 자세한 설명은 Amazon ECR 스캔의 스캔 동작 섹션을 참조하세요.

컨테이너 이미지에 대한 추가 세부 정보를 검토하려면 ECR 컨테이너 이미지 열의 링크를 선택하세요. 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 결과 세부 정보 섹션을 참조하세요.

Amazon ECR 컨테이너 이미지의 상태 값 스캔

Amazon Elastic 컨테이너 레지스트리 컨테이너 이미지의 경우 가능한 상태 값은 다음과 같습니다.

- 능동적 모니터링 (지속적) — Amazon Inspector는 지속적으로 모니터링하고 새로운 관련 CVE가 게시될 때마다 이미지와 새 스캔을 수행합니다. 이미지에 대한 Amazon ECR 재스캔 기간은 이미지를 푸시하거나 가져올 때마다 새로 고쳐집니다. 이미지를 저장하는 리포지토리에 대해 고급 스캔이 활성화되고 리포지토리의 고급 스캔 설정이 연속 스캔으로 설정되어 있습니다.
- 활성화됨 (푸시 시) — Amazon Inspector는 새 이미지가 푸시될 때마다 이미지를 자동으로 스캔합니다. 이미지를 저장하는 리포지토리에 대해 고급 스캔이 활성화되고 리포지토리의 고급 스캔 설정이 푸시할 때 스캔으로 설정되어 있습니다.
- 내부 오류 — Amazon Inspector에서 컨테이너 이미지 스캔을 시도할 때 내부 오류가 발생했습니다. Amazon Inspector에서 자동으로 오류를 해결하고 가능한 한 빨리 스캔을 재개합니다.
- 초기 스캔 보류 중 — Amazon Inspector는 초기 스캔을 위해 이미지를 대기열에 넣었습니다.
- 스캔 자격 만료 (계속) — Amazon Inspector에서 이미지 스캔을 일시 중단했습니다. 리포지토리 이미지의 자동 재스캔 기간 내에 이미지가 업데이트되지 않았습니다. 이미지를 밀거나 당겨 스캔을 재개할 수 있습니다.
- 스캔 자격 만료 (푸시 시) — Amazon Inspector에서 이미지 스캔을 일시 중단했습니다. 리포지토리 이미지의 자동 재스캔 기간 내에 이미지가 업데이트되지 않았습니다. 이미지를 푸시하여 스캔을 재개할 수 있습니다.

- 스캔 빈도 수동(수동) – Amazon Inspector에서 Amazon ECR 컨테이너 이미지를 스캔하지 않습니다. 이미지를 저장하는 리포지토리의 Amazon ECR 스캔 설정이 기본(수동 스캔)으로 설정되어 있습니다. 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 Amazon ECR 스캔을 지원하는 운영 체제 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 적용 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 설명은 스캔 기능 AWS Lambda 섹션을 참조하세요.

각 탭의 함수 이름 열은 Lambda 함수의 이름을 지정합니다. 계정 열은 함수를 소유하고 있는 AWS 계정을 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 조사 결과 메타데이터에 대한 업데이트가 있을 때, 리소스의 애플리케이션 인벤토리에 대한 업데이트가 있을 때, 새로운 CVE에 대한 대응으로 재스캔이 진행될 때 수행되는 검사가 포함될 수 있습니다. 자세한 설명은 Lambda 함수 스캔의 스캔 동작 섹션을 참조하세요.

함수의 상태 값을 스캔하는 중입니다. AWS Lambda

Lambda 함수의 경우 가능한 상태 값은 다음과 같습니다.

- 능동 모니터링 – Amazon Inspector에서 Lambda 함수를 지속적으로 모니터링하고 스캔합니다. 연속 스캔에는 새 함수를 리포지토리로 푸시할 때의 최초 스캔과 함수가 업데이트되거나 새로운 일반 취약성 및 노출(CVE)이 발표될 때 자동으로 함수를 재스캔하는 것이 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.

Amazon Inspector에서 조직을 사용하여 여러 계정 관리하기

Amazon Inspector를 사용하여 Organizations를 통해 연결된 여러 계정을 관리할 수 있습니다. AWS 여러 Amazon Inspector 계정을 관리하기 위해 조직 관리 계정은 조직 내 계정을 Amazon Inspector의 위임된 관리자 계정으로 지정합니다. 위임 관리자는 조직의 Amazon Inspector를 관리하며 조직을 대신하여 작업을 수행할 수 있는 특별 권한을 부여받습니다. 이러한 작업에는 구성원 계정 스캔 활성화 또는 비활성화, 조직 전체의 집계된 검색 결과 데이터 보기, 금지 규칙 생성 및 관리 등이 포함됩니다.

> **Note**
>
> 여러 계정의 여러 계정에 대해 Amazon Inspector를 프로그래밍 방식으로 활성화하려면 Amazon Inspector에서 AWS 리전 개발한 셸 스크립트를 사용할 수 있습니다. 이 스크립트 사용에 대한 자세한 내용은 웹 사이트의 inspector2-를 참조하십시오. enablement-with-cli GitHub

주제

- Amazon Inspector에서 관리자 계정과 멤버 계정 간의 관계 이해
- Amazon Inspector 위임 관리자 지정

Amazon Inspector에서 관리자 계정과 멤버 계정 간의 관계 이해

다중 계정 환경에서 Amazon Inspector를 사용하는 경우 Amazon Inspector 위임 관리자 계정은 특정 메타데이터에 액세스할 수 있습니다. 이 메타데이터에는 Amazon EC2 및 Amazon ECR 구성 데이터와 멤버 계정에 대한 보안 조사 결과가 포함됩니다. 관리자 계정은 멤버 계정에 적용될 결과 억제 규칙을 생성할 수도 있습니다. 자세한 설명은 억제 규칙을 사용하여 Amazon Inspector 결과를 숨기는 방법 섹션을 참조하세요.

위임 관리자 작업

일반적으로 위임된 관리자가 자신의 계정에 설정을 적용하면 해당 설정이 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 멤버에 대해 Amazon ECR 고급 스캔을 활성화합니다.
- 전체 조직의 리소스 적용 범위를 확인합니다.
- 조직 내 모든 멤버 계정에 대해 ECR 컨테이너 이미지 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 재스캔 기간을 다르게 설정할 수는 없습니다.
- 조직의 모든 계정에서 사용할 Amazon Inspector 심층 검사를 위한 다섯 가지 사용자 지정 경로를 지정하십시오. 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 을 참조하십시오. Amazon Inspector 심층 검사를 위한 사용자 지정 경로
- 회원 계정에 대한 Amazon Inspector 심층 검사를 활성화 및 비활성화합니다.
- 조직 내 모든 멤버 계정의 SBOM을 내보냅니다.
- 조직의 모든 멤버 계정에 Amazon EC2 스캔 모드를 설정합니다. 자세한 설명은 스캔 모드 관리 섹션을 참조하세요.
- 구성원 계정으로 만든 스캔 구성을 제외하고 조직 내 모든 계정에 대한 CIS 스캔 구성을 생성하고 관리합니다.

> **Note**
>
> 구성원 계정이 조직을 떠나는 경우 위임된 관리자는 해당 계정으로 예약된 스캔 구성을 더 이상 볼 수 없습니다.

- 조직 내 모든 계정에 대한 CIS 스캔 결과를 볼 수 있습니다.

멤버 계정 작업

멤버 계정은 Amazon Inspector에서 자신의 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.
- 본인 계정에 대한 ECR 컨테이너 이미지 자동 재스캔 기간 설정을 확인합니다.
- 개별 계정에 사용할 Amazon Inspector EC2 심층 검사를 위한 다섯 가지 사용자 지정 경로를 지정하십시오. 이러한 경로는 위임된 관리자가 조직에 지정한 모든 사용자 지정 경로와 함께 스캔됩니다. 심층 검사 경로 구성에 대한 자세한 내용은 을 참조하십시오. Amazon Inspector 심층 검사를 위한 사용자 지정 경로
- Amazon Inspector 심층 검사를 위해 위임된 관리자가 설정한 사용자 지정 경로를 확인하십시오.
- 자신의 계정과 연결된 리소스의 SBOM을 내보냅니다.
- 자신의 계정의 스캔 모드를 확인합니다.
- 해당 계정의 CIS 스캔 구성을 생성하고 관리합니다.
- 위임된 관리자가 예약한 리소스를 포함하여 해당 계정의 리소스에 대한 모든 CIS 스캔 결과를 볼 수 있습니다.

> **Note**
>
> Amazon Inspector는 활성화 후 위임 관리자 계정을 통해서만 비활성화할 수 있습니다.

Amazon Inspector 위임 관리자 지정

위임 관리자에 대한 중요 고려 사항

다음은 Amazon Inspector에서 위임 관리자의 운영 방식을 정의하는 요소입니다.

위임 관리자는 최대 5,000명의 멤버를 관리할 수 있습니다.

> Amazon Inspector 위임 관리자마다 5,000개의 회원 계정 할당량이 있습니다. 하지만 조직에 포함된 계정이 5,000개가 넘을 수도 있습니다. 회원 계정이 5,000개를 초과하는 경우 Amazon CloudWatch Personal Health Dashboard를 통해 알림을 받고 위임된 관리자 계정으로 이메일을 받게 됩니다.

위임 관리자는 리전별로 결정됩니다.

> AWS Organizations와 달리 Amazon Inspector는 리전 서비스입니다. 즉, Amazon Inspector에서 사용할 위임 관리자를 지정하고, 멤버 계정을 추가하고, 각 AWS 리전 계정에서 스캔 유형을 활성화해야 합니다.

조직의 위임 관리자는 한 명입니다.

Amazon Inspector 위임 관리자는 조직당 한 명만 있을 수 있습니다. 한 지역의 계정을 위임 관리자로 지정한 경우 다른 모든 지역의 위임 관리자여야 합니다.

위임 관리자를 변경해도 멤버 계정의 Amazon Inspector는 비활성화되지 않습니다.

위임된 관리자를 제거해도 Amazon Inspector는 해당 계정에서 비활성화되지 않으며 스캔 설정도 영향을 받지 않습니다.

AWS Organization의 의 모든 기능이 활성화되어 있어야 합니다.

의 기본 설정입니다. AWS Organizations 활성화되지 않은 경우 조직의 모든 기능 활성화를 참조하십시오.

위임 관리자를 지정하는 데 필요한 권한

Amazon Inspector를 활성화하고 Amazon Inspector 위임 관리자를 지정할 수 있는 권한이 있어야 합니다.

이러한 권한을 부여하려면 IAM 정책의 끝에 다음 문을 추가합니다.

```
{
    "Sid": "PermissionsForInspectorAdmin",
    "Effect": "Allow",
    "Action": [
        "inspector2:EnableDelegatedAdminAccount",
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganizationalUnit",
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization"
    ],
    "Resource": "*"
}
```

AWS 조직의 위임 관리자 지정

다음 절차는 AWS 조직의 위임 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 처음으로 활성화하면 계정에 대한 서비스 연결 역할(SLR`AWSServiceRoleForAmazonInspector`) 이 생성됩니다. Amazon Inspector에서 서비스 연결 역할을 사용하는 방법에 대한 자세한 내용은 Amazon Inspector에 서비스 연결 역할 사용 섹션을 참조하세요. 일반적인 서비스 연결 역할에 대한 자세한 내용은 IAM 사용 설명서에서 서비스 연결 역할 사용을 참조하세요.

Amazon Inspector 위임 관리자를 지정하려면

Console

콘솔에서 위임 관리자 지정

1. AWS Organizations 관리 계정을 사용하여 AWS Management Console에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 연 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 관리자를 지정하려는 지역을 지정합니다.
3. 위임된 관리자 창에서 조직의 Amazon Inspector 위임 관리자로 지정하려는 12자리 계정 ID를 입력합니다. AWS 계정 그런 다음 관리 위임을 선택합니다.
4. (권장) 각 AWS 리전에 대해 위의 단계를 반복합니다.

API

API를 사용하여 위임 관리자 지정

- Organizations 관리 계정의 자격 증명을 사용하여 EnableDelegatedAdminAccountAPI 작업을 실행합니다. AWS 계정 를 사용하여 다음 AWS Command Line Interface CLI 명령을 실행하여 이 작업을 수행할 수도 있습니다. `aws inspector2 enable-delegated-admin-account --delegated-admin-account-id 11111111111`

  > **Note**
  >
  > Amazon Inspector에서 위임 관리자로 지정하려는 계정의 계정 ID를 지정해야 합니다.

위임된 관리자를 지정한 후에는 위임된 관리자 계정을 변경하거나 제거할 때만 AWS Organizations 관리 계정을 사용해야 합니다.

멤버 계정에 대한 Amazon Inspector 스캔 활성화

조직의 위임 관리자는 AWS Organizations 관리 계정과 연결된 모든 멤버에 대해 Amazon EC2 스캔, Amazon ECR 스캔 또는 둘 다를 활성화할 수 있습니다. 멤버 계정에 대한 스캔을 활성화하면 해당 계정이 위임 관리자와 연결되고 Amazon Inspector가 자동으로 활성화되며 선택한 유형의 스캔이 즉시 시작됩니다. 스캔할 수 있는 리소스 및 스캔 구성 방법에 대한 자세한 내용은 을 참조하십시오. Amazon Inspector를 사용한 자동 리소스 스캔

Amazon Inspector는 멤버 계정에서 Amazon Inspector를 활성화할 수 있도록 허용하는 것을 포함하여 멤버 계정에 대한 스캔을 관리하고 활성화하기 위한 몇 가지 옵션을 제공합니다. 다음 옵션 중 하나를 사용하여 멤버 계정에 대한 스캔을 시작합니다.

모든 멤버 계정에 대한 스캔을 자동으로 활성화하려면

1. 위임된 관리자 계정에 로그인하기
2. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다. 그런 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 모든 구성원 계정에 대한 검색을 활성화할 지역을 지정합니다.
3. 탐색 창의 설정에서 계정 관리를 선택합니다. 계정 테이블에 AWS Organizations 관리 계정과 관련된 모든 멤버 계정이 표시됩니다.
4. 테이블 상단의 확인란을 선택하여 이 페이지에 있는 모든 계정을 선택합니다. 그런 다음 활성화를 선택하고 메뉴에서 원하는 스캔 유형 옵션을 선택합니다.

   > **Note**
   >
   > 페이지에 현재 표시된 계정만 선택됩니다. 계정 페이지가 여러 개인 경우 각 페이지에서 이 프로세스를 반복해야 합니다. 페이지에 표시된 계정 수를 변경하려면 톱니바퀴 아이콘을 선택합니다.

5. 새 구성원 계정에 대해 Inspector 자동 활성화 설정을 켠 다음 스캔 유형을 선택하여 기관에 추가된 새 구성원을 활성화합니다.
6. (권장) 구성원 계정을 스캔하려는 각 지역에서 이 단계를 반복하세요.

새 멤버 계정에 대해 자동으로 검사기 활성화 설정을 사용하면 조직의 이후 모든 멤버에 대해 Amazon Inspector가 활성화됩니다. 따라서 Amazon Inspector 위임 관리자가 조직에 추가된 새 멤버를 관리할

수 있습니다. 회원 계정 수가 할당량인 5,000개에 도달하면 이 설정은 자동으로 해제됩니다. 계정이 제거되어 총 멤버 수가 5,000개 미만으로 줄어들면 이 설정은 자동으로 다시 활성화됩니다.

