aws

Benutzerhandbuch

Amazon Inspector

Amazon Inspector: Benutzerhandbuch

Table of Contents

Was ist Amazon Inspector?

Amazon Inspector ist ein Schwachstellen-Management-Service, der Workloads automatisch erkennt und sie kontinuierlich auf Software-Schwachstellen und unbeabsichtigte Netzwerkbedrohungen überprüft. Amazon Inspector erkennt und scannt EC2 Amazon-Instances, Container-Images in Amazon ECR und Lambda-Funktionen. Wenn Amazon Inspector eine Sicherheitslücke in Software oder eine unbeabsichtigte Netzwerkgefährdung erkennt, erstellt Amazon Inspector einen Befund, bei dem es sich um einen detaillierten Bericht über das Problem handelt. Sie können Ergebnisse in der Amazon Inspector Inspector-Konsole oder API verwalten.

Themen

Funktionen von Amazon Inspector

Zentrales Verwalten mehrerer Amazon Inspector Inspector-Konten

Wenn Ihre AWS Umgebung über mehrere Konten verfügt, können Sie Ihre Umgebung mithilfe von AWS Organizations über ein einziges Konto zentral verwalten. Mit diesem Ansatz können Sie ein Konto als delegiertes Administratorkonto für Amazon Inspector festlegen.

Amazon Inspector kann mit einem einzigen Klick für Ihr gesamtes Unternehmen aktiviert werden. Darüber hinaus können Sie die Aktivierung des Dienstes für future Mitglieder automatisieren, wann immer diese Ihrer Organisation beitreten. Das delegierte Administratorkonto von Amazon Inspector kann Ergebnisdaten und bestimmte Einstellungen für Mitglieder der Organisation verwalten. Dazu gehören die Anzeige aggregierter Ergebnisdetails für alle Mitgliedskonten, die Aktivierung oder Deaktivierung von Scans für Mitgliedskonten und die Überprüfung gescannter Ressourcen innerhalb der Organisation. AWS

Scannen Sie Ihre Umgebung kontinuierlich auf Sicherheitslücken und Netzwerkgefährdungen

Mit Amazon Inspector müssen Sie Bewertungsscans nicht manuell planen oder konfigurieren. Amazon Inspector erkennt automatisch Ihre in Frage kommenden Ressourcen und beginnt mit dem Scannen. Amazon Inspector bewertet Ihre Umgebung weiterhin während des gesamten Lebenszyklus Ihrer Ressourcen, indem es automatisch Ressourcen als Reaktion auf Änderungen, die zu einer neuen Sicherheitslücke führen könnten, erneut scannt, z. B.: Installation eines neuen

Pakets in einer EC2 Instance, Installation eines Patches und wenn neue Common Vulnerabilities and Exposures (CVE), die sich auf die Ressource auswirken, veröffentlicht wird. Im Gegensatz zu herkömmlicher Sicherheitsscan-Software hat Amazon Inspector nur minimale Auswirkungen auf die Leistung Ihrer Flotte.

Wenn Sicherheitslücken oder offene Netzwerkpfade identifiziert werden, erstellt Amazon Inspector ein Ergebnis, das Sie untersuchen können. Das Ergebnis umfasst umfassende Informationen über die Sicherheitsanfälligkeit, die betroffene Ressource und Empfehlungen zur Behebung. Wenn Sie ein Ergebnis angemessen korrigieren, erkennt Amazon Inspector die Behebung automatisch und schließt das Ergebnis.

Beurteilen Sie Sicherheitslücken genau mit dem Amazon Inspector Risk Score

Amazon Inspector sammelt mithilfe von Scans Informationen über Ihre Umgebung und bietet Schweregrade, die speziell auf Ihre Umgebung zugeschnitten sind. Amazon Inspector untersucht die Sicherheitsmetriken, die den Basiswert der National Vulnerability Database (NVD) für eine Sicherheitslücke bilden, und passt sie an Ihre Computerumgebung an. Beispielsweise kann der Service den Amazon Inspector-Score eines Ergebnisses für eine EC2 Amazon-Instance senken, wenn die Sicherheitsanfälligkeit über das Netzwerk ausgenutzt werden kann, aber von der Instance aus kein offener Netzwerkpfad zum Internet verfügbar ist. Diese Bewertung ist im CVSS-Format und ist eine Modifikation der von NVD bereitgestellten Basisbewertung des Common Vulnerability Scoring System (CVSS).

Identifizieren Sie wichtige Ergebnisse mit dem Amazon Inspector-Dashboard

Das Amazon Inspector-Dashboard bietet einen umfassenden Überblick über die Ergebnisse aus Ihrer gesamten Umgebung. Über das Dashboard können Sie auf die detaillierten Details eines Ergebnisses zugreifen. Das Dashboard enthält übersichtliche Informationen zur Scanabdeckung in Ihrer Umgebung, zu Ihren wichtigsten Ergebnissen und zu den Ressourcen, bei denen die meisten Ergebnisse vorliegen. Das Fenster zur risikobasierten Behebung im Amazon Inspector-Dashboard zeigt die Ergebnisse, die sich auf die meisten Instances und Images auswirken. Dieses Fenster erleichtert es, die Ergebnisse mit den größten Auswirkungen auf Ihre Umgebung zu identifizieren, die Einzelheiten der Ergebnisse zu überprüfen und Lösungsvorschläge zu überprüfen.

Verwalten Sie Ihre Ergebnisse mithilfe anpassbarer Ansichten

Zusätzlich zum Dashboard bietet die Amazon Inspector Inspector-Konsole eine Ergebnisansicht. Diese Seite listet alle Ergebnisse für Ihre Umgebung auf und enthält Einzelheiten zu den einzelnen Ergebnissen. Sie können die Ergebnisse nach Kategorie oder Schwachstellentyp gruppiert anzeigen.

In jeder Ansicht können Sie Ihre Ergebnisse mithilfe von Filtern weiter anpassen. Sie können Filter auch verwenden, um Unterdrückungsregeln zu erstellen, die unerwünschte Ergebnisse in Ihren Ansichten verbergen.

Sie können Filter und Unterdrückungsregeln verwenden, um Ergebnisberichte zu erstellen, in denen alle Ergebnisse oder eine benutzerdefinierte Auswahl von Ergebnissen angezeigt werden. Berichte können im CSV- oder JSON-Format generiert werden.

Überwachen und verarbeiten Sie Ergebnisse mit anderen Diensten und Systemen

Um die Integration mit anderen Diensten und Systemen zu unterstützen, veröffentlicht Amazon Inspector die Ergebnisse EventBridge als Finding Events auf Amazon. EventBridge ist ein serverloser Eventbus-Service, der Ergebnisdaten an Ziele wie AWS Lambda Funktionen und Amazon Simple Notification Service (Amazon SNS) -Themen weiterleiten kann. Damit EventBridge können Sie die Ergebnisse im Rahmen Ihrer bestehenden Sicherheits- und Compliance-Workflows nahezu in Echtzeit überwachen und verarbeiten.

Wenn Sie aktiviert haben AWS Security Hub, veröffentlicht Amazon Inspector die Ergebnisse auch im Security Hub. Security Hub ist ein Service, der Ihnen einen umfassenden Überblick über Ihren Sicherheitsstatus in Ihrer gesamten AWS Umgebung bietet und Ihnen hilft, Ihre Umgebung anhand von Industriestandards und Best Practices zu überprüfen. Mit Security Hub können Sie Ihre Ergebnisse im Rahmen einer umfassenderen Analyse der Sicherheitslage Ihres Unternehmens in einfacher überwachen und verarbeiten AWS.

Zugreifen auf Amazon Inspector

Amazon Inspector ist in den meisten Fällen verfügbar AWS-Regionen. Eine Liste der Regionen, in denen Amazon Inspector derzeit verfügbar ist, finden Sie unter Amazon Inspector Inspector-Endpunkte und Kontingente in der Amazon Web Services General Reference. Weitere Informationen AWS-Regionen dazu finden Sie unter Managing AWS-Regionen in der Amazon Web Services General Reference. In jeder Region können Sie auf folgende Weise mit Amazon Inspector arbeiten.

AWS Management-Konsole

Die AWS Management Console ist eine browserbasierte Oberfläche, mit der Sie AWS Ressourcen erstellen und verwalten können. Als Teil dieser Konsole bietet die Amazon Inspector Inspector-Konsole Zugriff auf Ihr Amazon Inspector Inspector-Konto und Ihre Ressourcen. Sie können Amazon Inspector Inspector-Aufgaben von der Amazon Inspector Inspector-Konsole aus ausführen.

AWS Befehlszeilentools

Mit AWS Befehlszeilentools können Sie Befehle an der Befehlszeile Ihres Systems ausgeben, um Amazon Inspector Inspector-Aufgaben auszuführen. Die Verwendung der Befehlszeile kann schneller und bequemer sein als die Verwendung der Konsole. Die Befehlszeilen-Tools können auch beim Erstellen von Skripts für -Aufgaben hilfreich sein.

AWS stellt zwei Gruppen von Befehlszeilentools bereit: das AWS Command Line Interface (AWS CLI) und das AWS -Tools für PowerShell. Informationen zur Installation und Verwendung von finden Sie im AWS Command Line Interface User Guide. AWS CLI Informationen zur Installation und Verwendung der Tools für PowerShell finden Sie im AWS -Tools für PowerShell Benutzerhandbuch.

AWS SDKs

AWS bietet SDKs Bibliotheken und Beispielcode für verschiedene Programmiersprachen und Plattformen, darunter Java, Go, Python, C++ und .NET. SDKs Sie bieten bequemen, programmatischen Zugriff auf Amazon Inspector und andere AWS-Services. Sie erledigen auch Aufgaben wie das kryptografische Signieren von Anfragen, das Verwalten von Fehlern und das automatische Wiederholen von Anfragen. Informationen zur Installation und Verwendung von finden Sie unter Tools AWS SDKs, auf denen Sie aufbauen können. AWS

Amazon Inspector REST-API

Die Amazon Inspector REST-API bietet Ihnen umfassenden, programmatischen Zugriff auf Ihr Amazon Inspector Inspector-Konto und Ihre Ressourcen. Mit dieser API können Sie HTTPS-Anfragen direkt an Amazon Inspector senden. Im Gegensatz zu den AWS Befehlszeilentools und SDKs bei der Verwendung dieser API muss Ihre Anwendung jedoch Details auf niedriger Ebene verarbeiten, z. B. die Generierung eines Hashs zum Signieren einer Anfrage.

Erste Schritte mit Amazon Inspector

Dieser Abschnitt enthält Informationen, die Sie vor der Aktivierung von Amazon Inspector berücksichtigen sollten, sowie ein Tutorial für die ersten Schritte, in dem beschrieben wird, wie Sie Amazon Inspector aktivieren und Ihre Ergebnisse in der Amazon Inspector Inspector-Konsole und mit der Amazon Inspector Inspector-API anzeigen können.

Themen

Vor der Aktivierung von Amazon Inspector

Bevor Sie Amazon Inspector aktivieren, sollten Sie Folgendes beachten:

Amazon Inspector ist ein regionaler Service

Ihre Daten werden dort gespeichert, AWS-Region wo Sie Amazon Inspector aktivieren. Wiederholen Sie die Schritte im ersten Teil des Tutorials „Erste Schritte“ für alle, AWS-Regionen in denen Sie Amazon Inspector verwenden möchten.

Amazon Inspector erstellt die serviceverknüpften Rollen AWSService RoleForAmazonInspector 2 und 2Agentless AWSService RoleForAmazonInspector

Eine serviceverknüpfte Rolle ist eine Rolle in AWS Identity and Access Management (IAM), die mit einem Dienst verknüpft ist. AWS AWSServiceRoleForAmazonInspector2 und AWSServiceRoleForAmazonInspector2Agentless ermöglichen Amazon Inspector den Zugriff, der für die Durchführung von Sicherheitsbewertungen AWS-Services erforderlich ist.

IAM-Identitäten mit Administratorrechten können Amazon Inspector aktivieren

Schützen Sie Ihre Anmeldeinformationen, indem Sie Benutzer mit IAM oder erstellen. AWS IAM Identity Center Auf diese Weise können Sie sicherstellen, dass Benutzer nur über die Berechtigungen verfügen, die für die Verwaltung von Amazon Inspector erforderlich sind. Weitere Informationen finden Sie unter AWS Verwaltete Richtlinie: AmazonInspectorFullAccess.

Hybrid-Scan wird automatisch aktiviert

Hybrid-Scanning umfasst agentenbasiertes Scannen und agentenloses Scannen. Standardmäßig verwendet Amazon Inspector diese Scanmethoden für alle geeigneten EC2 Amazon-Instances. Weitere Informationen finden Sie unter EC2 Amazon-Instances mit Amazon Inspector scannen.

Für Amazon ECR- und Lambda-Funktionsscans ist kein SSM-Agent erforderlich

Beim agentenbasierten Scannen wird der SSM-Agent verwendet, um das Softwareinventar zu sammeln. Beim agentenlosen Scannen werden Amazon EBS-Snapshots verwendet, um Softwareinventar zu sammeln.

> **Note**
>
> Standardmäßig ist der SSM-Agent bereits in EC2 Amazon-Instances installiert, die auf Amazon Machine Images basieren. In einigen Fällen müssen Sie den SSM-Agenten jedoch möglicherweise manuell aktivieren. Weitere Informationen finden Sie im AWS Systems Manager Benutzerhandbuch unter Arbeiten mit dem SSM-Agenten.

Die monatlichen Kosten basieren auf den gescannten Workloads

Weitere Informationen erhalten Sie unter Amazon Inspector: Preise.

Tutorial „Erste Schritte“: Amazon Inspector aktivieren

In diesem Thema wird beschrieben, wie Amazon Inspector für eine eigenständige Kontoumgebung (Mitgliedskonto) und eine Umgebung mit mehreren Konten (delegiertes Administratorkonto) aktiviert wird. Wenn Sie Amazon Inspector aktivieren, erkennt Amazon Inspector automatisch Workloads und scannt sie auf Softwareschwachstellen und unbeabsichtigte Netzwerkgefährdung.

Standalone account environment

Das folgende Verfahren beschreibt, wie Sie Amazon Inspector in der Konsole für ein Mitgliedskonto aktivieren. Um Amazon Inspector programmatisch zu aktivieren, inspector2 -. enablement-with-cli

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie Get Started.

3. Wählen Sie Amazon Inspector aktivieren.

Wenn Sie Amazon Inspector für ein eigenständiges Konto aktivieren, sind standardmäßig alle Scantypen aktiviert. Informationen zu Mitgliedskonten finden Sie unter Grundlegendes zum delegierten Administratorkonto und zu Mitgliedskonten in Amazon Inspector.

Multi-account environment

Das folgende Verfahren beschreibt, wie Amazon Inspector in der Konsole für ein delegiertes Administratorkonto aktiviert wird. Verwenden Sie das Amazon Inspector2-Shell-Skript, um Amazon Inspector für mehrere Konten programmatisch zu aktivieren. enablement-with-cli

> **Note**
>
> Sie müssen das AWS Organizations Verwaltungskonto verwenden, um dieses Verfahren abzuschließen. Nur das AWS Organizations Verwaltungskonto kann einen delegierten Administrator benennen. Für die Benennung eines delegierten Administrators sind möglicherweise Berechtigungen erforderlich. Weitere Informationen finden Sie unter Erforderliche Berechtigungen zum designieren eines delegierten Administrators.

Wenn Sie Amazon Inspector zum ersten Mal aktivieren, erstellt Amazon Inspector die serviceverknüpfte Rolle `AWSServiceRoleForAmazonInspector` für das Konto. Informationen darüber, wie Amazon Inspector serviceverknüpfte Rollen verwendet, finden Sie unterVerwenden von serviceverknüpften Rollen für Amazon Inspector.

So benennen Sie einen delegierten Administrator für Amazon Inspector

1. Melden Sie sich beim AWS Organizations Verwaltungskonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie Erste Schritte.
3. Geben Sie unter Delegierter Administrator die 12-stellige ID des Benutzers ein, den AWS-Konto Sie als delegierten Administrator festlegen möchten.
4. Wählen Sie Delegieren und dann erneut Delegieren aus.
5. (Optional) Wenn Sie Amazon Inspector für das AWS Organizations Verwaltungskonto aktivieren möchten, wählen Sie unter Serviceberechtigungen die Option Amazon Inspector aktivieren aus.

Wenn Sie einen delegierten Administrator benennen, sind standardmäßig alle Scantypen für das Konto aktiviert. Informationen zum delegierten Administratorkonto finden Sie unter Grundlegendes zum delegierten Administratorkonto und zu Mitgliedskonten in Amazon Inspector.

Automatisierte Scantypen in Amazon Inspector

Amazon Inspector verwendet eine speziell entwickelte Scan-Engine, die Ihre Ressourcen auf Softwareschwachstellen und unbeabsichtigte Netzwerkbedrohungen überwacht. Wenn Amazon Inspector eine Sicherheitslücke in Software oder eine unbeabsichtigte Netzwerkgefährdung erkennt, wird ein Ergebnis erstellt. Wenn Sie Amazon Inspector zum ersten Mal aktivieren, wird Ihr Konto automatisch für alle Scantypen registriert, darunter Amazon EC2 Amazon-Scanning, Amazon ECR Scanning und Lambda Standard-Scanning.

> **Note**
>
> Das Lambda-Code-Scannen ist eine optionale Ebene des Lambda-Funktionsscannens, die Sie jederzeit aktivieren können.

Themen

Übersicht der Amazon Inspector-Scantypen

Amazon Inspector bietet verschiedene Scantypen, die sich auf bestimmte Ressourcentypen in Ihrer AWS Umgebung konzentrieren.

EC2 Amazon-Scannen

Wenn Sie EC2 Amazon-Scanning aktivieren, scannt Amazon Inspector Ihre EC2 Instances auf allgemeine Sicherheitslücken und Sicherheitslücken (CVEs), Netzwerkprobleme, Probleme mit der Netzwerkerreichbarkeit sowie Sicherheitslücken im Betriebssystem und in Programmiersprachenpaketen. Amazon Inspector führt Scans mithilfe des auf Ihrer Instance

installierten SSM-Agenten oder mithilfe von Amazon EBS-Snapshots von Instances durch. Weitere Informationen finden Sie unter Scannen von EC2 Amazon-Instances mit Amazon Inspector. Wenn Sie EC2 Amazon-Scanning aktivieren, aktivieren Sie standardmäßig automatisch den Hybrid-Scanmodus. Weitere Informationen finden Sie unter Agentloses Scannen.

Amazon ECR-Scannen

Wenn Sie das Amazon ECR-Scannen aktivieren, konvertiert Amazon Inspector alle Repositorys in Ihrer privaten Registrierung von einfachen Scan-Container-Repositorys in erweiterte Scan-Repositorys. Sie können diese Einstellung mit Einschlussregeln so konfigurieren, dass nur On-Push-Scans oder ausgewählte Repositorys gescannt werden. Amazon Inspector scannt alle Bilder, die innerhalb der letzten 30 Tage übertragen oder innerhalb der letzten 90 Tage abgerufen wurden. Amazon Inspector überwacht Bilder standardmäßig weiterhin 90 Tage lang. Sie können diese Einstellung jederzeit ändern. Weitere Informationen finden Sie unter Scannen von Amazon Elastic Container Registry-Container-Images mit Amazon Inspector.

Lambda-Standardscannen

Wenn Sie das Lambda-Standardscannen aktivieren, erkennt Amazon Inspector alle Lambda-Funktionen in Ihrem Konto und scannt sie sofort auf Sicherheitslücken. Amazon Inspector scannt neue Lambda-Funktionen und -Layer, wenn sie bereitgestellt werden. Amazon Inspector scannt sie erneut, wenn sie aktualisiert werden oder wenn neue veröffentlicht CVEs werden. Weitere Informationen zum Scannen finden Sie unterAWS Lambda Scanfunktionen mit Amazon Inspector.

Lambda-Standardscannen + Lambda-Code-Scannen

Wenn Sie das Lambda-Code-Scannen aktivieren, erkennt Amazon Inspector die Lambda-Funktionen und -Layer in Ihrem Konto und scannt sie auf Code-Schwachstellen. Diese Art des Scannens bewertet Abhängigkeiten von Anwendungspaketen, die in einer Lambda-Funktion für verwendet werden. CVEs Wenn Sie diesen Scantyp aktivieren, aktivieren Sie auch den Lambda-Standardscan. Weitere Informationen finden Sie unter AWS Lambda Scanfunktionen mit Amazon Inspector.

Codesicherheit für Amazon Inspector

Dieser Scantyp nutzt die Amazon Q Developer-Scan-Engine, um Anwendungscode von Erstanbietern, Abhängigkeiten von Drittanbieteranwendungen und Infrastructure as Code auf Sicherheitslücken zu scannen. Weitere Informationen finden Sie unter Code Security for Amazon Inspector.

Einen Scantyp aktivieren

Sie können einen Scan-Typ jederzeit aktivieren. Wenn Sie einen Scan-Typ aktivieren, beginnt Amazon Inspector mit dem Scannen geeigneter Ressourcen für diesen Scantyp.

EC2 Amazon-Scannen

Dieser Scantyp extrahiert Metadaten aus einer EC2 Amazon-Instance, bevor die Metadaten mit Regeln verglichen werden, die in Sicherheitsempfehlungen gesammelt wurden. Wenn Sie diesen Scantyp aktivieren, scannt Amazon Inspector alle in Frage kommenden EC2 Amazon-Instances in Ihrem Konto auf Paketschwachstellen und Probleme mit der Netzwerkerreichbarkeit. Nachdem Sie diesen Scan-Typ aktiviert haben, können Sie auf der Registerkarte Instances sehen, wie viele Instances gescannt werden.

Amazon ECR-Scannen

Dieser Scantyp scannt Container-Images und Container-Repositorys in Amazon ECR. Wenn Sie diesen Scantyp aktivieren, ändern Sie die Einstellung der Scan-Konfiguration für Ihre private Registrierung von einfachem Scannen zu erweitertem Scannen. Nachdem Sie das Amazon ECR-Scannen aktiviert haben, können Sie auf den Registerkarten Container-Images und Container-Repositorys sehen, wie viele Bilder und Repositorys gescannt werden.

Lambda-Standardscannen + Lambda-Code-Scannen

Lambda-Standardscan ist der standardmäßige Lambda-Scantyp. Wenn Sie das Lambda-Standardscannen aktivieren, werden alle Ihre Lambda-Funktionen auf Softwareschwachstellen gescannt, sofern sie in den letzten 90 Tagen aufgerufen oder aktualisiert wurden. Nachdem Sie den Lambda-Standardscan aktiviert haben, sehen Sie auf der Registerkarte Lambda-Funktionen, wie viele Lambda-Funktionen gescannt werden.

Das Scannen von Lambda-Code scannt benutzerdefinierten Anwendungscode in einer Lambda-Funktion. Wenn Sie das Lambda-Code-Scanning aktivieren, werden alle Ihre Lambda-Funktionen auf Code-Schwachstellen gescannt, sofern sie in den letzten 90 Tagen aufgerufen oder aktualisiert wurden. Nachdem Sie den Lambda-Standardscan aktiviert haben, können Sie auf der Registerkarte Lambda-Funktionen sehen, wie viele Lambda-Funktionen auf Code-Schwachstellen gescannt werden.

> **Note**
>
> Sie können das Lambda-Standardscannen und das Lambda-Code-Scannen unabhängig oder zusammen aktivieren.

[Amazon Inspector Codesicherheit]

Dieser Scantyp durchsucht Anwendungscode von Erstanbietern, Abhängigkeiten von Drittanbieteranwendungen und Infrastructure as Code auf Sicherheitslücken. Wenn Sie Code Security aktivieren, beginnt Amazon Inspector, Ihre Code-Repositorys auf der Grundlage Ihrer Scan-Konfigurationen auf Code-Schwachstellen zu scannen. Nachdem Sie Amazon Inspector Code Security aktiviert haben, können Sie auf der Registerkarte Code-Repositorys sehen, wie viele Code-Repositorys gescannt werden.

Scans aktivieren

Das folgende Verfahren beschreibt, wie Sie einen Scantyp in Amazon Inspector aktivieren.

> **Note**
>
> Wenn Sie der delegierte Administrator für eine AWS Organisation sind, können Sie Amazon Inspector-Scantypen für mehrere Konten in mehreren Regionen mithilfe eines Shell-Skripts aktivieren. Weitere Informationen finden Sie unter inspector2 - on. enablement-with-cli GitHub Andernfalls führen Sie die folgenden Schritte aus, während Sie als delegierter Amazon Inspector-Administrator angemeldet sind.

Console

Um Scans zu aktivieren

1. Öffnen Sie die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie mit der AWS-Region Auswahl in der oberen rechten Ecke der Seite die Region aus, in der Sie einen neuen Scantyp aktivieren möchten.
3. Wählen Sie im Navigationsbereich Kontoverwaltung aus.

4. Wählen Sie auf der Seite Kontoverwaltung die Konten aus, für die Sie einen Scantyp aktivieren möchten.
5. Wählen Sie Aktivieren und wählen Sie die Art des Scannens aus, den Sie aktivieren möchten.
6. (Empfohlen) Wiederholen Sie diese Schritte AWS-Region für jeden, für den Sie diesen Scantyp aktivieren möchten.

API

Führen Sie den Vorgang „API aktivieren“ aus. Geben Sie in der Anfrage das Konto an, für das IDs Sie Scans aktivieren, und das Idempotenz-Token und einen oder mehrere von`EC2`,, oder `LAMBDA_CODE` für`ECR`, oder`LAMBDA`, `resourceTypes` um Scans dieses Typs zu aktivieren.

Scannen von EC2 Amazon-Instances mit Amazon Inspector

Amazon Inspector Amazon EC2 Scanning extrahiert Metadaten aus Ihrer EC2 Instance, bevor die Metadaten mit Regeln verglichen werden, die in Sicherheitsempfehlungen gesammelt wurden. Amazon Inspector scannt Instances auf Sicherheitslücken in Paketen und Problemen mit der Erreichbarkeit des Netzwerks, um Ergebnisse zu erzielen. Amazon Inspector führt alle 12 Stunden Netzwerkerreichbarkeitsscans und Paketschwachstellenscans in einem variablen Rhythmus durch, der von der Scanmethode abhängt, die mit der Instance verknüpft ist. EC2

Scans nach Sicherheitslücken in Paketen können mit einer agentenbasierten oder agentenlosen Scanmethode durchgeführt werden. Beide Scanmethoden bestimmen, wie und wann Amazon Inspector das Softwareinventar von einer EC2 Instance für Paketschwachstellenscans erfasst. Agentenbasiertes Scannen erfasst Softwareinventar mithilfe des SSM-Agenten, und agentenloses Scannen erfasst Softwareinventar mithilfe von Amazon EBS-Snapshots.

Amazon Inspector verwendet die Scanmethoden, die Sie für Ihr Konto aktivieren. Wenn Sie Amazon Inspector zum ersten Mal aktivieren, wird Ihr Konto automatisch für das Hybrid-Scannen registriert, das beide Scanmethoden verwendet. Sie können diese Einstellung jedoch jederzeit ändern. Informationen zur Aktivierung eines Suchtyps finden Sie unter Einen Scantyp aktivieren. Dieser Abschnitt enthält Informationen zum EC2 Scannen durch Amazon.

> **Note**
>
> Amazon EC2 Scanning scannt keine Dateisystemverzeichnisse, die sich auf virtuelle Umgebungen beziehen, auch wenn sie im Rahmen einer gründlichen Inspektion bereitgestellt

wurden. Beispielsweise `/var/lib/docker/` wird der Pfad nicht gescannt, da er häufig für Container-Laufzeiten verwendet wird.

Agentengestütztes Scannen

Agentenbasierte Scans werden kontinuierlich mit dem SSM-Agenten auf allen geeigneten Instanzen durchgeführt. Für agentenbasierte Scans verwendet Amazon Inspector SSM-Verknüpfungen und über diese Verknüpfungen installierte Plugins, um Softwarebestand aus Ihren Instances zu sammeln. Zusätzlich zu Paket-Schwachstellenscans für Betriebssystempakete kann das agentenbasierte Scannen von Amazon Inspector auch Paketschwachstellen in Paketen in Programmiersprachenpaketen in Linux-basierten Instances erkennen. Tiefeninspektion von Amazon Inspector für Linux-basierte EC2 Amazon-Instances

Der folgende Prozess erklärt, wie Amazon Inspector SSM verwendet, um Inventar zu sammeln und agentenbasierte Scans durchzuführen:

1. Amazon Inspector erstellt SSM-Verknüpfungen in Ihrem Konto, um Inventar aus Ihren Instances zu sammeln. Bei einigen Instance-Typen (Windows und Linux) installieren diese Verknüpfungen Plugins auf einzelnen Instances, um Inventar zu sammeln.
2. Mithilfe von SSM extrahiert Amazon Inspector Paketinventar aus einer Instance.
3. Amazon Inspector bewertet das extrahierte Inventar und generiert Ergebnisse für alle erkannten Sicherheitslücken.

> **Note**
>
> Für agentenbasiertes Scannen muss die EC2 Amazon-Instance von SSM in derselben verwaltet werden. AWS-Konto

In Frage kommende Instanzen

Amazon Inspector verwendet die agentenbasierte Methode, um eine Instance zu scannen, wenn sie die folgenden Bedingungen erfüllt:

- Die Instance hat ein unterstütztes Betriebssystem. Eine Liste der unterstützten Betriebssysteme finden Sie in der Spalte Unterstützung für agentengestütztes Scannen unter. the section called “Unterstützte Betriebssysteme: EC2 Amazon-Scanning”

- Die Instance wird nicht von Scans durch Amazon Inspector EC2 Inspector-Ausschluss-Tags ausgeschlossen.
- Die Instance wird SSM-verwaltet. Anweisungen zur Überprüfung und Konfiguration des Agenten finden Sie unter. Den SSM-Agenten konfigurieren

Verhalten beim Scannen auf Agentenbasis

Bei Verwendung der agentenbasierten Scanmethode initiiert Amazon Inspector in den folgenden Situationen neue Schwachstellenscans von EC2 Instances:

- Wenn Sie eine neue EC2 Instance starten.
- Wenn Sie neue Software auf einer vorhandenen EC2 Instanz (Linux und Mac) installieren.
- Wenn Amazon Inspector seiner Datenbank ein neues CVE-Element (Common Vulnerabilities and Exposures) hinzufügt und dieses CVE für Ihre EC2 Instance (Linux und Mac) relevant ist.

Amazon Inspector aktualisiert das Feld Zuletzt gescannt für eine EC2 Instance, wenn ein erster Scan abgeschlossen ist. Danach wird das Feld Zuletzt gescannt aktualisiert, wenn Amazon Inspector das SSM-Inventar auswertet (standardmäßig alle 30 Minuten) oder wenn eine Instance erneut gescannt wird, weil ein neuer CVE, der sich auf diese Instance auswirkt, zur Amazon Inspector Inspector-Datenbank hinzugefügt wurde.

Sie können auf der Kontoverwaltungsseite auf der Registerkarte Instances überprüfen, wann eine EC2 Instance zuletzt auf Sicherheitslücken gescannt wurde, oder indem Sie den Befehl verwenden. ListCoverage

Den SSM-Agenten konfigurieren

Damit Amazon Inspector mithilfe der agentenbasierten Scanmethode Softwareschwachstellen für eine EC2 Amazon-Instance erkennen kann, muss es sich bei der Instance um eine verwaltete Instance in Amazon EC2 Systems Manager (SSM) handeln. Bei einer von SSM verwalteten Instance ist der SSM-Agent installiert und läuft, und SSM ist berechtigt, die Instance zu verwalten. Wenn Sie SSM bereits zur Verwaltung Ihrer Instanzen verwenden, sind für agentenbasierte Scans keine weiteren Schritte erforderlich.

Der SSM-Agent wird standardmäßig auf EC2 Instances installiert, die aus einigen Amazon Machine Images (AMIs) erstellt wurden. Weitere Informationen finden Sie unter Über SSM Agent im AWS Systems Manager Benutzerhandbuch. Selbst wenn er installiert ist, müssen Sie den SSM-Agenten möglicherweise manuell aktivieren und SSM die Berechtigung zur Verwaltung Ihrer Instanz erteilen.

Das folgende Verfahren beschreibt, wie Sie eine EC2 Amazon-Instance mithilfe eines IAM-Instance-Profils als verwaltete Instance konfigurieren. Das Verfahren enthält auch Links zu detaillierteren Informationen im AWS Systems Manager Benutzerhandbuch.

AmazonSSMManagedInstanceCoreist die empfohlene Richtlinie, die Sie verwenden sollten, wenn Sie ein Instanzprofil anhängen. Diese Richtlinie verfügt über alle Berechtigungen, die für das EC2 Scannen mit Amazon Inspector erforderlich sind.

> **Note**
>
> Mithilfe der SSM-Standardkonfiguration für die Hostverwaltung können Sie auch die SSM-Verwaltung all Ihrer EC2 Instances automatisieren, ohne IAM-Instanzprofile verwenden zu müssen. Weitere Informationen finden Sie unter Standardkonfiguration für die Host-Verwaltung.

Um SSM für eine EC2 Amazon-Instance zu konfigurieren

1. Wenn es noch nicht von Ihrem Betriebssystemanbieter installiert wurde, installieren Sie den SSM-Agent. Weitere Informationen finden Sie unter Arbeiten mit dem SSM-Agenten.
2. Verwenden Sie den AWS CLI , um zu überprüfen, ob der SSM-Agent ausgeführt wird. Weitere Informationen finden Sie unter Prüfen des Status des SSM-Agents und Starten des Agenten.
3. Erteilen Sie SSM die Erlaubnis, Ihre Instanz zu verwalten. Sie können die Erlaubnis erteilen, indem Sie ein IAM-Instanzprofil erstellen und es an Ihre Instanz anhängen. Wir empfehlen die Verwendung dieser AmazonSSMManagedInstanceCoreRichtlinie, da diese Richtlinie über die Berechtigungen für SSM Distributor, SSM Inventory und SSM State Manager verfügt, die Amazon Inspector für Scans benötigt. Anweisungen zum Erstellen eines Instanzprofils mit diesen Berechtigungen und zum Anhängen einer Instanz finden Sie unter Instanzberechtigungen für Systems Manager Systems Manager konfigurieren.
4. (Optional) Aktivieren Sie automatische Updates für den SSM-Agent. Weitere Informationen finden Sie unter Automatisieren von Updates für den SSM-Agenten.
5. (Optional) Konfigurieren Sie Systems Manager für die Verwendung eines Amazon Virtual Private Cloud (Amazon VPC) -Endpunkts. Weitere Informationen finden Sie unter Amazon VPC-Endpoints erstellen.

> **Important**
>
> Amazon Inspector benötigt eine Systems Manager State Manager-Zuordnung in Ihrem Konto, um den Bestand an Softwareanwendungen zu erfassen. Amazon Inspector erstellt automatisch eine Assoziation, die aufgerufen wird, `InspectorInventoryCollection-do-not-delete` falls noch keine vorhanden ist.
> Amazon Inspector benötigt außerdem eine Ressourcendatensynchronisierung und erstellt automatisch eine, die aufgerufen wird, `InspectorResourceDataSync-do-not-delete` falls noch keine vorhanden ist. Weitere Informationen finden Sie unter Konfiguration der Ressourcendatensynchronisierung für Inventar im AWS Systems Manager Benutzerhandbuch. Für jedes Konto kann eine festgelegte Anzahl von Ressourcendatensynchronisierungen pro Region festgelegt werden. Weitere Informationen finden Sie unter Maximale Anzahl von Ressourcendatensynchronisierungen (AWS-Konto pro Region) in SSM-Endpunkten und -Kontingenten.

Für das Scannen erstellte SSM-Ressourcen

Amazon Inspector benötigt eine Reihe von SSM-Ressourcen in Ihrem Konto, um EC2 Amazon-Scans auszuführen. Die folgenden Ressourcen werden erstellt, wenn Sie das Amazon EC2 Inspector-Scannen zum ersten Mal aktivieren:

> **Note**
>
> Wenn eine dieser SSM-Ressourcen gelöscht wird, während Amazon Inspector Amazon EC2 Scanning für Ihr Konto aktiviert ist, versucht Amazon Inspector, sie beim nächsten Scanintervall neu zu erstellen.

`InspectorInventoryCollection-do-not-delete`

Dies ist eine Systems Manager State Manager (SSM) -Zuordnung, die Amazon Inspector verwendet, um Softwareanwendungsinventar aus Ihren EC2 Amazon-Instances zu sammeln. Wenn Ihr Konto bereits über eine SSM-Verknüpfung für die Erfassung von Inventar verfügt`InstanceIds*`, verwendet Amazon Inspector diese, anstatt eine eigene zu erstellen.

`InspectorResourceDataSync-do-not-delete`

Dies ist eine Ressourcendatensynchronisierung, die Amazon Inspector verwendet, um gesammelte Inventardaten von Ihren EC2 Amazon-Instances an einen Amazon S3-Bucket zu senden, der Amazon Inspector gehört. Weitere Informationen finden Sie unter Konfiguration der Ressourcendatensynchronisierung für Inventar im AWS Systems Manager Benutzerhandbuch.

`InspectorDistributor-do-not-delete`

Dies ist eine SSM-Verknüpfung, die Amazon Inspector zum Scannen von Windows-Instances verwendet. Diese Assoziation installiert das Amazon Inspector SSM-Plugin auf Ihren Windows-Instances. Wenn die Plugin-Datei versehentlich gelöscht wird, wird sie durch diese Verknüpfung beim nächsten Zuordnungsintervall erneut installiert.

`InvokeInspectorSsmPlugin-do-not-delete`

Dies ist eine SSM-Verknüpfung, die Amazon Inspector zum Scannen von Windows-Instances verwendet. Diese Zuordnung ermöglicht es Amazon Inspector, Scans mithilfe des Plug-ins zu initiieren. Sie können damit auch benutzerdefinierte Intervalle für Scans von Windows-Instances festlegen. Weitere Informationen finden Sie unter Einstellung benutzerdefinierter Zeitpläne für Windows Instanzscans.

`InspectorLinuxDistributor-do-not-delete`

Dies ist eine SSM-Assoziation, die Amazon Inspector für Amazon EC2 Linux Deep Inspection verwendet. Diese Assoziation installiert das Amazon Inspector SSM-Plugin auf Ihren Linux-Instances.

`InvokeInspectorLinuxSsmPlugin-do-not-delete`

Dies ist eine SSM-Verbindung, die Amazon Inspector für Amazon EC2 Linux Deep Inspection verwendet. Diese Zuordnung ermöglicht es Amazon Inspector, Scans mithilfe des Plug-ins zu initiieren.

> **Note**
>
> Wenn Sie Amazon Inspector Amazon EC2 Scanning oder Deep Inspection deaktivieren, `InvokeInspectorLinuxSsmPlugin-do-not-delete` wird die SSM-Ressource nicht mehr aufgerufen.

Scannen ohne Agenten

Amazon Inspector verwendet die agentenlose Scanmethode für berechtigte Instances, wenn sich Ihr Konto im Hybrid-Scanmodus befindet. Der Hybrid-Scanmodus umfasst agentenbasierte und agentenlose Scans und wird automatisch aktiviert, wenn Sie EC2 Amazon-Scanning aktivieren.

Für Scans ohne Agenten verwendet Amazon Inspector EBS-Snapshots, um ein Softwareinventar aus Ihren Instances zu erfassen. Beim agentenlosen Scannen werden Instances nach Sicherheitslücken im Betriebssystem und in Paketen der Anwendungsprogrammiersprache durchsucht.

> **Note**
>
> Beim Scannen von Linux-Instances auf Sicherheitslücken in Paketen in der Programmiersprache werden mit der agentenlosen Methode alle verfügbaren Pfade gescannt, wohingegen das agentenbasierte Scannen nur die Standardpfade und zusätzliche Pfade scannt, die Sie als Teil angeben. Tiefeninspektion von Amazon Inspector für Linux-basierte EC2 Amazon-Instances Dies kann dazu führen, dass dieselbe Instanz unterschiedliche Ergebnisse erzielt, je nachdem, ob sie mit der agentenbasierten Methode oder der agentenlosen Methode gescannt wird.

Der folgende Prozess erklärt, wie Amazon Inspector EBS-Snapshots verwendet, um Inventar zu sammeln und agentenlose Scans durchzuführen:

1. Amazon Inspector erstellt einen EBS-Snapshot aller Volumes, die an die Instance angehängt sind. Während Amazon Inspector es verwendet, wird der Snapshot in Ihrem Konto gespeichert und mit `InspectorScan` einem Tag-Schlüssel und einer eindeutigen Scan-ID als Tag-Wert gekennzeichnet.
2. Amazon Inspector ruft mithilfe von EBS Direct Daten aus den Snapshots ab APIs und bewertet sie auf Sicherheitslücken. Die Ergebnisse werden für alle erkannten Sicherheitslücken generiert.
3. Amazon Inspector löscht die EBS-Snapshots, die es in Ihrem Konto erstellt hat.

In Frage kommende Instances

Amazon Inspector verwendet die agentenlose Methode, um eine Instance zu scannen, wenn sie die folgenden Bedingungen erfüllt:

- Die Instance hat ein unterstütztes Betriebssystem. Weitere Informationen finden Sie in der Spalte >Unterstützung für agentengestütztes Scannen von. the section called “Unterstützte Betriebssysteme: EC2 Amazon-Scanning”
- Die Instanz hat den Status `Unmanaged EC2 instanceStale inventory`, oder. `No inventory`
- Die Instance wird von Amazon EBS unterstützt und hat eines der folgenden Dateisystemformate:
  - `ext3`
  - `ext4`
  - `xfs`
- Die Instance ist nicht von Scans über EC2 Amazon-Ausschluss-Tags ausgeschlossen.
- Die Anzahl der an die Instance angehängten Volumes beträgt weniger als 8 und ihre Gesamtgröße beträgt höchstens 1200 GB.

Verhalten beim Scannen ohne Agenten

Wenn Ihr Konto für Hybrid-Scanning konfiguriert ist, führt Amazon Inspector alle 24 Stunden agentenlose Scans auf geeigneten Instances durch. Amazon Inspector erkennt und scannt jede Stunde neue infrage kommende Instances, einschließlich neuer Instances ohne SSM-Agenten oder bereits existierende Instances mit Status, der auf geändert wurde. `SSM_UNMANAGED`

Amazon Inspector aktualisiert das Feld Zuletzt gescannt für eine EC2 Amazon-Instance, wenn es nach einem agentenlosen Scan extrahierte Snapshots aus einer Instance scannt.

Sie können auf der Kontoverwaltungsseite auf der Registerkarte Instances überprüfen, wann eine EC2 Instance zuletzt auf Sicherheitslücken gescannt wurde, oder indem Sie den ListCoverageBefehl verwenden.

Den Scanmodus verwalten

Ihr EC2 Scanmodus bestimmt, welche Scanmethoden Amazon Inspector bei der Durchführung von EC2 Scans in Ihrem Konto verwendet. Sie können den Scanmodus für Ihr Konto auf der Seite mit den EC2 Scaneinstellungen unter Allgemeine Einstellungen einsehen. Eigenständige Konten oder von Amazon Inspector delegierte Administratoren können den Scanmodus ändern. Wenn Sie den Scanmodus als delegierter Administrator von Amazon Inspector festlegen, wird dieser Scanmodus für alle Mitgliedskonten in Ihrer Organisation festgelegt. Amazon Inspector bietet die folgenden Scanmodi:

Agentengestütztes Scannen — In diesem Scanmodus verwendet Amazon Inspector ausschließlich die agentenbasierte Scanmethode, um nach Sicherheitslücken in Paketen zu suchen. Dieser Scanmodus scannt nur SSM-verwaltete Instances in Ihrem Konto, bietet jedoch den Vorteil, dass als Reaktion auf neue CVES oder Änderungen an den Instances kontinuierliche Scans bereitgestellt werden. Agentenbasiertes Scannen bietet auch Amazon Inspector Deep Inspection für berechtigte Instances. Dies ist der Standard-Scanmodus für neu aktivierte Konten.

Hybrid-Scan — In diesem Scanmodus verwendet Amazon Inspector eine Kombination aus agentenbasierten und agentenlosen Methoden, um nach Sicherheitslücken in Paketen zu suchen. Für berechtigte EC2 Instances, auf denen der SSM-Agent installiert und konfiguriert ist, verwendet Amazon Inspector die agentenbasierte Methode. Für berechtigte Instances, die nicht über SSM verwaltet werden, verwendet Amazon Inspector die agentenlose Methode für berechtigte EBS-gestützte Instances.

Um den Scanmodus zu ändern

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie mit der AWS-Region Auswahl in der oberen rechten Ecke der Seite die Region aus, in der Sie Ihren Scanmodus ändern möchten. EC2
3. Wählen Sie im seitlichen Navigationsbereich unter Allgemeine Einstellungen die Option EC2 Scaneinstellungen aus.
4. Wählen Sie unter Scanmodus die Option Bearbeiten aus.
5. Wählen Sie einen Scanmodus und dann Änderungen speichern aus.

Instanzen von Amazon Inspector-Scans ausschließen

Sie können Windows Instances von Amazon Inspector-Scans ausschließenLinux, indem Sie diese Instances mit dem `InspectorEc2Exclusion` Schlüssel kennzeichnen. Die Angabe eines Tag-Werts ist optional. Informationen zum Hinzufügen von Tags finden Sie unter Taggen Sie Ihre EC2 Amazon-Ressourcen.

Wenn Sie eine Instance für den Ausschluss aus Amazon Inspector-Scans kennzeichnen, markiert Amazon Inspector die Instance als ausgeschlossen und erstellt keine Ergebnisse dafür. Das Amazon Inspector SSM-Plugin wird jedoch weiterhin aufgerufen. Um zu verhindern, dass das Plugin aufgerufen wird, müssen Sie den Zugriff auf Tags in den Instanz-Metadaten zulassen.

> **Note**
>
> Für ausgeschlossene Instanzen werden Ihnen keine Gebühren berechnet.

Darüber hinaus können Sie ein verschlüsseltes EBS-Volume von Scans ohne Agenten ausschließen, indem Sie den AWS KMS Schlüssel, mit dem das Volume verschlüsselt wurde, mit dem Tag kennzeichnen. `InspectorEc2Exclusion` Weitere Informationen finden Sie unter Kennzeichnen von Schlüsseln.

Unterstützte Betriebssysteme

Amazon Inspector scannt unterstützte Mac-, Windows- und EC2 Linux-Instances auf Sicherheitslücken in Betriebssystempaketen. Für Linux-Instances kann Amazon Inspector Ergebnisse für Anwendungsprogrammiersprachenpakete erstellen, die verwendet Tiefeninspektion von Amazon Inspector für Linux-basierte EC2 Amazon-Instances werden. Für Mac- und Windows-Instances werden nur Betriebssystempakete gescannt.

Informationen zu unterstützten Betriebssystemen, einschließlich der Betriebssysteme, die ohne SSM-Agent gescannt werden können, finden Sie unterStatuswerte für EC2 Amazon-Instances.

Tiefeninspektion von Amazon Inspector für Linux-basierte EC2 Amazon-Instances

Amazon Inspector erweitert die EC2 Scanabdeckung von Amazon um eine gründliche Inspektion. Mit einer gründlichen Inspektion erkennt Amazon Inspector Paketschwachstellen für Anwendungsprogrammiersprachenpakete in Ihren Linux-basierten EC2 Amazon-Instances. Amazon Inspector scannt Standardpfade für Programmiersprachen-Paketbibliotheken. Sie können jedoch zusätzlich zu den Pfaden, die Amazon Inspector standardmäßig scannt, benutzerdefinierte Pfade konfigurieren.

> **Note**
>
> Sie können Deep Inspection mit der Einstellung Standard-Host-Management-Konfiguration verwenden. Sie müssen jedoch eine Rolle erstellen oder verwenden, die mit den `ssm:GetParameter` Berechtigungen `ssm:PutInventory` und konfiguriert ist.

Um Deep Inspection-Scans für Ihre Linux-basierten EC2 Amazon-Instances durchzuführen, verwendet Amazon Inspector Daten, die mit dem Amazon Inspector SSM-Plugin gesammelt wurden. Um das Amazon Inspector SSM-Plug-In zu verwalten und Deep Inspection für Linux durchzuführen, erstellt Amazon Inspector automatisch die SSM-Verknüpfung `InvokeInspectorLinuxSsmPlugin-do-not-delete` in Ihrem Konto. Amazon Inspector sammelt alle 6 Stunden aktualisiertes Anwendungsinventar von Ihren Linux-basierten EC2 Amazon-Instances.

> **Note**
>
> Deep Inspection wird für Windows Mac-Instances nicht unterstützt.

In diesem Abschnitt wird beschrieben, wie Sie Amazon Inspector Deep Inspection für EC2 Amazon-Instances verwalten, einschließlich der Festlegung benutzerdefinierter Pfade für Amazon Inspector zum Scannen.

Themen

Auf Deep Inspection zugreifen oder diese deaktivieren

> **Note**
>
> Für Konten, die Amazon Inspector nach dem 17. April 2023 aktivieren, wird die Tiefeninspektion automatisch als Teil des EC2 Amazon-Scannings aktiviert.

Um die Tiefeninspektion zu verwalten

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home
2. Wählen Sie im Navigationsbereich Allgemeine Einstellungen und dann EC2 Amazon-Scaneinstellungen aus.

3. Unter Deep Inspection of Amazon EC2 Instance können Sie benutzerdefinierte Pfade für Ihre Organisation oder für Ihr eigenes Konto festlegen.

Sie können den Aktivierungsstatus programmgesteuert für ein einzelnes Konto mit der GetEcDeepInspectionConfiguration2-API überprüfen. Sie können den Aktivierungsstatus programmgesteuert für mehrere Konten mit der API überprüfen. BatchGetMemberEc2DeepInspectionStatus

Wenn Sie Amazon Inspector vor dem 17. April 2023 aktiviert haben, können Sie Deep Inspection über das Konsolenbanner oder die UpdateEc2DeepInspectionConfigurationAPI aktivieren. Wenn Sie der delegierte Administrator für eine Organisation in Amazon Inspector sind, können Sie die BatchUpdateMemberEc2DeepInspectionStatusAPI verwenden, um Deep Inspection für sich und Ihre Mitgliedskonten zu aktivieren.

Sie können Deep Inspection über die UpdateEc2DeepInspectionConfigurationAPI deaktivieren. Mitgliedskonten in einer Organisation können Deep Inspection nicht deaktivieren. Stattdessen muss das Mitgliedskonto von seinem delegierten Administrator mithilfe der BatchUpdateMemberEc2DeepInspectionStatusAPI deaktiviert werden.

Benutzerdefinierte Pfade für die Tiefeninspektion mit Amazon Inspector

Sie können benutzerdefinierte Pfade festlegen, damit Amazon Inspector bei der Tiefeninspektion Ihrer EC2 Linux-Amazon-Instances scannt. Wenn Sie einen benutzerdefinierten Pfad festlegen, scannt Amazon Inspector Pakete in diesem Verzeichnis und allen Unterverzeichnissen darin.

Alle Konten können bis zu 5 benutzerdefinierte Pfade definieren. Der delegierte Administrator einer Organisation kann 10 benutzerdefinierte Pfade definieren.

Amazon Inspector scannt alle benutzerdefinierten Pfade zusätzlich zu den folgenden Standardpfaden, die Amazon Inspector für alle Konten scannt:

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

> **Note**
>
> Benutzerdefinierte Pfade müssen lokale Pfade sein. Amazon Inspector scannt keine zugewiesenen Netzwerkpfade, wie z. B. Netzwerkdateisystem-Mounts oder Amazon S3 S3-Dateisystem-Mounts.

Formatieren benutzerdefinierter Pfade

Ein benutzerdefinierter Pfad darf nicht länger als 256 Zeichen sein. Im Folgenden finden Sie ein Beispiel dafür, wie ein benutzerdefinierter Pfad aussehen könnte:

Beispiel für einen Pfad

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

> **Note**
>
> Das Paketlimit pro Instanz beträgt 5.000. Die maximale Zeit für die Erfassung des Paketinventars beträgt 15 Minuten. Amazon Inspector empfiehlt, benutzerdefinierte Pfade zu wählen, um diese Beschränkungen zu umgehen.

Einen benutzerdefinierten Pfad in der Amazon Inspector Inspector-Konsole und mit der Amazon Inspector Inspector-API einrichten

Die folgenden Verfahren beschreiben, wie Sie einen benutzerdefinierten Pfad für Amazon Inspector Deep Inspection in der Amazon Inspector Inspector-Konsole und mit der Amazon Inspector Inspector-API festlegen. Nachdem Sie einen benutzerdefinierten Pfad festgelegt haben, nimmt Amazon Inspector den Pfad in die nächste Tiefeninspektion auf.

Console

1. Melden Sie sich AWS Management Console als delegierter Administrator an und öffnen Sie die Amazon Inspector Inspector-Konsole unter v2/home https://console.aws.amazon.com/inspector/
2. Verwenden Sie den AWS-Region Selektor, um die Region auszuwählen, in der Sie das Lambda-Standardscannen aktivieren möchten.

3. Wählen Sie im Navigationsbereich Allgemeine Einstellungen und anschließend EC2 Scaneinstellungen aus.
4. Wählen Sie unter Benutzerdefinierte Pfade für Ihr eigenes Konto die Option Bearbeiten aus.
5. Geben Sie in den Pfad-Textfeldern Ihre benutzerdefinierten Pfade ein.
6. Wählen Sie Speichern.

API

Führen Sie den Befehl UpdateEc2DeepInspectionConfiguration aus. `packagePaths`Geben Sie ein Array von Pfaden an, die gescannt werden sollen.

Benutzerdefinierte Zeitpläne für die Tiefeninspektion mit Amazon Inspector

Standardmäßig erfasst Amazon Inspector alle 6 Stunden ein Anwendungsinventar von EC2 Amazon-Instances. Sie können jedoch die folgenden Befehle ausführen, um zu steuern, wie oft Amazon Inspector dies tut.

Beispielbefehl 1: Zuordnungen auflisten, um die Zuordnungs-ID und das aktuelle Intervall anzuzeigen

Der folgende Befehl zeigt die Zuordnungs-ID für die Assoziation an`InvokeInspectorLinuxSsmPlugin-do-not-delete`.

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

Beispielbefehl 2: Die Zuordnung so aktualisieren, dass sie ein neues Intervall einschließt

Der folgende Befehl verwendet die Zuordnungs-ID für die Zuordnung`InvokeInspectorLinuxSsmPlugin-do-not-delete`. Sie können die Rate für einen Zeitraum `schedule-expression` zwischen 6 Stunden und einem neuen Intervall, z. B. 12 Stunden, festlegen.

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

```
--region your-Region
```

> **Note**
>
> Je nach Anwendungsfall können Sie das tägliche SSM-Inventarlimit überschreiten, wenn Sie den Tarif für `schedule-expression` einen Tarif zwischen 6 Stunden und einem Intervall wie 30 Minuten festlegen. Dies führt zu verzögerten Ergebnissen und es kann vorkommen, dass EC2 Amazon-Instances mit teilweisem Fehlerstatus auftreten.

Unterstützte Programmiersprachen

Bei Linux-Instances kann Amazon Inspector Deep Inspector Ergebnisse für Anwendungsprogrammiersprachenpakete und Betriebssystempakete liefern.

Bei Mac- und Windows-Instances kann Amazon Inspector Deep Inspector nur Ergebnisse für Betriebssystempakete liefern.

Weitere Informationen zu unterstützten Programmiersprachen finden Sie unter Unterstützte Programmiersprachen: Amazon EC2 Deep Inspection.

Windows EC2 Instanzen mit Amazon Inspector scannen

Amazon Inspector erkennt automatisch alle unterstützten Windows Instances und nimmt sie ohne zusätzliche Aktionen in kontinuierliche Scans auf. Informationen darüber, welche Instances unterstützt werden, finden Sie unter Von Amazon Inspector unterstützte Betriebssysteme und Programmiersprachen. Amazon Inspector führt in regelmäßigen Abständen Windows Scans durch. WindowsInstances werden bei Entdeckung und dann alle 6 Stunden gescannt. Sie können das Standard-Scan-Intervall jedoch nach dem ersten Scan anpassen.

Wenn EC2 Amazon-Scanning aktiviert ist, erstellt Amazon Inspector die folgenden SSM-Verknüpfungen für Ihre Windows Ressourcen: `InspectorDistributor-do-not-deleteInspectorInventoryCollection-do-not-delete`, und`InvokeInspectorSsmPlugin-do-not-delete`. Um das Amazon Inspector SSM-Plugin auf Ihren Windows Instances zu installieren, verwendet die `InspectorDistributor-do-not-delete` SSM-Zuordnung das `AWS-ConfigureAWSPackage`SSM-Dokument und das `AmazonInspector2-InspectorSsmPluginSSM` Distributor-Paket. Weitere Informationen finden Sie unter Das Amazon Inspector SSM-Plugin für Windows. Um Instance-Daten zu sammeln und Amazon Inspector-Ergebnisse zu generieren, führt die `InvokeInspectorSsmPlugin-do-not-`

`delete` SSM-Vereinigung das Amazon Inspector SSM-Plugin in Intervallen von 6 Stunden aus. Sie können diese Einstellung jedoch mithilfe eines Cron- oder Rate-Ausdrucks anpassen.

> **Note**
>
> Amazon Inspector stellt aktualisierte OVAL-Definitionsdateien (Open Vulnerability and Assessment Language) im S3-Bucket bereit`inspector2-oval-prod-`*`your-AWS-Region`*. Der Amazon S3 S3-Bucket enthält OVAL-Definitionen, die in Scans verwendet werden. Diese OVAL-Definitionen sollten nicht geändert werden. Andernfalls sucht Amazon Inspector bei der Veröffentlichung nicht nach neuen CVEs Produkten.

Amazon Inspector-Scananforderungen für Windows Instances

Um eine Windows Instance zu scannen, setzt Amazon Inspector voraus, dass die Instance die folgenden Kriterien erfüllt:

- Die Instance ist eine von SSM verwaltete Instance. Anweisungen zum Einrichten Ihrer Instanz für das Scannen finden Sie unterDen SSM-Agenten konfigurieren.
- Das Instanzbetriebssystem ist eines der unterstützten Windows Betriebssysteme. Eine vollständige Liste der unterstützten Betriebssysteme finden Sie unterStatuswerte für EC2 Amazon-Instances.
- Auf der Instance ist das Amazon Inspector SSM-Plugin installiert. Amazon Inspector installiert bei Entdeckung automatisch das Amazon Inspector SSM-Plugin für verwaltete Instances. Einzelheiten zum Plugin finden Sie im nächsten Thema.

> **Note**
>
> Wenn Ihr Host in einer Amazon VPC ohne ausgehenden Internetzugang läuft, erfordert das Windows Scannen, dass Ihr Host auf regionale Amazon S3 S3-Endpunkte zugreifen kann. Informationen zur Konfiguration eines Amazon S3 S3-Amazon-VPC-Endpunkts finden Sie unter Erstellen eines Gateway-Endpunkts im Amazon Virtual Private Cloud-Benutzerhandbuch. Wenn Ihre Amazon VPC-Endpunktrichtlinie den Zugriff auf externe S3-Buckets einschränkt, müssen Sie ausdrücklich den Zugriff auf den von Amazon Inspector verwalteten Bucket in Ihrem zulassen AWS-Region , in dem die zur Bewertung Ihrer Instance verwendeten OVAL-Definitionen gespeichert sind. Dieser Bucket hat das folgende Format:. `inspector2-oval-prod-`*`REGION`*

Einstellung benutzerdefinierter Zeitpläne für Windows Instanzscans

Sie können die Zeit zwischen Ihren Windows EC2 Amazon-Instance-Scans anpassen, indem Sie einen Cron-Ausdruck oder einen Rate-Ausdruck für die `InvokeInspectorSsmPlugin-do-not-delete` Zuordnung mithilfe von SSM festlegen. Weitere Informationen finden Sie unter Referenz: Cron- und Rate-Ausdrücke für Systems Manager im AWS Systems Manager Benutzerhandbuch oder verwenden Sie die folgenden Anweisungen.

Wählen Sie eines der folgenden Codebeispiele aus, um die Scan-Taktfrequenz für Windows Instances von der Standardeinstellung 6 Stunden auf 12 Stunden zu ändern, indem Sie entweder einen Rate- oder einen Cron-Ausdruck verwenden.

In den folgenden Beispielen müssen Sie die AssociationIdfür die angegebene Assoziation verwenden. `InvokeInspectorSsmPlugin-do-not-delete` Sie können Ihre abrufen, AssociationIdindem Sie den folgenden AWS CLI Befehl ausführen:

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

> **Note**
>
> Da AssociationIdes sich um Regional handelt, müssen Sie zunächst für jede ID eine eindeutige ID abrufen AWS-Region. Anschließend können Sie den Befehl ausführen, um die Scanfrequenz in jeder Region zu ändern, in der Sie einen benutzerdefinierten Scan-Zeitplan für Windows Instances festlegen möchten.