멤버 계정을 선택적으로 활성화하려면

1. 위임된 관리자 계정에 로그인하기
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 연 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 특정 회원 계정에 대한 스캔을 활성화하려는 지역을 지정합니다.
3. 탐색 창의 설정에서 계정 관리를 선택합니다. 계정 테이블에 AWS Organizations 관리 계정과 관련된 모든 멤버 계정이 표시됩니다.
4. 계정 관리 페이지에서 스캔을 활성화하려는 각 멤버 계정의 확인란을 선택합니다.
5. 활성화를 선택합니다.
6. 활성화 메뉴에서 선택한 계정에 대해 활성화할 스캔 유형을 선택합니다. 다음 옵션 중에서 선택할 수 있습니다.
   - 전체 스캔 — 모든 스캔 유형을 활성화합니다.
   - EC2 스캔 — Amazon EC2 인스턴스 스캔을 활성화합니다.
   - ECR 컨테이너 스캔 — ECR 컨테이너 이미지 스캔을 활성화합니다.
   - AWS Lambda표준 스캔 — Lambda 함수 스캔을 활성화합니다.
7. (권장) 특정 구성원에 대한 스캔을 활성화하려는 각 지역에서 이 단계를 반복합니다.

AWS Organizations관리 계정이 Amazon Inspector의 관리자를 위임한 경우, 자신의 계정을 회원으로 활성화하고 자신의 계정에 대한 스캔 세부 정보를 볼 수 있습니다.

멤버 계정으로 스캔을 활성화하려면

1. 자신의 계정에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 연 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 스캔을 활성화하려는 지역을 지정합니다.
3. 탐색 창의 설정에서 계정 관리를 선택합니다.
4. 계정 관리 페이지에서 계정의 확인란을 선택합니다.
5. 활성화 메뉴에서 활성화할 스캔 유형을 선택합니다. 다음 옵션 중에서 선택할 수 있습니다.
   - 전체 스캔 — 모든 스캔 유형을 활성화합니다.

- EC2 스캔 — Amazon EC2 인스턴스 스캔을 활성화합니다.
- ECR 컨테이너 스캔 — ECR 컨테이너 이미지 스캔을 활성화합니다.
- AWS Lambda표준 스캔 — Lambda 함수 스캔을 활성화합니다.

6. (권장) 스캔을 활성화하려는 각 지역에서 이 단계를 반복합니다.

Amazon Inspector에서 멤버 계정 연결 해제

다음 절차는 멤버 계정의 연결 해제 방법을 보여줍니다. 연결이 끊긴 멤버 계정은 AWS Organizations 조직에 독립형 Amazon Inspector 계정으로 남아 있습니다. Amazon Inspector의 위임 관리자는 더 이상 이러한 계정에 대해 Amazon Inspector를 활성화하고 관리할 권한이 없습니다. 연결이 끊긴 계정을 나중에 다시 구성원으로 추가할 수 있습니다.

> **Note**
> 계정을 분리해도 해당 계정에 대한 Amazon Inspector 스캔은 비활성화되지 않습니다.

Console

콘솔을 사용하여 멤버 계정의 연결을 해제하는 방법

1. 위임 관리자 계정에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 연 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 하나 이상의 회원 계정을 연결 해제하려는 지역을 지정합니다.
3. 탐색 창의 설정에서 계정 관리를 선택합니다.
4. 계정 관리 페이지에서 연결을 해제할 각 계정의 확인란을 선택합니다.
5. 작업 메뉴에서 계정 연결 해제를 선택합니다.
6. (권장) 계정 연결을 해제하려는 각 지역에서 이 단계를 반복합니다.

API

API를 사용하여 멤버 계정의 연결을 해제하는 방법

DisassociateMember API 작업을 실행합니다. 요청 시 연결 해제하려는 계정 ID를 제공하십시오.

Amazon Inspector 위임 관리자 제거

Amazon Inspector의 위임 관리자를 새로 할당해야 하는 경우 기존의 위임 관리자를 관리 계정에서 제거할 수 있습니다. AWS Organizations

위임된 관리자를 제거해도 해당 계정이나 조직 구성원 계정에서 Amazon Inspector가 비활성화되지는 않습니다. 조직 내 계정은 독립 실행형 계정으로 변환되며 위임된 관리자가 관리하기 전의 스캔 설정을 유지합니다.

위임 관리자를 제거하려면

1. AWS Organizations 관리 계정을 사용하여 AWS Management Console에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 연 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 위임된 관리자를 제거하려는 지역을 지정합니다.
3. 탐색 창의 설정에서 계정 관리를 선택합니다.
4. 위임된 관리자 섹션에서 제거를 선택한 다음 작업을 확인합니다.
5. 이 위임된 관리자를 등록한 각 지역에서 이 단계를 반복합니다.

Amazon Inspector의 위임 관리자를 새로 추가할 때는 조직 구성원을 새 관리자 계정에 수동으로 연결해야 합니다. 다음 단계를 사용하여 조직 구성원을 새 관리자 계정에 연결할 수 있습니다.

멤버를 새 위임 관리자와 연결하려면

1. 위임 관리자 계정을 사용하여 AWS Management Console에 로그인합니다.
2. https://console.aws.amazon.com/inspector/v2/home 에서 Amazon Inspector 콘솔을 연 다음 오른쪽 상단의 AWS 리전 선택기를 사용하여 구성원을 새로 위임된 관리자와 연결할 지역을 지정합니다.
3. 탐색 창의 설정에서 계정 관리를 선택합니다.
4. 상단 확인란을 사용하여 조직에 나열된 계정을 모두 선택합니다.
5. 작업 메뉴에서 멤버 추가를 선택합니다.
6. 구성원을 새로운 위임 관리자와 연결하려는 각 지역에서 이 단계를 반복합니다.

Amazon Inspector에서 사용량 및 비용 모니터링

Amazon Inspector 콘솔 및 API 작업을 사용하여 사용 환경에서 Amazon Inspector를 사용하는 데 드는 월별 비용을 예측할 수 있습니다. 다중 계정 환경의 Amazon Inspector 관리자인 경우 전체 환경의 총 비용과 각 멤버 계정의 비용 지표를 확인할 수 있습니다.

사용량 콘솔 사용

콘솔에서 Amazon Inspector의 사용량과 예상 비용을 평가할 수 있습니다.

사용량 통계에 액세스하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단에 있는 AWS 리전 선택기를 사용하여 비용을 모니터링할 리전을 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.

Amazon EC2 스캔

Amazon Inspector 사용 콘솔은 에이전트 기반 검사 및 에이전트 없는 검사에 대한 다음 지표를 추적합니다.

- 인스턴스 (평균) — Amazon Inspector는 적용 범위 시간을 사용하여 EC2 인스턴스 스캔에 필요한 평균 리소스 수를 계산합니다. 평균은 총 적용 범위 시간을 720시간(30일 기간의 시간 수)으로 나눈 값입니다.

- 적용 범위 시간 — Amazon EC2 스캔의 경우, 지난 30일 동안 Amazon Inspector가 계정 내 각 EC2 인스턴스에 대해 활성 적용 범위를 제공한 총 시간입니다. EC2 인스턴스의 경우, 적용 범위 시간은 Amazon Inspector에서 인스턴스를 발견한 시점부터 인스턴스가 종료 또는 중지되거나 태그에 의해 스캔에서 제외될 때까지의 시간입니다. 중지된 인스턴스를 다시 시작하거나 제외 태그를 제거하면 Amazon Inspector에서 적용을 재개하고 해당 인스턴스에 대한 적용 시간은 계속 누적됩니다.

CIS 인스턴스 스캔 — 계정 내 인스턴스에 대해 수행된 CIS 스캔의 총 수입니다.

Amazon 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 수행되는 스캔입니다. ECR 리포지토리를 연속 스캔하도록 구성한 경우 Amazon Inspector에서 새로운 일반적인 취약성 및 노출(CVE)을 데이터베이스에 추가하면 자동으로 재스캔이 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 AWS Billing 콘솔의 비용과 다를 수 있습니다.

사용량 페이지에서 Amazon Inspector가 비용을 계산하는 방법에 대해서는 다음 사항을 참고하세요.

- 사용 비용은 현재 리전에만 적용됩니다. 스캔 유형별 가격은 AWS 리전에 따라 다릅니다. 리전별 정확한 가격을 검토하려면 Amazon Inspector 요금을 참조하세요.

- 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.
  - EC2 스캔: 비용에는 지난 30일 동안 Amazon Inspector에서 적용한 평균 EC2 인스턴스 수가 반영됩니다.
  - ECR 컨테이너 스캔: 비용에는 지난 30일 동안의 최초 이미지 스캔 횟수 + 이미지 재스캔 횟수의 합이 반영됩니다.
  - 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 Inspector 스캔 유형을 활성화하면 해당 스캔 유형에 대한 15일 무료 평가판에 자동으로 등록됩니다. EC2 스캔, ECR 스캔, Lambda 표준 스캔 및 Lambda 코드 스캔 등의 스캔 유형별로 독립적인 무료 평가판이 있습니다.

> **Note**
>
> 무료 평가판은 CIS 스캔에는 적용되지 않습니다.

무료 평가판 사용 중에 스캔 유형을 비활성화하면 해당 스캔 유형에 대한 무료 평가판이 일시 중지됩니다. 해당 서비스를 다시 활성화하면 무료 평가판이 다시 시작되고 남은 기간 동안 무료 평가판을 사용할 수 있습니다.

Amazon Inspector의 보안

클라우드 AWS 보안이 최우선 과제입니다. AWS 고객은 가장 보안에 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 혜택을 누릴 수 있습니다.

보안은 기업과 기업 간의 공동 책임입니다. AWS 공동 책임 모델은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드 보안 - AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호하는 역할을 합니다 AWS 클라우드. AWS 또한 안전하게 사용할 수 있는 서비스를 제공합니다. Amazon Inspector에 적용되는 규정 준수 프로그램에 대해 자세히 알아보려면 규정 준수 프로그램별 범위 내AWS 서비스 규정 준수 참조하십시오.
- 클라우드에서의 보안 — 사용하는 AWS 서비스에 따라 책임이 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 Inspector의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든 모델을 실행하는 글로벌 인프라를 보호할 책임이 있습니다. AWS 클라우드이 인프라에서 호스팅되는 콘텐츠에 대한 제어를 유지하는 것은 사용자의 책임입니다. 사용하는 AWS 서비스 의 보안 구성

과 관리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시 FAQ를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은AWS 보안 블로그의 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하세요.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 AWS IAM Identity Center OR AWS Identity and Access Management (IAM) 을 사용하여 개별 사용자를 설정하는 것이 좋습니다. 이 방식을 사용하면 각 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 다중 인증(MFA)을 사용합니다.
- SSL/TLS를 사용하여 리소스와 통신할 수 있습니다. AWS TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다. AWS CloudTrail
- 포함된 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용하십시오 AWS 서비스.
- Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하여 Amazon S3에 저장된 민감한 데이터를 검색하고 보호합니다.
- 명령줄 인터페이스 또는 API를 AWS 통해 액세스할 때 FIPS 140-2로 검증된 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용하십시오. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 Federal Information Processing Standard(FIPS) 140-2를 참조하십시오.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon Inspector 또는 다른 곳에서 콘솔 AWS CLI, API 또는 AWS 서비스 SDK를 사용하여 작업하는 경우가 포함됩니다. AWS 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 보안 인증 정보를 URL에 포함시켜서는 안 됩니다.

주제

- 저장된 데이터 암호화
- 전송 중 암호화

저장된 데이터 암호화

Amazon Inspector는 기본적으로 AWS 암호화 솔루션을 사용하여 저장된 데이터를 안전하게 저장합니다. Amazon Inspector는 키 관리 서비스 () 에서 소유한 암호화 키를 AWS 사용하여 AWS Systems

Manager를 AWS 사용하여 수집한 리소스 인벤토리, Amazon ECR 이미지에서 파싱한 리소스 인벤토리, 생성된 보안 탐지 결과 등의 데이터를 암호화합니다.AWS KMS AWS 소유 키를 확인, 관리 또는 사용하거나 사용 여부를 감사할 수 없습니다. 하지만 데이터 암호화 키를 보호하기 위해 어떤 작업을 수행하거나 어떤 프로그램을 변경할 필요가 없습니다. 자세한 내용은 AWS 소유 키를 참조하세요.

Amazon Inspector를 비활성화하면 수집된 인벤토리 및 보안 결과와 같이 저장 또는 유지 관리되는 모든 리소스가 영구적으로 삭제됩니다.

결과 코드에 대한 저장 중 암호화

Amazon Inspector Lambda 코드 스캐닝의 경우, Amazon Inspector는 코드에 취약성이 있는지 검사하기 위해 협력합니다. CodeGuru 취약성이 감지되면 취약성이 포함된 코드 스니펫을 CodeGuru 추출하여 Amazon Inspector에서 액세스를 요청할 때까지 해당 코드를 저장합니다. 기본적으로 자체 키를 CodeGuru 사용하여 추출된 코드를 암호화하지만 암호화에 자체 고객 AWS KMS 관리 키를 사용하도록 Amazon Inspector를 구성할 수 있습니다. AWS

다음 워크플로우는 Amazon Inspector에서 사용자가 구성한 키를 사용하여 코드를 암호화하는 방법을 설명합니다.

1. 아마존 인스펙터 API를 사용하여 아마존 인스펙터에 AWS KMS 키를 제공합니다. UpdateEncryptionKey
2. Amazon Inspector는 키에 대한 정보를 다음 주소로 전달합니다. AWS KMS CodeGuru CodeGuru 나중에 사용할 수 있도록 정보를 저장합니다.
3. CodeGuru Amazon Inspector에서 구성한 AWS KMS 키에 대한 권한 부여를 요청합니다.
4. CodeGuru 키에서 암호화된 데이터 AWS KMS 키를 생성하여 저장합니다. 이 데이터 키는 에서 저장한 CodeGuru 코드 데이터를 암호화하는 데 사용됩니다.
5. Amazon Inspector는 코드 스캔에서 데이터를 요청할 때마다 권한 부여를 CodeGuru 사용하여 암호화된 데이터 키를 해독한 다음 해당 키를 사용하여 데이터를 복호화하여 검색할 수 있도록 합니다.

Lambda 코드 CodeGuru 스캔을 비활성화하면 권한 부여가 폐기되고 관련 데이터 키가 삭제됩니다.

고객 관리형 키를 사용한 코드 암호화에 대한 권한

암호화를 사용하려면 AWS KMS 작업에 대한 액세스를 허용하는 정책과 Amazon Inspector에 조건 키를 통해 해당 작업을 사용할 CodeGuru 권한을 부여하는 명령문이 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 정책에서 다음 작업을 수행할 수 있도록 허용해야 합니다.