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 * * ? *)"
```

Scannen von Amazon Elastic Container Registry-Container-Images mit Amazon Inspector

Amazon Inspector scannt Container-Images, die in Amazon Elastic Container Registry gespeichert sind, auf Softwareschwachstellen, um Sicherheitslücken in Paketen zu ermitteln. Wenn Sie das Amazon ECR-Scannen aktivieren, legen Sie Amazon Inspector als bevorzugten Scanservice für Ihre private Registrierung fest.

> **Note**
>
> Amazon ECR verwendet eine Registrierungsrichtlinie, um einem AWS Prinzipal Berechtigungen zu erteilen. Dieser Principal verfügt über die erforderlichen Berechtigungen, um Amazon Inspector APIs zum Scannen aufzurufen. Wenn Sie den Geltungsbereich Ihrer Registrierungsrichtlinie festlegen, dürfen Sie die `ecr:*` Aktion nicht hinzufügen oder `PutRegistryScanningConfiguration` eingebendeny. Dies führt zu Fehlern auf Registrierungsebene, wenn das Scannen für Amazon ECR aktiviert und deaktiviert wird.

Beim einfachen Scannen können Sie Ihre Repositorys so konfigurieren, dass sie bei Push-Scans scannen oder manuelle Scans durchführen. Mit der erweiterten Suchfunktion können Sie auf der Registrierungsebene nach Sicherheitslücken in Betriebssystemen und Programmiersprachenpaketen suchen. Einen side-by-side Vergleich der Unterschiede zwischen einfachem und erweitertem Scannen finden Sie in den häufig gestellten Fragen zu Amazon Inspector.

> **Note**
>
> Das einfache Scannen wird über Amazon ECR bereitgestellt und abgerechnet. Weitere Informationen finden Sie unter Amazon Elastic Container Registry — Preise. Erweitertes Scannen wird über Amazon Inspector bereitgestellt und abgerechnet. Weitere Informationen erhalten Sie unter Amazon Inspector: Preise.

Informationen zur Aktivierung des Amazon ECR-Scannens finden Sie unterEinen Scantyp aktivieren. Informationen darüber, wie Sie Ihre Ergebnisse einsehen können, finden Sie unterErgebnisse in

Amazon Inspector verwalten. Informationen dazu, wie Sie Ihre Ergebnisse auf Bildebene anzeigen können, finden Sie unter Scannen von Bildern im Amazon Elastic Container Registry User Guide. Sie können Ergebnisse auch verwalten, wenn sie AWS-Services nicht für einfaches Scannen verfügbar sind, wie AWS Security Hub bei Amazon EventBridge.

Dieser Abschnitt enthält Informationen zum Amazon ECR-Scannen und beschreibt, wie Sie das erweiterte Scannen für Amazon ECR-Repositorys konfigurieren.

Scanverhalten für Amazon ECR-Scans

Wenn Sie das Amazon ECR-Scannen zum ersten Mal aktivieren, erkennt Amazon Inspector Bilder, die innerhalb der letzten 14 Tage übertragen wurden. Amazon Inspector scannt dann die Bilder und setzt den Scanstatus auf`active`. Wenn kontinuierliches Scannen aktiviert ist, überwacht Amazon Inspector Bilder, sofern sie innerhalb von 14 Tagen (standardmäßig) übertragen wurden, das last-in-use Datum innerhalb von 14 Tagen liegt (standardmäßig) oder die Bilder innerhalb der konfigurierten Dauer des erneuten Scannens gescannt werden. Für Amazon Inspector Inspector-Konten, die vor dem 16. Mai 2025 erstellt wurden, ist die Standardkonfiguration ein erneutes Scannen, um Bilder zu überwachen, wenn sie innerhalb der letzten 90 Tage per Push oder Pull übertragen wurden. Weitere Informationen finden Sie unter Konfiguration der Dauer des Amazon ECR-Neuscans.

Für kontinuierliches Scannen initiiert Amazon Inspector in den folgenden Situationen neue Schwachstellenscans von Container-Images:

- Immer wenn ein neues Container-Image übertragen wird.
- Immer wenn Amazon Inspector seiner Datenbank ein neues CVE-Element (Common Vulnerabilities and Exposures) hinzufügt und dieses CVE für dieses Container-Image relevant ist (nur kontinuierliches Scannen).

Wenn Sie Ihr Repository für On-Push-Scannen konfigurieren, werden Bilder nur gescannt, wenn Sie sie per Push übertragen.

Sie können im Tab Container-Images auf der Kontoverwaltungsseite oder mithilfe der ListCoverageAPI überprüfen, wann ein Container-Image zuletzt auf Sicherheitslücken überprüft wurde. Amazon Inspector aktualisiert das Feld Zuletzt gescannt am eines Amazon ECR-Bilds als Reaktion auf die folgenden Ereignisse:

- Wenn Amazon Inspector einen ersten Scan eines Container-Images abschließt.

- Wenn Amazon Inspector ein Container-Image erneut scannt, weil ein neues CVE-Element (Common Vulnerabilities and Exposures), das sich auf dieses Container-Image auswirkt, zur Amazon Inspector Inspector-Datenbank hinzugefügt wurde.

Zuordnung von Container-Images zu laufenden Containern

Amazon Inspector bietet ein umfassendes Container-Sicherheitsmanagement, indem Container-Images laufenden Containern in Amazon Elastic Container Service (Amazon ECS) und Amazon Elastic Kubernetes Service (Amazon EKS) zugeordnet werden. Diese Zuordnungen bieten Einblicke in Sicherheitslücken bei Bildern auf laufenden Containern.

> **Note**
>
> Die verwaltete Richtlinie `AWSReadOnlyAccess` allein bietet keine ausreichenden Berechtigungen, um die Zuordnung zwischen Amazon ECR-Images und laufenden Containern anzuzeigen. Sie benötigen `AWSReadOnlyAccess` sowohl die als auch die `AWSInspector2ReadOnlyAccess` verwalteten Richtlinien, um Informationen zur Zuordnung von Container-Images anzuzeigen.

Mit dieser Funktion können Sie Abhilfemaßnahmen auf der Grundlage betrieblicher Risiken priorisieren und den Sicherheitsschutz im gesamten Container-Ökosystem aufrechterhalten. Sie können Container-Images überwachen, die aktuell verwendet werden und wann Container-Images in den letzten 24 Stunden zuletzt auf einem Amazon ECS- oder Amazon EKS-Cluster verwendet wurden. Bei neuen Images oder Konten kann es bis zu 36 Stunden dauern, bis Daten verfügbar sind. Danach werden diese Daten alle 24 Stunden aktualisiert. Diese Informationen sind in Ihren Ergebnissen über die Amazon Inspector Inspector-Konsole auf dem Detailbildschirm für Ihre Container-Image-Ergebnisse und in der Amazon Inspector Inspector-API über die `ecrImageLastInUseAt` Filter `ecrImageInUseCount` und verfügbar.

> **Note**
>
> Diese Daten werden automatisch an Amazon ECR Findings gesendet, wenn Sie das Amazon ECR-Scannen aktivieren und Ihr Repository für kontinuierliches Scannen konfigurieren. Kontinuierliches Scannen muss auf Amazon ECR-Repository-Ebene konfiguriert werden. Weitere Informationen finden Sie unter Verbessertes Scannen im Amazon Elastic Container Registry User Guide.

Sie können Container-Images aus Clustern auch anhand ihres last-in-use Datums erneut scannen.

Diese Funktion wird auch auf Amazon ECS Amazon EKS Fargate unterstützt.

Unterstützte Betriebssysteme und Medientypen

Informationen zu unterstützten Betriebssystemen finden Sie unterUnterstützte Betriebssysteme: Amazon ECR-Scannen mit Amazon Inspector.

Amazon Inspector-Scans von Amazon ECR-Repositorys decken die folgenden unterstützten Medientypen ab:

Image-Manifest

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

Image-Konfiguration

- `"application/vnd.docker.container.image.v1+json"`
- `"application/vnd.oci.image.config.v1+json"`

Bildebenen

- `"application/vnd.docker.image.rootfs.diff.tar"`
- `"application/vnd.docker.image.rootfs.diff.tar.gzip"`
- `"application/vnd.docker.image.rootfs.foreign.diff.tar.gzip"`
- `"application/vnd.oci.image.layer.v1.tar"`
- `"application/vnd.oci.image.layer.v1.tar+gzip"`
- `"application/vnd.oci.image.layer.v1.tar+zstd"`
- `"application/vnd.oci.image.layer.nondistributable.v1.tar"`
- `"application/vnd.oci.image.layer.nondistributable.v1.tar+gzip"`

> **Note**
>
> Amazon Inspector unterstützt den `"application/vnd.docker.distribution.manifest.list.v2+json"` Medientyp für das Scannen von Amazon ECR-Repositorys nicht.

Konfiguration der Dauer des erneuten Scans von Amazon ECR

Die Einstellung für die Dauer des erneuten Scans in Amazon ECR bestimmt, wie lange Amazon Inspector kontinuierlich Container-Images in Repositorys überwacht. Sie konfigurieren die Dauer des erneuten Scans für das last-in-use Image-Datum, das Datum des letzten Abrufs und das Push-Datum. Es hat sich bewährt, die Dauer des erneuten Scans so zu konfigurieren, dass sie am besten zu Ihrer Umgebung passt.

Wenn Sie häufig Images erstellen, wählen Sie eine kürzere Scandauer. Wählen Sie für Bilder, die über einen längeren Zeitraum verwendet werden, eine längere Scandauer. Die Standardscandauer für neue Konten, einschließlich neuer Konten, die zu einer Organisation hinzugefügt wurden, beträgt 14 Tage.

Amazon Inspector überwacht weiterhin ein Bild und scannt es erneut, solange es zuletzt in einem Cluster verwendet oder innerhalb von 14 Tagen übertragen wurde (standardmäßig). Wenn ein Bild innerhalb des konfigurierten Push- und letzten Nutzungsdatums nicht per Push übertragen oder zuletzt in einem laufenden Container verwendet wurde, beendet Amazon Inspector die Überwachung. Es besteht die Möglichkeit, die Einstellung so zu ändern, dass Bilder bei Bedarf nach dem Datum des letzten Abrufs statt nach dem Datum der letzten Verwendung überwacht werden. Wenn Amazon Inspector die Überwachung eines Bilds beendet, wird der Statuscode für den Bildscan auf inaktiv und der Ursachencode auf abgelaufen gesetzt. Amazon Inspector plant dann, dass alle zugehörigen Bildergebnisse geschlossen werden.

Wenn Sie die Dauer des Push-Datums verlängern, wendet Amazon Inspector die Änderung auf alle aktiv gescannten Bilder in Repositorys an, die für kontinuierliches Scannen konfiguriert sind. Inaktive Bilder bleiben jedoch inaktiv, auch wenn Sie sie innerhalb der neuen Dauer per Push übertragen haben.

> **Note**
>
> Wenn Sie die Dauer des erneuten Scans von einem delegierten Administratorkonto aus konfigurieren, wendet Amazon Inspector die Einstellung auf alle Mitgliedskonten in der

Organisation an. Wenn das delegierte Administratorkonto das Amazon ECR-Scannen nicht aktiviert, kann es keine Cluster für ein API-Image anzeigen.

> **Note**
>
> Alle Einstellungen für die Dauer des erneuten Scans, die vor dem 16. Mai 2025 konfiguriert wurden, bleiben unverändert. Sie können weiterhin alle zuvor konfigurierten Standardeinstellungen verwenden.

Dauer des erneuten Scans von Bildern

Die Dauer des erneuten Scans von Bildern bestimmt, wie lange Amazon Inspector Bilder überwacht. Die Dauer des erneuten Scannens von Bildern umfasst zwei Modi: Datum der letzten Verwendung (Standard) oder Datum des letzten Abrufs. Wählen Sie Datum der letzten Verwendung (Standard), wenn Sie das Datum der letzten Verwendung aus Ihrer Amazon ECS/Amazon EKS-Cluster-Aktivität verwenden möchten. Wählen Sie Letztes Abrufdatum, wenn Sie das Datum des letzten Abrufs aus Ihren Amazon ECR-Bildern verwenden möchten, um Bilder erneut zu scannen. Die folgenden Optionen sind für die Dauer des erneuten Scans verfügbar:

- 14 Tage (Standard)
- 30 Tage
- 60 Tage
- 90 Tage
- 180 Tage

Dauer des Bild-Push-Datums

Die Dauer des Image-Push-Datums bestimmt, wie lange Amazon Inspector Bilder kontinuierlich überwacht, nachdem sie in Repositorys übertragen wurden. Die folgenden Optionen sind für die Dauer des erneuten Scans verfügbar:

- 14 Tage (Standard)
- 30 Tage
- 60 Tage

- 90 Tage
- 180 Tage
- Nutzungsdauer

So konfigurieren Sie die Dauer des erneuten Scans von Amazon ECR

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie den AWS-Region Ort aus, an dem Sie die Dauer des Amazon ECR-Neuscans konfigurieren möchten.
3. Wählen Sie im Navigationsbereich Allgemeine Einstellungen und dann ECR-Scaneinstellungen aus.
4. Wählen Sie unter Dauer des erneuten ECR-Scans den Modus für das erneute Scannen von Bildern und dann die entsprechende Dauer aus.
5. Wählen Sie unter Bild-Push-Datum das Bild-Push-Datum aus.
6. Wählen Sie Speichern.

AWS Lambda Scanfunktionen mit Amazon Inspector

Die Unterstützung von AWS Lambda Funktionen und Ebenen durch Amazon Inspector ermöglicht kontinuierliche automatisierte Bewertungen von Sicherheitslücken. Amazon Inspector bietet zwei Arten von Lambda-Funktionsscans:

Standard-Scanning mit Amazon Inspector Lambda

Dies ist der standardmäßige Lambda-Scantyp. Das Lambda-Standardscannen scannt Anwendungsabhängigkeiten innerhalb einer Lambda-Funktion und -Layer auf Paketschwachstellen.

Scannen von Lambda-Code mit Amazon Inspector

Dieser Scantyp scannt den benutzerdefinierten Anwendungscode in Ihrer Lambda-Funktion und Ihren Lambda-Ebenen auf Code-Schwachstellen. Sie können entweder das Lambda-Standardscannen oder das Lambda-Standardscannen mit Lambda-Code-Scanning aktivieren.

Wenn Sie den Lambda-Funktionsscan aktivieren, erstellt Amazon Inspector die folgenden AWS CloudTrail serviceverknüpften Kanäle in

Ihrem Konto: `cloudtrail:CreateServiceLinkedChannel` und. `cloudtrail:DeleteServiceLinkedChannel` Amazon Inspector verwaltet diese Kanäle und verwendet sie, um Ihre CloudTrail Ereignisse im Hinblick auf Scans zu überwachen. Diese Kanäle ermöglichen es Ihnen, CloudTrail Ereignisse in Ihrem Konto zu verfolgen, als ob Sie eine Spur erhalten hätten CloudTrail. Wir empfehlen dir, deinen eigenen Trail zu erstellen CloudTrail , um Ereignisse für dein Konto zu verwalten.

Informationen zur Aktivierung des Lambda-Funktionsscanners finden Sie unter Einen Scantyp aktivieren. Dieser Abschnitt enthält Informationen zum Scannen von Lambda-Funktionen.

Scanverhalten beim Scannen mit Lambda-Funktionen

Nach der Aktivierung scannt Amazon Inspector alle Lambda-Funktionen, die in den letzten 90 Tagen in Ihrem Konto aufgerufen oder aktualisiert wurden. Amazon Inspector initiiert Schwachstellenscans von Lambda-Funktionen in den folgenden Situationen:

- Sobald Amazon Inspector eine bestehende Lambda-Funktion entdeckt.
- Wenn Sie eine neue Lambda-Funktion für den Lambda-Service bereitstellen.
- Wenn Sie ein Update für den Anwendungscode oder die Abhängigkeiten einer vorhandenen Lambda-Funktion oder ihrer Layer bereitstellen.
- Immer wenn Amazon Inspector seiner Datenbank ein neues CVE-Element (Common Vulnerabilities and Exposures) hinzufügt und dieses CVE für Ihre Funktion relevant ist.

Amazon Inspector überwacht jede Lambda-Funktion während ihrer gesamten Lebensdauer, bis sie entweder gelöscht oder vom Scannen ausgeschlossen wird.

Sie können im Tab Lambda-Funktionen auf der Kontoverwaltungsseite oder mithilfe der API überprüfen, wann eine Lambda-Funktion zuletzt auf Sicherheitslücken überprüft wurde. ListCoverage Amazon Inspector aktualisiert das Feld Zuletzt gescannt am für eine Lambda-Funktion als Reaktion auf die folgenden Ereignisse:

- Wenn Amazon Inspector einen ersten Scan einer Lambda-Funktion abschließt.
- Wenn eine Lambda-Funktion aktualisiert wird.
- Wenn Amazon Inspector eine Lambda-Funktion erneut scannt, weil ein neues CVE-Element, das sich auf diese Funktion auswirkt, zur Amazon Inspector Inspector-Datenbank hinzugefügt wurde.

Unterstützte Laufzeiten und geeignete Funktionen

Amazon Inspector unterstützt unterschiedliche Laufzeiten für Lambda-Standardscans und Lambda-Code-Scans. Eine Liste der unterstützten Laufzeiten für jeden Scan-Typ finden Sie unter und. Unterstützte Laufzeiten: Amazon Inspector Lambda Standard-Scanning Unterstützte Laufzeiten: Amazon Inspector Lambda-Code-Scanning

Zusätzlich zu einer unterstützten Laufzeit muss eine Lambda-Funktion die folgenden Kriterien erfüllen, um für Amazon Inspector-Scans in Frage zu kommen:

- Die Funktion wurde in den letzten 90 Tagen aufgerufen oder aktualisiert.
- Die Funktion ist markiert`$LATEST`.
- Die Funktion ist nicht von Scans nach Tags ausgeschlossen.

> **Note**
>
> Lambda-Funktionen, die in den letzten 90 Tagen nicht aufgerufen oder geändert wurden, werden automatisch von Scans ausgeschlossen. Amazon Inspector setzt das Scannen einer automatisch ausgeschlossenen Funktion fort, wenn sie erneut aufgerufen wird oder wenn Änderungen am Lambda-Funktionscode vorgenommen werden.

Standard-Scanning mit Amazon Inspector Lambda

Das Standard-Scannen von Amazon Inspector Lambda identifiziert Softwareschwachstellen in den Abhängigkeiten von Anwendungspaketen, die Sie Ihrem Lambda-Funktionscode und den Lambda-Funktionsschichten hinzufügen. Wenn Ihre Lambda-Funktion beispielsweise eine Version des `python-jwt` Pakets mit einer bekannten Sicherheitslücke verwendet, generiert der Lambda-Standardscan einen Befund für diese Funktion.

Wenn Amazon Inspector eine Sicherheitslücke in den Abhängigkeiten Ihrer Lambda-Funktionsanwendung feststellt, erstellt Amazon Inspector eine detaillierte Suche nach dem Typ der Sicherheitslücke in Paketen.

Anweisungen zur Aktivierung eines Scan-Typs finden Sie unterEinen Scantyp aktivieren.

> **Note**
>
> Das Lambda-Standardscannen scannt nicht die AWS SDK-Abhängigkeit, die standardmäßig in der Lambda-Laufzeitumgebung installiert ist. Amazon Inspector scannt nur Abhängigkeiten, die mit dem Funktionscode hochgeladen oder von einer Ebene übernommen wurden.

> **Note**
>
> Wenn Sie das Standardscannen von Amazon Inspector Lambda deaktivieren, wird auch das Scannen von Amazon Inspector Lambda-Code deaktiviert.

Funktionen vom Lambda-Standardscan ausschließen

Sie können Lambda-Funktionen Tags hinzufügen, sodass Sie sie von Amazon Inspector Lambda-Standardscans ausschließen können. Das Ausschließen von Funktionen aus Scans kann verhindern, dass Warnmeldungen nicht bearbeitet werden können. Wenn Sie eine Funktion für den Ausschluss kennzeichnen, muss das Tag das folgende Schlüssel-Wert-Paar haben.

- Schlüssel: `InspectorExclusion`
- Wert: `LambdaStandardScanning`

In diesem Thema wird beschrieben, wie Sie eine Funktion kennzeichnen, sodass sie von Scans ausgeschlossen wird. Weitere Informationen zum Hinzufügen von Tags in Lambda finden Sie unter Verwenden von Tags in Lambda-Funktionen.

So schließen Sie eine Funktion von Scans aus

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Lambda-Konsole unter https://console.aws.amazon.com/lambda/.
2. Wählen Sie im Navigationsbereich Funktionen aus.
3. Wählen Sie den Namen der Funktion, die Sie von Amazon Inspector Lambda-Standardscans ausschließen möchten.
4. Wählen Sie Configuraton (Konfiguration) und dann Tags aus.
5. Wählen Sie Tags verwalten und dann Neues Tag hinzufügen.

a. Geben Sie für Key (Schlüssel) `InspectorExclusion` ein.

b. Geben Sie für Wert `LambdaStandardScanning` ein

6. Wählen Sie Save (Speichern) aus.

Scannen von Lambda-Code mit Amazon Inspector

> **Important**
>
> Diese Funktion erfasst Ausschnitte von Lambda-Funktionen, um erkannte Sicherheitslücken hervorzuheben. Diese Snippets können fest codierte Anmeldeinformationen und andere vertrauliche Materialien enthalten.

Mit dieser Funktion scannt Amazon Inspector Anwendungscode in einer Lambda-Funktion auf Codeschwachstellen, die auf bewährten AWS Sicherheitsmethoden basieren, um Datenlecks, Injektionsfehler, fehlende Verschlüsselung und schwache Kryptografie zu erkennen. Amazon Inspector verwendet automatisiertes Denken und maschinelles Lernen, um Ihren Anwendungscode für Lambda-Funktionen zu bewerten. Es verwendet auch interne Detektoren, die in Zusammenarbeit mit Amazon entwickelt wurden CodeGuru , um Richtlinienverstöße und Sicherheitslücken zu identifizieren. Weitere Informationen finden Sie in der CodeGuru Detector Library.

Amazon Inspector generiert eine Code-Schwachstelle, wenn es eine Sicherheitslücke im Anwendungscode Ihrer Lambda-Funktion entdeckt. Dieser Erkennungstyp umfasst einen Codeausschnitt, der das Problem zeigt und wo Sie das Problem in Ihrem Code finden können. Außerdem wird vorgeschlagen, wie das Problem behoben werden kann. Der Vorschlag umfasst plug-and-play Codeblöcke, mit denen Sie anfällige Codezeilen ersetzen können. Diese Codekorrekturen werden zusätzlich zu den allgemeinen Anleitungen zur Codebehebung für diesen Befundtyp bereitgestellt.

Vorschläge zur Codekorrektur basieren auf automatisierten Argumenten und generativen Diensten für künstliche Intelligenz. Einige Vorschläge zur Codekorrektur funktionieren möglicherweise nicht wie beabsichtigt. Sie sind für die Vorschläge zur Codekorrektur verantwortlich, die Sie übernehmen. Lesen Sie sich die Vorschläge zur Codekorrektur immer durch, bevor Sie sie übernehmen. Möglicherweise müssen Sie sie bearbeiten, um sicherzustellen, dass Ihr Code wie vorgesehen funktioniert. Weitere Informationen finden Sie in der Richtlinie für verantwortungsvolle KI.

Das Lambda-Code-Scannen kann eigenständig oder zusammen mit dem Lambda-Standard-Scanning aktiviert werden. Weitere Informationen finden Sie unter Einen Scantyp aktivieren. Informationen darüber, welche diese Funktion AWS-Regionen unterstützen, finden Sie unterVerfügbarkeit regionsspezifischer Feature.

Verschlüsselung Ihres Codes im Code — Sicherheitslücken

CodeGuru speichert Codefragmente, bei denen festgestellt wurde, dass sie im Zusammenhang mit einer Code-Schwachstelle stehen, die mithilfe von Lambda-Code-Scanning gefunden wurde. CodeGuru Steuert standardmäßig den AWS eigenen Schlüssel, der zur Verschlüsselung Ihres Codes verwendet wird. Sie können jedoch Ihren eigenen, vom Kunden verwalteten Schlüssel für die Verschlüsselung über die Amazon Inspector API verwenden. Weitere Informationen finden Sie unter Verschlüsselung im Ruhezustand für den Code in Ihren Ergebnissen

Funktionen vom Lambda-Code-Scannen ausschließen

Sie können Lambda-Funktionen Tags hinzufügen, sodass Sie sie von Amazon Inspector Lambda-Codescans ausschließen können. Wenn Sie Funktionen von Scans ausschließen, können Sie verhindern, dass Warnmeldungen nicht bearbeitet werden können. Wenn Sie eine Funktion für den Ausschluss kennzeichnen, muss das Tag das folgende Schlüssel-Wert-Paar haben.

- Schlüssel: `InspectorCodeExclusion`
- Wert — `LambdaCodeScanning`

In diesem Thema wird beschrieben, wie Sie eine Funktion kennzeichnen, sodass sie von Codescans ausgeschlossen wird. Weitere Informationen zum Hinzufügen von Tags in Lambda finden Sie unter Verwenden von Tags in Lambda-Funktionen.

So schließen Sie eine Funktion von Codescans aus

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Lambda-Konsole unter https://console.aws.amazon.com/lambda/.
2. Wählen Sie im Navigationsbereich Funktionen aus.
3. Wählen Sie den Namen der Funktion, die Sie von Amazon Inspector Lambda-Codescans ausschließen möchten.
4. Wählen Sie Configuraton (Konfiguration) und dann Tags aus.
5. Wählen Sie „Tags verwalten“ und dann „Neues Tag hinzufügen“.

a. Geben Sie für Key (Schlüssel) `InspectorCodeExclusion` ein.

b. Geben Sie für Wert `LambdaCodeScanning` ein

6. Wählen Sie Save (Speichern) aus.

Deaktivieren eines Scantyps in Amazon Inspector

Wenn Sie einen Scan-Typ deaktivieren, verlieren Sie den Zugriff auf alle Ergebnisse, die der Scantyp erbracht hat. Wenn Sie den Scan-Typ reaktivieren, scannt Amazon Inspector alle infrage kommenden Ressourcen, um neue Ergebnisse zu generieren. Wenn Sie Ihre Ergebnisse aufzeichnen möchten, können Sie sie als Ergebnisbericht in einen Amazon Simple Storage Service (Amazon S3) -Bucket exportieren. Weitere Informationen finden Sie unter Exportieren von Amazon Inspector Inspector-Ergebnisberichten. Wenn Sie einen Scan-Typ deaktivieren, können in dem AWS Konto, in dem Sie den Scan-Typ deaktiviert haben, die folgenden Änderungen auftreten:

EC2 Amazon-Scannen

Wenn Sie das EC2 Scannen von Amazon Inspector Amazon nach einem Konto deaktivieren, werden die folgenden SSM-Verknüpfungen gelöscht:

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

Darüber hinaus wird das Amazon Inspector SSM-Plugin von allen Windows Hosts entfernt. Weitere Informationen finden Sie unter Windows EC2 Instanz wird gescannt.

Amazon ECR-Scannen

Wenn Sie das Amazon ECR-Scannen für ein Konto deaktivieren, ändert sich der Kontotyp Amazon ECR von Erweitertes Scannen mit Amazon Inspector zu Standard-Scannen mit Amazon ECR.

Lambda-Standardscannen

Wenn Sie den Lambda-Standardscan für ein Konto deaktivieren, deaktivieren Sie den Lambda-Code-Scan, wenn der Scantyp aktiviert war. Sie löschen auch den CloudTrail serviceverknüpften Kanal, den Amazon Inspector erstellt, wenn Sie das Lambda-Standardscannen aktivieren.

Amazon Inspector Codesicherheit

Wenn Sie Code Security für Ihr Konto deaktivieren, löschen Sie alle damit verbundenen Integrationen, Projekte und Scankonfigurationen. Wenn Ihr Konto der delegierte Administrator für eine Organisation ist, deaktivieren Sie Code Security nur für Ihr Konto, und Mitgliedskonten werden zu eigenständigen Konten.

Scans deaktivieren

Wenn Sie alle Scanarten für ein Konto deaktivieren, wird Amazon Inspector für dieses Konto in diesem Konto deaktiviert. AWS-Region Weitere Informationen finden Sie unter Amazon Inspector deaktivieren.

Um dieses Verfahren für eine Umgebung mit mehreren Konten abzuschließen, folgen Sie diesen Schritten, während Sie als delegierter Amazon Inspector-Administrator angemeldet sind.

Console

Um Scans zu deaktivieren

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie mithilfe der AWS-Region Auswahl in der oberen rechten Ecke der Seite die Region aus, in der Sie Scans deaktivieren möchten.
3. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus.
4. Wählen Sie die Registerkarte Konten, um den Scanstatus eines Kontos anzuzeigen.
5. Aktivieren Sie das Kontrollkästchen jedes Kontos, für das Sie Scans deaktivieren möchten.
6. Wählen Sie Aktionen und wählen Sie unter den Deaktivierungsoptionen den Scantyp aus, den Sie deaktivieren möchten.
7. (Empfohlen) Wiederholen Sie diese Schritte AWS-Region für jeden Scantyp, für den Sie diesen Scantyp deaktivieren möchten.

API

Führen Sie den Vorgang „API deaktivieren“ aus. Geben Sie in der Anfrage das Konto an, für das IDs Sie Scans deaktivieren möchten, und `resourceTypes` geben Sie einen oder mehrere der,, oder an, an, an, an, an`EC2`, an`ECR`, an`LAMBDA`, `LAMBDA_CODE` um Scans zu deaktivieren.

Das Center for Internet Security (CIS) scannt nach EC2 Amazon-Instance-Betriebssystemen

Amazon Inspector CIS-Scans (CIS-Scans) vergleichen Ihre EC2 Amazon-Instance-Betriebssysteme, um sicherzustellen, dass Sie sie gemäß den vom Center for Internet Security festgelegten Best-Practice-Empfehlungen konfiguriert haben. CIS Security Benchmarks bietet branchenübliche Konfigurationsgrundlagen und bewährte Methoden für die sichere Konfiguration eines Systems. Sie können CIS-Scans durchführen oder planen, nachdem Sie Amazon EC2 Inspector-Scans für ein Konto aktiviert haben. Informationen zur Aktivierung von EC2 Amazon-Scans finden Sie unter Einen Scantyp aktivieren.

> **Note**
>
> CIS-Standards sind für x86_64-Betriebssysteme vorgesehen. Einige Prüfungen werden möglicherweise nicht ausgewertet oder geben ungültige Anweisungen zur Behebung von ARM-basierten Ressourcen zurück.

Amazon Inspector führt CIS-Scans auf EC2 Amazon-Ziel-Instances auf der Grundlage von Instance-Tags und Ihrem definierten Scan-Zeitplan durch. Amazon Inspector führt für jede Ziel-Instance eine Reihe von Instance-Prüfungen durch. Bei jeder Prüfung wird bewertet, ob Ihre Systemkonfiguration den spezifischen CIS-Benchmark-Empfehlungen entspricht. Jeder Check hat eine CIS-Check-ID und einen Titel, die einer CIS-Benchmark-Empfehlung für diese Plattform entsprechen. Wenn ein CIS-Scan abgeschlossen ist, können Sie anhand der Ergebnisse sehen, welche Instanzprüfungen für dieses System bestanden, übersprungen oder fehlgeschlagen sind.

> **Note**
>
> Um CIS-Scans durchführen oder planen zu können, benötigen Sie eine sichere Internetverbindung. Wenn Sie jedoch CIS-Scans auf privaten Instances ausführen möchten, müssen Sie einen VPC-Endpunkt verwenden.

Themen

- Überlegungen zur Verwaltung von Amazon Inspector CIS-Scans mit AWS Organizations
- Amazon Inspector-eigene Amazon S3 S3-Buckets, die für Amazon Inspector CIS-Scans verwendet werden
- Eine CIS-Scankonfiguration erstellen
- CIS-Scanergebnisse anzeigen
- Bearbeitung einer CIS-Scankonfiguration
- CIS-Scanergebnisse herunterladen

EC2 Amazon-Instance-Anforderungen für Amazon Inspector CIS-Scans

Um einen CIS-Scan auf Ihrer EC2 Amazon-Instance auszuführen, muss die EC2 Amazon-Instance die folgenden Kriterien erfüllen:

- Das Instance-Betriebssystem ist eines der unterstützten Betriebssysteme für CIS-Scans. Weitere Informationen finden Sie unter Von Amazon Inspector unterstützte Betriebssysteme und Programmiersprachen.
- Die Instance ist eine Amazon EC2 Systems Manager Manager-Instance. Weitere Informationen finden Sie unter Arbeiten mit dem SSM-Agenten im AWS Systems Manager Benutzerhandbuch.
- Das Amazon Inspector SSM-Plugin ist auf der Instance installiert. Amazon Inspector installiert dieses Plugin automatisch auf verwalteten Instances.
- Die Instance verfügt über ein Instance-Profil, das SSM Berechtigungen zur Verwaltung der Instance und Amazon Inspector zur Ausführung von CIS-Scans für diese Instance gewährt. Um diese Berechtigungen zu gewähren, fügen Sie die ManagedCisPolicy Richtlinien Amazon SSMManaged InstanceCore und AmazonInspector2 einer IAM-Rolle hinzu. Fügen Sie Ihrer Instance dann die IAM-Rolle als Instance-Profil hinzu. Anweisungen zum Erstellen und Anhängen eines Instance-Profils finden Sie unter Arbeiten mit IAM-Rollen im EC2 Amazon-Benutzerhandbuch.

> **Note**
>
> Sie müssen Amazon Inspector Deep Inspection nicht aktivieren, bevor Sie einen CIS-Scan auf Ihrer EC2 Amazon-Instance ausführen. Wenn Sie Amazon Inspector Deep Inspection deaktivieren, installiert Amazon Inspector automatisch den SSM-Agenten, aber der SSM-

Agent wird nicht mehr aufgerufen, um Deep Inspection auszuführen. Infolgedessen ist die `InspectorLinuxDistributor-do-not-delete` Zuordnung jedoch in Ihrem Konto vorhanden.

Amazon Virtual Private Cloud Cloud-Endpunktanforderungen für die Ausführung von CIS-Scans auf privaten EC2 Amazon-Instances

Sie können CIS-Scans auf EC2 Amazon-Instances über ein Amazon-Netzwerk ausführen. Wenn Sie jedoch CIS-Scans auf privaten EC2 Amazon-Instances ausführen möchten, müssen Sie Amazon VPC-Endpoints erstellen. Die folgenden Endpunkte sind erforderlich, wenn Sie Amazon VPC-Endpoints für Systems Manager erstellen:

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

Weitere Informationen finden Sie unter Erstellen von Amazon VPC-Endpunkten für Systems Manager im AWS Systems Manager Benutzerhandbuch.

> **Note**
>
> Derzeit unterstützen einige den AWS-Regionen Endpunkt nicht. `amazonaws.com.region.inspector2`

CIS-Scans werden ausgeführt

Sie können einen CIS-Scan entweder einmal auf Anforderung oder als geplanten wiederkehrenden Scan ausführen. Um einen Scan auszuführen, müssen Sie zunächst eine Scankonfiguration erstellen.

Wenn Sie eine Scankonfiguration erstellen, geben Sie Tag-Schlüssel-Wert-Paare an, die für Ziel-Instances verwendet werden sollen. Wenn Sie der von Amazon Inspector delegierte Administrator für eine Organisation sind, können Sie in der Scan-Konfiguration mehrere Konten angeben, und Amazon

Inspector sucht in jedem dieser Konten nach Instances mit den angegebenen Tags. Sie wählen das CIS-Benchmark-Level für den Scan. Für jeden Benchmark unterstützt CIS ein Level-1- und Level-2-Profil, das als Ausgangsbasis für verschiedene Sicherheitsstufen dient, die in verschiedenen Umgebungen möglicherweise erforderlich sind.

- Stufe 1 — empfiehlt grundlegende Sicherheitseinstellungen, die auf jedem System konfiguriert werden können. Die Implementierung dieser Einstellungen sollte kaum oder gar nicht zu Betriebsunterbrechungen führen. Ziel dieser Empfehlungen ist es, die Anzahl der Eintrittspunkte in Ihre Systeme zu verringern und so Ihre allgemeinen Cybersicherheitsrisiken zu verringern.
- Stufe 2 — empfiehlt erweiterte Sicherheitseinstellungen für Hochsicherheitsumgebungen. Die Implementierung dieser Einstellungen erfordert Planung und Koordination, um das Risiko geschäftlicher Auswirkungen zu minimieren. Ziel dieser Empfehlungen ist es, Sie bei der Einhaltung gesetzlicher Vorschriften zu unterstützen.

Stufe 2 erweitert Ebene 1. Wenn Sie Level 2 wählen, sucht Amazon Inspector nach allen Konfigurationen, die für Level 1 und Level 2 empfohlen werden.

Nachdem Sie die Parameter für Ihren Scan definiert haben, können Sie wählen, ob Sie ihn als einmaligen Scan, der nach Abschluss der Konfiguration ausgeführt wird, oder als wiederkehrender Scan ausführen möchten. Wiederkehrende Scans können täglich, wöchentlich oder monatlich zu einem Zeitpunkt Ihrer Wahl ausgeführt werden.

> **Tip**
>
> Wir empfehlen, einen Tag und eine Uhrzeit zu wählen, die sich während der Ausführung des Scans am wenigsten auf Ihr System auswirken.

Überlegungen zur Verwaltung von Amazon Inspector CIS-Scans mit AWS Organizations

Wenn Sie CIS-Scans in einer Organisation ausführen, interagieren die von Amazon Inspector delegierten Administratoren und Mitgliedskonten unterschiedlich mit den CIS-Scankonfigurationen und Scanergebnissen.

So können delegierte Administratoren von Amazon Inspector mit CIS-Scankonfigurationen und Scanergebnissen interagieren

Wenn der delegierte Administrator eine Scan-Konfiguration erstellt, entweder für alle Konten oder für ein bestimmtes Mitgliedskonten, ist die Organisation für die Konfiguration verantwortlich. Scankonfigurationen, die einer Organisation gehören, haben einen ARN, in dem die Organisations-ID als Eigentümer angegeben ist:

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

Der delegierte Administrator kann Scankonfigurationen verwalten, die einer Organisation gehören, auch wenn sie von einem anderen Konto erstellt wurden.

Der delegierte Administrator kann die Scanergebnisse für jedes Konto in seiner Organisation einsehen.

Wenn der delegierte Administrator eine Scankonfiguration erstellt und `SELF` als Zielkonto angibt, ist der delegierte Administrator für die Scankonfiguration verantwortlich, auch wenn er das Unternehmen verlässt. Der delegierte Administrator kann das Ziel einer Scankonfiguration mit `SELF` dem Ziel jedoch nicht ändern.

> **Note**
>
> Der delegierte Administrator kann den CIS-Scankonfigurationen, die das Unternehmen besitzt, keine Tags hinzufügen.

So können Amazon Inspector Inspector-Mitgliedskonten mit CIS-Scankonfigurationen und Scanergebnissen interagieren

Wenn ein Mitgliedskonto eine CIS-Scan-Konfiguration erstellt, ist es für die Konfiguration verantwortlich. Der delegierte Administrator kann die Konfiguration jedoch einsehen. Wenn ein Mitgliedskonto die Organisation verlässt, kann der delegierte Administrator die Konfiguration nicht einsehen.

> **Note**
>
> Der delegierte Administrator kann eine vom Mitgliedskonto erstellte Scan-Konfiguration nicht bearbeiten.

Mitgliedskonten, delegierte Administratoren SELF als Ziel und eigenständige Konten erstellen alle ihre eigenen Scankonfigurationen. Diese Scankonfigurationen haben einen ARN, der die Konto-ID als Besitzer anzeigt:

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

Ein Mitgliedskonto kann die Scanergebnisse in seinem Konto einsehen, einschließlich der Scanergebnisse von CIS-Scans, die der delegierte Administrator geplant hat.

Amazon Inspector-eigene Amazon S3 S3-Buckets, die für Amazon Inspector CIS-Scans verwendet werden

Open Vulnerability and Assessment Language (OVAL) ist ein Projekt zur Informationssicherheit, das standardisiert, wie der Maschinenzustand von Computersystemen bewertet und gemeldet wird. In der folgenden Tabelle sind alle Amazon S3-Buckets mit OVAL-Definitionen aufgeführt, die Amazon Inspector gehören und für CIS-Scans verwendet werden. Amazon Inspector stellt OVAL-Definitionsdateien bereit, die für CIS-Scans erforderlich sind. Die Amazon S3-Buckets, die Amazon Inspector gehören, sollten bei VPCs Bedarf in die Zulassungsliste aufgenommen werden.

> **Note**
>
> Die Details für jeden der folgenden Amazon S3-Buckets, die Amazon Inspector gehören, können sich nicht ändern. Die Tabelle kann jedoch aktualisiert werden, um die neu unterstützten AWS-Regionen Funktionen widerzuspiegeln. Sie können Amazon S3-Buckets, die Amazon Inspector gehören, nicht für andere Amazon S3 S3-Operationen oder in Ihren eigenen Amazon S3 S3-Buckets verwenden.

CIS-Bucket	AWS-Region
`cis-datasets-prod-arn-5908f6f`	Europe (Stockholm)
`cis-datasets-prod-bah-8f88801`	Middle East (Bahrain)
`cis-datasets-prod-bjs-0f40506`	China (Peking)
`cis-datasets-prod-bom-435a167`	Asien-Pazifik (Mumbai)

CIS-Bucket	AWS-Region
cis-datasets-prod-cdg-f3a9c58	Europa (Paris)
cis-datasets-prod-cgk-09eb12f	Asien-Pazifik (Jakarta)
cis-datasets-prod-cmh-63030b9	USA Ost (Ohio)
cis-datasets-prod-cpt-02c5c6f	Afrika (Kapstadt)
cis-datasets-prod-dub-984936f	Europa (Irland)
cis-datasets-prod-fra-6eb96eb	Europa (Frankfurt)
cis-datasets-prod-gru-de69f99	Südamerika (São Paulo)
cis-datasets-prod-hkg-8e30800	Asien-Pazifik (Hongkong)
cis-datasets-prod-iad-8438411	USA Ost (Nord-Virginia)
cis-datasets-prod-icn-f4eff1c	Asien-Pazifik (Seoul)
cis-datasets-prod-kix-5743b21	Asien-Pazifik (Osaka)
cis-datasets-prod-lhr-8b1fbd0	Europa (London)
cis-datasets-prod-mxp-7b1bbce	Europa (Milan)
cis-datasets-prod-nrt-464f684	Asien-Pazifik (Tokio)
cis-datasets-prod-osu-5bead6f	AWS GovCloud (US-Ost)
cis-datasets-prod-pdt-adadf9c	AWS GovCloud (US-West)
cis-datasets-prod-pdx-acfb052	USA West (Oregon)
cis-datasets-prod-sfo-1515ba8	USA West (Nordkalifornien)
cis-datasets-prod-sin-309725b	Asien-Pazifik (Singapur)
cis-datasets-prod-syd-f349107	Asien-Pazifik (Sydney)

CIS-Bucket	AWS-Region
`cis-datasets-prod-yul-5e0c95e`	Kanada (Zentral)
`cis-datasets-prod-zhy-5a8eacb`	China (Ningxia)
`cis-datasets-prod-zrh-67e0e3d`	Europa (Zürich)

Eine CIS-Scankonfiguration erstellen

In diesem Thema wird beschrieben, wie eine CIS-Scankonfiguration erstellt wird.

Um einen CIS-Scan auszuführen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie in der AWS-Region Dropdownliste den Ort aus, AWS-Region an dem Sie einen CIS-Scan ausführen möchten.
3. Wählen Sie im Navigationsbereich die Option On-Demand-Scans und dann CIS-Scans aus.
4. Wählen Sie Neuen Scan erstellen aus.
5. Geben Sie unter Name der Scan-Konfiguration einen Namen für die Scan-Konfiguration ein.
6. Geben Sie für Target Resource Tags einen Schlüssel und den entsprechenden Wert für die Instances ein, die Sie scannen möchten. Sie können bis zu fünf verschiedene Werte für jeden Schlüssel und insgesamt 25 Tags angeben, die in den Scan aufgenommen werden sollen.
7. Für das CIS-Benchmark-Level können Sie Level 1 für grundlegende Sicherheitskonfigurationen oder Level 2 für erweiterte Sicherheitskonfigurationen wählen.
8. Geben Sie für Zielkonten an, welche Konten in den CIS-Scan aufgenommen werden sollen. Weitere Informationen finden Sie unter Überlegungen zur Verwaltung von Amazon Inspector CIS-Scans mit AWS Organizations.

   Wenn es sich bei Ihrem Konto um das delegierte Administratorkonto handelt, können Sie Alle Konten oder Konten angeben auswählen. Die Option Alle Konten zielt auf alle Konten in Ihrer Organisation ab. Die Option Konten angeben zielt nur auf einzelne Konten in Ihrer Organisation ab. Wenn Sie diese Option wählen, können Sie mehr als ein Konto angeben, indem Sie die Kontonummern durch ein Komma trennen. Sie können `SELF` anstelle einer Konto-ID auch eine Konto-ID eingeben, um eine Scan-Konfiguration für Ihr Konto zu erstellen

Wenn es sich bei Ihrem Konto um ein eigenständiges Konto oder ein Mitgliedskonto in einer Organisation handelt, können Sie Self auswählen, um eine Scankonfiguration für Ihr Konto zu erstellen.

9. Wählen Sie unter Zeitplan die Option Einmaliger Scan, der ausgeführt wird, sobald Sie die Erstellung Ihrer Scankonfiguration abgeschlossen haben, oder Wiederkehrende Scans, der zu dem von Ihnen angegebenen Zeitpunkt ausgeführt wird.
10. Bestätigen Sie Ihre Auswahl und wählen Sie dann Erstellen.

CIS-Scanergebnisse anzeigen

Amazon Inspector erstellt einen Scanauftrag für jede Scankonfiguration, die ausgeführt wird, und sammelt Ergebnisse eines Scans mit einer eindeutigen Scan-ID. Die CIS-Scanergebnisse sind 90 Tage lang verfügbar. Sie können die CIS-Scanergebnisse anhand ihrer Checks oder gescannten Ressourcen anzeigen:

- Nach Prüfungen aggregierte Scanergebnisse — Gruppiert die Ergebnisse eines Scans nach jeder einzelnen Prüfung, die während des Scans durchgeführt wurde. Für jede Prüfung erhalten Sie einen Bericht darüber, wie viele Ressourcen ausgefallen, übersprungen oder bestanden wurden.
- Nach gescannten Ressourcen aggregierte Suchergebnisse — Gruppiert die Ergebnisse eines Scans nach jeder gescannten Ressource, auf die der Scan während des Scans abzielt. Für jede Ressource erhalten Sie einen Bericht darüber, wie viele Prüfungen eine Ressource nicht bestanden, übersprungen oder bestanden hat.

In diesem Thema wird beschrieben, wie die Ergebnisse eines CIS-Scans angezeigt werden.

So zeigen Sie Scanergebnisse an

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie in der AWS-Region Dropdownliste den Ort aus, an AWS-Region dem Sie Ihre CIS-Scan-Konfiguration erstellt haben.
3. Wählen Sie im Navigationsbereich die Option On-Demand-Scans und dann CIS-Scans aus.
4. Wählen Sie die Registerkarte Scanergebnisse aus.

5. Wählen Sie in der Spalte Geplant nach die ID des Scanzeitplans aus, den Sie anzeigen möchten. Oder wählen Sie die Zeile mit der Scanzeitplan-ID aus, die Sie anzeigen möchten, und wählen Sie dann Details anzeigen.
6. Wählen Sie „Prüfungen“, um alle durchgeführten Prüfungen anzuzeigen, oder „Gescannte Ressourcen“, um alle gescannten Ressourcen anzuzeigen, die während des Scans als Ziel ausgewählt wurden.

Sie können auch Details für geplante CIS-Scans anzeigen.

Um Details für geplante CIS-Scans anzuzeigen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie in der AWS-Region Dropdownliste den Ort aus, an AWS-Region dem Sie Ihre CIS-Scan-Konfiguration erstellt haben.
3. Wählen Sie im Navigationsbereich die Option On-Demand-Scans und dann CIS-Scans aus.
4. Wählen Sie die Registerkarte Geplant.
5. Wählen Sie in der Spalte Name der Scankonfiguration den Namen der Scankonfiguration aus, die Sie anzeigen möchten. Oder wählen Sie die Zeile mit der Scankonfiguration aus, die Sie anzeigen möchten, und wählen Sie dann Details anzeigen.

Bearbeitung einer CIS-Scankonfiguration

In diesem Thema wird beschrieben, wie eine CIS-Scankonfiguration bearbeitet wird.

Um eine CIS-Scankonfiguration zu bearbeiten

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie in der AWS-Region Dropdownliste den Ort aus, an AWS-Region dem Sie Ihre CIS-Scan-Konfiguration erstellt haben.
3. Wählen Sie im Navigationsbereich die Option On-Demand-Scans und dann CIS-Scans aus.
4. Wählen Sie die Registerkarte Geplant.
5. Wählen Sie die Zeile mit der Scankonfiguration aus, die Sie bearbeiten möchten, und klicken Sie dann auf Bearbeiten.

CIS-Scanergebnisse herunterladen

Sie können eine PDF- oder CSV-Datei eines CIS-Scans mithilfe der Amazon Inspector Inspector-Konsole oder API herunterladen.

> **Note**
>
> Sie können nur eine CSV-Datei mit Ihren CIS-Scanergebnissen für CIS-Scans herunterladen, die nach dem 05.03.2024 erfasst wurden.

In diesem Thema wird beschrieben, wie Sie einen CIS-Scan mithilfe der Amazon Inspector Inspector-Konsole herunterladen.

Um CIS-Scanergebnisse von der Konsole herunterzuladen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie in der AWS-Region Dropdownliste den Ort aus, an AWS-Region dem Sie Ihre CIS-Scan-Konfiguration erstellt haben.
3. Wählen Sie im Navigationsbereich die Option On-Demand-Scans und dann CIS-Scans aus.
4. Wählen Sie die Registerkarte Scanergebnisse aus.
5. Wählen Sie in der Spalte Geplant von die ID des Scanzeitplans aus, die Sie anzeigen möchten. Oder wählen Sie die Zeile mit der Scanzeitplan-ID aus, die Sie anzeigen möchten, und wählen Sie dann Details anzeigen.
6. Wählen Sie „Herunterladen“ und dann „PDF“ oder „CSV“. Wenn es sich bei Ihrem Konto um das delegierte Administratorkonto handelt, können Sie Konto auswählen wählen, um Ergebnisse für ein bestimmtes Mitgliedskonto herunterzuladen.

Amazon Inspector Code-Sicherheit

Amazon Inspector ist ein Schwachstellen-Management-Service, der Workloads automatisch erkennt und sie kontinuierlich auf Software-Schwachstellen und unbeabsichtigte Netzwerkbedrohungen überprüft. Mit Code Security scannt Amazon Inspector den Quellcode von Erstanbieteranwendungen, Abhängigkeiten von Drittanbieteranwendungen und Infrastructure as Code auf Sicherheitslücken. Sie können Code Security in der Amazon Inspector Inspector-Konsole oder mit der Amazon Inspector Inspector-API aktivieren. Sobald Sie Code Security aktiviert haben, können Sie eine Scan-Konfiguration erstellen und auf Ihr Code-Repository anwenden, um festzulegen, wie oft und wann der Code gescannt wird. Sie können Ihre Scankonfiguration jederzeit anzeigen, bearbeiten und löschen. Informationen darüber, AWS-Regionen wo Code Security verfügbar ist, finden Sie unter Regionen und Endpunkte. Preisinformationen finden Sie unter Amazon Inspector — Preise.

> **Note**
>
> Von Amazon Inspector Code Security generierte Sicherheitsergebnisse sind für die Amazon Inspector Inspector-Integration mit Security Hub nicht verfügbar. Sie können jedoch in der Amazon Inspector-Konsole und über die Amazon Inspector-API auf diese speziellen Ergebnisse zugreifen.

Voraussetzungen für die Codesicherheit

Bevor Sie Code Security verwenden können, müssen Sie Code Security aktivieren und entscheiden, wie Sie Ihre Daten verschlüsseln möchten. Dies können Informationen wie Anmeldeinformationen für die Integration, Code oder andere Informationen sein, die sich auf Ihre Integrationen, Code-Repositorys und Projekte beziehen. Standardmäßig werden Ihre Daten mit einem AWS eigenen Schlüssel verschlüsselt. Das bedeutet, dass der Schlüssel vom Dienst erstellt wird, ihm gehört und von ihm verwaltet wird. Wenn Sie den Schlüssel, der zur Verschlüsselung Ihrer Daten verwendet wird, besitzen und verwalten möchten, können Sie einen vom Kunden verwalteten KMS-Schlüssel erstellen.

Code-Sicherheit aktivieren

Sie aktivieren Code Security auf die gleiche Weise wie alle automatisierten Scantypen. Weitere Informationen finden Sie unter Einen Suchtyp aktivieren.

Einen vom Kunden verwalteten Zugriffsschlüssel erstellen AWS KMS

Standardmäßig werden Ihre Daten mit einem AWS eigenen Schlüssel verschlüsselt. Das bedeutet, dass der Schlüssel vom Dienst erstellt wird, ihm gehört und von ihm verwaltet wird. Wenn Sie den Schlüssel, der zur Verschlüsselung Ihrer Daten verwendet wird, besitzen und verwalten möchten, können Sie einen vom Kunden verwalteten KMS-Schlüssel erstellen. Amazon Inspector interagiert nicht mit Ihren Daten. Amazon Inspector nimmt nur Metadaten aus Repositorys in Ihrem Quellcode-Anbieter auf. Informationen zum Erstellen eines vom Kunden verwalteten KMS-Schlüssels finden Sie unter Erstellen eines KMS-Schlüssels im AWS Key Management Service Benutzerhandbuch.

Beispiel für eine Richtlinie

Verwenden Sie beim Erstellen Ihres vom Kunden verwalteten Schlüssels die folgende Beispielrichtlinie.

```
{
    "Version": "2012-10-17",
    "Id": "key-policy",
    "Statement": [
        {
            "Sid": "Allow Q to use Encrypt Decrypt GenerateDataKey and
 GenerateDataKeyWithoutPlaintext",
            "Effect": "Allow",
            "Principal": {
                "Service": "q.amazonaws.com"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:qdeveloper:codesecurity-scope":
 "111122223333"
                },
                "ArnLike": {
```

```
                    "aws:SourceArn":
"arn:aws:inspector2:Region:111122223333:codesecurity-integration/*"
                }
            }
        },
        {
            "Sid": "Allow Q to use DescribeKey",
            "Effect": "Allow",
            "Principal": {
                "Service": "q.amazonaws.com"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*"
        },
        {
            "Sid": "Allow Inspector to use Encrypt Decrypt GenerateDataKey and
GenerateDataKeyWithoutPlaintext using FAS",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{111122223333}:role/inspectorCodeSecurity"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:GenerateDataKey",
                "kms:GenerateDataKeyWithoutPlaintext"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "inspector2.Region.amazonaws.com"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:qdeveloper:codesecurity-scope":
"111122223333"
                }
            }
        },
        {
            "Sid": "Allow Inspector to use DescribeKey using FAS",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{111122223333}:role/inspectorCodeSecurity"
            },
```

```
            "Action": [
                "kms:DescribeKey"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "inspector2.Region.amazonaws.com"
                }
            }
        }
    ]
}
```

Nachdem Sie Ihren KMS-Schlüssel erstellt haben, können Sie den folgenden Amazon Inspector verwenden APIs.

- UpdateEncryptionKey — Verwenden Sie mit `CODE_REPOSITORY` for `resourceType` und `CODE` als Scan-Typ, um die Verwendung Ihres vom Kunden verwalteten KMS-Schlüssels zu konfigurieren.
- GetEncryptionKey — Verwenden Sie mit `CODE_REPOSITORY` for `resourceType` und `CODE` als Suchtyp, um den Abruf Ihrer KMS-Schlüsselkonfiguration zu konfigurieren.
- ResetEncryptionKey — Verwenden Sie mit `CODE_REPOSITORY` for `resourceType` und`CODE`, um Ihre KMS-Schlüsselkonfiguration zurückzusetzen und einen AWS eigenen KMS-Schlüssel zu verwenden.

Eine Integration zwischen Amazon Inspector und Ihrem Code-Repository erstellen

Dieser Abschnitt enthält Themen, in denen beschrieben wird, wie Sie eine Integration zwischen Amazon Inspector und Ihrem Code-Repository erstellen. Wenn Sie eine Integration erstellen, werden alle Code-Repositorys in der Amazon Inspector Inspector-Konsole auf der Seite Code Security als Projekte aufgeführt. In anderen Themen in diesem Abschnitt wird beschrieben, wie Sie auf Ihre Integrationen und Projekte zugreifen können.

Code Security importiert nur bis zu 100.000 Projekte, und nur der Standardzweig für jedes Repository wird überwacht. Ein Projekt kann maximal drei Standard-Scankonfigurationen zugeordnet werden.

Code Security unterstützt nur maximal 100 Integrationen pro Konto. Bei den Integrationen von Code Security gibt es kein Konzept für die Beziehung zwischen delegierten account/member Administratorkonten.

Um Einschränkungen zu vermeiden, empfehlen wir, denselben Host nicht mehrmals für eine Integration zu verwenden.

Integrationen mit GitHub SaaSGitHub Enterprise Cloud, und GitHub Enterprise Server erfordern einen öffentlichen Internetzugang.

> **Important**
>
> Integrationen von Drittanbietern können ohne vorherige Ankündigung aus beliebigem Grund vorübergehend oder dauerhaft deaktiviert werden, z. B. um Sicherheitsbedenken auszuräumen.

Erstellen einer Integration zwischen Amazon Inspector und GitHub

In diesem Thema wird beschrieben, wie Sie eine Integration zwischen Amazon Inspector und erstellenGitHub.

> **Note**
>
> Wenn Sie zum ersten Mal eine Integration erstellen, werden Sie in Schritt 2 aufgefordert, eine Standard-Scan-Konfiguration zu erstellen. Wenn Sie eine Scankonfiguration erstellen, wählen Sie die Scanfrequenz, die Scananalyse und die zu scannenden Repositorys aus. Das Erstellen einer Standard-Scankonfiguration entspricht dem Erstellen einer allgemeinen Scankonfiguration. Die Standard-Scankonfiguration wird jedoch automatisch allen neuen und vorhandenen Projekten zugeordnet, die in Amazon Inspector importiert werden. Wenn Sie eine Standard-Scan-Konfiguration erstellen möchten, wählen Sie Mit dieser Konfiguration fortfahren. Sie können eine Standard-Scan-Konfiguration nur einmal erstellen. Wenn Sie eine Standard-Scankonfiguration erstellen, werden Sie nicht erneut aufgefordert, eine Standard-Scankonfiguration zu erstellen. Sie können eine Standard-Scankonfiguration nur einmal pro Konto und einmal pro Organisation erstellen. Wenn Sie keine Standard-Scan-Konfiguration konfigurieren möchten, wählen Sie Konfiguration überspringen. Sie werden jedoch aufgefordert, eine Standard-Scankonfiguration zu erstellen, wenn Sie das nächste Mal eine Integration erstellen. Nachdem Sie eine Standard-Scan-Konfiguration erstellt oder die

Erstellung einer Standard-Scan-Konfiguration übersprungen haben, werden Sie zu Schritt 3 des Integrations-Workflows weitergeleitet, wo Sie Ihre Integrationsdetails eingeben.

Integrationen mit GitHub SaaSGitHub Enterprise Cloud, und GitHub Enterprise Server erfordern einen öffentlichen Internetzugang.

> **Note**
>
> Amazon Inspector scannt und überwacht nur Ihre Standardfiliale. Wenn Sie einen neuen Standardzweig erstellen, scannt und aktualisiert Amazon Inspector den neuen Standardzweig.

> **Important**
>
> Bevor Sie mit der Erstellung der Integration fertig sind, werden Sie angewiesen, die Verbindung zwischen Amazon Inspector und GitHub zu autorisieren. Sie müssen diesen Schritt abschließen, um den Vorgang abzuschließen. Wenn Sie das Pop-up schließen, können Sie nicht fortfahren.

Um eine Integration zwischen Amazon Inspector und GitHub

1. Melden Sie sich mit Ihren Zugangsdaten an. Öffnen Sie die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus. Wählen Sie Connect und wählen SieGitHub.
3. Geben Sie unter Integrationsdetails den Namen Ihrer Integration ein und wählen Sie Connect aus GitHub.
4. Wählen Sie im Pop-up Autorisieren, um eine Verbindung zwischen Amazon Inspector und GitHub herzustellen.
5. Wählen Sie im Erfolgsbanner die Option Gehe zur Seite zur GitHub Verbindungserstellung.
6. Geben Sie die Installations-ID für die GitHub Anwendung ein. Wenn Sie die GitHub Anwendung installiert haben, finden Sie die Installations-ID auf GitHub der GitHub Apps-Seite oder am Ende der GitHub Anwendungs-URL. Wenn Sie die GitHub Anwendung nicht installiert haben, wählen

Sie Neue App installieren. Dadurch werden Sie zu GitHub dem Punkt weitergeleitet, an dem Sie die GitHub Organisation auswählen und den Repository-Bereich angeben.

7. Wählen Sie Connect GitHub.

Nachdem Sie die Integration erstellt haben, kann ein Szenario auftreten, in dem Amazon Inspector das Zugriffstoken nicht aktualisieren kann. Dies kann der Fall sein, wenn der Integrationshost nicht verfügbar ist oder Amazon Inspector andere Kommunikationsprobleme hat. Um das Problem zu beheben, können Sie die Verbindung über den Tab Integrationen auf der Seite Codesicherheit erneut authentifizieren. In der Spalte Status wird die Integration als Inaktiv angezeigt, und Amazon Inspector bietet die Option zur erneuten Authentifizierung. Wählen Sie Erneut authentifizieren. Sie werden zum Integrations-Workflow weitergeleitet, wo Sie die Verbindungseinrichtung abschließen können.

Wenn Sie die Systemeinstellungen für Ihre Integration löschen, kann die Verbindung auf unbestimmte Zeit unterbrochen werden. In diesem Fall müssen Sie die Integration löschen und eine neue Integration erstellen. Wenn Sie eine Integration löschen, verlieren Sie alle Projekte und Scankonfigurationen, die mit der Integration verknüpft sind.

Erstellen einer Integration zwischen Amazon Inspector und GitLab Self Managed

In diesem Thema wird beschrieben, wie Sie eine Integration zwischen Amazon Inspector und Ihrem Code-Repository in erstellenGitLab Self Managed.

Erforderliche Informationen

Folgendes ist erforderlich, wenn Sie eine Verbindung herstellen:

- Integrationsname — Dies ist der Name, der dem Hauptteil Ihrer Integration hinzugefügt wurde.
- Endpunkt-URL — Dies ist die URL, die für den Zugriff auf Ihre GitLab Self Managed Instanz verwendet wird.
- Persönliches Zugriffstoken — Das persönliche Zugriffstoken wird GitLab Self Managed von einem Administratorkonto aus erstellt und muss die folgenden Bereiche enthalten:`api`, `read_apiread_repository`, und`write_repository`.

> **Note**
>
> Amazon Inspector scannt und überwacht nur Ihre Standardfiliale. Wenn Sie einen neuen Standardzweig erstellen, scannt und aktualisiert Amazon Inspector den neuen Standardzweig.

Erstellen einer Integration zwischen Amazon Inspector und GitLab Self Managed

Das folgende Verfahren beschreibt, wie Sie eine Verbindung zwischen Amazon Inspector und Ihrem Code-Repository in herstellenGitLab Self Managed.

> **Note**
>
> Wenn Sie zum ersten Mal eine Integration erstellen, werden Sie in Schritt 2 aufgefordert, eine Standard-Scan-Konfiguration zu erstellen. Wenn Sie eine Scankonfiguration erstellen, wählen Sie die Scanfrequenz, die Scananalyse und die zu scannenden Repositorys aus. Das Erstellen einer Standard-Scankonfiguration entspricht dem Erstellen einer allgemeinen Scankonfiguration. Die Standard-Scankonfiguration wird jedoch automatisch allen neuen und vorhandenen Projekten zugeordnet, die in Amazon Inspector importiert werden. Wenn Sie eine Standard-Scan-Konfiguration erstellen möchten, wählen Sie Mit dieser Konfiguration fortfahren. Sie können eine Standard-Scan-Konfiguration nur einmal erstellen. Wenn Sie eine Standard-Scankonfiguration erstellen, werden Sie nicht erneut aufgefordert, eine Standard-Scankonfiguration zu erstellen. Sie können eine Standard-Scankonfiguration nur einmal pro Konto und einmal pro Organisation erstellen. Wenn Sie keine Standard-Scan-Konfiguration konfigurieren möchten, wählen Sie Konfiguration überspringen. Wenn Sie das nächste Mal eine Integration erstellen, werden Sie jedoch aufgefordert, eine Standard-Scankonfiguration zu erstellen. Nachdem Sie eine Standard-Scan-Konfiguration erstellt oder die Erstellung einer Standard-Scan-Konfiguration übersprungen haben, werden Sie zu Schritt 3 des Integrations-Workflows weitergeleitet, wo Sie Ihre Integrationsdetails eingeben.

> **Important**
>
> Bevor Sie mit der Erstellung der Integration fertig sind, werden Sie aufgefordert, die Verbindung zwischen Amazon Inspector und GitLab Self Managed zu autorisieren. Sie

müssen diesen Schritt abschließen, um den Vorgang abzuschließen. Wenn Sie das Pop-up schließen, können Sie nicht fortfahren.

Um eine Verbindung mit GitLab Self Managed herzustellen

1. Melden Sie sich mit Ihren Zugangsdaten an. Öffnen Sie die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus. Wählen Sie Connect und dann GitLab Self Managed aus.
3. Geben Sie unter Integrationsdetails Folgendes ein:
    a. Geben Sie unter Integrationsname den Namen ein, der dem Hauptteil Ihrer Integration hinzugefügt wurde.
    b. Geben Sie unter Endpunkt-URL die URL ein, die für den Zugriff auf Ihre GitLab selbstverwaltete Instanz verwendet wird.
    c. Geben Sie unter Persönliches Zugriffstoken Ihr persönliches Zugriffstoken mit den erforderlichen Gültigkeitsbereichen ein.
4. Wählen Sie Verbinden mit. GitLab
5. Wählen Sie im Popup-Fenster die Option Autorisieren, um die Herstellung einer Verbindung zwischen Amazon Inspector und GitLab abzuschließen.

Nachdem Sie die Integration erstellt haben, kann ein Szenario auftreten, in dem Amazon Inspector das Zugriffstoken nicht aktualisieren kann. Dies kann der Fall sein, wenn der Integrationshost nicht verfügbar ist oder Amazon Inspector andere Kommunikationsprobleme hat. Um das Problem zu beheben, können Sie die Verbindung über den Tab Integrationen auf der Seite Code Security erneut authentifizieren. In der Spalte Status wird die Integration als Inaktiv angezeigt, und Amazon Inspector bietet die Option zur erneuten Authentifizierung. Wählen Sie Erneut authentifizieren. Sie werden zum Integrations-Workflow weitergeleitet, wo Sie die Verbindungseinrichtung abschließen können.

Wenn Sie die Systemeinstellungen für Ihre Integration löschen, kann die Verbindung auf unbestimmte Zeit unterbrochen werden. In diesem Fall müssen Sie die Integration löschen und eine neue Integration erstellen. Wenn Sie eine Integration löschen, verlieren Sie alle Projekte und Scankonfigurationen, die mit der Integration verknüpft sind.

Integrationen mit Code-Repositorys anzeigen

In diesem Thema wird beschrieben, wie Integrationen in der Amazon Inspector Inspector-Konsole angezeigt werden.

So zeigen Sie Integrationen in der Amazon Inspector Inspector-Konsole an

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wähle Sie Integrations (Integrationen) aus. Auf dieser Registerkarte können Sie alle Ihre konfigurierten Integrationen und grundlegende Informationen zu all Ihren Integrationen überprüfen. Zu diesen Informationen gehören der Name der Integration, der Status der Integration und der Name des Quellcode-Anbieters.

Authentifizieren Sie sich erneut beim Anbieter

Nachdem Sie die Integration erstellt haben, kann ein Szenario auftreten, in dem Amazon Inspector das Zugriffstoken nicht aktualisieren kann. Dies kann der Fall sein, wenn der Integrationshost nicht verfügbar ist oder Amazon Inspector andere Kommunikationsprobleme hat. Um das Problem zu beheben, können Sie die Verbindung über den Tab Integrationen auf der Seite Codesicherheit erneut authentifizieren. In der Spalte Status wird die Integration als Inaktiv angezeigt, und Amazon Inspector bietet die Option zur erneuten Authentifizierung. Wählen Sie Erneut authentifizieren. Sie werden zum Integrations-Workflow weitergeleitet, wo Sie die Verbindungseinrichtung abschließen können.

Wenn Sie die Systemeinstellungen für Ihre Integration löschen, kann die Verbindung auf unbestimmte Zeit unterbrochen werden. In diesem Fall müssen Sie die Integration löschen und eine neue Integration erstellen. Wenn Sie eine Integration löschen, verlieren Sie alle Projekte und Scankonfigurationen, die mit der Integration verknüpft sind.

Code-Repositorys anzeigen

In diesem Thema wird beschrieben, wie Code-Repositorys in der Amazon Inspector Inspector-Konsole angezeigt werden.

Um Code-Repositorys in der Amazon Inspector Inspector-Konsole anzuzeigen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.

2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wählen Sie Code-Repositorys aus. Auf dieser Registerkarte können Sie alle Ihre Code-Repositorys, die als Projekte aufgeführt sind, sowie grundlegende Informationen zu ihnen überprüfen. Zu diesen Informationen gehören der Name und der Scanstatus für jedes Projekt. Sie können auch die Konfigurationen überprüfen, die Ihren Projekten zugeordnet sind und wann Ihre Projekte zuletzt gescannt wurden. Sie können Ihre Projekte sogar in der Suchleiste filtern.

Details für ein Projekt anzeigen

In diesem Thema wird beschrieben, wie Sie Details für ein Projekt in der Amazon Inspector Inspector-Konsole anzeigen. Wenn Ihr Konto der delegierte Administrator für eine Organisation ist, können Sie Details zu Projekten anzeigen, die zu Mitgliedskonten gehören.

So zeigen Sie Codeprojekte in der Amazon Inspector Inspector-Konsole an

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wählen Sie Code-Repositorys aus. Auf dieser Registerkarte können Sie alle Ihre Code-Repositorys, die als Projekte aufgeführt sind, sowie grundlegende Informationen zu ihnen überprüfen. Zu diesen Informationen gehören der Name und der Scanstatus für jedes Projekt. Sie können auch die mit Ihren Projekten verknüpften Konfigurationen und den Zeitpunkt überprüfen, an dem Ihre Projekte zuletzt gescannt wurden. Sie können Ihre Projekte sogar in der Suchleiste filtern.
4. Wählen Sie ein Projekt aus. Oder wählen Sie ein Projekt aus und wählen Sie Details anzeigen. Auf dem Bildschirm mit den Projektdetails können Sie grundlegende Informationen über das Projekt überprüfen. Zu diesen Informationen gehören der Name und die ID für das Projekt sowie der Integrations-ARN. Sie enthalten Informationen darüber, wann das Projekt gescannt wurde und um welchen Anbietertyp es sich handelt. Sie können sogar die mit dem Projekt verknüpften Ergebnisse überprüfen, Ergebnisse exportieren und Regeln zur Unterdrückung von Ergebnissen erstellen.

Eine Integration löschen

Das folgende Verfahren beschreibt, wie Sie eine Integration in der Amazon Inspector Inspector-Konsole löschen. Wenn Sie eine Integration löschen, verlieren Sie alle Projekte und Scankonfigurationen, die mit der Integration verknüpft sind.

Um eine Integration in der Amazon Inspector Inspector-Konsole zu löschen.

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wähle Sie Integrations (Integrationen) aus. Auf dieser Registerkarte können Sie alle Ihre konfigurierten Integrationen und grundlegende Informationen zu all Ihren Integrationen überprüfen. Zu diesen Informationen gehören der Name der Integration, der Status der Integration und der Typ des Integrationsanbieters.
4. Wählen Sie eine Integration aus und klicken Sie auf Löschen.

Eine Scan-Konfiguration erstellen

Bevor Sie eine Scankonfiguration erstellen, müssen Sie eine Integration mit Amazon Inspector erstellen. Wenn Sie zum ersten Mal eine Integration erstellen, werden Sie aufgefordert, eine Standard-Scan-Konfiguration zu erstellen. In diesem Thema wird beschrieben, wie Sie eine allgemeine Scankonfiguration erstellen. Der Unterschied zwischen einer Standard-Scankonfiguration und einer allgemeinen Scankonfiguration besteht darin, dass neuen Projekten automatisch eine Standardscankonfiguration zugewiesen wird. Sie können die Erstellung einer Standard-Scankonfiguration überspringen.

Code Security unterstützt nur maximal 500 allgemeine Scankonfigurationen. Code Security unterstützt nur eine Standard-Scankonfiguration pro Konto und Organisation. Eine Scankonfiguration kann nur mit maximal 100.000 Projekten verknüpft werden.

Ein Projekt kann insgesamt maximal 4 Scankonfigurationen zugeordnet werden. Dies beinhaltet eine Standard-Scankonfiguration, falls eine Standard-Scankonfiguration erstellt wurde. Scankonfigurationen für eine Organisation können nicht markiert werden.

Wenn der delegierte Administrator für eine Organisation eine Scankonfiguration erstellt, wird die Scankonfiguration auf Organisationsebene erstellt und auf alle Mitgliedskonten in der Organisation

angewendet. Das Gleiche gilt, wenn der delegierte Administrator eine Standardscankonfiguration erstellt.

Wenn Sie eine Scankonfiguration erstellen, wählen Sie die Scanfrequenz, die Scananalyse und die zu scannenden Repositorys aus. Die Scanfrequenz kann je nach Bedarf und in regelmäßigen Abständen oder individuell angepasst werden. Bei änderungsbasiertem und regelmäßigem Scannen haben Sie die Möglichkeit, das regelmäßige Scannen zu aktivieren. Wenn Sie das regelmäßige Scannen aktivieren, legen Sie die Scanfrequenz auf den Wochentag oder den Monat fest, an dem ein Scan durchgeführt wird. Benutzerdefiniertes Scannen bietet Ihnen die Möglichkeit, das Scannen bei Codeänderungen und das regelmäßige Scannen zu aktivieren. Wenn Sie das Scannen bei Codeänderungen aktivieren, geben Sie den Scan-Trigger an, der in Merge- und Pull-Anfragen aufgenommen werden soll.

Scans können übersprungen werden, wenn sich eine Commit-ID innerhalb eines festgelegten Zeitraums nicht geändert hat. Bei regelmäßigen Scans werden Scans übersprungen, wenn sich eine Commit-ID zwischen den Scans innerhalb einer Woche nicht geändert hat. Bei On-Demand-Scans werden Scans übersprungen, wenn sich eine Commit-ID zwischen den Scans innerhalb von 24 Stunden nicht geändert hat.

> **Note**
>
> Wenn eine Scankonfiguration nur Trigger für Merge- und Pull-Requests enthält, sind nur die 25 wichtigsten Ergebnisse nur in der Quellcode-Management-Plattform sichtbar. Keine wird in Amazon Inspector sichtbar sein.

Um eine allgemeine Scan-Konfiguration zu erstellen

1. Melden Sie sich mit Ihren Zugangsdaten an. Öffnen Sie die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wählen Sie Konfigurationen und anschließend Scankonfiguration erstellen aus.
4. Gehen Sie unter Scandetails wie folgt vor:
   - Geben Sie unter Konfigurationsname einen Namen für die Scankonfiguration ein.
5. Geben Sie unter Scan-Frequenz an, wie oft Code gescannt wird, indem Sie auf Änderung basierendes und periodisches Scannen oder Benutzerdefinierte Scantypen und -auslöser auswählen.

a. (Option 1) Wenn Sie auf Änderungen basierendes und periodisches Scannen wählen, wählen Sie Periodische Suche aktivieren oder Periodische Suche deaktivieren aus.

   - . Wenn Sie Periodischen Scan aktivieren wählen, legen Sie die Scanfrequenz fest, indem Sie die Woche und den Tag auswählen, an dem der Code gescannt werden soll.

b. (Option 2) Wenn Sie Benutzerdefiniertes Scannen wählen, entscheiden Sie, ob das Scannen bei Codeänderungen und das regelmäßige Scannen aktiviert werden sollen.

   i. Wählen Sie Scannen bei Codeänderung aktivieren oder Scannen bei Codeänderung deaktivieren. Wenn Sie „Scannen bei Codeänderung aktivieren“ wählen, geben Sie in der Dropdownliste an, wann Scans ausgelöst werden sollen.

   ii. Wählen Sie Periodischen Scan aktivieren oder Periodischen Scan deaktivieren. Wenn Sie Periodischen Scan aktivieren wählen, legen Sie die Scan-Frequenz fest, indem Sie die Woche und den Tag auswählen, an dem der Code gescannt werden soll. Sie können auch nach ereignisbasierten Triggern scannen. Zu diesen Ereignissen gehört, wenn eine Pull-Anfrage für den Standard-Branch geöffnet ist und wenn ein Commit an den Standard-Branch weitergeleitet wird.

6. Entscheiden Sie unter Scan-Analyse, ob Sie eine vollständige Scan-Analyse oder eine benutzerdefinierte Scan-Analyse konfigurieren möchten:

   a. (Option 1) Wenn Sie Vollständige Scananalyse wählen, wenden Sie alle der folgenden Scananalysen an:

      - Statische Anwendungssicherheitstests — Analysiert den Quellcode auf Sicherheitslücken.
      - IaC-Scanning — Analysiert Skripts und Code, mit denen die Infrastruktur konfiguriert und bereitgestellt wird.
      - Statische Analyse der Softwarezusammensetzung — Untersucht Open-Source-Pakete in Anwendungen.

   b. (Option 2) Wenn Sie die Option Benutzerdefinierte Scananalyse wählen, müssen Sie mindestens einen Typ der zuvor genannten Scananalysetypen aus dem Drop-down-Menü auswählen:

7. (Optional) Erstellen Sie für Tags ein Schlüssel-Wert-Paar, das auf Ihr Projekt angewendet werden soll. Sie können bis zu 50 Tags erstellen.

8. Wählen Sie Weiter aus.

9. Wählen Sie unter Repository-Auswahl die Option Alle Repositorys oder Spezifische Repositorys aus.

    a. (Option 1) Wenn Sie Alle Repositorys wählen, ist das Scannen für alle vorhandenen Repositorys aktiviert.

    b. (Option 2) Wenn Sie Spezifische Repositorys wählen, ist das Scannen nur für die von Ihnen angegebenen Repositorys aktiviert.

10. Wählen Sie Weiter aus.
11. Überprüfen Sie Ihre Auswahl und wählen Sie dann Scankonfiguration erstellen aus.

> **Note**
>
> Allgemeine Scankonfigurationen werden nur auf alle vorhandenen Code-Repositorys angewendet. Sie werden nicht auf neue Code-Repositorys angewendet.

Scan-Konfigurationen anzeigen

Das folgende Verfahren beschreibt, wie Sie Scankonfigurationen in der Amazon Inspector Inspector-Konsole anzeigen.

> **Note**
>
> Wenn Sie sich Ihre Scan-Konfiguration auf Organisationsebene ansehen, können einige Details auf dem Code-Sicherheitsbildschirm je nach Ihren Anforderungen abweichen AWS-Konto.

Um Details zu einer Scan-Konfiguration anzuzeigen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wählen Sie Konfigurationen, um eine Liste Ihrer Scankonfigurationen anzuzeigen. Wenn Sie der delegierte Administrator sind, enthält die Liste die Scankonfigurationen Ihres Unternehmens. Sie können den Namen jeder Scan-Konfiguration sehen und sehen, wer die einzelnen Scan-

Konfigurationen erstellt hat (AWS-Konto ID oder Organisations-ID). Sie können auch sehen, welche Scantypen und welcher Scananalysetyp auf die Konfiguration angewendet werden. Sie können Ihre Scankonfiguration sogar nach verschiedenen Feldern in der Suchleiste filtern.

Details für eine Scan-Konfiguration anzeigen

Das folgende Verfahren beschreibt, wie Sie Details für eine Scan-Konfiguration in der Amazon Inspector Inspector-Konsole anzeigen.

Um Details zu einer Scan-Konfiguration anzuzeigen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wählen Sie Konfigurationen aus.
4. Wählen Sie die Konfiguration aus, für die Sie Details anzeigen möchten. Der Bildschirm mit den Details zur Scankonfiguration bietet einen Überblick über die Scankonfiguration. Auf diesem Bildschirm können Sie den ARN für die Scankonfiguration einsehen, welche Scanfrequenztypen aktiviert sind und welche Scananalysetypen aktiviert sind. Sie können die Scankonfiguration auch von diesem Bildschirm aus löschen. Wenn Sie sich eine Scankonfiguration ansehen, die zu Ihrer Organisation gehört, können Sie sie auch von diesem Bildschirm aus bearbeiten.

Bearbeitung einer Scan-Konfiguration

Sie können eine Scankonfiguration jederzeit bearbeiten. Wenn Sie eine Scankonfiguration bearbeiten, können Sie die Scanfrequenz, die Scananalyse, die Tags und die zu scannenden Repositorys ändern. Sie bearbeiten beispielsweise eine Scankonfiguration, um das Scannen für ein bestimmtes Repository anzuhalten. Das folgende Verfahren beschreibt, wie Sie eine Scankonfiguration bearbeiten.

Um eine Scankonfiguration zu bearbeiten

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Code Security aus.
3. Wählen Sie Konfigurationen aus.

4. Wählen Sie die Konfiguration aus, die Sie bearbeiten möchten, und klicken Sie dann auf Bearbeiten. Sie können auch die Konfiguration auswählen, die Sie bearbeiten möchten, und dann Bearbeiten wählen.

Löschen einer Scan-Konfiguration

Sie können eine Scankonfiguration jederzeit löschen. In diesem Thema wird beschrieben, wie eine Scankonfiguration gelöscht wird.

Um eine Scan-Konfiguration zu löschen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich die Option Codesicherheit aus.
3. Wählen Sie Konfigurationen aus.
4. Wählen Sie die Konfiguration aus, die Sie löschen möchten, und wählen Sie dann Löschen. Oder wählen Sie die Konfiguration aus, die Sie löschen möchten, und wählen Sie dann Löschen.

Einen Scan auf Anforderung durchführen

Sie können einen On-Demand-Service für Ihre Projekte durchführen. Wenn Sie einen Scan auf Anforderung durchführen, wird eine Vereinigung all Ihrer konfigurierten Scankonfigurationen auf Ihr ausgewähltes Projekt angewendet. Wenn Ihr Konto das delegierte Administratorkonto für eine Organisation ist, können Sie einen On-Demand-Scan für Projekte durchführen, die zu Mitgliedskonten gehören. Das folgende Verfahren beschreibt, wie Sie einen On-Demand-Scan in der Amazon Inspector Inspector-Konsole durchführen.

Um einen On-Demand-Scan durchzuführen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich die Option Code-Sicherheit aus.
3. Wählen Sie Code-Repositorys aus.
4. Wählen Sie das Projekt aus, das Sie scannen möchten, und wählen Sie dann Scannen auf Anforderung.

Unterstützte Sprachen für Amazon Inspector Inspector-Codesicherheit

Dieses Thema umfasst die unterstützten Sprachen für Amazon Inspector Code Security.

Unterstützte Sprachen für SAST

- C#(Alle Versionen außer .Net 6.0 und höher werden empfohlen)
- C(C11 oder früher)
- C++(C++17 oder früher)
- Go(nur Go 1.18)
- Java(Java17 oder früher)
- JavaScript(EMCMAScript 2021 oder früher)
- JSX(React 17 oder früher)
- Kotlin(Kotlin2.0 oder früher)
- PHP(PHP8.2 oder früher)
- Python(Python3.11 oder früher innerhalb der Serie Python 3)
- Ruby(nur Ruby 2.7 und 3.2)
- Rust
- Scala(Scala3.2.2 oder früher)
- Shell
- TSX
- TypeScript (alle Versionen)

Unterstützte Sprachen für die Analyse der Softwarekomposition

- Go(nur Go 1.18)
- Java(Java17 oder früher)
- JavaScript(EMCMAScript 2021 oder früher)
- PHP(PHP8.2 oder früher)
- Python(Python3.11 oder früher innerhalb der Serie Python 3)
- .Net

- Ruby(nur Ruby 2.7 und 3.2)
- Rust

Sprachen für Infrastruktur als Code

- AWS CDK (PythonundTypeScript)
- AWS CloudFormation (2010—09—09)
- Terraform(1.6.2 oder früher)

Deaktivierung der Codesicherheit

Weitere Informationen zur Deaktivierung von Code Security finden Sie unter Deaktivieren eines Scantyps.

Die Ergebnisse von Amazon Inspector verstehen

Amazon Inspector generiert ein Ergebnis, wenn es eine Sicherheitslücke in EC2 Amazon-Instances, Amazon ECR-Containern, Images und Lambda-Funktionen entdeckt. Es generiert auch Ergebnisse für Code-Schwachstellen, die im Quellcode von Erstanbieter-Anwendungen, in Abhängigkeiten von Drittanbieteranwendungen und in Infrastructure as Code entdeckt wurden. Ein Ergebnis ist ein detaillierter Bericht über eine Sicherheitslücke, die sich auf eine Ihrer AWS Ressourcen auswirkt.

Die Ergebnisse sind nach Sicherheitslücken benannt und enthalten Bewertungen des Schweregrads, Informationen zu betroffenen AWS Ressourcen und Ressourcen, die nicht zur AWS Verfügung stehen, sowie Einzelheiten zur Behebung erkannter Sicherheitslücken. Amazon Inspector speichert all Ihre aktiven Ergebnisse, bis Sie sie korrigieren.

Wenn eine Ressource gelöscht, beendet oder nicht mehr gescannt werden kann, schließt Amazon Inspector automatisch die mit der Ressource verknüpften Ergebnisse und löscht die Ergebnisse dann nach 3 Tagen. Wenn Ergebnisse aus einem anderen Grund geschlossen werden, werden sie nach 30 Tagen gelöscht.

> **Note**
>
> Amazon Inspector öffnet ein behobenes Ergebnis innerhalb von sieben Tagen nach Abschluss des Fundes erneut, falls das Problem, das die Sicherheitsanfälligkeit verursacht hat, erneut auftritt.

Wenn Sie Amazon Inspector deaktivieren, werden die Ergebnisse nach 24 Stunden entfernt. Wenn eine Ressource beendet wird, werden alle Ergebnisse, die sich auf die Ressource beziehen, nach 3 Tagen entfernt. Das Gleiche gilt für alle Ergebnisse, die mit einer Ressource verknüpft sind und für die das Scannen nicht mehr zulässig ist. Wenn AWS Ihr Konto gesperrt wird, werden die Ergebnisse nach 90 Tagen entfernt. Die Ergebnisse für gestoppte Instances bleiben aktiv.

Die Ergebnisse besagen

Amazon Inspector kategorisiert Ergebnisse in die folgenden Bundesstaaten.

Aktiv

Amazon Inspector stuft ein Ergebnis, das nicht behoben wurde, als Aktiv ein.

Unterdrückt

Amazon Inspector stuft ein Ergebnis, das einer oder mehreren Unterdrückungsregeln unterliegt, als Unterdrückt ein.

Closed (Abgeschlossen)

Wenn ein Ergebnis behoben wurde, stuft Amazon Inspector das Ergebnis als geschlossen ein.

Themen

Suchtypen von Amazon Inspector

In diesem Abschnitt werden die verschiedenen Findetypen in Amazon Inspector beschrieben.

Themen

Sicherheitslücke im Package

Package von Ergebnissen zu Sicherheitslücken in Paketen werden Softwarepakete in Ihrer AWS Umgebung identifiziert, die häufig auftretenden Sicherheitslücken und Risiken ausgesetzt sind (CVEs). Angreifer können diese ungepatchten Sicherheitslücken ausnutzen, um die Vertraulichkeit, Integrität oder Verfügbarkeit von Daten zu gefährden oder auf andere Systeme zuzugreifen. Das CVE-System ist eine Referenzmethode für öffentlich bekannte Sicherheitslücken und Sicherheitslücken. Weitere Informationen finden Sie unter https://www.cve.org/.

Amazon Inspector kann Ergebnisse zu Sicherheitslücken in Paketen für EC2 Instances, ECR-Container-Images und Lambda-Funktionen generieren. Die Ergebnisse der Sicherheitslücken

von Paketen enthalten zusätzliche Details, die für diesen Befundtyp einzigartig sind, nämlich den Inspector-Score und die Schwachstelleninformationen.

Sicherheitslücke im Code

Die Ergebnisse von Sicherheitslücken im Code helfen dabei, Codezeilen zu identifizieren, die ausgenutzt werden können. Zu den Sicherheitslücken im Code gehören fehlende Verschlüsselung, Datenlecks, Injektionsfehler und schwache Kryptografie. Amazon Inspector generiert mithilfe von Lambda-Funktionsscans und seiner Code Security-Funktion Erkenntnisse zu Sicherheitslücken im Code.

Amazon Inspector bewertet den Anwendungscode der Lambda-Funktion mithilfe automatisierter Argumentation und maschinellem Lernen, um den Anwendungscode auf allgemeine Sicherheitsbestimmungen hin zu analysieren. Es identifiziert Richtlinienverstöße und Sicherheitslücken auf der Grundlage interner Detektoren, die in Zusammenarbeit mit Amazon entwickelt wurden CodeGuru. Eine Liste möglicher Erkennungen finden Sie unter CodeGuru Detector Library.

Beim Codescan werden Codefragmente erfasst, um erkannte Sicherheitslücken hervorzuheben. Beispielsweise kann ein Codeausschnitt hartcodierte Anmeldeinformationen oder andere vertrauliche Materialien im Klartext enthalten. CodeGuru speichert Codefragmente, die mit Sicherheitslücken im Zusammenhang stehen. Standardmäßig ist Ihr Code mit einem AWS eigenen Schlüssel verschlüsselt. Sie können jedoch einen vom Kunden verwalteten Schlüssel erstellen, um Ihren Code zu verschlüsseln, wenn Sie mehr Kontrolle über diese Informationen haben möchten. Weitere Informationen finden Sie unter Verschlüsselung im Ruhezustand für den Code in Ihren Ergebnissen.

> **Note**
>
> Der delegierte Administrator einer Organisation kann keine Codefragmente einsehen, die zu Mitgliedskonten gehören.

Erreichbarkeit über das Netzwerk

Die Ergebnisse der Netzwerkerreichbarkeit deuten darauf hin, dass es in Ihrer Umgebung offene Netzwerkpfade zu EC2 Amazon-Instances gibt. Diese Ergebnisse treten auf, wenn Ihre TCP- und UDP-Ports von den VPC-Edges aus erreichbar sind, z. B. ein Internet-Gateway (einschließlich Instances hinter Application Load Balancers oder Classic Load Balancers), eine VPC-Peering-Verbindung oder ein VPN über ein virtuelles Gateway. Diese Ergebnisse heben

Netzwerkkonfigurationen hervor, die möglicherweise zu freizügig sind, wie z. B. schlecht verwaltete Sicherheitsgruppen, Zugriffskontrolllisten oder Internet-Gateways, oder die potenziell böswilligen Zugriff ermöglichen.

Amazon Inspector generiert nur Ergebnisse zur Netzwerkerreichbarkeit für EC2 Amazon-Instances. Amazon Inspector führt alle 12 Stunden Scans nach Ergebnissen der Netzwerkerreichbarkeit durch, sobald Amazon Inspector aktiviert ist.

Amazon Inspector bewertet beim Scannen nach Netzwerkpfaden die folgenden Konfigurationen:

- EC2 Amazon-Instanzen
- Application Load Balancer
- Direct Connect
- Elastic Load Balancers
- Elastic-Network-Schnittstellen
- Internet-Gateways
- Listen zur Netzwerkzugriffskontrolle
- Routing-Tabellen
- Sicherheitsgruppen
- Subnets
- Virtuelle private Clouds
- Virtuelle private Gateways
- VPC-Endpunkte
- VPC-Gateway-Endpunkte
- VPC-Peering-Verbindungen
- VPN-Verbindungen

Ihre Amazon Inspector Inspector-Ergebnisse anzeigen

Sie können Ihre Amazon Inspector Inspector-Ergebnisse in der Amazon Inspector Inspector-Konsole und mit der Amazon Inspector `ListFindings`Inspector-API anzeigen. In der Amazon Inspector-Konsole können Sie Ihre Ergebnisse im Amazon Inspector-Dashboard und auf dem Findings-Bildschirm einsehen. Sie können Ihre Ergebnisse auch in AWS Security Hub Amazon Elastic Container Registry (Amazon ECR) einsehen. Standardmäßig zeigen das Amazon Inspector-

Dashboard und der Ergebnisbildschirm Ihre aktiven Ergebnisse an. Sie können Ihre Ergebnisse auch nach Kategorien sortiert anzeigen. Die Verfahren in diesem Abschnitt beschreiben, wie Sie Ihre Ergebnisse in der Amazon Inspector Inspector-Konsole und mit der Amazon Inspector Inspector-API anzeigen können.

Console

Um die Ergebnisse von Amazon Inspector einzusehen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. (Optional) Wählen Sie im Navigationsbereich Dashboard aus. Das Dashboard zeigt einen Überblick über den Versicherungsschutz für Ihre Umgebung und nur Ihre wichtigsten Ergebnisse.
3. (Optional) Wählen Sie im Navigationsbereich Findings aus. Auf dem Bildschirm Ergebnisse werden alle Ihre aktiven Ergebnisse in einer Tabelle angezeigt, in der Sie Ihre Ergebnisse nach Status und Filterkriterien filtern können. Sie können auch Unterdrückungsregeln erstellen, um Ergebnisse aus der Ansicht auszuschließen. Sie können Details zu einem Ergebnis anzeigen, indem Sie den Namen des Ergebnisses auswählen.
4. (Optional) Wählen Sie im Navigationsbereich eine der folgenden Optionen, um Ihre Ergebnisse nach Kategorien geordnet anzuzeigen:

   - Nach Sicherheitslücke — Zeigt Ihre kritischsten Sicherheitslücken an.
   - Nach Konto — Zeigt alle Ihre Konten sowie den Umfang der Scans und die Gesamtzahl der Ergebnisse mit den Schweregraden „Kritisch“ und „Hoch“ an.

     > **Note**
     >
     > Diese Kategorie steht nur delegierten Administratoren zur Verfügung.

   - Nach Instanz — Zeigt Ihre anfälligsten EC2 Amazon-Instances an.

     > **Note**
     >
     > Die in dieser Kategorie zusammengefassten Ergebnisse enthalten keine Informationen zur Netzwerkverfügbarkeit.

   - Nach Container-Image — Zeigt Ihre anfälligsten Amazon ECR-Container-Images an.

- Nach Container-Repository — Zeigt Ihre anfälligsten Repositorys an.
- Nach Lambda-Funktion — Zeigt Ihre anfälligsten Lambda-Funktionen an.

API

Um die Ergebnisse von Amazon Inspector einzusehen

- Führen Sie den ListFindingsAPI-Vorgang aus. Geben Sie in der Anfrage FilterCriteria an, um bestimmte Ergebnisse zurückzugeben.

Details zu Ihren Amazon Inspector Inspector-Ergebnissen anzeigen

Das Verfahren in diesem Abschnitt beschreibt, wie Sie Details zu den Ergebnissen von Amazon Inspector anzeigen können.

Um die Details zu einem Ergebnis anzuzeigen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home
2. Wählen Sie die Region aus, in der Sie sich die Ergebnisse ansehen möchten.
3. Wählen Sie im Navigationsbereich Findings aus, um die Ergebnisliste anzuzeigen
4. (Optional) Verwenden Sie die Filterleiste, um ein bestimmtes Ergebnis auszuwählen. Weitere Informationen finden Sie unter Filtern Ihrer Amazon Inspector Inspector-Ergebnisse.
5. Wählen Sie ein Ergebnis aus, um dessen Detailbereich anzuzeigen.

Der Bereich mit den Ergebnisdetails enthält die grundlegenden Erkennungsmerkmale des Ergebnisses. Dazu gehören der Titel des Befundes sowie eine grundlegende Beschreibung der identifizierten Sicherheitslücke, Vorschläge zur Behebung und eine Bewertung des Schweregrads. Informationen zur Bewertung finden Sie unterErläuterung der Schweregrade Ihrer Amazon Inspector Inspector-Ergebnisse.

Die für ein Ergebnis verfügbaren Details variieren je nach Art des Ergebnisses und der betroffenen Ressource.

Alle Ergebnisse enthalten die AWS-Konto ID-Nummer, für die das Ergebnis identifiziert wurde, einen Schweregrad, einen Befundtyp, das Datum, an dem das Ergebnis erstellt wurde, und einen Abschnitt „Betroffene Ressource“ mit Details zu dieser Ressource.

Der Typ des Ergebnisses bestimmt, welche Informationen zur Behebung und zur Schwachstellenanalyse für das Ergebnis verfügbar sind. Je nach Art des Ergebnisses sind unterschiedliche Ergebnisdetails verfügbar.

Sicherheitslücke im Package

Ergebnisse zu Sicherheitslücken in Paketen sind für EC2 Instances, ECR-Container-Images und Lambda-Funktionen verfügbar. Weitere Informationen finden Sie unter Sicherheitslücke im Package.

Zu den Ergebnissen der Paketschwachstelle gehören auchDen Amazon Inspector-Score anzeigen und Details zur Schwachstellenanalyse verstehen.

Dieser Befundtyp enthält die folgenden Details:

- Fix verfügbar — Zeigt an, ob die Sicherheitsanfälligkeit in einer neueren Version der betroffenen Pakete behoben wurde. Hat einen der folgenden Werte:
  - `YES`, was bedeutet, dass alle betroffenen Pakete eine feste Version haben.
  - `NO`, was bedeutet, dass keine betroffenen Pakete eine feste Version haben.
  - `PARTIAL`, was bedeutet, dass eines oder mehrere (aber nicht alle) der betroffenen Pakete eine feste Version haben.
- Exploit verfügbar — Zeigt an, dass es sich bei der Sicherheitsanfälligkeit um einen bekannten Exploit handelt.
  - `YES`, was bedeutet, dass es sich bei der in Ihrer Umgebung entdeckten Sicherheitslücke um einen bekannten Exploit handelt. Amazon Inspector hat keinen Einblick in die Verwendung von Exploits in einer Umgebung.
  - `NO`, was bedeutet, dass für diese Sicherheitsanfälligkeit kein Exploit bekannt ist.
- Betroffene Pakete — Listet jedes Paket auf, das bei der Entdeckung als gefährdet identifiziert wurde, sowie die Einzelheiten zu jedem Paket:
- Dateipfad — Die EBS-Volume-ID und die Partitionsnummer, die mit einem Ergebnis verknüpft sind. Dieses Feld ist in den Ergebnissen für EC2 Instanzen enthalten, die mit gescannt wurden. Scannen ohne Agenten
- Installierte Version//Behobene Version — Die Versionsnummer des aktuell installierten Pakets, für das eine Sicherheitslücke entdeckt wurde. Vergleichen Sie die installierte Versionsnummer mit dem Wert nach dem Schrägstrich (/). Der zweite Wert ist die Versionsnummer des Pakets, das die entdeckte Sicherheitsanfälligkeit behebt, wie in den allgemeinen Sicherheitslücken (CVEs) oder in der zugehörigen Empfehlung angegeben. Wenn die Sicherheitsanfälligkeit in

mehreren Versionen behoben wurde, wird in diesem Feld die neueste Version aufgeführt, die den Fix enthält. Wenn ein Fix nicht verfügbar ist, ist dieser Wert`None available`.

> **Note**
>
> Wenn ein Ergebnis erkannt wurde, bevor Amazon Inspector begann, dieses Feld in die Ergebnisse aufzunehmen, ist der Wert für dieses Feld leer. Möglicherweise ist jedoch ein Fix verfügbar.

- Paketmanager — Der Paketmanager, der zur Konfiguration dieses Pakets verwendet wurde.
- Problembehebung — Wenn ein Update über ein aktualisiertes Paket oder eine aktualisierte Programmierbibliothek verfügbar ist, enthält dieser Abschnitt die Befehle, die Sie ausführen können, um das Update durchzuführen. Sie können den bereitgestellten Befehl kopieren und in Ihrer Umgebung ausführen.

> **Note**
>
> Die Befehle zur Problembehebung werden aus Datenfeeds von Anbietern bereitgestellt und können je nach Systemkonfiguration variieren. Genauere Hinweise finden Sie in den Referenzen oder in der Dokumentation zum Betriebssystem.

- Details zur Sicherheitslücke — bietet einen Link zur bevorzugten Quelle von Amazon Inspector für das im Ergebnis identifizierte CVE, z. B. National Vulnerability Database (NVD), REDHAT oder ein anderer Betriebssystemanbieter. Darüber hinaus finden Sie die Schweregrade für das Ergebnis. Weitere Informationen zur Bewertung des Schweregrads finden Sie beispielsweise unterErläuterung der Schweregrade Ihrer Amazon Inspector Inspector-Ergebnisse. Die folgenden Werte sind enthalten, einschließlich der jeweiligen Bewertungsvektoren:
  - Ergebnis des Exploit Prediction Scoring Systems (EPSS)
  - Ergebnis des Inspector
  - CVSS 3.1 von Amazon CVE
  - CVSS 3.1 von NVD
  - CVSS 2.0 von NVD (falls zutreffend, für ältere Versionen) CVEs
- Verwandte Sicherheitslücken — Spezifiziert weitere Sicherheitslücken im Zusammenhang mit der Entdeckung. In der Regel handelt es sich CVEs dabei um andere, die dieselbe Paketversion betreffen, oder um andere CVEs , die zur selben Gruppe gehören wie das gefundene CVE, je nach Angabe des Herstellers.

Sicherheitslücke im Code

Die Ergebnisse von Sicherheitslücken im Code sind nur für Lambda-Funktionen verfügbar. Weitere Informationen finden Sie unter Sicherheitslücke im Code. Dieser Erkennungstyp enthält die folgenden Details:

- Fix verfügbar — Bei Code-Schwachstellen gilt dieser Wert immer`YES`.
- Name des Detektors — Der Name des CodeGuru Detektors, der zur Erkennung der Sicherheitslücke im Code verwendet wurde. Eine Liste möglicher Erkennungen finden Sie in der Q Detector Library.
- Melder-Tags — Die mit dem Detektor verknüpften CodeGuru Tags CodeGuru verwenden Tags, um Erkennungen zu kategorisieren.
- Relevante CWE — IDs der Common Weakness Enumeration (CWE), die mit der Sicherheitslücke im Code verknüpft sind.
- Dateipfad — Der Dateispeicherort der Code-Sicherheitslücke.
- Ort der Sicherheitslücke — Für Sicherheitslücken beim Scannen von Lambda-Code zeigt dieses Feld die genauen Codezeilen an, in denen Amazon Inspector die Sicherheitsanfälligkeit gefunden hat.
- Vorgeschlagene Behebung — Hier wird vorgeschlagen, wie der Code bearbeitet werden kann, um das Problem zu beheben.

Erreichbarkeit über das Netzwerk

Ergebnisse zur Erreichbarkeit des Netzwerks sind nur für Instanzen verfügbar. EC2 Weitere Informationen finden Sie unter Erreichbarkeit über das Netzwerk. Dieser Ergebnistyp enthält die folgenden Details:

- Offener Portbereich — Der Portbereich, über den auf die EC2 Instance zugegriffen werden konnte.
- Netzwerkpfade öffnen — Zeigt den Open-Access-Pfad zur EC2 Instance an. Wählen Sie ein Element im Pfad aus, um weitere Informationen zu erhalten.
- Behebung — Empfiehlt eine Methode zum Schließen des offenen Netzwerkpfads.

Den Amazon Inspector-Score anzeigen und Details zur Schwachstellenanalyse verstehen

Amazon Inspector erstellt eine Bewertung für die Ergebnisse der Amazon Elastic Compute Cloud (Amazon EC2) -Instance. Sie können den Amazon Inspector-Score und die Informationen zu Sicherheitslücken in der Amazon Inspector Inspector-Konsole einsehen. Der Amazon Inspector Score bietet Ihnen Details, die Sie mit den Kennzahlen im Common Vulnerability Scoring System vergleichen können. Diese Details sind nur für die Entdeckung von Sicherheitslücken in Paketen verfügbar. In diesem Abschnitt wird beschrieben, wie Sie den Amazon Inspector-Score interpretieren und Informationen zu Sicherheitslücken verstehen.

Amazon Inspector-Punktzahl

Der Amazon Inspector-Score ist ein kontextualisierter Score, den Amazon Inspector für jedes EC2 gefundene Instance erstellt. Der Amazon Inspector-Score wird bestimmt, indem die Basisinformationen des CVSS v3.1-Scores mit Informationen korreliert werden, die während der Scans aus Ihrer Computerumgebung gesammelt wurden, wie z. B. Ergebnisse zur Netzwerkerreichbarkeit und Daten zur Ausnutzbarkeit. Beispielsweise kann der Amazon Inspector-Score eines Ergebnisses niedriger sein als der Basiswert, wenn die Sicherheitsanfälligkeit über das Netzwerk ausgenutzt werden kann, Amazon Inspector jedoch feststellt, dass kein offener Netzwerkpfad zur anfälligen Instance über das Internet verfügbar ist.

Die Basisbewertung für ein Ergebnis ist die vom Anbieter bereitgestellte CVSS v3.1-Basisbewertung. RHEL-, Debian- oder Amazon-Hersteller-Basiswerte werden unterstützt. Für andere Anbieter oder für Fälle, in denen der Anbieter keine Bewertung angegeben hat, verwendet Amazon Inspector die Basisbewertung aus der National Vulnerability Database (NVD). Amazon Inspector verwendet den Common Vulnerability Scoring System Version 3.1 Calculator, um den Score zu berechnen. Sie können die Quelle der Basisbewertung eines einzelnen Ergebnisses in den Details des Ergebnisses unter den Schwachstellendetails als Quelle der Sicherheitslücke (oder packageVulnerabilityDetails.source in der Ergebnis-JSON) sehen

> **Note**
>
> Der Amazon Inspector Score ist für Linux-Instances, auf denen Ubuntu ausgeführt wird, nicht verfügbar. Das liegt daran, dass Ubuntu seinen eigenen Schweregrad für Sicherheitslücken definiert, der sich vom zugehörigen CVE-Schweregrad unterscheiden kann.

Einzelheiten zur Amazon Inspector-Punktzahl

Wenn Sie die Detailseite eines Befundes öffnen, können Sie die Registerkarte Inspector Score und Vulnerability Intelligence auswählen. Dieses Feld zeigt den Unterschied zwischen dem Basiswert und dem Inspector-Score. In diesem Abschnitt wird erklärt, wie Amazon Inspector den Schweregrad auf der Grundlage einer Kombination aus dem Amazon Inspector-Score und dem Hersteller-Score für das Softwarepaket zugewiesen hat. Wenn sich die Punktzahlen unterscheiden, wird in diesem Bereich erklärt, warum.

Im Abschnitt CVSS-Score-Metriken finden Sie eine Tabelle mit Vergleichen zwischen den CVSS-Basisscore-Metriken und dem Inspector-Score. Bei den verglichenen Kennzahlen handelt es sich um die Basiskennzahlen, die im CVSS-Spezifikationsdokument definiert sind, das von verwaltet wird. first.org Im Folgenden finden Sie eine Zusammenfassung der Basiskennzahlen:

Angriffsvektor

Der Kontext, in dem eine Sicherheitslücke ausgenutzt werden kann. Bei Ergebnissen von Amazon Inspector kann dies „Netzwerk“, „Angrenzendes Netzwerk“ oder „Lokal“ sein.

Komplexität des Angriffs

Dies beschreibt den Schwierigkeitsgrad, mit dem ein Angreifer konfrontiert sein wird, wenn er die Sicherheitsanfälligkeit ausnutzt. Eine niedrige Punktzahl bedeutet, dass der Angreifer nur wenige oder keine zusätzlichen Bedingungen erfüllen muss, um die Sicherheitsanfälligkeit auszunutzen. Eine hohe Punktzahl bedeutet, dass ein Angreifer erhebliche Anstrengungen unternehmen muss, um einen erfolgreichen Angriff mit dieser Sicherheitsanfälligkeit durchzuführen.

Privileg erforderlich

Dies beschreibt die Rechte, die ein Angreifer benötigt, um eine Sicherheitslücke auszunutzen.

Interaktion mit dem Benutzer

Diese Metrik gibt an, ob für einen erfolgreichen Angriff, der diese Sicherheitsanfälligkeit ausnutzt, ein anderer menschlicher Benutzer als der Angreifer erforderlich ist.

Scope

Dies gibt an, ob sich eine Sicherheitsanfälligkeit in einer anfälligen Komponente auf Ressourcen in Komponenten auswirkt, die über den Sicherheitsbereich der anfälligen Komponente hinausgehen. Wenn dieser Wert Unverändert ist, sind die betroffene Ressource und die betroffene Ressource identisch. Wenn dieser Wert geändert ist, kann die anfällige Komponente ausgenutzt werden, um Ressourcen zu beeinträchtigen, die von verschiedenen Sicherheitsbehörden verwaltet werden.

Vertraulichkeit

Dabei wird das Ausmaß der Auswirkungen auf die Vertraulichkeit von Daten innerhalb einer Ressource gemessen, wenn die Sicherheitsanfälligkeit ausgenutzt wird. Dies reicht von „Keine“, bei der keine Vertraulichkeit verloren geht, bis hin zu „Hoch“, bei der alle Informationen innerhalb einer Ressource weitergegeben werden oder vertrauliche Informationen wie Passwörter oder Verschlüsselungsschlüssel preisgegeben werden können.

Integrität

Dabei wird das Ausmaß der Auswirkungen auf die Integrität der Daten innerhalb der betroffenen Ressource gemessen, wenn die Sicherheitsanfälligkeit ausgenutzt wird. Die Integrität ist gefährdet, wenn der Angreifer Dateien innerhalb der betroffenen Ressourcen verändert. Die Bewertung reicht von „Keine“, wobei der Angriff es einem Angreifer nicht ermöglicht, Informationen zu ändern, bis hin zu „Hoch“, bei dem die Sicherheitsanfälligkeit es einem Angreifer ermöglichen würde, einige oder alle Dateien zu ändern, oder die Dateien, die geändert werden könnten, schwerwiegende Folgen haben könnten.

Verfügbarkeit

Damit wird das Ausmaß der Auswirkungen auf die Verfügbarkeit der betroffenen Ressource gemessen, wenn die Sicherheitsanfälligkeit ausgenutzt wird. Die Bewertung reicht von „Keine“, wenn die Sicherheitsanfälligkeit die Verfügbarkeit überhaupt nicht beeinträchtigt, bis hin zu „Hoch“, bei dem der Angreifer bei Ausnutzung die Verfügbarkeit der Ressource vollständig verweigern oder dafür sorgen kann, dass ein Dienst nicht verfügbar ist.

Informationen zu Sicherheitslücken

In diesem Abschnitt werden die verfügbaren Informationen über das CVE von Amazon sowie branchenübliche Quellen für Sicherheitsinformationen wie Recorded Future und Cybersecurity and Infrastructure Security Agency (CISA) zusammengefasst.

> **Note**
>
> Intel von CISA, Amazon oder Recorded Future wird nicht für alle CVEs verfügbar sein.

Sie können sich die Informationen zu Sicherheitslücken in der Konsole oder mithilfe der BatchGetFindingDetailsAPI ansehen. Die folgenden Details sind in der Konsole verfügbar:

ATT&CK

In diesem Abschnitt werden die Taktiken, Techniken und Verfahren (TTPs) von MITRE im Zusammenhang mit dem CVE beschrieben. Die zugehörigen TTPs werden angezeigt. Wenn mehr als zwei zutreffen, können TTPs Sie den Link auswählen, um eine vollständige Liste anzuzeigen. Wenn Sie eine Taktik oder Technik auswählen, werden Informationen zu dieser Taktik oder Technik auf der MITRE-Website geöffnet.

CISA

Dieser Abschnitt behandelt relevante Daten im Zusammenhang mit der Sicherheitsanfälligkeit. Das Datum, an dem die Cybersecurity and Infrastructure Security Agency (CISA) die Sicherheitsanfälligkeit aufgrund von Hinweisen auf eine aktive Ausnutzung der Sicherheitslücke in den Katalog der bekannten Sicherheitslücken aufgenommen hat, und das Fälligkeitsdatum, bis zu dem die Systeme gepatcht werden sollen. Diese Informationen stammen von CISA.

Bekannte Schadsoftware

In diesem Abschnitt sind bekannte Exploit-Kits und Tools aufgeführt, die diese Sicherheitsanfälligkeit ausnutzen.

Beweise

In diesem Abschnitt werden die wichtigsten Sicherheitsereignisse im Zusammenhang mit dieser Sicherheitsanfälligkeit zusammengefasst. Wenn mehr als 3 Ereignisse dieselbe Kritikalitätsstufe haben, werden die drei jüngsten Ereignisse angezeigt.

Zuletzt gemeldet

In diesem Abschnitt wird das Datum des letzten bekannten öffentlichen Exploits für diese Sicherheitsanfälligkeit angezeigt.

Erläuterung der Schweregrade Ihrer Amazon Inspector Inspector-Ergebnisse

Wenn Amazon Inspector ein Ergebnis generiert, weist es dem Ergebnis einen Schweregrad zu. Schweregrade helfen Ihnen dabei, Ihre Ergebnisse zu bewerten und zu priorisieren. Der Schweregrad eines Ergebnisses entspricht einem numerischen Wert und einer Stufe: informativ, niedrig, mittel, hoch und kritisch. Amazon Inspector bestimmt den Schweregrad eines Ergebnisses anhand des Ergebnistyps. In diesem Abschnitt wird beschrieben, wie Amazon Inspector für jeden Befundtyp eine Schweregradbewertung ermittelt.

Schweregrad der Sicherheitslücke im Softwar

Amazon Inspector verwendet die NVD/CVSS Bewertung als Grundlage für die Bewertung des Schweregrads von Sicherheitslücken in Softwarepaketen. Bei der NVD/CVSS Bewertung handelt es sich um den Schweregrad der Sicherheitslücke, der vom NVD veröffentlicht und vom CVSS definiert wurde. Die NVD/CVSS Bewertung setzt sich aus Sicherheitsmetriken wie der Komplexität des Angriffs, dem Reifegrad des Exploit-Codes und den erforderlichen Rechten zusammen. Amazon Inspector erstellt eine numerische Bewertung von 1 bis 10, die den Schweregrad der Sicherheitsanfälligkeit widerspiegelt. Amazon Inspector stuft dies als Basiswert ein, da er den Schweregrad einer Sicherheitslücke anhand ihrer intrinsischen Merkmale widerspiegelt, die im Laufe der Zeit konstant sind. Bei dieser Bewertung wird auch davon ausgegangen, dass die Auswirkungen im schlimmsten Fall auf verschiedene bereitgestellte Umgebungen angemessen sind. Der CVSS v3-Standard ordnet die CVSS-Scores den folgenden Schweregraden zu.

Ergebnis	Bewertung
0	Informativ
0,1—3,9	Niedrig
4,0—6,9	Mittelschwer
7,0—8,9	Hoch
9,0—10,0	Kritisch

Die gefundenen Sicherheitslücken in Paketen können auch den Schweregrad Untriaged haben. Das bedeutet, dass der Anbieter noch keinen Schwachstellen-Score für die entdeckte Sicherheitslücke festgelegt hat. In diesem Fall empfehlen wir, die Referenz URLs für das Ergebnis zu verwenden, um diese Sicherheitsanfälligkeit zu untersuchen und entsprechend zu reagieren.

Zu den Ergebnissen der Paketschwachstellen gehören die folgenden Bewertungen und die zugehörigen Bewertungsvektoren als Teil der Ergebnisdetails:

- EPSS-Score
- Ergebnis des Inspector
- CVSS 3.1 von Amazon CVE
- CVSS 3.1 von NVD

- CVSS 2.0 von NVD (falls zutreffend)

Schweregrad der Sicherheitslücke

Für die Suche nach Sicherheitslücken im Code verwendet Amazon Inspector die Schweregrade, die von den CodeGuru Amazon-Detektoren definiert wurden, die den Befund generiert haben. Jedem Detektor wird mithilfe des CVSS v3-Bewertungssystems ein Schweregrad zugewiesen. Eine Erläuterung der CodeGuru verwendeten Schweregrade finden Sie unter Schweregraddefinitionen im CodeGuru Leitfaden. Eine Liste der Melder nach Schweregrad finden Sie, wenn Sie eine der folgenden unterstützten Programmiersprachen auswählen:

- Python-Detektoren nach Schweregrad
- Java-Detektoren nach Schweregrad

Schweregrad der Netzwerkerreichbarkeit

Amazon Inspector bestimmt den Schweregrad einer Sicherheitslücke im Netzwerk auf der Grundlage der offengelegten Services, Ports und Protokolle sowie der Art des offenen Pfads. In der folgenden Tabelle werden diese Schweregrade definiert. Der Wert in der Spalte Bewertung offener Pfade steht für offene Pfade von virtuellen Gateways, Peering-Verbindungen und VPCs AWS Direct Connect Netzwerken. Für alle anderen exponierten Dienste, Ports und Protokolle wurde der Schweregrad „Information“ eingestuft.

Service	TCP-Ports	UDP-Anschlüsse	Bewertung des Internetpfads	Pfadbewertung öffnen
DHCP	67, 68, 546, 547	67, 68, 546, 547	Mittelschwer	Informativ
Elasticsearch	9300, 9200	N/A	Mittelschwer	Informativ
FTP	21	21	Hoch	Mittelschwer
Global Catalog LDAP	3268	N/A	Mittelschwer	Informativ
Global Catalog LDAP über TLS	3269	N/A	Mittelschwer	Informativ

HTTP	80	80	Niedrig	Informativ
HTTPS	443	443	Niedrig	Informativ
Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752	Mittelschwer	Informativ
LDAP	389	389	Mittelschwer	Informativ
LDAP über TLS	636	N/A	Mittelschwer	Informativ
MongoDB	27017, 27018, 27019, 28017	N/A	Mittelschwer	Informativ
MySQL	3306	N/A	Mittelschwer	Informativ
NetBIOS	137, 139	137, 138	Mittelschwer	Informativ
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110	Mittelschwer	Informativ
Oracle	1521, 1630	N/A	Mittelschwer	Informativ
PostgreSQL	5432	N/A	Mittelschwer	Informativ
Druckdienste	515	N/A	Hoch	Mittelschwer
RDP	3389	3389	Mittelschwer	Niedrig
RPC	111, 135, 530	111, 135, 530	Mittelschwer	Informativ
SMB	445	445	Mittelschwer	Informativ
SSH	22	22	Mittelschwer	Niedrig
SQL Server	1433	1434	Mittelschwer	Informativ
Syslog	601	514	Mittelschwer	Informativ
Telnet	23	23	Hoch	Mittelschwer
WINS	1512, 42	1512, 42	Mittelschwer	Informativ

Ergebnisse in Amazon Inspector verwalten

Mit Amazon Inspector können Sie Ihre Ergebnisse auf unterschiedliche Weise verwalten. Sie können Ihre Ergebnisse nach ihrem Status filtern. Sie können Ihre Ergebnisse anhand von Filterkriterien durchsuchen. Sie können Unterdrückungsregeln erstellen, um Ergebnisse aus Ihrer Ergebnisliste auszuschließen. Sie können Ergebnisse auch nach AWS Security Hub Amazon und Amazon EventBridge Simple Storage Service (Amazon S3) exportieren.

Themen

Filtern Ihrer Amazon Inspector Inspector-Ergebnisse

Sie können Ihre Amazon Inspector Inspector-Ergebnisse mithilfe von Filterkriterien filtern. Wenn ein Ergebnis nicht Ihren Filterkriterien entspricht, schließt Amazon Inspector das Ergebnis aus der Ansicht aus. In diesem Abschnitt wird beschrieben, wie Sie Ihre Amazon Inspector Inspector-Ergebnisse mithilfe von Filterkriterien filtern können.

Filter in der Amazon Inspector Inspector-Konsole erstellen

In jeder Ergebnisansicht können Sie die Filterfunktion verwenden, um Ergebnisse mit bestimmten Merkmalen zu finden. Filter werden entfernt, wenn Sie zu einer anderen Ansicht mit Registerkarten wechseln.

Ein Filter besteht aus einem Filterkriterium, das aus einem Filterattribut und einem Filterwert besteht. Ergebnisse, die Ihren Filterkriterien nicht entsprechen, werden von der Ergebnisliste ausgeschlossen. Um beispielsweise alle Ergebnisse zu sehen, die mit Ihrem Administratorkonto verknüpft sind, können Sie das AWS Konto-ID-Attribut auswählen und es mit dem Wert Ihrer zwölfstelligen AWS Konto-ID verknüpfen.

Einige Filterkriterien gelten für alle Ergebnisse, während andere nur für bestimmte Ressourcentypen oder nur für Suchtypen verfügbar sind.

Um einen Filter auf die Ergebnisansicht anzuwenden

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Findings aus. In der Standardansicht werden alle Ergebnisse mit dem Status Aktiv angezeigt.
3. Um Ergebnisse nach Kriterien zu filtern, wählen Sie die Filterleiste Hinzufügen aus, um eine Liste aller zutreffenden Filterkriterien für diese Ansicht anzuzeigen. Verschiedene Filterkriterien sind in verschiedenen Ansichten verfügbar.
4. Wählen Sie aus der Liste ein Kriterium aus, nach dem Sie filtern möchten.
5. Geben Sie im Kriterieneingabebereich die gewünschten Filterwerte ein, um dieses Kriterium zu definieren.
6. Wählen Sie Anwenden, um dieses Filterkriterium auf Ihre aktuellen Ergebnisse anzuwenden. Sie können weitere Filterkriterien hinzufügen, indem Sie erneut die Filtereingabeleiste auswählen.
7. (Optional) Um Ihre unterdrückten oder geschlossenen Ergebnisse anzuzeigen, wählen Sie in der Filterleiste Aktiv und dann Unterdrückt oder Geschlossen aus. Wählen Sie „Alle anzeigen“, um aktive, unterdrückte und geschlossene Ergebnisse in derselben Ansicht anzuzeigen.

Unterdrückung der Ergebnisse von Amazon Inspector

Sie können Unterdrückungsregeln erstellen, um Ergebnisse auszublenden, die den Kriterien entsprechen. Sie können beispielsweise eine Unterdrückungsregel erstellen, um Ergebnisse anhand ihres Schweregrads auszublenden. Wenn Amazon Inspector ein Ergebnis generiert, das Ihrer Unterdrückungsregel entspricht, unterdrückt Amazon Inspector das Ergebnis und blendet es aus. Amazon Inspector speichert unterdrückte Ergebnisse, bis sie behoben sind. Sobald ein unterdrücktes Ergebnis behoben wurde, schließt Amazon Inspector das Ergebnis. Sie können unterdrückte Ergebnisse in der Konsole einsehen.

Sie erstellen Unterdrückungsregeln, um Ihre wichtigsten Ergebnisse zu priorisieren. Unterdrückungsregeln haben keine Auswirkungen auf Ihre Ergebnisse, da sie nur Ergebnisse vor dem Zugriff verbergen. Sie können keine Unterdrückungsregel erstellen, die Ergebnisse schließt oder behebt. Sie können unerwünschte Ergebnisse auch AWS Security Hub mit einer EventBridge Amazon-Regel unterdrücken. Die Verfahren in diesem Abschnitt beschreiben, wie Sie eine Unterdrückungsregel erstellen, anzeigen, bearbeiten und löschen.

> **Note**
>
> Nur der delegierte Administrator einer Organisation kann Unterdrückungsregeln erstellen und verwalten.

Eine Unterdrückungsregel erstellen

Sie können Unterdrückungsregeln erstellen, um die Liste der Ergebnisse zu filtern, die standardmäßig angezeigt werden. Sie können eine Unterdrückungsregel programmgesteuert erstellen, indem Sie die CreateFilterAPI verwenden und `SUPRESS` als Wert für angeben. `action`

> **Note**
>
> Nur eigenständige Konten und delegierte Amazon Inspector-Administratoren können Unterdrückungsregeln erstellen und verwalten. Mitgliedern einer Organisation wird im Navigationsbereich keine Option für Unterdrückungsregeln angezeigt.

Um eine Unterdrückungsregel zu erstellen (Konsole)

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich die Option Suppression Rules aus. Wählen Sie dann Create rule (Regel erstellen) aus.
3. Gehen Sie für jedes Kriterium wie folgt vor:
   - Wählen Sie die Filterleiste aus, um eine Liste mit Filterkriterien anzuzeigen, die Sie zu Ihrer Unterdrückungsregel hinzufügen können.
   - Wählen Sie die Filterkriterien für Ihre Unterdrückungsregel aus.
4. Wenn Sie mit dem Hinzufügen von Kriterien fertig sind, geben Sie einen Namen für die Regel und optional eine Beschreibung ein.
5. Wählen Sie Save rule (Regel speichern). Amazon Inspector wendet sofort die neue Unterdrückungsregel an und verbirgt alle Ergebnisse, die den Kriterien entsprechen.

Unterdrückte Ergebnisse anzeigen

Standardmäßig zeigt Amazon Inspector keine unterdrückten Ergebnisse in der Amazon Inspector Inspector-Konsole an. Sie können sich jedoch die Ergebnisse ansehen, die durch eine bestimmte Regel unterdrückt wurden.

Um unterdrückte Ergebnisse anzuzeigen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Unterdrückungsregeln aus.
3. Wählen Sie in der Liste der Unterdrückungsregeln den Titel der Regel aus.

Eine Unterdrückungsregel bearbeiten

Sie können jederzeit Änderungen an den Unterdrückungsregeln vornehmen.

Um Unterdrückungsregeln zu ändern

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich die Option Suppression Rules aus.
3. Wählen Sie den Namen der Unterdrückungsregel aus, die Sie ändern möchten, und klicken Sie dann auf Bearbeiten.
4. Nehmen Sie die gewünschten Änderungen vor und wählen Sie dann Speichern.

Löschen einer Unterdrückungsregel

Sie können Unterdrückungsregeln löschen. Wenn Sie eine Unterdrückungsregel löschen, beendet Amazon Inspector die Unterdrückung neuer und vorhandener Ergebnisse, die die Regelkriterien erfüllen und nicht durch andere Regeln unterdrückt werden.

Nachdem Sie eine Unterdrückungsregel gelöscht haben, haben neue und bestehende Ergebnisse, die die Kriterien der Regel erfüllen, den Status Aktiv. Das bedeutet, dass sie standardmäßig auf der Amazon Inspector Inspector-Konsole angezeigt werden. Darüber hinaus veröffentlicht Amazon Inspector diese Ergebnisse im EventBridge Rahmen von Veranstaltungen an AWS Security Hub und Amazon.

Um eine Unterdrückungsregel zu löschen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Unterdrückungsregeln aus.
3. Aktivieren Sie das Kontrollkästchen neben dem Titel der Unterdrückungsregel, die Sie löschen möchten.
4. Wählen Sie Löschen und bestätigen Sie dann Ihre Auswahl, um die Regel dauerhaft zu löschen.

Exportieren von Amazon Inspector Inspector-Ergebnisberichten

Ein Ergebnisbericht ist eine CSV- oder JSON-Datei, die einen detaillierten Überblick über Ihre Ergebnisse bietet. Sie können einen Ergebnisbericht nach AWS Security Hub Amazon und Amazon EventBridge Simple Storage Service (Amazon S3) exportieren. Wenn Sie einen Ergebnisbericht konfigurieren, geben Sie an, welche Ergebnisse darin enthalten sein sollen. Standardmäßig enthält Ihr Ergebnisbericht Daten für alle Ihre aktiven Ergebnisse. Wenn Sie der delegierte Administrator für eine Organisation sind, enthält Ihr Ergebnisbericht Daten für alle Mitgliedskonten in der Organisation. Um einen Ergebnisbericht anzupassen, erstellen Sie einen Filter und wenden Sie ihn an.

Wenn Sie einen Ergebnisbericht exportieren, verschlüsselt Amazon Inspector Ihre Ergebnisdaten mit einer AWS KMS key , die Sie angeben. Nachdem Amazon Inspector Ihre Ergebnisdaten verschlüsselt hat, speichert es Ihren Befundbericht in einem von Ihnen angegebenen Amazon S3 S3-Bucket. Ihr AWS KMS Schlüssel muss in demselben AWS-Region wie Ihr Amazon S3 S3-Bucket verwendet werden. Ihre AWS KMS Schlüsselrichtlinie muss Amazon Inspector erlauben, sie zu verwenden, und Ihre Amazon S3 S3-Bucket-Richtlinie muss es Amazon Inspector ermöglichen, ihr Objekte hinzuzufügen. Nachdem Sie Ihren Ergebnisbericht exportiert haben, können Sie ihn aus Ihrem Amazon S3 S3-Bucket herunterladen oder an einen neuen Speicherort übertragen. Sie können Ihren Amazon S3 S3-Bucket auch als Repository für andere exportierte Ergebnisberichte verwenden.

In diesem Abschnitt wird beschrieben, wie Sie einen Ergebnisbericht in der Amazon Inspector Inspector-Konsole exportieren. Für die folgenden Aufgaben müssen Sie Ihre Berechtigungen überprüfen, einen Amazon S3 S3-Bucket konfigurieren AWS KMS key, einen konfigurieren und einen Ergebnisbericht konfigurieren und exportieren.

> **Note**
>
> Wenn Sie einen Ergebnisbericht mit der Amazon Inspector CreateFindingsReportAPI exportieren, können Sie nur Ihre aktiven Ergebnisse anzeigen. Wenn Sie Ihre unterdrückten oder geschlossenen Ergebnisse anzeigen möchten, müssen Sie dies `CLOSED` als Teil Ihrer Filterkriterien angeben`SUPPRESSED`.

Aufgaben

- Schritt 1: Überprüfen Sie Ihre Berechtigungen
- Schritt 2: Konfigurieren Sie einen S3-Bucket
- Schritt 3: Konfigurieren Sie eine AWS KMS key
- Schritt 4: Konfigurieren und exportieren Sie einen Ergebnisbericht
- Beheben Sie Exportfehler

Schritt 1: Überprüfen Sie Ihre Berechtigungen

> **Note**
>
> Nachdem Sie einen Ergebnisbericht zum ersten Mal exportiert haben, sind die Schritte 1—3 optional. Das Befolgen dieser Schritte hängt davon ab, ob Sie denselben Amazon S3 S3-Bucket und AWS KMS key für andere exportierte Ergebnisberichte verwenden möchten. Wenn Sie nach Abschluss der Schritte 1—3 einen Ergebnisbericht programmgesteuert exportieren möchten, verwenden Sie die Amazon CreateFindingsReportInspector API.

Bevor Sie einen Ergebnisbericht aus Amazon Inspector exportieren, stellen Sie sicher, dass Sie über die erforderlichen Berechtigungen verfügen, um sowohl Ergebnisberichte zu exportieren als auch Ressourcen für die Verschlüsselung und Speicherung der Berichte zu konfigurieren. Um Ihre Berechtigungen zu überprüfen, verwenden Sie AWS Identity and Access Management (IAM), um die IAM-Richtlinien zu überprüfen, die mit Ihrer IAM-Identität verknüpft sind. Vergleichen Sie dann die Informationen in diesen Richtlinien mit der folgenden Liste von Aktionen, die Sie ausführen dürfen müssen, um einen Ergebnisbericht zu exportieren.

Amazon Inspector

Stellen Sie für Amazon Inspector sicher, dass Sie die folgenden Aktionen ausführen dürfen:

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

Diese Aktionen ermöglichen es Ihnen, Ergebnisdaten für Ihr Konto abzurufen und diese Daten in Ergebnisberichten zu exportieren.

Wenn Sie planen, umfangreiche Berichte programmgesteuert zu exportieren, können Sie auch überprüfen, ob Sie die folgenden Aktionen ausführen dürfen:`inspector2:GetFindingsReportStatus`, um den Status von Berichten zu überprüfen und`inspector2:CancelFindingsReport`, um laufende Exporte abzubrechen.

AWS KMS

Stellen Sie sicher AWS KMS, dass Sie die folgenden Aktionen ausführen dürfen:

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

Mit diesen Aktionen können Sie die Schlüsselrichtlinie für das abrufen und aktualisieren AWS KMS key , das Amazon Inspector zur Verschlüsselung Ihres Berichts verwenden soll.

Um die Amazon Inspector Inspector-Konsole zum Exportieren eines Berichts zu verwenden, stellen Sie außerdem sicher, dass Sie die folgenden AWS KMS Aktionen ausführen dürfen:

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

Mit diesen Aktionen können Sie Informationen über das AWS KMS keys für Ihr Konto abrufen und anzeigen. Sie können dann einen dieser Schlüssel auswählen, um Ihren Bericht zu verschlüsseln.

Wenn Sie vorhaben, einen neuen KMS-Schlüssel für die Verschlüsselung Ihres Berichts zu erstellen, müssen Sie auch berechtigt sein, die `kms:CreateKey` Aktion auszuführen.

Amazon S3

Stellen Sie für Amazon S3 sicher, dass Sie die folgenden Aktionen ausführen dürfen:

- `s3:CreateBucket`

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

  Mit diesen Aktionen können Sie den S3-Bucket erstellen und konfigurieren, in dem Amazon Inspector Ihren Bericht speichern soll. Sie ermöglichen Ihnen auch das Hinzufügen und Löschen von Objekten aus dem Bucket.

  Wenn Sie planen, Ihren Bericht mit der Amazon Inspector Inspector-Konsole zu exportieren, überprüfen Sie auch, ob Sie die `s3:ListAllMyBuckets` `s3:GetBucketLocation` Aktionen ausführen dürfen. Mit diesen Aktionen können Sie Informationen zu den S3-Buckets für Ihr Konto abrufen und anzeigen. Sie können dann einen dieser Buckets auswählen, um den Bericht zu speichern.

Wenn Sie eine oder mehrere der erforderlichen Aktionen nicht ausführen dürfen, bitten Sie Ihren AWS Administrator um Unterstützung, bevor Sie mit dem nächsten Schritt fortfahren.

Schritt 2: Konfigurieren Sie einen S3-Bucket

Nachdem Sie Ihre Berechtigungen überprüft haben, können Sie den S3-Bucket konfigurieren, in dem Sie Ihren Ergebnisbericht speichern möchten. Dabei kann es sich um einen vorhandenen Bucket für Ihr eigenes Konto oder um einen vorhandenen Bucket handeln, der einem anderen gehört AWS-Konto und auf den Sie zugreifen dürfen. Wenn Sie Ihren Bericht in einem neuen Bucket speichern möchten, erstellen Sie den Bucket, bevor Sie fortfahren.

Der S3-Bucket muss sich im selben AWS-Region Verzeichnis befinden wie die Ergebnisdaten, die Sie exportieren möchten. Wenn Sie beispielsweise Amazon Inspector in der Region USA Ost (Nord-Virginia) verwenden und Ergebnisdaten für diese Region exportieren möchten, muss sich der Bucket auch in der Region USA Ost (Nord-Virginia) befinden.

Darüber hinaus muss die Richtlinie des Buckets Amazon Inspector das Hinzufügen von Objekten zum Bucket ermöglichen. In diesem Thema wird erklärt, wie die Bucket-Richtlinie aktualisiert wird, und es gibt ein Beispiel für die Anweisung, die der Richtlinie hinzugefügt werden soll. Ausführliche

Informationen zum Hinzufügen und Aktualisieren von Bucket-Richtlinien finden Sie unter Verwenden von Bucket-Richtlinien im Amazon Simple Storage Service-Benutzerhandbuch.

Wenn Sie Ihren Bericht in einem S3-Bucket speichern möchten, der einem anderen Konto gehört, arbeiten Sie mit dem Besitzer des Buckets zusammen, um die Richtlinie des Buckets zu aktualisieren. Rufen Sie auch den URI für den Bucket ab. Sie müssen diesen URI eingeben, wenn Sie Ihren Bericht exportieren.

Um die Bucket-Richtlinie zu aktualisieren

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon S3 S3-Konsole unter https://console.aws.amazon.com/s3.
2. Wählen Sie im Navigationsbereich die Option Buckets aus.
3. Wählen Sie den S3-Bucket aus, in dem Sie den Ergebnisbericht speichern möchten.
4. Wählen Sie die Registerkarte Berechtigungen.
5. Wählen Sie im Abschnitt Bucket-Richtlinie die Option Bearbeiten aus.
6. Kopieren Sie die folgende Beispielanweisung in Ihre Zwischenablage:

```
{
 "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:::amzn-s3-demo-bucket/*",
   "Condition": {
    "StringEquals": {
     "aws:SourceAccount": "111122223333"
    },
    "ArnLike": {
     "aws:SourceArn": "arn:aws:inspector2:Region:111122223333:report/*"
    }
```

```
    }
   }
  ]
}
```

7. Fügen Sie im Bucket-Policy-Editor auf der Amazon S3 S3-Konsole die vorherige Anweisung in die Richtlinie ein, um sie der Richtlinie hinzuzufügen.

   Wenn Sie die Anweisung hinzufügen, stellen Sie sicher, dass die Syntax gültig ist. Bucket-Richtlinien verwenden das JSON-Format. Das bedeutet, dass Sie vor oder nach der Anweisung ein Komma hinzufügen müssen, je nachdem, wo Sie die Anweisung zur Richtlinie hinzufügen. Wenn Sie die Anweisung als letzte Anweisung hinzufügen, fügen Sie hinter der schließenden Klammer für die vorherige Anweisung ein Komma hinzu. Wenn Sie sie als erste Anweisung oder zwischen zwei vorhandenen Anweisungen hinzufügen, fügen Sie hinter der schließenden Klammer für die Anweisung ein Komma ein.

8. Aktualisieren Sie die Anweisung mit den richtigen Werten für Ihre Umgebung, wobei:

   - *`amzn-s3-demo-bucket`*ist der Name des Buckets.
   - *`111122223333`*ist die Konto-ID für Ihre AWS-Konto.
   - *`Region`*ist der, AWS-Region in dem Sie Amazon Inspector verwenden und Amazon Inspector erlauben möchten, Berichte zum Bucket hinzuzufügen. Zum Beispiel `us-east-1` für die Region USA Ost (Nord-Virginia).

   > **Note**
   >
   > Wenn Sie Amazon Inspector in einem manuell aktivierten System verwenden AWS-Region, fügen Sie dem Wert für das `Service` Feld auch den entsprechenden Regionalcode hinzu. Dieses Feld gibt den Amazon Inspector Service Principal an. Wenn Sie beispielsweise Amazon Inspector in der Region Naher Osten (Bahrain) verwenden, die den Regionalcode hat`me-south-1`, `inspector2.amazonaws.com` ersetzen Sie ihn `inspector2.me-south-1.amazonaws.com` in der Anweisung durch.

   Beachten Sie, dass die Beispielanweisung Bedingungen definiert, die zwei globale IAM-Bedingungsschlüssel verwenden:

- aws: SourceAccount — Diese Bedingung ermöglicht es Amazon Inspector, dem Bucket Berichte nur für Ihr Konto hinzuzufügen. Es verhindert, dass Amazon Inspector dem Bucket Berichte für andere Konten hinzufügt. Genauer gesagt gibt die Bedingung an, welches Konto den Bucket für die in der `aws:SourceArn` Bedingung angegebenen Ressourcen und Aktionen verwenden kann.

  Um Berichte für weitere Konten im Bucket zu speichern, fügen Sie dieser Bedingung die Konto-ID für jedes weitere Konto hinzu. Zum Beispiel:

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

- aws: SourceArn — Diese Bedingung schränkt den Zugriff auf den Bucket basierend auf der Quelle der Objekte ein, die dem Bucket hinzugefügt werden. Sie verhindert, dass andere AWS-Services Objekte zum Bucket hinzufügen. Es verhindert auch, dass Amazon Inspector Objekte zum Bucket hinzufügt und gleichzeitig andere Aktionen für Ihr Konto ausführt. Genauer gesagt erlaubt die Bedingung Amazon Inspector, Objekte nur dann zum Bucket hinzuzufügen, wenn es sich bei den Objekten um Ergebnisberichte handelt, und nur, wenn diese Berichte von dem Konto und in der Region erstellt wurden, die in der Bedingung angegeben sind.

  Damit Amazon Inspector die angegebenen Aktionen für weitere Konten ausführen kann, fügen Sie Amazon Resource Names (ARNs) für jedes weitere Konto zu dieser Bedingung hinzu. Zum Beispiel:

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

  Die in den `aws:SourceArn` Bedingungen `aws:SourceAccount` und angegebenen Konten müssen übereinstimmen.

Beide Bedingungen verhindern, dass Amazon Inspector bei Transaktionen mit Amazon S3 als verwirrter Stellvertreter eingesetzt wird. Obwohl wir dies nicht empfehlen, können Sie diese Bedingungen aus der Bucket-Richtlinie entfernen.

9. Wenn Sie mit der Aktualisierung der Bucket-Richtlinie fertig sind, wählen Sie Änderungen speichern aus.

Schritt 3: Konfigurieren Sie eine AWS KMS key

Nachdem Sie Ihre Berechtigungen überprüft und den S3-Bucket konfiguriert haben, legen AWS KMS key Sie fest, welchen Code Amazon Inspector zur Verschlüsselung Ihres Ergebnisberichts verwenden soll. Bei dem Schlüssel muss es sich um einen vom Kunden verwalteten KMS-Schlüssel mit symmetrischer Verschlüsselung handeln. Darüber hinaus muss sich der Schlüssel in demselben AWS-Region S3-Bucket befinden, den Sie zum Speichern des Berichts konfiguriert haben.

Der Schlüssel kann ein vorhandener KMS-Schlüssel aus Ihrem eigenen Konto oder ein vorhandener KMS-Schlüssel sein, den ein anderes Konto besitzt. Wenn Sie einen neuen KMS-Schlüssel verwenden möchten, erstellen Sie den Schlüssel, bevor Sie fortfahren. Wenn Sie einen vorhandenen Schlüssel verwenden möchten, der einem anderen Konto gehört, rufen Sie den Amazon-Ressourcennamen (ARN) des Schlüssels ab. Sie müssen diesen ARN eingeben, wenn Sie Ihren Bericht aus Amazon Inspector exportieren. Informationen zum Erstellen und Überprüfen der Einstellungen für KMS-Schlüssel finden Sie unter Schlüssel verwalten im AWS Key Management Service Entwicklerhandbuch.

Nachdem Sie festgelegt haben, welchen KMS-Schlüssel Sie verwenden möchten, erteilen Sie Amazon Inspector die Erlaubnis, den Schlüssel zu verwenden. Andernfalls kann Amazon Inspector den Bericht nicht verschlüsseln und exportieren. Um Amazon Inspector die Erlaubnis zur Verwendung des Schlüssels zu erteilen, aktualisieren Sie die Schlüsselrichtlinie für den Schlüssel. Ausführliche Informationen zu wichtigen Richtlinien und zur Verwaltung des Zugriffs auf KMS-Schlüssel finden Sie unter Wichtige Richtlinien AWS KMS im AWS Key Management Service Entwicklerhandbuch.

> **Note**
>
> Das folgende Verfahren dient der Aktualisierung eines vorhandenen Schlüssels, damit Amazon Inspector ihn verwenden kann. Wenn Sie noch keinen Schlüssel haben, finden Sie weitere Informationen unter Schlüssel erstellen im AWS Key Management Service Entwicklerhandbuch.

So aktualisieren Sie die Schlüsselrichtlinie

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die AWS KMS Konsole unter https://console.aws.amazon.com/kms.
2. Klicken Sie im Navigationsbereich auf Kundenverwaltete Schlüssel.
3. Wählen Sie den KMS-Schlüssel aus, den Sie zum Verschlüsseln des Berichts verwenden möchten. Der Schlüssel muss ein symmetrischer Verschlüsselungsschlüssel (SYMMETRIC_DEFAULT) sein.
4. Wählen Sie im Tab Schlüsselrichtlinie die Option Bearbeiten aus. Wenn Sie keine Schlüsselrichtlinie mit der Schaltfläche Bearbeiten sehen, müssen Sie zuerst Zur Richtlinienansicht wechseln auswählen.
5. Kopieren Sie die folgende Beispielanweisung in Ihre Zwischenablage:

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

6. Fügen Sie im Editor für Schlüsselrichtlinien auf der AWS KMS Konsole die vorherige Anweisung in die Schlüsselrichtlinie ein, um sie der Richtlinie hinzuzufügen.

   Stellen Sie beim Hinzufügen der Anweisung sicher, dass die Syntax gültig ist. Wichtige Richtlinien verwenden das JSON-Format. Das bedeutet, dass Sie vor oder nach der Anweisung ein Komma hinzufügen müssen, je nachdem, wo Sie die Anweisung zur Richtlinie hinzufügen.

Wenn Sie die Anweisung als letzte Anweisung hinzufügen, fügen Sie hinter der schließenden Klammer für die vorherige Anweisung ein Komma hinzu. Wenn Sie sie als erste Anweisung oder zwischen zwei vorhandenen Anweisungen hinzufügen, fügen Sie hinter der schließenden Klammer für die Anweisung ein Komma ein.

7. Aktualisieren Sie die Anweisung mit den richtigen Werten für Ihre Umgebung, wobei:

   - *111122223333*ist die Konto-ID für Ihr AWS-Konto.
   - *Region*ist der, AWS-Region in dem Sie Amazon Inspector erlauben möchten, Berichte mit dem Schlüssel zu verschlüsseln. Zum Beispiel `us-east-1` für die Region USA Ost (Nord-Virginia).

   > **Note**
   >
   > Wenn Sie Amazon Inspector in einem manuell aktivierten System verwenden AWS-Region, fügen Sie dem Wert für das `Service` Feld auch den entsprechenden Regionalcode hinzu. Wenn Sie beispielsweise Amazon Inspector in der Region Naher Osten (Bahrain) verwenden, `inspector2.amazonaws.com` ersetzen Sie es durch`inspector2.me-south-1.amazonaws.com`.

   Wie die Beispielanweisung für die Bucket-Richtlinie im vorherigen Schritt verwenden die `Condition` Felder in diesem Beispiel zwei globale IAM-Bedingungsschlüssel:

   - aws: SourceAccount — Diese Bedingung ermöglicht es Amazon Inspector, die angegebenen Aktionen nur für Ihr Konto durchzuführen. Insbesondere bestimmt sie, welches Konto die angegebenen Aktionen für die in der `aws:SourceArn` Bedingung angegebenen Ressourcen und Aktionen ausführen kann.

     Damit Amazon Inspector die angegebenen Aktionen für weitere Konten ausführen kann, fügen Sie dieser Bedingung die Konto-ID für jedes weitere Konto hinzu. Zum Beispiel:

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

   - aws: SourceArn — Diese Bedingung verhindert, dass andere AWS-Services die angegebenen Aktionen ausführen. Außerdem wird verhindert, dass Amazon Inspector den Schlüssel verwendet, während andere Aktionen für Ihr Konto ausgeführt werden. Mit anderen Worten, es ermöglicht Amazon Inspector, S3-Objekte nur dann mit dem Schlüssel zu verschlüsseln, wenn

es sich bei den Objekten um Ergebnisberichte handelt, und nur wenn diese Berichte von dem Konto und in der Region erstellt wurden, die in der Bedingung angegeben sind.

Damit Amazon Inspector die angegebenen Aktionen für weitere Konten ausführen kann, fügen Sie ARNs für jedes weitere Konto diese Bedingung hinzu. Zum Beispiel:

```
"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/*"
]
```

Die in den `aws:SourceArn` Bedingungen `aws:SourceAccount` und angegebenen Konten müssen übereinstimmen.

Diese Bedingungen verhindern, dass Amazon Inspector bei Transaktionen mit als verwirrter Stellvertreter eingesetzt wird AWS KMS. Wir empfehlen dies zwar nicht, Sie können diese Bedingungen jedoch aus der Erklärung entfernen.

8. Wenn Sie mit der Aktualisierung der wichtigsten Richtlinie fertig sind, wählen Sie Änderungen speichern.

Schritt 4: Konfigurieren und exportieren Sie einen Ergebnisbericht

> **Note**
>
> Sie können jeweils nur einen Ergebnisbericht exportieren. Wenn gerade ein Export ausgeführt wird, müssen Sie warten, bis der Export abgeschlossen ist, bevor Sie einen weiteren Ergebnisbericht exportieren.

Nachdem Sie Ihre Berechtigungen überprüft und Ressourcen zum Verschlüsseln und Speichern Ihres Ergebnisberichts konfiguriert haben, können Sie den Bericht konfigurieren und exportieren.

Um einen Ergebnisbericht zu konfigurieren und zu exportieren

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.

2. Wählen Sie im Navigationsbereich unter Ergebnisse die Option Alle Ergebnisse aus.
3. (Optional) Fügen Sie mithilfe der Filterleiste über der Tabelle Ergebnisse Filterkriterien hinzu, die angeben, welche Ergebnisse in den Bericht aufgenommen werden sollen. Wenn Sie Kriterien hinzufügen, aktualisiert Amazon Inspector die Tabelle, sodass sie nur die Ergebnisse enthält, die den Kriterien entsprechen. Die Tabelle bietet eine Vorschau der Daten, die Ihr Bericht enthalten wird.

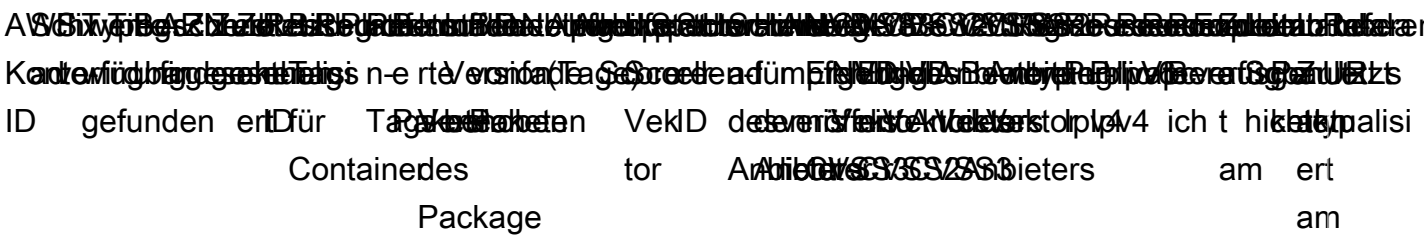
   > **Note**
   >
   > Wir empfehlen, dass Sie Filterkriterien hinzufügen. Wenn Sie dies nicht tun, enthält der Bericht Daten für alle Ihre aktuellen Ergebnisse AWS-Region , die den Status Aktiv haben. Wenn Sie der Amazon Inspector-Administrator für eine Organisation sind, umfasst dies Ergebnisdaten für alle Mitgliedskonten in Ihrer Organisation.
   > Wenn ein Bericht Daten für alle oder viele Ergebnisse enthält, kann es sehr lange dauern, den Bericht zu erstellen und zu exportieren, und Sie können jeweils nur einen Bericht exportieren.

4. Wählen Sie Ergebnisse exportieren aus.
5. Geben Sie im Abschnitt Exporteinstellungen für den Exportdateityp ein Dateiformat für den Bericht an:

   - Um eine JavaScript Objektnotationsdatei (.json) zu erstellen, die die Daten enthält, wählen Sie JSON aus.

     Wenn Sie die JSON-Option wählen, enthält der Bericht alle Felder für jedes Ergebnis. Eine Liste möglicher JSON-Felder finden Sie unter Finding data type in der Amazon Inspector API-Referenz.
   - Um eine Datei mit kommagetrennten Werten (.csv) zu erstellen, die die Daten enthält, wählen Sie CSV.

     Wenn Sie die CSV-Option wählen, enthält der Bericht nur eine Teilmenge der Felder für jedes Ergebnis, d. h. ungefähr 45 Felder, die die wichtigsten Attribute eines Ergebnisses angeben. Zu den Feldern gehören: Befundtyp, Titel, Schweregrad, Status, Beschreibung, Zuerst gesehen, Zuletzt gesehen, Korrektur verfügbar, AWS Konto-ID, Ressourcen-ID, Ressourcen-Tags und Problembehebung. Diese Felder ergänzen die Felder, in denen Bewertungsdetails und Referenzen URLs für jedes Ergebnis erfasst werden. Im Folgenden finden Sie ein Beispiel für die CSV-Header in einem Ergebnisbericht:



6. Geben Sie unter Exportort für S3-URI den S3-Bucket an, in dem Sie den Bericht speichern möchten:

   - Um den Bericht in einem Bucket zu speichern, der Ihrem Konto gehört, wählen Sie Browse S3 aus. Amazon Inspector zeigt eine Tabelle der S3-Buckets für Ihr Konto an. Wählen Sie die Zeile für den gewünschten Bucket aus und klicken Sie dann auf Auswählen.

     > **Tip**
     >
     > Um auch ein Amazon S3 S3-Pfadpräfix für den Bericht anzugeben, fügen Sie einen Schrägstrich (/) und das Präfix an den Wert im Feld S3-URI an. Amazon Inspector fügt dann das Präfix hinzu, wenn der Bericht dem Bucket hinzugefügt wird, und Amazon S3 generiert den durch das Präfix angegebenen Pfad.
     > Wenn Sie beispielsweise Ihre AWS-Konto ID als Präfix verwenden möchten und Ihre Konto-ID 111122223333 lautet, fügen Sie sie an den Wert im Feld **`/111122223333`** S3-URI an.
     > Ein Präfix ähnelt einem Verzeichnispfad innerhalb eines S3-Buckets. Es ermöglicht Ihnen, ähnliche Objekte in einem Bucket zu gruppieren, ähnlich wie Sie ähnliche Dateien zusammen in einem Ordner auf einem Dateisystem speichern könnten. Weitere Informationen finden Sie unter Organisieren von Objekten in der Amazon S3 S3-Konsole mithilfe von Ordnern im Amazon Simple Storage Service-Benutzerhandbuch.

   - Um den Bericht in einem Bucket zu speichern, der einem anderen Konto gehört, geben Sie den URI für den Bucket ein — zum Beispiel **`s3://DOC-EXAMPLE_BUCKET`**, wobei DOC-EXAMPLE_BUCKET der Name des Buckets ist. Der Bucket-Besitzer kann diese Informationen für Sie in den Eigenschaften des Buckets finden.

7. Geben Sie für den KMS-Schlüssel den an AWS KMS key , den Sie zum Verschlüsseln des Berichts verwenden möchten:

- Um einen Schlüssel aus Ihrem eigenen Konto zu verwenden, wählen Sie den Schlüssel aus der Liste aus. In der Liste werden vom Kunden verwaltete KMS-Schlüssel mit symmetrischer Verschlüsselung für Ihr Konto angezeigt.
- Um einen Schlüssel zu verwenden, der einem anderen Konto gehört, geben Sie den Amazon-Ressourcennamen (ARN) des Schlüssels ein. Der Schlüsselinhaber kann diese Informationen für Sie in den Eigenschaften des Schlüssels finden. Weitere Informationen finden Sie unter Suchen der Schlüssel-ID und des Schlüssel-ARN im AWS Key Management Service Entwicklerhandbuch.

8. Wählen Sie Export aus.

Amazon Inspector generiert den Ergebnisbericht, verschlüsselt ihn mit dem von Ihnen angegebenen KMS-Schlüssel und fügt ihn dem von Ihnen angegebenen S3-Bucket hinzu. Abhängig von der Anzahl der Ergebnisse, die Sie in den Bericht aufnehmen möchten, kann dieser Vorgang mehrere Minuten oder Stunden dauern. Wenn der Export abgeschlossen ist, zeigt Amazon Inspector eine Meldung an, dass Ihr Ergebnisbericht erfolgreich exportiert wurde. Wählen Sie optional Bericht anzeigen in der Nachricht, um zu dem Bericht in Amazon S3 zu navigieren.

Beachten Sie, dass Sie jeweils nur einen Bericht exportieren können. Wenn gerade ein Export ausgeführt wird, warten Sie, bis der Export abgeschlossen ist, bevor Sie versuchen, einen weiteren Bericht zu exportieren.

Beheben Sie Exportfehler

Wenn beim Versuch, einen Ergebnisbericht zu exportieren, ein Fehler auftritt, zeigt Amazon Inspector eine Meldung an, in der der Fehler beschrieben wird. Sie können die Informationen in diesem Thema als Leitfaden verwenden, um mögliche Ursachen und Lösungen für den Fehler zu ermitteln.

Stellen Sie beispielsweise sicher, dass sich der S3-Bucket im aktuellen Bucket befindet AWS-Region und die Bucket-Richtlinie Amazon Inspector erlaubt, Objekte zum Bucket hinzuzufügen. Stellen Sie außerdem sicher, dass der in der aktuellen Region aktiviert AWS KMS key ist, und stellen Sie sicher, dass die Schlüsselrichtlinie Amazon Inspector die Verwendung des Schlüssels ermöglicht.

Nachdem Sie den Fehler behoben haben, versuchen Sie erneut, den Bericht zu exportieren.

Der Fehler kann nicht mehrere Berichte haben

Wenn Sie versuchen, einen Bericht zu erstellen, Amazon Inspector jedoch bereits einen Bericht generiert, erhalten Sie eine Fehlermeldung mit der Angabe Grund: Es können nicht mehrere Berichte

in Bearbeitung sein. Dieser Fehler tritt auf, weil Amazon Inspector jeweils nur einen Bericht für ein Konto erstellen kann.

Um den Fehler zu beheben, können Sie warten, bis der andere Bericht abgeschlossen ist, oder ihn stornieren, bevor Sie einen neuen Bericht anfordern.

Sie können den Status eines Berichts überprüfen, indem Sie den GetFindingsReportStatusVorgang verwenden. Dieser Vorgang gibt die Berichts-ID jedes Berichts zurück, der gerade generiert wird.

Bei Bedarf können Sie mithilfe der `GetFindingsReportStatus` Operation die vom Vorgang angegebene Berichts-ID verwenden, um einen Export abzubrechen, der CancelFindingsReportgerade ausgeführt wird.

Mit Amazon benutzerdefinierte Antworten auf Ergebnisse von Amazon Inspector erstellen EventBridge

Amazon Inspector erstellt in Amazon ein Ereignis EventBridge für neu generierte Ergebnisse und aggregierte Ergebnisse. Amazon Inspector erstellt auch ein Ereignis für alle Änderungen am Status eines Ergebnisses. Das bedeutet, dass Amazon Inspector ein neues Ereignis für ein Ergebnis erstellt, wenn Sie Aktionen wie den Neustart einer Ressource oder das Ändern von mit einer Ressource verknüpften Tags ergreifen. Wenn Amazon Inspector ein neues Ereignis für ein aktualisiertes Ergebnis erstellt, `id` bleibt das Ergebnis gleich.

> **Note**
>
> Wenn es sich bei Ihrem Konto um ein delegiertes Administratorkonto von Amazon Inspector handelt, EventBridge veröffentlicht Ereignisse auf Ihrem Konto und dem Mitgliedskonto, von dem die Ereignisse ihren Ursprung haben.

Wenn Sie EventBridge Ereignisse mit Amazon Inspector verwenden, können Sie Aufgaben automatisieren, um auf Sicherheitsprobleme zu reagieren, die Ihre Ergebnisse aufdecken. Um Benachrichtigungen über Ergebnisse von Amazon Inspector auf der Grundlage von EventBridge Ereignissen zu erhalten, müssen Sie eine EventBridge Regel erstellen und ein Ziel für Amazon Inspector angeben. Die EventBridge Regel ermöglicht EventBridge das Senden von Benachrichtigungen über Ergebnisse von Amazon Inspector, und das Ziel gibt an, wohin die Benachrichtigungen gesendet werden sollen.

Amazon Inspector sendet Ereignisse an den Standardereignisbus in dem Bereich, in AWS-Region dem Sie Amazon Inspector derzeit verwenden. Das bedeutet, dass Sie für jeden Fall, in AWS-Region dem Sie Amazon Inspector aktiviert und Amazon Inspector für den Empfang von EventBridge Ereignissen konfiguriert haben, Ereignisregeln konfigurieren müssen. Amazon Inspector sendet Ereignisse nach bestem Wissen und Gewissen aus.

Dieser Abschnitt enthält ein Beispiel für ein Ereignisschema und beschreibt, wie Sie eine EventBridge Regel erstellen.

Ereignisschema

Im Folgenden finden Sie ein Beispiel für das Amazon Inspector Inspector-Ereignisformat für ein EC2 Findereignis. Ein Beispiel für ein Schema anderer Such- und Ereignistypen finden Sie unterEventBridge Schema.