- kms:CreateGrant
- kms:Decrypt
- kms:DescribeKey
- kms:GenerateDataKeyWithoutPlainText
- kms:Encrypt
- kms:RetireGrant

정책에 올바른 AWS KMS 권한이 있는지 확인한 후에는 Amazon Inspector에서 암호화에 키를 사용할 수 있도록 허용하는 설명을 첨부해야 합니다. CodeGuru 다음 정책 설명을 첨부합니다.

> **Note**
>
> 지역을 Amazon Inspector Lambda 코드 스캔이 활성화된 AWS 지역으로 바꾸십시오.

```
{
            "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. 정책에는 JSON 형식이 사용됩니다. 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 검색 결과와 동일한 AWS 지역의 AWS KMS 키 또는 다중 지역 키가 필요합니다. 계정의 기존 대칭 키를 사용하거나 AWS 관리 콘솔 또

는 API를 사용하여 대칭 고객 관리 키를 생성할 수 있습니다. AWS KMS 자세한 내용은 사용 설명서의 대칭 암호화 AWS KMS 키 만들기를 참조하십시오. AWS KMS

Amazon Inspector API를 사용하여 암호화 구성

Amazon Inspector 관리자로 로그인한 상태에서 Amazon Inspector API의 UpdateEncryptionKey작동을 암호화하기 위한 키를 설정하려면 API 요청에서 `kmsKeyId` 필드를 사용하여 사용하려는 AWS KMS 키의 ARN을 지정합니다. `scanType`에 `CODE`를 입력하고 `resourceType`에 `AWS_LAMBDA_FUNCTION`을 입력합니다.

UpdateEncryptionKeyAPI를 사용하여 Amazon Inspector가 암호화에 사용하는 AWS KMS 키를 뷰에서 확인할 수 있습니다.

> **Note**
>
> 고객 관리 키를 설정하지 않은 `GetEncryptionKey` 상태에서 사용을 시도하면 `ResourceNotFoundException` 오류가 반환되며, 이는 AWS 소유한 키가 암호화에 사용되고 있음을 의미합니다.

키를 삭제하거나 변경하면 Amazon CodeGuru Inspector에 대한 액세스를 거부하도록 정책이 설정되어 있습니다. 그렇지 않으면 코드 취약성 발견에 액세스할 수 없으며 계정에 대한 Lambda 코드 스캔이 실패합니다.

Amazon `ResetEncryptionKey` Inspector 검색 결과의 일부로 추출된 코드를 암호화하기 위해 AWS 소유 키를 다시 사용하는 데 사용할 수 있습니다.

전송 중 암호화

AWS AWS 내부 시스템과 다른 서비스 간에 전송되는 모든 데이터를 암호화합니다. AWS

인벤토리 수집을 위해 Systems Manager는 고객 소유의 EC2 인스턴스에서 원격 분석 데이터를 수집하여 평가를 위해 전송 계층 보안 (TLS) 보호 채널을 AWS 통해 다시 전송합니다. SSM이 전송 중인 데이터를 암호화하는 방법을 이해하려면 Systems Manager의 데이터 보호를 참조하세요.

마찬가지로, Security Hub로 전송되는 Amazon ECR 및 AWS Lambda 함수 스캔 결과는 TLS 보호 채널을 사용하여 암호화됩니다.

Amazon Inspector용 Identity and Access Management

AWS Identity and Access Management (IAM) 은 관리자가 리소스에 대한 액세스를 안전하게 AWS 서비스 제어할 수 있도록 도와줍니다. AWS IAM 관리자는 인증(로그인) 및 권한 부여(권한 보유)를 통해 Amazon Inspector 리소스를 사용할 수 있는 사용자를 제어합니다. IAM은 추가 AWS 서비스 비용 없이 사용할 수 있습니다.

주제

고객

사용 방법 AWS Identity and Access Management (IAM) 은 Amazon Inspector에서 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 IAM의 기본 개념을 이해하세요. 회사가 Amazon Inspector에서 IAM을 사용하는 방법에 대해 자세히 알아보려면 Amazon Inspector에서 IAM을 사용하는 방법 섹션을 참조하세요.

IAM 관리자 - IAM 관리자라면 Amazon Inspector에 대한 액세스 관리 정책 작성 방법을 자세히 알고 싶을 수도 있습니다. IAM에서 사용할 수 있는 Amazon Inspector 자격 증명 기반 정책 예제를 보려면 Amazon Inspector의 자격 증명 기반 정책 예 섹션을 참조하세요.

자격 증명을 통한 인증

인증은 ID 자격 증명을 AWS 사용하여 로그인하는 방법입니다. IAM 사용자로 인증 (로그인 AWS) 하거나 IAM 역할을 맡아 인증 (로그인) 해야 합니다. AWS 계정 루트 사용자

ID 소스를 통해 제공된 자격 증명을 사용하여 페더레이션 ID로 로그인할 수 있습니다. AWS AWS IAM Identity Center (IAM ID 센터) 사용자, 회사의 싱글 사인온 인증, Google 또는 Facebook 자격 증명이 페더레이션 ID의 예입니다. 연동 자격 증명으로 로그인할 때 관리자가 이전에 IAM 역할을 사용하여 ID 페더레이션을 설정했습니다. 페더레이션을 사용하여 액세스하는 경우 AWS 간접적으로 역할을 맡게 됩니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. 로그인에 대한 자세한 내용은 AWS 로그인 사용 설명서의 내 로그인 방법을 참조하십시오. AWS AWS 계정

AWS 프로그래밍 방식으로 액세스하는 경우 자격 증명을 사용하여 요청에 암호화 방식으로 서명할 수 있는 소프트웨어 개발 키트 (SDK) 와 명령줄 인터페이스 (CLI) 를 AWS 제공합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. 권장 방법을 사용하여 직접 요청에 서명하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 AWS API 요청 서명을 참조하십시오.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, AWS 계정의 보안을 강화하기 위해 다단계 인증 (MFA) 을 사용할 것을 권장합니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서의 다중 인증 및 IAM 사용자 설명서의 AWS에서 다중 인증(MFA) 사용을 참조하세요.

AWS 계정 루트 사용자

계정을 AWS 계정만들 때는 먼저 계정의 모든 AWS 서비스 리소스에 대한 완전한 액세스 권한을 가진 하나의 로그인 ID로 시작합니다. 이 ID를 AWS 계정 루트 사용자라고 하며, 계정을 만들 때 사용한 이메일 주소와 비밀번호로 로그인하여 액세스할 수 있습니다. 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 태스크의 전체 목록은 IAM 사용자 설명서의 루트 사용자 보안 인증이 필요한 태스크를 참조하세요.

연동 보안 인증

가장 좋은 방법은 관리자 액세스가 필요한 사용자를 비롯한 수동 AWS 서비스 사용자가 ID 공급자와의 페더레이션을 사용하여 임시 자격 증명을 사용하여 액세스하도록 하는 것입니다.

페더레이션 ID는 기업 사용자 디렉토리, 웹 ID 공급자, Identity Center 디렉터리의 사용자 또는 ID 소스를 통해 제공된 자격 증명을 사용하여 액세스하는 AWS 서비스 모든 사용자를 말합니다. AWS

Directory Service페더레이션 ID에 AWS 계정액세스하면 이들이 역할을 맡고 역할은 임시 자격 증명을 제공합니다.

중앙 집중식 액세스 관리를 위해 AWS IAM Identity Center을 사용하는 것이 좋습니다. IAM Identity Center에서 사용자 및 그룹을 생성하거나 자체 ID 소스의 사용자 및 그룹 집합에 연결하고 동기화하여 모든 사용자 및 애플리케이션에서 사용할 수 있습니다. AWS 계정 IAM Identity Center에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서에서 IAM Identity Center란 무엇입니까?를 참조하세요.

IAM 사용자 및 그룹

IAM 사용자는 단일 사용자 또는 애플리케이션에 대한 특정 권한을 AWS 계정 가진 사용자 내 자격 증명입니다. 가능하면 암호 및 액세스 키와 같은 장기 자격 증명이 있는 IAM 사용자를 생성하는 대신 임시 자격 증명을 사용하는 것이 좋습니다. 하지만 IAM 사용자의 장기 자격 증명이 필요한 특정 사용 사례가 있는 경우 액세스 키를 교체하는 것이 좋습니다. 자세한 내용은 IAM 사용자 설명서의 장기 보안 인증이 필요한 사용 사례의 경우 정기적으로 액세스 키 교체를 참조하세요.

IAM 그룹은 IAM 사용자 컬렉션을 지정하는 보안 인증입니다. 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, IAMAdmins라는 그룹이 있고 이 그룹에 IAM 리소스를 관리할 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 인증 정보만 제공합니다. 자세한 정보는 IAM 사용자 설명서의 IAM 사용자를 만들어야 하는 경우(역할이 아님)를 참조하세요.

IAM 역할

IAM 역할은 특정 권한을 가진 사용자 AWS 계정 내의 자격 증명입니다. IAM 사용자와 유사하지만, 특정 개인과 연결되지 않습니다. 역할을 AWS Management Console 전환하여 에서 일시적으로 IAM 역할을 맡을 수 있습니다. AWS CLI 또는 AWS API 작업을 호출하거나 사용자 지정 URL을 사용하여 역할을 수임할 수 있습니다. 역할 사용 방법에 대한 자세한 정보는 IAM 사용자 설명서의 IAM 역할 사용을 참조하세요.

임시 보안 인증 정보가 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 연동 자격 증명에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 연동 자격 증명이 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 역할에 대한 자세한 내용은 IAM 사용자 설명서의 Creating a role for a third-party Identity

Provider(서드 파티 자격 증명 공급자의 역할 만들기) 부분을 참조하세요. IAM 자격 증명 센터를 사용하는 경우 권한 집합을 구성합니다. 인증 후 아이덴티티가 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 세트를 IAM의 역할과 연관 짓습니다. 권한 세트에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 권한 세트 섹션을 참조하세요.

- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할은 IAM 역할을 수임하여 특정 작업에 대한 다양한 권한을 임시로 받을 수 있습니다.
- 크로스 계정 액세스 - IAM 역할을 사용하여 다른 계정의 사용자(신뢰할 수 있는 보안 주체)가 내 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 계정 간 액세스를 부여하는 기본적인 방법입니다. 그러나 일부 AWS 서비스경우에는 역할을 프록시로 사용하는 대신 정책을 리소스에 직접 연결할 수 있습니다. 교차 계정 액세스를 위한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용자 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.
- 서비스 간 액세스 — 일부는 다른 AWS 서비스서비스의 기능을 AWS 서비스 사용합니다. 예를 들어 서비스에서 직접 호출을 수행하면 일반적으로 해당 서비스는 Amazon EC2에서 애플리케이션을 실행하거나 Amazon S3에 객체를 저장합니다. 서비스는 직접적으로 호출하는 보안 주체의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 태스크를 수행할 수 있습니다.
  - 순방향 액세스 세션 (FAS) — IAM 사용자 또는 역할을 사용하여 작업을 수행하는 경우 보안 AWS 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는 전화를 거는 주체의 권한을 다운스트림 AWS 서비스서비스에 AWS 서비스 요청하기 위한 요청과 결합하여 사용합니다. FAS 요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 전달 액세스 세션을 참조하세요.
  - 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 태스크를 수행하기 위해 맡는 IAM 역할입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 참조하세요.
  - 서비스 연결 역할 — 서비스 연결 역할은 에 연결된 서비스 역할의 한 유형입니다. AWS 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM 관리자는 서비스 링크 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.
- Amazon EC2에서 실행되는 애플리케이션 — IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 API 요청을 AWS CLI 하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. AWS 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있게 하려면 인스턴스에 연결된 인스턴스 프로필을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 보안

인증 정보를 얻을 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하세요.

IAM 역할을 사용할지 또는 IAM 사용자를 사용할지를 알아보려면 IAM 사용자 설명서의 IAM 역할(사용자 대신)을 생성하는 경우를 참조하세요.

정책을 사용한 액세스 관리

정책을 생성하고 이를 AWS ID 또는 리소스에 AWS 연결하여 액세스를 제어할 수 있습니다. 정책은 ID 또는 리소스와 연결될 때 AWS 해당 권한을 정의하는 객체입니다. AWS 주도자 (사용자, 루트 사용자 또는 역할 세션) 가 요청할 때 이러한 정책을 평가합니다. 정책의 권한이 요청 허용 또는 거부 여부를 결정합니다. 대부분의 정책은 JSON 문서로 AWS 저장됩니다. JSON 정책 문서의 구조와 콘텐츠에 대한 자세한 정보는 IAM 사용자 설명서의 JSON 정책 개요를 참조하세요.

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 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 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다. 그런 다음 관리자가 IAM 정책을 역할에 추가하고, 사용자가 역할을 수임할 수 있습니다.

IAM 정책은 작업을 수행하기 위해 사용하는 방법과 상관없이 작업에 대한 권한을 정의합니다. 예를 들어, `iam:GetRole` 작업을 허용하는 정책이 있다고 가정합니다. 해당 정책을 사용하는 사용자는 AWS Management Console, AWS CLI, 또는 AWS API에서 역할 정보를 가져올 수 있습니다.

보안 인증 기반 정책

보안 인증 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 보안 인증에 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 사용자 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 내 여러 사용자, 그룹 및 역할에 연결할 수 있는 독립형 정책입니다. AWS 계정관리형 정책에는 AWS 관리형 정책과 고객 관리형 정책이 포함됩니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용자 설명서의 관리형 정책과 인라인 정책의 선택을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 리소스 기반 정책의 예는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책입니다. 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 태스크를 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 연동 사용자 등이 포함될 수 있습니다. AWS 서비스

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. IAM의 AWS 관리형 정책은 리소스 기반 정책에 사용할 수 없습니다.

액세스 제어 목록(ACL)

액세스 제어 목록(ACL)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACL은 JSON 정책 설명서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

ACL을 지원하는 서비스의 예로는 아마존 S3와 아마존 VPC가 있습니다. AWS WAF ACL에 대해 자세히 알아보려면 Amazon Simple Storage Service 개발자 안내서의 ACL(액세스 제어 목록) 개요 섹션을 참조하세요.

기타 정책 타입

AWS 일반적이지 않은 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 타입에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 – 권한 경계는 자격 증명 기반 정책에 따라 IAM 엔터티(IAM 사용자 또는 역할)에 부여할 수 있는 최대 권한을 설정하는 고급 특성입니다. 개체에 대한 권한 경계를 설정할 수 있습니다. 그 결과로 얻는 권한은 개체의 보안 인증 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 보안 주체로 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 정보는 IAM 사용자 설명서의 IAM 엔터티에 대한 권한 경계를 참조하세요.
- 서비스 제어 정책 (SCP) - SCP는 조직 또는 조직 단위 (OU) 에 대한 최대 권한을 지정하는 JSON 정책입니다. AWS Organizations AWS Organizations 사업체가 소유한 여러 AWS 계정 개를 그룹화하고 중앙에서 관리하는 서비스입니다. 조직에서 모든 특성을 활성화할 경우 서비스 제어 정책 (SCP)을 임의의 또는 모든 계정에 적용할 수 있습니다. SCP는 구성원 계정의 엔티티 (각 엔티티 포함) 에 대한 권한을 제한합니다. AWS 계정 루트 사용자조직 및 SCP에 대한 자세한 정보는 AWS Organizations 사용 설명서의 SCP 작동 방식을 참조하세요.

- 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 자격 증명 기반 정책의 교집합과 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 정보는 IAM 사용자 설명서의 세션 정책을 참조하세요.

여러 정책 타입

여러 정책 타입이 요청에 적용되는 경우 결과 권한은 이해하기가 더 복잡합니다. 여러 정책 유형이 관련된 경우 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하십시오.

Amazon Inspector에서 IAM을 사용하는 방법

IAM을 사용하여 Amazon Inspector에 대한 액세스를 관리하기 전에 Amazon Inspector에서 사용할 수 있는 IAM 기능에 대해 알아봅니다.

Amazon Inspector에서 사용할 수 있는 IAM 기능

IAM 특성	Amazon Inspector 지원
ID 기반 정책	예
리소스 기반 정책	아니요
정책 작업	예
정책 리소스	예
정책 조건 키(서비스별)	예
ACL	아니요
ABAC(정책 내 태그)	부분
임시 보안 인증 정보	예
보안 주체 권한	예
서비스 역할	아니요

IAM 특성	Amazon Inspector 지원
서비스 링크 역할	예

Amazon Inspector 및 AWS 서비스 기타 제품이 대부분의 IAM 기능과 어떻게 작동하는지 자세히 알아보려면 IAM 사용 설명서의 IAM과 연동되는 기능을AWS 서비스 참조하십시오.

Amazon Inspector의 자격 증명 기반 정책

ID 기반 정책 지원	예

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 ID에 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 사용자 설명서의 IAM 정책 생성 (Creating IAM policies)을 참조합니다.

IAM ID 기반 정책을 사용하면 허용되거나 거부되는 태스크와 리소스 뿐만 아니라 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. 자격 증명 기반 정책에서는 보안 주체가 연결된 사용자 또는 역할에 적용되므로 보안 주체를 지정할 수 없습니다. JSON 정책에서 사용하는 모든 요소에 대해 알아보려면 IAM 사용자 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 예는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책입니다. 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 태스크를 수행할 수 있는지를 정의합니

다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 연동 사용자 등이 포함될 수 있습니다. AWS 서비스

크로스 계정 액세스를 활성화하려는 경우 전체 계정이나 다른 계정의 IAM 개체를 리소스 기반 정책의 보안 주체로 지정할 수 있습니다. 리소스 기반 정책에 크로스 계정 보안 주체를 추가하는 것은 트러스트 관계 설정의 절반밖에 되지 않는다는 것을 유념합니다. 보안 주체와 리소스가 다른 AWS 계정경우 신뢰할 수 있는 계정의 IAM 관리자는 보안 주체 개체 (사용자 또는 역할) 에게 리소스에 액세스할 수 있는 권한도 부여해야 합니다. 개체에 자격 증명 기반 정책을 연결하여 권한을 부여합니다. 하지만 리소스 기반 정책이 동일 계정의 보안 주체에 액세스를 부여하는 경우 추가 자격 증명 기반 정책이 필요하지 않습니다. 자세한 정보는 IAM 사용자 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.

Amazon Inspector에 대한 정책 작업

정책 작업 지원	예

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 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 관련 AWS API 작업과 이름이 같습니다. 일치하는 API 작업이 없는 권한 전용 작업 같은 몇 가지 예외도 있습니다. 정책에서 여러 작업이 필요한 몇 가지 작업도 있습니다. 이러한 추가 작업을 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 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

`Resource` JSON 정책 요소는 작업이 적용되는 하나 이상의 객체를 지정합니다. 보고서에는 `Resource` 또는 `NotResource` 요소가 반드시 추가되어야 합니다. 모범 사례에 따라 Amazon 리소스 이름(ARN)을 사용하여 리소스를 지정합니다. 리소스 수준 권한이라고 하는 특정 리소스 유형을 지원하는 작업에 대해 이 작업을 수행할 수 있습니다.

작업 나열과 같이 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": "*"
```

Amazon Inspector리소스 유형 및 해당 ARN의 목록을 보려면 서비스 승인 참조에서 Amazon Inspector에서 정의한 리소스를 참조하세요. 각 리소스의 ARN을 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 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 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 적음 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 (은)는 논리적 AND 작업을 사용하여 평가합니다. 단일 조건 키에 여러 값을 지정하는 경우 는 논리적 OR 연산을 사용하여 조건을 AWS 평가합니다. 명령문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어, IAM 사용자에게 IAM 사용자 이름으로 태그가 지정된 경우에만 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 IAM 정책 요소: 변수 및 태그를 참조하세요.

AWS 글로벌 조건 키 및 서비스별 조건 키를 지원합니다. 모든 AWS 글로벌 조건 키를 보려면 IAM 사용 AWS 설명서의 글로벌 조건 컨텍스트 키를 참조하십시오.

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.

Amazon Inspector의 ACL

ACL 지원	아니요

액세스 제어 목록(ACL)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACL은 JSON 정책 설명서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

Amazon Inspector를 사용한 ABAC

ABAC(정책 내 태그) 지원	부분

속성 기반 액세스 제어(ABAC)는 속성을 기반으로 권한을 정의하는 권한 부여 전략입니다. AWS에서는 이러한 속성을 태그라고 합니다. IAM 엔티티 (사용자 또는 역할) 및 여러 AWS 리소스에 태그를 첨부할

수 있습니다. ABAC의 첫 번째 단계로 개체 및 리소스에 태그를 지정합니다. 그런 다음 보안 주체의 태그가 액세스하려는 리소스의 태그와 일치할 때 작업을 허용하도록 ABAC 정책을 설계합니다.

ABAC는 빠르게 성장하는 환경에서 유용하며 정책 관리가 번거로운 상황에 도움이 됩니다.

태그를 기반으로 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 예(Yes)입니다. 서비스가 일부 리소스 유형에 대해서만 세 가지 조건 키를 모두 지원하는 경우 값은 부분적(Partial)입니다.

ABAC에 대한 자세한 정보는 IAM 사용자 설명서의 ABAC란 무엇인가요?를 참조하세요. ABAC 설정 단계가 포함된 자습서를 보려면 IAM 사용자 설명서의 속성 기반 액세스 제어(ABAC) 사용을 참조하세요.

Amazon Inspector에서 임시 보안 인증 정보 사용

임시 보안 인증 정보 지원	예

임시 자격 증명을 사용하여 로그인하면 작동하지 AWS 서비스 않는 것도 있습니다. 임시 자격 증명을 사용하는 방법을 AWS 서비스 비롯한 추가 정보는 IAM 사용 설명서의 IAM과AWS 서비스 연동되는 내용을 참조하십시오.

사용자 이름과 암호를 제외한 다른 방법을 AWS Management Console 사용하여 로그인하면 임시 자격 증명을 사용하는 것입니다. 예를 들어 회사의 SSO (Single Sign-On) 링크를 AWS 사용하여 액세스하는 경우 이 프로세스에서 자동으로 임시 자격 증명을 생성합니다. 또한 콘솔에 사용자로 로그인한 다음 역할을 전환할 때 임시 보안 인증 정보를 자동으로 생성합니다. 역할 전환에 대한 자세한 정보는 IAM 사용자 설명서의 역할로 전환(콘솔)을 참조하세요.

또는 API를 사용하여 임시 자격 증명을 수동으로 생성할 수 있습니다 AWS CLI . AWS 그런 다음 해당 임시 자격 증명을 사용하여 액세스할 수 AWS있습니다. AWS 장기 액세스 키를 사용하는 대신 임시 자격 증명을 동적으로 생성할 것을 권장합니다. 자세한 정보는 IAM의 임시 보안 인증 정보 섹션을 참조하세요.

Amazon Inspector에 대한 교차 서비스 보안 주체 권한

전달 액세스 세션(FAS) 지원	예

IAM 사용자 또는 역할을 사용하여 작업을 수행하는 AWS경우 보안 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는 전화를 거는 주체의 권한을 다운스트림 서비스에 AWS 서비스 요청하라는 요청과 결합하여 사용합니다. AWS 서비스 FAS 요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 전달 액세스 세션을 참조하세요.

Amazon Inspector의 서비스 역할

서비스 역할 지원	아니요

서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하기 위해 수임하는 IAM role(IAM 역할)입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 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 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM 관리자는 서비스 링크 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.

서비스 연결 역할 생성 또는 관리에 대한 자세한 내용은 IAM으로 작업하는AWS 서비스를 참조하세요. 서비스 연결 역할 열에서 Yes가 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 AWS API를 사용하여 작업을 수행할 수 없습니다. 사용자에게 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다. 그런 다음 관리자가 IAM 정책을 역할에 추가하고, 사용자가 역할을 맡을 수 있습니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM ID 기반 정책을 생성하는 방법을 알아보려면 IAM 사용자 설명서의 IAM 정책 생성을 참조하세요.

각 리소스 유형에 대한 ARN 형식을 포함하여 Amazon Inspector에서 정의한 작업 및 리소스 유형에 대한 자세한 내용은 서비스 승인 참조에서 Amazon Inspector에 사용되는 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. 자격 증명 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따릅니다.

- AWS 관리형 정책으로 시작하고 최소 권한 권한으로 이동 — 사용자와 워크로드에 권한을 부여하려면 여러 일반적인 사용 사례에 권한을 부여하는 AWS 관리형 정책을 사용하세요. 해당 내용은 에서 사용할 수 있습니다. AWS 계정사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 더 줄이는 것이 좋습니다. 자세한 정보는 IAM 사용자 설명서의 AWS managed policies(관리형 정책) 또는 AWS managed policies for job functions(직무에 대한 관리형 정책)를 참조하세요.
- 최소 권한 적용 – IAM 정책을 사용하여 권한을 설정하는 경우 작업을 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 least-privilege permissions(최소 권한)으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. IAM을 사용하여 권한을 적용하는 방법에 대한 자세

한 정보는 IAM 사용자 설명서에 있는 Policies and permissions in IAM(IAM의 정책 및 권한)을 참조하세요.

- Use conditions in IAM policies to further restrict access(IAM 정책의 조건을 사용하여 액세스 추가 제한) – 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어 SSL을 사용하여 모든 요청을 전송해야 한다고 지정하는 정책 조건을 생성할 수 있습니다. 예를 AWS 서비스들어 특정 작업을 통해 서비스 작업을 사용하는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 AWS CloudFormation있습니다. 자세한 정보는 IAM 사용자 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.
- IAM Access Analyzer를 통해 IAM 정책을 검증하여 안전하고 기능적인 권한 보장 – IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 신규 및 기존 정책을 검증합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 권장 사항을 제공하여 안전하고 기능적인 정책을 생성하도록 돕습니다. 자세한 정보는 IAM 사용자 설명서의 IAM Access Analyzer policy validation(IAM Access Analyzer 정책 검증)을 참조하세요.
- 멀티 팩터 인증 (MFA) 필요 - IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 AWS 계정 MFA를 활성화하십시오. API 작업을 직접적으로 호출할 때 MFA가 필요하면 정책에 MFA 조건을 추가합니다. 자세한 정보는 IAM 사용자 설명서의 Configuring MFA-protected API access(MFA 보호 API 액세스 구성)를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용자 설명서의 IAM의 보안 모범 사례를 참조하세요.

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 ID 기반 정책을 만들면 콘솔이 해당 정책에 연결된 개체(사용자 또는 역할)에 대해 의도대로 작동하지 않습니다.

또는 API만 호출하는 사용자에게 최소 콘솔 권한을 허용할 필요는 없습니다. AWS CLI AWS 그 대신, 수행하려는 API 작업과 일치하는 작업에만 액세스할 수 있도록 합니다.

사용자와 역할이 Amazon Inspector 콘솔을 계속 사용할 수 있도록 하려면 Amazon Inspector `ReadOnly` AWS 또는 관리형 정책도 `ConsoleAccess` 엔티티에 연결하십시오. 자세한 내용은 IAM 사용 설명서의 사용자에게 권한 추가를 참조하세요.