```
{
    "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"
    }
}
```


Eine EventBridge Regel erstellen, um Sie über Ergebnisse von Amazon Inspector zu informieren

Um die Sichtbarkeit der Ergebnisse von Amazon Inspector EventBridge zu erhöhen, können Sie automatische Suchwarnungen einrichten, die an einen Messaging-Hub gesendet werden. In diesem Thema erfahren Sie, wie Sie Benachrichtigungen `CRITICAL` und `HIGH` Schweregrade per E-Mail, Slack oder Amazon Chime senden. Sie erfahren, wie Sie ein Amazon Simple Notification Service-Thema einrichten und dieses Thema dann mit einer EventBridge Ereignisregel verbinden.

Schritt 1. Ein Amazon SNS SNS-Thema und einen Endpunkt einrichten

Um automatische Benachrichtigungen einzurichten, müssen Sie zunächst ein Thema in Amazon Simple Notification Service einrichten und einen Endpunkt hinzufügen. Weitere Informationen finden Sie im SNS-Handbuch.

Dieses Verfahren legt fest, wohin Sie Amazon Inspector Inspector-Ergebnisdaten senden möchten. Das SNS-Thema kann während oder nach der Erstellung der EventBridge Ereignisregel zu einer Ereignisregel hinzugefügt werden.

Email setup

Erstellen eines SNS-Themas

1. Melden Sie sich bei der Amazon SNS SNS-Konsole unter https://console.aws.amazon.com/sns/v3/home an.
2. Wählen Sie im Navigationsbereich Themen und dann Thema erstellen aus.
3. Wählen Sie im Abschnitt Thema erstellen die Option Standard aus. Geben Sie als Nächstes einen Themennamen ein, z. **`Inspector_to_Email`** B. Weitere Angaben sind optional.
4. Wählen Sie Create Topic (Thema erstellen) aus. Dadurch wird ein neues Fenster mit Details zu Ihrem neuen Thema geöffnet.
5. Wählen Sie im Abschnitt Abonnements die Option Abonnement erstellen aus.
6. a. Wählen Sie im Menü Protocol (Protokoll) die Option Email (E-Mail) aus.
   b. Geben Sie im Feld Endpoint die E-Mail-Adresse ein, an die Sie Benachrichtigungen erhalten möchten.

   > **Note**
   >
   > Nach der Erstellung des Abonnements müssen Sie Ihr Abonnement über Ihren E-Mail-Client bestätigen.

   c. Wählen Sie Create subscription (Abonnement erstellen) aus.
7. Suchen Sie in Ihrem Posteingang nach einer Abonnementnachricht und wählen Sie Abonnement bestätigen.

Slack setup

Erstellen eines SNS-Themas

1. Melden Sie sich bei der Amazon SNS SNS-Konsole unter https://console.aws.amazon.com/sns/v3/home an.
2. Wählen Sie im Navigationsbereich Themen und dann Thema erstellen aus.
3. Wählen Sie im Abschnitt Thema erstellen die Option Standard aus. Geben Sie als Nächstes einen Themennamen ein, z. **`Inspector_to_Slack`** B. Weitere Angaben sind optional. Wählen Sie Thema erstellen, um die Endpunkterstellung abzuschließen.

Konfiguration eines Amazon Q Developer im Client für Chat-Anwendungen

1. Navigieren Sie zur Amazon Q Developer in Chat-Anwendungskonsole unterhttps://console.aws.amazon.com/chatbot/.
2. Wählen Sie im Bereich Konfigurierte Clients die Option Neuen Client konfigurieren aus.
3. Wähle Slack und dann zur Bestätigung Configure.

> **Note**
>
> Wenn du dich für Slack entscheidest, musst du bestätigen, dass Amazon Q Developer in Chat-Anwendungen auf deinen Kanal zugreifen darf, indem du Zulassen auswählst.

4. Wählen Sie Neuen Kanal konfigurieren aus, um den Bereich mit den Konfigurationsdetails zu öffnen.
   a. Geben Sie einen Namen für den Kanal ein.
   b. Wählen Sie für den Slack-Kanal den Kanal aus, den Sie verwenden möchten.
   c. Kopiere in Slack die Kanal-ID des privaten Channels, indem du mit der rechten Maustaste auf den Kanalnamen klickst und Link kopieren auswählst.
   d. Fügen Sie im Fenster Amazon Q Developer in Chat-Anwendungen die Kanal-ID, die Sie aus Slack kopiert haben, in das Feld Private Channel-ID ein. AWS Management Console
   e. Wählen Sie unter Berechtigungen aus, ob Sie mithilfe einer Vorlage eine IAM-Rolle erstellen möchten, falls Sie noch keine Rolle haben.
   f. Wählen Sie für Richtlinienvorlagen die Option Benachrichtigungsberechtigungen aus. Dies ist die IAM-Richtlinienvorlage für Amazon Q Developer in Chat-Anwendungen. Diese Richtlinie bietet die erforderlichen Lese- und Listenberechtigungen für CloudWatch Alarme, Ereignisse und Protokolle sowie für Amazon SNS SNS-Themen.
   g. Wählen Sie für Channel-Guardrail-Richtlinien die Option 2. AmazonInspector ReadOnlyAccess
   h. Wählen Sie die Region aus, in der Sie zuvor Ihr SNS-Thema erstellt haben, und wählen Sie dann das Amazon SNS SNS-Thema aus, das Sie erstellt haben, um Benachrichtigungen an den Slack-Kanal zu senden.
5. Wählen Sie Konfigurieren.

Amazon Chime setup

Erstellen eines SNS-Themas

1. Melden Sie sich bei der Amazon SNS SNS-Konsole unter https://console.aws.amazon.com/sns/v3/home an.
2. Wählen Sie im Navigationsbereich Themen und dann Thema erstellen aus.
3. Wählen Sie im Abschnitt Thema erstellen die Option Standard aus. Geben Sie als Nächstes einen Themennamen ein, z. **`Inspector_to_Chime`** B. Weitere Angaben sind optional. Wählen Sie Thema erstellen, um den Vorgang abzuschließen.

Konfiguration eines Amazon Q Developer im Client für Chat-Anwendungen

1. Navigieren Sie zur Amazon Q Developer in Chat-Anwendungskonsole unterhttps://console.aws.amazon.com/chatbot/.
2. Wählen Sie im Bereich Konfigurierte Clients die Option Neuen Client konfigurieren.
3. Wählen Sie Chime und anschließend zur Bestätigung Configure.
4. Geben Sie im Bereich mit den Konfigurationsdetails einen Namen für den Kanal ein.
5. Öffnen Sie in Amazon Chime den gewünschten Chatroom.
   a. Wählen Sie das Zahnradsymbol rechts oben und danach Manage webhooks and bots aus.
   b. Wählen Sie URL kopieren, um die Webhook-URL in Ihre Zwischenablage zu kopieren.
6. Fügen Sie im Fenster Amazon Q Developer in Chat-Anwendungen die URL ein, die Sie kopiert haben, in das Feld Webhook-URL. AWS Management Console
7. Wählen Sie unter Berechtigungen aus, ob Sie eine IAM-Rolle mithilfe einer Vorlage erstellen möchten, falls Sie noch keine Rolle haben.
8. Wählen Sie für Richtlinienvorlagen die Option Benachrichtigungsberechtigungen aus. Dies ist die IAM-Richtlinienvorlage für Amazon Q Developer in Chat-Anwendungen. Es bietet die erforderlichen Lese- und Listenberechtigungen für CloudWatch Alarme, Ereignisse und Protokolle sowie für Amazon SNS SNS-Themen.
9. Wählen Sie die Region aus, in der Sie zuvor Ihr SNS-Thema erstellt haben, und wählen Sie dann das Amazon SNS SNS-Thema aus, das Sie erstellt haben, um Benachrichtigungen an den Amazon Chime Chime-Raum zu senden.
10. Wählen Sie Konfigurieren.

Schritt 2. Eine EventBridge Regel für Amazon Inspector Inspector-Ergebnisse erstellen

1. Melden Sie sich mit Ihren Zugangsdaten an.
2. Öffnen Sie die EventBridge Amazon-Konsole unter https://console.aws.amazon.com/events/.
3. Wählen Sie im Navigationsbereich Regeln und dann Regel erstellen aus.
4. Geben Sie einen Namen und optional eine Beschreibung für Ihre Regel ein.
5. Wählen Sie Regel mit einem Ereignismuster und dann Weiter aus.
6. Wählen Sie im Bereich „Ereignismuster“ die Option Benutzerdefinierte Muster (JSON-Editor) aus.
7. Fügen Sie den folgenden JSON-Code in den Editor ein.

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

> **Note**
>
> Dieses Muster sendet Benachrichtigungen für alle von Amazon Inspector festgestellten aktiven `CRITICAL` oder `HIGH` schwerwiegenden Ergebnisse.

   Wählen Sie Weiter, wenn Sie mit der Eingabe des Ereignismusters fertig sind.
8. Wählen Sie auf der Seite Ziele auswählen aus AWS-Service. Wählen Sie dann für Zieltyp auswählen die Option SNS-Thema aus.
9. Wählen Sie unter Thema den Namen des SNS-Themas aus, das Sie in Schritt 1 erstellt haben. Wählen Sie anschließend Weiter.
10. Fügen Sie bei Bedarf optionale Tags hinzu und wählen Sie Weiter.
11. Überprüfen Sie Ihre Regel und wählen Sie dann Regel erstellen aus.

EventBridge für Amazon Inspector Inspector-Umgebungen mit mehreren Konten

Wenn Sie ein delegierter Administrator von Amazon Inspector sind, werden auf Ihrem Konto EventBridge Regeln angezeigt, die auf den entsprechenden Ergebnissen Ihrer Mitgliedskonten basieren. Wenn Sie EventBridge in Ihrem Administratorkonto Benachrichtigungen über Ergebnisse einrichten, wie im vorherigen Abschnitt beschrieben, erhalten Sie Benachrichtigungen über mehrere Konten. Mit anderen Worten, Sie werden über Ergebnisse und Ereignisse informiert, die von Ihren Mitgliedskonten generiert wurden, zusätzlich zu den Ergebnissen und Ereignissen, die von Ihrem eigenen Konto generiert wurden.

Sie können die JSON-Details `accountId` aus den Ergebnissen verwenden, um das Mitgliedskonto zu identifizieren, von dem das Amazon Inspector Inspector-Ergebnis stammt.

Arbeiten mit dem Dashboard in Amazon Inspector

Das Dashboard bietet eine Momentaufnahme der aggregierten Statistiken für Ressourcen, die Amazon Inspector scannt. Verwenden Sie das Dashboard, um mehr über die Abdeckung Ihrer Umgebung und wichtige Ergebnisse zu erfahren.

> **Note**
>
> Wenn es sich bei Ihrem Konto um das delegierte Administratorkonto für eine Organisation handelt, werden im Dashboard Informationen zu Ihrem Konto und allen anderen Konten in der Organisation angezeigt.

In diesem Thema wird beschrieben, wie Sie das Dashboard anzeigen und die Komponenten verstehen, aus denen das Dashboard besteht.

Themen

Das Dashboard anzeigen

Das Dashboard bietet einen Überblick über den Versicherungsschutz für Ihre Umgebung und wichtige Ergebnisse. Das Dashboard aktualisiert die Daten automatisch alle fünf Minuten. Sie können Daten manuell aktualisieren, indem Sie das Aktualisierungssymbol in der oberen rechten Ecke des Bildschirms auswählen. Sie können unterstützende Daten zu einem Artikel anzeigen, indem Sie den Artikel auswählen.

> **Note**
>
> Wenn es sich bei Ihrem Konto um das delegierte Administratorkonto für eine Organisation handelt, können Sie aggregierte Statistiken für ein Mitgliedskonto anzeigen, indem Sie die Mitgliedskonto-ID in das Feld Konto eingeben.

So zeigen Sie das Dashboard an:

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Dashboard aus.

Dashboard-Komponenten verstehen und Daten interpretieren

Jeder Abschnitt des Dashboards bietet Einblicke in wichtige Kennzahlen und Ergebnisdaten, sodass Sie sich ein Bild von der aktuellen Sicherheitsanfälligkeit Ihrer AWS Ressourcen machen können AWS-Region.

Abdeckung der Umwelt

Der Abschnitt Umweltberichterstattung enthält Statistiken über die von Amazon Inspector gescannten Ressourcen. In diesem Abschnitt können Sie die Anzahl und den Prozentsatz der EC2 Amazon-Instances, Amazon ECR-Bilder und AWS Lambda Funktionen sehen, die von Amazon Inspector gescannt wurden. Wenn Sie AWS Organizations als delegierter Administrator von Amazon Inspector mehrere Konten verwalten, werden Ihnen auch die Gesamtzahl der Organisationskonten, die Anzahl mit aktiviertem Amazon Inspector und der daraus resultierende Deckungsgrad für die Organisation angezeigt. In diesem Abschnitt können Sie auch feststellen, welche Ressourcen nicht von Amazon Inspector abgedeckt werden. Diese Ressourcen können Sicherheitslücken enthalten, die ausgenutzt werden könnten, um Ihr Unternehmen zu gefährden. Weitere Details finden Sie unter Bewertung der Abdeckung Ihrer AWS Umgebung durch Amazon Inspector.

Wenn Sie eine Deckungsgruppe auswählen, gelangen Sie zur Kontoverwaltungsseite für die von Ihnen gewählte Gruppierung. Auf der Kontoverwaltungsseite finden Sie Details darüber, welche Konten, EC2 Amazon-Instances und Amazon ECR-Repositorys von Amazon Inspector abgedeckt werden.

Die folgenden Deckungsgruppen sind verfügbar:

- Account
- Instances
- Container-Repositorien
- Container-Images
- Lambda

Kritische Ergebnisse

Der Abschnitt Kritische Ergebnisse enthält eine Anzahl der kritischen Sicherheitslücken in Ihrer Umgebung und eine Gesamtzahl aller Ergebnisse in Ihrer Umgebung. In diesem Abschnitt werden die Zahlen pro Ressource und Bewertungstyp angezeigt. Weitere Informationen zu kritischen Ergebnissen und dazu, wie Amazon Inspector die Kritikalität bestimmt, finden Sie unterDie Ergebnisse von Amazon Inspector verstehen.

Wenn Sie eine kritische Ergebnisgruppe auswählen, gelangen Sie zur Seite Alle Ergebnisse und wendet automatisch Filter an, um alle kritischen Ergebnisse anzuzeigen, die der von Ihnen ausgewählten Gruppierung entsprechen.

Die folgenden Gruppen kritischer Ergebnisse sind verfügbar:

- Ergebnisse des Amazon Inspector Inspector-Codescans
- Ergebnisse der EC2 Amazon-Instance
- Ergebnisse des Amazon ECR-Container-Images
- Ergebnisse der Lambda-Funktion

Risikobasierte Abhilfemaßnahmen

Im Abschnitt Risikobasierte Problembehebungen werden die fünf Softwarepakete mit den wichtigsten Sicherheitslücken aufgeführt, von denen die meisten Ressourcen in Ihrer Umgebung betroffen sind. Durch die Behebung dieser Pakete kann die Anzahl kritischer Risiken für Ihre Umgebung erheblich reduziert werden. Wählen Sie den Namen des Softwarepakets, um die zugehörigen Sicherheitslücken und die betroffenen Ressourcen zu sehen.

Konten mit den wichtigsten Ergebnissen

Im Abschnitt Konten mit den kritischsten Ergebnissen werden die fünf AWS Konten in Ihrer Umgebung mit den kritischsten Ergebnissen sowie die Gesamtzahl der Ergebnisse für dieses Konto angezeigt. Dieser Abschnitt ist nur vom delegierten Administratorkonto aus sichtbar, wenn Amazon Inspector für das Scannen mehrerer Konten mit konfiguriert ist. AWS Organizations Diese Ansicht hilft delegierten Administratoren zu verstehen, welche Konten innerhalb des Unternehmens möglicherweise am stärksten gefährdet sind.

Wählen Sie Konto-ID, um weitere Informationen über das betroffene Mitgliedskonto zu erhalten.

Amazon ECR-Repositorys mit den wichtigsten Ergebnissen

Im Abschnitt Elastic Container Registry (ECR) -Repositorien mit den wichtigsten Ergebnissen werden die fünf Amazon ECR-Repositorys in Ihrer Umgebung mit den kritischsten Container-

Image-Ergebnissen angezeigt. In der Ansicht werden der Repository-Name, die AWS Konto-ID, das Erstellungsdatum des Repositorys, die Anzahl der kritischen Sicherheitslücken und die Gesamtzahl der Sicherheitslücken angezeigt. Anhand dieser Ansicht können Sie ermitteln, welche Repositorys möglicherweise am stärksten gefährdet sind.

Wählen Sie den Repository-Namen, um weitere Informationen über das betroffene Repository zu erhalten.

Container-Images mit den wichtigsten Ergebnissen

Im Abschnitt Container-Images mit den kritischsten Ergebnissen werden die fünf Container-Images in Ihrer Umgebung mit den kritischsten Ergebnissen angezeigt. In der Ansicht werden Image-Tag-Daten, Repository-Name, Image-Digest, AWS Konto-ID, Anzahl kritischer Sicherheitslücken und Gesamtzahl der Sicherheitslücken angezeigt. Anhand dieser Ansicht können Anwendungsbesitzer erkennen, welche Container-Images möglicherweise neu erstellt und neu gestartet werden müssen.

Wählen Sie Container-Image, um weitere Informationen über das betroffene Container-Image zu erhalten.

Fälle mit den kritischsten Ergebnissen

Der Abschnitt Instances mit den kritischsten Ergebnissen zeigt die fünf EC2 Amazon-Instances mit den kritischsten Ergebnissen. Die Ansicht zeigt die Instanz-ID, die AWS Konto-ID, die Amazon Machine Image (AMI) -ID, die Anzahl kritischer Sicherheitslücken und die Gesamtzahl der Sicherheitslücken. Diese Ansicht hilft Infrastrukturbesitzern dabei, zu erkennen, welche Instances möglicherweise gepatcht werden müssen.

Wählen Sie Instance-ID, um weitere Informationen über die betroffene EC2 Amazon-Instance zu erhalten.

Amazon Machine Images (AMI) mit den wichtigsten Ergebnissen

Im Abschnitt Amazon Machine Images (AMIs) mit den kritischsten Ergebnissen werden die fünf wichtigsten Ergebnisse AMIs in Ihrer Umgebung angezeigt. Die Ansicht zeigt die AMI-ID, die AWS Konto-ID, die Anzahl der betroffenen EC2 Instances, die in der Umgebung ausgeführt werden, das AMI-Erstellungsdatum, die Betriebssystemplattform des AMI, die Anzahl der kritischen Sicherheitslücken und die Gesamtzahl der Sicherheitslücken. Diese Ansicht hilft Infrastrukturbesitzern dabei, herauszufinden, welche Infrastruktur AMIs möglicherweise neu aufgebaut werden muss.

Wählen Sie Betroffene Instances aus, um weitere Informationen zu den Instances zu erhalten, die über das betroffene AMI gestartet wurden.

AWS Lambda Funktionen mit den wichtigsten Ergebnissen

Im Abschnitt AWS Lambda Funktionen mit den kritischsten Ergebnissen werden die fünf wichtigsten Lambda-Funktionen in Ihrer Umgebung mit den kritischsten Ergebnissen angezeigt. Die Ansicht zeigt den Namen der Lambda-Funktion, die AWS Konto-ID, die Laufzeitumgebung, die Anzahl kritischer Sicherheitslücken, die Anzahl der schwerwiegenden Sicherheitslücken und die Gesamtzahl der Sicherheitslücken. Diese Ansicht hilft Infrastrukturbesitzern dabei, zu erkennen, welche Lambda-Funktionen möglicherweise behoben werden müssen.

Wählen Sie Funktionsname, um weitere Informationen über die betroffene AWS Lambda Funktion zu erhalten.

Amazon Inspector Inspector-Codescans mit den wichtigsten Ergebnissen

Der Abschnitt Projekte mit den kritischsten Code-Schwachstellen zeigt die fünf Projekte mit den wichtigsten Ergebnissen. Sie können ein Projekt auswählen, um Details zu den Ergebnissen anzuzeigen. Wenn Sie ein Projekt auswählen, werden Sie zu dem Repository weitergeleitet, in dem sich die Ergebnisse befinden. Auf der Registerkarte „Ergebnisse“ werden die Namen Ihrer Ergebnisse und deren Schweregrad angezeigt. Es zeigt, welche Art von Analyse zur Generierung Ihrer Ergebnisse verwendet wurde. Es zeigt auch, wie alt Ihre Ergebnisse sind und welchen Status sie haben.

Durchsuchen der Amazon Inspector Inspector-Schwachstellendatenbank

Sie können die Sicherheitslückendatenbank von Amazon Inspector nach häufigen Sicherheitslücken und Risiken (CVE) durchsuchen. Amazon Inspector verwendet Informationen aus der Schwachstellendatenbank, um Details zu einer CVE-ID zu erstellen. Sie können diese Details auf dem Bildschirm mit den CVE-Details einsehen. Amazon Inspector verfolgt und erstellt Ergebnisse für Softwareschwachstellen in der Schwachstellendatenbank. Amazon Inspector unterstützt CVEs nur Plattformen, die im Abschnitt Erkennungsplattformen des CVE-Detailbildschirms aufgeführt sind. In diesem Abschnitt wird beschrieben, wie Sie die Amazon Inspector Inspector-Schwachstellendatenbank mithilfe einer CVE-ID durchsuchen.

> **Note**
>
> Derzeit wird die CVE-Suche nicht unterstützt. Microsoft Windows

Die Schwachstellen-Datenbank durchsuchen

In diesem Abschnitt wird beschrieben, wie Sie die Schwachstellendatenbank in der Konsole und mit der Amazon Inspector API durchsuchen.

> **Note**
>
> Sie müssen Amazon Inspector in Ihrem aktuellen System aktivieren, AWS-Region bevor Sie die Schwachstellendatenbank durchsuchen können.

Console

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home
2. Wählen Sie im Navigationsbereich die Option Vulnerability database search aus.
3. Geben Sie in der Suchleiste eine CVE-ID ein und wählen Sie Suchen aus.

API

Führen Sie die Amazon Inspector SearchVulnerabilitiesInspector-API aus und geben Sie eine einzelne CVE-ID wie `filterCriteria` im folgenden Format ein:`CVE-<year>-<ID>`.

CVE-Details verstehen

In diesem Abschnitt wird beschrieben, wie die CVE-Detailseite zu interpretieren ist.

CVE-Details

Der Abschnitt mit den CVE-Details enthält die folgenden Informationen:

- Beschreibung und ID des CVE
- CVE-Schweregrad
- Bewertungen des Common Vulnerability Scoring System (CVSS) und des Exploit Prediction Scoring System (EPSS)
- Plattformen zur Erkennung

> **Note**
>
> Wenn dieses Feld leer ist, unterstützt Amazon Inspector keine Erkennung für Ihre CVE-ID.

- Aufzählung der häufigsten Schwächen (CWE)
- Datum der Erstellung und Aktualisierung durch den Anbieter

Informationen zu Sicherheitslücken

Der Bereich Vulnerability Intelligence bietet Bedrohungsdaten wie Exploit-Ziele und das Datum des letzten bekannten öffentlichen Exploits.

Es enthält auch Daten der Cybersecurity and Infrastructure Security Agency (CISA), darunter die Abhilfemaßnahmen, das Datum, an dem das CVE in den Katalog der bekannten Sicherheitslücken aufgenommen wurde, und das Datum, zu dem die CISA erwartet, dass Bundesbehörden das CVE beheben werden.

Referenzen

Der Abschnitt „Referenzen“ enthält Links zu Ressourcen mit weiteren Informationen über das CVE.

Exportieren SBOMs mit Amazon Inspector

Eine Software-Stückliste (SBOM) ist ein verschachteltes Inventar aller Open-Source-Softwarekomponenten und Softwarekomponenten von Drittanbietern in Ihrer Codebasis. Amazon Inspector stellt SBOMs einzelne Ressourcen in Ihrer Umgebung bereit. Sie können die Amazon Inspector Inspector-Konsole oder die Amazon Inspector Inspector-API verwenden, um SBOMs für Ihre Ressourcen zu generieren. Sie können SBOMs für alle Ressourcen exportieren, die Amazon Inspector unterstützt und überwacht. Exportiert SBOMs bieten Informationen zu Ihrem Softwareangebot. Sie können den Status Ihrer Ressourcen überprüfen, indem Sie die Abdeckung Ihrer AWS Umgebung bewerten. In diesem Abschnitt wird die Konfiguration und der Export beschrieben SBOMs.

> **Note**
>
> Derzeit unterstützt Amazon Inspector den Export SBOMs für EC2 Windows-Amazon-Instances nicht.

Amazon Inspector Inspector-Formate

Amazon Inspector unterstützt den Export SBOMs in den mit CyclonedX 1.4 und SPDX 2.3 kompatiblen Formaten. Amazon Inspector exportiert SBOMs als `JSON` Dateien in den Amazon S3 S3-Bucket Ihrer Wahl.

> **Note**
>
> Exporte im SPDX-Format von Amazon Inspector sind mit Systemen kompatibel, die SPDX 2.3 verwenden, enthalten jedoch nicht das Feld Creative Commons Zero (CC0). Dies liegt daran, dass die Aufnahme dieses Felds es Benutzern ermöglichen würde, das Material weiterzuverteilen oder zu bearbeiten.

Beispiel für das CyclonedX 1.4 SBOM-Format von Amazon Inspector

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


Beispiel für das SPDX 2.3 SBOM-Format von Amazon Inspector

```
{
 "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"
}
```

Filtert für SBOMs

Beim Exportieren können SBOMs Sie Filter einbeziehen, um Berichte für bestimmte Teilmengen von Ressourcen zu erstellen. Wenn Sie keinen Filter angeben, werden die SBOMs für alle aktiven, unterstützten Ressourcen exportiert. Und wenn Sie ein delegierter Administrator sind, umfasst dies auch Ressourcen für alle Mitglieder. Die folgenden Filter sind verfügbar:

- AccountID — Dieser Filter kann SBOMs für den Export aller Ressourcen verwendet werden, die mit einer bestimmten Konto-ID verknüpft sind.
- EC2 Instanz-Tag — Dieser Filter kann SBOMs für den Export von EC2 Instanzen mit bestimmten Tags verwendet werden.

- Funktionsname — Dieser Filter kann SBOMs für den Export bestimmter Lambda-Funktionen verwendet werden.
- Bild-Tag — Dieser Filter kann SBOMs für den Export von Container-Images mit bestimmten Tags verwendet werden.
- Lambda-Funktions-Tag — Dieser Filter kann SBOMs für den Export von Lambda-Funktionen mit bestimmten Tags verwendet werden.
- Ressourcentyp — Dieser Filter kann verwendet werden, um den Ressourcentyp zu filtern: EC2 / ecr/Lambda.
- Ressourcen-ID — Dieser Filter kann verwendet werden, um eine SBOM für eine bestimmte Ressource zu exportieren.
- Repository-Name — Dieser Filter kann verwendet werden, um Container-Images in bestimmten Repositorys zu generieren SBOMs .

Konfigurieren und exportieren SBOMs

Für den Export SBOMs müssen Sie zunächst einen Amazon S3 S3-Bucket und einen AWS KMS Schlüssel konfigurieren, den Amazon Inspector verwenden darf. Sie können Filter verwenden, um SBOMs für bestimmte Teilmengen Ihrer Ressourcen zu exportieren. Um SBOMs für mehrere Konten in einer AWS Organisation zu exportieren, folgen Sie diesen Schritten, während Sie als delegierter Amazon Inspector-Administrator angemeldet sind.

Voraussetzungen

- Unterstützte Ressourcen, die aktiv von Amazon Inspector überwacht werden.
- Ein Amazon S3 S3-Bucket, der mit einer Richtlinie konfiguriert ist, die es Amazon Inspector ermöglicht, Objekte hinzuzufügen. Informationen zur Konfiguration der Richtlinie finden Sie unter Exportberechtigungen konfigurieren.
- Ein AWS KMS Schlüssel, der mit einer Richtlinie konfiguriert ist, die es Amazon Inspector ermöglicht, Ihre Berichte zu verschlüsseln. Informationen zur Konfiguration der Richtlinie finden Sie unter Einen AWS KMS Schlüssel für den Export konfigurieren.

> **Note**
>
> Wenn Sie zuvor einen Amazon S3 S3-Bucket und einen AWS KMS Schlüssel für den Ergebnisexport konfiguriert haben, können Sie denselben Bucket und Schlüssel für den SBOM-Export verwenden.

Wählen Sie Ihre bevorzugte Zugriffsmethode für den Export einer SBOM.

Console

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie mit der AWS-Region Auswahl in der oberen rechten Ecke der Seite die Region mit den Ressourcen aus, für die Sie SBOM exportieren möchten.
3. Wählen Sie im Navigationsbereich die Option Export (Exportieren) SBOMs aus.
4. (Optional) Verwenden Sie auf der SBOMsExportseite das Menü Filter hinzufügen, um eine Teilmenge von Ressourcen auszuwählen, für die Berichte erstellt werden sollen. Wenn kein Filter angegeben ist, exportiert Amazon Inspector Berichte für alle aktiven Ressourcen. Wenn Sie ein delegierter Administrator sind, umfasst dies alle aktiven Ressourcen in Ihrer Organisation.
5. Wählen Sie unter Exporteinstellung das gewünschte Format für die SBOM aus.
6. Geben Sie eine Amazon S3-URI ein oder wählen Sie Amazon S3 durchsuchen, um einen Amazon S3 S3-Standort zum Speichern der SBOM auszuwählen.
7. Geben Sie einen AWS KMS Schlüssel ein, der für Amazon Inspector konfiguriert ist, um Ihre Berichte zu verschlüsseln.

API

- Verwenden Sie den CreateSbomExportBetrieb der Amazon Inspector Inspector-API, um Ihre Ressourcen programmgesteuert zu exportieren SBOMs .

  Verwenden Sie in Ihrer Anfrage den `reportFormat` Parameter, um das SBOM-Ausgabeformat anzugeben, und wählen Sie oder. `CYCLONEDX_1_4` `SPDX_2_3` Der `s3Destination` Parameter ist erforderlich, und Sie müssen einen S3-Bucket angeben, der mit einer Richtlinie konfiguriert ist, die es Amazon Inspector ermöglicht, in diesen Bucket

zu schreiben. Verwenden Sie optional `resourceFilterCriteria` Parameter, um den Umfang des Berichts auf bestimmte Ressourcen zu beschränken.

AWS CLI

- AWS Command Line Interface Führen Sie den folgenden Befehl aus, um SBOMs für Ihre Ressourcen mit dem folgenden Befehl zu exportieren:

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

  Ersetzen Sie es in Ihrer Anfrage *FORMAT* durch das Format Ihrer Wahl, `CYCLONEDX_1_4` oder`SPDX_2_3`. Ersetzen Sie dann das *user input placeholders* für das S3-Ziel durch den Namen des S3-Buckets, in den exportiert werden soll, das Präfix, das für die Ausgabe in S3 verwendet werden soll, und den ARN für den KMS-Schlüssel, den Sie zum Verschlüsseln der Berichte verwenden.

EventBridge Amazon-Ereignisschema für Amazon Inspector-Ereignisse

Amazon EventBridge liefert einen Stream von Echtzeitdaten aus Anwendungen und anderen Anwendungen AWS-Services an Ziele wie AWS Lambda Funktionen, Amazon Simple Notification Service-Themen und Datenstreams in Amazon Kinesis Data Streams. Um die Integration mit anderen Anwendungen, Diensten und Systemen zu unterstützen, veröffentlicht Amazon Inspector die Ergebnisse automatisch EventBridge als Ereignisse. Sie können Amazon Inspector verwenden, um Ereignisse für Ergebnisse, Berichterstattung und Scans zu veröffentlichen. Dieser Abschnitt enthält Beispielschemas für EventBridge Ereignisse.

Themen

EventBridge Amazon-Basisschema für Amazon Inspector

Das Folgende ist ein Beispiel für das grundlegende Schema für ein EventBridge Ereignis für Amazon Inspector. Die Veranstaltungsdetails unterscheiden sich je nach Art des Ereignisses.

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

```
}
```

Beispiel für das Auffinden von Ereignissen in Amazon Inspector

Im Folgenden finden Sie Beispiele für das Schema für ein EventBridge Ereignis mit Ergebnissen von Amazon Inspector. Findungsereignisse werden ausgelöst, wenn Amazon Inspector eine Softwareschwachstelle oder ein Netzwerkproblem in einer Ihrer Ressourcen identifiziert. Eine Anleitung zum Erstellen von Benachrichtigungen als Reaktion auf diese Art von Ereignis finden Sie unterMit Amazon benutzerdefinierte Antworten auf Ergebnisse von Amazon Inspector erstellen EventBridge.

Die folgenden Felder identifizieren ein Findereignis:

- `detail-type`ist auf gesetzt`Inspector2 Finding`.
- `detail`beschreibt den Befund.
- `detail.resources.tags`ist der Ort, an dem Schlüsselwertdaten gespeichert werden.

Sie können die Registerkarten filtern, um nach Ereignisschemas für verschiedene Ressourcen und Suchtypen zu suchen.

Amazon EC2 package vulnerability finding

```
{
    "version": "0",
    "id": "4d621919-f1f4-4201-a0e2-37e4e330ff51",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-09-04T17:00:36Z",
    "region": "eu-central-1",
    "resources": [
        "i-12345678901234567"
    ],
    "detail": {
        "awsAccountId": "123456789012",
        "description": "In snapd versions prior to 2.62, snapd failed to properly
 check the destination of symbolic links when extracting a snap. The snap format
 is a squashfs file-system image and so can contain symbolic links and other file
 types. Various file entries within the snap squashfs image (such as icons and
```

```
 desktop files etc) are directly read by snapd when it is extracted. An attacker who
 could convince a user to install a malicious snap which contained symbolic links
 at these paths could then cause snapd to write out the contents of the symbolic
 link destination into a world-readable directory. This in-turn could allow an
 unprivileged user to gain access to privileged information.",
        "epss": {
            "score": 0.00043
        },
        "exploitAvailable": "NO",
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Wed Sep 04 16:59:44.356 UTC 2024",
        "fixAvailable": "YES",
        "inspectorScore": 4.8,
        "inspectorScoreDetails": {
            "adjustedCvss": {
                "adjustments": [],
                "cvssSource": "UBUNTU_CVE",
                "score": 4.8,
                "scoreSource": "UBUNTU_CVE",
                "scoringVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:L",
                "version": "3.1"
            }
        },
        "lastObservedAt": "Wed Sep 04 16:59:44.476 UTC 2024",
        "packageVulnerabilityDetails": {
            "cvss": [
                {
                    "baseScore": 4.8,
                    "scoringVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:L",
                    "source": "UBUNTU_CVE",
                    "version": "3.1"
                },
                {
                    "baseScore": 7.3,
                    "scoringVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:R/S:U/C:H/I:H/A:H",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://www.cve.org/CVERecord?id=CVE-2024-29069",
                "https://ubuntu.com/security/notices/USN-6940-1"
            ],
```

```
            "relatedVulnerabilities": [
                "USN-6940-1"
            ],
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2024/
CVE-2024-29069.html",
            "vendorCreatedAt": "Thu Jul 25 20:15:00.000 UTC 2024",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2024-29069",
            "vulnerablePackages": [
                {
                    "arch": "ALL",
                    "epoch": 0,
                    "fixedInVersion": "0:2.63+22.04ubuntu0.1",
                    "name": "snapd",
                    "packageManager": "OS",
                    "remediation": "apt-get update && apt-get upgrade",
                    "version": "2.63"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsEc2Instance": {
                        "iamInstanceProfileArn":
 "arn:aws:iam::123456789012:instance-profile/AmazonSSMRoleForInstancesQuickSetup",
                        "imageId": "ami-02ff980600c693b38",
                        "ipV4Addresses": [
                            "1.23.456.789",
                            "123.45.67.890"
                        ],
                        "ipV6Addresses": [],
                        "launchedAt": "Wed Sep 04 16:57:40.000 UTC 2024",
                        "platform": "UBUNTU_22_04",
                        "subnetId": "subnet-12345678",
                        "type": "t2.small",
                        "vpcId": "vpc-12345678"
                    }
```

```
                },
                "id": "i-12345678901234567",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_EC2_INSTANCE"
            }
        ],
        "severity": "MEDIUM",
        "status": "CLOSED",
        "title": "CVE-2024-29069 - snapd",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 17:00:36.951 UTC 2024"
    }
}
```

Amazon EC2 network reachability finding

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

```
                    "componentId": "acl-171b527d",
                    "componentType": "AWS::EC2::NetworkAcl"
                }, {
                    "componentId": "sg-0d34debf87410f2d9",
                    "componentType": "AWS::EC2::SecurityGroup"
                }, {
                    "componentId": "eni-094ad651219472857",
                    "componentType": "AWS::EC2::NetworkInterface"
                }, {
                    "componentId": "i-12345678901234567",
                    "componentType": "AWS::EC2::Instance"
                }]
            },
            "openPortRange": {
                "begin": 22,
                "end": 22
            },
            "protocol": "TCP"
        },
        "remediation": {
            "recommendation": {
                "text": "You can restrict access to your instance by modifying the
 Security Groups or ACLs in the network path."
            }
        },
        "resources": [{
            "details": {
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::123456789012:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-02ff980600c693b38",
                    "ipV4Addresses": ["1.23.456.789", "123.45.67.890"],
                    "ipV6Addresses": [],
                    "launchedAt": "Wed Sep 04 17:41:24.000 UTC 2024",
                    "platform": "UBUNTU_22_04",
                    "subnetId": "subnet-12345678",
                    "type": "t2.small",
                    "vpcId": "vpc-12345678"
                }
            },
            "id": "i-12345678901234567",
            "partition": "aws",
            "region": "eu-central-1",
            "type": "AWS_EC2_INSTANCE"
```

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

Amazon ECR package vulnerability finding

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

```
            ],
            "relatedVulnerabilities": [
                "USN-6986-1"
            ],
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2024/
CVE-2024-6119.html",
            "vendorCreatedAt": "Tue Sep 03 00:00:00.000 UTC 2024",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2024-6119",
            "vulnerablePackages": [
                {
                    "arch": "ARM64",
                    "epoch": 0,
                    "fixedInVersion": "0:3.0.13-0ubuntu3.4",
                    "name": "libssl3t64",
                    "packageManager": "OS",
                    "release": "0ubuntu3.2",
                    "remediation": "apt-get update && apt-get upgrade",
                    "sourceLayerHash":
 "sha256:1567e7ea90b67fc95ccdeeec39bdc3045098dee7e0c604975b957a9f8c0e9616",
                    "version": "3.0.13"
                },
                {
                    "arch": "ARM64",
                    "epoch": 0,
                    "fixedInVersion": "0:3.0.13-0ubuntu3.4",
                    "name": "openssl",
                    "packageManager": "OS",
                    "release": "0ubuntu3.2",
                    "remediation": "apt-get update && apt-get upgrade",
                    "sourceLayerHash":
 "sha256:1567e7ea90b67fc95ccdeeec39bdc3045098dee7e0c604975b957a9f8c0e9616",
                    "version": "3.0.13"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
```

```
                "details": {
                    "awsEcrContainerImage": {
                        "architecture": "arm64",
                        "imageHash":
 "sha256:84f507df33c6864d49c296fb734192696e4cb6f78166ac51ac8b9b118181085d",
                        "imageTags": [
                            "ubuntu_latest"
                        ],
                        "platform": "UBUNTU_24_04",
                        "pushedAt": "Wed Sep 04 16:55:28.000 UTC 2024",
                        "registry": "123456789012",
                        "repositoryName": "inspector2"
                    }
                },
                "id": "arn:aws:ecr:eu-central-1:123456789012:repository/inspector2/
sha256:84f507df33c6864d49c296fb734192696e4cb6f78166ac51ac8b9b118181085d",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_ECR_CONTAINER_IMAGE"
            }
        ],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2024-6119 - libssl3t64, openssl",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 16:55:38.411 UTC 2024"
    }
}
```


Lambda package vulnerability finding


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

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

```
                    "scoringVector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N",
                    "source": "NVD",
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://www.debian.org/security/2023/dsa-5442",
                "https://lists.debian.org/debian-lts-announce/2023/08/msg00024.html"
            ],
            "relatedVulnerabilities": [],
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2023-30861",
            "vendorCreatedAt": "Tue May 02 18:15:52.000 UTC 2023",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Sun Aug 20 21:15:09.000 UTC 2023",
            "vulnerabilityId": "CVE-2023-30861",
            "vulnerablePackages": [
                {
                    "epoch": 0,
                    "filePath": "requirements.txt",
                    "fixedInVersion": "2.3.2",
                    "name": "flask",
                    "packageManager": "PIP",
                    "version": "2.0.0"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
                            "X86_64"
                        ],
                        "codeSha256": "O7jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
                        "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
                        "functionName": "VulnerableFunction",
```

```
                        "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
                        "packageType": "ZIP",
                        "runtime": "PYTHON_3_11",
                        "version": "$LATEST"
                    }
                },
                "id": "arn:aws:lambda:eu-
central-1:123456789012:function:VulnerableFunction:$LATEST",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_LAMBDA_FUNCTION"
            }
        ],
        "severity": "HIGH",
        "status": "ACTIVE",
        "title": "CVE-2023-30861 - flask",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 16:50:37.627 UTC 2024"
    }
}
```

Lambda code vulnerability finding

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

```
            "detectorId": "python/hardcoded-credentials@v1.0",
            "detectorName": "Hardcoded credentials",
            "detectorTags": [
                "secrets",
                "security",
                "owasp-top10",
                "top25-cwes",
                "cwe-798",
                "Python"
            ],
            "filePath": {
                "endLine": 6,
                "fileName": "lambda_function.py",
                "filePath": "lambda_function.py",
                "startLine": 6
            },
            "ruleId": "python-detect-hardcoded-aws-credentials"
        },
        "description": "Access credentials, such as passwords and access keys,
 should not be hardcoded in source code. Hardcoding credentials may cause leaks even
 after removing them. This is because version control systems might retain older
 versions of the code. Credentials should be stored securely and obtained from the
 runtime environment.",
        "findingArn": "arn:aws:inspector2:eu-
central-1:123456789012:finding/FINDING_ID",
        "firstObservedAt": "Wed Sep 04 16:51:01.869 UTC 2024",
        "lastObservedAt": "Wed Sep 04 16:51:01.869 UTC 2024",
        "remediation": {
            "recommendation": {
                "text": "Your code uses hardcoded AWS credentials which might
 allow unauthorized users access to your AWS account. These attacks can occur
 a long time after the credentials are removed from the code. We recommend that
 you set AWS credentials with environment variables or an AWS profile instead.
 You should consider deleting the affected account or rotating the secret key
 and then monitoring Amazon CloudWatch for unexpected activity.\n[https://
boto3.amazonaws.com/v1/documentation/api/latest/guide/credentials.html](https://
boto3.amazonaws.com/v1/documentation/api/latest/guide/credentials.html)"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
```

```
                            "X86_64"
                        ],
                        "codeSha256": "O7jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
                        "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
                        "functionName": "VulnerableFunction",
                        "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
                        "packageType": "ZIP",
                        "runtime": "PYTHON_3_11",
                        "version": "$LATEST"
                    }
                },
                "id": "arn:aws:lambda:eu-
central-1:123456789012:function:VulnerableFunction:$LATEST",
                "partition": "aws",
                "region": "eu-central-1",
                "type": "AWS_LAMBDA_FUNCTION"
            }
        ],
        "severity": "CRITICAL",
        "status": "ACTIVE",
        "title": "CWE-798 - Hardcoded credentials",
        "type": "CODE_VULNERABILITY",
        "updatedAt": "Wed Sep 04 16:51:01.869 UTC 2024"
    }
}
```


> **Note**
>
> Der Detailwert gibt die JSON-Details eines einzelnen Ergebnisses als Objekt zurück. Es wird nicht die gesamte Antwortsyntax für Ergebnisse zurückgegeben, die mehrere Ergebnisse innerhalb eines Arrays unterstützt.

Beispiel für ein vollständiges Amazon Inspector Inspector-Ereignisschema für den ersten Scan

Im Folgenden finden Sie ein Beispiel für das EventBridge Ereignisschema für ein Amazon Inspector Inspector-Ereignis zum Abschluss eines ersten Scans. Dieses Ereignis wird ausgelöst, wenn Amazon Inspector einen ersten Scan einer Ihrer Ressourcen abschließt.

Die folgenden Felder kennzeichnen ein Ereignis, bei dem der erste Scan abgeschlossen wurde:

- Das `detail-type` Feld ist auf eingestelltInspector2 Scan.
- Das `detail` Objekt enthält ein `finding-severity-counts` Objekt, das die Anzahl der Ergebnisse in den entsprechenden Schweregradkategorien detailliert beschreibtCRITICAL, z. B.HIGH, undMEDIUM.

Wählen Sie eine der Optionen aus, um je nach Ressourcentyp unterschiedliche Schemas für den ersten Scan anzuzeigen.

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"
  }
}
```

Beispiel für ein Amazon Inspector Inspector-Abdeckungsereignisschema

Im Folgenden finden Sie ein Beispiel für das EventBridge Ereignisschema für ein Amazon Inspector Inspector-Ereignis zur Berichterstattung. Dieses Ereignis wird ausgelöst, wenn die Scanabdeckung von Amazon Inspector für eine Ressource geändert wird. Die folgenden Felder kennzeichnen ein Abdeckungsereignis:

- Das `detail-type` Feld ist auf eingestelltInspector2 Coverage.
- Das `detail` Objekt enthält ein `scanStatus` Objekt, das den neuen Scanstatus für die Ressource angibt.

```
{
```

```
    "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"
    }
}
```

Beispiel für ein Schema zur auto Aktivierung von Amazon Inspector

Das automatische Aktivierungsereignis wird an den delegierten Administrator gesendet, wenn Amazon Inspector die Anzahl der Mitglieder in einer Organisation nicht unterstützen kann. Die folgenden Felder identifizieren ein automatisches Aktivierungsereignis:

- Das `detail-type` Feld ist auf eingestellt. `Inspector2 AutoEnable`
- Das `detail` Objekt beschreibt, warum das auto Aktivierungsereignis fehlgeschlagen ist.

```
{
    "version": "0",
    "id": "85fc3613-e913-7fc4-a80c-a3753e4aa9ae",
    "detail-type": "Inspector2 AutoEnable",
    "source": "aws.inspector2",
    "account": "123456789012",
    "time": "2024-08-21T02:36:48Z",
    "region": "us-east-1",
    "detail": {
```

```
        “version”: “1.0.0”,
        “AutoEnableStatus”: “Failed”,
        “Reason”: "The number of member accounts enabled with AWS Inspector has reached
 the maximum limit of 10,000"
        }
}
```

Das Amazon Inspector SSM-Plugin für Linux and Windows

Dieses Thema beschreibt das Amazon Inspector SSM-Plugin für Linux and Windows Instanzen.

Das Amazon Inspector SSM-Plugin für Linux

Amazon Inspector verwendet das Amazon Inspector SSM-Plugin, um Deep Inspection-Scans auf Linux-Instances durchzuführen. Das Amazon Inspector SSM-Plugin wird automatisch auf Linux-Instances im `/opt/aws/inspector/bin` Verzeichnis installiert. Der Name der ausführbaren Datei lautet`inspectorssmplugin`.

Amazon Inspector verwendet Systems Manager Distributor, um das Plugin auf Ihrer Instance bereitzustellen. Um Deep Inspection Scans durchführen zu können, müssen Systems Manager Distributor und Amazon Inspector Ihr EC2 Amazon-Instance-Betriebssystem unterstützen. Informationen zu den Betriebssystemen, die Systems Manager Distributor unterstützt, finden Sie im AWS Systems Manager Benutzerhandbuch unter Unterstützte Paketplattformen und Architekturen.

Amazon Inspector erstellt Dateiverzeichnisse, um Daten zu verwalten, die für die eingehende Prüfung mit dem Amazon Inspector SSM-Plugin gesammelt wurden. Zu diesen Dateiverzeichnissen gehören `/opt/aws/inspector/var/input` und`/opt/aws/inspector/var/output`.

In der `packages.txt` Datei `/opt/aws/inspector/var/output` werden die vollständigen Pfade zu Paketen gespeichert, die bei einer Tiefeninspektion entdeckt wurden. Wenn Amazon Inspector dasselbe Paket mehrmals auf Ihrer Instance erkennt, werden in der `packages.txt` Datei alle Standorte aufgeführt, an denen das Paket gefunden wurde.

Amazon Inspector speichert Protokolle für das Plugin im `/var/log/amazon/inspector` Verzeichnis.

Deinstallation des Amazon Inspector SSM-Plug-ins

Wenn die `inspectorssmplugin` Datei versehentlich gelöscht wird, versucht die SSM-Zuordnung, die `inspectorssmplugin` Datei beim nächsten `InspectorLinuxDistributor-do-not-delete` Scanintervall erneut zu installieren.

Wenn Sie das EC2 Amazon-Scannen deaktivieren, wird das Plugin automatisch von allen Linux-Hosts deinstalliert.

Das Amazon Inspector SSM-Plugin für Windows

Das Amazon Inspector SSM-Plugin ist erforderlich, damit Amazon Inspector Ihre Daten scannen kann Windows Instanzen. Das Amazon Inspector SSM-Plugin wird automatisch auf Ihrem installiert Windows Instanzen in`C:\Program Files\Amazon\Inspector`, und die ausführbare Binärdatei wird benannt`InspectorSsmPlugin.exe`.

Die folgenden Dateispeicherorte werden erstellt, um Daten zu speichern, die das Amazon Inspector SSM-Plugin sammelt:

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

> **Note**
>
> Standardmäßig wird das Amazon Inspector SSM-Plugin mit niedrigerer Priorität ausgeführt.

> **Note**
>
> Sie können Folgendes verwenden … Windows Instances mit der Standardeinstellung für die Host-Management-Konfiguration. Sie müssen jedoch eine Rolle erstellen oder verwenden, die mit den `ssm:GetParameter` Berechtigungen `ssm:PutInventory` und konfiguriert ist.

Deinstallation des Amazon Inspector SSM-Plug-ins

Wenn die `InspectorSsmPlugin.exe` Datei versehentlich gelöscht wird, installiert die `InspectorDistributor-do-not-delete` Assoziation die Datei beim nächsten Mal neu `InspectorSsmPlugin.exe` Windows Scan-Intervall. Wenn Sie das Amazon Inspector SSM-Plugin deinstallieren möchten, können Sie die Aktion Deinstallieren im `AmazonInspector2-ConfigureInspectorSsmPlugin` Dokument verwenden. Das Amazon Inspector SSM-Plugin wird jedoch automatisch von allen deinstalliert Windows hostet, wenn Sie das EC2 Amazon-Scannen deaktivieren.

> **Note**
>
> Wenn Sie den SSM Agent deinstallieren, bevor Sie Amazon Inspector deaktivieren, verbleibt das Amazon Inspector SSM-Plugin auf dem Windows hostet, sendet aber keine Daten an das Amazon Inspector SSM-Plugin. Weitere Informationen finden Sie unter Amazon Inspector deaktivieren.

Amazon Inspector SBOM-Generator

Eine Softwareliste (Software Bill of Materials, SBOM) ist eine formal strukturierte Liste von Komponenten, Bibliotheken und Modulen, die zur Erstellung einer Software erforderlich sind. Der Amazon Inspector SBOM Generator (Sbomgen) ist ein Tool, das eine SBOM für Archive, Container-Images, Verzeichnisse, lokale Systeme sowie kompilierte Dateien und Binärdateien erstellt. Go Rust Sbomgensucht nach Dateien, die Informationen über installierte Pakete enthalten. Wenn Sbomgen eine relevante Datei gefunden wird, extrahiert es Paketnamen, Versionen und andere Metadaten. Sbomgenwandelt dann Paketmetadaten in eine CycloneDX SBOM um. Sie können Sbomgen damit die CycloneDX SBOM als Datei oder in STDOUT generieren und zur Schwachstellenerkennung SBOMs an Amazon Inspector senden. Sie können es auch Sbomgen als Teil der CI/CD Integration verwenden, bei der Container-Images automatisch als Teil Ihrer Bereitstellungspipeline gescannt werden.

Unterstützte Pakettypen

Sbomgensammelt Inventar für die folgenden Pakettypen:

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

Unterstützte Konfigurationsprüfungen für Container-Images

Sbomgenkann eigenständige Docker-Dateien scannen und den Verlauf anhand vorhandener Images auf Sicherheitsprobleme aufbauen. Weitere Informationen finden Sie unter Amazon Inspector Dockerfile-Checks.

Installation von Sbomgen

Sbomgenist nur für Linux-Betriebssysteme verfügbar.

Sie müssen Docker installiert sein, wenn Sie lokal zwischengespeicherte Bilder analysieren möchtenSbomgen. Dockerist nicht erforderlich, um Bilder zu analysieren, die als `.tar` Dateien exportiert wurden, oder Bilder, die in Remote-Container-Registern gehostet werden.

Amazon Inspector empfiehlt, dass Sie von einem System Sbomgen aus arbeiten, das mindestens die folgenden Hardwarespezifikationen aufweist:

- 4-fache Kern-CPU
- 8 GB RAM

So installieren Sie Sbomgen

1. Laden Sie die neueste Sbomgen ZIP-Datei von der richtigen URL für Ihre Architektur herunter:

   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

   Alternativ können Sie frühere Versionen der Amazon Inspector SBOM Generator-Zip-Datei herunterladen.
2. Entpacken Sie den Download mit dem folgenden Befehl:

   ```
   unzip inspector-sbomgen.zip
   ```
3. Suchen Sie im entpackten Verzeichnis nach den folgenden Dateien:

   - `inspector-sbomgen`— Dies ist das Tool, das Sie ausführen werden, um es zu generieren SBOMs.

- `README.txt`— Dies ist die Dokumentation zur VerwendungSbomgen.
- `LICENSE.txt`— Diese Datei enthält die Softwarelizenz fürSbomgen.
- `licenses`— Dieser Ordner enthält Lizenzinformationen für Pakete von Drittanbietern, die von verwendet werdenSbomgen.
- `checksums.txt`— Diese Datei enthält Hashes des Sbomgen Tools.
- `sbom.json`— Dies ist eine CycloneDX SBOM für das Tool. Sbomgen
- `WhatsNew.txt`— Diese Datei enthält ein zusammengefasstes Änderungsprotokoll, sodass Sie wichtige Änderungen und Verbesserungen zwischen den Sbomgen Versionen schnell einsehen können.

4. (Optional) Überprüfen Sie die Authentizität und Integrität des Tools mit dem folgenden Befehl:

   ```
   sha256sum < inspector-sbomgen
   ```

   - Vergleichen Sie die Ergebnisse mit dem Inhalt der `checksums.txt` Datei.

5. Erteilen Sie dem Tool mit dem folgenden Befehl die Rechte zur ausführbaren Datei:

   ```
   chmod +x inspector-sbomgen
   ```

6. Stellen Sie mit Sbomgen dem folgenden Befehl sicher, dass die Installation erfolgreich abgeschlossen wurde:

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

   Sie sollten die Ausgabe ähnlich der folgenden sehen:

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

Verwenden von Sbomgen

In diesem Abschnitt werden verschiedene Verwendungsmöglichkeiten beschriebenSbomgen. Anhand der integrierten Beispiele können Sie mehr Sbomgen über die Verwendung erfahren. Führen Sie den folgenden `list-examples` Befehl aus, um sich diese Beispiele anzusehen:

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

Generieren Sie eine SBOM für ein Container-Image und geben Sie das Ergebnis aus

Sie können Sbomgen es verwenden, um Bilder SBOMs für Container zu generieren und das Ergebnis in eine Datei auszugeben. Diese Funktion kann mit dem `container` Unterbefehl aktiviert werden.

-Beispielbefehl

Im folgenden Codeausschnitt können Sie es durch die ID Ihres Bilds und *`image:tag output_path.json`* durch den Pfad zu der Ausgabe, die Sie speichern möchten, ersetzen.

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

> **Note**
>
> Die Dauer und Leistung des Scans hängen von der Bildgröße und der geringen Anzahl der Ebenen ab. Kleinere Bilder verbessern nicht nur die Sbomgen Leistung, sondern reduzieren auch die potenzielle Angriffsfläche. Kleinere Images verbessern auch die Dauer der Erstellung, des Herunterladens und Uploads von Images.

Bei Verwendung Sbomgen mit `ScanSbom`verarbeitet die Amazon Inspector Scan API keine Pakete SBOMs , die mehr als 5.000 Pakete enthalten. In diesem Szenario gibt die Amazon Inspector Scan API eine HTTP 400-Antwort zurück.

Wenn ein Bild Massenmediendateien oder -verzeichnisse enthält, sollten Sie erwägen, sie von der Sbomgen Verwendung des `--skip-files` Arguments auszuschließen.

Beispiel: Häufige Fehlerfälle

Das Scannen von Container-Images kann aufgrund der folgenden Fehler fehlschlagen:

- `InvalidImageFormat`— Tritt auf, wenn falsch formatierte Container-Images mit beschädigten TAR-Headern, Manifestdateien oder Konfigurationsdateien gescannt werden.
- `ImageValidationFailure`— Tritt auf, wenn die Überprüfung der Prüfsumme oder der Inhaltslänge für Container-Image-Komponenten fehlschlägt, wie z. B. nicht

übereinstimmende Content-Length-Header, falsche Manifest-Digests oder eine fehlgeschlagene Prüfsummenüberprüfung. SHA256

- `ErrUnsupportedMediaType`— Tritt auf, wenn Bildkomponenten Medientypen enthalten, die nicht unterstützt werden. Informationen zu unterstützten Medientypen finden Sie unter Unterstützte Betriebssysteme und Medientypen.

Amazon Inspector unterstützt den `application/vnd.docker.distribution.manifest.list.v2+json` Medientyp nicht. Amazon Inspector unterstützt jedoch Manifestlisten. Beim Scannen von Bildern, die Manifestlisten verwenden, können Sie mit dem `--platform` Argument explizit angeben, welche Plattform verwendet werden soll. Wenn das `--platform` Argument nicht angegeben ist, wählt der Amazon Inspector SBOM Generator das Manifest automatisch basierend auf der Plattform aus, auf der es ausgeführt wird.

Generieren Sie eine SBOM aus Verzeichnissen und Archiven

Sie können sie verwendenSbomgen, um SBOMs aus Verzeichnissen und Archiven zu generieren. Diese Funktion kann mit den `archive` Unterbefehlen `directory` oder aktiviert werden. Amazon Inspector empfiehlt, diese Funktion zu verwenden, wenn Sie eine SBOM aus einem Projektordner, z. B. einem heruntergeladenen Git-Repository, generieren möchten.

Beispielbefehl 1

Der folgende Ausschnitt zeigt einen Unterbefehl, der eine SBOM aus einer Verzeichnisdatei generiert.

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

Beispiel für Befehl 2

Der folgende Ausschnitt zeigt einen Unterbefehl, der eine SBOM aus einer Archivdatei generiert. Die einzigen unterstützten Archivformate sind, und. `.zip` `.tar` `.tar.gz`

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

Generieren Sie eine SBOM aus Go oder Rust kompilierten Binärdateien

Sie können es verwenden, Sbomgen um SBOMs aus kompilierten Go und binären Dateien zu generieren. Rust Sie können diese Fähigkeit mit dem folgenden Unterbefehl aktivieren: `binary`

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

Generieren Sie eine SBOM aus bereitgestellten Volumes

Sie können Amazon Inspector SBOM Generator verwenden, um SBOMs aus bereitgestellten Volumes zu generieren. Diese Funktion kann mit dem Unterbefehl aktiviert werden. `volume` Wir empfehlen, diese Funktion zu verwenden, wenn Sie Speichervolumes analysieren möchten, z. B. Amazon EBS-Volumes, die auf Ihrem System bereitgestellt wurden. Im Gegensatz zum Unterbefehl directory werden beim Scannen von Mounted Volumes Betriebssystempakete und Betriebssysteminformationen erkannt.

Sie können ein Amazon EBS-Volume scannen, indem Sie es an eine EC2 Amazon-Instance anhängen, auf der Amazon Inspector SBOM Generator installiert ist, und es auf dieser Instance mounten. Für Amazon EBS-Volumes, die derzeit von anderen EC2 Amazon-Instances verwendet werden, können Sie einen Amazon EBS-Snapshot des Volumes erstellen und dann aus diesem Snapshot ein neues Amazon EBS-Volume für Scanzwecke erstellen. Weitere Informationen zu Amazon EBS finden Sie unter Was ist Amazon EBS? im Amazon Elastic Block Store-Benutzerhandbuch.

-Beispielbefehl

Der folgende Ausschnitt zeigt einen Unterbefehl, der aus einem bereitgestellten Volume eine SBOM generiert. Das `--path` Argument sollte das Stammverzeichnis angeben, in dem das Volume eingehängt ist.

```
# generate SBOM from mounted volume
./inspector-sbomgen volume --path /mount/point/of/volume/root
```

-Beispielbefehl

Der folgende Ausschnitt zeigt einen Unterbefehl, der eine SBOM aus einem bereitgestellten Volume generiert und dabei bestimmte Dateipfade mit dem Argument ausschließt. `--exclude-suffix` Das `--exclude-suffix` Argument ist besonders nützlich, wenn ein Volume Massendateien (wie

Protokolldateien oder Mediendateien) enthält. Dateien und Verzeichnisse, deren Pfade mit den angegebenen Suffixen enden, werden vom Scannen ausgeschlossen, wodurch die Scanzeit und der Speicherverbrauch reduziert werden können.

```
# generate SBOM from mounted volume with exclusions
./inspector-sbomgen volume --path /mount/point/of/volume/root \
--exclude-suffix .log \
--exclude-suffix cache
```

Alle Dateipfade im Zielvolume werden auf ihre ursprünglichen Pfade normalisiert. Wenn beispielsweise ein Volume gescannt wird, auf dem `/mnt/volume` sich eine Datei befindet`/mnt/volume/var/lib/rpm/rpmdb.sqlite`, wird der Pfad `/var/lib/rpm/rpmdb.sqlite` in der generierten SBOM auf den Pfad normalisiert.

Senden Sie eine SBOM zur Identifizierung von Sicherheitslücken an Amazon Inspector

Sie können nicht nur eine SBOM generieren, sondern auch eine SBOM zum Scannen mit einem einzigen Befehl aus der Amazon Inspector Scan API senden. Amazon Inspector bewertet den Inhalt der SBOM auf Sicherheitslücken, bevor die Ergebnisse an zurückgegeben werden. Sbomgen Abhängig von Ihren Eingaben können die Ergebnisse angezeigt oder in eine Datei geschrieben werden.

> **Note**
>
> Sie müssen über einen aktiven Benutzer AWS-Konto mit Leseberechtigungen verfügen, `InspectorScan-ScanSbom` um diese Funktion nutzen zu können.

Um diese Funktion zu aktivieren, übergeben Sie das `--scan-sbom` Argument an die Sbomgen CLI. Sie können das `--scan-sbom` Argument auch an einen der folgenden Sbomgen Unterbefehle übergeben:`archive`,,`binary`, `containerdirectory`,`localhost`.

> **Note**
>
> Die Amazon Inspector Scan API verarbeitet SBOMs nicht mehr als 2.000 Pakete. In diesem Szenario gibt die Amazon Inspector Scan API eine HTTP 400-Antwort zurück.

Sie können sich bei Amazon Inspector über ein AWS Profil oder eine IAM-Rolle mit den folgenden AWS CLI Argumenten authentifizieren:

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

Sie können sich auch bei Amazon Inspector authentifizieren, indem Sie die folgenden Umgebungsvariablen angeben. Sbomgen

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

Um das Antwortformat anzugeben, verwenden Sie das `--scan-sbom-output-format cyclonedx` Argument oder das `--scan-sbom-output-format inspector` Argument.

Beispielbefehl 1

Dieser Befehl erstellt eine SBOM für die neueste Alpine Linux Version, scannt die SBOM und schreibt die Ergebnisse der Sicherheitslücke in eine JSON-Datei.

```
./inspector-sbomgen container --image alpine:latest \
                          --scan-sbom \
                          --aws-profile your_profile \
                          --aws-region your_region \
                          --scan-sbom-output-format cyclonedx \
                          --outfile /tmp/inspector_scan.json
```

Beispielbefehl 2

Dieser Befehl authentifiziert sich bei Amazon Inspector und verwendet AWS Anmeldeinformationen als Umgebungsvariablen.

```
AWS_ACCESS_KEY_ID=$your_access_key \
AWS_SECRET_ACCESS_KEY=$your_secret_key \
```

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

Beispielbefehl 3

Dieser Befehl authentifiziert sich bei Amazon Inspector mithilfe des ARN für eine IAM-Rolle.

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

Verwenden Sie zusätzliche Scanner, um die Erkennungsfunktionen zu verbessern

Der Amazon Inspector SBOM Generator wendet vordefinierte Scanner auf der Grundlage des verwendeten Befehls an.

Standard-Scanner-Gruppen

Jeder Amazon Inspector SBOM Generator-Unterbefehl wendet die folgenden Standard-Scanner-Gruppen automatisch an.

- Für den `directory` Unterbefehl: binary, programming-language-packages dockerfile scanner groups
- Für den `localhost` Unterbefehl: os, Scanner-Gruppen für programming-language-packages Extra-Ökosysteme
- Für den `container` Unterbefehl: os, extra-ecosystems programming-language-packages, dockerfile, binary scanner groups

Spezielle Scanner

Verwenden Sie die `--additional-scanners` Option, gefolgt vom Namen des Scanners, der hinzugefügt werden soll, um andere Scanner als die Standard-Scanner-Gruppen einzubeziehen. Im Folgenden finden Sie einen Beispielbefehl, der zeigt, wie das geht.

```
# Add WordPress installation scanner to directory scan
./inspector-sbomgen directory --path /path/to/directory/ --additional-scanners
 wordpress-installation -o output.json
```

Im Folgenden finden Sie einen Beispielbefehl, der zeigt, wie Sie mehrere Scanner mit einer durch Kommas getrennten Liste hinzufügen.

```
./inspector-sbomgen container --image image:tag --additional-scanners scanner1,scanner2
 -o output.json
```

Optimieren Sie Containerscans, indem Sie die maximale zu scannende Dateigröße anpassen

Wenn Sie ein Container-Image analysieren und verarbeiten, werden standardmäßig Dateien mit einer Größe von 200 MB oder weniger Sbomgen gescannt. Dateien, die größer als 200 MB sind, enthalten selten Paketmetadaten. Fehler können auftreten, wenn Sie eine Datei Go oder Rust Binärdatei mit mehr als 200 MB inventarisieren. Verwenden Sie das `--max-file-size` Argument, um die Größenbeschränkung anzupassen. Auf diese Weise können Sie das Limit erhöhen, um große Dateien einzuschließen, und das Limit verringern, um den Ressourcenverbrauch zu reduzieren, indem Sie große Dateien ausschließen.

Beispiel

Das folgende Beispiel zeigt, wie das `--max-file-size` Argument verwendet wird, um die Dateigröße zu erhöhen.

```
# Increase the file size limit to scan files up to 300 MB
./inspector-sbomgen container --image alpine:latest \
--outfile /tmp/sbom.json \
--max-file-size 300000000
```

Durch das Anpassen dieser Einstellung können Sie die Festplattennutzung, den Speicherverbrauch und die Gesamtdauer des Scans kontrollieren.

Deaktivieren Sie die Fortschrittsanzeige

Sbomgenzeigt eine sich drehende Fortschrittsanzeige an, die in CI/CD Umgebungen zu viele Schrägstriche führen kann.

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

Sie können die Fortschrittsanzeige mit dem folgenden `--disable-progress-bar` Argument deaktivieren:

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

Authentifizierung bei privaten Registern mit Sbomgen

Wenn Sie Ihre Anmeldedaten für die private Registrierung angeben, können Sie Daten SBOMs aus Containern generieren, die in privaten Registern gehostet werden. Sie können diese Anmeldeinformationen mit den folgenden Methoden bereitstellen:

Authentifizieren Sie sich mit zwischengespeicherten Anmeldeinformationen (empfohlen)

Für diese Methode authentifizieren Sie sich bei Ihrer Container-Registry. Wenn Sie dies beispielsweise verwendenDocker, können Sie sich mit dem Docker Loging-Befehl bei Ihrer Container-Registry authentifizieren:. `docker login`

1. Authentifizieren Sie sich bei Ihrer Container-Registry. Wenn Sie dies beispielsweise verwendenDocker, können Sie sich mit dem folgenden Befehl bei Ihrer Registrierung authentifizieren: Docker `login`

2. Nachdem Sie sich bei Ihrer Container-Registry authentifiziert haben, verwenden Sie es Sbomgen auf einem Container-Image, das sich in der Registrierung befindet. Um das folgende Beispiel zu verwenden, *image:tag* ersetzen Sie es durch den Namen des zu scannenden Bilds:

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

Authentifizieren Sie sich mit der interaktiven Methode

Geben Sie für diese Methode Ihren Benutzernamen als Parameter an und Sie Sbomgen werden bei Bedarf zur sicheren Passworteingabe aufgefordert.

Um das folgende Beispiel zu verwenden, *image:tag* ersetzen Sie es durch den Namen des Bilds, das Sie scannen möchten, und *your_username* durch einen Benutzernamen, der Zugriff auf das Bild hat:

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

Authentifizieren Sie sich mit der nicht interaktiven Methode

Speichern Sie für diese Methode Ihr Passwort oder Ihr Registrierungstoken in einer `.txt` Datei.

> **Note**
>
> Der aktuelle Benutzer sollte diese Datei nur lesen können. Die Datei sollte auch Ihr Passwort oder Token in einer einzigen Zeile enthalten.

Um das folgende Beispiel zu verwenden, *your_username* ersetzen Sie es durch Ihren Benutzernamen, *password.txt* durch die `.txt` Datei, die Ihr Passwort oder Token in einer einzigen Zeile enthält, und *image:tag* durch den Namen des zu scannenden Bilds:

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

Beispielausgaben von Sbomgen

Im Folgenden finden Sie ein Beispiel für eine SBOM für ein Container-Image, das mithilfe von inventarisiert wurde. Sbomgen

Container-Image (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"
        }
      ]
    }
  ]
}
```

Frühere Versionen des Amazon Inspector SBOM-Generators

Dieses Thema enthält Links zu den neuesten und früheren Versionen des Amazon Inspector SBOM Generators. Informationen zur Installation von Sbomgen finden Sie im Abschnitt Installieren der Sbomgen.

Aktuelle Version

- https://amazon-inspector-sbomgen.s3.amazonaws.com/latest/linux/amd64/inspector-sbomgen.zip
- https://amazon-inspector-sbomgen.s3.amazonaws.com/latest/linux/arm64/inspector-sbomgen.zip

Sbomgen1.8.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.8.0/ linux/amd64/inspector

- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.8.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.7.3

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.3/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.3/linux/arm64/inspector-sbomgen.zip

Sbomgen1.7.2

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.2/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.2/linux/arm64/inspector-sbomgen.zip

Sbomgen1.7.1

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.1/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.7.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.0/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.7.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.6.3

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.3/ linux/amd64/inspector

- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.3/linux/arm64/inspector-sbomgen.zip

Sbomgen1.6.2

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.2/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.2/linux/arm64/inspector-sbomgen.zip

Sbomgen1.6.1

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.1/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.6.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.0/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.6.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.5.5

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.5/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.5/linux/arm64/inspector-sbomgen.zip

Sbomgen1.5.4

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.4/ linux/amd64/inspector

- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.4/linux/arm64/inspector-sbomgen.zip

Sbomgen1.5.3

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.3/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.3/linux/arm64/inspector-sbomgen.zip

Sbomgen1.5.2

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.2/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.2/linux/arm64/inspector-sbomgen.zip

Sbomgen1.5.1

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.1/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.5.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.0/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.5.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.4.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.4.0/ linux/amd64/inspector

- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.4.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.3.2

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.2/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.2/linux/arm64/inspector-sbomgen.zip

Sbomgen1.3.1

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.1/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.3.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.0/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.3.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.2.1

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.1/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.2.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.0/ linux/amd64/inspector

- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.2.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.1.1

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.1/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.1/linux/arm64/inspector-sbomgen.zip

Sbomgen1.1.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.0/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.1.0/linux/arm64/inspector-sbomgen.zip

Sbomgen1.0.0

- Linux AMD64: -sbomgen.zip https://amazon-inspector-sbomgen.s3.amazonaws.com/1.0.0/ linux/amd64/inspector
- Linux ARM64: https://amazon-inspector-sbomgen.s3.amazonaws.com/1.0.0/linux/arm64/inspector-sbomgen.zip

Umfassende Betriebssystemsammlung von Amazon Inspector SBOM Generator

Der Amazon Inspector SBOM Generator scannt verschiedene Betriebssysteme, um eine robuste und detaillierte Analyse der Systemkomponenten zu gewährleisten. Das Generieren einer SBOM hilft Ihnen, die Zusammensetzung Ihres Betriebssystems zu verstehen, sodass Sie Sicherheitslücken in vom System verwalteten Paketen identifizieren können. In diesem Thema werden die wichtigsten Funktionen verschiedener Betriebssystem-Paketsammlungen beschrieben, die der Amazon Inspector SBOM Generator unterstützt. Informationen zu den Betriebssystemen, die Amazon Inspector unterstützt, finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.

Unterstützte Betriebssystemartefakte

Der Amazon Inspector SBOM Generator unterstützt die folgenden Betriebssystemartefakte:

Plattform	Binär	Quelle	Stream
Alma Linux	N/A	Ja	Ja
Alpine Linux	Ja	Ja	–
Amazon Linux	N/A	Ja	–
CentOS	N/A	Ja	N/A
Chainguard	Ja	Ja	N/A
Debian	Ja	Ja	N/A
Distroless	Ja	Ja	–
Fedora	N/A	Ja	N/A
MinimOS	Ja	Ja	–
OpenSUSE	N/A	Ja	–
Oracle Linux	N/A	Ja	–
Photon OS	N/A	Ja	–
RHEL	N/A	Ja	Ja
Rocky Linux	N/A	Ja	Ja
SLES	N/A	Ja	N/A
Ubuntu	Ja	Ja	N/A

APK-basierte Sammlung von Betriebssystem-Paketen

Dieser Abschnitt enthält die unterstützten Plattformen und die wichtigsten Funktionen für die Sammlung von Paketen APK auf Basis von Betriebssystemen. Weitere Informationen finden Sie unter Alpine Package Keeper auf der Alpine Linux Website.

Unterstützte Plattformen

Die folgenden Plattformen werden unterstützt.

- Alpine Linux

> **Note**
>
> Bei APK basierten Systemen sammelt der Amazon Inspector SBOM Generator Paketmetadaten aus der `/lib/apk/db/`Datei.

Schlüsselfeatures

- Sammlung von Paketnamen — Extrahiert den Namen jedes installierten Pakets
- Versionssammlung — Extrahiert die Version jedes installierten Pakets
- Identifizierung des Quellpakets — Identifiziert das Quellpaket für jedes installierte Paket

Beispiel

Der folgende Ausschnitt ist ein Beispiel für eine APK Datenbankdatei.