사용자가 자신의 고유한 권한을 볼 수 있도록 허용

이 예제는 IAM 사용자가 자신의 사용자 자격 증명에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 API를 사용하여 프로그래밍 방식으로 이 작업을 완료할 수 있는 AWS CLI 권한이 포함됩니다. AWS

```
{
    "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 아마존 인스펙터의 관리형 정책

AWS 관리형 정책은 에서 생성하고 관리하는 독립 실행형 정책입니다. AWS AWS 관리형 정책은 많은 일반 사용 사례에 대한 권한을 제공하도록 설계되었으므로 사용자, 그룹 및 역할에 권한을 할당하기 시작할 수 있습니다.

AWS 관리형 정책은 모든 AWS 고객이 사용할 수 있으므로 특정 사용 사례에 대해 최소 권한 권한을 부여하지 않을 수도 있다는 점에 유의하세요. 사용 사례에 고유한 고객 관리형 정책을 정의하여 권한을 줄이는 것이 좋습니다.

관리형 정책에 정의된 권한은 변경할 수 없습니다. AWS AWS 관리형 정책에 정의된 권한을 업데이트하는 경우 AWS 해당 업데이트는 정책이 연결된 모든 주체 ID (사용자, 그룹, 역할) 에 영향을 미칩니다. AWS 새 API 작업이 시작되거나 기존 서비스에 새 AWS 서비스 API 작업을 사용할 수 있게 되면 AWS 관리형 정책을 업데이트할 가능성이 가장 높습니다.

자세한 내용은 IAM 사용자 설명서의 AWS 관리형 정책을 참조하세요.

AWS 관리형 정책: AmazonInspector2FullAccess

`AmazonInspector2FullAccess` 정책을 IAM 보안 인증에 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 Inspector에서 서비스 연결 역할인 `AmazonInspector2AgentlessServiceRole`을 생성할 수 있도록 허용합니다. 이 권한은 Amazon Inspector에서 Amazon EC2 인스턴스, Amazon ECR 리포지토리 및 컨테이너 이미지에 대한 정보를 검색하고, VPC 네트워크를 분석하며, 조직과 연결된 계정을 설명하는 등의 작업을 수행하는 데 필요합니다. 자세한 설명은 Amazon Inspector에 서비스 연결 역할 사용 섹션을 참조하세요.
- `organizations` - 관리자가 Amazon Inspector를 AWS Organizations의 조직에 사용할 수 있도록 허용합니다. AWS Organizations에서 Amazon Inspector에 대한 신뢰할 수 있는 액세스를 활성화한 후, 위임된 관리자 계정의 구성원은 조직 전체의 설정을 관리하고 결과를 볼 수 있습니다.
- `codeguru-security`— 관리자가 Amazon Inspector를 사용하여 정보 코드 스니펫을 검색하고 보안에서 저장한 코드의 암호화 설정을 변경할 수 있습니다. CodeGuru 자세한 설명은 결과 코드에 대한 저장 중 암호화 섹션을 참조하세요.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": "inspector2:*",
   "Resource": "*"
```

```
  },
  {
   "Effect": "Allow",
   "Action": [
    "codeguru-security:BatchGetFindings",
    "codeguru-security:GetAccountConfiguration",
    "codeguru-security:UpdateAccountConfiguration"
   ],
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": "inspector2.amazonaws.com"
    }
   }
  },
  {
   "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 보안 인증에 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`— 조직의 Amazon Inspector 적용 범위에 대한 세부 정보를 볼 AWS Organizations 수 있습니다.
- `codeguru-security`— 보안에서 코드 스니펫을 검색할 수 있습니다. CodeGuru 또한 CodeGuru 보안에 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 IAM 엔터티에 연결할 수 있습니다. Amazon EC2 인스턴스에 인스턴스의 CIS 스캔을 실행할 권한을 부여하는 역할에 이 정책을 연결해야 합니다. IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 API 요청을 하는 애플리케이션의 임시 자격 증명을 관리

할 수 있습니다. AWS CLI AWS 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있게 하려면 인스턴스에 연결된 인스턴스 프로필을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 보안 인증 정보를 얻을 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하세요.

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `inspector2`— CIS 스캔을 실행하는 데 사용된 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 IAM 엔터티에 연결할 수 없습니다. 이 정책을 서비스 연결 역할에 연결하면 Amazon Inspector가 사용자를 대신하여 작업을 수행할 수 있습니다. 자세한 설명은 Amazon Inspector에 서비스 연결 역할 사용 섹션을 참조하세요.

AWS 관리형 정책: AmazonInspector2AgentlessServiceRolePolicy

`AmazonInspector2AgentlessServiceRolePolicy` 정책을 IAM 엔터티에 연결할 수 없습니다. 이 정책을 서비스 연결 역할에 연결하면 Amazon Inspector가 사용자를 대신하여 작업을 수행할 수 있습니다. 자세한 설명은 Amazon Inspector에 서비스 연결 역할 사용 섹션을 참조하세요.

Amazon Inspector의 관리형 정책 업데이트 AWS

이 서비스가 변경 사항을 추적하기 시작한 이후 Amazon Inspector의 AWS 관리형 정책 업데이트에 대한 세부 정보를 확인하십시오. 이 페이지의 변경 사항에 대한 자동 알림을 받아보려면 Amazon Inspector 문서 기록 페이지에서 RSS 피드를 구독하세요.

변경 사항	설명	날짜
AmazonInspector2 ManagedCisPolicy — 새 정책	Amazon Inspector는 인스턴스에서 CIS 스캔을 허용하기 위해 인스턴스 프로필의 일부로 사용할 수 있는 새로운 관리형 정책을 추가했습니다.	2024년 1월 23일
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 대상 인스턴스에서 CIS 스캔을 시작할 수 있는 새로운 권한을 추가했습니다.	2024년 1월 23일
AmazonInspector2 — 새 정책 AgentlessServiceRolePolicy	에이전트 없는 EC2 인스턴스 스캔을 허용하도록 Amazon Inspector에 새 서비스 연결 역할 정책을 추가했습니다.	2023년 11월 27일
AmazonInspector2 ReadOnlyAccess — 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
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector에서 Elastic Load Balancing 대상 그룹에 속하는 Amazon EC2 인스턴스의 네트워크 구성을 스캔할 수 있는 새로운 권한이 추가되었습니다.	2023년 8월 31일

변경 사항	설명	날짜
AmazonInspector2 ReadOnlyAccess — 기존 정책 업데이트	Amazon Inspector에서 읽기 전용 사용자가 리소스에 대한 Software Bill of Materials (SBOM)를 내보낼 수 있는 새로운 권한을 추가했습니다.	2023년 6월 29일
AmazonInspector2 ReadOnlyAccess — 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
AmazonInspector2 FullAccess — 기존 정책 업데이트	Amazon Inspector에서 사용자가 Lambda 코드 스캔 결과의 코드를 암호화하도록 고객 관리형 KMS 키를 구성할 수 있는 새로운 권한을 추가했습니다.	2023년 6월 13일
AmazonInspector2 ReadOnlyAccess — 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
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Lambda 스캔을 활성화할 때 Amazon Inspector가 사용자 계정에 서비스 연결 채널을 AWS CloudTrail 생성할 수 있는 새로운 권한을 추가했습니다. 이렇게 하면 Amazon Inspector에서 사용자 계정의 CloudTrail 이벤트를 모니터링할 수 있습니다.	2023년 4월 30일

변경 사항	설명	날짜
AmazonInspector2 FullAccess — 기존 정책 업데이트	Amazon Inspector에서 사용자가 Lambda 코드 스캔의 코드 취약성 결과에 대한 세부 정보를 검색할 수 있는 새로운 권한을 추가했습니다.	2023년 4월 21일
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 Amazon EC2 심층 검사를 위해 고객이 정의한 사용자 지정 경로에 대한 정보를 Amazon EC2 Systems Manager에 전송할 수 있는 새로운 권한을 추가했습니다.	2023년 4월 17일
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Lambda 스캔을 활성화할 때 Amazon Inspector가 사용자 계정에 서비스 연결 채널을 AWS CloudTrail 생성할 수 있는 새로운 권한을 추가했습니다. 이렇게 하면 Amazon Inspector에서 사용자 계정의 CloudTrail 이벤트를 모니터링할 수 있습니다.	2023년 4월 30일

변경 사항	설명	날짜
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector가 함수의 개발자 코드 스캔을 요청하고 Amazon Security로부터 스캔 데이터를 수신할 수 AWS Lambda 있는 새로운 권한을 추가했습니다. CodeGuru 또한 Amazon Inspector에서 IAM 정책을 검토할 수 있는 권한이 추가되었습니다. Amazon Inspector는 이 정보를 사용하여 Lambda 함수의 코드 취약성을 스캔합니다.	2023년 2월 28일
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Amazon CloudWatch Inspector에서 함수가 마지막으로 호출된 시기에 AWS Lambda 대한 정보를 검색할 수 있도록 하는 새로운 설명을 추가했습니다. Amazon Inspector는 이 정보를 사용하여 사용자 환경에서 지난 90일 동안 활성화된 Lambda 함수에 초점을 맞추어 스캔을 수행합니다.	2023년 2월 20일
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector는 Amazon Inspector에서 각 함수와 관련된 각 레이어 버전을 포함하여 함수에 AWS Lambda 대한 정보를 검색할 수 있는 새로운 설명을 추가했습니다. Amazon Inspector는 이 정보를 사용하여 Lambda 함수의 보안 취약성을 스캔합니다.	2022년 11월 28일

변경 사항	설명	날짜
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector에서 SSM 연결 실행을 설명할 수 있는 새로운 작업이 추가되었습니다. 또한 Amazon Inspector에서 `AmazonInspector2` 소유 SSM 문서와 SSM 연결을 생성, 업데이트, 삭제 및 시작할 수 있도록 리소스 범위가 추가되었습니다.	2022년 8월 31일
AmazonInspector2 기존 정책 ServiceRolePolicy 업데이트	Amazon Inspector는 Amazon Inspector가 다른 파티션의 소프트웨어 인벤토리를 수집할 수 있도록 정책의 리소스 범위를 업데이트했습니다. AWS	2022년 8월 12일
AmazonInspector2 ServiceRolePolicy — 기존 정책 업데이트	Amazon Inspector에서 SSM 연결을 생성, 삭제 및 업데이트할 수 있도록 작업의 리소스 범위가 재구성되었습니다.	2022년 8월 10일
AmazonInspector2 ReadOnlyAccess — 새 정책	Amazon Inspector 기능에 대한 읽기 전용 액세스를 허용하는 새로운 정책이 추가되었습니다.	2022년 1월 21일
AmazonInspector2 FullAccess — 새 정책	Amazon Inspector 기능에 대한 전체 액세스를 허용하는 새로운 정책이 추가되었습니다.	2021년 11월 29일
AmazonInspector2 ServiceRolePolicy — 새 정책	Amazon Inspector에서 사용자를 대신하여 다른 서비스의 작업을 수행할 수 있도록 허용하는 새로운 정책이 추가되었습니다.	2021년 11월 29일

변경 사항	설명	날짜
Amazon Inspector에서 변경 사항 추적 시작	Amazon Inspector는 AWS 관리형 정책의 변경 사항을 추적하기 시작했습니다.	2021년 11월 29일

Amazon Inspector에 서비스 연결 역할 사용

Amazon Inspector는 이름이 지정된 AWS Identity and Access Management (IAM) 서비스 연결 역할을 사용합니다. `AWSServiceRoleForAmazonInspector2` 이 서비스 연결 역할은 Amazon Inspector에 직접 연결되는 IAM 역할입니다. Amazon Inspector에서 미리 정의하며 Amazon Inspector에서 사용자를 대신하여 다른 사람에게 전화를 거는 AWS 서비스 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 정책과 권한 정책이 포함되며 이 권한 정책은 다른 IAM 엔터티에 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 기타 서비스에 대한 자세한 내용은 IAM으로 작업하는AWS 서비스를 참조하고 서비스 연결 역할 열에 예가 있는 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(Amazon EC2) 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 검색할 수 있습니다.

- AWS Systems Manager SendCommand작업을 사용하여 대상 인스턴스에 대한 CIS 스캔을 호출할 수 있습니다.
- Amazon Elastic Container Registry 작업을 사용하여 컨테이너 이미지에 대한 정보를 검색합니다.
- AWS Lambda 작업을 사용하여 Lambda 함수에 대한 정보를 검색하십시오.
- AWS Organizations 작업을 사용하여 관련 계정을 설명하십시오.
- CloudWatch 작업을 사용하여 Lambda 함수가 마지막으로 호출된 시간에 대한 정보를 검색할 수 있습니다.
- 일부 IAM 작업을 사용하여 Lambda 코드에 보안 취약성을 일으킬 수 있는 IAM 정책에 대한 정보를 검색합니다.
- CodeGuru 보안 작업을 사용하여 Lambda 함수의 코드 스캔을 수행할 수 있습니다. Amazon Inspector는 다음과 같은 CodeGuru 보안 조치를 사용합니다.
  - codeguru-security: CreateScan — 보안 스캔을 생성할 권한을 부여합니다. CodeGuru
  - codeguru-security: GetScan — 보안 스캔 메타데이터를 검색할 권한을 부여합니다. CodeGuru
  - codeguru-security: ListFindings — 보안에서 생성된 결과를 검색할 수 있는 권한을 부여합니다. CodeGuru
  - codeguru-security: DeleteScansByCategory — Amazon Inspector에서 시작한 스캔을 삭제할 수 있는 권한을 CodeGuru 보안 팀에 부여합니다.
  - codeguru-security: BatchGetFindings — 보안에서 생성한 특정 결과를 일괄 검색할 수 있는 권한을 부여합니다. CodeGuru
- 일부 Elastic Load Balancing 작업을 사용하여 Elastic Load Balancing 대상 그룹에 속하는 EC2 인스턴스의 네트워크 스캔을 수행합니다.

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",
    "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 CLI, 또는 AWS API에서 Amazon Inspector를 활성화하면 Amazon Inspector가 사용자를 대신하여 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 IAM을 사용하여 역할의 설명을 편집할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 서비스 연결 역할 편집을 참조하십시오.

Amazon Inspector의 서비스 연결 역할 삭제

Amazon Inspector를 더 이상 사용하지 않을 경우에는 `AWSServiceRoleForAmazonInspector2` 서비스 연결 역할을 삭제하는 것이 좋습니다. 역할을 삭제하려면 먼저 활성화된 AWS 리전 각 위치에서 Amazon Inspector를 비활성화해야 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 CLI, 또는 AWS API를 사용하여 서비스 연결 역할을 삭제할 수 있습니다. `AWSServiceRoleForAmazonInspector2` 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. 이 SLR을 사용하면 Amazon Inspector가 사용자 계정에서 Amazon

EBS 볼륨 스냅샷을 생성한 다음 해당 스냅샷의 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 에이전트 없는 스캔을 수행하는 것을 방지합니다. 또한 이 문은 암호화하는 데 사용된 KMS 키에 `InspectorEc2Exclusion` 태그가 있는 경우 Amazon Inspector가 볼륨의 암호화된 데이터에 액세스하는 것을 방지합니다. 자세한 설명은 Amazon Inspector 스캔에서 인스턴스 제외 섹션을 참조하세요.

역할에 대한 권한 정책(`AmazonInspector2AgentlessServiceRolePolicy`)을 통해 Amazon Inspector에서는 다음과 같은 작업을 수행할 수 있습니다.

- Amazon Elastic Compute Cloud(Amazon EC2) 작업을 사용하여 EC2 인스턴스, 볼륨 및 스냅샷에 관한 정보를 검색합니다.
  - Amazon EC2 태그 지정 작업을 사용하여 스캔을 위해 `InspectorScan` 태그 키로 스냅샷에 태그를 지정합니다.
  - Amazon EC2 스냅샷 작업을 사용하여 스냅샷을 생성하고, `InspectorScan` 태그 키로 스냅샷에 태그를 지정한 다음, `InspectorScan` 태그 키로 태그가 지정된 Amazon EBS 볼륨의 스냅샷을 삭제합니다.
- Amazon EBS 작업을 사용하면 `InspectorScan` 태그 키로 태그가 지정된 스냅샷에서 정보를 검색합니다.
- 일부 AWS KMS 암호 해독 작업을 사용하여 고객 관리 키로 암호화된 스냅샷을 해독할 수 있습니다. AWS KMS Amazon Inspector는 스냅샷을 암호화하는 데 사용된 KMS 키에 `InspectorEc2Exclusion` 태그가 지정된 경우 스냅샷을 복호화하지 않습니다.

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 CLI, 또는 AWS API에서 Amazon Inspector를 활성화하면 Amazon Inspector가 사용자를 대신하여 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 IAM을 사용하여 역할의 설명을 편집할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 서비스 연결 역할 편집을 참조하십시오.

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. 자세한 내용은 [스캔 모드 설정 링크 미정] 섹션을 참조하세요.

IAM을 사용하여 수동으로 서비스 링크 역할을 삭제하려면

IAM 콘솔 AWS CLI, 또는 AWS API를 사용하여 서비스 연결 역할을 삭제하십시오. AWSServiceRoleForAmazonInspector2Agentless 자세한 내용은 IAM 사용 설명서에서 서비스 연결 역할 삭제를 참조하세요.

Amazon Inspector 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 Amazon Inspector 및 IAM으로 작업할 때 발생할 수 있는 일반적인 문제를 진단하고 수정할 수 있습니다.

주제

- Amazon Inspector에서 작업을 수행할 권한이 없음
- 저는 IAM을 수행할 권한이 없습니다. PassRole
- 외부 사용자가 내 Amazon Inspector AWS 계정 리소스에 액세스할 수 있도록 허용하고 싶습니다.

Amazon Inspector에서 작업을 수행할 권한이 없음

작업을 수행할 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 없을 때 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 필요하면 관리자에게 문의하세요. AWS 관리자는 로그인 자격 증명을 제공한 사람입니다.

저는 IAM을 수행할 권한이 없습니다. PassRole

`iam:PassRole` 작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 Amazon Inspector에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

새 서비스 역할 또는 서비스 연결 역할을 만드는 대신 기존 역할을 해당 서비스에 전달할 AWS 서비스 수 있는 기능도 있습니다. 이렇게 하려면 사용자가 서비스에 역할을 전달할 수 있는 권한을 가지고 있어야 합니다.

다음 예제 오류는 marymajor라는 IAM 사용자가 콘솔을 사용하여 Amazon Inspector에서 태스크를 수행하려고 하는 경우에 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 필요하면 관리자에게 문의하세요. AWS 관리자는 로그인 자격 증명을 제공한 사람입니다.

외부 사용자가 내 Amazon Inspector AWS 계정 리소스에 액세스할 수 있도록 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(ACL)을 지원하는 서비스의 경우 이러한 정책을 사용하여 다른 사람에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세히 알아보려면 다음을 참조하세요.

- Amazon Inspector에서 이러한 기능을 지원하는지 여부를 알아보려면 Amazon Inspector에서 IAM을 사용하는 방법 섹션을 참조하세요.
- 소유하고 AWS 계정 있는 모든 리소스에 대한 액세스 권한을 AWS 계정 부여하는 방법을 알아보려면 IAM 사용 설명서의 다른 IAM 사용자에게 액세스 권한 제공을 참조하십시오.
- 제3자에게 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사 AWS 계정AWS 계정 소유에 대한 액세스 제공을 참조하십시오.
- 자격 증명 연동을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용자 설명서의 외부에서 인증된 사용자에게 액세스 권한 제공(자격 증명 연동)을 참조하세요.
- 교차 계정 액세스를 위한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용자 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.

Amazon Inspector 모니터링

모니터링은 Amazon Inspector 및 기타 AWS 솔루션의 안정성, 가용성 및 성능을 유지하는 데 있어 중요한 부분입니다. AWS Amazon Inspector를 관찰하고, 문제 발생 시 보고하고, 적절한 경우 자동 조치를 취할 수 있는 모니터링 도구를 제공합니다.

- EventBridgeAmazon은 다양한 소스의 데이터에 애플리케이션을 쉽게 연결할 수 있게 해주는 서버리스 이벤트 버스 서비스입니다. EventBridge 자체 애플리케이션, oftware-as-a S-Service (SaaS) 애플리케이션 AWS 및 서비스에서 실시간 데이터 스트림을 제공하고 해당 데이터를 Lambda와 같은 대상으로 라우팅합니다. 이를 통해 서비스에서 발생하는 이벤트를 모니터링하고 이벤트 기반 아키텍처를 구축할 수 있습니다. 자세한 내용은 Amazon EventBridge 사용 설명서를 참조하십시오.
- AWS CloudTrail는 AWS 계정에서 또는 이 계정을 대신하여 수행된 API 호출 및 관련 이벤트를 캡처합니다. CloudTrail 그런 다음 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 어떤 사용자와 계정이 전화를 걸었는지 AWS, 어떤 소스 IP 주소에서 호출이 이루어졌는지, 언제 호출이 발생했는지 확인할 수 있습니다. 자세한 내용은 AWS CloudTrail 사용 설명서를 참조하십시오.

AWS CloudTrail을 사용하여 Amazon Inspector API 호출 로깅

Amazon Inspector는 Amazon Inspector에서 IAM 사용자 또는 역할 또는 사용자가 수행한 작업에 대한 기록을 제공하는 서비스와 통합되어 있습니다. AWS CloudTrail AWS 서비스 CloudTrail Amazon Inspector에 대한 모든 API 호출을 이벤트로 캡처합니다. 캡처되는 호출에는 Amazon Inspector 콘솔로부터의 호출과 Amazon Inspector API 작업에 대한 호출이 포함됩니다. 트레일을 생성하면 Amazon Inspector에 대한 CloudTrail 이벤트를 포함하여 Amazon S3 버킷으로 이벤트를 지속적으로 전송할 수 있습니다. 추적을 구성하지 않은 경우 이벤트 기록에서 CloudTrail 콘솔의 최신 이벤트를 볼 수도 있습니다. 에서 수집한 정보를 사용하여 CloudTrail 다음을 확인할 수 있습니다.

- Amazon Inspector에 보낸 요청
- 요청이 발생한 IP 주소
- 요청한 사람
- 요청이 발생한 시간

에 대해 자세히 CloudTrail 알아보려면 AWS CloudTrail 사용 설명서를 참조하십시오.

아마존 인스펙터 정보 CloudTrail

CloudTrail 계정을 생성할 AWS 계정 때 활성화됩니다. Amazon Inspector에서 활동이 발생하면 해당 활동이 이벤트 기록의 다른 CloudTrail 이벤트와 함께 AWS 서비스 이벤트에 기록됩니다. 에서 최근 이벤트를 보고, 검색하고, 다운로드할 수 있습니다. AWS 계정자세한 내용은 이벤트 기록으로 CloudTrail 이벤트 보기를 참조하십시오.

Amazon Inspector의 이벤트를 AWS 계정포함하여 귀하의 이벤트에 대한 지속적인 기록을 보려면 트레일을 생성하십시오. 트레일을 사용하면 CloudTrail Amazon S3 버킷으로 로그 파일을 전송할 수 있

습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 지역의 이벤트를 로깅하고 지정된 Amazon S3 버킷으로 로그 파일을 전송합니다. 또한 CloudTrail 로그에서 수집된 이벤트 데이터를 추가로 분석하고 이에 따라 조치를 AWS 서비스 취하도록 기타를 구성할 수 있습니다. 자세한 정보는 다음 주제를 참조하십시오.

- 추적 생성 개요
- CloudTrail 지원되는 서비스 및 통합
- CloudTrail에 대한 Amazon SNS 알림 구성
- 여러 계정에서 CloudTrail 로그 파일 받기
- 여러 지역에서 CloudTrail 로그 파일 받기

모든 Amazon Inspector 작업은 로깅에 의해 기록됩니다. CloudTrail Amazon Inspector가 수행할 수 있는 모든 작업은 Amazon Inspector API 참조에 문서화되어 있습니다. 예를 들어, `CreateFindingsReport`, `ListCoverage` 및 `UpdateOrganizationConfiguration` 작업을 호출하면 CloudTrail 로그 파일에 항목이 생성됩니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 보안 인증 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트 사용자로 했는지 아니면 IAM 사용자 자격 증명으로 했는지 여부.
- 역할 또는 연합된 사용자에 대한 임시 보안 자격 증명을 사용하여 요청했는지 여부
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail userIdentity 요소를 참조하십시오.

Amazon Inspector 로그 파일 항목 이해

트레일은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 전송할 수 있는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함되어 있습니다. 이벤트는 모든 소스로부터 단일 요청을 나타냅니다. 이벤트에는 요청된 작업, 작업 날짜 및 시간, 요청 파라미터 등에 대한 정보가 포함되어 있습니다. CloudTrail 로그 파일은 공개 API 호출의 정렬된 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

Amazon Inspector 스캔 정보 입력 CloudTrail

Amazon Inspector Scan은 과 통합되어 있습니다. CloudTrail 모든 Amazon Inspector 스캔 API 작업은 관리 이벤트로 로깅됩니다. 아마존 인스펙터가 CloudTrail 기록하는 아마존 인스펙터 스캔 API 작업 목록은 아마존 인스펙터 API 레퍼런스의 아마존 인스펙터 스캔을 참조하십시오.

다음 예는 작업을 보여주는 CloudTrail 로그 항목을 보여줍니다. `ScanSbom`