```
C:Q1JlboSJkrN4qkDcokr4zenpcWEXQ=
P:zlib
V:1.2.13-r1
A:x86_64
S:54253
I:110592
T:A compression/decompression Library
U:https://zlib.net/
L:Zlib
```

```
o:zlib
```

DPKG-basierte Betriebssystem-Paketsammlung

Dieser Abschnitt enthält die unterstützten Plattformen und die wichtigsten Funktionen für die Sammlung von Paketen auf DPKG Basis von Betriebssystemen. Weitere Informationen finden Sie im Debian-Paket auf der Debian Website.

Unterstützte Plattformen

Die folgenden Plattformen werden unterstützt.

- Debian
- Ubuntu

> **Note**
>
> Bei DPKG basierten Systemen sammelt der Amazon Inspector SBOM Generator Paketmetadaten aus der `/var/lib/dpkg/status`Datei.

Schlüsselfeatures

Im Folgenden sind die wichtigsten Funktionen für DPKG basierte Betriebssystempakete aufgeführt.

- Sammlung von Paketnamen — Extrahiert den Namen jedes installierten Pakets
- Versionssammlung — Extrahiert die Version jedes installierten Pakets
- Identifizierung des Quellpakets — Identifiziert das Quellpaket für jedes installierte Paket

Beispiel

Der folgende Ausschnitt ist ein Beispiel für eine Datei. `/var/lib/dpkg/`

```
Package: zlib1g
Status: install ok installed
```

```
Priority: optional
Section: libs
Installed-Size: 168
Maintainer: Mark Brown <broonie@debian.org>
Architecture: amd64
Multi-Arch: same
Source: zlib
Version: 1:1.2.13.dfsg-1
Provides: libz1
Depends: libc6 (>= 2.14)
Breaks: libxml2 (<< 2.7.6.dfsg-2), texlive-binaries (<< 2009-12)
Conflicts: zlib1 (<= 1:1.0.4-7)
Description: compression library - runtime
 zlib is a library implementing the deflate compression method found
 in gzip and PKZIP.  This package includes the shared library.
Homepage: http://zlib.net/
```

RPM-basierte Betriebssystem-Paketsammlung

Dieser Abschnitt enthält die unterstützten Plattformen und die wichtigsten Funktionen für die Sammlung von Paketen RPM auf Basis von Betriebssystemen. Weitere Informationen finden Sie unter RPM Package Manager auf der RPM Website.

Unterstützte Plattformen

Die folgenden Plattformen werden unterstützt.

- Alma Linux
- Amazon Linux
- CentOS
- Fedora
- OpenSUSE
- Oracle Linux
- PhotonOS
- RedHat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server

> **Note**
>
> Bei RPM basierten Systemen sammelt der Amazon Inspector SBOM Generator Paketmetadaten aus der `/var/lib/rpm`Datei.

Schlüsselfeatures

Im Folgenden sind die wichtigsten Funktionen für Paketsammlungen RPM auf Basis von Betriebssystemen aufgeführt.

- Sammlung von Paketnamen — Extrahiert den Namen jedes installierten Pakets
- Versionssammlung — Extrahiert die Version jedes installierten Pakets
- Identifizierung des Quellpakets — Identifiziert das Quellpaket für jedes installierte Paket
- Stream-Unterstützung — Extrahiert Stream-Metadaten jedes installierten Pakets

Beispiel

Im Folgenden finden Sie ein Beispiel für einen RPM Datenbankdateiausschnitt.

```
/usr/lib/sysimage/rpm/rpmdb.sqlite
/usr/lib/sysimage/rpm/Packages
/usr/lib/sysimage/rpm/Packages.db
/var/lib/rpm/rpmdb.sqlite
/var/lib/rpm/Packages
/var/lib/rpm/Packages.db
```

Sammlung von Chainguard-Image-Paketen

Dieser Abschnitt enthält die unterstützten Plattformen und wichtigsten Funktionen für die Sammlung von Chainguard Image-Paketen. Weitere Informationen finden Sie unter Bilder auf der Chainguard Website.

Unterstützte Plattformen

Die folgenden Plattformen werden unterstützt

- Wolfi Linux

> **Note**
>
> Für Chainguard Bilder sammelt der Amazon Inspector SBOM Generator Paketmetadaten aus der `/lib/apk/db/installed` Datei.

Schlüsselfeatures

Im Folgenden sind die wichtigsten Funktionen aufgeführt.

- Sammlung von Paketnamen — Extrahiert den Namen jedes installierten Pakets
- Versionssammlung — Extrahiert die Version jedes installierten Pakets
- Identifizierung des Quellpakets — Identifiziert das Quellpaket für jedes installierte Paket

Beispiel

Der folgende Ausschnitt ist ein Beispiel für eine Chainguard Bilddatei.

```
P:wolfi-keys
V:1-r8
A:x86_64
L:MIT
T:Wolfi signing keyring
o:wolfi-keys
```

Sammlung von Image-Paketen ohne Distribution

DistrolessContainer sind Container-Images, die Paketmanager, Shells und andere Dienstprogramme in Linux Distributionen ausschließen. DistrolessContainer enthalten nur wichtige Abhängigkeiten, die für die Ausführung der Anwendung und die Verbesserung von Leistung und Sicherheit erforderlich sind.

> **Note**
>
> Für DistrolessBilder sammelt der Amazon Inspector SBOM Generator Paketmetadaten aus der `/var/lib/dpkg/status.d` Datei. Es werden nur Distributionen Debian unterstützt, die auf dem Server Ubuntu basieren. Diese können anhand des `NAME` Felds im `/etc/os-release` Dateisystem identifiziert werden, das "" oder Debian "“ anzeigtUbuntu.

Schlüsselfeatures

- Sammlung von Paketnamen — Extrahiert den Namen jedes installierten Pakets
- Versionssammlung — Extrahiert die Version jedes installierten Pakets

Beispiel

Im Folgenden finden Sie ein Beispiel für eine Distroless Image-Datei.

```
Package: tzdata
Version: 2021a-1+deb11u10
Architecture: all
Maintainer: GNU Libc Maintainers <debian-glibc@lists.debian.org>
Installed-Size: 3413
Depends: debconf (>= 0.5) | debconf-2.0
Provides: tzdata-bullseye
Section: localization
Priority: required
Multi-Arch: foreign
Homepage: https://www.iana.org/time-zones
Description: time zone and daylight-saving time data
 This package contains data required for the implementation of
 standard local time for many representative locations around the
 globe. It is updated periodically to reflect changes made by
 political bodies to time zone boundaries, UTC offsets, and
 daylight-saving rules.
```

MiniMOS-Paketsammlung

Dieser Abschnitt enthält die unterstützten Plattformen und wichtigsten Funktionen für die Sammlung von Minimus Image-Paketen. Weitere Informationen finden Sie auf der Minimus-Website.

Unterstützte Plattformen

Die folgenden Plattformen werden unterstützt.

- MinimOS

> **Note**
>
> Für Minimus Bilder sammelt der Amazon Inspector SBOM Generator Paketmetadaten aus der `/lib/apk/db/installed` Datei.

Schlüsselfeatures

Im Folgenden sind die wichtigsten Funktionen aufgeführt.

- Sammlung von Paketnamen — Extrahiert den Namen jedes installierten Pakets
- Versionssammlung — Extrahiert den Namen jedes installierten Pakets
- Identifizierung des Quellpakets — Identifiziert das Quellpaket für jedes installierte Paket

Das Folgende ist ein Ausschnitt aus einer Minimus Bilddatei.

```
P:ca-certificates-bundle
V:20241121-r1
A:aarch64
L:MPL-2.0 AND MIT
T:
o:ca-certificates
```

Sammlung von Abhängigkeiten zu Programmiersprachen

Der Amazon Inspector SBOM Generator unterstützt verschiedene Programmiersprachen und Frameworks, die eine robuste und detaillierte Sammlung von Abhängigkeiten bilden. Die Generierung einer SBOM hilft Ihnen dabei, die Zusammensetzung Ihrer Software zu verstehen, sodass Sie

Schwachstellen identifizieren und die Einhaltung der Sicherheitsstandards sicherstellen können. Der Amazon Inspector SBOM Generator unterstützt die folgenden Programmiersprachen und Dateiformate.

Gehen Sie zum Scannen von Abhängigkeiten

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
Go	Go	`go.mod`	N/A	–	–	–	Ja
		`go.sum`	N/A	–	–	–	Ja
		Go Binaries	Ja	–	–	–	Ja
		`GOMODCACHE`	–	–	–	N/A	Nein

go.mod/go.sum

Verwenden Sie `go.sum` Dateien, um Abhängigkeiten in Projekten zu definieren `go.mod` und zu sperren. Go Der Amazon Inspector SBOM Generator verwaltet diese Dateien je nach Version der Go Toolchain unterschiedlich.

Schlüsselfeatures

- Sammelt Abhängigkeiten von `go.mod` (wenn die Go Toolketten-Version 1.17 oder höher ist)
- Sammelt Abhängigkeiten von `go.sum` (wenn die Go Toolketten-Version 1.17 oder niedriger ist)
- Analysiert`go.mod`, um alle deklarierten Abhängigkeiten und Abhängigkeitsversionen zu identifizieren

`go.mod`-Beispieldatei

Das Folgende ist ein Beispiel für eine `go.mod` Datei.

```
module example.com/project

go 1.17

require (
github.com/gin-gonic/gin v1.7.2
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123
)
```

`go.sum`-Beispieldatei

Das Folgende ist ein Beispiel für eine `go.sum` Datei.

```
github.com/gin-gonic/gin v1.7.2 h1:VZ7DdRl0sghbA6lVGSkX+UXO2+J0aH7RbsNugG+FA8Q=
github.com/gin-gonic/gin v1.7.2/go.mod h1:ILZ1Ngh2f1pL1ASUj7gGk8lGFeNC8cRTaN2ZhsBNbXU=
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123 h1:b6rCu+qHze
+BUsmC3CZzH8aNu8LzPZTVsNTo64OypSc=
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123/go.mod h1:K5Dkpb0Q4ewZW/
EzWlQphgJcUMBCzoWrLfDOVzpTGVQ=
```

> **Note**
>
> Jede dieser Dateien erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Gehe zu Binärdateien

Der Amazon Inspector SBOM Generator extrahiert Abhängigkeiten aus kompilierten Go Binärdateien, um sicherzustellen, dass der verwendete Code verwendet wird.

> **Note**
>
> Der Amazon Inspector SBOM Generator unterstützt das Erfassen und Auswerten von Toolchainversionen aus Go Binärdateien, die mit dem offiziellen Compiler erstellt wurden.

Go Weitere Informationen finden Sie auf der Website unter Herunterladen und Installieren. Go Wenn Sie beispielsweise die Go Toolchain eines anderen Anbieters verwenden, ist die Bewertung aufgrund möglicher Unterschiede bei der Verteilung und der Verfügbarkeit von Metadaten möglicherweise nicht korrekt. Red Hat

Schlüsselfeatures

- Extrahiert Abhängigkeitsinformationen direkt aus Binärdateien Go
- Sammelt Abhängigkeiten, die in die Binärdatei eingebettet sind
- Erkennt und extrahiert die Version der Go Toolchain, die zum Kompilieren der Binärdatei verwendet wurde.

GOMODCACHE

Der Amazon Inspector SBOM Generator scannt den Go Modul-Cache, um Informationen über installierte Abhängigkeiten zu sammeln. In diesem Cache werden heruntergeladene Module gespeichert, um sicherzustellen, dass dieselben Versionen in verschiedenen Builds verwendet werden.

Schlüsselfeatures

- Durchsucht das `GOMODCACHE` Verzeichnis, um zwischengespeicherte Module zu identifizieren
- Extrahiert detaillierte Metadaten, einschließlich Modulnamen, Versionen und Quelle URLs

Beispiel für eine Struktur

Das Folgende ist ein Beispiel für die `GOMODCACHE` Struktur.

```
~/go/pkg/mod/
### github.com/gin-gonic/gin@v1.7.2
### golang.org/x/crypto@v0.0.0-20210616213533-5cf6c0f8e123
```

> **Note**
>
> Diese Struktur erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Scannen von Java-Abhängigkeiten

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
`Java`	`Maven`	Kompilierte Java Anwendungen (.jar/.war/.ear) `pom.xml`	N/A –	– N/A	Ja Ja	– N/A	Ja Ja

Der Amazon Inspector SBOM Generator führt Java Abhängigkeitsscans durch, indem er kompilierte Java Anwendungen und `pom.xml` Dateien analysiert. Beim Scannen kompilierter Anwendungen generiert der Scanner SHA-1-Hashes zur Integritätsprüfung, extrahiert eingebettete `pom.properties` Dateien und analysiert verschachtelte Dateien. `pom.xml`

SHA-1-Hash-Sammlung (für kompilierte JAR-, .WAR-, .EAR-Dateien)

Der Amazon Inspector SBOM Generator versucht, SHA-1-Hashes für alle`.ear`,, und `.war` Dateien in einem Projekt zu sammeln`.jar`, um die Integrität und Rückverfolgbarkeit der kompilierten Artefakte zu gewährleisten. Java

Schlüsselfeatures

- Generiert SHA—1-Hashes für alle kompilierten Artefakte Java

Beispiel für ein Artefakt

Das Folgende ist ein Beispiel für ein SHA-1-Artefakt.

```
{
  "bom-ref": "comp-52",
  "type": "library",
  "name": "jul-to-slf4j",
  "version": "2.0.6",
  "hashes": [
    {
      "alg": "SHA-1",
      "content": ""
    }
  ],
  "purl": "pkg:maven/jul-to-slf4j@2.0.6",
  "properties": [
    {
      "name": "amazon:inspector:sbom_generator:source_path",
      "value": "test-0.0.1-SNAPSHOT.jar/BOOT-INF/lib/jul-to-slf4j-2.0.6.jar"
    }
  ]
}
```

> **Note**
>
> Dieses Artefakt erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

pom.properties

Die `pom.properties` Datei wird in Maven Projekten verwendet, um Projektmetadaten, einschließlich Paketnamen und Paketversionen, zu speichern. Der Amazon Inspector SBOM Generator analysiert diese Datei, um Projektinformationen zu sammeln.

Schlüsselfeatures

- Analysiert und extrahiert Paketartefakte, Paketgruppen und Paketversionen

`pom.properties`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `pom.properties`-Datei dargestellt.

```
#Generated by Maven
#Tue Mar 16 15:44:02 UTC 2021

version=1.6.0
groupId=net.datafaker
artifactId=datafaker
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Ohne verschachteltes Parsing **`pom.xml`**

Wenn Sie das `pom.xml` Parsen beim Scannen kompilierter Java Anwendungen ausschließen möchten, verwenden Sie das Argument. `--skip-nested-pomxml`

pom.xml

Die `pom.xml` Datei ist die zentrale Konfigurationsdatei für Maven Projekte. Sie enthält Informationen über Projekte und Projektabhängigkeiten. Der Amazon Inspector SBOM Generator analysiert

pom.xml Dateien, um Abhängigkeiten zu erfassen. Dabei werden eigenständige Dateien in Repositorys und Dateien in kompilierten Dateien gescannt. .jar

Schlüsselfeatures

- Analysiert und extrahiert Paketartefakte, Paketgruppen und Paketversionen aus Dateien. `pom.xml`

Unterstützte Maven Bereiche und Tags

Abhängigkeiten werden in den folgenden Maven Bereichen erfasst:

- compile
- bereitgestellt
- runtime
- Test
- system
- einführen

Abhängigkeiten werden mit dem folgenden Maven Tag gesammelt:`<optional>true</optional>`.

`pom.xml`Beispieldatei mit einem Bereich

Das Folgende ist ein Beispiel für eine `pom.xml` Datei mit einem Bereich.

```
<dependency>
<groupId>jakarta.servlet</groupId>
<artifactId>jakarta.servlet-api</artifactId>
</version>6.0.0</version>
<scope>provided</scope>
</dependency>
<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>8.0.28</version>
<scope>runtime</scope>
</dependency>
```

`pom.xml`Beispieldatei ohne Gültigkeitsbereich

Im Folgenden finden Sie ein Beispiel für eine `pom.xml` Datei ohne Bereich.

```
<dependency>
<groupId>com.fasterxml.jackson.core</groupId>
<artifactId>jackson-databind</artifactId>
<version>2.17.1</version>
</dependency>

<dependency>
<groupId>org.jenkins-ci.plugins</groupId>
<artifactId>plain-credentials</artifactId>
<version>183.va_de8f1dd5a_2b_</version>
</dependency>

<dependency>
<groupId>org.jenkins-ci.plugins</groupId>
<artifactId>jackson2-api</artifactId>
<version>2.15.2-350.v0c2f3f8fc595</version>
</dependency>
```

> **Note**
>
> Jede dieser Dateien erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

JavaScript Scannen von Abhängigkeiten

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
`Javascript`	`Node Modules`	`node_modules/`	N/A	N/A	Ja	Ja	Ja

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolkette-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
	NPM	*/package.json	–	Ja	–	–	Nein
	PNPM		–	Ja	–	–	Nein
	YARN	package-lock.json (v1, v2, and v3) / npm-shrinkwrap.json pnpm-lock.yaml yarn.lock	–	Ja	N/A	N/A	Nein

package.json

Die `package.json` Datei ist eine Kernkomponente von Projekten. Node.js Sie enthält Metadaten zu installierten Paketen. Der Amazon Inspector SBOM Generator scannt diese Datei, um Paketnamen und Paketversionen zu identifizieren.

Schlüsselfeatures

- Analysiert die JSON-Dateistruktur, um Paketnamen und Versionen zu extrahieren
- Identifiziert private Pakete mit privaten Werten

`package.json`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `package.json`-Datei dargestellt.

```
{
"name": "arrify",
"private": true,
"version": "2.0.1",
"description": "Convert a value to an array",
"license": "MIT",
"repository": "sindresorhus/arrify"
}
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

package-lock.json

Die `package-lock.json` Datei wird automatisch von npm generiert, um genaue Versionen der für ein Projekt installierten Abhängigkeiten zu sperren. Es gewährleistet die Konsistenz in Umgebungen, indem exakte Versionen aller Abhängigkeiten und ihrer Unterabhängigkeiten gespeichert werden. Diese Datei kann zwischen regulären Abhängigkeiten und Entwicklungsabhängigkeiten unterscheiden.

Schlüsselfeatures

- Analysiert die JSON-Dateistruktur, um Paketnamen und Paketversionen zu extrahieren
- Unterstützt die Erkennung von Abhängigkeiten von Entwicklern

`package-lock.json`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `package-lock.json`-Datei dargestellt.

```
"verror": {
"version": "1.10.0",
"resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
"integrity": "sha1-OhBcoXBTr1XW4nDB+CiGguGNpAA=",
"requires": {
    "assert-plus": "^1.0.0",
    "core-util-is": "1.0.2",
    "extsprintf": "^1.2.0"
}
},
"wrappy": {
"version": "1.0.2",
"resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",
"integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",
"dev": true
},
"yallist": {
"version": "3.0.2",
"resolved": "https://registry.npmjs.org/yallist/-/yallist-3.0.2.tgz",
"integrity": "sha1-hFK0u36Dx8GI2AQcGoN8dz1ti7k="
}
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

npm-shrinkwrap.json

npmgeneriert `package-lock.json` automatisch `npm-shrinkwrap.json` Dateien, um exakte Versionen der für ein Projekt installierten Abhängigkeiten zu sperren. Dies garantiert Konsistenz in Umgebungen, indem exakte Versionen aller Abhängigkeiten und Unterabhängigkeiten gespeichert werden. Die Dateien unterscheiden zwischen regulären Abhängigkeiten und Entwicklungsabhängigkeiten.

Schlüsselfeatures

- Analysieren Sie die `package-lock` Versionen 1, 2 und 3 der JSON Dateistruktur, um den Paketnamen und die Version zu extrahieren
- Die Erkennung von Abhängigkeiten zwischen Entwicklern wird unterstützt (`package-lock.json`erfasst Produktions- und Entwicklungsabhängigkeiten, sodass Tools erkennen können, welche Pakete in Entwicklungsumgebungen verwendet werden)
- Die `npm-shrinkwrap.json` Datei hat Vorrang vor der `package-lock.json` Datei

Beispiel

Im Folgenden wird ein Beispiel für eine `package-lock.json`-Datei dargestellt.

```
"verror": {
            "version": "1.10.0",
            "resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
            "integrity": "sha1-OhBcoXBTr1XW4nDB+CiGguGNpAA=",
            "requires": {
                "assert-plus": "^1.0.0",
                "core-util-is": "1.0.2",
                "extsprintf": "^1.2.0"
            }
        },
        "wrappy": {
            "version": "1.0.2",
            "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",
            "integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",
            "dev": true
        },
        "yallist": {
            "version": "3.0.2",
            "resolved": "https://registry.npmjs.org/yallist/-/yallist-3.0.2.tgz",
            "integrity": "sha1-hFK0u36Dx8GI2AQcGoN8dz1ti7k="
}
```

pnpm-yaml.lock

Die `pnpm-lock.yaml` Datei wird von pnpm generiert, um eine Aufzeichnung der installierten Abhängigkeitsversionen zu führen. Außerdem werden Entwicklungsabhängigkeiten separat verfolgt.

Schlüsselfeatures

- Analysiert die YAML-Dateistruktur, um Paketnamen und Versionen zu extrahieren
- Unterstützt die Erkennung von Abhängigkeiten von Entwicklern

Beispiel

Im Folgenden wird ein Beispiel für eine `pnpm-lock.yaml`-Datei dargestellt.

```
lockfileVersion: 5.3
importers:
my-project:
dependencies:
  lodash: 4.17.21
devDependencies:
  jest: 26.6.3
specifiers:
  lodash: ^4.17.21
  jest: ^26.6.3
packages:
/lodash/4.17.21:
resolution:
  integrity: sha512-xyz
engines:
  node: '>=6'
dev: false
/jest/26.6.3:
resolution:
  integrity: sha512-xyz
dev: true
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

yarn.lock

Der Amazon Inspector SBOM Generator versucht, SHA-1-Hashes für `.ear`, und `.war` Dateien in einem Projekt zu sammeln `.jar`, um die Integrität und Rückverfolgbarkeit kompilierter Artefakte zu gewährleisten. Java

Schlüsselfeatures

- Generiert SHA—1-Hashes für alle kompilierten Artefakte Java

Beispiel für ein SHA-1-Artefakt

Im Folgenden finden Sie ein Beispiel für ein SHA-1-Artefakt.

```
"@ampproject/remapping@npm:^2.2.0":
version: 2.2.0
resolution: "@ampproject/remapping@npm:2.2.0"
dependencies:
"@jridgewell/gen-mapping": ^0.1.0
"@jridgewell/trace-mapping": ^0.3.9
checksum:
 d74d170d06468913921d72430259424b7e4c826b5a7d39ff839a29d547efb97dc577caa8ba3fb5cf023624e9af9d09
languageName: node
linkType: hard

"@babel/code-frame@npm:^7.0.0, @babel/code-frame@npm:^7.12.13, @babel/code-
frame@npm:^7.18.6, @babel/code-frame@npm:^7.21.4":
version: 7.21.4
resolution: "@babel/code-frame@npm:7.21.4"
dependencies:
"@babel/highlight": ^7.18.6
checksum:
 e5390e6ec1ac58dcef01d4f18eaf1fd2f1325528661ff6d4a5de8979588b9f5a8e852a54a91b923846f7a5c681b217
languageName: node
linkType: hard
```

> **Note**
>
> Dieses Artefakt erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer

Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

.NET-Abhängigkeiten scannen

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
.NET	.NET Core	*.deps.json	N/A	–	–	–	Ja
	Nuget	Packages.config	–	–	N/A	–	Ja
	Nuget	packages.lock.json	–	–	Ja	–	Ja
	.NET	.csproj	–	–	–	N/A	Ja

Packages.config

Die `Packages.config` Datei ist eine XML-Datei, die von einer älteren Version von Nuget zur Verwaltung von Projektabhängigkeiten verwendet wird. Sie listet alle Pakete auf, auf die das Projekt verweist, einschließlich bestimmter Versionen.

Schlüsselfeatures

- Analysiert die XML-Struktur, um das Paket IDs und die Versionen zu extrahieren

Beispiel

Im Folgenden wird ein Beispiel für eine `Packages.config`-Datei dargestellt.

```
<?xml version="1.0" encoding="utf-8"? >
<packages>
<package id="FluentAssertions" version="5.4.1" targetFramework="net461" />
<package id="Newtonsoft.Json" version="11.0.2" targetFramework="net461" />
<package id="SpecFlow" version="2.4.0" targetFramework="net461" />
<package id="SpecRun.Runner" version="1.8.0" targetFramework="net461" />
<package id="SpecRun.SpecFlow" version="1.8.0" targetFramework="net461" />
<package id="SpecRun.SpecFlow.2-4-0" version="1.8.0" targetFramework="net461" />
<package id="System.ValueTuple" version="4.5.0" targetFramework="net461" />
</packages>
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

*.deps.json

Die `*.deps.json` Datei wird von .NET Core Projekten generiert und enthält detaillierte Informationen zu allen Abhängigkeiten, einschließlich Pfaden, Versionen und Laufzeitabhängigkeiten. Diese Datei stellt sicher, dass die Runtime über die notwendigen Informationen verfügt, um die richtigen Versionen von Abhängigkeiten zu laden.

Schlüsselfeatures

- Analysiert die JSON-Struktur nach umfassenden Abhängigkeitsdetails
- Extrahiert Paketnamen und Versionen in eine `libraries` Liste.

`.deps.json`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `.deps.json`-Datei dargestellt.

```
{
```

```
"runtimeTarget": {
    "name": ".NETCoreApp,Version=v7.0",
    "signature": ""
},
"libraries": {
    "sample-Nuget/1.0.0": {
        "type": "project",
        "serviceable": false,
        "sha512": ""
    },
    "Microsoft.EntityFrameworkCore/7.0.5": {
        "type": "package",
        "serviceable": true,
        "sha512": "sha512-
RXbRLHHWP2Z3pq8qcL5nQ6LPeoOyp8hasM5bd0Te8PiQi3RjWQR4tcbdY5XMqQ+oTO9wA8/RLhZRn/
hnxlTDnQ==",
        "path": "microsoft.entityframeworkcore/7.0.5",
        "hashPath": "microsoft.entityframeworkcore.7.0.5.nupkg.sha512"
    },
}
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

packages.lock.json

Die `packages.lock.json` Datei wird von neueren Versionen von verwendetNuget, um genaue Versionen von Abhängigkeiten für ein .NET Projekt zu sperren, um sicherzustellen, dass dieselben Versionen in verschiedenen Umgebungen konsistent verwendet werden.

Schlüsselfeatures

- Analysiert die JSON-Struktur, um gesperrte Abhängigkeiten aufzulisten
- Unterstützt sowohl direkte als auch transitive Abhängigkeiten
- Extrahiert den Paketnamen und die aufgelösten Versionen

packages.lock.json-Beispieldatei

Im Folgenden wird ein Beispiel für eine `packages.lock.json`-Datei dargestellt.

```
{
"version": 1,
"dependencies": {
"net7.0": {
  "Microsoft.EntityFrameworkCore": {
    "type": "Direct",
    "requested": "[7.0.5, )",
    "resolved": "7.0.5",
    "contentHash": "RXbRLHHWP2Z3pq8qcL5nQ6LPeoOyp8hasM5bd0Te8PiQi3RjWQR4tcbdY5XMqQ
+oTO9wA8/RLhZRn/hnxlTDnQ==",
    "dependencies": {
      "Microsoft.EntityFrameworkCore.Abstractions": "7.0.5",
      "Microsoft.EntityFrameworkCore.Analyzers": "7.0.5",
      "Microsoft.Extensions.Caching.Memory": "7.0.0",
      "Microsoft.Extensions.DependencyInjection": "7.0.0",
      "Microsoft.Extensions.Logging": "7.0.0"
    }
  },
  "Newtonsoft.Json": {
    "type": "Direct",
    "requested": "[13.0.3, )",
    "resolved": "13.0.3",
    "contentHash": "HrC5BXdl00IP9zeV+0Z848QWPAoCr9P3bDEZguI+gkLcBKAOxix/tLEAAHC
+UvDNPv4a2d18lOReHMOagPa+zQ=="
  },
  "Microsoft.Extensions.Primitives": {
    "type": "Transitive",
    "resolved": "7.0.0",
    "contentHash": "um1KU5kxcRp3CNuI8o/GrZtD4AIOXDk
+RLsytjZ9QPok3ttLUelLKpilVPuaFT3TFjOhSibUAso0odbOaCDj3Q=="
  }
}
}
}
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

.csproj

Die `.csproj` Datei ist in XML geschrieben und die Projektdatei für Projekte. .NET Sie enthält Verweise auf Nuget Pakete, Projekteigenschaften und Build-Konfigurationen.

Schlüsselfeatures

- Analysiert die XML-Struktur, um Paketverweise zu extrahieren

`.csproj`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `.csproj`-Datei dargestellt.

```
<Project Sdk="Microsoft.NET.Sdk">
<PropertyGroup>
<TargetFramework>net7.0</TargetFramework>
<RootNamespace>sample_Nuget</RootNamespace>
<ImplicitUsings>enable</ImplicitUsings>
<Nullable>enable</Nullable>
<RestorePackagesWithLockFile>true</RestorePackagesWithLockFile>
</PropertyGroup>
<ItemGroup>
</ItemGroup>
<ItemGroup>
<PackageReference Include="Newtonsoft.Json" Version="13.0.3" />
<PackageReference Include="Microsoft.EntityFrameworkCore" Version="7.0.5" />
</ItemGroup>
</Project>
```

`.csproj`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `.csproj`-Datei dargestellt.

```
<PackageReference Include="ExamplePackage" Version="6.*" />
<PackageReferencePackageReference Include="ExamplePackage" Version="(4.1.3,)" />
<PackageReference Include="ExamplePackage" Version="(,5.0)" />
<PackageReference Include="ExamplePackage" Version="[1,3)" />
<PackageReference Include="ExamplePackage" Version="[1.3.2,1.5)" />
```

> **Note**
>
> Jede dieser Dateien erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Scannen von PHP-Abhängigkeiten

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
PHP	Composer	`composer.lock` `/vendor/composer/installed.json`	N/A –	– N/A	Ja Ja	– N/A	Ja Ja

composer.lock

Die `composer.lock` Datei wird automatisch generiert, wenn die Befehle composer install oder composer update ausgeführt werden. Diese Datei garantiert, dass in jeder Umgebung dieselben Versionen von Abhängigkeiten installiert sind. Dies sorgt für einen konsistenten und zuverlässigen Erstellungsprozess.

Schlüsselfeatures

- Analysiert das JSON-Format für strukturierte Daten
- Extrahiert Namen und Versionen von Abhängigkeiten

`composer.lock`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `composer.lock`-Datei dargestellt.

```
{
"packages": [
    {
        "name": "nesbot/carbon",
        "version": "2.53.1",
        // TRUNCATED
    },
    {
        "name": "symfony/deprecation-contracts",
        "version": "v3.2.1",
        // TRUNCATED
    },
    {
        "name": "symfony/polyfill-mbstring",
        "version": "v1.27.0",
        // TRUNCATED
    }
]
// TRUNCATED
}
```

> **Note**
>
> Dadurch wird eine Ausgabe erzeugt, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

/.json vendor/composer/installed

Die `/vendor/composer/installed.json` Datei befindet sich im `vendor/composer` Verzeichnis und bietet eine umfassende Liste aller installierten Pakete und Paketversionen.

Schlüsselfeatures

- Analysiert das JSON-Format für strukturierte Daten
- Extrahiert die Namen und die Version der Abhängigkeiten

`/vendor/composer/installed.json`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `/vendor/composer/installed.json`-Datei dargestellt.

```
{
"packages": [
    {
        "name": "nesbot/carbon",
        "version": "2.53.1",
        // TRUNCATED
    },
    {
        "name": "symfony/deprecation-contracts",
        "version": "v3.2.1",
        // TRUNCATED
    },
    {
        "name": "symfony/polyfill-mbstring",
        "version": "v1.27.0",
        // TRUNCATED
    }
]
```

```
// TRUNCATED
}
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Scannen von Python-Abhängigkeiten

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
Python	pip	requirements.txt	N/A	–	–	–	Ja
	Poetry	Poetry.lock	–	–	–	–	Ja
	Pipenv	Pipfile.lock	–	–	–	–	Ja
	Egg/Wheel	.egg-info/PKG-INFO	–	–	–	N/A	Ja
		.dist-info/METADATA					

requirements.txt

Die `requirements.txt` Datei ist ein in Python Projekten weit verbreitetes Format, um Projektabhängigkeiten zu spezifizieren. Jede Zeile in dieser Datei enthält ein Paket mit seinen Versionseinschränkungen. Der Amazon Inspector SBOM Generator analysiert diese Datei, um Abhängigkeiten genau zu identifizieren und zu katalogisieren.

Schlüsselfeatures

- Unterstützt Versionsbezeichner (== und =)
- Unterstützt Kommentare und komplexe Abhängigkeitszeilen

> **Note**
>
> Die Versionsbezeichner <= und => werden nicht unterstützt.

`requirements.txt`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `requirements.txt`-Datei dargestellt.

```
flask==1.1.2
requests==2.24.0
numpy==1.18.5
foo~=1.2.0
# Comment about a dependency
scipy. # invalid
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Pipfile.lock

Pipenvist ein Tool, das das Beste aus allen Verpackungswelten bietet (gebündelt, gepinnt und unverheftet). Es `Pipfile.lock` sperrt genaue Versionen von Abhängigkeiten, um deterministische Builds zu ermöglichen. Der Amazon Inspector SBOM Generator liest diese Datei, um Abhängigkeiten und ihre aufgelösten Versionen aufzulisten.

Schlüsselfeatures

- Analysiert das JSON-Format auf die Auflösung von Abhängigkeiten
- Unterstützt Standard- und Entwicklungsabhängigkeiten

`Pipfile.lock`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `Pipfile.lock`-Datei dargestellt.

```
{
"default": {
    "requests": {
        "version": "==2.24.0",
        "hashes": [
            "sha256:cc718bb187e53b8d"
        ]
    }
},
"develop": {
    "blinker": {
        "hashes": [
            "sha256:1779309f71bf239144b9399d06ae925637cf6634cf6bd131104184531bf67c01",
            "sha256:8f77b09d3bf7c795e969e9486f39c2c5e9c39d4ee07424be2bc594ece9642d83"
        ],
        "markers": "python_version >= '3.8'",
        "version": "==1.8.2"
    }
}
}
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Poetry.lock

Poetryist ein Tool zur Verwaltung und Paketierung von Abhängigkeiten für Python. Die `Poetry.lock` Datei sperrt genaue Versionen von Abhängigkeiten, um konsistente Umgebungen zu ermöglichen. Der Amazon Inspector SBOM Generator extrahiert detaillierte Abhängigkeitsinformationen aus dieser Datei.

Schlüsselfeatures

- Analysiert das TOML-Format für strukturierte Daten
- Extrahiert Namen und Versionen von Abhängigkeiten

`Poetry.lock`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `Poetry.lock`-Datei dargestellt.

```
[[package]]
name = "flask"
version = "1.1.2"
description = "A simple framework for building complex web applications."
category = "main"
optional = false
python-versions = ">=3.5"
[[package]]
name = "requests"
version = "2.24.0"
description = "Python HTTP for Humans."
category = "main"
optional = false
python-versions = ">=3.5"
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Eier/Rad

Für global installierte Python-Pakete unterstützt der Amazon Inspector SBOM Generator das Parsen von Metadatendateien in den `.egg-info/PKG-INFO` Verzeichnissen und. `.dist-info/METADATA` Diese Dateien enthalten detaillierte Metadaten zu installierten Paketen.

Schlüsselfeatures

- Extrahiert den Paketnamen und die Version
- Unterstützt sowohl Eier- als auch Radformate

`PKG-INFO/METADATA`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `PKG-INFO/METADATA`-Datei dargestellt.

```
Metadata-Version: 1.2
Name: Flask
Version: 1.1.2
Summary: A simple framework for building complex web applications.
Home-page: https://palletsprojects.com/p/flask/
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Ruby-Abhängigkeitsscan

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolchaining-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
Ruby	Bundler	`Gemfile.lock`	N/A	–	Ja	–	Ja
		`.gemspec`	–	–	–	–	Ja
		`globall installed Gems`	–	N/A	–	N/A	Ja

GemFile.lock

Die `Gemfile.lock` Datei sperrt die exakten Versionen aller Abhängigkeiten, um sicherzustellen, dass in jeder Umgebung dieselben Versionen verwendet werden.

Schlüsselfeatures

- Analysiert die `Gemfile.lock` Datei, um Abhängigkeiten und Abhängigkeitsversionen zu identifizieren
- Extrahiert detaillierte Paketnamen und Paketversionen

`Gemfile.lock`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `Gemfile.lock`-Datei dargestellt.

```
GEM
remote: https://rubygems.org/
specs:
ast (2.4.2)
awesome_print (1.9.2)
```

```
diff-lcs (1.5.0)
json (2.6.3)
parallel (1.22.1)
parser (3.2.2.0)
nokogiri (1.16.6-aarch64-linux)
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

.gemspec

Die `.gemspec` Datei ist eine RubyGem Datei, die Metadaten zu einem Edelstein enthält. Der Amazon Inspector SBOM Generator analysiert diese Datei, um detaillierte Informationen über einen Edelstein zu sammeln.

Schlüsselfeatures

- Analysiert und extrahiert den Edelsteinnamen und die Edelsteinversion

> **Note**
>
> Die Referenzspezifikation wird nicht unterstützt.

`.gemspec`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `.gemspec`-Datei dargestellt.

```
Gem::Specification.new do |s|
s.name        = "generategem"
s.version     = "2.0.0"
s.date        = "2020-06-12"
s.summary     = "generategem"
```

```
s.description = "A Gemspec Builder"
s.email       = "edersondeveloper@gmail.com"
s.files       = ["lib/generategem.rb"]
s.homepage    = "https://github.com/edersonferreira/generategem"
s.license     = "MIT"
s.executables = ["generategem"]
s.add_dependency('colorize', '~> 0.8.1')
end
```

```
# Not supported

Gem::Specification.new do |s|
s.name        = &class1
s.version     = &foo.bar.version
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Weltweit installierte Edelsteine

Der Amazon Inspector SBOM Generator unterstützt das Scannen global installierter Gems, die sich in Standardverzeichnissen wie `/usr/local/lib/ruby/gems/<ruby_version>/gems/` Amazon EC2 /Amazon ECR und Lambda befinden. `ruby/gems/<ruby_version>/gems/` Dadurch wird sichergestellt, dass alle global installierten Abhängigkeiten identifiziert und katalogisiert werden.

Schlüsselfeatures

- Identifiziert und scannt alle global installierten Gems in Standardverzeichnissen
- Extrahiert Metadaten und Versionsinformationen für jedes global installierte Gem

Beispiel für eine Verzeichnisstruktur

Das Folgende ist ein Beispiel für eine Verzeichnisstruktur.

```
.
### /usr/local/lib/ruby/3.5.0/gems/
### actrivesupport-6.1.4
### concurrent-ruby-1.1.9
### i18n-1.8.10
```

> **Note**
>
> Diese Struktur erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben, und sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Scannen von Abhängigkeiten auf Rost

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolketten-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
Rust	Cargo.toml	Cargo.toml	N/A	–	N/A	–	Ja
		Cargo.lock	N/A	–	Ja	–	Ja
		Rust binary (built with cargo-	Ja	–	–	N/A	Ja

Programmiersprache	Paketmanager	Unterstützte Artefakte	Toolkette-Unterstützung	Abhängigkeiten bei der Entwicklung	Transitive Abhängigkeiten	Private Flagge	Rekursiv
		`auditable`)					

Fracht.toml

Die `Cargo.toml` Datei ist die Manifestdatei für Projekte. Rust

Schlüsselfeatures

- Analysiert und extrahiert die `Cargo.toml` Datei, um den Namen und die Version des Projektpakets zu identifizieren.

`Cargo.toml`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `Cargo.toml`-Datei dargestellt.

```
[package]
name = "wait-timeout"
version = "0.2.0"
description = "A crate to wait on a child process with a timeout specified across Unix
 and\nWindows platforms.\n"
homepage = "https://github.com/alexcrichton/wait-timeout"
documentation = "https://docs.rs/wait-timeout"
readme = "README.md"
categories = ["os"]
license = "MIT/Apache-2.0"
repository = "https://github.com/alexcrichton/wait-timeout"
[target."cfg(unix)".dependencies.libc]
version = "0.2"
[badges.appveyor]
repository = "alexcrichton/wait-timeout"
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Cargo.lock

Die `Cargo.lock` Datei sperrt Abhängigkeitsversionen, um sicherzustellen, dass bei der Erstellung eines Projekts dieselben Versionen verwendet werden.

Schlüsselfeatures

- Analysiert die `Cargo.lock` Datei, um alle Abhängigkeiten und Abhängigkeitsversionen zu identifizieren.

`Cargo.lock`-Beispieldatei

Im Folgenden wird ein Beispiel für eine `Cargo.lock`-Datei dargestellt.

```
# This file is automatically @generated by Cargo.
# It is not intended for manual editing.
[[package]]
name = "adler32"
version = "1.0.3"
source = "registry+https://github.com/rust-lang/crates.io-index"

[[package]]
name = "aho-corasick"
version = "0.7.4"
source = "registry+https://github.com/rust-lang/crates.io-index"
```

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste

anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Rust-Binärdateien mit überprüfbarer Ladung

Der Amazon Inspector SBOM Generator sammelt Abhängigkeiten von Rust Binärdateien, die mit der Bibliothek erstellt wurden. `cargo-auditable` Dadurch werden zusätzliche Abhängigkeitsinformationen bereitgestellt, indem die Extraktion von Abhängigkeiten aus kompilierten Binärdateien aktiviert wird.

Schlüsselfeatures

- Extrahiert Abhängigkeitsinformationen direkt aus Rust Binärdateien, die mit der Bibliothek erstellt wurden `cargo-auditable`
- Ruft Metadaten und Versionsinformationen für Abhängigkeiten ab, die in den Binärdateien enthalten sind

> **Note**
>
> Diese Datei erzeugt eine Ausgabe, die eine Paket-URL enthält. Diese URL kann verwendet werden, um Informationen zu Softwarepaketen bei der Generierung einer Softwareliste anzugeben. Sie kann in die ScanSbomAPI aufgenommen werden. Weitere Informationen finden Sie unter package-url auf der GitHub Website.

Nicht unterstützte Artefakte

In diesem Abschnitt werden nicht unterstützte Artefakte beschrieben.

Java

Der Amazon Inspector SBOM Generator Generator unterstützt nur die Erkennung von Sicherheitslücken für Abhängigkeiten, die aus dem Maven Mainstream-Repository stammen. Private oder benutzerdefinierte Maven Repositorys wie Red Hat Maven und werden nicht Jenkins unterstützt. Stellen Sie für eine genaue Erkennung von Sicherheitslücken sicher, dass Java Abhängigkeiten aus dem Maven Mainstream-Repository abgerufen werden. Abhängigkeiten von anderen Repositorys werden bei Schwachstellenscans nicht berücksichtigt.

JavaScript

Esbuild-Pakete

Bei esbuild minimierten Paketen unterstützt der Amazon Inspector SBOM Generator keinen Abhängigkeitsscan für Projekte mit. esbuild Quellzuordnungen, die von generiert wurdenesbuild, enthalten nicht genügend Metadaten (Namen und Versionen von Abhängigkeiten), die für eine genaue Generierung erforderlich sind. Sbomgen Um zuverlässige Ergebnisse zu erzielen, scannen Sie vor dem Bündelungsprozess die ursprünglichen Projektdateien`package-lock.json`, z. B. die Dateien `node_modules/directory` und.

package.json

Der Amazon Inspector SBOM Generator unterstützt das Scannen der package.json-Datei auf Stammebene nicht nach Abhängigkeitsinformationen. Diese Datei spezifiziert nur Paketnamen und Versionsbereiche, enthält jedoch keine vollständig aufgelösten Paketversionen. Verwenden Sie `package.json` oder andere Sperrdateien wie `yarn.lock` und, die aufgelöste Versionen enthaltenpnpm`.lock`, um genaue Scanergebnisse zu erzielen.

Punktnetz

Wenn Sie schwebende Versionen oder Versionsbereiche in verwenden`PackageReference`, wird es schwieriger, die genaue Paketversion zu ermitteln, die in einem Projekt verwendet wird, ohne die Paketauflösung durchzuführen. Schwebende Versionen und Versionsbereiche ermöglichen es Entwicklern, anstelle einer festen Version eine Reihe akzeptabler Paketversionen anzugeben.

Gehen Sie zu Binärdateien

Der Amazon Inspector SBOM Generator scannt keine Go Binärdateien, die mit Build-Flags erstellt wurden, die so konfiguriert sind, dass sie die Build-ID ausschließen. Diese Build-Flags Bomerman verhindern, dass die Binärdatei ihrer ursprünglichen Quelle genau zugeordnet wird. Unklare Go Binärdateien werden nicht unterstützt, da Paketinformationen nicht extrahiert werden können. Stellen Sie sicher, dass Go Binärdateien mit Standardeinstellungen, einschließlich der Build-ID, erstellt wurden, um genaue Abhängigkeiten zu überprüfen.

Rust-Binärdateien

Der Amazon Inspector SBOM Generator scannt nur Binärdateien, wenn die Rust Binärdateien mit der Cargo-Auditable Library erstellt wurden. RustBinärdateien, die diese Bibliothek nicht verwenden,

verfügen nicht über die notwendigen Metadaten für eine genaue Extraktion von Abhängigkeiten. Der Amazon Inspector SBOM Generator extrahiert die kompilierte Rust Toolchainversion ab Version Rust 1.7.3, jedoch nur für Binärdateien in einer Umgebung. Linux Für umfassende Scans sollten Sie Binärdateien mithilfe von cargo-auditable erstellenRust. Linux

> **Note**
>
> Die Erkennung von Sicherheitslücken für die Rust Toolchain selbst wird nicht unterstützt, selbst wenn die Version der Toolchain extrahiert wurde.

Umfassende Ökosystemsammlung von Amazon Inspector SBOM Generator

Der Amazon Inspector SBOM Generator ist ein Tool zum Erstellen einer Software-Stückliste (SBOM) und zum Durchführen von Sicherheitslücken nach unterstützten Paketen aus Betriebssystemen und Programmiersprachen. Er unterstützt auch das Scannen verschiedener Ökosysteme außerhalb der Kernbetriebssysteme und gewährleistet so eine robuste und detaillierte Analyse der Infrastrukturkomponenten. Durch die Generierung einer SBOM können Benutzer die Zusammensetzung ihrer modernen Technologie-Stacks nachvollziehen, Schwachstellen in Ökosystemkomponenten identifizieren und sich Einblick in Software von Drittanbietern verschaffen.

Unterstützte Ökosysteme

Die Ökosystemsammlung erweitert die SBOM-Generierung über Pakete hinaus, die über Betriebssystem-Paketmanager installiert wurden. Dies erfolgt durch die Sammlung von Anwendungen, die mit alternativen Methoden, wie z. B. manueller Installation, bereitgestellt werden. Der Amazon Inspector SBOM Generator unterstützt das Scannen für die folgenden Ökosysteme:

Ökosysteme	Anwendungen
Oracle Java	JDK JRE Amazon Corretto
Apache	httpd

Ökosysteme	Anwendungen
	Tomcat
WordPress	core Plugin Thema
Google	Chrome
Node.JS	node

ApacheSammlung Ökosysteme

Der Amazon Inspector SBOM Generator sucht nach Apache Installationen, die plattformübergreifend gemeinsame Installationspfade haben:

- macOS: `/Library/`
- Linux: `/etc/`, `/usr/share`, `/usr/lib`, `/usr/local`, `/var`, `/opt`

Unterstützte Anwendungen

- httpd
- tomcat

Schlüsselfeatures

- Apache httpd— Analysiert die `/include/ap_release.h` Datei, um Installationsmakros zu extrahieren, die Hauptkennungszeichenfolgen, Nebenkennungszeichenfolgen und Patch-Identifikationszeichenfolgen enthalten.
- Apache tomcat— Entpackt die `catalina.jar` Datei, um die Installationsmakros aus der Datei (`META-INF/MANIFEST.MF`) zu extrahieren, die die Versionszeichenfolge enthält.

`ap_release.h`-Beispieldatei

Das Folgende ist ein Beispiel für den Inhalt der `ap_release.h` Datei.

```
//truncated

#define AP_SERVER_BASEVENDOR "Apache Software Foundation"
#define AP_SERVER_BASEPROJECT "Apache HTTP Server"
#define AP_SERVER_BASEPRODUCT "Apache"

#define AP_SERVER_MAJORVERSION_NUMBER 2
#define AP_SERVER_MINORVERSION_NUMBER 4
#define AP_SERVER_PATCHLEVEL_NUMBER   1
#define AP_SERVER_DEVBUILD_BOOLEAN    0

//truncated
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für eine `Apache httpd` Anwendung.

```
Sample PURL: pkg:generic/apache/httpd@2.4.1
```

`catalina.jar/META-INF/MANIFEST.MF`-Beispieldatei

Das Folgende ist ein Beispiel für den Inhalt der `catalina.jar/META-INF/MANIFEST.MF` Datei.

```
//truncated

Implementation-Title: Apache Tomcat
Implementation-Vendor: Apache Software Foundation
Implementation-Version: 10.1.31

//truncated
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für eine `Apache Tomcat` Anwendung.

```
Sample PURL: pkg:generic/apache/tomcat@10.1.31
```

JavaSammlung von Ökosystemen

Unterstützte Anwendungen

- Oracle JDK
- Oracle JRE
- Amazon Corretto

Schlüsselfeatures

- Extrahiert die Zeichenfolge der Java Installation.
- Identifiziert den Verzeichnispfad, der die Java Laufzeit enthält.
- Identifiziert den Anbieter als Oracle JDKOracle JRE, undAmazon Corretto.

Der Amazon Inspector SBOM Generator sucht auf den folgenden Installationspfaden und Plattformen nach Java Installationen:

- macOS: `/Library/Java/JavaVirtualMachines`
- Linux 32-bit: `/usr/lib/jvm`
- Linux 64-bit: `/usr/lib64/jvm`
- Linux (generic): `/usr/java and /opt/java`

Beispiel für Versionsinformationen Java

Im Folgenden finden Sie ein Beispiel für eine Oracle Java Version.

```
// Amazon Corretto
IMPLEMENTOR="Amazon.com Inc."
IMPLEMENTOR_VERSION="Corretto-17.0.11.9.1"
JAVA_RUNTIME_VERSION="17.0.11+9-LTS"
JAVA_VERSION="17.0.11"
JAVA_VERSION_DATE="2024-04-16"
LIBC="default"
```

```
MODULES="java.base java.compiler java.datatransfer java.xml java.prefs java.desktop
 java.instrument java.logging java.management java.security.sasl java.naming
 java.rmi java.management.rmi java.net.http java.scripting java.security.jgss
 java.transaction.xa java.sql java.sql.rowset java.xml.crypto java.se java.smartcardio
 jdk.accessibility jdk.internal.jvmstat jdk.attach jdk.charsets jdk.compiler
 jdk.crypto.ec jdk.crypto.cryptoki jdk.dynalink jdk.internal.ed jdk.editpad
 jdk.hotspot.agent jdk.httpserver jdk.incubator.foreign jdk.incubator.vector
 jdk.internal.le jdk.internal.opt jdk.internal.vm.ci jdk.internal.vm.compiler
 jdk.internal.vm.compiler.management jdk.jartool jdk.javadoc jdk.jcmd jdk.management
 jdk.management.agent jdk.jconsole jdk.jdeps jdk.jdwp.agent jdk.jdi jdk.jfr jdk.jlink
 jdk.jpackage jdk.jshell jdk.jsobject jdk.jstatd jdk.localedata jdk.management.jfr
 jdk.naming.dns jdk.naming.rmi jdk.net jdk.nio.mapmode jdk.random jdk.sctp
 jdk.security.auth jdk.security.jgss jdk.unsupported jdk.unsupported.desktop
 jdk.xml.dom jdk.zipfs"
OS_ARCH="x86_64"
OS_NAME="Darwin"
SOURCE=".:git:7917f11551e8+"

// JDK
IMPLEMENTOR="Oracle Corporation"
JAVA_VERSION="19"
JAVA_VERSION_DATE="2022-09-20"
LIBC="default"
MODULES="java.base java.compiler java.datatransfer java.xml java.prefs java.desktop
 java.instrument java.logging java.management java.security.sasl java.naming
 java.rmi java.management.rmi java.net.http java.scripting java.security.jgss
 java.transaction.xa java.sql java.sql.rowset java.xml.crypto java.se java.smartcardio
 jdk.accessibility jdk.internal.jvmstat jdk.attach jdk.charsets jdk.zipfs jdk.compiler
 jdk.crypto.ec jdk.crypto.cryptoki jdk.dynalink jdk.internal.ed jdk.editpad
 jdk.hotspot.agent jdk.httpserver jdk.incubator.concurrent jdk.incubator.vector
 jdk.internal.le jdk.internal.opt jdk.internal.vm.ci jdk.internal.vm.compiler
 jdk.internal.vm.compiler.management jdk.jartool jdk.javadoc jdk.jcmd jdk.management
 jdk.management.agent jdk.jconsole jdk.jdeps jdk.jdwp.agent jdk.jdi jdk.jfr jdk.jlink
 jdk.jpackage jdk.jshell jdk.jsobject jdk.jstatd jdk.localedata jdk.management.jfr
 jdk.naming.dns jdk.naming.rmi jdk.net jdk.nio.mapmode jdk.random jdk.sctp
 jdk.security.auth jdk.security.jgss jdk.unsupported jdk.unsupported.desktop
 jdk.xml.dom"
OS_ARCH="x86_64"
OS_NAME="Darwin"
SOURCE=".:git:53b4a11304b0 open:git:967a28c3d85f"
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für eine Oracle Java Version.

```
Sample PURL:
# Amazon Corretto
pkg:generic/amazon/amazon-corretto@21.0.3
# Oracle JDK
pkg:generic/oracle/jdk@11.0.16
# Oracle JRE
pkg:generic/oracle/jre@20
```

GoogleSammlung von Ökosystemen

Unterstützte Anwendung

- Google Chrome

Unterstützte Artefakte

Amazon Inspector sammelt Google Chrome Informationen aus den folgenden Quellen:

- Die `chrome/VERSION` Datei (Build-Quelle)
- Die `puppeteer` Datei (Installation)

Der Amazon Inspector SBOM Generator analysiert und sammelt die entsprechenden Versionen der einzelnen unterstützten Artefakte.

Beispiel für eine Versionsdatei **`chrome/VERSION`**

Im Folgenden finden Sie ein Beispiel für die `chrome/VERSION` Versionsdatei.

```
MAJOR=130
MINOR=0
BUILD=6723
PATCH=58
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für eine `chrome/VERSION` Versionsdatei.

```
Sample PURL: pkg:generic/google/chrome@131.0.6778.87
```

Beispiel für **`puppeteer`** eine Versionsdatei

Im Folgenden finden Sie ein Beispiel für die `puppeteer` Versionsdatei.

```
{
"name": "puppeteer",
"version": "23.9.0",
"description": "A high-level API to control headless Chrome over the DevTools
 Protocol",
"keywords": [
  "puppeteer",
  "chrome",
  "headless",
  "automation"
]
}
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für eine `puppeteer` Versionsdatei.

```
Sample PURL: pkg:generic/google/puppeteer@23.9.0
```

WordPressSammlung von Ökosystemen

Unterstützte Komponenten

- WordPress-Core
- WordPressPlugins
- WordPressThemen

Schlüsselfeatures

- WordPresscore — analysiert die `/wp-includes/version.php` Datei, um den Versionswert aus der Variablen $wp_version zu extrahieren.
- WordPressplugins — analysiert die `/wp-content/plugins/<WordPress Plugin>/readme.txt` Datei oder `/wp-content/plugins/<WordPress Plugin>/readme.md` Datei, um das `Stable` Tag als Versionsstring zu extrahieren.
- WordPressthemes — analysiert die `/wp-content/themes/<WordPress Theme>/style.css` Datei, um die Version aus den Versionsmetadaten zu extrahieren.

`version.php`-Beispieldatei

Das Folgende ist ein Beispiel für eine WordPress `version.php` Core-Datei.

```
// truncated

/**
* The WordPress version string.
*
* Holds the current version number for WordPress core. Used to bust caches
* and to enable development mode for scripts when running from the /src directory.
*
* @global string $wp_version
*/
$wp_version = '6.5.5';

// truncated
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für WordPress Core.

```
Sample PURL: pkg:generic/wordpress/core/wordpress@6.5.5
```

`readme.txt`-Beispieldatei

Das Folgende ist ein Beispiel für eine WordPress `readme.txt` Plugin-Datei.

```
=== Plugin Name ===
Contributors: (this should be a list of wordpress.org userid's)
Donate link: https://example.com/
Tags: tag1, tag2
Requires at least: 4.7
Tested up to: 5.4
Stable tag: 4.3
Requires PHP: 7.0
License: GPLv2 or later
License URI: https://www.gnu.org/licenses/gpl-2.0.html

// truncated
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für ein WordPress Plugin.

```
Sample PURL: pkg:generic/wordpress/plugin/exclusive-addons-for-elementor@1.0.0
```

`style.css`-Beispieldatei

Das Folgende ist ein Beispiel für eine WordPress `style.css` Theme-Datei.

```
/*
Author: the WordPress team
Author URI: https://wordpress.org
Description: Twenty Twenty-Four is designed to be flexible, versatile and applicable
 to any website. Its collection of templates and patterns tailor to different needs,
 such as presenting a business, blogging and writing or showcasing work. A multitude
 of possibilities open up with just a few adjustments to color and typography. Twenty
 Twenty-Four comes with style variations and full page designs to help speed up the
 site building process, is fully compatible with the site editor, and takes advantage
 of new design tools introduced in WordPress 6.4.
Requires at least: 6.4
Tested up to: 6.5
Requires PHP: 7.0
Version: 1.2
```

```
License: GNU General Public License v2 or later
License URI: http://www.gnu.org/licenses/gpl-2.0.html
Text Domain: twentytwentyfour
Tags: one-column, custom-colors, custom-menu, custom-logo, editor-style, featured-
images, full-site-editing, block-patterns, rtl-language-support, sticky-post,
 threaded-comments, translation-ready, wide-blocks, block-styles, style-variations,
 accessibility-ready, blog, portfolio, news
*/
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für ein WordPress Thema.

```
Sample PURL: pkg:generic/wordpress/theme/avada@1.0.0
```

Node.JSRuntime-Sammlung

Unterstützte Anwendungen

- Knotenlaufzeit-Binärdatei für Node.JS

Unterstützte Artefakte

- MacOSund Linux — Erkennung von `node` Binärdateien anhand von Binärdetails, die mit `asdf`fnm,nvm,volta, oder offiziellen Node.JS Containern installiert wurden.

Beispiel MacOS und Linux Pfade

Im Folgenden finden Sie ein Beispiel für Pfade für MacOS undLinux.

```
NVM:   ~/.nvm/, /usr/local/nvm
FNM:   ~/.local/share/fnm/
ASDF:  ~/.asdf/
MISE:  ~/.local/share/mise/
VOLTA: ~/.volta/
```

Beispiel PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL fürNode.JS.

```
Sample PURL: pkg:generic/nodejs/node@20.18.0
```

OpenSSL-Ökosystemsammlung

Unterstützte Anwendungen

Die Support für OpenSSL-Bibliotheken und Entwicklungspakete ist auf Software beschränkt, die mit offiziellem OpenSSL für 3.0.0-Versionen und höher erstellt wurde. Die Software muss außerdem der semantischen Versionierung folgen. Benutzerdefinierte oder geforkte OpenSSL-Varianten und Versionen unter 3.0.0 werden nicht unterstützt.

Der Amazon Inspector SBOM Generator extrahiert wichtige Paketinformationen für jede installierte OpenSSL-Instance.

Schlüsselfeatures

- Extrahiert die Basiszeichenfolge der SEMVER-Version aus der OpenSSL-Header-Datei
- Identifiziert den Verzeichnispfad, der die OpenSSL-Installation enthält

Der Amazon Inspector SBOM Generator sucht nach OpenSSL-Installationen, indem er plattformübergreifend in gängigen Installationspfaden nach der `openssl.h` Datei sucht.

Beispiel für einen Installationspfad für Linux/Unix

Im Folgenden finden Sie ein Beispiel für einen Installationspfad für Linux/Unix.

```
/usr/local/include/openssl/opensslv.h
/usr/local/ssl/include/openssl/opensslv.h
/usr/local/openssl/include/openssl/opensslv.h
/usr/local/opt/openssl/include/openssl/opensslv.h
/usr/include/openssl/opensslv.h
```

Der Amazon Inspector SBOM Generator extrahiert Versionsinformationen, indem er die `openssl.h` Datei analysiert und nach den Versionsdefinitionen sucht.

```
# define OPENSSL_VERSION_MAJOR  3
# define OPENSSL_VERSION_MINOR  4
# define OPENSSL_VERSION_PATCH  0
```

Beispiel für eine PURL

Im Folgenden finden Sie ein Beispiel für eine Paket-URL für die OpenSSL-Version.

```
Sample PURL: pkg:generic/openssl/openssl@3.4.0
```

Sammlung von Amazon Inspector SBOM Generator-Lizenzen

Der Amazon Inspector SBOM Generator hilft dabei, Lizenzinformationen in einer Software-Stückliste (SBOM) nachzuverfolgen. Er sammelt Lizenzinformationen aus unterstützten Paketen für alle Betriebssysteme und Programmiersprachen. Mit standardisierten Lizenzausdrücken in Ihrer generierten SBOM können Sie Ihre Lizenzverpflichtungen nachvollziehen.

Sammeln Sie Lizenzinformationen

-Beispielbefehl

Das folgende Beispiel zeigt, wie Lizenzinformationen aus einem Verzeichnis gesammelt werden.

```
./inspector-sbomgen directory --path /path/to/your/directory/ --collect-licenses
```

Beispiel für eine SBOM-Komponente

Das folgende Beispiel zeigt einen Komponenteneintrag in der generierten SBOM.

```
"components": [
    {
      "bom-ref": "comp-2",
      "type": "application",
      "name": "sample-js-pkg",
      "version": "1.2.3",
      "licenses": [
        {
          "expression": "Apache-2.0 AND (MIT OR GPL-2.0-only)"
        }
```

```
      ],
      "purl": "pkg:npm/sample-js-pkg@1.2.3",
    }
  ]
```

Unterstützte Pakete

Die folgenden Programmiersprachen und Betriebssystempakete werden für die Lizenzerfassung unterstützt.

Ziel	Paketmanager	Quelle für Lizenzinformationen	Typ
Alma Linux	RPM	• /usr/lib/sysimage/rpm/rpmdb.sqlite • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS
Amazon Linux	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb	BS

Ziel	Paketmanager	Quelle für Lizenzinformationen	Typ
		• /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	
CentOS	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS
Fedora	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS

Ziel	Paketmanager	Quelle für Lizenzinformationen	Typ
OpenSUSE	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS
Oracle Linux	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS

Ziel	Paketmanager	Quelle für Lizenzinformationen	Typ
Photon OS	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS
RHEL	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS

Ziel	Paketmanager	Quelle für Lizenzinformationen	Typ
Rocky Linux	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS
SLES	RPM	• /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages	BS
Alpine Linux	APK	`/lib/apk/db/installed`	BS

Ziel	Paketmanager	Quelle für Lizenzinformationen	Typ
Chainguard	APK	`/lib/apk/db/installed`	BS
Debian	DPKG	`/usr/share/doc/*/copyright`	BS
Ubuntu	DPKG	`/usr/share/doc/*/copyright`	BS
Node.js	Javascript	`node_modules/*/package.json`	Programmiersprache
PHP	Composer-Paket	• `composer.lock` • `/vendor/composer/installed.json`	Programmiersprache
Go	Go	`LICENSE`	Programmiersprache
Python	Python/Egg/Wheel	• `.dist-info/METADATA` • `.egg-info` • `.egg-info/PKG-INFO`	Programmiersprache
Ruby	RubyGem	`*.gemspec`	Programmiersprache
Rust	crate	`Cargo.toml`	Programmiersprache

Standardisierung von Lizenzausdrücken

Das SPDX-Lizenzausdrucksformat bietet eine genaue Darstellung der Lizenzbedingungen, die in Open-Source-Software zu finden sind. Der Amazon Inspector SBOM Generator standardisiert

alle Lizenzinformationen anhand der in diesem Abschnitt beschriebenen Regeln in SPDX-Lizenzausdrücken. Die Regeln sorgen für Konsistenz und Kompatibilität aller Lizenzinformationen.

Zuordnung von SPDX-Kurzform-IDs

Alle Lizenznamen sind SPDX-Kurzformkennungen zugeordnet. `MIT License` wurde beispielsweise zu `MIT` verkürzt.

Kombination aus mehreren Lizenzen

Sie können mehr als eine Lizenz mit dem `AND` Betreiber kombinieren. Im Folgenden finden Sie einen Beispielbefehl, der zeigt, wie Sie Ihren Befehl formatieren.

```
MIT AND Apache-2.0
```

Benutzerdefiniertes Lizenzpräfix

Benutzerdefinierten Lizenzen wird ein Präfix `LicenseRef` vorangestellt, wie `LicenseRef-CompanyPrivate` z.

Benutzerdefiniertes Ausnahmepräfix

Benutzerdefinierten Ausnahmen wird ein Präfix `AdditionRef-` vorangestellt, wie `AdditionRef-CustomException` z.

Was ist eine Paket-URL?

Eine Paket-URL oder PURL ist ein standardisiertes Format, das zur Identifizierung von Softwarepaketen, Komponenten und Bibliotheken in verschiedenen Paketverwaltungssystemen verwendet wird. Das Format erleichtert das Nachverfolgen, Analysieren und Verwalten von Abhängigkeiten in Softwareprojekten, insbesondere bei der Generierung einer Softwareliste (SBOMs).

PURL-Struktur

Die PURL-Struktur ähnelt einer URL und besteht aus mehreren Komponenten:

- `pkg`— Das wörtliche Präfix

- `type`— Der Pakettyp
- `namespace`— Die Gruppierung
- `name`— Der Paketname
- `version`— Die Paketversion
- `qualifiers`— Zusätzliche Schlüssel-Wert-Paare
- `subpath`— Der Dateipfad im Paket

Beispiel für eine PURL

Das Folgende ist ein Beispiel dafür, wie eine PURL aussehen könnte.

```
pkg:<type>/<namespace>/<name>@<version>?<qualifiers>#<subpath>
```

Die generische PURL

Eine generische PURL wird verwendet, um Softwarepakete und Komponenten darzustellen, die nicht in etablierte Paket-Ökosysteme passen, wie npmpypi, oder. maven Sie identifiziert Softwarekomponenten und erfasst Metadaten, die möglicherweise nicht mit bestimmten Paketverwaltungssystemen übereinstimmen. Eine generische PURL ist für eine Vielzahl von Softwareprojekten nützlich, von kompilierten Binärdateien bis hin zu Plattformen wie Apache und. WordPress Sie ermöglicht die Anwendung in einer Vielzahl von Anwendungsfällen, einschließlich kompilierter Binärdateien, Webplattformen und kundenspezifischer Softwareverteilungen.

Wichtige Anwendungsfälle

- Unterstützt kompilierte Binärdateien und ist nützlich für Go und Rust
- Unterstützt Webplattformen wie Apache undWordPress, bei denen ein Paket möglicherweise nicht mit herkömmlichen Paketmanagern verknüpft ist.
- Unterstützt benutzerdefinierte Legacy-Software, indem Organisationen auf intern entwickelte Software oder auf Systeme verweisen können, denen formelle Pakete fehlen.

Beispielformat

Das Folgende ist ein Beispiel für das generische PURL-Format.

```
pkg:generic/<namespace>/<name>@<version>?<qualifiers>
```

Zusätzliche Beispiele für das generische PURL-Format

Im Folgenden finden Sie weitere Beispiele für das generische PURL-Format.

Kompiliertes Binär Go

Das Folgende steht für das `inspector-sbomgen binary` Kompilierte mit einemGo.

```
pkg:generic/inspector-sbomgen?go_toolchain=1.22.5
```

Kompilierte Rust Binär

Im Folgenden wird die `myrustapp` Binärdatei dargestellt, mit der kompiliert wurdeRust.

```
pkg:generic/myrustapp?rust_toolchain=1.71.0
```

Apache-Projekt

Das Folgende bezieht sich auf ein HTTP-Projekt unter dem Apache Namespace.

```
pkg:generic/apache/httpd@1.0.0
```

WordPress-Software

Das Folgende bezieht sich auf eine WordPress Kernsoftware.

```
pkg:generic/wordpress/core/wordpress@6.0.0
```

WordPressThema

Das Folgende bezieht sich auf ein benutzerdefiniertes WordPress Thema.

```
pkg:generic/wordpress/theme/mytheme@1.0.0
```

WordPress-Plugin

Das Folgende bezieht sich auf ein benutzerdefiniertes WordPress Plugin.

```
pkg:generic/wordpress/plugin/myplugin@1.0.0
```

Umgang mit ungelösten oder nicht standardmäßigen Versionsreferenzen im Amazon Inspector SBOM Generator

Der Amazon Inspector SBOM Generator lokalisiert und analysiert unterstützte Artefakte innerhalb eines Systems, indem er Abhängigkeiten direkt aus Quelldateien identifiziert. Er ist kein Paketmanager und löst keine Versionsbereiche auf, leitet keine Versionen auf der Grundlage dynamischer Verweise ab und verarbeitet auch keine Abfragen in der Registrierung. Es erfasst Abhängigkeiten nur so, wie sie in den Artefakten der Projektquelle definiert sind. In vielen Fällen werden Abhängigkeiten in Paketmanifesten, wie`package.json`, oder `pom.xml`requirements.txt, mit ungelösten oder bereichsbasierten Versionen spezifiziert. Dieses Thema enthält Beispiele dafür, wie diese Abhängigkeiten aussehen könnten.

Empfehlungen

Der Amazon Inspector SBOM Generator extrahiert Abhängigkeiten aus Quellartefakten, löst oder interpretiert jedoch weder Versionsbereiche noch dynamische Verweise. Für genauere Sicherheitslücken empfehlen wir die Verwendung von bearbeiteten, semantischen Versions-IDs in Projektabhängigkeiten. SBOMs

Java

Denn Maven Projekte können Versionsbereiche verwendenJava, um Abhängigkeiten in der `pom.xml` Datei zu definieren.

```
<dependency>
    <groupId>org.inspector</groupId>
    <artifactId>inspector-api</artifactId>
    <version>(,1.0]</version>
</dependency>
```

Der Bereich gibt an, dass jede Version bis einschließlich 1.0 zulässig ist. Wenn es sich bei einer Version jedoch nicht um eine aufgelöste Version handelt, erfasst der Amazon Inspector SBOM Generator sie nicht, da sie keiner bestimmten Version zugeordnet werden kann.

JavaScript

Denn JavaScript die `package.json` Datei kann Versionsbereiche enthalten, die den folgenden ähneln:

```
"dependencies": {
    "ky": "^1.2.0",
    "registry-auth-token": "^5.0.2",
    "registry-url": "^6.0.1",
    "semver": "^7.6.0"
}
```

Der ^ Operator gibt an, dass jede Version, die größer oder gleich der angegebenen Version ist, zulässig ist. Wenn es sich bei der angegebenen Version jedoch nicht um eine aufgelöste Version handelt, erfasst der Amazon Inspector SBOM Generator sie nicht, da dies zu Fehlalarmen bei der Erkennung von Sicherheitslücken führen kann.

Python

Denn Python die `requirements.txt` Datei kann Einträge mit einem booleschen Ausdruck enthalten.

```
requests>=1.0.0
```

Der >= Operator gibt an, dass jede Version, die größer oder gleich `1.0.0` ist, zulässig ist. Da dieser spezielle Ausdruck keine genaue Version angibt, kann der Amazon Inspector SBOM Generator keine zuverlässige Version für die Schwachstellenanalyse sammeln.

Der Amazon Inspector SBOM Generator unterstützt keine nicht standardmäßigen oder mehrdeutigen Versionskennungen wie Beta, Latest oder Snapshot.

```
pkg:maven/org.example.com/testmaven@1.0.2%20Beta-RC-1_Release
```

> **Note**
>
> Die Verwendung eines nicht standardmäßigen Suffixes, wie Beta-RC-1_Release z. B., entspricht nicht der standardmäßigen semantischen Versionierung und kann nicht auf Sicherheitslücken innerhalb der Amazon Inspector Detection Engine untersucht werden.

Die Verwendung von CycloneDX Namespaces mit Amazon Inspector

Amazon Inspector bietet Ihnen CycloneDX Namespaces und Eigenschaftsnamen, die Sie zusammen verwenden können. SBOMs In diesem Abschnitt werden alle benutzerdefinierten Schlüssel-/Werteigenschaften beschrieben, die Komponenten in hinzugefügt werden können CycloneDX SBOMs. Weitere Informationen finden Sie unter CyclonedX-Eigenschaftstaxonomie auf der GitHub Webseite.

`amazon:inspector:sbom_scanner`Namespace-Taxonomie

Die Amazon Inspector Scan API verwendet den `amazon:inspector:sbom_scanner` Namespace und hat die folgenden Eigenschaften:

Eigenschaft	Beschreibung
`amazon:inspector:sbom_scanner:cisa_kev_date_added`	Gibt an, wann die Sicherheitsanfälligkeit dem Katalog der CISA Known Exploited Vulnerabilities hinzugefügt wurde.
`amazon:inspector:sbom_scanner:cisa_kev_date_due`	Gibt an, wann die Behebung der Sicherheitslücke gemäß dem CISA-Katalog mit den bekannten Sicherheitslücken fällig ist.
`amazon:inspector:sbom_scanner:critical_vulnerabilities`	Anzahl der Gesamtzahl der in der SBOM gefundenen Sicherheitslücken mit kritischem Schweregrad.
`amazon:inspector:sbom_scanner:exploit_available`	Gibt an, ob ein Exploit für die angegebene Sicherheitsanfälligkeit verfügbar ist.