```
{
    "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

특정 규정 준수 프로그램의 범위 내에 AWS 서비스 있는지 알아보려면AWS 서비스 규정 준수 프로그램의AWS 서비스 범위별, 규정 참조하여 관심 있는 규정 준수 프로그램을 선택하십시오. 일반 정보는 AWS 규정 준수 프로그램AWS 보증 프로그램 규정AWS 참조하십시오.

를 사용하여 AWS Artifact타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 의 보고서 https://docs.aws.amazon.com/artifact/latest/ug/downloading-documents.html 참조하십시오 AWS Artifact.

사용 시 규정 준수 AWS 서비스 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 결정됩니다. AWS 규정 준수에 도움이 되는 다음 리소스를 제공합니다.

- 보안 및 규정 준수 퀵 스타트 가이드 - 이 배포 가이드에서는 아키텍처 고려 사항을 설명하고 보안 및 규정 준수에 AWS 중점을 둔 기본 환경을 배포하기 위한 단계를 제공합니다.
- Amazon Web Services의 HIPAA 보안 및 규정 준수를 위한 설계 — 이 백서에서는 기업이 HIPAA 적격 애플리케이션을 만드는 AWS 데 사용할 수 있는 방법을 설명합니다.

> **Note**
>
> 모든 AWS 서비스 사람이 HIPAA 자격을 갖춘 것은 아닙니다. 자세한 내용은 HIPAA 적격 서비스 참조를 참조하십시오.

- AWS 규정 준수 리소스AWS — 이 워크북 및 가이드 모음은 해당 산업 및 지역에 적용될 수 있습니다.
- AWS 고객 규정 준수 가이드 — 규정 준수의 관점에서 공동 책임 모델을 이해하십시오. 이 가이드에서는 보안을 유지하기 위한 모범 사례를 AWS 서비스 요약하고 여러 프레임워크 (미국 표준 기술 연구소 (NIST), 결제 카드 산업 보안 표준 위원회 (PCI), 국제 표준화기구 (ISO) 등) 에서 보안 제어에 대한 지침을 매핑합니다.
- 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— 이를 AWS 서비스 통해 내부 AWS보안 상태를 포괄적으로 파악할 수 있습니다. Security Hub는 보안 통제를 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 Security Hub 제어 참조를 참조하십시오.
- AWS Audit Manager— 이를 AWS 서비스 통해 AWS 사용량을 지속적으로 감사하여 위험을 관리하고 규정 및 업계 표준을 준수하는 방법을 단순화할 수 있습니다.

Amazon Inspector의 복원성

AWS 글로벌 인프라는 가용 영역을 중심으로 AWS 리전 구축됩니다. AWS 리전 물리적으로 분리되고 격리된 여러 가용 영역을 제공합니다. 이 가용 영역은 지연 시간이 짧고 처리량이 높으며 중복성이 높은 네트워킹으로 연결됩니다. 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

Amazon Inspector는 관리형 서비스로서 AWS 글로벌 네트워크 보안의 보호를 받습니다. AWS 보안 서비스 및 인프라 AWS 보호 방법에 대한 자세한 내용은 AWS 클라우드 보안을 참조하십시오. 인프라 보안 모범 사례를 사용하여 AWS 환경을 설계하려면 Security Pillar AWS Well-Architected Framework의 인프라 보호를 참조하십시오.

AWS 게시된 API 호출을 사용하여 네트워크를 통해 Amazon Inspector에 액세스할 수 있습니다. 고객은 다음을 지원해야 합니다.

- 전송 계층 보안(TLS). TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 비밀 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 인증 정보를 생성하여 요청에 서명할 수 있습니다.

Amazon Inspector의 인시던트 대응

AWS에서는 보안을 가장 중요하게 생각합니다. AWS 클라우드 공동 책임 모델의 일환으로 가장 보안에 민감한 조직의 요구 사항을 충족하는 데이터 센터, 네트워크 및 소프트웨어 아키텍처를 AWS 관리합니다. AWS 서비스 자체와 관련된 모든 사고 대응을 담당합니다. AWS Config 또한 AWS 고객으로서 클라우드의 보안을 유지할 책임도 있습니다. 즉, 액세스할 수 있는 AWS 도구 및 기능 중에서 구현하기로 선택한 보안을 제어하고 공동 책임 모델에 따라 사고 대응을 책임지는 사람이 됩니다.

클라우드에서 실행되는 애플리케이션의 목표를 충족하는 보안 기준을 설정하면 대응할 수 있는 편차를 감지할 수 있습니다. 보안 사고 대응은 복잡한 주제일 수 있으므로 IR (사고 대응) 과 선택이 기업 목표에 미치는 영향을 더 잘 이해할 수 있도록 AWS 보안 사고 대응 가이드, 보안 모범 사례 백서,AWS AWS 클라우드 채택 프레임워크 (CAF) 의 보안 관점 백서 등의 리소스를 검토하는 것이 좋습니다.

Amazon Inspector 통합

Amazon Inspector는 다른 AWS 서비스와 통합됩니다. 이러한 서비스는 Amazon Inspector에서 데이터를 수집하여 새로운 방식으로 결과를 확인할 수 있도록 합니다. 해당 서비스가 Amazon Inspector와 함께 작동하도록 설정하는 방법에 대해 자세히 알아보려면 다음 통합 옵션을 검토하세요.

Amazon Inspector와 Amazon ECR 통합

Amazon Elastic Container Registry(Amazon ECR)는 컨테이너 이미지를 간편하게 저장, 공유 및 배포할 수 있는 완전 관리형 Docker 컨테이너 레지스트리입니다. Amazon ECR 프라이빗 레지스트리는 가용성 및 확장성이 뛰어난 아키텍처에서 이미지를 호스팅합니다. Amazon Inspector를 사용하여 Amazon ECR 리포지토리에 있는 컨테이너 이미지에서 취약한 운영 체제 패키지 및 프로그래밍 언어 패키지가 있는지 스캔할 수 있습니다.

Amazon Inspector와 함께 Amazon ECR을 사용하는 방법에 대한 자세한 내용은 Amazon Inspector와 Amazon Elastic Container Registry(Amazon ECR) 통합 섹션을 참조하세요.

Amazon Inspector와 AWS Security Hub 통합

AWS Security Hub는 AWS 계정, 서비스 및 기타 지원되는 제품 전반에서 보안 데이터를 수집하여 업계 표준 및 모범 사례에 따라 환경의 보안 상태를 평가합니다. 보안 태세를 평가하는 것 외에도 Security Hub는 모든 통합 AWS 서비스 및 AWS 파트너 네트워크 제품 전반의 결과를 중앙에서 확인할 수 있는 위치를 생성합니다. Amazon Inspector로 Security Hub를 활성화하면 Security Hub에서 Amazon Inspector 결과 데이터를 자동으로 수집할 수 있습니다.

Amazon Inspector와 함께 Security Hub를 사용하는 방법에 대한 자세한 내용은 Amazon Inspector와 AWS Security Hub 통합 섹션을 참조하세요.

Amazon Inspector와 Amazon Elastic Container Registry(Amazon ECR) 통합

Amazon ECR은 AWS에서 Docker 및 OCI 이미지와 아티팩트를 지원하는 완전 관리형 컨테이너 레지스트리입니다. Amazon ECR을 사용하는 경우, 레지스트리에 대한 고급 검색을 활성화하여 Amazon Inspector에서 컨테이너 이미지를 자동으로 탐지하고 취약한 운영 체제 패키지 및 프로그래밍 언어 패키지가 있는지 스캔할 수 있습니다.

이 통합을 통해 Amazon ECR 콘솔 내에서 컨테이너 이미지에 대한 Amazon Inspector 결과를 볼 수 있습니다. 또한 Amazon ECR 콘솔에서 스캔 빈도를 관리하고 포함 필터를 생성하여 스캔 범위를 조정할 수 있습니다.

통합 활성화

Amazon Inspector 콘솔 또는 API를 통해 Amazon Inspector 스캔을 활성화하거나, Amazon ECR 콘솔 또는 API를 통해 Amazon Inspector의 고급 스캔을 사용하도록 리포지토리를 구성하여 통합을 활성화할 수 있습니다.