Eigenschaft	Beschreibung
`amazon:inspector:sbom_scanner:exploit_last_seen_in_public`	Gibt an, wann ein Exploit für die angegebene Sicherheitsanfälligkeit zuletzt öffentlich bekannt wurde.
`amazon:inspector:sbom_scanner:fixed_version: component_bom_ref`	Stellt die behobene Version der angegebenen Komponente für die angegebene Sicherheitsanfälligkeit bereit.
`amazon:inspector:sbom_scanner:high_vulnerabilities`	Anzahl der Gesamtzahl der in der SBOM gefundenen Sicherheitslücken mit hohem Schweregrad.
`amazon:inspector:sbom_scanner:info`	Stellt den Scankontext für eine bestimmte Komponente bereit, zum Beispiel: „Komponente gescannt: Keine Sicherheitslücken gefunden“.
`amazon:inspector:sbom_scanner:is_malicious`	Zeigt an, ob OpenSSF die betroffenen Komponenten als bösartig identifiziert.
`amazon:inspector:sbom_scanner:low_vulnerabilities`	Anzahl der Gesamtzahl der in der SBOM gefundenen Sicherheitslücken mit geringem Schweregrad.
`amazon:inspector:sbom_scanner:medium_vulnerabilities`	Anzahl der Gesamtzahl der in der SBOM gefundenen Sicherheitslücken mit mittlerem Schweregrad.
`amazon:inspector:sbom_scanner:path`	Der Pfad zu der Datei, die die Betreff-Paketinformationen enthält.
`amazon:inspector:sbom_scanner:priority`	Die empfohlene Priorität für die Behebung einer bestimmten Sicherheitsanfälligkeit. Die Werte in absteigender Reihenfolge sind „IMMEDIATE“, „URGENT“, „MODERATE“ und „STANDARD“.

Eigenschaft	Beschreibung
`amazon:inspector:sbom_scanner:priority_intelligence`	Die Qualität der Informationen, anhand derer die Priorität einer bestimmten Sicherheitslücke bestimmt wird. Zu den Werten gehören „VERIFIED“ oder „UNVERIFIED“.
`amazon:inspector:sbom_scanner:warning`	Stellt einen Kontext dafür bereit, warum eine bestimmte Komponente nicht gescannt wurde, zum Beispiel: „Komponente übersprungen: keine URL angegeben“.

`amazon:inspector:sbom_generator`Namespace-Taxonomie

Der Amazon Inspector SBOM Generator verwendet den `amazon:inspector:sbom_generator` Namespace und hat die folgenden Eigenschaften:

Eigenschaft	Beschreibung
`amazon:inspector:sbom_generator:cpu_architecture`	Die CPU-Architektur des Systems, das inventarisiert wird (x86_64).
`amazon:inspector:sbom_generator:ec2:instance_id`	Die EC2 Amazon-Instance-ID.
`amazon:inspector:sbom_generator:live_patching_enabled`	Ein boolescher Wert, der angibt, ob Live-Patching bei Amazon Amazon aktiviert ist EC2 Linux.
`amazon:inspector:sbom_generator:live_patched_cves`	Eine Liste der durch CVEs Live-Patches bei Amazon gepatchten EC2 Linux.
`amazon:inspector:sbom_generator:dockerfile_finding: inspector_finding_id`	Zeigt an, dass ein Amazon Inspector Inspector-Ergebnis in einer Komponente zusammenhängt mit Dockerfile prüft.

Eigenschaft	Beschreibung
`amazon:inspector:sbom_generator:image_id`	Der Hash, der zur Container-Image-Konfigurationsdatei gehört (auch bekannt als Image-ID).
`amazon:inspector:sbom_generator:image_arch`	Die Architektur des Container-Images.
`amazon:inspector:sbom_generator:image_author`	Der Autor des Container-Images.
`amazon:inspector:sbom_generator:image_docker_version`	Die Docker-Version, die zum Erstellen des Container-Images verwendet wurde.
`amazon:inspector:sbom_generator:is_duplicate_package`	Zeigt an, dass das Betreff-Paket von mehr als einem Dateiscanner gefunden wurde.
`amazon:inspector:sbom_generator:duplicate_purl`	Zeigt die duplizierte Paket-URL an, die von einem anderen Scanner gefunden wurde.
`amazon:inspector:sbom_generator:kernel_name`	Der Kernelname des Systems, das inventarisiert wird.
`amazon:inspector:sbom_generator:kernel_version`	Die Kernelversion des Systems, das inventarisiert wird.
`amazon:inspector:sbom_generator:kernel_component`	Ein boolescher Wert, der angibt, ob es sich bei einem Betreff-Paket um eine Kernelkomponente handelt
`amazon:inspector:sbom_generator:running_kernel`	Ein boolescher Wert, der angibt, ob es sich bei einem Betreff-Paket um den laufenden Kernel handelt
`amazon:inspector:sbom_generator:layer_diff_id`	Der Hash der unkomprimierten Container-Image-Ebene.
`amazon:inspector:sbom_generator:replaced_by`	Der Wert, der den aktuellen ersetzt Go Modul.

Eigenschaft	Beschreibung
`amazon:inspector:sbom_generator:os_hostname`	Der Hostname des Systems, das inventarisiert wird.
`amazon:inspector:sbom_generator:source_file_scanner`	Der Scanner, der die Datei gefunden hat, die Paketinformationen enthält, zum Beispiel:. `/var/lib/dpkg/status`
`amazon:inspector:sbom_generator:source_package_collector`	Der Collector, der den Paketnamen und die Version aus einer bestimmten Datei extrahiert hat.
`amazon:inspector:sbom_generator:source_path`	Der Pfad zu der Datei, aus der die Betreff-Paketinformationen extrahiert wurden.
`amazon:inspector:sbom_generator:file_size_bytes`	Gibt die Dateigröße eines bestimmten Artefakts an.
`amazon:inspector:sbom_generator:unresolved_version`	Zeigt eine Versionszeichenfolge an, die nicht vom Paketmanager aufgelöst wurde.
`amazon:inspector:sbom_generator:experimental:transitive_dependency`	Weist auf indirekte Abhängigkeiten von einem Paketmanager hin.

Integration von Amazon Inspector-Scans in Ihre CI/CD Pipeline

Die Amazon CI/CD Inspector-Integration verwendet den Amazon Inspector SBOM Generator und die Amazon Inspector Scan API, um Schwachstellenberichte für Container-Images zu erstellen. Der Amazon Inspector SBOM Generator erstellt eine Software-Stückliste (SBOM) für Archive, Container-Images, Verzeichnisse, lokale Systeme sowie kompilierte Dateien und Binärdateien. Go Rust Die Amazon Inspector Scan API scannt die SBOM, um einen Bericht mit Details zu erkannten Sicherheitslücken zu erstellen. Sie können Amazon Inspector Container-Image-Scans in Ihre CI/CD Pipeline integrieren, um nach Softwareschwachstellen zu suchen und Schwachstellenberichte zu erstellen, mit denen Sie Risiken vor der Bereitstellung untersuchen und beheben können. Um Ihre CI/CD Integration einzurichten, können Sie Plugins verwenden oder mit dem Amazon Inspector SBOM Generator und der Amazon Inspector Scan API eine benutzerdefinierte CI/CD Integration erstellen.

Themen

Plugin-Integration

Amazon Inspector bietet Plugins für unterstützte CI/CD Lösungen. Sie können diese Plugins von ihren jeweiligen Marketplaces aus installieren und sie dann verwenden, um Amazon Inspector Scans als Build-Schritt in Ihre Pipeline aufzunehmen. Im Schritt zur Plugin-Erstellung wird der Amazon

Inspector SBOM-Generator auf dem von Ihnen bereitgestellten Bild ausgeführt und anschließend die Amazon Inspector Scan-API auf der generierten SBOM ausgeführt.

Im Folgenden finden Sie eine Übersicht darüber, wie eine Amazon Inspector CI/CD Inspector-Integration über Plugins funktioniert:

1. Sie konfigurieren eine AWS-Konto , um den Zugriff auf die Amazon Inspector Scan API zu ermöglichen. Detaillierte Anweisungen finden Sie unter Einrichtung eines AWS Kontos für die Nutzung der Amazon Inspector CI/CD Inspector-Integration.
2. Sie installieren das Amazon Inspector-Plugin vom Marketplace.
3. Sie installieren und konfigurieren die Amazon Inspector SBOM Generator-Binärdatei. Detaillierte Anweisungen finden Sie unter Amazon Inspector SBOM-Generator.
4. Sie fügen Amazon Inspector Scans als Build-Schritt in Ihre CI/CD Pipeline ein und konfigurieren den Scan.
5. Wenn Sie einen Build ausführen, verwendet das Plugin Ihr Container-Image als Eingabe und führt dann den Amazon Inspector SBOM Generator auf dem Image aus, um eine CycloneDX kompatible SBOM zu generieren.
6. Von dort aus sendet das Plugin die generierte SBOM an einen Amazon Inspector Scan API-Endpunkt, der jede SBOM-Komponente auf Sicherheitslücken überprüft.
7. Die Antwort der Amazon Inspector Scan API wird in einen Schwachstellenbericht in den Formaten CSV, SBOM, JSON und HTML umgewandelt. Der Bericht enthält Details zu allen Sicherheitslücken, die Amazon Inspector gefunden hat.

Unterstützte CI/CD Lösungen

Amazon Inspector unterstützt derzeit die folgenden CI/CD Lösungen. Vollständige Anweisungen zur Einrichtung der CI/CD Integration mithilfe eines Plug-ins erhalten Sie, wenn Sie das Plug-in für Ihre CI/CD-Lösung auswählen:

- Jenkins-Plugin
- TeamCity-Plugin
- GitHub Aktionen

Maßgeschneiderte Integration

Wenn Amazon Inspector keine Plug-ins für Ihre CI/CD Lösung bereitstellt, können Sie mithilfe einer Kombination aus dem Amazon Inspector SBOM Generator und der Amazon Inspector Scan API Ihre eigene benutzerdefinierte CI/CD Integration erstellen. Sie können auch eine benutzerdefinierte Integration verwenden, um Scans mithilfe der im Amazon Inspector SBOM Generator verfügbaren Optionen zu optimieren.

Im Folgenden finden Sie einen Überblick darüber, wie eine benutzerdefinierte Amazon Inspector CI/CD Inspector-Integration funktioniert:

1. Sie konfigurieren eine AWS-Konto , um den Zugriff auf die Amazon Inspector Scan API zu ermöglichen. Detaillierte Anweisungen finden Sie unter Einrichtung eines AWS Kontos für die Nutzung der Amazon Inspector CI/CD Inspector-Integration.
2. Sie installieren und konfigurieren die Amazon Inspector SBOM Generator-Binärdatei. Detaillierte Anweisungen finden Sie unter Amazon Inspector SBOM-Generator.
3. Sie verwenden den Amazon Inspector SBOM Generator, um eine CycloneDX kompatible SBOM für Ihr Container-Image zu generieren.
4. Sie verwenden die Amazon Inspector Scan API für die generierte SBOM, um einen Schwachstellenbericht zu erstellen.

Anweisungen zum Einrichten einer benutzerdefinierten Integration finden Sie unter. Erstellen einer benutzerdefinierten CI/CD Pipeline-Integration mit Amazon Inspector Scan

Einrichtung eines AWS Kontos für die Nutzung der Amazon Inspector CI/CD Inspector-Integration

Um die Amazon Inspector CI/CD Inspector-Integration nutzen zu können, müssen Sie sich für eine registrieren AWS-Konto. Sie AWS-Konto müssen über eine IAM-Rolle verfügen, die Ihrer CI/CD Pipeline Zugriff auf die Amazon Inspector Scan API gewährt. Führen Sie die Aufgaben in den folgenden Themen aus, um sich für eine IAM-Rolle zu registrieren AWS-Konto, einen Administratorbenutzer zu erstellen und eine IAM-Rolle für die Integration zu konfigurieren. CI/CD

> **Note**
>
> Wenn Sie sich bereits für eine angemeldet haben AWS-Konto, können Sie direkt zuKonfigurieren Sie eine IAM-Rolle für die CI/CD-Integration.

Themen

Melde dich für eine an AWS-Konto

Wenn Sie noch keine haben AWS-Konto, führen Sie die folgenden Schritte aus, um eine zu erstellen.

Um sich für eine anzumelden AWS-Konto

1. Öffnen Sie https://portal.aws.amazon.com/billing/die Anmeldung.
2. Folgen Sie den Online-Anweisungen.

   Ein Teil des Anmeldevorgangs umfasst den Empfang eines Telefonanrufs oder einer Textnachricht und die Eingabe eines Bestätigungscodes auf der Telefontastatur.

   Wenn Sie sich für eine anmelden AWS-Konto, wird eine Root-Benutzer des AWS-Kontoserstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem Administratorbenutzer Administratorzugriff zu und verwenden Sie nur den Root-Benutzer, um Aufgaben auszuführen, die Root-Benutzerzugriff erfordern.

AWS sendet Ihnen nach Abschluss des Anmeldevorgangs eine Bestätigungs-E-Mail. Du kannst jederzeit deine aktuellen Kontoaktivitäten einsehen und dein Konto verwalten, indem du zu https://aws.amazon.com/gehst und Mein Konto auswählst.

Erstellen eines Benutzers mit Administratorzugriff

Nachdem Sie sich für einen angemeldet haben AWS-Konto, sichern Sie Ihren Root-Benutzer des AWS-Kontos AWS IAM Identity Center, aktivieren und erstellen Sie einen Administratorbenutzer, sodass Sie den Root-Benutzer nicht für alltägliche Aufgaben verwenden.

Sichern Sie Ihre Root-Benutzer des AWS-Kontos

1. Melden Sie sich AWS Management Consoleals Kontoinhaber an, indem Sie Root-Benutzer auswählen und Ihre AWS-Konto E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

   Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter Anmelden als Root-Benutzer im AWS-Anmeldung Benutzerhandbuch zu.
2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

   Anweisungen finden Sie unter Aktivieren eines virtuellen MFA-Geräts für Ihren AWS-Konto Root-Benutzer (Konsole) im IAM-Benutzerhandbuch.

Erstellen eines Benutzers mit Administratorzugriff

1. Aktivieren Sie das IAM Identity Center.

   Anweisungen finden Sie unter Aktivieren AWS IAM Identity Center im AWS IAM Identity Center Benutzerhandbuch.
2. Gewähren Sie einem Administratorbenutzer im IAM Identity Center Benutzerzugriff.

   Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie IAM-Identity-Center-Verzeichnis im Benutzerhandbuch unter Benutzerzugriff mit der Standardeinstellung konfigurieren.AWS IAM Identity Center

Anmelden als Administratorbenutzer

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

  Hilfe bei der Anmeldung mit einem IAM Identity Center-Benutzer finden Sie im AWS-Anmeldung Benutzerhandbuch unter Anmeldung beim AWS Access-Portal.

Weiteren Benutzern Zugriff zuweisen

1. Erstellen Sie im IAM-Identity-Center einen Berechtigungssatz, der den bewährten Vorgehensweisen für die Anwendung von geringsten Berechtigungen folgt.

   Anweisungen hierzu finden Sie unter Berechtigungssatz erstellen im AWS IAM Identity Center Benutzerhandbuch.
2. Weisen Sie Benutzer einer Gruppe zu und weisen Sie der Gruppe dann Single Sign-On-Zugriff zu.

   Eine genaue Anleitung finden Sie unter Gruppen hinzufügen im AWS IAM Identity Center Benutzerhandbuch.

Konfigurieren Sie eine IAM-Rolle für die CI/CD-Integration

Um Amazon Inspector-Scanning in Ihre CI/CD Pipeline zu integrieren, müssen Sie eine IAM-Richtlinie erstellen, die den Zugriff auf die Amazon Inspector Scan-API ermöglicht, die die Softwareliste scannt (SBOMs). Anschließend können Sie diese Richtlinie einer IAM-Rolle zuordnen, die Ihr Konto für die Ausführung der Amazon Inspector Scan API übernehmen kann.

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die IAM-Konsole unter. https://console.aws.amazon.com/iam/
2. Klicken Sie im Navigationsbereich der IAM-Konsole auf Richtlinien und wählen Sie dann Richtlinie erstellen aus.
3. Wählen Sie im Policy Editor JSON aus und fügen Sie die folgende Anweisung ein:

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

4. Wählen Sie Weiter aus.
5. Geben Sie der Richtlinie beispielsweise `InspectorCICDscan-policy` einen Namen und fügen Sie eine optionale Beschreibung hinzu. Wählen Sie dann Create Policy aus. Diese Richtlinie wird der Rolle angehängt, die Sie in den nächsten Schritten erstellen werden.
6. Wählen Sie im Navigationsbereich der IAM-Konsole Rollen und dann Neue Rolle erstellen aus.
7. Wählen Sie unter Vertrauenswürdiger Entitätstyp die Option Benutzerdefinierte Vertrauensrichtlinie aus und fügen Sie die folgende Richtlinie ein:

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

8. Wählen Sie Weiter aus.
9. Suchen Sie unter Berechtigungen hinzufügen nach der Richtlinie, die Sie zuvor erstellt haben, wählen Sie sie aus und klicken Sie dann auf Weiter.
10. Geben Sie der Rolle beispielsweise `InspectorCICDscan-role` einen Namen und fügen Sie eine optionale Beschreibung hinzu`Create Role`. Wählen Sie dann.

Amazon Inspector Dockerfile-Prüfungen

In diesem Abschnitt wird beschrieben, wie Sie den Amazon Inspector SBOM Generator verwenden, um Bilder zu scannen Dockerfiles und Docker zu speichern, um Fehlkonfigurationen zu erkennen, die zu Sicherheitslücken führen.

Themen

- Dockerfile-Prüfungen verwenden Sbomgen

- Unterstützte Dockerfile-Prüfungen

Dockerfile-Prüfungen verwenden Sbomgen

Dockerfile-Prüfungen werden automatisch durchgeführt, wenn eine Datei mit dem Namen `Dockerfile` oder entdeckt `*.Dockerfile` wird und wenn ein Docker-Image gescannt wird.

Sie können Dockerfile-Prüfungen mit dem Argument deaktivieren. `--skip-scanners dockerfile` Sie können Dockerfile-Prüfungen auch mit allen verfügbaren Scannern kombinieren, z. B. mit Betriebssystemen oder Paketen von Drittanbietern.

Beispiele für Docker-Checkbefehle

Die folgenden Beispielbefehle zeigen, wie SBOMs für Dockerfiles und Docker-Container-Images sowie für Betriebssysteme und Pakete von Drittanbietern generiert werden.

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

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

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

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

Beispiel für eine Dateikomponente

Im Folgenden finden Sie ein Beispiel für eine Dockerfile-Suche nach einer Dateikomponente.

```
{
     "bom-ref": "comp-2",
      "name": "dockerfile:data/docker/Dockerfile",
      "properties": [
        {
          "name": "amazon:inspector:sbom_scanner:dockerfile_finding:IN-DOCKER-001",
```

```
          "value": "affected_lines:27-27"
        }
      ],
      "type": "file"
    },
```

Beispiel einer Komponente zur Reaktion auf Sicherheitslücken

Im Folgenden finden Sie ein Beispiel für eine Dockerfile-Suche nach einer Komponente zur Reaktion auf Sicherheitslücken.

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

```
    },
    "updated": "2024-03-27T14:36:39Z"
  },
```

> **Note**
>
> Wenn Sie Sbomgen ohne das `--scan-sbom` Flag aufrufen, können Sie nur Rohergebnisse von Dockerfile anzeigen.

Unterstützte Dockerfile-Prüfungen

SbomgenDockerfile-Prüfungen werden für Folgendes unterstützt:

- Das Sudo-Binärpaket
- APT-Dienstprogramme von Debian
- Hartcodierte Geheimnisse
- Root-Container
- Schwächung der Befehlsflags zur Laufzeit
- Umgebungsvariablen, die die Laufzeit schwächen

Jede dieser Dockerfile-Prüfungen hat einen entsprechenden Schweregrad, der oben in den folgenden Themen angegeben ist.

> **Note**
>
> Die in den folgenden Themen beschriebenen Empfehlungen basieren auf bewährten Verfahren der Branche.

Das Sudo-Binärpaket

> **Note**
>
> Der Schweregrad dieser Prüfung ist Info.

Wir empfehlen, das Sudo-Binärpaket nicht zu installieren oder zu verwenden, da es ein unvorhersehbares TTY- und Signalweiterleitungsverhalten aufweist. Weitere Informationen finden Sie unter Benutzer auf der Docker Docs-Website. Wenn Ihr Anwendungsfall eine ähnliche Funktionalität wie das Sudo-Binärpaket erfordert, empfehlen wir die Verwendung von Gosu.

DebianAPT-Dienstprogramme

> **Note**
>
> Der Schweregrad dieser Prüfung ist Hoch.

Im Folgenden finden Sie bewährte Methoden für die Verwendung von Debian APT-Dienstprogrammen.

Kombinieren von **`apt-get`** Befehlen in einer einzigen **`Run`** Anweisung, um Probleme beim Zwischenspeichern zu vermeiden

Wir empfehlen, `apt-get` Befehle in einer einzigen RUN-Anweisung innerhalb Ihres Docker-Containers zu kombinieren. Die `apt-get update` alleinige Verwendung führt zu Caching-Problemen und nachfolgenden `apt-get install` Anweisungen schlagen fehl. Weitere Informationen finden Sie unter apt-get auf der Docker Docs-Website.

> **Note**
>
> Das beschriebene Caching-Verhalten kann auch innerhalb Ihres Docker Containers auftreten, wenn die Docker-Container-Software veraltet ist.

Verwenden Sie das APT-Befehlszeilenprogramm auf nicht interaktive Weise

Wir empfehlen, das APT-Befehlszeilenprogramm interaktiv zu verwenden. Das APT-Befehlszeilenprogramm ist als Tool für Endbenutzer konzipiert, und sein Verhalten ändert sich zwischen den Versionen. Weitere Informationen finden Sie auf der Debian-Website unter Verwendung von Skripten und Unterschiede zu anderen APT-Tools.

Hartcodierte Geheimnisse

> **Note**
>
> Der Schweregrad dieser Prüfung ist Kritisch.

Vertrauliche Informationen in Ihrem Dockerfile werden als fest codiertes Geheimnis betrachtet. Die folgenden hartcodierten Geheimnisse können durch Sbomgen Docker-Dateiprüfungen identifiziert werden:

- AWS Zugriffsschlüssel — IDs `AKIAIOSFODNN7EXAMPLE`
- AWS geheime Schlüssel — `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`
- DockerHub persönliche Zugriffstoken — `dckr_pat_thisisa27charexample1234567`
- GitHub persönliche Zugriffstoken — `ghp_examplev61wY7Pj1YnotrealUoY123456789`
- GitLab persönliche Zugriffstoken — `glpat-12345example12345678`

Root-Container

> **Note**
>
> Der Schweregrad für diese Prüfung ist Info.

Wir empfehlen, Docker-Container ohne Root-Rechte auszuführen. Für containerisierte Workloads, die ohne Root-Rechte nicht ausgeführt werden können, empfehlen wir, Ihre Anwendungen nach einem Prinzip mit den geringsten Rechten zu erstellen. Weitere Informationen finden Sie unter Benutzer auf der Docker Docs-Website.

Umgebungsvariablen, die die Laufzeit schwächen

> **Note**
>
> Der Schweregrad dieser Prüfung ist Hoch.

Verschiedene Befehlszeilenprogramme oder Runtimes in Programmiersprachen unterstützen die Umgehung sicherer Standardeinstellungen, wodurch die Ausführung mit unsicheren Methoden ermöglicht wird.

NODE_TLS_REJECT_UNAUTHORIZED=0

Wenn Node.js Prozesse mit der Einstellung auf ausgeführt werden, ist die TLS-Zertifikatsvalidierung deaktiviert. `NODE_TLS_REJECT_UNAUTHORIZED 0` Weitere Informationen finden Sie unter NODE_TLS_REJECT_UNAUTHORIZED=0 auf der Website Node.js.

GIT_SSL_NO_VERIFY=*

Wenn Git-Befehlszeilenprozesse mit `GIT_SSL_NO_VERIFY` set ausgeführt werden, überspringt Git die Überprüfung von TLS-Zertifikaten. Weitere Informationen finden Sie unter Umgebungsvariablen auf der Git-Website.

PIP_TRUSTED_HOST=*

Wenn Python Pip-Befehlszeilenprozesse mit `PIP_TRUSTED_HOST` set ausgeführt werden, überspringt Pip die Überprüfung von TLS-Zertifikaten auf der angegebenen Domain. Weitere Informationen finden Sie unter --trusted-host auf der Pip-Website.

npm_config_strict_ssl=False

Wenn Node.js npm-Befehlszeilenprozesse mit dem `NPM_CONFIG_STRICT_SSL` Wert false ausgeführt werden, stellt das Node Package Manager (npm) -Hilfsprogramm eine Verbindung zur NPM-Registrierung her, ohne die TLS-Zertifikate zu überprüfen. Weitere Informationen finden Sie unter strict-ssl auf der Website von npm Docs.

Befehlsflags werden zur Laufzeit geschwächt

> **Note**
>
> Der Schweregrad dieser Prüfung ist Hoch.

Ähnlich wie bei Umgebungsvariablen, die die Laufzeit schwächen, unterstützen mehrere Befehlszeilenprogramme oder Laufzeitumgebungen in Programmiersprachen die Umgehung sicherer Standardwerte, was die Ausführung mit unsicheren Methoden ermöglicht.

`npm --strict-ssl=false`

Wenn die npm-Befehlszeilenprozesse von Node.js mit dem `--strict-ssl=false` Flag ausgeführt werden, stellt das Node Package Manager (npm) -Hilfsprogramm eine Verbindung zur NPM-Registrierung her, ohne die TLS-Zertifikate zu überprüfen. Weitere Informationen finden Sie unter strict-ssl auf der npm Docs-Website.

`apk --allow-untrusted`

Wenn das Alpine Package Keeper Hilfsprogramm mit dem `--allow-untrusted` Flag ausgeführt wird, `apk` werden Pakete ohne oder ohne vertrauenswürdige Signaturen installiert. Weitere Informationen finden Sie im folgenden Repository auf der Apline-Website.

`apt-get --allow-unauthenticated`

Wenn das `apt-get` Debian-Paket-Hilfsprogramm mit dem `--allow-unauthenticated` Flag ausgeführt wird, überprüft `apt-get` es nicht die Gültigkeit des Pakets. Weitere Informationen finden Sie unter apt-Get (8) auf der Debian-Webseite.

`pip --trusted-host`

Wenn das Python Pip-Hilfsprogramm mit dem `--trusted-host` Flag ausgeführt wird, umgeht der angegebene Hostname die Validierung des TLS-Zertifikats. Weitere Informationen finden Sie unter --trusted-host auf der Pip-Website.

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

Wenn der RPM-basierte Paketmanager mit den `--nofiledigest` Flags,, und ausgeführt `rpm` wird `--nodigest` `--nosignature``--noverify`, validiert der RPM-Paketmanager bei der Installation eines Pakets keine Paket-Header, Signaturen oder Dateien. Weitere Informationen finden Sie auf der folgenden RPM-Handbuchseite auf der RPM-Website.

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

Wenn der RPM-basierte Paketmanager mit dem `--setopt=sslverify` Flag auf False ausgeführt `yum-config-manager` wird, validiert der YUM-Paketmanager keine TLS-Zertifikate. Weitere Informationen finden Sie auf der folgenden YUM-Handbuchseite auf der Man7-Website.

`yum --nogpgcheck`

Wenn der RPM-basierte Paketmanager mit dem `--nogpgcheck` Flag ausgeführt `yum` wird, überspringt der YUM-Paketmanager die Überprüfung der GPG-Signaturen auf Paketen. Weitere Informationen finden Sie unter yum (8) auf der Man7-Website.

`curl --insecure, curl -k`

Wenn sie mit dem `-k` Flag `--insecure` oder ausgeführt `curl` wird, ist die TLS-Zertifikatsvalidierung deaktiviert. Standardmäßig wird jede sichere Verbindung, die `curl` hergestellt wird, als sicher verifiziert, bevor die Übertragung stattfindet. Bei dieser Option können `curl` Sie den Bestätigungsschritt überspringen und ohne Überprüfung fortfahren. Weitere Informationen finden Sie auf der folgenden Curl-Handbuchseite auf der Curl-Website.

`wget --no-check-certificate`

Wenn mit dem `--no-check-certificate` Flag ausgeführt `wget` wird, ist die TLS-Zertifikatsvalidierung deaktiviert. Weitere Informationen finden Sie auf der folgenden Wget-Handbuchseite auf der GNU-Website.

Beim Entfernen werden Betriebssystem-Paketdatenbanken in Containern überprüft

> **Note**
>
> Der Schweregrad dieser Prüfung ist Info.

Das Entfernen einer Betriebssystem-Paketdatenbank reduziert die Möglichkeit, das gesamte Inventar der Software eines Container-Images zu scannen. Diese Datenbanken sollten während der Container-Build-Schritte intakt bleiben.

Entfernungsprüfungen für eine Betriebssystem-Paketdatenbank werden für die folgenden Paketmanager unterstützt:

Alpine Package Keeper (APK)

Container-Images, die den APK-Paketmanager für installierte Software verwenden, müssen sicherstellen, dass APK-Systemdateien während eines Builds nicht entfernt werden. Weitere Informationen finden Sie in der Dokumentation zu den APK-Manpages-Systemdateien auf der Arch Linux Website.

Debian-Paketmanager (DPKG)

Container, die den DPKG-Paketmanager verwenden, wie z. B. Debian-, Ubuntu- oder Distroless-basierte Images, müssen sicherstellen, dass die DPKG-Datenbank während eines Container-Builds nicht entfernt wird. Weitere Informationen finden Sie in der Dokumentation zu den DPKG-Manpages-Systemdateien auf der Website. Ubuntu

RPM-Paketmanager (RPM)

Container, die den RPM Package Manager (yum/dnf) verwenden, wie Amazon Linux oder Red Hat Enterprise Linux, müssen sicherstellen, dass die RPM-Datenbank während eines Container-Builds nicht entfernt wird. Weitere Informationen finden Sie in der Dokumentation zu den RPM-Manpages-Systemdateien auf der RPM-Website.

Erstellen einer benutzerdefinierten CI/CD Pipeline-Integration mit Amazon Inspector Scan

Wir empfehlen Ihnen, die Amazon CI/CD Inspector-Plugins zu verwenden, wenn die Amazon CI/CD Inspector-Plugins für Ihre CI/CD Lösung verfügbar sind. Wenn die Amazon CI/CD Inspector-Plugins für Ihre CI/CD Lösung nicht verfügbar sind, können Sie eine Kombination aus dem Amazon Inspector SBOM Generator und der Amazon Inspector Scan API verwenden, um eine benutzerdefinierte CI/CD Integration zu erstellen. In den folgenden Schritten wird beschrieben, wie Sie eine benutzerdefinierte CI/CD Pipeline-Integration mit Amazon Inspector Scan erstellen.

> **Tip**
>
> Sie können den Amazon Inspector SBOM Generator (Sbomgen) verwenden, um Schritt 3 und Schritt 4 zu überspringen, wenn Sie Ihre SBOM mit einem einzigen Befehl generieren und scannen möchten.

Schritt 1. Konfiguration AWS-Konto

Konfigurieren Sie einen AWS-Konto , der Zugriff auf die Amazon Inspector Scan API bietet. Weitere Informationen finden Sie unter Einrichtung eines AWS Kontos für die Nutzung der Amazon Inspector CI/CD Inspector-Integration.

Schritt 2. SbomgenBinärdatei installieren

Installieren und konfigurieren Sie die Sbomgen Binärdatei. Weitere Informationen finden Sie unter Installieren der Sbomgen.

Schritt 3. Verwenden von Sbomgen

Verwenden Sie dieSbomgen, um eine SBOM-Datei für ein Container-Image zu erstellen, das Sie scannen möchten.

Sie können das folgende Beispiel verwenden. *image:id*Ersetzen Sie es durch den Namen des Bilds, das Sie scannen möchten. *sbom_path.json*Ersetzen Sie durch den Speicherort, an dem Sie die SBOM-Ausgabe speichern möchten.

Beispiel

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

Schritt 4. Aufrufen der Amazon Inspector Scan API

Rufen Sie die `inspector-scan` API auf, um die generierte SBOM zu scannen und einen Schwachstellenbericht zu erstellen.

Sie können das folgende Beispiel verwenden. *sbom_path.json*Ersetzen Sie durch den Speicherort einer gültigen CyclonedX-kompatiblen SBOM-Datei. Ersetzen Sie es *ENDPOINT* durch den API-Endpunkt für den Ort, an AWS-Region dem Sie derzeit authentifiziert sind. Ersetzen Sie es *REGION* durch die entsprechende Region.

Beispiel

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

Eine vollständige Liste der AWS-Regionen Endpunkte finden Sie unter Regionen und Endpunkte.

(Optional) Schritt 5. Generieren und scannen Sie SBOM mit einem einzigen Befehl

> **Note**
>
> Führen Sie diesen Schritt nur aus, wenn Sie Schritt 3 und Schritt 4 übersprungen haben.

Generieren und scannen Sie Ihre SBOM mit einem einzigen Befehl mithilfe der Markierung. `--scan-bom`

Sie können das folgende Beispiel verwenden. *image:id*Ersetzen Sie es durch den Namen des Bilds, das Sie scannen möchten. *profile*Ersetzen Sie es durch das entsprechende Profil. *REGION*Durch die entsprechende Region ersetzen. */tmp/scan.json*Ersetzen Sie durch den Speicherort der Datei scan.json im Verzeichnis tmp.

Beispiel

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

Eine vollständige Liste der Endpunkte finden Sie unter Regionen AWS-Regionen und Endpunkte.

API-Ausgabeformate

Die Amazon Inspector Scan API kann einen Schwachstellenbericht im CycloneDX 1.5-Format oder Amazon Inspector Finding JSON ausgeben. Die Standardeinstellung kann mithilfe des `--output-format` Flags geändert werden.

Beispiel für eine Ausgabe im CycloneDX 1.5-Format


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

Beispiel für eine Ausgabe im Inspector-Format

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

Verwenden des Amazon Jenkins Inspector-Plug-ins

Das Jenkins Plugin nutzt die Amazon Inspector SBOM Generator-Binärdatei und die Amazon Inspector Scan API, um am Ende Ihres Builds detaillierte Berichte zu erstellen, sodass Sie Risiken vor der Bereitstellung untersuchen und beheben können. Mit dem Amazon Jenkins Inspector-Plugin können Sie Amazon Inspector Inspector-Schwachstellenscans zu Ihrer Jenkins Pipeline hinzufügen. Amazon Inspector Vulnerability Scans können so konfiguriert werden, dass Pipeline-Ausführungen je nach Anzahl und Schweregrad der erkannten Sicherheitslücken bestanden oder fehlschlagen. Die neueste Version des Jenkins Plug-ins finden Sie im Jenkins Marketplace unter https://plugins.jenkins.io/amazon-inspector-image-scanner/. In den folgenden Schritten wird beschrieben, wie Sie das Amazon Jenkins Inspector-Plugin einrichten.

> **Important**
>
> Bevor Sie die folgenden Schritte ausführen, müssen Sie Jenkins auf Version 2.387.3 oder höher aktualisieren, damit das Plugin ausgeführt werden kann.

Schritt 1. Richten Sie ein AWS-Konto

Konfigurieren Sie eine AWS-Konto mit einer IAM-Rolle, die den Zugriff auf die Amazon Inspector Scan API ermöglicht. Detaillierte Anweisungen finden Sie unter Einrichtung eines AWS Kontos für die Nutzung der Amazon Inspector CI/CD Inspector-Integration.

Schritt 2. Installieren Sie das Amazon Inspector Jenkins-Plugin

Das folgende Verfahren beschreibt, wie Sie das Amazon Inspector Jenkins-Plugin vom Jenkins Dashboard aus installieren.

1. Wählen Sie im Jenkins-Dashboard Manage Jenkins und anschließend Manage Plugins aus.
2. Wählen Sie „Verfügbar“.
3. Suchen Sie auf der Registerkarte Verfügbar nach Amazon Inspector Scans und installieren Sie dann das Plugin.

(Optional) Schritt 3. Fügen Sie Docker-Anmeldeinformationen hinzu Jenkins

> **Note**
>
> Fügen Sie nur Docker-Anmeldeinformationen hinzu, wenn sich das Docker-Image in einem privaten Repository befindet. Andernfalls überspringen Sie diesen Schritt.

Das folgende Verfahren beschreibt, wie Sie Docker-Anmeldeinformationen vom Jenkins Dashboard Jenkins aus hinzufügen.

1. Wählen Sie im Jenkins-Dashboard Manage Jenkins, Credentials und dann System aus.
2. Wählen Sie Globale Anmeldeinformationen und dann Anmeldeinformationen hinzufügen aus.
3. Wählen Sie unter Kind die Option Nutzername mit Passwort aus.
4. Wählen Sie unter Bereich die Option Global (Jenkins, Knoten, Elemente, alle untergeordneten Elemente usw.) aus.
5. Geben Sie Ihre Daten ein und wählen Sie dann OK.

(Optional) Schritt 4. Fügen Sie AWS Anmeldeinformationen hinzu

> **Note**
>
> Fügen Sie nur AWS Anmeldeinformationen hinzu, wenn Sie sich anhand eines IAM-Benutzers authentifizieren möchten. Andernfalls überspringen Sie diesen Schritt.

Im folgenden Verfahren wird beschrieben, wie Sie AWS Anmeldeinformationen über das Jenkins Dashboard hinzufügen.

1. Wählen Sie im Jenkins-Dashboard Manage Jenkins, Credentials und dann System aus.
2. Wählen Sie Globale Anmeldeinformationen und dann Anmeldeinformationen hinzufügen aus.
3. Wählen Sie für Kind die Option AWS-Anmeldeinformationen aus.
4. Geben Sie Ihre Daten ein, einschließlich Ihrer Zugangsschlüssel-ID und Ihres geheimen Zugangsschlüssels, und wählen Sie dann OK.

Schritt 5. Fügen Sie CSS-Unterstützung in einem Jenkins Skript hinzu

Das folgende Verfahren beschreibt, wie CSS-Unterstützung in einem Jenkins Skript hinzugefügt wird.

1. Starten Sie Jenkins neu.
2. Wählen Sie im Dashboard Manage Jenkins, Nodes, Built-In Node und dann Script Console aus.
3. Fügen Sie im Textfeld die Zeile hinzu und wählen Sie **`System.setProperty("hudson.model.DirectoryBrowserSupport.CSP", "")`** dann Ausführen aus.

Schritt 6: Fügen Sie Amazon Inspector Scan zu Ihrem Build hinzu

Sie können Amazon Inspector Scan zu Ihrem Build hinzufügen, indem Sie Ihrem Projekt einen Build-Schritt hinzufügen oder die Jenkins deklarative Pipeline verwenden.

Amazon Inspector Scan zu Ihrem Build, indem Sie Ihrem Projekt einen Build-Schritt hinzufügen

1. Scrollen Sie auf der Konfigurationsseite nach unten zu Build Steps und wählen Sie Build-Schritt hinzufügen aus. Wählen Sie dann Amazon Inspector Scan aus.
2. Wählen Sie zwischen zwei Inspector-Sbomgen-Installationsmethoden: Automatisch oder Manuell. Die automatische Option ermöglicht es dem Plugin, die neueste Version herunterzuladen. Außerdem wird sichergestellt, dass Sie immer über die neuesten Funktionen, Sicherheitsupdates und Bugfixes verfügen.

a. (Option 1) Wählen Sie Automatisch, um die neueste Version von inspector-sbomgen herunterzuladen. Diese Option erkennt automatisch das Betriebssystem und die CPU-Architektur, die derzeit verwendet werden.

b. (Option 2) Wählen Sie Manuell, wenn Sie die Amazon Inspector SBOM Generator-Binärdatei für das Scannen einrichten möchten. Wenn Sie diese Methode wählen, stellen Sie sicher, dass Sie den vollständigen Pfad zu einer zuvor heruntergeladenen Version von inspector-sbomgen angeben.

Weitere Informationen finden Sie unter Installation von Amazon Inspector SBOM Generator (Sbomgen) in Amazon Inspector SBOM Generator.

3. Gehen Sie wie folgt vor, um die Konfiguration des Amazon Inspector Scan-Build-Schritts abzuschließen:

a. Geben Sie Ihre Bild-ID ein. Das Bild kann lokal, remote oder archiviert sein. Bildnamen sollten der Docker Benennungskonvention entsprechen. Wenn Sie ein exportiertes Bild analysieren, geben Sie den Pfad zur erwarteten TAR-Datei an. Sehen Sie sich das folgende Beispiel für Image-ID-Pfade an:

   i. Für lokale oder Remote-Container: `NAME[:TAG|@DIGEST]`

   ii. Für eine TAR-Datei: `/path/to/image.tar`

b. Wählen Sie einen aus AWS-Region, über den die Scananforderung gesendet werden soll.

c. (Optional) Geben Sie unter Berichtsartefaktname einen benutzerdefinierten Namen für die während des Erstellungsprozesses generierten Artefakte ein. Auf diese Weise können sie eindeutig identifiziert und verwaltet werden.

d. (Optional) Geben Sie unter Dateien überspringen ein oder mehrere Verzeichnisse an, die Sie vom Scannen ausschließen möchten. Ziehen Sie diese Option für Verzeichnisse in Betracht, die aufgrund ihrer Größe nicht gescannt werden müssen.

e. (Optional) Wählen Sie für Docker-Anmeldeinformationen Ihren Docker Benutzernamen aus. Tun Sie dies nur, wenn sich Ihr Container-Image in einem privaten Repository befindet.

f. (Optional) Sie können die folgenden unterstützten AWS Authentifizierungsmethoden bereitstellen:

   i. (Optional) Geben Sie für die IAM-Rolle einen Rollen-ARN an (arn:aws:iam: :role/). *AccountNumber RoleName*

ii. (Optional) Geben Sie für AWS-Anmeldeinformationen AWS Anmeldeinformationen für die Authentifizierung auf der Grundlage eines IAM-Benutzers an.

iii. (Optional) Geben Sie als AWS Profilname den Namen eines Profils an, für das Sie sich mithilfe eines Profilnamens authentifizieren möchten.

g. (Optional) Wählen Sie Schwellenwerte für Sicherheitslücken aktivieren aus. Mit dieser Option können Sie feststellen, ob Ihr Build fehlschlägt, wenn eine gescannte Sicherheitslücke einen Wert überschreitet. Wenn alle Werte gleich sind`0`, ist der Build erfolgreich, unabhängig davon, wie viele Sicherheitslücken gescannt werden. Für den EPSS-Score kann der Wert zwischen 0 und 1 liegen. Wenn eine gescannte Sicherheitslücke einen Wert überschreitet, schlägt der Build fehl, und alle, CVEs deren EPSS-Score über dem Wert liegt, werden in der Konsole angezeigt.

4. Wählen Sie Speichern.

Fügen Sie Amazon Inspector Scan mithilfe der Jenkins deklarativen Pipeline zu Ihrem Build hinzu

Sie können Amazon Inspector Scan mithilfe der deklarativen Jenkins-Pipeline automatisch oder manuell zu Ihrem Build hinzufügen.

Um die deklarative Pipeline automatisch herunterzuladen SBOMGen

- Verwenden Sie die folgende Beispielsyntax, um Amazon Inspector Scan zu einem Build hinzuzufügen. Basierend auf Ihrer bevorzugten Betriebssystemarchitektur des Amazon Inspector SBOM Generator-Downloads, *`SBOMGEN_SOURCE`* ersetzen Sie ihn durch LinuxAMD64 oder LinuxARM64. *`IMAGE_PATH`*Ersetzen Sie durch den Pfad zu Ihrem Image (z. B.*`alpine:latest`*), durch den *`IAM_ROLE`* ARN der IAM-Rolle, die Sie in Schritt 1 konfiguriert haben, und *`ID`* durch Ihre Docker Anmeldeinformations-ID, wenn Sie ein privates Repository verwenden. Sie können optional Schwellenwerte für Sicherheitslücken aktivieren und Werte für jeden Schweregrad angeben.

```
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,
               ])
            }
          }
        }
    }
}
```

Um die SBOMGen deklarative Pipeline manuell herunterzuladen

- Verwenden Sie die folgende Beispielsyntax, um Amazon Inspector Scan zu einem Build hinzuzufügen. *SBOMGEN_PATH*Ersetzen Sie durch den Pfad zum Amazon Inspector SBOM Generator, den Sie in Schritt 3 installiert haben, *IMAGE_PATH* durch den Pfad zu Ihrem Image (z. B.*alpine:latest*), durch den *IAM_ROLE* ARN der IAM-Rolle, die Sie in Schritt 1 konfiguriert haben, und *ID* durch Ihre Docker Anmeldeinformations-ID, wenn Sie ein privates Repository verwenden. Sie können optional Schwellenwerte für Sicherheitslücken aktivieren und Werte für jeden Schweregrad angeben.

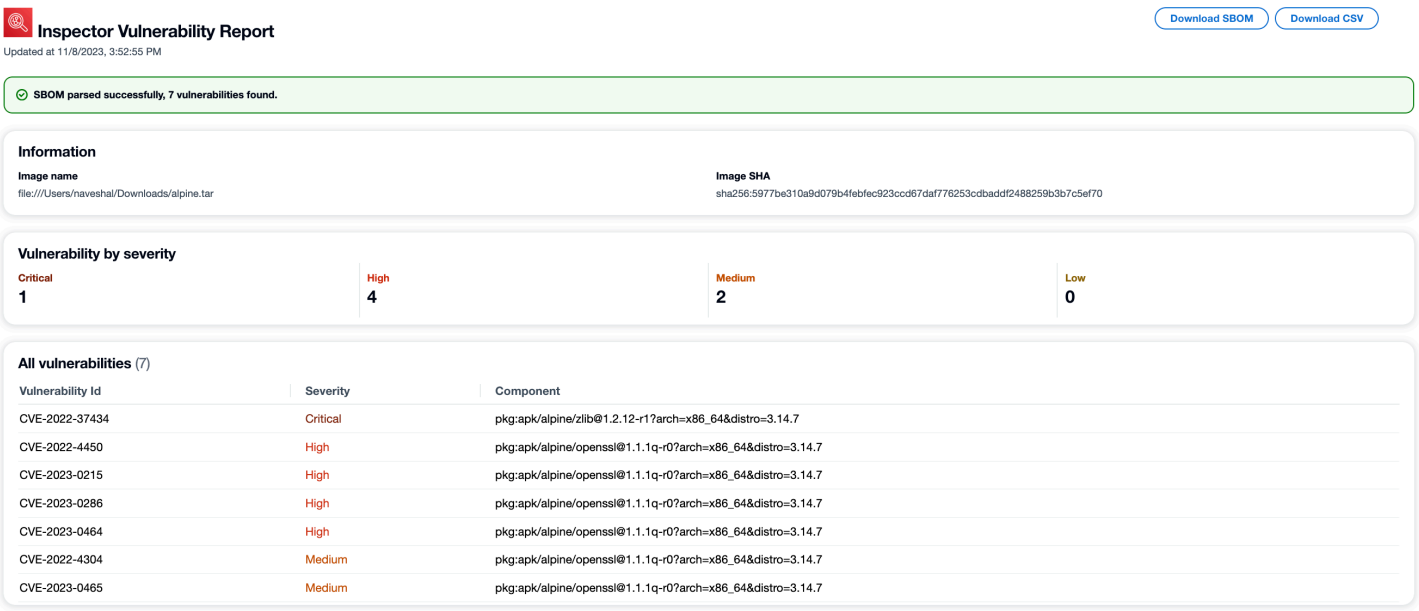
> **Note**
>
> Platzieren Sie es Sbomgen im Jenkins-Verzeichnis und geben Sie den Pfad zum Jenkins-Verzeichnis im Plugin an (z. B.). */opt/folder/arm64/inspector-sbomgen*

```
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,
               ])
            }
          }
        }
    }
 }
```

Schritt 7. Sehen Sie sich Ihren Amazon Inspector Inspector-Schwachstellenbericht an

1. Vervollständigen Sie einen neuen Build Ihres Projekts.
2. Wählen Sie nach Abschluss des Builds ein Ausgabeformat aus den Ergebnissen aus. Wenn Sie HTML auswählen, haben Sie die Möglichkeit, eine JSON-, SBOM- oder CSV-Version des Berichts herunterzuladen. Im Folgenden wird ein Beispiel für einen HTML-Bericht gezeigt:



Fehlerbehebung

Im Folgenden sind häufig auftretende Fehler aufgeführt, die bei der Verwendung des Amazon Inspector Scan-Plug-ins auftreten könnenJenkins.

Anmeldeinformationen konnten nicht geladen werden oder STS-Ausnahmefehler

Fehler:

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

Auflösung

Holen Sie sich `aws_access_key_id` und `aws_secret_access_key` für Ihr AWS Konto. Aufstellen `aws_access_key_id` und `aws_secret_access_key` rein`~/.aws/credentials`.

Das Bild konnte nicht aus Tarball-, lokalen oder Remote-Quellen geladen werden

Fehler:

```
2024/10/16 02:25:17 [ImageDownloadFailed]: failed to load image from tarball, local, or remote sources.
```

> **Note**
>
> Dieser Fehler kann auftreten, wenn das Jenkins-Plugin das Container-Image nicht lesen kann, das Container-Image nicht in der Docker Engine gefunden wird und das Container-Image nicht in der Remote-Container-Registry gefunden wird.

Auflösung

Überprüfen Sie Folgendes:

- Der Benutzer des Jenkins-Plugins hat Leseberechtigungen für das Bild, das Sie scannen möchten.
- Das Bild, das Sie scannen möchten, ist in Docker der Engine vorhanden.
- Ihre Remote-Bild-URL ist korrekt.
- Sie sind bei der Remote-Registrierung authentifiziert (falls zutreffend).

Inspector-SBOMGen-Pfadfehler

Fehler:

```
Exception:com.amazon.inspector.jenkins.amazoninspectorbuildstep.exception.Sbomge
There was an issue running inspector-sbomgen, is /opt/inspector/inspector-
sbomgen the correct path?
```

Auflösung

Gehen Sie wie folgt vor, um das Problem zu beheben.

1. Platzieren Sie die richtige Betriebssystemarchitektur Inspector-SBOMGen im Jenkins Verzeichnis Weitere Informationen finden Sie unter Amazon Inspector SBOM Generator.
2. Erteilen Sie mit dem folgenden Befehl ausführbare Rechte für die Binärdatei:. `chmod +x inspector-sbomgen`
3. Geben Sie im Plugin den richtigen Jenkins Computerpfad an, z. `/opt/folder/arm64/inspector-sbomgen` B.
4. Speichern Sie die Konfiguration und führen Sie den Jenkins Job aus.

Verwenden des Amazon TeamCity Inspector-Plug-ins

Das Amazon TeamCity Inspector-Plugin nutzt die Amazon Inspector SBOM Generator-Binärdatei und die Amazon Inspector Scan API, um am Ende Ihres Builds detaillierte Berichte zu erstellen, sodass Sie Risiken vor der Bereitstellung untersuchen und beheben können. Mit dem Amazon TeamCity Inspector-Plugin können Sie Amazon Inspector Inspector-Schwachstellenscans zu Ihrer TeamCity Pipeline hinzufügen. Amazon Inspector Vulnerability Scans können so konfiguriert werden, dass Pipeline-Ausführungen je nach Anzahl und Schweregrad der erkannten Sicherheitslücken bestanden oder fehlschlagen. Sie können die neueste Version des Amazon TeamCity Inspector-Plug-ins im TeamCity Marketplace unter https://plugins.jetbrains.com/plugin/23236- amazon-inspector-scanner einsehen. Informationen zur Integration von Amazon Inspector Scan in Ihre CI/CD Pipeline finden Sie unter Amazon Inspector-Scans in Ihre CI/CD Pipeline integrieren. Eine Liste der Betriebssysteme und Programmiersprachen, die Amazon Inspector unterstützt, finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen. In den folgenden Schritten wird beschrieben, wie Sie das Amazon TeamCity Inspector-Plugin einrichten.

1. Richten Sie ein AWS-Konto.

   - Konfigurieren Sie eine AWS-Konto mit einer IAM-Rolle, die den Zugriff auf die Amazon Inspector Scan API ermöglicht. Detaillierte Anweisungen finden Sie unter Einrichtung eines AWS Kontos für die Nutzung der Amazon Inspector CI/CD Inspector-Integration.

2. Installieren Sie das Amazon TeamCity Inspector-Plugin.

   a. Gehen Sie in Ihrem Dashboard zu Administration > Plugins.

   b. Suchen Sie nach Amazon Inspector Scans.

   c. Installieren Sie das -Plug-in.

3. Installieren Sie den Amazon Inspector SBOM Generator.

   - Installieren Sie die Amazon Inspector SBOM Generator-Binärdatei in Ihrem Teamcity-Serververzeichnis. Detaillierte Anweisungen finden Sie unter Installation von Sbomgen.

4. Fügen Sie Ihrem Projekt einen Amazon Inspector Scan-Build-Schritt hinzu.

   a. Scrollen Sie auf der Konfigurationsseite nach unten zu Build Steps, wählen Sie Build-Schritt hinzufügen und dann Amazon Inspector Scan aus.

   b. Konfigurieren Sie den Erstellungsschritt Amazon Inspector Scan, indem Sie die folgenden Details eingeben:

- Fügen Sie einen Schrittnamen hinzu.
- Wählen Sie zwischen zwei Installationsmethoden für Amazon Inspector SBOM Generator: Automatisch oder Manuell.
  - Lädt automatisch die neueste Version von Amazon Inspector SBOM Generator herunter, die auf Ihrer System- und CPU-Architektur basiert.
  - Für das Handbuch müssen Sie einen vollständigen Pfad zu einer zuvor heruntergeladenen Version von Amazon Inspector SBOM Generator angeben.

  Weitere Informationen finden Sie unter Installation von Amazon Inspector SBOM Generator (SBOM Generator) in Amazon Inspector SBOM Generator.
- Geben Sie Ihre Bild-ID ein. Ihr Bild kann lokal, remote oder archiviert sein. Bildnamen sollten der Docker Benennungskonvention entsprechen. Wenn Sie ein exportiertes Bild analysieren, geben Sie den Pfad zur erwarteten TAR-Datei an. Sehen Sie sich das folgende Beispiel für Image-ID-Pfade an:
  - Für lokale oder Remote-Container: `NAME[:TAG|@DIGEST]`
  - Für eine TAR-Datei: `/path/to/image.tar`
- Geben Sie für IAM-Rolle den ARN für die Rolle ein, die Sie in Schritt 1 konfiguriert haben.
- Wählen Sie eine aus AWS-Region, über die die Scananforderung gesendet werden soll.
- (Optional) Geben Sie für die Docker-Authentifizierung Ihren Docker-Benutzernamen und Ihr Docker-Passwort ein. Tun Sie dies nur, wenn sich Ihr Container-Image in einem privaten Repository befindet.
- (Optional) Geben Sie für die AWS Authentifizierung Ihre AWS Zugriffsschlüssel-ID und Ihren AWS geheimen Schlüssel ein. Tun Sie dies nur, wenn Sie sich anhand von AWS Anmeldeinformationen authentifizieren möchten.
- (Optional) Geben Sie die Schwellenwerte für Sicherheitslücken pro Schweregrad an. Wenn die von Ihnen angegebene Zahl während eines Scans überschritten wird, schlägt die Image-Erstellung fehl. Wenn die Werte alle sind, ist `0` der Build unabhängig von der Anzahl der gefundenen Sicherheitslücken erfolgreich.

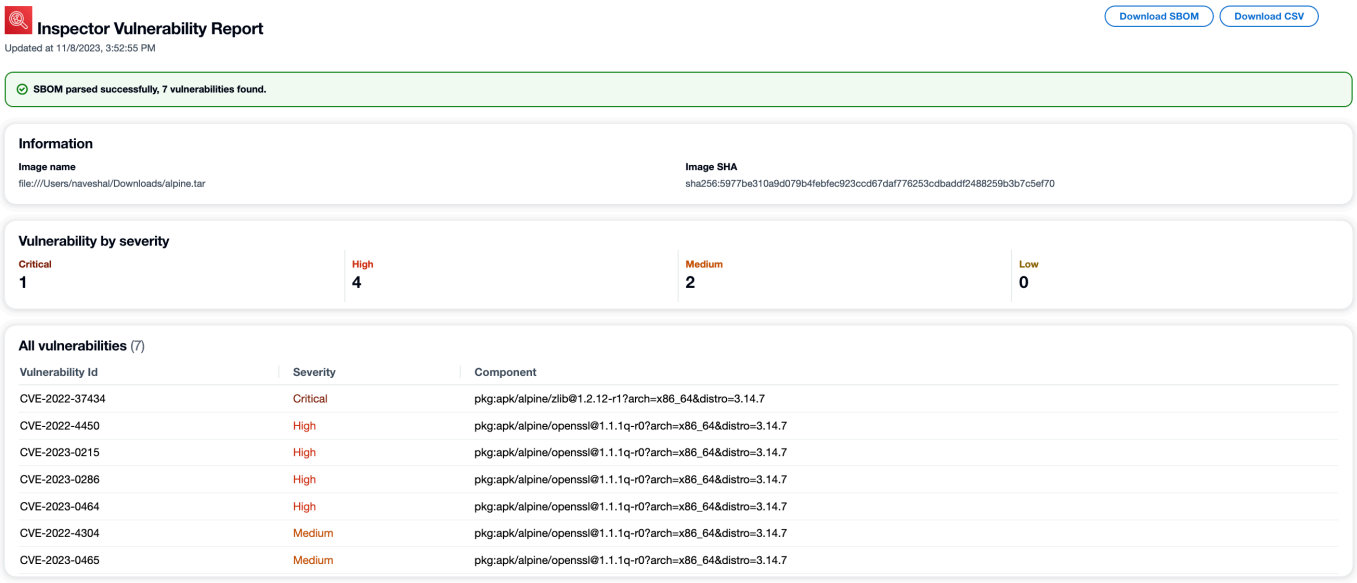
c. Wählen Sie Save (Speichern).

5. Sehen Sie sich Ihren Amazon Inspector-Sicherheitslückenbericht an.

   a. Vervollständigen Sie einen neuen Build Ihres Projekts.

   b. Wenn der Build abgeschlossen ist, wählen Sie ein Ausgabeformat aus den Ergebnissen aus. Wenn Sie HTML auswählen, haben Sie die Möglichkeit, eine JSON-, SBOM- oder CSV-

Version des Berichts herunterzuladen. Im Folgenden finden Sie ein Beispiel für einen HTML-Bericht:



Amazon Inspector mit GitHub Aktionen verwenden

Sie können Amazon Inspector mit verwenden GitHub actions, um Amazon Inspector Inspector-Schwachstellenscans zu Ihren GitHub Workflows hinzuzufügen. Dabei werden der Amazon Inspector SBOM Generator und die Amazon Inspector Scan API genutzt, um am Ende Ihres Builds detaillierte Berichte zu erstellen, sodass Sie Risiken vor der Bereitstellung untersuchen und beheben können. Amazon Inspector Vulnerability Scans können je nach Anzahl und Schweregrad der erkannten Sicherheitslücken so konfiguriert werden, dass Workflows erfolgreich oder nicht bestanden werden. Sie können die neueste Version der Amazon Inspector Inspector-Aktion auf der GitHubWebsite einsehen. Informationen zur Integration von Amazon Inspector Scan in Ihre CI/CD Pipeline finden Sie unter Amazon Inspector-Scans in Ihre CI/CD Pipeline integrieren. Eine Liste der Betriebssysteme und Programmiersprachen, die Amazon Inspector unterstützt, finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen.

Amazon Inspector mit GitLab Komponenten verwenden

Sie können Amazon Inspector mit GitLab CI/CD-Komponenten verwenden, um Amazon Inspector Inspector-Schwachstellenscans zu Ihren GitLab Projekten hinzuzufügen. Dabei werden der Amazon Inspector SBOM Generator und die Amazon Inspector Scan API genutzt, um am Ende Ihres Builds

detaillierte Berichte zu erstellen, sodass Sie Risiken vor der Bereitstellung untersuchen und beheben können. Amazon Inspector Vulnerability Scans können je nach Anzahl und Schweregrad der erkannten Sicherheitslücken so konfiguriert werden, dass Workflows erfolgreich oder nicht bestanden werden. Sie können die neueste Version der Amazon Inspector Inspector-Komponente auf der GitLabWebsite einsehen. Informationen zur Integration von Amazon Inspector Scan in Ihre CI/CD Pipeline finden Sie unter Amazon Inspector-Scans in Ihre CI/CD Pipeline integrieren. Eine Liste der Betriebssysteme und Programmiersprachen, die Amazon Inspector unterstützt, finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen.

CodeCatalystAktionen mit Amazon Inspector verwenden

Sie können Amazon Inspector mit Amazon verwenden CodeCatalyst, um Amazon Inspector Inspector-Schwachstellenscans zu Ihren CodeCatalyst Workflows hinzuzufügen. Dabei werden der Amazon Inspector SBOM Generator und die Amazon Inspector Scan API genutzt, um am Ende Ihres Builds detaillierte Berichte zu erstellen, sodass Sie Risiken vor der Bereitstellung untersuchen und beheben können. Amazon Inspector Vulnerability Scans können je nach Anzahl und Schweregrad der erkannten Sicherheitslücken so konfiguriert werden, dass Workflows erfolgreich oder nicht bestanden werden. Informationen zur Integration von Amazon Inspector Scan in Ihre CI/CD Pipeline finden Sie unter Amazon Inspector-Scans in Ihre CI/CD Pipeline integrieren. Eine Liste der Betriebssysteme und Programmiersprachen, die Amazon Inspector unterstützt, finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen.

Verwenden von Amazon Inspector Scan-Aktionen mit CodePipeline

Sie können Amazon Inspector mit verwenden, AWS CodePipeline indem Sie Ihren Workflows Schwachstellenscans hinzufügen. Diese Integration nutzt den Amazon Inspector SBOM Generator und die Amazon Inspector Scan API, um am Ende Ihres Builds detaillierte Berichte zu erstellen. Die Integration hilft Ihnen, Risiken vor der Bereitstellung zu untersuchen und zu beheben. Bei der `InspectorScan` Aktion handelt es sich um eine verwaltete Rechenaktion CodePipeline , mit der die Erkennung und Behebung von Sicherheitslücken in Ihrem Open-Source-Code automatisiert wird. Du kannst diese Aktion mit Anwendungsquellcode in deinem Drittanbieter-Repository wie Bitbucket Cloud GitHub oder mit Bildern für Containeranwendungen verwenden. Weitere Informationen findest du in der Referenz zum InspectorScan Aufrufen von Aktionen im AWS CodePipeline Benutzerhandbuch.

Bewertung der Abdeckung Ihrer AWS Umgebung durch Amazon Inspector

Sie können den Umfang Ihrer AWS Umgebung durch Amazon Inspector auf dem Bildschirm Kontoverwaltung in der Amazon Inspector-Konsole beurteilen. Dort werden Details und Statistiken zum Status der Amazon Inspector-Scans für Ihre Konten und Ressourcen angezeigt.

> **Note**
>
> Wenn Sie der delegierte Administrator einer Organisation sind, können Sie Details und Statistiken für alle Konten in der Organisation einsehen.

Das folgende Verfahren beschreibt, wie Sie die Abdeckung Ihrer Amazon Inspector Inspector-Umgebung beurteilen können.

Um die Abdeckung Ihrer AWS Umgebung durch Amazon Inspector zu beurteilen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich Kontoverwaltung aus.
3. Um den Versicherungsschutz zu überprüfen, wählen Sie eine der folgenden Registerkarten:

   - Wählen Sie Konten, um den Versicherungsschutz auf Kontoebene zu überprüfen.
   - Wählen Sie Instances aus, um den Versicherungsschutz für Amazon Elastic Compute Cloud (Amazon EC2) -Instances zu überprüfen.
   - Wählen Sie Container-Repositorys aus, um den Umfang der Amazon Elastic Container Registry (Amazon ECR) -Repositorys zu überprüfen.
   - Wählen Sie Container-Images aus, um die Abdeckung von Amazon ECR-Container-Images zu überprüfen.
   - Wählen Sie Lambda-Funktionen, um den Umfang der Lambda-Funktionen zu überprüfen.

In den folgenden Themen werden die Informationen beschrieben, die jede dieser Registerkarten enthält.

Themen

Bewertung der Deckung auf Kontoebene

Wenn Ihr Konto nicht Teil einer Organisation ist oder nicht das delegierte Amazon Inspector-Administratorkonto für eine Organisation ist, finden Sie auf der Registerkarte Konten Informationen über Ihr Konto und den Status des Ressourcenscans für Ihr Konto. Auf dieser Registerkarte können Sie das Scannen für alle oder nur bestimmte Arten von Ressourcen für Ihr Konto aktivieren oder deaktivieren. Weitere Informationen finden Sie unter Automatisierte Scantypen in Amazon Inspector.

Wenn Ihr Konto das delegierte Amazon Inspector-Administratorkonto für eine Organisation ist, bietet die Registerkarte Konten automatische Aktivierungseinstellungen für Konten in Ihrer Organisation und listet alle Konten in Ihrer Organisation auf. Für jedes Konto gibt die Liste an, ob Amazon Inspector für das Konto aktiviert ist, und wenn ja, welche Arten von Ressourcenscans für das Konto aktiviert sind. Als delegierter Administrator können Sie auf dieser Registerkarte die Einstellungen für die automatische Aktivierung für Ihr Unternehmen ändern. Sie können auch bestimmte Arten des Ressourcenscans für einzelne Mitgliedskonten aktivieren oder deaktivieren. Weitere Informationen finden Sie unter Die Aktivierung von Amazon Inspector scannt nach Mitgliedskonten.

Bewertung der Abdeckung von EC2 Amazon-Instances

Auf der Registerkarte Instances werden EC2 Amazon-Instances in Ihrer AWS Umgebung angezeigt. Die Listen sind auf den folgenden Registerkarten in Gruppen unterteilt:

- Alle — Zeigt alle Instanzen in Ihrer Umgebung an. In der Spalte Status wird der aktuelle Scanstatus für eine Instance angezeigt.
- Scannen — Zeigt alle Instances an, die Amazon Inspector in Ihrer Umgebung aktiv überwacht und scannt.
- Nicht scannen — Zeigt alle Instances an, die Amazon Inspector in Ihrer Umgebung nicht überwacht und scannt. Die Spalte Grund gibt an, warum Amazon Inspector eine Instance nicht überwacht und scannt.

Eine EC2 Instance kann aus verschiedenen Gründen auf der Registerkarte Nicht gescannt angezeigt werden. Amazon Inspector verwendet AWS Systems Manager (SSM) und den SSM-Agenten, um Ihre EC2 Instances automatisch zu überwachen und auf Sicherheitslücken zu scannen. Wenn auf einer Instance der SSM-Agent nicht ausgeführt wird, sie keine AWS Identity and Access Management (IAM-) Rolle hat, die Systems Manager unterstützt, oder auf der kein unterstütztes Betriebssystem oder keine unterstützte Architektur ausgeführt wird, kann Amazon Inspector die Instance nicht überwachen und scannen. Weitere Informationen finden Sie unter Scannen Amazon EC2 Amazon-Instanzen.

Auf jeder Registerkarte gibt die Spalte Konto an, wer Eigentümer einer AWS-Konto Instance ist.

EC2 Instanz-Tags — In dieser Spalte werden die mit der Instance verknüpften Tags angezeigt. Anhand dieser Spalte können Sie feststellen, ob Ihre Instance anhand von Stichwörtern von Scans ausgeschlossen wurde.

Betriebssystem — In dieser Spalte wird der Betriebssystemtyp angezeigt. Dieser kann`WINDOWS`, `MACLINUX`, oder sein`UNKNOWN`.

Überwacht mit — In dieser Spalte wird angezeigt, ob Amazon Inspector für diese Instance die agentenbasierte oder die agentenlose Scanmethode verwendet.

Zuletzt gescannt — In dieser Spalte wird angezeigt, wann Amazon Inspector diese Ressource zuletzt auf Sicherheitslücken überprüft hat. Die Häufigkeit, mit der Amazon Inspector Scans durchführt, hängt von der Scanmethode ab, die zum Scannen der Instance verwendet wird.

Um weitere Details zu einer EC2 Instance zu überprüfen, wählen Sie den Link in der EC2 Instance-Spalte. Amazon Inspector zeigt dann Details zur Instance und aktuelle Ergebnisse für die Instance an. Um die Details eines Ergebnisses zu überprüfen, wählen Sie den Link in der Titelspalte. Informationen zu diesen Details finden Sie unterDetails zu Ihren Amazon Inspector Inspector-Ergebnissen anzeigen.

Statuswerte für EC2 Amazon-Instances scannen

Für eine Amazon Elastic Compute Cloud (Amazon EC2) -Instance sind die möglichen Statuswerte wie folgt:

- Aktive Überwachung — Amazon Inspector überwacht und scannt die Instance kontinuierlich.

- Speicherlimit für agentenlose Instances überschritten — Amazon Inspector verwendet diesen Status, wenn die Gesamtgröße aller an eine Instance angehängten Volumes mehr als 1200 GB beträgt oder wenn an eine Instance mehr als 8 Volumes angehängt sind.
- Zeitlimit für die Erfassung agentenloser Instances überschritten — Amazon Inspector tritt beim Versuch, einen agentenlosen Scan auf einer Instance auszuführen, ein Timeout auf.
- EC2 Instance gestoppt — Amazon Inspector hat den Scanvorgang für die Instance angehalten, da sich die Instance in einem gestoppten Zustand befindet. Alle vorhandenen Ergebnisse bleiben bestehen, bis die Instance beendet wird. Wenn die Instance neu gestartet wird, fährt Amazon Inspector automatisch mit dem Scannen nach der Instance fort.
- Interner Fehler — Ein interner Fehler ist aufgetreten, als Amazon Inspector versuchte, die Instance zu scannen. Amazon Inspector behebt den Fehler automatisch und setzt den Scanvorgang so schnell wie möglich fort.
- Kein Inventar — Amazon Inspector konnte das Inventar der Softwareanwendung, das nach der Instance gescannt werden soll, nicht finden. Die Amazon Inspector Inspector-Verknüpfungen für die Instance wurden möglicherweise gelöscht oder konnten möglicherweise nicht ausgeführt werden.

  Um dieses Problem zu beheben, stellen Sie bitte AWS Systems Manager sicher, dass die `InspectorInventoryCollection-do-not-delete` Zuordnung besteht und ihr Zuordnungsstatus erfolgreich ist. Verwenden Sie außerdem AWS Systems Manager Fleet Manager, um das Softwareanwendungsinventar für die Instanz zu überprüfen.
- Ausstehende Deaktivierung — Amazon Inspector hat das Scannen der Instance beendet. Die Instance wird deaktiviert, bis die Bereinigungsaufgaben abgeschlossen sind.
- Erster Scan steht aus — Amazon Inspector hat die Instance für einen ersten Scan in die Warteschlange gestellt.
- Ressource beendet — Die Instance wurde beendet. Amazon Inspector bereinigt derzeit die vorhandenen Ergebnisse und Deckungsdaten für die Instanz.
- Veraltetes Inventar — Amazon Inspector war nicht in der Lage, ein aktualisiertes Softwareanwendungsinventar zu sammeln, das innerhalb der letzten 7 Tage für die Instance erfasst wurde.

  Um dieses Problem zu beheben, stellen Sie AWS Systems Manager sicher, dass die erforderlichen Amazon Inspector Inspector-Verknüpfungen vorhanden sind und für die Instance ausgeführt werden. Verwenden Sie außerdem AWS Systems Manager Fleet Manager, um den Bestand der Softwareanwendungen für die Instance zu überprüfen.

- Nicht verwaltete EC2 Instance — Amazon Inspector überwacht oder scannt die Instance nicht. Die Instance wird nicht von AWS Systems Manager verwaltet.

  Um dieses Problem zu beheben, können Sie das von AWS Systems Manager Automation AWSSupport-TroubleshootManagedInstance runbookbereitgestellte Tool verwenden. Nachdem Sie die Konfiguration AWS Systems Manager für die Verwaltung der Instance vorgenommen haben, beginnt Amazon Inspector automatisch, die Instance kontinuierlich zu überwachen und zu scannen.
- Nicht unterstütztes Betriebssystem — Amazon Inspector überwacht oder scannt die Instance nicht. Die Instance verwendet ein Betriebssystem oder eine Architektur, die Amazon Inspector nicht unterstützt. Eine Liste der Betriebssysteme, die Amazon Inspector unterstützt, finden Sie unterStatuswerte für EC2 Amazon-Instances.
- Aktive Überwachung mit teilweisen Fehlern — Dieser Status bedeutet, dass das EC2 Scannen zwar aktiv ist, aber es liegen Fehler vorTiefeninspektion von Amazon Inspector für Linux-basierte EC2 Amazon-Instances. Bei tiefgreifenden Inspektionen kann es sich um folgende Fehler handeln:
  - Das Limit für die Erfassung von Paketen mit Tiefeninspektion wurde überschritten — Die Instance hat das Limit von 5000 Paketen für die Tiefeninspektion von Amazon Inspector überschritten. Um die Tiefeninspektion für diese Instance fortzusetzen, können Sie versuchen, die mit dem Konto verknüpften benutzerdefinierten Pfade anzupassen.
  - Das tägliche SSM-Inventarlimit für Deep Inspection wurde überschritten — Der SSM-Agent konnte kein Inventar an Amazon Inspector senden, da das SSM-Kontingent für die pro Instance und Tag gesammelten SSM-Inventardaten für diese Instance bereits erreicht wurde. Weitere Informationen finden Sie unter Amazon EC2 Systems Manager Manager-Endpunkte und Kontingente.
  - Zeitlimit für die Abholung bei Tiefeninspektion überschritten — Amazon Inspector konnte den Paketbestand nicht extrahieren, da die Paketabholzeit den maximalen Schwellenwert von 15 Minuten überschritten hat.
  - Deep Inspection hat kein Inventar — Das Amazon Inspector SSM-Plugin war noch nicht in der Lage, ein Inventar von Paketen für diese Instanz zu sammeln. Dies ist normalerweise das Ergebnis eines ausstehenden Scans. Wenn dieser Status jedoch nach 6 Stunden weiterhin besteht, verwenden Sie Amazon EC2 Systems Manager, um sicherzustellen, dass die erforderlichen Amazon Inspector Inspector-Verknüpfungen vorhanden sind und für die Instance ausgeführt werden.

Einzelheiten zur Konfiguration der Scaneinstellungen für eine EC2 Instance finden Sie unterScannen Amazon EC2 Amazon-Instanzen.

Bewertung der Abdeckung von Amazon ECR-Repositorien

Auf der Registerkarte Repositorys werden Amazon ECR-Repositorys in Ihrer Umgebung angezeigt. AWS Die Listen sind auf den folgenden Registerkarten in Gruppen unterteilt:

- Alle — Zeigt alle Repositorys in Ihrer Umgebung an. In der Spalte Status wird der aktuelle Scanstatus für ein Repository angezeigt.
- Aktiviert — Zeigt alle Repositorys an, für deren Überwachung und Scannen Amazon Inspector in Ihrer Umgebung konfiguriert ist. Die Spalte Status zeigt den aktuellen Scanstatus für ein Repository an.
- Nicht aktiviert — Zeigt alle Repositorys an, die Amazon Inspector in Ihrer Umgebung nicht überwacht und scannt. Die Spalte Grund gibt an, warum Amazon Inspector ein Repository nicht überwacht und scannt.

Auf jeder Registerkarte gibt die Spalte Konto an, wer Eigentümer eines Repositorys ist. AWS-Konto

Um weitere Details zu einem Repository zu überprüfen, wählen Sie den Namen des Repositorys. Amazon Inspector zeigt dann eine Liste der Container-Images im Repository sowie Details zu jedem Bild an. Zu den Details gehören das Image-Tag, der Image-Digest und der Scanstatus. Sie enthalten auch wichtige Ergebnisstatistiken, wie z. B. die Anzahl kritischer Ergebnisse für das Bild. Wählen Sie das Bild-Tag für das Bild aus, um weitere Informationen zu erhalten und die unterstützenden Daten für die Suche nach Statistiken zu überprüfen.

Statuswerte für Amazon ECR-Repositorys scannen

Für ein Amazon Elastic Container Registry (Amazon ECR) -Repository sind die möglichen Statuswerte:

- Aktiviert (Kontinuierlich) — Für ein Repository überwacht Amazon Inspector kontinuierlich die Bilder in diesem Repository. Die erweiterte Scan-Einstellung für das Repository ist auf kontinuierliches Scannen eingestellt. Amazon Inspector scannt neue Bilder zunächst, wenn sie übertragen werden, und scannt Bilder erneut, wenn ein neues CVE veröffentlicht wird, das für dieses Bild relevant ist. Amazon Inspector überwacht weiterhin die Bilder in diesem Repository für die Dauer des Amazon ECR-Neuscans, die Sie konfigurieren.
- Aktiviert (bei Push) — Amazon Inspector scannt automatisch einzelne Container-Images im Repository, wenn ein neues Image übertragen wird. Das erweiterte Scannen ist für das Repository aktiviert und so eingestellt, dass bei Push gescannt wird.

- Zugriff verweigert — Amazon Inspector darf weder auf das Repository noch auf Container-Images im Repository zugreifen.

  Um dieses Problem zu beheben, stellen Sie sicher, dass AWS Identity and Access Management (IAM-) Richtlinien für das Repository Amazon Inspector den Zugriff auf das Repository ermöglichen.
- Deaktiviert (Manuell) — Amazon Inspector überwacht oder scannt keine Container-Images im Repository. Die Amazon ECR-Scaneinstellung für das Repository ist auf einfaches manuelles Scannen eingestellt.

  Um mit dem Scannen von Bildern im Repository mit Amazon Inspector zu beginnen, ändern Sie die Scan-Einstellung für das Repository auf Erweitertes Scannen und wählen Sie dann, ob Bilder kontinuierlich oder nur dann gescannt werden sollen, wenn ein neues Bild übertragen wird.
- Aktiviert (bei Push) — Amazon Inspector scannt automatisch einzelne Container-Images im Repository, wenn ein neues Image übertragen wird. Die erweiterte Scan-Einstellung für das Repository ist auf Scan on Push eingestellt.
- Interner Fehler — Ein interner Fehler ist aufgetreten, als Amazon Inspector versuchte, das Repository zu scannen. Amazon Inspector behebt den Fehler automatisch und setzt den Scanvorgang so schnell wie möglich fort.

Für Einzelheiten zur Konfiguration der Scaneinstellungen für RepositorysScannen von Amazon ECR-Container-Bildern.

Bewertung der Reichweite von Amazon ECR-Container-Images

Auf der Registerkarte Images werden Amazon ECR-Container-Images in Ihrer AWS Umgebung angezeigt. Die Listen sind auf den folgenden Registerkarten in Gruppen unterteilt:

- Alle — Zeigt alle Container-Images in Ihrer Umgebung an. In der Spalte Status wird der aktuelle Scanstatus für ein Bild angezeigt.
- Scannen — Zeigt alle Container-Images an, für deren Überwachung und Scannen Amazon Inspector in Ihrer Umgebung konfiguriert ist. Die Spalte Status zeigt den aktuellen Scanstatus für ein Bild an.
- Nicht scannen — Zeigt alle Container-Images an, die Amazon Inspector in Ihrer Umgebung nicht überwacht und scannt. Die Spalte Grund gibt an, warum Amazon Inspector ein Bild nicht überwacht und scannt.

Ein Container-Bild kann aus verschiedenen Gründen auf der Registerkarte Nicht aktiviert angezeigt werden. Das Bild ist möglicherweise in einem Repository gespeichert, für das Amazon Inspector-Scans nicht aktiviert sind, oder Amazon ECR-Filterregeln verhindern, dass dieses Repository gescannt wird. Oder das Bild wurde nicht innerhalb der Anzahl von Tagen übertragen oder abgerufen, die Sie für die Dauer des erneuten ECR-Scans konfiguriert haben. Weitere Informationen finden Sie unter Konfiguration der Dauer des Amazon ECR-Neuscans.

Auf jeder Registerkarte gibt die Spalte Repository-Name den Namen des Repositorys an, das ein Container-Image speichert. In der Spalte Konto wird der AWS-Konto Eigentümer des Repositorys angegeben. In der Spalte Zuletzt gescannt wird angezeigt, wann Amazon Inspector diese Ressource zuletzt auf Sicherheitslücken überprüft hat. Dies kann Prüfungen beinhalten, wenn die Suchmetadaten aktualisiert wurden, wenn das Anwendungsinventar der Ressource aktualisiert wurde oder wenn als Reaktion auf einen neuen CVE ein erneuter Scan durchgeführt wird. Weitere Informationen finden Sie unter Scanverhalten für Amazon ECR-Scans.

Um weitere Details zu einem Container-Image zu überprüfen, wählen Sie den Link in der Spalte ECR-Container-Image. Amazon Inspector zeigt dann Details zum Bild und aktuelle Ergebnisse für das Bild an. Um die Details eines Ergebnisses zu überprüfen, wählen Sie den Link in der Titelspalte. Informationen zu diesen Details finden Sie unterDetails zu Ihren Amazon Inspector Inspector-Ergebnissen anzeigen.

Statuswerte für Amazon ECR-Container-Images scannen

Für ein Amazon Elastic Container Registry-Container-Image sind die möglichen Statuswerte wie folgt:

- Aktive Überwachung (kontinuierlich) — Amazon Inspector überwacht kontinuierlich das Bild und es werden neue Scans durchgeführt, wenn ein neuer relevanter CVE veröffentlicht wird. Die Dauer des Amazon ECR-Rescans für das Bild wird jedes Mal aktualisiert, wenn das Bild übertragen oder abgerufen wird. Das erweiterte Scannen ist für das Repository aktiviert, in dem das Bild gespeichert ist, und die erweiterte Scan-Einstellung für das Repository ist auf kontinuierliches Scannen eingestellt.
- Aktiviert (bei Push) — Amazon Inspector scannt das Bild automatisch jedes Mal, wenn ein neues Bild übertragen wird. Das erweiterte Scannen ist für das Repository aktiviert, in dem das Bild gespeichert ist, und die erweiterte Scan-Einstellung für das Repository ist auf Scannen bei Push eingestellt.

- Interner Fehler — Ein interner Fehler ist aufgetreten, als Amazon Inspector versuchte, das Container-Image zu scannen. Amazon Inspector behebt den Fehler automatisch und setzt den Scanvorgang so schnell wie möglich fort.
- Erster Scan steht aus — Amazon Inspector hat das Bild für einen ersten Scan in die Warteschlange gestellt.
- Die Scanberechtigung ist abgelaufen (Fortlaufend) — Amazon Inspector hat den Scanvorgang für das Bild ausgesetzt. Das Bild wurde nicht innerhalb der Dauer aktualisiert, die Sie für automatische erneute Scans von Bildern im Repository angegeben haben. Sie können das Bild verschieben oder ziehen, um den Scanvorgang fortzusetzen.
- Die Scanberechtigung ist abgelaufen (On Push) — Amazon Inspector hat den Scanvorgang für das Bild ausgesetzt. Das Bild wurde innerhalb des Zeitraums, den Sie für automatische erneute Scans von Bildern im Repository angegeben haben, nicht aktualisiert. Sie können das Bild per Push übertragen, um den Scanvorgang fortzusetzen.
- Manuelles Scannen der Frequenz (Manuell) — Amazon Inspector scannt das Amazon ECR-Container-Image nicht. Die Amazon ECR-Scaneinstellung für das Repository, in dem das Bild gespeichert ist, ist auf einfaches manuelles Scannen eingestellt. Um das Bild automatisch mit Amazon Inspector zu scannen, ändern Sie die Repository-Einstellung auf Verbessertes Scannen und wählen Sie dann, ob Bilder kontinuierlich oder nur gescannt werden sollen, wenn ein neues Bild übertragen wird.
- Nicht unterstütztes Betriebssystem — Amazon Inspector überwacht oder scannt das Bild nicht. Das Bild basiert auf einem Betriebssystem, das Amazon Inspector nicht unterstützt, oder es verwendet einen Medientyp, den Amazon Inspector nicht unterstützt.

  Eine Liste der Betriebssysteme, die Amazon Inspector unterstützt, finden Sie unterUnterstützte Betriebssysteme: Amazon ECR-Scannen mit Amazon Inspector. Eine Liste der Medientypen, die Amazon Inspector unterstützt, finden Sie unter Unterstützte Medientypen.

Einzelheiten zur Konfiguration der Scaneinstellungen für Repositorys und Bilder finden Sie unterScannen von Amazon ECR-Container-Bildern.

Bewertung des Funktionsumfangs AWS Lambda

Auf der Registerkarte Lambda werden Lambda-Funktionen in Ihrer AWS Umgebung angezeigt. Auf dieser Seite gibt es zwei Tabellen, eine mit Details zur Funktionsabdeckung für das Lambda-Standardscannen und eine andere für das Scannen von Lambda-Code. Sie können Funktionen auf der Grundlage der folgenden Registerkarten gruppieren:

- Alle — Zeigt alle Lambda-Funktionen in Ihrer Umgebung an. Die Spalte Status zeigt den aktuellen Scanstatus für eine Lambda-Funktion an.
- Scannen — Zeigt die Lambda-Funktionen an, für deren Scannen Amazon Inspector konfiguriert ist. Die Spalte Status zeigt den aktuellen Scanstatus für jede Lambda-Funktion an.
- Nicht scannen — Zeigt die Lambda-Funktionen an, für deren Scannen Amazon Inspector nicht konfiguriert ist. Die Spalte Grund gibt an, warum Amazon Inspector eine Funktion nicht überwacht und scannt.

  Eine Lambda-Funktion kann aus verschiedenen Gründen auf der Registerkarte Nicht scannen angezeigt werden. Die Lambda-Funktion gehört möglicherweise zu einem Konto, das nicht zu Amazon Inspector hinzugefügt wurde, oder Filterregeln verhindern, dass diese Funktion gescannt wird. Weitere Informationen finden Sie unter Scannen mit Lambda-Funktionen.

Auf jeder Registerkarte gibt die Spalte Funktionsname den Namen der Lambda-Funktion an. In der Spalte Konto wird der Eigentümer AWS-Konto der Funktion angegeben. Runtime gibt die Laufzeit der Funktion an. Die Spalte Status zeigt den aktuellen Scanstatus für jede Lambda-Funktion an. Resource Tags zeigt die Tags an, die auf die Funktion angewendet wurden. In der Spalte Zuletzt gescannt wird angezeigt, wann Amazon Inspector diese Ressource zuletzt auf Sicherheitslücken überprüft hat. Dies kann Prüfungen beinhalten, wenn die Suchmetadaten aktualisiert wurden, wenn das Anwendungsinventar der Ressource aktualisiert wurde oder wenn als Reaktion auf einen neuen CVE ein erneuter Scan durchgeführt wird. Weitere Informationen finden Sie unter Scanverhalten beim Scannen mit Lambda-Funktionen.

Statuswerte für Funktionen werden gescannt AWS Lambda

Für eine Lambda-Funktion sind folgende Statuswerte möglich:

- Aktive Überwachung — Amazon Inspector überwacht und scannt kontinuierlich Lambda-Funktionen. Kontinuierliches Scannen umfasst einen ersten Scan neuer Funktionen, wenn sie in das Repository übertragen werden, und automatische erneute Scans von Funktionen, wenn sie aktualisiert werden oder wenn neue Common Vulnerabilities and Exposures (CVEs) veröffentlicht werden.
- Nach Tag ausgeschlossen — Amazon Inspector scannt diese Funktion nicht, da sie von Tag-Scans ausgeschlossen wurde.
- Die Scanberechtigung ist abgelaufen — Amazon Inspector überwacht diese Funktion nicht, da seit dem letzten Aufruf oder der letzten Aktualisierung mindestens 90 Tage vergangen sind.

- Interner Fehler — Beim Versuch von Amazon Inspector, die Funktion zu scannen, ist ein interner Fehler aufgetreten. Amazon Inspector behebt den Fehler automatisch und setzt den Scanvorgang so schnell wie möglich fort.
- Erster Scan steht aus — Amazon Inspector hat die Funktion für einen ersten Scan in die Warteschlange gestellt.
- Nicht unterstützt — Die Lambda-Funktion hat eine Laufzeit, die nicht unterstützt wird.

Verwaltung mehrerer Konten in Amazon Inspector mit AWS Organizations

Sie können Amazon Inspector verwenden, um mehrere Konten in einer Organisation zu verwalten. Dazu müssen Sie Amazon Inspector mit dem AWS Organizations Verwaltungskonto aktivieren und einen delegierten Administrator angeben. Der delegierte Administrator verwaltet Amazon Inspector für eine Organisation und kann Aufgaben im Namen der Organisation ausführen. In den folgenden Themen wird der Unterschied zwischen einem delegierten Administratorkonto und einem Mitgliedskonto beschrieben, wie ein delegierter Administrator benannt und entfernt wird und wie Mitgliedskonten verwaltet werden.

Themen

Grundlegendes zum delegierten Administratorkonto und Mitgliedskonto in Amazon Inspector

Wenn Sie Amazon Inspector in einer Umgebung mit mehreren Konten verwenden, hat das delegierte Administratorkonto Zugriff auf bestimmte Metadaten. Die Metadaten umfassen Standardscans für Amazon EC2, Amazon ECR und Lambda sowie Lambda-Code-Scans. Es enthält auch Ergebnisse von Sicherheitsüberprüfungen für Mitgliedskonten. Dieser Abschnitt enthält Informationen darüber, welche Aktionen das delegierte Administratorkonto und die Mitgliedskonten ausführen können.

Delegierte Administratoraktionen

Wenn der delegierte Administrator Einstellungen auf sein Konto anwendet, werden diese Einstellungen im Allgemeinen auf alle anderen Konten in der Organisation angewendet. Der delegierte Administrator kann auch Informationen für sein eigenes Konto und jedes zugeordnete Mitglied anzeigen und abrufen. Ein delegiertes Administratorkonto von Amazon Inspector kann die folgenden Aktionen ausführen:

- Nur das AWS Organizations Verwaltungskonto kann einen delegierten Administrator benennen und entfernen.

- Wenn Sie einen delegierten Administrator benennen, müssen Sie derselben Organisation angehören wie die Mitgliedskonten, die Sie verwalten möchten.
- Den Status von Amazon Inspector für zugehörige Konten anzeigen und verwalten, einschließlich der Aktivierung und Deaktivierung von Amazon Inspector.
- Aktivieren oder deaktivieren Sie die Scantypen für alle Mitgliedskonten in der Organisation.
- Zeigen Sie aggregierte Suchdaten für die gesamte Organisation und Suchdetails für alle Mitgliedskonten innerhalb der Organisation an.
- Erstellen und verwalten Sie Unterdrückungsregeln, die für Ergebnisse aller Konten in der Organisation gelten.
- Aktivieren Sie das erweiterte Scannen von Amazon ECR für alle Mitglieder der Organisation.
- Zeigen Sie die Ressourcenabdeckung für die gesamte Organisation an.
- Definieren Sie die Dauer für automatische erneute Scans von ECR-Container-Images für alle Mitgliedskonten in der Organisation. Die Einstellung für die Scandauer des delegierten Administrators hat Vorrang vor allen Einstellungen, die das Mitgliedskonto zuvor festgelegt hat. Alle Konten in der Organisation teilen sich die Dauer der automatisierten erneuten Scans von Amazon ECR für die delegierten Administratoren. Sie können für einzelne Konten keine unterschiedlichen Dauern für erneute Scans festlegen.
- Geben Sie fünf benutzerdefinierte Pfade für Amazon Inspector Deep Inspection for Amazon an EC2 , die für alle Konten in der Organisation verwendet werden. Dies gilt zusätzlich zu den fünf benutzerdefinierten Pfaden, die ein delegierter Administrator für sein individuelles Konto festlegen kann. Weitere Informationen zur Konfiguration benutzerdefinierter Pfade für Deep Inspection finden Sie unterBenutzerdefinierte Pfade für die Tiefeninspektion mit Amazon Inspector.
- Aktivieren und deaktivieren Sie Amazon Inspector Deep Inspection für Mitgliedskonten.
- Export SBOMs für alle Mitgliedskonten in der Organisation.
- Stellen Sie den EC2 Amazon-Scanmodus für alle Mitgliedskonten in der Organisation ein. Weitere Informationen finden Sie unter Den Scanmodus verwalten.
- Erstellen und verwalten Sie CIS-Scankonfigurationen für alle Konten in der Organisation, mit Ausnahme von Scankonfigurationen, die von Mitgliedskonten erstellt wurden.

> **Note**
>
> Wenn ein Mitgliedskonto die Organisation verlässt, kann der delegierte Administrator die von diesem Konto geplanten Scankonfigurationen nicht mehr sehen.

- Zeigen Sie die CIS-Scanergebnisse für alle Konten in der Organisation an.

Aktionen für Mitgliedskonten

Ein Mitgliedskonto kann Informationen zu seinem Konto in Amazon Inspector einsehen und abrufen, während die Einstellungen für sein Konto vom delegierten Administrator verwaltet werden. Mitgliedskonten innerhalb einer Organisation können die folgenden Aktionen in Amazon Inspector ausführen:

- Aktivieren Sie Amazon Inspector für ihr eigenes Konto.
- Sehen Sie sich die Ressourcenabdeckung für ihr eigenes Konto an.
- Details zu den Ergebnissen für ihr eigenes Konto anzeigen.
- Sehen Sie sich die Einstellung für die Dauer des automatischen erneuten Scans des ECR-Container-Images für ihr eigenes Konto an.
- Geben Sie fünf benutzerdefinierte Pfade für die Tiefeninspektion von Amazon Inspector an EC2 , die für ihr individuelles Konto verwendet werden. Diese Pfade werden zusätzlich zu allen benutzerdefinierten Pfaden gescannt, die der delegierte Administrator für die Organisation angegeben hat. Weitere Informationen zur Konfiguration von Deep-Inspection-Pfaden finden Sie unterBenutzerdefinierte Pfade für die Tiefeninspektion mit Amazon Inspector.
- Sehen Sie sich die benutzerdefinierten Pfade an, die von Ihrem delegierten Administrator für Amazon Inspector Deep Inspection festgelegt wurden.
- Exportieren Sie SBOMs für alle Ressourcen, die mit ihrem Konto verknüpft sind.
- Sehen Sie sich den Scanmodus für ihr Konto an.
- Erstellen und verwalten Sie CIS-Scankonfigurationen für ihr Konto.
- Sehen Sie sich die Ergebnisse aller CIS-Scans nach Ressourcen in ihrem Konto an, einschließlich der vom delegierten Administrator geplanten Scans.

> **Note**
>
> Nach der Aktivierung kann Amazon Inspector nur durch ein delegiertes Administratorkonto deaktiviert werden.

Benennen eines delegierten Administratorkontos für Amazon Inspector

Der delegierte Administrator ist ein Konto, das einen Dienst für eine Organisation verwaltet. In diesem Thema wird beschrieben, wie Sie einen delegierten Administrator für Amazon Inspector bestimmen.

Überlegungen

Bevor Sie einen delegierten Administrator benennen, beachten Sie Folgendes:

Der delegierte Administrator kann maximal 10.000 Mitglieder verwalten.

- Wenn Sie mehr als 10.000 Mitgliedskonten haben, erhalten Sie eine Benachrichtigung über das Amazon CloudWatch Personal Health Dashboard und eine E-Mail an das delegierte Administratorkonto.

Der delegierte Administrator ist Regional.

- Amazon Inspector ist ein regionaler Service. Sie müssen die Schritte des Verfahrens überall wiederholen, AWS-Region wo Sie Amazon Inspector verwenden möchten.

Eine Organisation kann nur einen delegierten Administrator haben.

- Wenn Sie in einem Konto ein Konto als delegierten Administrator festlegen AWS-Region, muss dieses Konto in allen anderen Konten der delegierte Administrator sein. AWS-Regionen

Durch das Ändern eines delegierten Administrators wird Amazon Inspector für Mitgliedskonten nicht deaktiviert.

- Wenn Sie einen delegierten Administrator entfernen, werden Mitgliedskonten zu eigenständigen Konten, und die Scaneinstellungen sind davon nicht betroffen.

In Ihrem AWS Unternehmen müssen alle Funktionen aktiviert sein.

- Dies ist die Standardeinstellung für AWS Organizations. Falls sie nicht aktiviert ist, finden Sie weitere Informationen unter Alle Funktionen in Ihrer Organisation aktivieren.

Erforderliche Berechtigungen zum designieren eines delegierten Administrators

Sie benötigen die Erlaubnis, Amazon Inspector zu aktivieren und einen delegierten Amazon Inspector-Administrator zu benennen. Fügen Sie am Ende Ihrer IAM-Richtlinie die folgende Erklärung

hinzu, um diese Berechtigungen zu gewähren. Weitere Informationen finden Sie unter Verwaltung von IAM-Richtlinien.

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

Benennen eines delegierten Administrators für Ihre Organisation AWS

Im folgenden Verfahren wird beschrieben, wie Sie einen delegierten Administrator für Ihre Organisation bestimmen. Bevor Sie das Verfahren abschließen, stellen Sie sicher, dass Sie sich in derselben Organisation befinden wie die Mitgliedskonten, die der delegierte Administrator verwalten soll.

> **Note**
>
> Sie müssen das AWS Organizations Verwaltungskonto verwenden, um dieses Verfahren abzuschließen. Nur das AWS Organizations Verwaltungskonto kann einen delegierten Administrator benennen. Für die Benennung eines delegierten Administrators sind möglicherweise Berechtigungen erforderlich. Weitere Informationen finden Sie unter Erforderliche Berechtigungen zum designieren eines delegierten Administrators.

Wenn Sie Amazon Inspector zum ersten Mal aktivieren, erstellt Amazon Inspector die serviceverknüpfte Rolle `AWSServiceRoleForAmazonInspector` für das Konto. Informationen darüber, wie Amazon Inspector serviceverknüpfte Rollen verwendet, finden Sie unterVerwenden von serviceverknüpften Rollen für Amazon Inspector.

Console

So benennen Sie einen delegierten Administrator für Amazon Inspector

1. Melden Sie sich beim AWS Organizations Verwaltungskonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Verwenden Sie den AWS-Region Selektor, um den Ort anzugeben, an AWS-Region dem Sie den delegierten Administrator benennen möchten.
3. Wählen Sie im Navigationsbereich Allgemeine Einstellungen aus.
4. Geben Sie unter Delegierter Administrator die 12-stellige ID des Administrators ein, den AWS-Konto Sie als delegierten Administrator festlegen möchten.
5. Wählen Sie Delegieren und dann erneut Delegieren aus.

Wenn Sie einen delegierten Administrator benennen, sind standardmäßig alle Scantypen für das Konto aktiviert. Wenn Sie Amazon Inspector für das AWS Organizations Verwaltungskonto aktivieren möchten, gehen Sie wie folgt vor.

Um Amazon Inspector für das AWS Organizations Verwaltungskonto zu aktivieren

1. Melden Sie sich mit dem delegierten Administratorkonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus.
3. Wählen Sie unter Konten das AWS Organizations Verwaltungskonto aus und wählen Sie dann Aktivieren aus.
4. Wählen Sie aus, welche Scantypen Sie für das AWS Organizations Verwaltungskonto aktivieren möchten, und klicken Sie dann auf Absenden.

API

Benennen Sie einen delegierten Administrator, der die API verwendet

- Führen Sie den EnableDelegatedAdminAccountAPI-Vorgang mit den Anmeldeinformationen AWS-Konto des Verwaltungskontos der Organizations. Sie können dazu auch den verwenden AWS Command Line Interface , indem Sie den folgenden CLI-Befehl

ausführen:`aws inspector2 enable-delegated-admin-account --delegated-admin-account-id 11111111111`.

> **Note**
>
> Stellen Sie sicher, dass Sie die Konto-ID des Kontos angeben, das Sie zu einem von Amazon Inspector delegierten Administrator machen möchten.

Die Aktivierung von Amazon Inspector scannt nach Mitgliedskonten

Wenn Sie der delegierte Administrator einer Organisation sind, können Sie Amazon EC2 - und Amazon ECR-Scans für Mitgliedskonten in der Organisation aktivieren. Sobald Sie das Scannen für ein Mitgliedskonto aktiviert haben, wird Amazon Inspector automatisch für dieses Konto aktiviert und das Konto wird mit dem delegierten Administratorkonto verknüpft. Informationen zu den Scantypen von Amazon Inspector finden Sie unterAutomatisierte Scantypen in Amazon Inspector. In diesem Abschnitt wird beschrieben, wie Sie das Scannen für Mitgliedskonten aktivieren.

Aktivieren Sie das Scannen nach Mitgliedskonten

Sie können das Scannen nach Mitgliedskonten auf verschiedene Arten aktivieren. In den folgenden Verfahren wird beschrieben, wie Sie als delegierter Administrator das Scannen für alle Mitgliedskonten und für bestimmte Mitgliedskonten aktivieren und wie Sie das Scannen als Mitgliedskonto aktivieren.

Um das Scannen automatisch für alle Mitgliedskonten zu aktivieren

1. Melden Sie sich mit den Anmeldeinformationen für das delegierte Administratorkonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Verwenden Sie die Regionsauswahl, um den Ort auszuwählen, an AWS-Region dem Sie das Scannen für alle Mitgliedskonten aktivieren möchten.
3. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus. Auf der Registerkarte Konten werden alle Mitgliedskonten angezeigt, die dem AWS Organizations Verwaltungskonto zugeordnet sind.
4. Wählen Sie unter Organisation das Kästchen neben Kontonummer aus. Wählen Sie dann Aktivieren, um auszuwählen, welche Scanoptionen Sie auf Mitgliedskonten anwenden möchten. Sie können die folgenden Scanarten auswählen:

- EC2 Amazon-Scannen
- Amazon ECR-Scannen
- Lambda-Standardscannen
- Scannen von Lambda-Code

- Nachdem Sie Ihre bevorzugten Scantypen ausgewählt haben, wählen Sie Speichern.

> **Note**
>
> Wenn Sie mehrere Seiten mit Konten haben, müssen Sie diesen Schritt auf jeder Seite wiederholen. Sie können das Zahnradsymbol wählen, um die Anzahl der auf jeder Seite angezeigten Konten zu ändern.

5. Aktivieren Sie die Einstellung Inspector automatisch für neue Mitgliedskonten aktivieren und wählen Sie aus, welche Scanoptionen Sie auf neue Mitgliedskonten anwenden möchten, die zu Ihrer Organisation hinzugefügt wurden. Sie können die folgenden Scanarten auswählen:

   - EC2 Amazon-Scannen
   - Amazon ECR-Scannen
   - Lambda-Standardscannen
   - Scannen von Lambda-Code

   - Nachdem Sie Ihre bevorzugten Scantypen ausgewählt haben, wählen Sie Aktivieren.

   > **Note**
   >
   > Die Einstellung Inspector automatisch für neue Mitgliedskonten aktivieren aktiviert Amazon Inspector für alle future Mitglieder Ihrer Organisation.
   > Wenn die Anzahl der Mitgliedskonten mehr als 5.000 beträgt, wird diese Einstellung automatisch deaktiviert. Wenn die Gesamtzahl der Mitgliedskonten auf weniger als 5.000 sinkt, wird die Einstellung automatisch wieder aktiviert.

6. (Empfohlen) Wiederholen Sie jeden dieser Schritte an allen Stellen AWS-Region , an denen Sie die Suche nach Mitgliedskonten aktivieren möchten.

Um das Scannen für bestimmte Mitgliedskonten zu aktivieren

1. Melden Sie sich mit den Anmeldeinformationen für das delegierte Administratorkonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Verwenden Sie die Regionsauswahl, um den Ort auszuwählen, an AWS-Region dem Sie das Scannen für alle Mitgliedskonten aktivieren möchten.
3. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus. Auf der Registerkarte Konten werden alle Mitgliedskonten angezeigt, die dem AWS Organizations Verwaltungskonto zugeordnet sind.
4. Wählen Sie unter Organisation das Kästchen neben jeder Mitgliedskontonummer aus, für die Sie die Suche aktivieren möchten. Wählen Sie dann Aktivieren aus, um auszuwählen, welche Scanoptionen Sie auf Mitgliedskonten anwenden möchten. Sie können die folgenden Scanarten auswählen:

   - EC2 Amazon-Scannen
   - Amazon ECR-Scannen
   - Lambda-Standardscannen
   - Scannen von Lambda-Code

   - Nachdem Sie Ihre bevorzugten Scantypen ausgewählt haben, wählen Sie Speichern.

   > **Note**
   >
   > Wenn Sie mehrere Seiten mit Konten haben, müssen Sie diesen Schritt auf jeder Seite wiederholen. Sie können das Zahnradsymbol wählen, um die Anzahl der auf jeder Seite angezeigten Konten zu ändern.

5. (Empfohlen) Wiederholen Sie jeden dieser Schritte in allen Bereichen AWS-Region , in denen Sie das Scannen für bestimmte Mitglieder aktivieren möchten.

Um das Scannen als Mitgliedskonto zu aktivieren

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.

2. Verwenden Sie die Regionsauswahl, um den Ort auszuwählen, AWS-Region an dem Sie das Scannen für alle Mitgliedskonten aktivieren möchten.
3. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus. Auf der Registerkarte Konten werden alle Mitgliedskonten angezeigt, die dem AWS Organizations Verwaltungskonto zugeordnet sind.
4. Wählen Sie unter Organisation das Kästchen neben Ihrer Kontonummer aus. Wählen Sie dann Aktivieren, um auszuwählen, welche Scanoptionen Sie anwenden möchten. Sie können die folgenden Scantypen auswählen:

   - EC2 Amazon-Scannen
   - Amazon ECR-Scannen
   - Lambda-Standardscannen
   - Scannen von Lambda-Code

   - Nachdem Sie Ihre bevorzugten Scantypen ausgewählt haben, wählen Sie Speichern.
5. (Empfohlen) Wiederholen Sie diese Schritte in jeder Region, in der Sie das Scannen für Ihr Mitgliedskonto aktivieren möchten.

> **Note**
>
> Wenn Ihr AWS Organizations Verwaltungskonto über ein delegiertes Administratorkonto für Amazon Inspector verfügt, können Sie Ihr Konto als Mitgliedskonto aktivieren, um die Scandetails einzusehen.

Verknüpfung von Mitgliedskonten in Amazon Inspector aufheben

Als delegierter Administrator müssen Sie möglicherweise ein Mitgliedskonto von Ihrem Konto trennen. Wenn Sie die Verknüpfung mit einem Mitgliedskonto aufheben, ist Amazon Inspector weiterhin für das Konto aktiviert und das Konto wird zu einem eigenständigen Konto. Sie sind auch nicht mehr berechtigt, Amazon Inspector für das Konto zu verwalten. Sie können Ihrem Konto jedoch jederzeit zuvor getrennte Mitgliedskonten zuordnen. In diesem Abschnitt wird beschrieben, wie Sie als delegierter Administrator die Zuordnung von Mitgliedskonten aufheben können.

Console

So trennen Sie die Zuordnung von Mitgliedskonten mithilfe der Konsole

1. Melden Sie sich mit den Anmeldeinformationen für das delegierte Administratorkonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter v2/home https://console.aws.amazon.com/inspector/
2. Verwenden Sie die Regionsauswahl, um das Land auszuwählen, für das Sie die Zuordnung von AWS-Region Mitgliedskonten aufheben möchten.
3. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus.
4. Wählen Sie unter Organisation das Kästchen neben jeder Kontonummer aus, deren Zuordnung Sie aufheben möchten.
5. Wählen Sie im Menü Aktionen die Option Konto trennen aus.

API

So trennen Sie die Zuordnung von Mitgliedskonten mithilfe der API

Führen Sie den DisassociateMemberAPI-Vorgang aus. Geben Sie in der Anfrage das Konto an, dessen Verknüpfung IDs Sie aufheben möchten.

Den delegierten Administrator in Amazon Inspector entfernen

Möglicherweise müssen Sie das delegierte Administratorkonto von Amazon Inspector entfernen. Sie können dies über das AWS Organizations Verwaltungskonto tun. Wenn Sie das delegierte Administratorkonto von Amazon Inspector entfernen, ist Amazon Inspector weiterhin für das Konto und alle Mitgliedskonten aktiviert. Das delegierte Administratorkonto und alle zugehörigen Mitgliedskonten werden zu eigenständigen Konten und behalten ihre ursprünglichen Scaneinstellungen bei. In diesem Abschnitt wird beschrieben, wie das delegierte Administratorkonto entfernt wird.

Den delegierten Amazon Inspector-Administrator entfernen

Die folgenden Verfahren beschreiben, wie Sie den delegierten Amazon Inspector-Administrator entfernen und Mitgliedskonten mit dem delegierten Administratorkonto verknüpfen.

Informationen zum Zuweisen eines delegierten Amazon Inspector-Administrators finden Sie unter Benennen eines delegierten Administratorkontos für Amazon Inspector.

> **Note**
>
> Nachdem Sie einen delegierten Amazon Inspector-Administrator zugewiesen haben, muss der delegierte Amazon Inspector-Administrator die Mitgliedskonten manuell zuordnen.

Um den delegierten Administrator zu entfernen

1. Melden Sie sich AWS Management Console mit dem AWS Organizations Verwaltungskonto an.
2. Öffnen Sie die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
3. Verwenden Sie die Regionsauswahl, um den Ort auszuwählen, aus AWS-Region dem Sie den delegierten Administrator entfernen möchten.
4. Wählen Sie im Navigationsbereich Allgemeine Einstellungen aus.
5. Wählen Sie unter Delegierter Administrator die Option Entfernen aus, und bestätigen Sie dann Ihre Aktion.

Um Mitglieder einem neuen delegierten Administrator zuzuordnen

1. Melden Sie sich mit den Anmeldeinformationen für das delegierte Administratorkonto an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Verwenden Sie die Regionsauswahl, um auszuwählen, AWS-Region wo Sie Mitglieder zuordnen möchten.
3. Wählen Sie im Navigationsbereich die Option Kontoverwaltung aus.
4. Wählen Sie unter Organisation das Kästchen neben Kontonummer aus.
5. Wählen Sie Aktionen und dann Mitglied hinzufügen aus.

Taggen von Amazon Inspector Inspector-Ressourcen

Ein Tag ist eine Bezeichnung, die Sie einer AWS Ressource hinzufügen. Mithilfe von Tags können Sie AWS Ressourcen anhand bestimmter Kriterien kategorisieren. Tags bestehen aus einem Schlüssel-Wert-Paar. Der Tag-Schlüssel ist eine allgemeine Bezeichnung. Der Tag-Wert ist eine Beschreibung des Tag-Schlüssels. Mit Amazon Inspector können Sie Unterdrückungsregeln und CIS-Scankonfigurationen kennzeichnen. Sie können jeder Ihrer Amazon Inspector Inspector-Ressourcen bis zu 50 Tags hinzufügen.

Grundlagen des Kennzeichnens

Ein Tag besteht aus einem Schlüssel-Wert-Paar. Der Tag-Schlüssel ist eine allgemeine Bezeichnung. Der Tag-Wert ist eine Beschreibung des Tag-Schlüssels. In diesem Thema werden die Grundlagen der Kennzeichnung von Amazon Inspector Inspector-Ressourcen beschrieben. Beachten Sie beim Taggen von Amazon Inspector Inspector-Ressourcen Folgendes:

- Sie können Regeln zur Unterdrückung von Tags und CIS-Scankonfigurationen kennzeichnen.
- Sie können jeder Ihrer Amazon Inspector Inspector-Ressourcen bis zu 50 Tags hinzufügen.
- Tag-Schlüssel müssen eindeutig sein.
- Ein Tag-Schlüssel kann nur einen Tag-Wert haben.
- Tag-Schlüssel und Tag-Werte können maximal 128 UTF-8-Zeichen enthalten. Bei den Zeichen kann es sich um Buchstaben, Zahlen, Leerzeichen oder die folgenden Symbole handeln:. `_ . : / = + - @`
- Sie können das `aws` Präfix in keinem Ihrer Tags verwenden oder Tags mit diesem Präfix ändern. Tags mit dem `aws` Präfix sind für die Verwendung durch reserviert AWS.
- Tags, die einer Amazon Inspector Inspector-Ressource zugewiesen wurden, sind nur in Ihrem AWS Konto und dort verfügbar AWS-Region , wo Sie sie erstellt haben.
- Wenn Sie eine Ressource löschen, werden auch alle damit verknüpften Tags gelöscht.

Weitere Informationen zu Tags finden Sie unter Bewährte Methoden und Strategien im Tagging AWS Resources and Tag Editor User Guide.

> **Note**
>
> Tags sind nicht dazu bestimmt, vertrauliche oder sensible Informationen zu speichern. Verwenden Sie niemals Tags, um diese Art von Daten zu speichern. Auf Tags kann von anderen AWS Diensten aus zugegriffen werden.

Hinzufügen von Tags

Sie können Tags zu Amazon Inspector Inspector-Ressourcen hinzufügen. Zu diesen Ressourcen gehören Unterdrückungsregeln und CIS-Scankonfigurationen. Mithilfe von Tags können Sie AWS Ressourcen anhand bestimmter Kriterien kategorisieren. In diesem Thema wird beschrieben, wie Tags zu Amazon Inspector Inspector-Ressourcen hinzugefügt werden.

Hinzufügen von Tags zu Amazon Inspector Inspector-Ressourcen

Sie können Regeln zur Unterdrückung von Tags und CIS-Scankonfigurationen festlegen. Die folgenden Verfahren beschreiben, wie Sie Tags in der Konsole und mit der Amazon Inspector Inspector-API hinzufügen.

Hinzufügen von Tags in der Konsole

Sie können Tags zu Amazon Inspector Inspector-Ressourcen in der Konsole hinzufügen.

Hinzufügen von Tags zu Unterdrückungsregeln

Sie können den Unterdrückungsregeln während der Erstellung Tags hinzufügen. Weitere Informationen finden Sie unter Unterdrückungsregel erstellen.

Sie können eine Unterdrückungsregel auch bearbeiten, um Tags einzubeziehen. Weitere Informationen finden Sie unter Eine Unterdrückungsregel bearbeiten.

Hinzufügen von Tags zu einer CIS-Scankonfiguration

Sie können einer CIS-Scankonfiguration während der Erstellung Tags hinzufügen. Weitere Informationen finden Sie unter Eine CIS-Scankonfiguration erstellen.

Sie können eine CIS-Scankonfiguration auch so bearbeiten, dass sie Tags enthält. Weitere Informationen finden Sie unter Eine CIS-Scankonfiguration bearbeiten.

Hinzufügen von Tags mit der Amazon Inspector API

Mit der Amazon Inspector-API können Sie Amazon Inspector-Ressourcen Tags hinzufügen.

Hinzufügen von Tags zu Amazon Inspector Inspector-Ressourcen

Verwenden Sie die `TagResource` API, um Tags zu Amazon Inspector Inspector-Ressourcen hinzuzufügen. Sie müssen den ARN der Ressource und das Schlüssel-Wert-Paar für das Tag in den Befehl aufnehmen. Der folgende Beispielbefehl verwendet einen leeren Ressourcen-ARN für einen Unterdrückungsfilter. Der Schlüssel ist `CostAllocation` und der Wert ist`dev`. Informationen zu Ressourcentypen für Amazon Inspector finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Inspector2 in der Service Authorization Reference.

```
aws inspector2 tag-resource \
--resource-arn "arn:${Partition}:inspector2:${Region}:${Account}:owner/${OwnerId}/
filter/${FilterId}" \
--tags CostAllocation=dev \
--region us-west-2
```

Hinzufügen von Tags zu Unterdrückungsregeln während der Erstellung

Verwenden Sie die `CreateFilter` API, um einer Unterdrückungsregel während der Erstellung Tags hinzuzufügen.

```
aws inspector2 create-filter \
--name "ExampleSuppressionRuleECR" \
--action SUPPRESS \
--filter-criteria 'resourceType=[{comparison="EQUALS", value="AWS_ECR_IMAGE"}]' \
--tags Owner=ApplicationSecurity \
--region us-west-2
```

Hinzufügen von Tags zu einer CIS-Scankonfiguration

Verwenden Sie die `CreateCisScanConfiguration` API, um einer CIS-Scankonfiguration ein Tag hinzuzufügen.

```
aws inspector2 create-cis-scan-configuration \
--scan-name "CreateConfigWithTagsSample" \
--security-level LEVEL_2 \
--targets accountIds=SELF,targetResourceTags={InspectorCisScan=True} \
--schedule 'daily={startTime={timeOfDay=11:10,timezone=UTC}}' \
```

```
--tags Owner=SecurityEngineering \
--region us-west-2
```

Entfernen von Tags

Sie können Tags aus den Ressourcen von Amazon Inspector entfernen. Zu diesen Ressourcen gehören Unterdrückungsregeln und CIS-Scankonfigurationen. Mithilfe von Tags können Sie AWS Ressourcen anhand bestimmter Kriterien kategorisieren. In diesem Thema wird beschrieben, wie Sie Tags aus den Amazon Inspector Inspector-Ressourcen entfernen.

Tags aus Amazon Inspector Inspector-Ressourcen entfernen

Sie können Tags aus Unterdrückungsregeln und CIS-Scankonfigurationen entfernen. Die folgenden Verfahren beschreiben, wie Sie Tags in der Konsole und mit der Amazon Inspector Inspector-API entfernen.

Entfernen von Tags in der Konsole

Sie können Tags aus den Amazon Inspector Inspector-Ressourcen in der Konsole entfernen.

Tags aus den Unterdrückungsregeln entfernen

Sie können ein Tag aus einer Unterdrückungsregel entfernen, indem Sie die Unterdrückungsregel so bearbeiten, dass das Tag nicht mehr enthalten ist. Weitere Informationen finden Sie unter Eine Unterdrückungsregel bearbeiten.

Tags aus einer CIS-Scankonfiguration entfernen

Sie können ein Tag aus einer CIS-Scankonfiguration entfernen, indem Sie die CIS-Scankonfiguration so bearbeiten, dass das Tag nicht mehr enthalten ist. Weitere Informationen finden Sie unter Eine CIS-Scankonfiguration bearbeiten.

Entfernen von Tags mit der Amazon Inspector API

Mit der Amazon Inspector-API können Sie ein Tag aus einer Amazon Inspector Inspector-Ressource entfernen.

Tags aus Amazon Inspector Inspector-Ressourcen entfernen

Verwenden Sie die `UntagResource` API, um Tags aus den Amazon Inspector Inspector-Ressourcen zu entfernen.

Das folgende Snippet zeigt ein Beispiel dafür, wie Sie Tags mithilfe von einer Amazon Inspector Inspector-Ressource entfernen können. `UntagResource` Sie müssen den ARN der Ressource und den Schlüssel für das Tag in den Befehl aufnehmen. Im folgenden Beispiel wird ein leerer Ressourcen-ARN für einen Unterdrückungsfilter verwendet. Der Schlüssel lautet `CostAllocation`. Informationen zu Ressourcentypen für Amazon Inspector finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Inspector2 in der Service Authorization Reference.

```
aws inspector2 untag-resource \
--resource-arn "arn:${Partition}:inspector2:${Region}:${Account}:owner/${OwnerId}/cis-configuration/${CISScanConfigurationId}" \
--tag-keys CostAllocation \
--region us-west-2
```

Überwachung von Nutzung und Kosten in Amazon Inspector

Sie können die Amazon Inspector Inspector-Konsole und die API verwenden, um die monatlichen Amazon Inspector Inspector-Kosten für Ihre Umgebung zu prognostizieren. Wenn Sie der Amazon Inspector-Administrator für eine Umgebung mit mehreren Konten sind, können Sie die Gesamtkosten für Ihre Umgebung und die Kostenkennzahlen für alle Mitgliedskonten einsehen. In diesem Abschnitt wird beschrieben, wie Sie auf Nutzungsstatistiken zugreifen und die Nutzungskosten berechnen.

Verwenden der Nutzungskonsole

Sie können die Nutzung und die voraussichtlichen Kosten für Amazon Inspector von der Konsole aus beurteilen.

Um auf Nutzungsstatistiken zuzugreifen

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie mithilfe der AWS-Region Auswahl in der oberen rechten Ecke der Seite die Region aus, in der Sie die Kosten überwachen möchten.
3. Wählen Sie im Navigationsbereich Benutzer.

Auf der Registerkarte „Nach Konto“ finden Sie die voraussichtlichen Gesamtkosten auf der Grundlage des Zeitraums von 30 Tagen, der unter Kontonutzung aufgeführt ist. Wählen Sie in der Tabelle unter der Spalte Voraussichtliche Kosten einen Wert aus, um eine Aufschlüsselung der Nutzung nach Scantyp für dieses Konto anzuzeigen. In diesem Detailbereich können Sie auch sehen, für welche Scantypen eine kostenlose Testversion für dieses Konto aktiv ist.

Wenn Sie der delegierte Administrator für eine Organisation sind, wird in der Tabelle für jedes Konto in Ihrer Organisation eine Zeile angezeigt. Wenn die Zuordnung zu einem Konto in Ihrer Organisation aufgehoben wird, zeigt die Konsole die voraussichtlichen Kosten als - an.

Auf der Registerkarte Nach Scan-Typ finden Sie eine Aufschlüsselung der tatsächlichen Nutzung im aktuellen Zeitraum von 30 Tagen nach Scantyp. Diese Informationen werden zur Berechnung der voraussichtlichen Kosten auf der Registerkarte „Nach Konto“ verwendet.

Wenn Sie der delegierte Administrator für eine Organisation sind, können Sie die Nutzung für jedes Konto in Ihrer Organisation einsehen.

Auf dieser Registerkarte können Sie jeden der folgenden Bereiche für Nutzungsstatistiken erweitern:

EC2 Amazon-Scannen

Die Amazon Inspector Inspector-Nutzungskonsole verfolgt die folgenden Metriken für agentenbasiertes Scannen und agentenloses Scannen:

- Instances (Durchschnitt) — Amazon Inspector berechnet anhand der Servicezeiten die durchschnittliche Anzahl der Ressourcen für das Scannen von EC2 Instanzen. Der Durchschnitt ergibt sich aus der Gesamtzahl der abgedeckten Stunden geteilt durch 720 Stunden (die Anzahl der Stunden in einem Zeitraum von 30 Tagen).
- Servicezeiten — Für EC2 Amazon-Scans ist dies die Gesamtzahl der Stunden innerhalb der letzten 30 Tage, in denen Amazon Amazon Inspector für jede EC2 Instanz in einem Konto eine aktive Abdeckung bereitgestellt hat. Bei EC2 Instances sind die Stunden zwischen dem Zeitpunkt, an dem Amazon Inspector die Instance entdeckt hat, bis sie beendet oder gestoppt oder von Tag-Scans ausgeschlossen wird. (Wenn Sie eine gestoppte Instance neu starten oder ein Ausschluss-Tag entfernen, nimmt Amazon Inspector den Versicherungsschutz wieder auf und es fallen weiterhin Versicherungsstunden für diese Instance an).

CIS-Instance-Scans — Die Gesamtzahl der CIS-Scans, die für Instances im Konto durchgeführt wurden.

Amazon ECR-Scannen

Erste Scans — Die Summe der ersten Scans von Bildern im Konto innerhalb der letzten 30 Tage.

Rescans — Die Summe der Rescans von Bildern im Konto innerhalb der letzten 30 Tage. Ein erneuter Scan ist jeder Scan, der an einem ECR-Bild durchgeführt wird, das Amazon Inspector zuvor gescannt hat. Wenn Sie Ihr ECR-Repository für kontinuierliches Scannen konfiguriert haben, werden erneute Scans automatisch durchgeführt, wenn Amazon Inspector seiner Datenbank neue Common Vulnerabilities and Exposures (CVE) hinzufügt.

Lambda-Scannen

Die Amazon Inspector Inspector-Nutzungskonsole verfolgt die folgenden Metriken für Lambda-Standardscans und Lambda-Code-Scans:

- Anzahl der Lambda-Funktionen (Durchschnitt) — Amazon Inspector berechnet anhand der Empfangsstunden die durchschnittliche Anzahl von Funktionen für das Scannen von Lambda-Funktionen. Der Durchschnitt ist die Gesamtzahl der abgedeckten Stunden geteilt durch 720 Stunden (die Anzahl der Stunden in einem Zeitraum von 30 Tagen).

- Servicezeiten — Für Lambda-Funktionsscans ist dies die Gesamtzahl der Stunden innerhalb der letzten 30 Tage, in denen Amazon Amazon Inspector für jede Lambda-Funktion in einem Konto eine aktive Abdeckung bereitgestellt hat. Für AWS Lambda Funktionen werden die Empfangszeiten von dem Zeitpunkt, an dem Amazon Inspector eine Funktion entdeckt, bis zu dem Zeitpunkt berechnet, zu dem sie gelöscht oder von Scans ausgeschlossen wird. Wenn eine ausgeschlossene Funktion erneut aufgenommen wird, fallen die für diese Funktion verfügbaren Stunden weiterhin an.

Verstehen, wie Amazon Inspector die Nutzungskosten berechnet

Bei den von Amazon Inspector angegebenen Kosten handelt es sich um Schätzungen, nicht um tatsächliche Kosten. Sie können daher von denen auf Ihrer AWS Billing Konsole abweichen.

Beachten Sie auf der Seite „Nutzung“ Folgendes zur Berechnung der Kosten durch Amazon Inspector:

- Die Nutzungskosten beziehen sich nur auf die aktuelle Region. Die Preise pro Scantyp variieren je nach AWS Region. Die genauen Preise pro Region finden Sie unter Preise für Amazon Inspector
- Alle Nutzungsprognosen werden auf den nächsten US-Dollar gerundet.
- Rabatte sind nicht in den voraussichtlichen Kosten enthalten.
- Die voraussichtlichen Kosten stellen die Gesamtkosten für den 30-tägigen Nutzungszeitraum pro Scantyp dar. Wenn ein Konto weniger als 30 Tage genutzt wurde, berechnet Amazon Inspector die Kosten nach 30 Tagen so, als ob alle derzeit abgedeckten Ressourcen für den Rest des 30-tägigen Zeitraums abgedeckt bleiben würden.
- Die Kosten pro Scan-Typ werden auf der Grundlage der folgenden Faktoren berechnet:
  - EC2 Scannen: Die Kosten geben die durchschnittliche Anzahl der EC2 Instances an, die Amazon Inspector in den letzten 30 Tagen bearbeitet hat.
  - Scannen von ECR-Containern: Die Kosten entsprechen der Summe der ersten Bildscans und der erneuten Bildscans in den letzten 30 Tagen.
  - Lambda-Standardscan: Die Kosten spiegeln die durchschnittliche Anzahl der Lambda-Funktionen wider, die Amazon Inspector in den letzten 30 Tagen abgedeckt hat.
  - Lambda-Code-Scanning: Die Kosten spiegeln die durchschnittliche Anzahl der Lambda-Funktionen wider, die Amazon Inspector in den letzten 30 Tagen abgedeckt hat.

Über die kostenlose Testversion von Amazon Inspector

In Amazon Inspector hat jeder Scan-Typ eine freie Spur. Wenn Sie einen Scan-Typ aktivieren, melden Sie sich automatisch für eine kostenlose 15-Tage-Testversion für diesen Scantyp an. Sobald die kostenlose Testversion gestartet wird, läuft sie automatisch nach 15 Tagen ab, auch wenn Sie den Scan-Typ deaktivieren.

> **Note**
>
> Die kostenlose Testversion gilt nicht für das CIS-Scannen.

Sicherheit in Amazon Inspector

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von Rechenzentren und Netzwerkarchitekturen, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame AWS Verantwortung von Ihnen und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud selbst und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der AWS Dienste in der ausgeführt AWS Cloud werden. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der AWS . Weitere Informationen zu den Compliance-Programmen, die für Amazon Inspector gelten, finden Sie unter AWS Services im Bereich nach Compliance-Programm AWS .
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Service, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von Amazon Inspector anwenden können. In den folgenden Themen erfahren Sie, wie Sie Amazon Inspector konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, mit denen Sie Ihre Amazon Inspector Inspector-Ressourcen überwachen und sichern können.

Themen

Datenschutz in Amazon Inspector

Das Modell der AWS gemeinsamen Verantwortung gilt für den Datenschutz in Amazon Inspector. Wie in diesem Modell beschrieben, AWS ist verantwortlich für den Schutz der globalen Infrastruktur, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz. Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag AWS -Modell der geteilten Verantwortung und in der DSGVO im AWS -Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Wird verwendet SSL/TLS , um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail. Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter Arbeiten mit CloudTrail Pfaden im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter Federal Information Processing Standard (FIPS) 140-3.

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit Amazon Inspector oder anderen AWS-Services über die Konsole AWS CLI, API oder arbeiten AWS SDKs. Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie eine URL für einen

externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Themen

Verschlüsselung im Ruhezustand

Standardmäßig speichert Amazon Inspector Daten im Ruhezustand mithilfe von AWS Verschlüsselungslösungen. Amazon Inspector verschlüsselt Daten wie die folgenden:

- Ressourcenbestand, gesammelt mit AWS Systems Manager.
- Aus Amazon Elastic Container Registry-Images analysiertes Ressourceninventar
- Generierte Sicherheitsergebnisse unter Verwendung AWS eigener Verschlüsselungsschlüssel von AWS Key Management Service

Sie können AWS eigene Schlüssel nicht verwalten, verwenden oder anzeigen. Sie müssen jedoch keine Maßnahmen ergreifen oder Programme ändern, um Schlüssel zu schützen, die Ihre Daten verschlüsseln. Weitere Informationen finden Sie unter Eigene AWS Schlüssel.

Wenn Sie Amazon Inspector deaktivieren, werden alle Ressourcen, die es für Sie speichert oder verwaltet, dauerhaft gelöscht, z. B. gesammeltes Inventar und Sicherheitserkenntnisse.

Verschlüsselung im Ruhezustand für den Code in Ihren Ergebnissen

Beim Scannen von Amazon Inspector Lambda-Code arbeitet Amazon Inspector mit Amazon Inspector zusammen, CodeGuru um Ihren Code auf Sicherheitslücken zu scannen. Wenn eine Sicherheitslücke erkannt wird, wird ein Codeausschnitt, der die Sicherheitslücke enthält, CodeGuru extrahiert und gespeichert, bis Amazon Inspector Zugriff anfordert. Standardmäßig wird ein AWS eigener Schlüssel CodeGuru verwendet, um den extrahierten Code zu verschlüsseln. Sie können Amazon Inspector jedoch so konfigurieren, dass Ihr eigener, vom Kunden verwalteter AWS KMS Schlüssel für die Verschlüsselung verwendet wird.

Der folgende Arbeitsablauf erklärt, wie Amazon Inspector den von Ihnen konfigurierten Schlüssel verwendet, um Ihren Code zu verschlüsseln:

1. Sie stellen Amazon Inspector mithilfe der Amazon Inspector UpdateEncryptionKeyInspector-API einen AWS KMS Schlüssel zur Verfügung.
2. Amazon Inspector leitet die Informationen zu Ihrem AWS KMS Schlüssel weiter an CodeGuru. CodeGuru speichert die Informationen für die future Verwendung.
3. CodeGuru fordert ein Zuschussformular AWS KMS für den Schlüssel an, den Sie in Amazon Inspector konfiguriert haben.
4. CodeGuru erstellt aus Ihrem Schlüssel einen verschlüsselten AWS KMS Datenschlüssel und speichert ihn. Dieser Datenschlüssel wird verwendet, um Ihre von CodeGuru gespeicherten Codedaten zu verschlüsseln.
5. Immer wenn Amazon Inspector Daten aus Codescans anfordert, CodeGuru verwendet Amazon Inspector den Grant, um den verschlüsselten Datenschlüssel zu entschlüsseln, und verwendet diesen Schlüssel dann, um die Daten zu entschlüsseln, sodass sie abgerufen werden können.

Wenn Sie das Lambda-Code-Scannen deaktivieren, wird CodeGuru der Grant zurückgezogen und der zugehörige Datenschlüssel gelöscht.

Berechtigungen für die Codeverschlüsselung mit einem vom Kunden verwalteten Schlüssel

Um Verschlüsselung verwenden zu können, benötigen Sie eine Richtlinie, die den Zugriff auf AWS KMS Aktionen ermöglicht, sowie eine Erklärung, die Amazon Inspector die CodeGuru Erlaubnis erteilt, diese Aktionen über Bedingungsschlüssel zu verwenden.

Wenn Sie den Verschlüsselungsschlüssel für Ihr Konto einrichten, aktualisieren oder zurücksetzen, müssen Sie eine Amazon Inspector-Administratorrichtlinie verwenden, z. B. AWS verwaltete Richtlinie: AmazonInspector2FullAccess Außerdem müssen Sie Benutzern mit Lesezugriff, die Codefragmente aus Ergebnissen oder Daten über den für die Verschlüsselung ausgewählten Schlüssel abrufen müssen, die folgenden Berechtigungen gewähren.

Für KMS muss die Richtlinie es Ihnen ermöglichen, die folgenden Aktionen auszuführen:

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

- `kms:RetireGrant`

Sobald Sie überprüft haben, dass Sie in Ihrer Richtlinie über die richtigen AWS KMS Berechtigungen verfügen, müssen Sie eine Erklärung beifügen, die Amazon Inspector und CodeGuru die Verwendung Ihres Schlüssels für die Verschlüsselung gestattet. Fügen Sie die folgende Grundsatzerklärung bei:

> **Note**
>
> Ersetzen Sie Region durch die AWS Region, in der Sie das Amazon Inspector Lambda-Code-Scannen aktiviert haben.

```
{
            "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**
>
> Wenn Sie die Anweisung hinzufügen, stellen Sie sicher, dass die Syntax gültig ist. Richtlinien verwenden das JSON-Format. Das bedeutet, dass Sie vor oder nach der Anweisung ein Komma hinzufügen müssen, je nachdem, wo Sie die Anweisung zur Richtlinie hinzufügen. Wenn Sie die Anweisung als letzte Anweisung hinzufügen, fügen Sie hinter der schließenden Klammer für die vorhergehende Anweisung ein Komma hinzu. Wenn Sie sie als erste Anweisung oder zwischen zwei vorhandenen Anweisungen hinzufügen, fügen Sie hinter der schließenden Klammer für die Anweisung ein Komma ein.

Konfiguration der Verschlüsselung mit einem vom Kunden verwalteten Schlüssel

Um die Verschlüsselung für Ihr Konto mit einem vom Kunden verwalteten Schlüssel zu konfigurieren, müssen Sie ein Amazon Inspector-Administrator mit den unter beschriebenen Berechtigungen seinBerechtigungen für die Codeverschlüsselung mit einem vom Kunden verwalteten Schlüssel. Darüber hinaus benötigen Sie einen AWS KMS Schlüssel in derselben AWS Region wie Ihre Ergebnisse oder einen Schlüssel für mehrere Regionen. Sie können einen vorhandenen symmetrischen Schlüssel in Ihrem Konto verwenden oder einen symmetrischen, vom Kunden verwalteten Schlüssel mithilfe der AWS Management-Konsole oder der erstellen. AWS KMS APIs Weitere Informationen finden Sie im Benutzerhandbuch unter Erstellen symmetrischer AWS KMSAWS KMS Verschlüsselungsschlüssel.

> **Note**
>
> Ab dem 13. Juni 2025 ändert sich der Dienstprinzipal bei AWS KMS Anfragen, die CloudTrail während des encryption/decryption Codeausschnitts angemeldet wurden, von „codeguru-reviewer“ zu „q“.

Verwenden der Amazon Inspector API zur Konfiguration der Verschlüsselung

Um einen Schlüssel für die Verschlüsselung für den UpdateEncryptionKeyBetrieb der Amazon Inspector-API festzulegen, während Sie als Amazon Inspector-Administrator angemeldet sind. Verwenden Sie in der API-Anfrage das `kmsKeyId` Feld, um den ARN des AWS KMS Schlüssels anzugeben, den Sie verwenden möchten. Für `scanType` Enter `CODE` und für `resourceType` Enter`AWS_LAMBDA_FUNCTION`.

Sie können die UpdateEncryptionKeyAPI verwenden, um zu überprüfen, welchen AWS KMS Schlüssel Amazon Inspector für die Verschlüsselung verwendet.

> **Note**
>
> Wenn Sie versuchen zu verwenden, `GetEncryptionKey` obwohl Sie keinen vom Kunden verwalteten Schlüssel eingerichtet haben, gibt der Vorgang einen `ResourceNotFoundException` Fehler zurück, was bedeutet, dass ein AWS eigener Schlüssel für die Verschlüsselung verwendet wird.

Wenn Sie den Schlüssel löschen oder seine Richtlinie ändern, sodass der Zugriff auf Amazon Inspector verweigert wird, können CodeGuru Sie andernfalls nicht auf Ihre gefundenen Sicherheitslücken zugreifen und der Lambda-Code-Scan schlägt für Ihr Konto fehl.

Sie können `ResetEncryptionKey` damit fortfahren, einen AWS eigenen Schlüssel zum Verschlüsseln von Code zu verwenden, der im Rahmen Ihrer Amazon Inspector Inspector-Ergebnisse extrahiert wurde.

Verschlüsselung während der Übertragung

AWS verschlüsselt alle Daten, die zwischen AWS internen Systemen und anderen AWS Diensten übertragen werden. AWS Systems Manager sammelt Telemetriedaten von kundeneigenen EC2

Instanzen, an die es AWS über einen mit Transport Layer Security (TLS) geschützten Kanal sendet, um sie zu bewerten. Die Ergebnisse der Amazon ECR- und AWS Lambda-Funktionsscans, die an Security Hub gesendet werden, werden über einen TLS-geschützten Kanal verschlüsselt. Weitere Informationen finden Sie unter Datenschutz in Systems Manager, um zu erfahren, wie SSM Daten bei der Übertragung verschlüsselt.

Identity and Access Management für Amazon Inspector

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Amazon Inspector Inspector-Ressourcen zu verwenden. IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie in Amazon Inspector ausführen.

Servicebenutzer — Wenn Sie den Amazon Inspector-Service für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die Anmeldeinformationen und Berechtigungen zur Verfügung, die Sie benötigen. Da Sie für Ihre Arbeit mehr Funktionen von Amazon Inspector verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Funktionsweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern

müssen. Wenn Sie in Amazon Inspector nicht auf eine Funktion zugreifen können, finden Sie weitere Informationen unterFehlerbehebung bei Identität und Zugriff auf Amazon Inspector.

Service-Administrator — Wenn Sie in Ihrem Unternehmen für die Amazon Inspector-Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf Amazon Inspector. Es ist Ihre Aufgabe, zu bestimmen, auf welche Funktionen und Ressourcen von Amazon Inspector Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anforderungen an Ihren IAM-Administrator senden, um die Berechtigungen der Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM mit Amazon Inspector verwenden kann, finden Sie unterSo arbeitet Amazon Inspector mit IAM.

IAM-Administrator — Wenn Sie ein IAM-Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff auf Amazon Inspector zu verwalten. Beispiele für identitätsbasierte Amazon Inspector Inspector-Richtlinien, die Sie in IAM verwenden können, finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode für die Selbstsignierung

von Anforderungen finden Sie unter AWS Signature Version 4 für API-Anforderungen im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen bereitstellen. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-Faktor-Authentifizierung im AWS IAM Identity Center - Benutzerhandbuch und AWS Multi-Faktor-Authentifizierung (MFA) in IAM im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen. Verwenden Sie diese nur, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, übernehmen sie Rollen, und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter Was ist IAM Identity Center? im AWS IAM Identity Center - Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein IAM-Benutzer ist eine Identität innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern im IAM-Benutzerhandbuch.

Eine IAM-Gruppe ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise eine Gruppe benennen IAMAdminsund dieser Gruppe Berechtigungen zur Verwaltung von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter Anwendungsfälle für IAM-Benutzer im IAM-Benutzerhandbuch.

IAM-Rollen

Eine IAM-Rolle ist eine Identität innerhalb von Ihnen AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, jedoch nicht mit einer bestimmten Person verknüpft. Um vorübergehend eine IAM-Rolle in der zu übernehmen AWS Management Console, können Sie von einer Benutzer- zu einer IAM-Rolle (Konsole) wechseln. Sie können eine Rolle übernehmen, indem Sie eine AWS CLI oder AWS API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter Methoden für die Übernahme einer Rolle im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- Verbundbenutzerzugriff – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter Erstellen von Rollen für externe

Identitätsanbieter (Verbund) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter Berechtigungssätze im AWS IAM Identity Center -Benutzerhandbuch.

- Temporäre IAM-Benutzerberechtigungen – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- Kontoübergreifender Zugriff – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.
- Serviceübergreifender Zugriff — Einige AWS-Services verwenden Funktionen in anderen AWS-Services. Wenn Sie beispielsweise in einem Service einen Anruf tätigen, ist es üblich, dass dieser Service Anwendungen in Amazon ausführt EC2 oder Objekte in Amazon S3 speichert. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
  - Forward Access Sessions (FAS) — Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.
  - Servicerolle – Eine Servicerolle ist eine IAM-Rolle, die ein Service übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.
  - Dienstbezogene Rolle — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und

gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

- Auf Amazon ausgeführte Anwendungen EC2 — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instance vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Die Berechtigungen in den Richtlinien legen fest, ob eine Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter Übersicht über JSON-Richtlinien im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Rolleninformationen von der AWS Management Console AWS CLI, der oder der AWS API abrufen.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem System zuordnen können AWS-Konto. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer Inline-Richtlinie wählen, finden Sie unter Auswählen zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffskontrolllisten () ACLs

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3 und Amazon VPC sind Beispiele für Dienste, die Unterstützung ACLs bieten. AWS WAF Weitere Informationen finden Sie unter Übersicht über ACLs die Zugriffskontrollliste (ACL) im Amazon Simple Storage Service Developer Guide.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- Berechtigungsgrenzen – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im IAM-Benutzerhandbuch.
- Dienststeuerungsrichtlinien (SCPs) — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie alle Funktionen in einer Organisation aktivieren, können Sie Richtlinien zur Servicesteuerung (SCPs) auf einige oder alle Ihre Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Root-Benutzer des AWS-Kontos Entitäten. Weitere Informationen zu Organizations und SCPs finden Sie unter Richtlinien zur Servicesteuerung im AWS Organizations Benutzerhandbuch.
- Ressourcenkontrollrichtlinien (RCPs) — RCPs sind JSON-Richtlinien, mit denen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten festlegen können, ohne die IAM-Richtlinien aktualisieren zu müssen, die jeder Ressource zugeordnet sind, deren Eigentümer Sie sind. Das RCP schränkt die Berechtigungen für Ressourcen in Mitgliedskonten ein und kann sich auf die effektiven Berechtigungen für Identitäten auswirken, einschließlich der Root-Benutzer des AWS-Kontos, unabhängig davon, ob sie zu Ihrer Organisation gehören. Weitere Informationen zu Organizations RCPs, einschließlich einer Liste AWS-Services dieser Support-Leistungen RCPs, finden Sie unter Resource Control Policies (RCPs) im AWS Organizations Benutzerhandbuch.

- Sitzungsrichtlinien – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter Bewertungslogik für Richtlinien.

So arbeitet Amazon Inspector mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf Amazon Inspector zu verwalten, sollten Sie sich darüber informieren, welche IAM-Funktionen mit Amazon Inspector verwendet werden können.

IAM-Funktionen, die Sie mit Amazon Inspector verwenden können

IAM-Feature	Unterstützung für Amazon Inspector
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Nein
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (services pezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Teilweise
Temporäre Anmeldeinformationen	Ja

IAM-Feature	Unterstützung für Amazon Inspector
Prinzipalberechtigungen	Ja
Servicerollen	Nein
Serviceverknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie Amazon Inspector und andere AWS-Services mit den meisten IAM-Funktionen funktionieren AWS-Services , finden Sie im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien für Amazon Inspector

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet ist. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der IAM-Referenz für JSON-Richtlinienelemente im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

Beispiele für identitätsbasierte Richtlinien von Amazon Inspector finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

Ressourcenbasierte Richtlinien in Amazon Inspector

Unterstützt ressourcenbasierte Richtlinien: Nein

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und

Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie einen Prinzipal angeben. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource unterscheiden AWS-Konten, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich. Weitere Informationen finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Politische Maßnahmen für Amazon Inspector

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der Amazon Inspector-Aktionen finden Sie unter Von Amazon Inspector definierte Aktionen in der Service Authorization Reference.

Richtlinienaktionen in Amazon Inspector verwenden das folgende Präfix vor der Aktion:

```
inspector2
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [
      "inspector2:action1",
      "inspector2:action2"
         ]
```

Beispiele für identitätsbasierte Richtlinien von Amazon Inspector finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

Richtlinienressourcen für Amazon Inspector

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein – `Resource`oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen Amazon-Ressourcennamen (ARN) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

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

Eine Liste der Amazon Inspector-Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter Von Amazon Inspector definierte Ressourcen in der Service Authorization Reference. Informationen darüber, mit welchen Aktionen Sie den ARN jeder Ressource angeben können, finden Sie unter Von Amazon Inspector definierte Aktionen.

Beispiele für identitätsbasierte Richtlinien von Amazon Inspector finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

Schlüssel zu den Richtlinienbedingungen für Amazon Inspector

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition` block) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. Sie können bedingte Ausdrücke erstellen, die Bedingungsoperatoren verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen AND-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, AWS wertet die Bedingung mithilfe einer logischen OR Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter IAM-Richtlinienelemente: Variablen und Tags im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Eine Liste der Amazon Inspector-Bedingungsschlüssel finden Sie unter Bedingungsschlüssel für Amazon Inspector in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter Von Amazon Inspector definierte Aktionen.

Beispiele für identitätsbasierte Richtlinien von Amazon Inspector finden Sie unter. Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

ACLs bei Amazon Inspector

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit Amazon Inspector

Unterstützt ABAC (Tags in Richtlinien): Teilweise

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen definiert werden. In AWS werden diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und an viele AWS Ressourcen anhängen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter Definieren von Berechtigungen mit ABAC-Autorisierung im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe Attributbasierte Zugriffskontrolle (ABAC) verwenden im IAM-Benutzerhandbuch.

Temporäre Anmeldeinformationen mit Amazon Inspector verwenden

Unterstützt temporäre Anmeldeinformationen: Ja

Einige funktionieren AWS-Services nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich Informationen, die mit temporären

Anmeldeinformationen AWS-Services funktionieren AWS-Services , finden Sie im IAM-Benutzerhandbuch unter Diese Option funktioniert mit IAM.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen AWS Management Console Methode als einem Benutzernamen und einem Passwort anmelden. Wenn Sie beispielsweise AWS über den Single Sign-On-Link (SSO) Ihres Unternehmens darauf zugreifen, werden bei diesem Vorgang automatisch temporäre Anmeldeinformationen erstellt. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter Wechseln von einer Benutzerrolle zu einer IAM-Rolle (Konsole) im IAM-Benutzerhandbuch.

Mithilfe der AWS API AWS CLI oder können Sie temporäre Anmeldeinformationen manuell erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff verwenden AWS. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen in IAM.

Serviceübergreifende Hauptberechtigungen für Amazon Inspector

Unterstützt Forward Access Sessions (FAS): Ja

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, kombiniert mit der Anforderung, Anfragen an nachgelagerte Dienste AWS-Service zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für Amazon Inspector

Unterstützt Servicerollen: Nein

Eine Servicerolle ist eine IAM-Rolle, die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.

> **Warning**
>
> Das Ändern der Berechtigungen für eine Servicerolle kann die Funktionalität von Amazon Inspector beeinträchtigen. Bearbeiten Sie Servicerollen nur, wenn Amazon Inspector Sie dazu anleitet.

Servicebezogene Rollen für Amazon Inspector

Unterstützt serviceverknüpfte Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Einzelheiten zum Erstellen oder Verwalten von dienstbezogenen Rollen finden Sie unter AWS-Services Diese Rollen funktionieren mit IAM. Suchen Sie in der Tabelle nach einem Service mit einem `Yes` in der Spalte Service-linked role (Serviceverknüpfte Rolle). Wählen Sie den Link Yes (Ja) aus, um die Dokumentation für die serviceverknüpfte Rolle für diesen Service anzuzeigen.

Beispiele für identitätsbasierte Richtlinien für Amazon Inspector

Standardmäßig sind Benutzer und Rollen nicht berechtigt, Amazon Inspector Inspector-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mithilfe der AWS Management Console, AWS Command Line Interface (AWS CLI) oder AWS API ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter Erstellen von IAM-Richtlinien (Konsole) im IAM-Benutzerhandbuch.

Einzelheiten zu den von Amazon Inspector definierten Aktionen und Ressourcentypen, einschließlich des Formats ARNs für die einzelnen Ressourcentypen, finden Sie unter Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Inspector in der Service Authorization Reference.

Themen

- Bewährte Methoden für Richtlinien
- Verwenden der Amazon Inspector Inspector-Konsole
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer
- Nur-Lese-Zugriff auf alle Amazon Inspector Inspector-Ressourcen zulassen
- Vollzugriff auf alle Amazon Inspector Inspector-Ressourcen zulassen

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand Amazon Inspector Inspector-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder diese löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um Ihren Benutzern und Workloads zunächst Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter AWS -verwaltete Richtlinien oder AWS -verwaltete Richtlinien für Auftrags-Funktionen im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.

- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung mit IAM Access Analyzer im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter Sicherer API-Zugriff mit MFA im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter Bewährte Methoden für die Sicherheit in IAM im IAM-Benutzerhandbuch.

Verwenden der Amazon Inspector Inspector-Konsole

Um auf die Amazon Inspector Inspector-Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den Amazon Inspector Inspector-Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen die Amazon Inspector-Konsole weiterhin verwenden können, fügen Sie den Entitäten auch den Amazon Inspector `ConsoleAccess` oder die `ReadOnly` AWS verwaltete Richtlinie hinzu. Weitere Informationen finden Sie unter Hinzufügen von Berechtigungen zu einem Benutzer im IAM-Benutzerhandbuch.

Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer

Benutzeridentität angefügt sind. Diese Richtlinie beinhaltet Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder 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": "*"
        }
    ]
}
```

Nur-Lese-Zugriff auf alle Amazon Inspector Inspector-Ressourcen zulassen

Dieses Beispiel zeigt eine Richtlinie, die nur Lesezugriff auf alle Amazon Inspector Inspector-Ressourcen ermöglicht.

```
{
    "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": "*"
        }
    ]
}
```

Vollzugriff auf alle Amazon Inspector Inspector-Ressourcen zulassen

Dieses Beispiel zeigt eine Richtlinie, die vollen Zugriff auf alle Amazon Inspector Inspector-Ressourcen ermöglicht.

```
{
    "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 verwaltete Richtlinien für Amazon Inspector

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet wird AWS. AWS Verwaltete Richtlinien dienen dazu, Berechtigungen für viele gängige Anwendungsfälle bereitzustellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

AWS verwaltete Richtlinie: AmazonInspector2FullAccess_v2

Sie können die `AmazonInspector2FullAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt vollen Zugriff auf Amazon Inspector und Zugriff auf andere verwandte Dienste.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `inspector2`— Ermöglicht den vollständigen Zugriff auf Amazon Inspector APIs.
- `codeguru-security`— Ermöglicht Administratoren das Abrufen von Sicherheitsergebnissen und Konfigurationseinstellungen für ein Konto.
- `iam`— Ermöglicht Amazon Inspector, die serviceverknüpften Rollen zu erstellen `AWSServiceRoleForAmazonInspector2` und`AWSServiceRoleForAmazonInspector2Agentless`. `AWSServiceRoleForAmazonInspector2`ist erforderlich, damit Amazon Inspector Vorgänge wie das Abrufen von Informationen über EC2 Amazon-Instances, Amazon ECR-Repositorys und Amazon ECR-Container-Images ausführen kann. Es ist auch erforderlich, um mit Schlüsseln verschlüsselte Amazon EBS-Snapshots zu entschlüsseln. AWS KMS Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für Amazon Inspector.
- `organizations`— `AllowServicePrincipalBasedAccessToOrganizationApis` ermöglicht es nur Service Principals, dienstbezogene Rollen für eine Organisation zu erstellen AWS-Konten, sie AWS-Konto als delegierten Administrator für eine Organisation zu registrieren und delegierte Administratoren in einer Organisation aufzulisten. `AllowOrganizationalBasedAccessToOrganizationApis`ermöglicht es dem Versicherungsnehmer, Informationen über eine Organisationseinheit abzurufen, insbesondere Informationen auf Ressourcenebene ARNs. `AllowAccountsBasedAccessToOrganizationApis`ermöglicht dem Versicherungsnehmer das Abrufen von Informationen, insbesondere auf Ressourcenebene ARNs, über eine. AWS-Konto`AllowAccessToOrganizationApis`ermöglicht es dem Versicherungsnehmer, AWS-Services integrierte Informationen zu einer Organisation und Organisation einzusehen.

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "AllowFullAccessToInspectorApis",
      "Effect" : "Allow",
      "Action" : "inspector2:*",
      "Resource" : "*"
    },
    {
      "Sid" : "AllowAccessToCodeGuruApis",
      "Effect" : "Allow",
      "Action" : [
        "codeguru-security:BatchGetFindings",
        "codeguru-security:GetAccountConfiguration"
      ],
      "Resource" : "*"
    },
    {
      "Sid" : "AllowAccessToCreateSlr",
      "Effect" : "Allow",
      "Action" : "iam:CreateServiceLinkedRole",
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "iam:AWSServiceName" : [
            "agentless.inspector2.amazonaws.com",
            "inspector2.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid" : "AllowServicePrincipalBasedAccessToOrganizationApis",
      "Effect" : "Allow",
      "Action" : [
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
        "organizations:ListDelegatedAdministrators"
      ],
      "Resource" : "*",
      "Condition": {
        "StringEquals": {
```

```
          "organizations:ServicePrincipal": [
            "inspector2.amazonaws.com",
            "agentless.inspector2.amazonaws.com"
          ]
        }
      }
    },
    {
      "Sid" : "AllowOrganizationalBasedAccessToOrganizationApis",
      "Effect" : "Allow",
      "Action" : [
        "organizations:DescribeOrganizationalUnit"
      ],
      "Resource" : "arn:*:organizations::*:ou/o-*/ou-*"
    },
    {
      "Sid" : "AllowAccountsBasedAccessToOrganizationApis",
      "Effect" : "Allow",
      "Action" : [
        "organizations:DescribeAccount"
      ],
      "Resource" : "arn:*:organizations::*:account/o-*/*"
    },
    {
      "Sid" : "AllowAccessToOrganizationApis",
      "Effect" : "Allow",
      "Action" : [
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganization"
      ],
      "Resource" : "*"
    }
  ]
}
```

AWS verwaltete Richtlinie: AmazonInspector2FullAccess

Sie können die `AmazonInspector2FullAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt Administratorberechtigungen, die vollen Zugriff auf Amazon Inspector ermöglichen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `inspector2`— Ermöglicht vollen Zugriff auf die Funktionen von Amazon Inspector.
- `iam`— Ermöglicht Amazon Inspector, die serviceverknüpften Rollen zu erstellen `AWSServiceRoleForAmazonInspector2` und`AWSServiceRoleForAmazonInspector2Agentless`. `AWSServiceRoleForAmazonInspector2`ist erforderlich, damit Amazon Inspector Vorgänge wie das Abrufen von Informationen über Ihre EC2 Amazon-Instances, Amazon ECR-Repositorys und Container-Images ausführen kann. Es ist auch erforderlich, dass Amazon Inspector Ihr VPC-Netzwerk analysiert und Konten beschreibt, die mit Ihrer Organisation verknüpft sind. `AWSServiceRoleForAmazonInspector2Agentless`ist erforderlich, damit Amazon Inspector Vorgänge wie das Abrufen von Informationen über Ihre EC2 Amazon-Instances und Amazon EBS-Snapshots ausführen kann. Es ist auch erforderlich, Amazon EBS-Snapshots zu entschlüsseln, die mit Schlüsseln verschlüsselt sind. AWS KMS Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für Amazon Inspector.
- `organizations`— Ermöglicht Administratoren die Verwendung von Amazon Inspector für eine Organisation in AWS Organizations. Wenn Sie den vertrauenswürdigen Zugriff für Amazon Inspector in aktivieren AWS Organizations, können Mitglieder des delegierten Administratorkontos Einstellungen verwalten und Ergebnisse in ihrer gesamten Organisation einsehen.
- `codeguru-security`— Ermöglicht Administratoren, Amazon Inspector zu verwenden, um Informationscodefragmente abzurufen und die Verschlüsselungseinstellungen für Code zu ändern, den CodeGuru Security speichert. Weitere Informationen finden Sie unter Verschlüsselung im Ruhezustand für den Code in Ihren Ergebnissen.