Amazon Inspector를 통한 통합 활성화에 대한 자세한 내용은 Amazon Inspector를 사용한 자동 리소스 스캔 섹션을 참조하세요.

Amazon ECR에서 고급 스캔을 활성화하고 구성하는 방법에 대한 자세한 내용은 Amazon ECR 사용 설명서에서 고급 스캔을 참조하세요.

다중 계정 환경과의 통합 사용

다중 계정 환경의 멤버인 경우 Amazon ECR을 통해 고급 검색을 활성화할 수 있습니다. 하지만 활성화한 후에는 Amazon Inspector 위임 관리자만 비활성화할 수 있습니다. 비활성화되면 기본 스캔으로 돌아갑니다. 자세한 내용은 Amazon Inspector 비활성화 섹션을 참조하세요.

Amazon Inspector와 AWS Security Hub 통합

Security Hub에서는 AWS에서 보안 상태를 포괄적으로 파악할 수 있으며 보안 업계 표준 및 모범 사례와 비교하여 환경을 확인할 수 있습니다. Security Hub는 AWS 계정, 서비스 및 지원되는 추가 제품 전반에서 보안 데이터를 수집합니다. 제공하는 정보를 사용하여 보안 추세를 분석하고 우선순위가 가장 높은 보안 문제를 식별할 수 있습니다.

Amazon Inspector와 Security Hub 통합을 사용하면 Amazon Inspector의 결과를 Security Hub로 보낼 수 있습니다. 그러면 Security Hub의 보안 태세 분석에 이러한 결과가 포함됩니다.

AWS Security Hub의 경우, 보안 문제를 결과로 추적합니다. 일부 결과는 다른 AWS 서비스 또는 타사 제품에서 탐지한 문제로 인해 발생합니다. Security Hub에는 보안 문제를 감지하고 결과를 생성하는 데 사용하는 규칙 집합도 있습니다. Security Hub는 이러한 모든 출처를 총망라하여 결과를 관리할 도구를 제공합니다. 사용자는 결과 목록을 조회하고 필터링할 수 있으며 결과 세부 정보를 조회할 수도 있습니다. Security Hub의 결과에 대한 자세한 내용은 AWS Security Hub 사용 설명서에서 결과 보기를 참조하세요. 또한 주어진 결과에 대한 조사 상태를 추적할 수도 있습니다. AWS Security Hub 사용 설명서에서 결과에 대한 작업 수행을 참조하세요.

Security Hub의 모든 결과는 표준 JSON 형식을 사용합니다. 이를 AWS Security Finding Format(ASFF)이라고 합니다. ASFF에는 문제의 출처, 영향을 받은 리소스와 결과의 현재 상태 등에 관한 세부 정보가 포함됩니다. AWS Security Hub 사용 설명서에서 AWS Security Finding 형식(ASFF)을 참조하세요.

Amazon Inspector 결과가 Amazon Inspector에서 해결되고 종결되면 Security Hub에서 해당 결과를 보관합니다.

AWS Security Hub에서 Amazon Inspector 결과 보기

Amazon Inspector Classic과 새로운 Amazon Inspector에서 제공하는 결과는 Security Hub의 동일한 패널에서 확인할 수 있습니다. 하지만 필터 막대에 `"aws/inspector/ProductVersion": "2"`를 추가하면 새 Amazon Inspector의 결과를 필터링할 수 있습니다. 이 필터를 추가하면 Amazon Inspector Classic의 결과가 Security Hub 대시보드에서 제외됩니다.

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"
}
```


통합 활성화 및 구성

Amazon Inspector와 AWS Security Hub의 통합을 사용하려면 Security Hub를 활성화해야 합니다. Security Hub를 활성화하는 방법에 대한 자세한 내용은 AWS Security Hub 사용 설명서에서 Security Hub 설정을 참조하세요.

Amazon Inspector와 Security Hub를 모두 활성화하면 통합이 자동으로 활성화되고 Amazon Inspector에서 결과를 Security Hub로 보내기 시작합니다. Amazon Inspector는 ASFF(AWS Security Finding Format)를 사용하여 생성되는 모든 결과를 Security Hub로 보냅니다.

AWS Security Hub로의 결과 게시 중지

결과 전송을 중지하는 방법

Security Hub로 결과를 전송하는 작업을 중지하려면 Security Hub 콘솔 또는 API를 사용하면 됩니다.

AWS Security Hub 사용 설명서에서 통합에서 결과의 흐름 사용 중지 및 사용 설정(콘솔) 또는 통합에서 결과의 흐름 해제(Security Hub API, AWS CLI)를 참조하세요.

Amazon Inspector에서 지원하는 운영 체제 및 프로그래밍 언어

Amazon Inspector는 아마존 Elastic Compute Cloud (Amazon EC2) 인스턴스에 설치된 소프트웨어 애플리케이션, 아마존 ECR (엘라스틱 컨테이너 레지스트리) 리포지토리에 저장된 컨테이너 이미지 및 함수를 스캔할 수 있습니다. AWS Lambda ECR 컨테이너 이미지의 경우 Amazon Inspector는 운영 체제 및 프로그래밍 언어 패키지 취약성을 모두 스캔할 수 있습니다. Lambda 함수의 경우 Amazon Inspector는 코드 취약성을 스캔할 수 있습니다. Amazon Inspector에서 리소스를 스캔할 때 특별히 제작된 자체 스캔 엔진을 사용하며 50개 이상의 데이터 피드를 소싱하여 일반적인 취약성 및 노출(CVE)에 대한 결과를 생성합니다. 소스에는 공급업체 보안 권고, NVD, MITRE, 오픈 소스 피드, 내부 연구 및 라이선스가 부여된 데이터 피드가 포함됩니다.

Amazon Inspector에서 리소스를 스캔하려면 리소스가 지원되는 운영 체제를 실행 중이거나 지원되는 프로그래밍 언어를 사용해야 합니다. 이 섹션의 주제에는 Amazon Inspector가 현재 다양한 리소스 및 스캔 유형에 지원하는 운영 체제, 런타임 및 프로그래밍 언어가 나열되어 있습니다. 또한 Amazon Inspector가 이전에 지원했지만 공급업체에서 중단한 운영 체제도 나열되어 있습니다. 공급업체가 지원을 중단한 운영 체제에 대해서는 Amazon Inspector에서 제한된 지원만 제공할 수 있습니다.

주제

지원되는 운영 체제: Amazon EC2 스캔

다음 표에는 Amazon Inspector에서 현재 Amazon EC2 인스턴스 스캔을 지원하는 운영 체제가 나와 있습니다. 또한 각 공급업체 보안 권고 사항의 출처와 에이전트 기반 또는 에이전트 없는 스캔 방법을 사용하여 해당 운영 체제를 스캔할 수 있는지 여부도 나열합니다. 스캔 방법에 대한 자세한 내용은 에이전트 기반 스캔 및 에이전트 없는 스캔 섹션을 참조하세요.

> **Note**
>
> Linux 운영 체제 탐지는 기본 패키지 관리자 리포지토리에서만 지원되며 타사 애플리케이션, 확장 지원 리포지토리 (예: BYOS RHEL, PAYG RHEL, SAP용 RHEL) 및 Red Hat 애플리케이션 스트림과 같은 선택적 리포지토리는 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
Amazon Linux 2023(AL2023)	AL2023	ALAS	예	예
Bottlerocket	1.7.0 이상	GHSA, CVE	아니요	예
CentOS Linux(CentOS)	7	CESA	예	예
Debian Server(Buster)	10	DSA	예	예
Debian Server(Bullseye)	11	DSA	예	예
Debian Server(Bookworm)	12	DSA	예	예
Fedora	38	CVE	예	예
Fedora	39	CVE	예	예

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
OpenSUSE	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)	7	RHSA	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
Rocky Linux	9	RLSA	예	예
SUSE Linux Enterprise Server(SLES)	12.4	SUSE CVE	예	예
SUSE Linux Enterprise Server(SLES)	12.5	SUSE CVE	예	예

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
SUSE Linux Enterprise Server(SLES)	15.3	SUSE CVE	예	예
SUSE Linux Enterprise Server(SLES)	15.4	SUSE CVE	예	예
SUSE Linux Enterprise Server(SLES)	15.5	SUSE CVE	예	예
Ubuntu(Trusty)	14.04(ESM)	USN, Ubuntu Pro	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(Mantic Minotaur)	23.10	USN	예	예
Windows Server	2016	MSKB	아니요	예
Windows Server	2019	MSKB	아니요	예
Windows Server	2022	MSKB	아니요	예
macOS (모하비)	10.14	APPLE-SA	아니요	예

운영 체제	버전	공급업체 보안 권고	에이전트 없는 스캔 지원	에이전트 기반 스캔 지원
macOS (카탈리나)	10.15	APPLE-SA	아니요	예
macOS (빅서)	11	APPLE-SA	아니요	예
macOS (몬터레이)	12	APPLE-SA	아니요	예
macOS (벤츄라)	13	APPLE-SA	아니요	예

지원되는 프로그래밍 언어: Amazon EC2 심층 검사

Amazon Inspector는 현재 Amazon EC2 Linux 인스턴스를 스캔하여 타사 소프트웨어 패키지의 취약성을 검사할 때 다음과 같은 프로그래밍 언어를 지원합니다.

- Java
- JavaScript
- Python

Amazon Inspector는 Systems Manager 디스트리뷰터를 사용하여 Amazon EC2 인스턴스의 심층 검사에 사용되는 플러그인을 배포합니다. Systems Manager Distributor는 Systems Manager 설명서에서 지원되는 패키지 플랫폼 및 아키텍처로 나열된 운영 체제를 지원합니다. Amazon Inspector에서 심층 검사 스캔을 수행하려면 Systems Manager Distributor 및 Amazon Inspector에서 Amazon EC2 인스턴스의 운영 체제를 지원해야 합니다.

> **Note**
>
> Bottlerocket 운영 체제에서는 심층 검사가 지원되지 않습니다.

지원되는 운영 체제: CIS 스캐닝

다음 표에는 Amazon Inspector에서 현재 CIS 스캔을 지원하는 운영 체제가 나와 있습니다. 이 표에는 해당 운영 체제의 스캔을 수행하는 데 사용되는 CIS 벤치마크 버전도 포함되어 있습니다.

운영 체제	버전	CIS 벤치마크 버전
Amazon Linux 2	AL2	2.0.0
Amazon Linux 2023	AL2023	1.0.0
Windows Server	2019	2.0.0
Windows Server	2022	2.0.0

지원되는 운영 체제: 아마존 인스펙터를 사용한 아마존 ECR 스캔

Amazon Inspector는 현재 Amazon ECR 리포지토리에서 컨테이너 이미지를 스캔할 때 다음 운영 체제 스캔을 지원합니다. 이 표에는 각 운영 체제에 대한 공급업체 보안 권고 사항의 출처도 나와 있습니다.

운영 체제	버전	공급업체 보안 권고
Alpine Linux (Alpine)	3.16	Alpine SecDB
Alpine Linux (Alpine)	3.17	Alpine SecDB
Alpine Linux (Alpine)	3.18	Alpine SecDB
Alpine Linux (Alpine)	3.19	Alpine SecDB
AlmaLinux	8	ALSA
AlmaLinux	9	ALSA
Amazon Linux (AL2)	AL2	ALAS
Amazon Linux 2023 (AL2023)	AL2023	ALAS
CentOS Linux (CentOS)	7	CESA
Debian Server (Buster)	10	DSA
Debian Server (Bullseye)	11	DSA

운영 체제	버전	공급업체 보안 권고
Debian Server (Bookworm)	12	DSA
Fedora	38	CVE
Fedora	39	CVE
OpenSUSE	15.5	CVE
Oracle Linux (Oracle)	7	ELSA
Oracle Linux (Oracle)	8	ELSA
Oracle Linux (Oracle)	9	ELSA
Photon OS	3	PHSA
Photon OS	4	PHSA
Photon OS	5	PHSA
Red Hat Enterprise Linux (RHEL)	7	RHSA
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)	12.4	SUSE CVE
SUSE Linux Enterprise Server (SLES)	12.5	SUSE CVE

운영 체제	버전	공급업체 보안 권고
SUSE Linux Enterprise Server (SLES)	15.3	SUSE CVE
SUSE Linux Enterprise Server (SLES)	15.4	SUSE CVE
SUSE Linux Enterprise Server (SLES)	15.5	SUSE CVE
Ubuntu (Trusty)	14.04 (ESM)	USN, Ubuntu Pro
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 (Mantic Minotaur)	23.10	USN

지원되는 프로그래밍 언어: Amazon ECR 스캔

Amazon Inspector는 현재 Amazon ECR 리포지토리에서 컨테이너 이미지를 스캔할 때 다음과 같은 프로그래밍 언어를 지원합니다.

- C#
- Go
- Java
- JavaScript
- PHP
- Python
- Ruby
- Rust

지원되는 런타임: Amazon Inspector Lambda 표준 스캔

Amazon Inspector Lambda 표준 스캐닝은 현재 Lambda 함수에서 타사 소프트웨어 패키지의 취약점을 스캔할 때 다음과 같은 프로그래밍 언어를 지원합니다.

- Java
  - java8
  - java8.al2
  - java11
  - java17
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
- Go
  - go1.x
- Ruby
  - ruby2.7
  - ruby3.2
- .NET
  - .NET 6

지원되는 런타임: Amazon Inspector Lambda 코드 스캔

Amazon Inspector Lambda 코드 스캔은 현재 Lambda 함수에서 코드의 취약성을 검사할 때 다음과 같은 프로그래밍 언어를 지원합니다.

- Java
  - java8
  - java8.al2
  - java11
  - java17
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
- Ruby
  - ruby2.7
  - ruby3.2

중단된 운영 체제

다음 표에 나열된 운영 체제에 대한 표준 공급업체 지원은 공급업체에 의해 중단되었습니다. 표에서 중단 열은 공급업체가 운영 체제에 대한 표준 지원을 중단한 시기를 나타냅니다.

Amazon Inspector는 이전에 이러한 운영 체제를 완벽하게 지원했으며 계속해서 Amazon EC2 인스턴스와 이를 실행하는 Amazon ECR 컨테이너 이미지를 스캔할 예정입니다. 그러나 공급업체 정책에 따

라 운영 체제는 더 이상 패치로 업데이트되지 않으므로 대부분의 경우 해당 운영 체제에 대한 새로운 보안 권고도 더 이상 발표되지 않습니다. 또한 일부 공급업체의 경우 영향을 받는 운영 체제의 표준 지원이 종료되면 피드에서 기존 보안 권고 및 탐지를 제거합니다. 따라서 Amazon Inspector에서는 알려진 CVE에 대한 결과 생성을 중단할 수 있습니다. 중단된 운영 체제에 대해 Amazon Inspector가 생성하는 모든 결과는 정보 제공 목적으로만 사용해야 합니다.

보안 모범 사례 및 지속적인 Amazon Inspector 적용을 위해, 지원되는 최신 버전의 운영 체제로 전환하는 것이 좋습니다.

중단된 운영 체제: Amazon EC2 스캔

운영 체제	버전	중단됨
Amazon Linux(AL1)	2012	2021년 12월 31일
CentOS Linux(CentOS)	8	2021년 12월 31일
Debian Server(Stretch)	9	2022년 6월 30일
Fedora	35	2022년 12월 13일
Fedora	36	2023년 5월 16일
Fedora	37	2023년 12월 5일
OpenSUSE	15.3	2022년 12월 1일
OpenSUSE	15.4	2023년 12월 7일
OpenSUSE Leap(SUSE Leap)	15.2	2021년 12월 1일
Oracle Linux(Oracle)	6	2021년 3월 1일
SUSE Linux Enterprise Server(SLES)	12	2019년 7월 1일
SUSE Linux Enterprise Server(SLES)	12.1	2020년 5월 31일
SUSE Linux Enterprise Server(SLES)	12.2	2021년 3월 31일

운영 체제	버전	중단됨
SUSE Linux Enterprise Server(SLES)	12.3	2022년 6월 30일
SUSE Linux Enterprise Server(SLES)	15	2019년 12월 31일
SUSE Linux Enterprise Server(SLES)	15.1	2021년 1월 31일
SUSE Linux Enterprise Server(SLES)	15.2	2021년 12월 31일
Ubuntu(Groovy)	20.10	2021년 7월 22일
Ubuntu(Hirsute)	21.04	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
Windows Server	2012	2023년 10월 10일
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

운영 체제	버전	중단됨
Amazon Linux(AL1)	2012	2021년 12월 31일
CentOS Linux(CentOS)	8	2021년 12월 31일
Debian Server(Stretch)	9	2022년 6월 30일
Fedora	35	2022년 12월 13일
Fedora	36	2023년 5월 16일
OpenSUSE	15.3	2022년 12월 1일
OpenSUSE	15.4	December 7, 2023
OpenSUSE Leap(SUSE Leap)	15.2	2021년 12월 1일
Oracle Linux(Oracle)	6	2021년 3월 1일
SUSE Linux Enterprise Server(SLES)	12	2019년 7월 1일
SUSE Linux Enterprise Server(SLES)	12.1	2020년 5월 31일
SUSE Linux Enterprise Server(SLES)	12.2	2021년 3월 31일
SUSE Linux Enterprise Server(SLES)	12.3	2022년 6월 30일
SUSE Linux Enterprise Server(SLES)	15	2019년 12월 31일
SUSE Linux Enterprise Server(SLES)	15.1	2021년 1월 31일
SUSE Linux Enterprise Server(SLES)	15.2	2021년 12월 31일

운영 체제	버전	중단됨
Ubuntu(Groovy)	20.10	2021년 7월 22일
Ubuntu(Hirsute)	21.04	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

Amazon Inspector 비활성화

Amazon Inspector 콘솔 또는 API를 사용하여 모든 AWS 리전에서 Amazon Inspector를 비활성화할 수 있습니다. Amazon Inspector를 비활성화하려면 이 주제의 끝 부분에 설명된 지침을 따르세요. AWS 계정에 대한 모든 Amazon Inspector 스캔을 비활성화하면 이 계정에 대해 Amazon Inspector가 자동으로 비활성화됩니다. 다양한 리소스의 스캔 유형 비활성화에 대한 자세한 내용은 Amazon Inspector를 사용한 자동 리소스 스캔 섹션을 참조하세요.

계정에 대해 Amazon Inspector가 비활성화되면 해당 리전의 해당 계정에 대한 모든 스캔 유형이 비활성화됩니다. 또한 해당 리전의 계정에 대한 모든 Amazon Inspector 스캔 설정, 억제 규칙, 필터 및 결과도 삭제됩니다.

해당 리전의 계정에 대해 Amazon Inspector가 비활성화되어 있는 동안에는 사용 요금이 부과되지 않습니다. Amazon Inspector를 비활성화한 후 나중에 다시 활성화할 수 있습니다.

> **Note**
>
> Amazon Inspector를 비활성화하기 전에 결과를 내보내는 것이 좋습니다. 자세한 설명은 Amazon Inspector에서 결과 보고서 내보내기 섹션을 참조하세요.

Amazon Inspector Amazon EC2 스캔을 비활성화하면 Amazon Inspector에서 사용하는 다음과 같은 SSM 연결이 삭제됩니다.

- `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 설명은 Windows 인스턴스 스캔 섹션을 참조하세요.

사전 조건

계정 유형에 따라 Amazon Inspector를 비활성화하기 전에 다음과 같이 추가 단계를 수행해야 할 수 있습니다.

- 독립 실행형 Amazon Inspector 계정의 경우 언제든지 계정을 비활성화할 수 있습니다.
- Amazon Inspector 다중 계정 환경의 멤버 계정인 경우 자체 서비스를 비활성화할 수 없습니다. 이 서비스를 비활성화하려면 조직의 위임 관리자에게 문의해야 합니다.

- 위임 관리자인 경우 Amazon Inspector를 비활성화하기 전에 모든 멤버 계정을 연결 해제해야 합니다. 자세한 설명은 Amazon Inspector에서 멤버 계정 연결 해제 섹션을 참조하세요.

> **Note**
>
> 계정을 연결 해제해도 해당 계정의 Amazon Inspector는 비활성화되지 않으며, 대신 연결이 해제된 멤버 계정이 독립 실행형 계정이 됩니다.

> **Note**
>
> 위임 관리자로서 Amazon Inspector를 비활성화하면 소속 조직의 자동 활성화 기능이 비활성화됩니다.

Amazon Inspector 비활성화

Console

Amazon Inspector를 비활성화하려면

1. Amazon Inspector 콘솔(https://console.aws.amazon.com/inspector/v2/home)을 엽니다.
2. 페이지 오른쪽 상단의 AWS 리전 선택기를 사용하여 Amazon Inspector를 비활성화할 리전을 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

Disable API 작업을 실행합니다. 요청에 비활성화할 계정 ID를 제공하고 `EC2, ECR, LAMBDA`를 `resourceTypes`로 제공하여 모든 스캔을 비활성화하면 해당 계정이 비활성화됩니다.

Amazon Inspector 할당량

AWS 계정에는 리전별로 다음과 같은 Amazon Inspector 할당량이 있습니다.

Resource	기본값	설명
억제 규칙	500	리전별로 AWS 계정당 저장된 최대 억제 규칙 수입니다. 할당량 증가를 요청할 수 없습니다.
Amazon EC2 네트워크 결과	10,000개	AWS 계정당 최대 Amazon EC2 네트워크 결과 수입니다. 할당량 증가를 요청할 수 없습니다.
멤버 계정	10000	Amazon Inspector 위임 관리자 계정과 연결된 최대 멤버 계정 수입니다. 이 한도는 AWS Organizations를 기준으로 합니다. AWS Organizations에 대한 할당량을 참조하세요.
CIS 스캔 구성	500	CIS 스캔 구성의 최대 수입니다. 할당량 증가를 요청할 수 없습니다.

Amazon Inspector Classic과 관련된 할당량 목록은 AWS 일반 참조에서 Amazon Inspector 서비스 할당량을 참조하세요.

Organizations와 관련된 할당량 목록은 AWS 일반 참조에서 Organizations 서비스 할당량을 참조하세요.

지역 및 엔드포인트

Amazon EC2용 Amazon Inspector 에이전트 없는 스캔은 미리 보기로 출시되었습니다. 에이전트 없는 Amazon EC2 스캔 기능 사용에는 AWS 서비스 약관 섹션 2('베타 및 미리 보기')가 적용됩니다.

Amazon Inspector를 사용할 수 있는 AWS 리전을 확인하려면 Amazon Web Services 일반 참조에서 Amazon Inspector 엔드포인트를 참조하세요.

Amazon Inspector 스캔 API용 엔드포인트

다음 표에는 Amazon Inspector 스캔 API를 호출할 때 사용할 수 있는 리전 엔드포인트가 나와 있습니다. 이 API를 사용할 때는 엔드포인트와 함께 현재 인증된 AWS 리전에 해당하는 엔드포인트의 리전을 제공해야 합니다.

Amazon Inspector 스캔 엔드포인트의 명명 규칙은 `inspector-scan.region.amazonaws.com`입니다. 예를 들어, `us-west-2`에서 인증된 경우 엔드포인트 `inspector-scan.us-west-2.amazonaws.com`을 사용하여 `inspector-scan` API를 호출합니다.

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
Africa (Cape Town)	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
Asia Pacific (Osaka)	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
Canada (Central)	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
Europe (Milan)	eu-south-1	inspector-scan.eu-south-1.amazonaws.com	HTTPS

리전 이름	지역	엔드포인트	프로토콜
Europe (Paris)	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	inspector-scan.us-gov-east-1.amazonaws.com inspector-scan-fips.us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud (US-West)	us-gov-west-1	inspector-scan.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 EC2 리전에 대한 에이전트 없는 EC2 스캔

다음 표는 Amazon EC2에 대한 에이전트 없는 스캔을 현재 사용할 수 있는 AWS 리전을 보여줍니다.

리전 이름	리전 코드
미국 동부(버지니아 북부)	us-east-1
미국 서부(오레곤)	us-west-2
유럽(아일랜드)	eu-west-1

Lambda 코드 스캔 리전

다음 표는 Lambda 코드 스캔을 현재 사용할 수 있는 AWS 리전을 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
유럽(스톡홀름)	eu-north-1

리전 이름	리전 코드
아시아 태평양(싱가포르)	ap-southeast-1

AWS GovCloud (US) 리전

자세한 내용은 AWS GovCloud (US) 사용 설명서에서 Amazon Inspector 섹션을 참조하세요.

Amazon Inspector 사용 설명서의 문서 기록

다음 표에서는 Amazon Inspector의 최신 릴리스 이후 이 설명서에서 변경된 중요한 사항에 대해 설명합니다. 이 설명서에 대한 업데이트 알림을 받으려면 RSS 피드를 구독하면 됩니다.

변경 사항	설명	날짜
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 인스턴스에 대한 CIS 스캔을 시작할 수 있습니다.	2024년 1월 23일
새 정책	Amazon Inspector는 인스턴스에 대한 CIS 스캔을 허용하기 위해 인스턴스 프로필의 일부로 사용할 수 있는 새 AmazonInspector2ManagedCisPolicy정책인 정책을 추가했습니다.	2024년 1월 23일
새 기능	Amazon Inspector는 이제 컨테이너 이미지를 가져올 때 해당 이미지의 ECR 재스캔 기간을 새로 고칩니다. 푸시 또는 풀 날짜를 기준으로 재스캔 기간을 변경하려면 ECR 재스캔 기간 구성을 참조하십시오.	2024년 1월 23일

새 기능	Amazon Inspector는 이제 EC2 인스턴스에서 인터넷 보안 센터 (CIS) 스캔을 실행할 수 있습니다. 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 EC2 인스턴스에서 Amazon EBS 스냅샷을 스캔할 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 Inspector에서 이제 MacOS Amazon EC2 인스턴스 스캔을 지원합니다. 지원되는 macOS 버전은 지원되는 운영 체제: Amazon EC2 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 Inspector에서 Elastic Load Balancing 대상 그룹에 속하는 Amazon EC2 인스턴스의 네트워크 구성을 스캔할 수 있는 새로운 권한이 추가되었습니다.	2023년 8월 31일
새 기능	Amazon Inspector에서 이제 패키지 취약성 발견에 대한 취약성 인텔리전스 세부 정보를 제공합니다.	2023년 7월 31일
업데이트된 기능	Amazon Inspector에서 읽기 전용 사용자가 리소스에 대한 Software Bill of Materials (SBOM)를 내보낼 수 있는 새로운 권한을 추가했습니다.	2023년 6월 29일
새 기능	이제 Amazon Inspector에서 스캔 중인 리소스에 대해 SBOM을 내보낼 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 Inspector에서 특정 CVE를 처리하는지 확인할 수 있는 취약성 데이터베이스 검색 기능이 추가되었습니다.	2023년 5월 1일
업데이트된 기능	Lambda 스캔을 활성화할 때 Amazon Inspector에서 사용자 계정에 AWS CloudTrail 서비스 연결 채널을 생성할 수 있는 새로운 권한이 AmazonInspector2ServiceRolePolicy 정책에 추가되었습니다. 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 Amazon EC2 심층 검사를 위해 정의한 사용자 지정 경로에 대한 정보를 Amazon EC2 Systems Manager에 보낼 수 있습니다.	2023년 4월 17일
새 기능	Amazon Inspector는 애플리케이션 프로그래밍 언어 패키지의 패키지 취약성이 있는지 인스턴스를 검사하는 Amazon Inspector 심층 검사 형태로 Linux EC2 인스턴스에 대한 지원을 추가합니다.	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 Inspector에서 IAM 정책을 검토할 수 있는 권한이 추가되었습니다. 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 Inspector에서 AWS Lambda 함수 스캔에 대한 지원이 추가되었습니다.	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 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 작업은 Amazon Inspector에서 SSM 연결 실행을 설명할 수 있도록 허용합니다. 또한 Amazon Inspector에서 `AmazonInspector2` 소유 SSM 문서와 SSM 연결을 생성, 업데이트, 삭제 및 시작할 수 있도록 리소스 범위가 추가되었습니다.	2022년 8월 31일

새 기능	Amazon Inspector에서 이제 Windows 인스턴스에 대한 스캔을 지원합니다. 지원되는 Windows 운영 체제를 실행하는 SSM 관리형 인스턴스를 Amazon Inspector에서 스캔할 수 있습니다. Windows호스트 스캔은 Amazon Inspector SSM 플러그인에 의해 수행되며, 이 플러그인은 Amazon Inspector에서 자동으로 생성한 새로운 SSM 연결을 통해 설치 및 호출됩니다.	2022년 8월 31일
업데이트된 기능	Amazon Inspector에서 다른 AWS 파티션의 소프트웨어 인벤토리를 수집할 수 있도록 AmazonInspector2ServiceRolePolicy 정책의 리소스 범위가 업데이트되었습니다.	2022년 8월 12일
업데이트된 기능	AmazonInspector2ServiceRolePolicy 정책에서 Amazon Inspector에서 SSM 연결을 생성, 삭제 및 업데이트할 수 있도록 작업의 리소스 범위가 재구성되었습니다.	2022년 8월 10일

새 기능	Amazon Inspector에서 이제 ECR 자동 재스캔 기간 설정의 변경을 지원합니다. Amazon ECR 자동 재스캔 기간 설정에 따라 Amazon Inspector에서 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 ECR 자동 재스캔 기간은 자동으로 전체 기간으로 설정됩니다. 이전에 생성한 계정의 ECR 자동 재스캔 기간은 30일이었지만 이제는 30일, 180일 또는 전체 스캔 기간 중에서 선택할 수 있습니다.	2022년 6월 25일
새로운 기능	Amazon Inspector 기능에 대한 읽기 전용 액세스를 허용하는 새로운 AWS 관리형 정책인 AmazonInspector2ReadOnlyAccess 정책이 추가되었습니다.	2022년 1월 21일
정식 출시	이는 Amazon Inspector 사용 설명서의 최초 공개 릴리스입니다.	2021년 11월 29일

AWS 용어집

최신 AWS 용어는 AWS 용어집 참조서의 AWS 용어집을 참조하세요.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.