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

```
  {
   "Sid": "AllowAccessToCodeGuruApis",
   "Effect": "Allow",
   "Action": [
    "codeguru-security:BatchGetFindings",
    "codeguru-security:GetAccountConfiguration"
   ],
   "Resource": "*"
  },
  {
   "Sid": "AllowAccessToCreateSlr",
   "Effect": "Allow",
   "Action": "iam:CreateServiceLinkedRole",
   "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": [
      "agentless.inspector2.amazonaws.com",
      "inspector2.amazonaws.com"
     ]
    }
   }
  },
  {
   "Sid": "AllowAccessToOrganizationApis",
   "Effect": "Allow",
   "Action": [
    "organizations:EnableAWSServiceAccess",
    "organizations:RegisterDelegatedAdministrator",
    "organizations:ListDelegatedAdministrators",
    "organizations:ListAWSServiceAccessForOrganization",
    "organizations:DescribeOrganizationalUnit",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization"
   ],
   "Resource": "*"
  }
 ]
}
```

AWS verwaltete Richtlinie: AmazonInspector2ReadOnlyAccess

Sie können die `AmazonInspector2ReadOnlyAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt Berechtigungen, die nur Lesezugriff auf Amazon Inspector ermöglichen.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `inspector2`— Ermöglicht den schreibgeschützten Zugriff auf die Funktionen von Amazon Inspector.
- `organizations`— Ermöglicht das Einsehen von Details zum Versicherungsschutz durch Amazon Inspector für ein Unternehmen. AWS Organizations
- `codeguru-security`— Ermöglicht das Abrufen von Codefragmenten aus CodeGuru der Sicherheitsabteilung. Ermöglicht auch die Anzeige der Verschlüsselungseinstellungen für Ihren in CodeGuru Security gespeicherten Code.

```
{
 "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 verwaltete Richtlinie: AmazonInspector2ManagedCisPolicy

Sie können die `AmazonInspector2ManagedCisPolicy`-Richtlinie auch Ihren IAM-Entitäten anfügen. Diese Richtlinie sollte einer Rolle zugeordnet werden, die Ihren EC2 Amazon-Instances die Erlaubnis erteilt, CIS-Scans der Instance durchzuführen. Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2 Instance ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der EC2 Instanz vorzuziehen. Um einer EC2 Instanz eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instanzprofil, das an die Instanz angehängt ist. Ein Instanzprofil enthält die Rolle und ermöglicht Programmen, die auf der EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM-Rolle, um Berechtigungen für Anwendungen zu gewähren, die auf EC2 Amazon-Instances ausgeführt werden.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `inspector2`— Ermöglicht den Zugriff auf Aktionen, die zur Ausführung von CIS-Scans verwendet werden.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
            "inspector2:StartCisSession",
            "inspector2:StopCisSession",
            "inspector2:SendCisSessionTelemetry",
            "inspector2:SendCisSessionHealth"
            ],
            "Resource": "*",
        }
     ]
 }
```

AWS verwaltete Richtlinie: AmazonInspector2ServiceRolePolicy

Sie können die `AmazonInspector2ServiceRolePolicy`-Richtlinie Ihren IAM-Entitäten nicht anfügen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, die es Amazon Inspector ermöglicht, Aktionen in Ihrem Namen durchzuführen. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für Amazon Inspector.

AWS verwaltete Richtlinie: AmazonInspector2AgentlessServiceRolePolicy

Sie können die `AmazonInspector2AgentlessServiceRolePolicy`-Richtlinie Ihren IAM-Entitäten nicht anfügen. Diese Richtlinie ist mit einer dienstbezogenen Rolle verknüpft, die es Amazon Inspector ermöglicht, Aktionen in Ihrem Namen durchzuführen. Weitere Informationen finden Sie unter Verwenden von serviceverknüpften Rollen für Amazon Inspector.

Amazon Inspector aktualisiert AWS verwaltete Richtlinien

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für Amazon Inspector an, seit dieser Service begonnen hat, diese Änderungen zu verfolgen. Um automatische Benachrichtigungen über Änderungen an dieser Seite zu erhalten, abonnieren Sie den RSS-Feed auf der Amazon Inspector Document History-Seite.

Änderung	Beschreibung	Datum
AmazonInspector2 FullAccess_v2 — Neue Richtlinie	Amazon Inspector hat eine neue verwaltete Richtlinie hinzugefügt, die vollen Zugriff auf Amazon Inspector und Zugriff auf andere verwandte Dienste bietet.	03. Juli 2025
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat eine neue Berechtigung hinzugefügt, die es Amazon Inspector ermöglicht, IP-Adressen und Internet-Gateways zu beschreiben.	29. April 2025

Änderung	Beschreibung	Datum
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die nur Lesezugriff auf Amazon ECS- und Amazon EKS-Aktionen ermöglichen.	25. März 2025
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, Funktions-Tags zurückzugeben AWS Lambda.	31. Juli 2024
AmazonInspector2 FullAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, die serviceverknüpfte Rolle zu erstellen . `AWSServiceRoleForAmazonInspector2Agentless` Dadurch können Benutzer agentenbasiertes Scannen und agentenloses Scannen durchführen, wenn sie Amazon Inspector aktivieren.	24. April 2024
AmazonInspector2 ManagedCisPolicy — Neue Richtlinie	Amazon Inspector hat eine neue verwaltete Richtlinie hinzugefügt, die Sie als Teil eines Instance-Profils verwenden können, um CIS-Scans auf einer Instance zuzulassen.	23. Januar 2024

Änderung	Beschreibung	Datum
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, CIS-Scans auf Ziel-Instances zu starten.	23. Januar 2024
AmazonInspector2 AgentlessServiceRolePolicy — Neue Richtlinie	Amazon Inspector hat eine neue servicebezogene Rollenrichtlinie hinzugefügt, um das Scannen von EC2 Instances ohne Agenten zu ermöglichen.	8. November 2023
AmazonInspector2 ReadOnlyAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Benutzern mit Lesezugriff ermöglichen, Informationen zu Sicherheitslücken für gefundene Sicherheitslücken in Paketen abzurufen.	22. September 2023
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, Netzwerkkonfigurationen von EC2 Amazon-Instances zu scannen, die Teil der Elastic Load Balancing Balancing-Zielgruppen sind.	31. August 2023

Änderung	Beschreibung	Datum
AmazonInspector2 ReadOnlyAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Benutzern mit Lesezugriff ermöglichen, Software Bill of Materials (SBOM) für ihre Ressourcen zu exportieren.	29. Juni 2023
AmazonInspector2 ReadOnlyAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Benutzern mit Lesezugriff ermöglichen, Details der Verschlüsselungseinstellungen für Lambda-Code-Scanergebnisse für ihr Konto abzurufen.	13. Juni 2023
AmazonInspector2 FullAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, mit denen Benutzer einen vom Kunden verwalteten KMS-Schlüssel konfigurieren können, um Code in Ergebnissen aus Lambda-Code-Scans zu verschlüsseln.	13. Juni 2023
AmazonInspector2 ReadOnlyAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Benutzern mit Lesezugriff ermöglichen, Details zum Status und zu den Ergebnissen des Lambda-Code-Scans für ihr Konto abzurufen.	02. Mai 2023

Änderung	Beschreibung	Datum
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, AWS CloudTrail serviceverknüpfte Kanäle in Ihrem Konto zu erstellen, wenn Sie Lambda-Scanning aktivieren. Auf diese Weise kann Amazon Inspector CloudTrail Ereignisse in Ihrem Konto überwachen.	30. April 2023
AmazonInspector2 FullAccess — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Benutzern ermöglichen, Details zu den beim Lambda-Code-Scannen gefundenen Sicherheitslücken abzurufen.	21. April 2023
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, Informationen über die benutzerdefinierten Pfade, die ein Kunde für Amazon EC2 Deep Inspection definiert hat, an Amazon EC2 Systems Manager zu senden.	17. April 2023

Änderung	Beschreibung	Datum
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, AWS CloudTrail serviceverknüpfte Kanäle in Ihrem Konto zu erstellen, wenn Sie Lambda-Scanning aktivieren. Auf diese Weise kann Amazon Inspector CloudTrail Ereignisse in Ihrem Konto überwachen.	30. April 2023
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, Scans des Entwicklercodes in AWS Lambda Funktionen anzufordern und Scandaten von Amazon CodeGuru Security zu empfangen. Darüber hinaus hat Amazon Inspector Berechtigungen zur Überprüfung von IAM-Richtlinien hinzugefügt. Amazon Inspector verwendet diese Informationen, um Lambda-Funktionen auf Code-Schwachstellen zu überprüfen.	28. Februar 2023

Änderung	Beschreibung	Datum
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat eine neue Anweisung hinzugefügt, die es Amazon Inspector ermöglicht, Informationen CloudWatch darüber abzurufen, wann eine AWS Lambda Funktion zuletzt aufgerufen wurde. Amazon Inspector verwendet diese Informationen, um Scans auf die Lambda-Funktionen in Ihrer Umgebung zu konzentrieren, die in den letzten 90 Tagen aktiv waren.	20. Februar 2023
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat eine neue Anweisung hinzugefügt, die es Amazon Inspector ermöglicht, Informationen über AWS Lambda Funktionen abzurufen, einschließlich jeder Layer-Version, die jeder Funktion zugeordnet ist. Amazon Inspector verwendet diese Informationen, um Lambda-Funktionen auf Sicherheitslücken zu überprüfen.	28. November 2022

Änderung	Beschreibung	Datum
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat eine neue Aktion hinzugefügt, mit der Amazon Inspector die Ausführung von SSM-Verknüpfungen beschreiben kann. Darüber hinaus hat Amazon Inspector zusätzlichen Ressourcenbereich hinzugefügt, damit Amazon Inspector SSM-Verknüpfungen mit `AmazonInspector2` eigenen SSM-Dokumenten erstellen, aktualisieren, löschen und starten kann.	31. August 2022
AmazonInspector2 ServiceRolePolicy Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat den Ressourcenbereich der Richtlinie aktualisiert, sodass Amazon Inspector Softwareinventar in anderen AWS Partitionen erfassen kann.	12. August 2022
AmazonInspector2 ServiceRolePolicy — Aktualisierungen einer bestehenden Richtlinie	Amazon Inspector hat den Ressourcenbereich der Aktionen neu strukturiert, sodass Amazon Inspector SSM-Verknüpfungen erstellen, löschen und aktualisieren kann.	10. August 2022

Änderung	Beschreibung	Datum
AmazonInspector2 — Neue Richtlinie ReadOnlyAccess	Amazon Inspector hat eine neue Richtlinie hinzugefügt, um den schreibgeschützten Zugriff auf die Funktionen von Amazon Inspector zu ermöglichen.	21. Januar 2022
AmazonInspector2 FullAccess — Neue Richtlinie	Amazon Inspector hat eine neue Richtlinie hinzugefügt, um vollen Zugriff auf die Funktionen von Amazon Inspector zu ermöglichen.	29. November 2021
AmazonInspector2 ServiceRolePolicy — Neue Richtlinie	Amazon Inspector hat eine neue Richtlinie hinzugefügt, die es Amazon Inspector ermöglicht, in Ihrem Namen Aktionen in anderen Diensten durchzuführen.	29. November 2021
Amazon Inspector hat begonnen, Änderungen nachzuverfolgen	Amazon Inspector hat damit begonnen, Änderungen an seinen AWS verwalteten Richtlinien nachzuverfolgen.	29. November 2021

Verwenden von serviceverknüpften Rollen für Amazon Inspector

Amazon Inspector verwendet eine AWS Identity and Access Management (IAM) -Serviceverknüpfte Rolle mit dem Namen. `AWSServiceRoleForAmazonInspector2` Bei dieser serviceverknüpften Rolle handelt es sich um eine IAM-Rolle, die direkt mit Amazon Inspector verknüpft ist. Es ist von Amazon Inspector vordefiniert und beinhaltet alle Berechtigungen, die Amazon Inspector benötigt, um andere in AWS-Services Ihrem Namen anzurufen.

Eine serviceverknüpfte Rolle erleichtert die Einrichtung von Amazon Inspector, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon Inspector definiert die

Berechtigungen seiner serviceverknüpften Rolle. Sofern nicht anders definiert, kann nur Amazon Inspector die Rolle übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Sie müssen Berechtigungen konfigurieren, damit eine IAM-Entität (z. B. eine Gruppe oder Rolle) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter serviceverknüpfte Rollenberechtigung im IAM-Benutzerhandbuch. Sie können eine dienstverknüpfte Rolle erst löschen, nachdem Sie die zugehörigen Ressourcen gelöscht haben. Dadurch werden Ihre Amazon Inspector Inspector-Ressourcen geschützt, da Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entziehen können.

Informationen zu anderen Diensten, die serviceverknüpfte Rollen unterstützen, finden Sie unter AWS Dienste, die mit IAM funktionieren. Suchen Sie in der Spalte Serviceverknüpfte Rollen nach den Diensten, für die Ja steht. Wählen Sie Ja mit einem Link aus, um die Dokumentation zu serviceverknüpften Rollen für diesen Dienst zu lesen.

Servicebezogene Rollenberechtigungen für Amazon Inspector

Amazon Inspector verwendet die verwaltete Richtlinie mit dem Namen `AWSServiceRoleForAmazonInspector2`. Diese dienstbezogene Rolle vertraut darauf, dass der `inspector2.amazonaws.com` Service die Rolle übernimmt.

Die Berechtigungsrichtlinie für die Rolle, die benannt ist `AmazonInspector2ServiceRolePolicy`, ermöglicht es Amazon Inspector, Aufgaben wie die folgenden auszuführen:

- Verwenden Sie Amazon Elastic Compute Cloud (Amazon EC2) -Aktionen, um Informationen über Ihre Instances und Netzwerkpfade abzurufen.
- Verwenden Sie AWS Systems Manager Aktionen, um Inventar von Ihren EC2 Amazon-Instances abzurufen und Informationen über Pakete von Drittanbietern aus benutzerdefinierten Pfaden abzurufen.
- Verwenden Sie die AWS Systems Manager `SendCommand` Aktion, um CIS-Scans für Ziel-Instances aufzurufen.
- Verwenden Sie Amazon Elastic Container Registry-Aktionen, um Informationen über Ihre Container-Images abzurufen.
- Verwenden Sie AWS Lambda Aktionen, um Informationen über Ihre Lambda-Funktionen abzurufen.
- Verwenden Sie AWS Organizations Aktionen, um zugehörige Konten zu beschreiben.

- Verwenden Sie CloudWatch Aktionen, um Informationen darüber abzurufen, wann Ihre Lambda-Funktionen zuletzt aufgerufen wurden.
- Verwenden Sie ausgewählte IAM-Aktionen, um Informationen über Ihre IAM-Richtlinien abzurufen, die zu Sicherheitslücken in Ihrem Lambda-Code führen könnten.
- Verwenden Sie CodeGuru Sicherheitsaktionen, um den Code in Ihren Lambda-Funktionen zu scannen. Amazon Inspector verwendet die folgenden CodeGuru Sicherheitsaktionen:
  - codeguru-security: CreateScan — Erteilt die Erlaubnis, einen Sicherheitsscan zu erstellen CodeGuru .
  - codeguru-security: GetScan — Erteilt die Erlaubnis, Metadaten des Sicherheitsscans abzurufen. CodeGuru
  - codeguru-security: ListFindings — Erteilt die Erlaubnis, von Security generierte Ergebnisse abzurufen. CodeGuru
  - codeguru-security: DeleteScansByCategory — Erteilt der Sicherheitsabteilung die Erlaubnis, von Amazon CodeGuru Inspector initiierte Scans zu löschen.
  - codeguru-security: BatchGetFindings — Erteilt die Erlaubnis, eine Reihe von spezifischen Ergebnissen abzurufen, die von Security generiert wurden. CodeGuru
- Verwenden Sie ausgewählte Elastic Load Balancing Balancing-Aktionen, um Netzwerkscans von EC2 Instances durchzuführen, die Teil der Elastic Load Balancing Balancing-Zielgruppen sind.
- Verwenden Sie Amazon ECS- und Amazon EKS-Aktionen, um nur Lesezugriff zu gewähren, um Cluster und Aufgaben anzuzeigen und Aufgaben zu beschreiben.

Die Rolle ist mit der folgenden Berechtigungsrichtlinie konfiguriert.

```
    {
"Version": "2012-10-17",
"Statement": [
 {
  "Sid": "TirosPolicy",
  "Effect": "Allow",
  "Action": [
   "directconnect:DescribeConnections",
   "directconnect:DescribeDirectConnectGatewayAssociations",
   "directconnect:DescribeDirectConnectGatewayAttachments",
   "directconnect:DescribeDirectConnectGateways",
   "directconnect:DescribeVirtualGateways",
   "directconnect:DescribeVirtualInterfaces",
```

```
    "ec2:DescribeAddresses",
    "ec2:DescribeAvailabilityZones",
    "ec2:DescribeCustomerGateways",
    "ec2:DescribeEgressOnlyInternetGateways",
    "ec2:DescribeInstances",
    "ec2:DescribeInternetGateways",
    "ec2:DescribeManagedPrefixLists",
    "ec2:DescribeNatGateways",
    "ec2:DescribeNetworkAcls",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribePrefixLists",
    "ec2:DescribeRegions",
    "ec2:DescribeRouteTables",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeTransitGatewayAttachments",
    "ec2:DescribeTransitGatewayConnects",
    "ec2:DescribeTransitGatewayPeeringAttachments",
    "ec2:DescribeTransitGatewayRouteTables",
    "ec2:DescribeTransitGatewayVpcAttachments",
    "ec2:DescribeTransitGateways",
    "ec2:DescribeVpcEndpointServiceConfigurations",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcPeeringConnections",
    "ec2:DescribeVpcs",
    "ec2:DescribeVpnConnections",
    "ec2:DescribeVpnGateways",
    "ec2:GetManagedPrefixListEntries",
    "ec2:GetTransitGatewayRouteTablePropagations",
    "ec2:SearchTransitGatewayRoutes",
    "elasticloadbalancing:DescribeListeners",
    "elasticloadbalancing:DescribeLoadBalancerAttributes",
    "elasticloadbalancing:DescribeLoadBalancers",
    "elasticloadbalancing:DescribeRules",
    "elasticloadbalancing:DescribeTags",
    "elasticloadbalancing:DescribeTargetGroups",
    "elasticloadbalancing:DescribeTargetGroupAttributes",
    "elasticloadbalancing:DescribeTargetHealth",
    "network-firewall:DescribeFirewall",
    "network-firewall:DescribeFirewallPolicy",
    "network-firewall:DescribeResourcePolicy",
    "network-firewall:DescribeRuleGroup",
    "network-firewall:ListFirewallPolicies",
    "network-firewall:ListFirewalls",
```

```
    "network-firewall:ListRuleGroups",
    "tiros:CreateQuery",
    "tiros:GetQueryAnswer"
   ],
   "Resource": [
    "*"
   ]
  },
  {
   "Sid": "PackageVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "ecr:BatchGetImage",
    "ecr:BatchGetRepositoryScanningConfiguration",
    "ecr:DescribeImages",
    "ecr:DescribeRegistry",
    "ecr:DescribeRepositories",
    "ecr:GetAuthorizationToken",
    "ecr:GetDownloadUrlForLayer",
    "ecr:GetRegistryScanningConfiguration",
    "ecr:ListImages",
    "ecr:PutRegistryScanningConfiguration",
    "organizations:DescribeAccount",
    "organizations:DescribeOrganization",
    "organizations:ListAccounts",
    "ssm:DescribeAssociation",
    "ssm:DescribeAssociationExecutions",
    "ssm:DescribeInstanceInformation",
    "ssm:ListAssociations",
    "ssm:ListResourceDataSync"
   ],
   "Resource": "*"
  },
  {
   "Sid": "LambdaPackageVulnerabilityScanning",
   "Effect": "Allow",
   "Action": [
    "lambda:ListFunctions",
    "lambda:GetFunction",
    "lambda:GetLayerVersion",
    "lambda:ListTags",
    "cloudwatch:GetMetricData"
   ],
   "Resource": "*"
```

```
  },
  {
   "Sid": "GatherInventory",
   "Effect": "Allow",
   "Action": [
    "ssm:CreateAssociation",
    "ssm:StartAssociationsOnce",
    "ssm: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": "GatherInventoryDeleteAssociation",
   "Effect": "Allow",
   "Action": [
    "ssm:DeleteAssociation"
   ],
   "Resource": [
    "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"
     }
    }
   },
   {
    "Sid": "AllowListAccessToECSAndEKS",
    "Effect": "Allow",
    "Action": [
        "ecs:ListClusters",
        "ecs:ListTasks",
        "eks:ListClusters"
    ],
    "Resource": [
        "*"
    ],
    "Condition": {
        "StringEquals": {
            "aws:ResourceAccount": "${aws:PrincipalAccount}"
        }
    }
    },
    {
    "Sid": "AllowAccessToECSTasks",
    "Effect": "Allow",
    "Action": [
        "ecs:DescribeTasks"
    ],
    "Resource": "arn:aws:ecs:*:*:task/*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceAccount": "${aws:PrincipalAccount}"
     }
    }
   }
  ]
}
```

Eine serviceverknüpfte Rolle für Amazon Inspector erstellen

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie Amazon Inspector in der AWS Management Console, der oder der AWS API aktivieren AWS CLI, erstellt Amazon Inspector die serviceverknüpfte Rolle für Sie.

Bearbeiten einer serviceverknüpften Rolle für Amazon Inspector

Amazon Inspector erlaubt Ihnen nicht, die `AWSServiceRoleForAmazonInspector2` serviceverknüpfte Rolle zu bearbeiten. Nachdem eine serviceverknüpfte Rolle erstellt wurde, können Sie den Namen der Rolle nicht mehr ändern, da verschiedene Entitäten möglicherweise auf die Rolle verweisen. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für Amazon Inspector

Wenn Sie Amazon Inspector nicht mehr verwenden müssen, empfehlen wir Ihnen, die `AWSServiceRoleForAmazonInspector2` serviceverknüpfte Rolle zu löschen. Bevor Sie die Rolle löschen können, müssen Sie Amazon Inspector in allen Bereichen deaktivieren, in AWS-Region denen sie aktiviert ist. Wenn Sie Amazon Inspector deaktivieren, wird die Rolle nicht für Sie gelöscht. Wenn Sie Amazon Inspector erneut aktivieren, kann Amazon Inspector daher die bestehende Rolle verwenden. Auf diese Weise können Sie vermeiden, dass eine ungenutzte Entität nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch die Ressourcen für Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie manuell löschen können.

Wenn Sie diese serviceverknüpfte Rolle löschen und dann erneut erstellen müssen, können Sie die Rolle in Ihrem Konto mit demselben Verfahren neu anlegen. Wenn Sie Amazon Inspector aktivieren, erstellt Amazon Inspector die serviceverknüpfte Rolle für Sie neu.

> **Note**
>
> Wenn der Amazon Inspector-Service die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt das Löschen möglicherweise fehl. Warten Sie in diesem Fall einige Minuten und führen Sie den Vorgang dann erneut aus.

Sie können die IAM-Konsole, die oder die AWS API verwenden AWS CLI, um die `AWSServiceRoleForAmazonInspector2` serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Servicebezogene Rollenberechtigungen für agentenlose Amazon Inspector-Scans

Das agentenlose Scannen von Amazon Inspector verwendet die angegebene, mit dem Service verknüpfte Rolle. `AWSServiceRoleForAmazonInspector2Agentless` Mit dieser Spiegelreflexkamera kann Amazon Inspector einen Amazon EBS-Volume-Snapshot in Ihrem Konto erstellen und dann auf die Daten aus diesem Snapshot zugreifen. Diese dienstbezogene Rolle vertraut darauf, dass der `agentless.inspector2.amazonaws.com` Service die Rolle übernimmt.

> **Important**
>
> Die Anweisungen in dieser servicebezogenen Rolle verhindern, dass Amazon Inspector agentenlose Scans für alle EC2 Instances durchführt, die Sie mithilfe des `InspectorEc2Exclusion` Tags von Scans ausgeschlossen haben. Darüber hinaus verhindern die Anweisungen, dass Amazon Inspector auf verschlüsselte Daten von einem Volume zugreift, wenn der für die Verschlüsselung verwendete KMS-Schlüssel das `InspectorEc2Exclusion` Tag trägt. Weitere Informationen finden Sie unter Instanzen von Amazon Inspector-Scans ausschließen.

Die Berechtigungsrichtlinie für die Rolle, die benannt ist`AmazonInspector2AgentlessServiceRolePolicy`, ermöglicht es Amazon Inspector, Aufgaben wie die folgenden auszuführen:

- Verwenden Sie Amazon Elastic Compute Cloud (Amazon EC2) -Aktionen, um Informationen über Ihre EC2 Instances, Volumes und Snapshots abzurufen.
  - Verwenden Sie EC2 Amazon-Tagging-Aktionen, um Schnappschüsse für Scans mit dem `InspectorScan` Tag-Schlüssel zu kennzeichnen.
  - Verwenden Sie EC2 Amazon-Snapshot-Aktionen, um Snapshots zu erstellen, sie mit dem `InspectorScan` Tag-Schlüssel zu kennzeichnen und anschließend Snapshots von Amazon EBS-Volumes zu löschen, die mit dem `InspectorScan` Tag-Schlüssel gekennzeichnet wurden.
- Verwenden Sie Amazon EBS-Aktionen, um Informationen aus Snapshots abzurufen, die mit dem `InspectorScan` Tag-Schlüssel gekennzeichnet sind.
- Verwenden Sie ausgewählte AWS KMS Entschlüsselungsaktionen, um Snapshots zu entschlüsseln, die mit vom Kunden verwalteten Schlüsseln verschlüsselt wurden. AWS KMS Amazon Inspector entschlüsselt keine Snapshots, wenn der KMS-Schlüssel, mit dem sie verschlüsselt wurden, mit dem Tag gekennzeichnet ist. `InspectorEc2Exclusion`

Die Rolle ist mit der folgenden Berechtigungsrichtlinie konfiguriert.

```
{
 "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/*"
  }
 ]
}
```

Erstellung einer dienstbezogenen Rolle für agentenloses Scannen

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie Amazon Inspector in der AWS Management Console, der oder der AWS API aktivieren AWS CLI, erstellt Amazon Inspector die serviceverknüpfte Rolle für Sie.

Bearbeitung einer serviceverknüpften Rolle für agentenloses Scannen

Amazon Inspector erlaubt Ihnen nicht, die `AWSServiceRoleForAmazonInspector2Agentless` serviceverknüpfte Rolle zu bearbeiten. Nachdem eine serviceverknüpfte Rolle erstellt wurde, können Sie den Namen der Rolle nicht mehr ändern, da verschiedene Entitäten möglicherweise auf die Rolle verweisen. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie unter Bearbeiten einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für das Scannen ohne Agenten

Wenn Sie ein Feature oder einen Service, die bzw. der eine serviceverknüpfte Rolle erfordert, nicht mehr benötigen, sollten Sie diese Rolle löschen. Auf diese Weise haben Sie keine ungenutzte Entität, die nicht aktiv überwacht oder verwaltet wird.

> **Important**
>
> Um die `AWSServiceRoleForAmazonInspector2Agentless` Rolle zu löschen, müssen Sie Ihren Scanmodus in allen Regionen, in denen agentenloses Scannen verfügbar ist, auf agentenbasiert einstellen.

So löschen Sie die serviceverknüpfte Rolle mit IAM

Verwenden Sie die IAM-Konsole, die oder die AWS API, um die AWS CLI serviceverknüpfte AWSService RoleForAmazonInspector 2Agentless-Rolle zu löschen. Weitere Informationen finden Sie unter Löschen einer serviceverknüpften Rolle im IAM-Benutzerhandbuch.

Fehlerbehebung bei Identität und Zugriff auf Amazon Inspector

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit Amazon Inspector und IAM auftreten können.

Themen

- Ich bin nicht berechtigt, eine Aktion in Amazon Inspector durchzuführen
- Ich bin nicht berechtigt, iam auszuführen: PassRole
- Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Amazon Inspector Inspector-Ressourcen ermöglichen

Ich bin nicht berechtigt, eine Aktion in Amazon Inspector durchzuführen

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *`my-example-widget`*-Ressource anzuzeigen, jedoch nicht über `inspector2:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 inspector2:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `inspector2:GetWidget`-Aktion auf die *`my-example-widget`*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht berechtigt sind, die `iam:PassRole` Aktion durchzuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an Amazon Inspector übergeben können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Service zu übergeben, anstatt eine neue Servicerolle oder eine dienstbezogene Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in Amazon Inspector auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Amazon Inspector Inspector-Ressourcen ermöglichen

Sie können eine Rolle erstellen, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem

die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob Amazon Inspector diese Funktionen unterstützt, finden Sie unterSo arbeitet Amazon Inspector mit IAM.
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter Kontoübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Überwachung von Amazon Inspector

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Verfügbarkeit, Zuverlässigkeit und Leistung von Amazon Inspector und anderen AWS Lösungen. AWS bietet Tools zur Überwachung von Amazon Inspector, zur Meldung auftretender Probleme und zur Ergreifung von Maßnahmen zur Behebung dieser Probleme:

- Amazon EventBridge ist ein AWS Service, der Ereignisse verwendet, um Anwendungskomponenten miteinander zu verbinden, sodass Sie leichter skalierbare, ereignisgesteuerte Anwendungen erstellen können. EventBridge stellt einen Stream von Echtzeitdaten aus Ihren Anwendungen, Software-as-a-Service (SaaS) -Anwendungen sowie AWS Diensten und Routen bereit, sodass Sie Ereignisse überwachen können, die in Diensten auftreten, und ereignisgesteuerte Architekturen erstellen können.
- AWS CloudTrailist ein AWS Dienst, der API-Aufrufe und zugehörige Ereignisse erfasst, die von Ihnen oder in Ihrem Namen getätigt wurden. AWS-Konto CloudTrail übermittelt die Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket, sodass Sie feststellen können, welche

Benutzer und Konten angerufen wurden AWS, von welcher Quell-IP-Adresse aus Anrufe getätigt wurden und wann die Anrufe stattfanden.

Protokollieren Amazon Inspector Inspector-API-Aufrufen mit AWS CloudTrail

Amazon Inspector ist in einen Service integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem IAM-Benutzer oder einer IAM-Rolle oder einem AWS-Service in Amazon Inspector ausgeführt wurden. CloudTrail erfasst alle API-Aufrufe für Amazon Inspector als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der Amazon Inspector Inspector-Konsole und Aufrufe der Amazon Inspector Inspector-API-Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Übermittlung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für Amazon Inspector. Auch wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse in der CloudTrail-Konsole in Event history (Ereignisverlauf) anzeigen. Anhand der von CloudTrail gesammelten Informationen können Sie Folgendes bestimmen:

- Die Anfrage, die an Amazon Inspector gestellt wurde.
- Die IP-Adresse, von der die Anforderung erfolgt ist.
- Wer die Anfrage gestellt hat.
- Wann die Anfrage gestellt wurde.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

Informationen zu Amazon Inspector in CloudTrail

CloudTrail ist auf Ihrem aktiviert AWS-Konto , wenn Sie das Konto erstellen. Wenn in Amazon Inspector eine Aktivität auftritt, wird diese Aktivität zusammen mit anderen AWS-Service Ereignissen in der CloudTrail Ereignishistorie in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem anzeigen, suchen und herunterladen AWS-Konto. Weitere Informationen finden Sie unter Ereignisse mit CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS-Konto, einschließlich Ereignissen für Amazon Inspector, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole anlegen, gilt dieser für alle AWS-Regionen-Regionen. Der Trail protokolliert Ereignisse aus allen Regionen in der AWS -Partition und stellt die Protokolldateien in dem von Ihnen angegebenen Amazon-S3-Bucket

bereit. Darüber hinaus können Sie andere konfigurieren, AWS-Services um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie unter den folgenden Themen:

- Übersicht zum Erstellen eines Trails
- CloudTrail unterstützte Dienste und Integrationen
- Konfigurieren von Amazon SNS-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien von mehreren Konten
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen

Alle Amazon Inspector Inspector-Aktionen werden von protokolliert CloudTrail. Alle Aktionen, die Amazon Inspector ausführen kann, sind in der Amazon Inspector API-Referenz dokumentiert. Zum Beispiel werden durch Aufrufe der `CreateFindingsReport`-, `ListCoverage`- und `UpdateOrganizationConfiguration`-Aktionen Einträge in den CloudTrail -Protokolldateien generiert.

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Anmeldeinformationen des Stammbenutzers oder des IAM-Benutzers gestellt wurde.
- Ob die Anfrage mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen verbundenen Benutzer ausgeführt wurde.
- Ob die Anforderung aus einem anderen AWS-Service gesendet wurde.

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

Grundlegendes zu Amazon Inspector Inspector-Protokolldateieinträgen

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar. Zu den Ereignissen gehören Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Amazon Inspector Scaninformationen in CloudTrail

Amazon Inspector Scan ist in integriert CloudTrail. Alle Amazon Inspector Scan API-Operationen werden als Verwaltungsereignisse protokolliert. Eine Liste der Amazon Inspector Scan API-Operationen, bei denen Amazon Inspector protokolliert CloudTrail, finden Sie unter Amazon Inspector Scan in der Amazon Inspector API-Referenz.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die `ScanSbom` Aktion demonstriert:


```
{
    "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"
}
```


Konformitätsvalidierung für Amazon Inspector

Informationen darüber, ob AWS-Service ein AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter Umfang nach Compliance-Programm AWS-Services unter . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter AWS Compliance-Programme AWS .

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen unter .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen

und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- Compliance und Governance im Bereich Sicherheit – In diesen Anleitungen für die Lösungsimplementierung werden Überlegungen zur Architektur behandelt. Außerdem werden Schritte für die Bereitstellung von Sicherheits- und Compliance-Features beschrieben.
- Referenz für berechtigte HIPAA-Services – Listet berechtigte HIPAA-Services auf. Nicht alle AWS-Services sind HIPAA-fähig.
- AWS Compliance-Ressourcen — Diese Sammlung von Arbeitsmappen und Leitfäden gilt möglicherweise für Ihre Branche und Ihren Standort.
- AWS Leitfäden zur Einhaltung von Vorschriften für Kunden — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den Leitfäden werden die bewährten Verfahren zur Sicherung zusammengefasst AWS-Services und die Leitlinien den Sicherheitskontrollen in verschiedenen Frameworks (einschließlich des National Institute of Standards and Technology (NIST), des Payment Card Industry Security Standards Council (PCI) und der International Organization for Standardization (ISO)) zugeordnet.
- Evaluierung von Ressourcen anhand von Regeln im AWS Config Entwicklerhandbuch — Der AWS Config Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- AWS Security Hub— Auf diese AWS-Service Weise erhalten Sie einen umfassenden Überblick über Ihren internen Sicherheitsstatus. AWS Security Hub verwendet Sicherheitskontrollen, um Ihre AWS -Ressourcen zu bewerten und Ihre Einhaltung von Sicherheitsstandards und bewährten Methoden zu überprüfen. Die Liste der unterstützten Services und Kontrollen finden Sie in der Security-Hub-Steuerelementreferenz.
- Amazon GuardDuty — Dies AWS-Service erkennt potenzielle Bedrohungen für Ihre Workloads AWS-Konten, Container und Daten, indem es Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen wie PCI DSS zu erfüllen, indem es die in bestimmten Compliance-Frameworks vorgeschriebenen Anforderungen zur Erkennung von Eindringlingen erfüllt.
- AWS Audit Manager— Auf diese AWS-Service Weise können Sie Ihre AWS Nutzung kontinuierlich überprüfen, um das Risikomanagement und die Einhaltung von Vorschriften und Industriestandards zu vereinfachen.

Resilienz in Amazon Inspector

Die AWS globale Infrastruktur basiert auf Availability AWS-Regionen Zones. AWS-Regionen bieten mehrere, physisch getrennte und isolierte Availability Zones, die mit Netzwerken mit niedriger Latenz, hohem Durchsatz und hoher Redundanz verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Infrastruktursicherheit in Amazon Inspector

Als verwalteter Service ist Amazon Inspector durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe, um über das Netzwerk auf Amazon Inspector zuzugreifen. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Alternativ können Sie mit AWS Security Token Service (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

Reaktion auf Vorfälle in Amazon Inspector

Sicherheit hat bei höchste Priorität AWS. Wie im Modell der AWS gemeinsamen Verantwortung unter „Sicherheit der Cloud“ erwähnt, AWS ist er für den Schutz der Infrastruktur verantwortlich, auf der alle Dienste in der AWS Cloud ausgeführt werden. AWS ist auch für die Reaktion auf alle Vorfälle im Zusammenhang mit dem Amazon Inspector-Service verantwortlich.

Als AWS Kunde tragen Sie gemeinsam die Verantwortung für die Aufrechterhaltung der Sicherheit in der AWS Cloud. Das bedeutet, dass Sie die Sicherheit kontrollieren, die Sie implementieren möchten. Dazu gehören alle AWS Tools und Funktionen, auf die Sie zugreifen. Das bedeutet auch, dass Sie im Rahmen des Modells der gemeinsamen Verantwortung für die Reaktion auf Vorfälle verantwortlich sind.

Indem Sie eine Sicherheitsbasis einrichten, die alle Ziele für Ihre in der AWS Cloud ausgeführten Anwendungen erfüllt, können Sie Abweichungen erkennen, auf die Sie reagieren können. Da es sich bei der Reaktion auf Vorfälle um ein komplexes Thema handelt, sollten Sie sich die folgenden Ressourcen ansehen, um besser zu verstehen, welche Auswirkungen die Reaktion auf Vorfälle hat und wie sich Ihre Entscheidungen auf Ihre Unternehmensziele auswirken könnten: Leitfaden zur Reaktion auf AWSAWS Sicherheitsvorfälle, Best Practices im Bereich Sicherheit und AWS Cloud Adoption Framework: Security Perspective.

Greifen Sie über einen Schnittstellenendpunkt auf Amazon Inspector zu (AWS PrivateLink

Sie können AWS PrivateLink es verwenden, um eine private Verbindung zwischen Ihrer VPC und Amazon Inspector herzustellen. Sie können auf Amazon Inspector zugreifen, als wäre es in Ihrer VPC, ohne ein Internet-Gateway, ein NAT-Gerät, eine VPN-Verbindung oder AWS Direct Connect eine Verbindung zu verwenden. Instances in Ihrer VPC benötigen keine öffentlichen IP-Adressen, um auf Amazon Inspector zuzugreifen.

Sie stellen diese private Verbindung her, indem Sie einen Schnittstellen-Endpunkt erstellen, der von AWS PrivateLink unterstützt wird. Wir erstellen eine Endpunkt-Netzwerkschnittstelle in jedem Subnetz, das Sie für den Schnittstellen-Endpunkt aktivieren. Dabei handelt es sich um vom Anforderer verwaltete Netzwerkschnittstellen, die als Einstiegspunkt für den Datenverkehr dienen, der für Amazon Inspector bestimmt ist.

Weitere Informationen finden Sie AWS PrivateLink im Handbuch unter Access AWS-Services through.AWS PrivateLink

Überlegungen zu Amazon Inspector

Bevor Sie einen Schnittstellenendpunkt für Amazon Inspector einrichten, lesen Sie die Überlegungen im AWS PrivateLink Handbuch.

Amazon Inspector unterstützt Aufrufe all seiner API-Aktionen über den Schnittstellenendpunkt.

VPC-Endpunktrichtlinien werden für Amazon Inspector nicht unterstützt. Standardmäßig ist der vollständige Zugriff auf Amazon Inspector über den Schnittstellenendpunkt zulässig. Alternativ können Sie den Endpunkt-Netzwerkschnittstellen eine Sicherheitsgruppe zuordnen, um den Datenverkehr zu Amazon Inspector über den Schnittstellenendpunkt zu steuern.

Erstellen Sie einen Schnittstellenendpunkt für Amazon Inspector

Sie können einen Schnittstellenendpunkt für Amazon Inspector entweder mit der Amazon VPC-Konsole oder mit AWS Command Line Interface (AWS CLI) erstellen. Weitere Informationen finden Sie unter Erstellen eines Schnittstellenendpunkts im AWS PrivateLink -Leitfaden.

Wenn Sie einen Schnittstellenendpunkt für Amazon Inspector erstellen, verwenden Sie einen der folgenden Servicenamen:

```
com.amazonaws.region.inspector2
```

```
com.amazonaws.region.inspector-scan
```

*region*Ersetzen Sie es durch den entsprechenden AWS-Region Code AWS-Region.

Wenn Sie privates DNS für den Schnittstellenendpunkt aktivieren, können Sie API-Anfragen an Amazon Inspector richten, indem Sie beispielsweise dessen standardmäßigen regionalen DNS-Namen verwenden, `service-name.us-east-1.amazonaws.com` oder `service-name.us-east-1.api.aws.com` für die USA Ost (Nord-Virginia).

Amazon Inspector Inspector-Integrationen

Amazon Inspector lässt sich in andere AWS Dienste integrieren. Diese Dienste können Daten von Amazon Inspector aufnehmen, sodass Sie Ihre Ergebnisse auf unterschiedliche Weise einsehen können. Sehen Sie sich die folgenden Integrationsoptionen an, um mehr zu erfahren.

Integration von Amazon Inspector mit Amazon ECR

Amazon Elastic Container Registry (Amazon ECR) ist eine AWS verwaltete Container-Image-Registry, die private Registrierungen unterstützt. Private Registrys von Amazon ECR hosten Container-Images in einer hochverfügbaren und skalierbaren Architektur. Sie können Amazon Inspector verwenden, um Container-Images, die sich in Ihrem Amazon ECR-Repository befinden, nach anfälligen Betriebssystempaketen und Programmiersprachenpaketen zu durchsuchen. Weitere Informationen finden Sie unter Integration von Amazon Inspector mit Amazon Elastic Container Registry (Amazon ECR).

Amazon Inspector Inspector-Integration mit AWS Security Hub

AWS Security Hubbietet einen umfassenden Überblick über Ihren Sicherheitsstatus AWS und hilft Ihnen dabei, Ihre Umgebung anhand von Industriestandards und bewährten Methoden zu überprüfen. Security Hub sammelt Sicherheitsdaten von AWS Konten, Diensten und unterstützten Produkten. Sie können Security Hub verwenden, um die Ergebnisdaten von Amazon Inspector aufzunehmen und einen zentralen Speicherort für Ergebnisse in all Ihren integrierten AWS Services und AWS Partner Network-Produkten zu schaffen. Weitere Informationen finden Sie unter Amazon Inspector Inspector-Integration mit AWS Security Hub.

Integration von Amazon Inspector mit Amazon Elastic Container Registry (Amazon ECR)

Amazon Elastic Container Registry ist eine vollständig verwaltete Container-Registry, die Docker- und OCI-Images und AWS -Artefakte unterstützt. Wenn Sie Amazon ECR verwenden, können Sie Enhanced Scanning für Ihr Container-Register aktivieren. Wenn Sie das erweiterte Scannen aktivieren, erkennt Amazon Inspector Ihre Container-Images automatisch und scannt sie nach anfälligen Betriebssystemen und Programmiersprachenpaketen. Diese Integration ermöglicht es Ihnen, die Ergebnisse von Amazon Inspector für Container-Images einzusehen und die Häufigkeit

und den Umfang der Scans in der Amazon ECR-Konsole zu verwalten. Weitere Informationen finden Sie unter Amazon ECR-Container-Images mit Amazon Inspector scannen.

Aktivierung der Integration

Sie können die Integration aktivieren, indem Sie das Amazon Inspector-Scannen über die Amazon Inspector-Konsole oder API aktivieren oder indem Sie Ihr Repository so konfigurieren, dass es Enhanced Scanning mit Amazon Inspector über die Amazon ECR-Konsole oder API verwendet.

Weitere Informationen zur Aktivierung der Integration über Amazon Inspector finden Sie unterAutomatisierte Scantypen in Amazon Inspector.

Informationen zur Aktivierung und Konfiguration von Enhanced Scanning in Amazon ECR finden Sie unter Enhanced Scanning im Amazon ECR-Benutzerhandbuch.

Verwendung der Integration in einer Umgebung mit mehreren Konten

Wenn Sie Mitglied in einer Umgebung mit mehreren Konten sind, können Sie das erweiterte Scannen über Amazon ECR aktivieren. Nach der Aktivierung kann es jedoch nur von Ihrem delegierten Amazon Inspector-Administrator deaktiviert werden. Wenn es deaktiviert ist, kehrt es zum normalen Scannen zurück. Weitere Informationen finden Sie unter Amazon Inspector deaktivieren.

Amazon Inspector Inspector-Integration mit AWS Security Hub

Security Hub bietet einen umfassenden Überblick über Ihren Sicherheitsstatus in AWS. Auf diese Weise können Sie Ihre Umgebung anhand von Industriestandards und Best Practices im Bereich Sicherheit überprüfen. Security Hub sammelt Sicherheitsdaten von AWS Konten, Diensten und unterstützten Produkten. Sie können diese Informationen verwenden, um Sicherheitstrends zu analysieren und Sicherheitsprobleme zu identifizieren. Wenn Sie die Amazon Inspector-Integration mit Security Hub aktivieren, kann Amazon Inspector Ergebnisse an Security Hub senden, und Security Hub kann diese Ergebnisse im Rahmen Ihrer Sicherheitslage analysieren.

Security Hub verfolgt Sicherheitsprobleme als Ergebnisse. Einige Ergebnisse können auf Sicherheitsprobleme zurückzuführen sein, die in anderen AWS Diensten oder Produkten von Drittanbietern festgestellt wurden. Security Hub verwendet eine Reihe von Regeln, um Sicherheitsprobleme zu erkennen und Ergebnisse zu generieren, und stellt Tools bereit, mit denen Sie die Ergebnisse verwalten können. Security Hub archiviert die Ergebnisse von Amazon Inspector, sobald die Ergebnisse in Amazon Inspector geschlossen wurden. Sie können auch eine Historie Ihrer

Ergebnisse und Details zu den Ergebnissen einsehen sowie den Status einer Untersuchung zu einem Ergebnis verfolgen.

Security Hub verarbeitet Ergebnisse im AWS Security Finding Format (ASFF). Dieses Format enthält Details wie eindeutige Identifikatoren, Schweregrade, betroffene Ressourcen, Hinweise zur Problembehebung, Workflow-Status und Kontextinformationen.

> **Note**
>
> Von Amazon Inspector Code Security generierte Sicherheitsergebnisse sind für diese Integration nicht verfügbar. Sie können jedoch in der Amazon Inspector-Konsole und über die Amazon Inspector-API auf diese speziellen Ergebnisse zugreifen.

Themen

Ergebnisse von Amazon Inspector anzeigen in AWS Security Hub

Sie können die Ergebnisse von Amazon Inspector Classic und Amazon Inspector im Security Hub einsehen.

> **Note**
>
> Um nur nach Ergebnissen von Amazon Inspector `"aws/inspector/ProductVersion": "2"` zu filtern, fügen Sie es der Filterleiste hinzu. Dieser Filter schließt Amazon Inspector Classic-Ergebnisse aus dem Security Hub-Dashboard aus.

Beispiel für ein Ergebnis von Amazon Inspector

```
{
  "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"
}
```

Aktivierung und Konfiguration der Amazon Inspector Inspector-Integration mit Security Hub

Sie können die Amazon Inspector Inspector-Integration mit aktivieren, AWS Security Hub indem Sie Security Hub aktivieren. Nachdem Sie Security Hub aktiviert haben, AWS Security Hub wird die Amazon Inspector-Integration mit automatisch aktiviert und Amazon Inspector beginnt, alle Ergebnisse unter Verwendung des Security Finding Formats (ASFF) an AWS Security Hub zu senden.

Deaktivierung des Flusses von Ergebnissen aus einer Integration

Um zu verhindern, dass Amazon Inspector Ergebnisse an Security Hub sendet, können Sie die Security Hub Hub-Konsole oder API verwenden und AWS CLI...

Sicherheitskontrollen für Amazon Inspector im Security Hub anzeigen

Security Hub analysiert Ergebnisse von unterstützten Produkten AWS und Produkten von Drittanbietern und führt automatisierte und kontinuierliche Sicherheitsprüfungen anhand von Regeln durch, um eigene Ergebnisse zu generieren. Die Regeln werden durch Sicherheitskontrollen dargestellt, anhand derer Sie feststellen können, ob die Anforderungen eines Standards erfüllt werden.

Amazon Inspector verwendet Sicherheitskontrollen, um zu überprüfen, ob die Funktionen von Amazon Inspector aktiviert sind oder aktiviert werden sollten. Nachstehend sind einige dieser Features aufgeführt:

- EC2 Amazon-Scannen
- Amazon ECR-Scannen
- Lambda-Standardscannen
- Scannen von Lambda-Code

Weitere Informationen finden Sie unter Amazon Inspector-Steuerelemente im AWS Security Hub Benutzerhandbuch.

Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector

Amazon Inspector kann Softwareanwendungen scannen, die auf folgenden Geräten installiert sind:

- Amazon Elastic Compute Cloud (Amazon EC2) -Instanzen

  > **Note**
  >
  > Für EC2 Amazon-Instances kann Amazon Inspector nach Paketschwachstellen in Betriebssystemen suchen, die agentenbasiertes Scannen unterstützen. Amazon Inspector kann auch nach Paketschwachstellen in Betriebssystemen und Programmiersprachen suchen, die Hybrid-Scans unterstützen. Amazon Inspector sucht nicht nach Schwachstellen in der Toolchain. Die Version des Programmiersprachen-Compilers, der zur Erstellung der Anwendung verwendet wurde, führt zu diesen Sicherheitslücken.

- Container-Images, die in Amazon Elastic Container Registry (Amazon ECR) -Repositorys gespeichert sind

  > **Note**
  >
  > Bei ECR-Container-Images kann Amazon Inspector nach Sicherheitslücken in Betriebssystemen und Programmiersprachenpaketen suchen. Amazon Inspector sucht nicht nach Schwachstellen in Rust der Toolchain. Die Version des Programmiersprachen-Compilers, der zur Erstellung der Anwendung verwendet wurde, führt zu diesen Sicherheitslücken.

- AWS Lambda Funktionen

  > **Note**
  >
  > Für Lambda-Funktionen kann Amazon Inspector nach Sicherheitslücken in Programmiersprachenpaketen und Code-Schwachstellen suchen. Amazon Inspector sucht nicht nach Schwachstellen in der Toolchain. Die Version des Programmiersprachen-Compilers, der zur Erstellung der Anwendung verwendet wurde, führt zu diesen Sicherheitslücken.

Wenn Amazon Inspector Ressourcen scannt, bezieht Amazon Inspector mehr als 50 Datenfeeds, um Ergebnisse für häufig auftretende Sicherheitslücken und Risiken zu generieren (CVEs). Zu diesen Quellen gehören beispielsweise Sicherheitsratschläge von Anbietern, Datenfeeds und Feeds mit Bedrohungsinformationen sowie die National Vulnerability Database (NVD) und MITRE. Amazon Inspector aktualisiert Sicherheitslückendaten aus Quell-Feeds mindestens einmal täglich.

Damit Amazon Inspector eine Ressource scannen kann, muss auf der Ressource ein unterstütztes Betriebssystem oder eine unterstützte Programmiersprache ausgeführt werden. In den Themen in diesem Abschnitt sind die Betriebssysteme, Programmiersprachen und Laufzeiten aufgeführt, die Amazon Inspector für verschiedene Ressourcen und Scantypen unterstützt. Sie listen auch eingestellte Betriebssysteme auf.

> **Note**
>
> Amazon Inspector kann nur eingeschränkten Support für ein Betriebssystem anbieten, nachdem ein Anbieter den Support für das Betriebssystem eingestellt hat.

Themen

Unterstützte Betriebssysteme

In diesem Abschnitt sind die Betriebssysteme aufgeführt, die Amazon Inspector unterstützt.

Unterstützte Betriebssysteme: EC2 Amazon-Scanning

In der folgenden Tabelle sind die Betriebssysteme aufgeführt, die Amazon Inspector für das Scannen von EC2 Amazon-Instances unterstützt. Sie gibt die Sicherheitsempfehlungen des Anbieters für jedes Betriebssystem an und gibt an, welche Betriebssysteme agentenbasiertes Scannen und agentenloses Scannen unterstützen.

Wenn Sie die agentenbasierte Scanmethode verwenden, konfigurieren Sie den SSM-Agent so, dass er kontinuierliche Scans auf allen geeigneten Instanzen durchführt. Amazon Inspector empfiehlt,

eine Version des SSM-Agenten zu konfigurieren, die höher als 3.2.2086.0 ist. Weitere Informationen finden Sie unter Arbeiten mit dem SSM-Agenten im Amazon EC2 Systems Manager Manager-Benutzerhandbuch.

Erkennungen von Linux-Betriebssystemen werden nur für das Standard-Paket-Manager-Repository (rpm und dpkg) unterstützt und schließen keine Anwendungen von Drittanbietern, Repositorys mit erweitertem Support (RHEL EUS, E4S, AUS und TUS) und optionale Repositorys (Anwendungsstreams) ein. Amazon Inspector scannt den laufenden Kernel auf Sicherheitslücken. Bei einigen Betriebssystemen ist beispielsweise ein Neustart erforderlichUbuntu, damit Upgrades in aktiven Ergebnissen angezeigt werden.

Betriebssystem	Version	Sicherheitsempfehlungen von Anbietern	Unterstützung für agentenloses Scannen	Unterstützung für agentengestütztes Scannen
AlmaLinux	8	ALSA	Ja	Ja
AlmaLinux	9	ALSA	Ja	Ja
Amazon Linux (AL2)	AL2	LEIDER	Ja	Ja
Amazon Linux 2023 (AL2023)	AL2023	LEIDER	Ja	Ja
Bottlerocket	1.7.0 und später	GHSA, CVE	Nein	Ja
Debian-Server (Bullseye)	11	DSA	Ja	Ja
Debian-Server (Bücherwurm)	12	DSA	Ja	Ja
Fedora	41	CVE	Ja	Ja
Fedora	42	CVE	Ja	Ja
openSUSE Leap	15.6	CVE	Ja	Ja

Betriebssystem	Version	Sicherheitsempfehlungen von Anbietern	Unterstützung für agentenloses Scannen	Unterstützung für agentengestütztes Scannen
Oracle Linux (Oracle)	8	ELSA	Ja	Ja
Oracle Linux (Oracle)	9	ELSA	Ja	Ja
Red Hat Enterprise Linux (RHEL)	8	RHSA	Ja	Ja
Red Hat Enterprise Linux (RHEL)	9	RHSA	Ja	Ja
Rocky Linux	8	RLSA	Ja	Ja
Rocky Linux	9	RLSA	Ja	Ja
SUSE Linux Enterprise Server (SLES)	15,6	HÖHLE SUES	Ja	Ja
Ubuntu (Xenial)	16,04	USN, Ubuntu Pro (esm-infra und esm-apps)	Ja	Ja
Ubuntu (Bionisch)	18,04	USN, Ubuntu Pro (esm-infra und esm-apps)	Ja	Ja
Ubuntu (fokal)	20.04	USN, Ubuntu Pro (esm-infra und esm-apps)	Ja	Ja

Betriebssystem	Version	Sicherheitsempfehlungen von Anbietern	Unterstützung für agentenloses Scannen	Unterstützung für agentengestütztes Scannen
Ubuntu (Jammy)	22,04	USN, Ubuntu Pro (esm-infra und esm-apps)	Ja	Ja
Ubuntu (Noble Numbat)	24.04	USN, Ubuntu Pro (esm-infra & esm-apps)	Ja	Ja
Ubuntu (Oracular Oriole)	24.10	USN	Ja	Ja
Ubuntu (Plucky Puffin)	25.04	USN	Ja	Ja
Windows Server	2016	MSKB	Nein	Ja
Windows Server	2019	MSKB	Nein	Ja
Windows Server	2022	MSKB	Nein	Ja
Windows Server	2025	MSKB	Nein	Ja
macOS (Mojave)	10.14	APPLE-SA	Nein	Ja
macOS (Catalina)	10.15	APPLE-SA	Nein	Ja
macOS (Big Sur)	11	APPLE-SA	Nein	Ja
macOS (Monterey)	12	APPLE-SA	Nein	Ja
macOS (Ventura)	13	APPLE-SA	Nein	Ja

Betriebssystem	Version	Sicherheitsempfehlungen von Anbietern	Unterstützung für agentenloses Scannen	Unterstützung für agentengestütztes Scannen
macOS (Sonoma)	14	APPLE-SA	Nein	Ja

Unterstützte Betriebssysteme: Amazon ECR-Scannen mit Amazon Inspector

In der folgenden Tabelle sind die Betriebssysteme aufgeführt, die Amazon Inspector für das Scannen von Container-Images in Amazon ECR-Repositorys unterstützt. Sie enthält auch die Sicherheitsempfehlungen des Anbieters für jedes Betriebssystem.

Betriebssystem	Version	Sicherheitsempfehlungen des Anbieters
Alpine Linux (Alpine)	3.19	Alpine SecDB
Alpine Linux (Alpine)	3.20	Alpine SecDB
Alpine Linux (Alpine)	3.21	Alpine SecDB
Alpine Linux (Alpine)	3.22	Alpine SecDB
AlmaLinux	8	ALSA
AlmaLinux	9	ALSA
Amazon Linux (AL2)	AL2	ALAS
Amazon Linux 2023 (AL2023)	AL2023	ALAS
BusyBox	—	—
Chainguard	–	CVE
Debian Server (Bullseye)	11	DSA

Betriebssystem	Version	Sicherheitsempfehlungen des Anbieters
Debian Server (Bookworm)	12	DSA
Fedora	41	CVE
Fedora	42	CVE
openSUSE Leap	15.6	CVE
Oracle Linux (Oracle)	8	ELSA
Oracle Linux (Oracle)	9	ELSA
Photon OS	4	PHSA
Photon OS	5	PHSA
Red Hat Enterprise Linux (RHEL)	8	RHSA
Red Hat Enterprise Linux (RHEL)	9	RHSA
Rocky Linux	8	RLSA
Rocky Linux	9	RLSA
SUSE Linux Enterprise Server (SLES)	15.6	SUSE CVE
Ubuntu (Xenial)	16.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Bionic)	18.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Focal)	20.04	USN, Ubuntu Pro (esm-infra & esm-apps)

Betriebssystem	Version	Sicherheitsempfehlungen des Anbieters
Ubuntu (Jammy)	22.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Noble Numbat)	24.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Oracular Oriole)	24.10	USN
Ubuntu (Plucky Puffin)	25.04	USN
Wolfi	–	CVE

Unterstützte Betriebssysteme: CIS-Scanning

In der folgenden Tabelle sind die Betriebssysteme aufgeführt, die Amazon Inspector für CIS-Scans unterstützt. Außerdem wird die CIS-Benchmark-Version für jedes Betriebssystem angegeben.

> **Note**
>
> CIS-Standards sind für x86_64-Betriebssysteme vorgesehen. Einige Prüfungen werden möglicherweise nicht ausgewertet oder geben ungültige Anweisungen zur Behebung von ARM-basierten Ressourcen zurück.

Betriebssystem	Version	CIS-Benchmark-Version
Amazon Linux 2	AL2	3.0.0
Amazon Linux 2023	AL2023	1.0.0
Red Hat Enterprise Linux (RHEL)	8	3.0.0
Red Hat Enterprise Linux (RHEL)	9	2.0.0

Betriebssystem	Version	CIS-Benchmark-Version
Rocky Linux	8	2.0.0
Rocky Linux	9	1.0.0
SUSE-Linux-Enterprise-Server	15	2.0.1
Ubuntu (Bonic)	18,04	2.1.0
Ubuntu (fokal)	20.04	2.0.1
Ubuntu (Jammy)	22,04	1.0.0
Ubuntu (Edler Numbat)	24.04	1.0.0
Windows Server	2016	3.0.0
Windows Server	2019	2.0.0
Windows Server	2022	2.0.0

Eingestellte Betriebssysteme

In den folgenden Tabellen ist aufgeführt, welche Betriebssysteme eingestellt wurden und wann sie eingestellt wurden.

Auch wenn Amazon Inspector keine vollständige Unterstützung für die folgenden ausgelaufenen Betriebssysteme bietet, scannt Amazon Inspector weiterhin die EC2 Amazon-Instances und Amazon ECR-Container-Images, auf denen sie ausgeführt werden. Aus Sicherheitsgründen empfehlen wir, zur unterstützten Version eines eingestellten Betriebssystems zu wechseln. Ergebnisse, die Amazon Inspector für ein eingestelltes Betriebssystem generiert, sollten nur zu Informationszwecken verwendet werden.

Gemäß den Herstellerrichtlinien erhalten die folgenden Betriebssysteme keine Patch-Updates mehr. Für nicht mehr eingestellte Betriebssysteme werden möglicherweise keine neuen Sicherheitsempfehlungen veröffentlicht. Anbieter können bestehende Sicherheitsempfehlungen und Sicherheitswarnungen für Betriebssysteme, deren Standardsupport ausläuft, aus ihren Feeds

entfernen. Infolgedessen kann Amazon Inspector die Generierung von Ergebnissen für bekannte Personen beenden CVEs.

Eingestellte Betriebssysteme: EC2 Amazon-Scanning

Betriebssystem	Version	Nicht mehr angeboten
Amazon Linux (AL1)	2012	31. Dezember 2021
CentOS Linux (CentOS)	7	30. Juni 2024
CentOS Linux (CentOS)	8	31. Dezember 2021
Debian-Server (Jessie)	8	30. Juni 2020
Debian-Server (Stretch)	9	30. Juni 2022
Debian-Server (Buster)	10	30. Juni 2024
Fedora	33	30. November 2021
Fedora	34	7. Juni 2022
Fedora	35	13. Dezember 2022
Fedora	36	16. Mai 2023
Fedora	37	15. Dezember 2023
Fedora	38	21. Mai 2024
Fedora	39	26. November 2024
Fedora	40	13. Mai 2025
openSUSE Leap	15.2	1. Dezember 2021
OpenSUSE Leap	15.3	01. Dezember 2022
openSUSE Leap	15.4	07. Dezember 2023
OpenSUSE Leap	15.5	December 31, 2024

Betriebssystem	Version	Nicht mehr angeboten
Oracle Linux (Oracle)	6	1. März 2021
Oracle Linux (Oracle)	7	31. Dezember 2024
Red Hat Enterprise Linux (RHEL)	6	30. November 2020
Red Hat Enterprise Linux (RHEL)	7	30. Juni 2024
SUSE Linux Enterprise Server (SLES)	12	30. Juni 2016
SUSE Linux Enterprise Server (SLES)	12.1	31. Mai 2017
SUSE Linux Enterprise Server (SLES)	12.2	31. März 2018
SUSE Linux Enterprise Server (SLES)	12.3	30. Juni 2019
SUSE Linux Enterprise Server (SLES)	12.4	30. Juni 2020
SUSE Linux Enterprise Server (SLES)	12,5	31. Oktober 2024
SUSE Linux Enterprise Server (SLES)	15	31. Dezember 2019
SUSE Linux Enterprise Server (SLES)	15,1	31. Januar 2021
SUSE Linux Enterprise Server (SLES)	15.2	31. Dezember 2021

Betriebssystem	Version	Nicht mehr angeboten
SUSE Linux Enterprise Server (SLES)	15.3	31. Dezember 2022
SUSE Linux Enterprise Server (SLES)	15,4	31. Dezember 2023
SUSE Linux Enterprise Server (SLES)	15,5	31. Dezember 2024
Ubuntu (vertrauenswürdig)	12.04	28. April 2017
Ubuntu (Vertrauenswürdig)	14.04	1. April 2024
Ubuntu (Groovy)	20,10	22. Juli 2021
Ubuntu (Hirsute)	21,04	20. Januar 2022
Ubuntu (Impish)	21.10	31. Juli 2022
Ubuntu (Kinetic)	22.10	July 20, 2023
Ubuntu (Lunar Lobster)	23.04	January 25, 2024
Ubuntu (Mantischer Minotauru s)	23,10	11. Juli 2024
Windows Server	2012	10. Oktober 2023
Windows Server	2012 R2	10. Oktober 2023

Eingestellte Betriebssysteme: Amazon ECR-Scanning

Betriebssystem	Version	Nicht mehr angeboten
Alpine Linux (Alpine)	3.2	1. Mai 2017
Alpines Linux (Alpin)	3.3	1. November 2017

Betriebssystem	Version	Nicht mehr angeboten
Alpines Linux (Alpin)	3.4	1. Mai 2018
Alpines Linux (Alpin)	3.5	1. November 2018
Alpines Linux (Alpin)	3.6	1. Mai 2019
Alpines Linux (Alpin)	3.7	1. November 2019
Alpines Linux (Alpin)	3.8	1. Mai 2020
Alpines Linux (Alpin)	3.9	1. November 2020
Alpines Linux (Alpin)	3.10	1. Mai 2021
Alpines Linux (Alpin)	3.11	1. November 2021
Alpines Linux (Alpin)	3.12	01.Mai 2022
Alpines Linux (Alpin)	3.13	1. November 2022
Alpine Linux (Alpine)	3.14	May 1, 2023
Alpine Linux (Alpine)	3.15	November 1, 2023
Alpine Linux (Alpine)	3.16	May 23, 2024
Alpine Linux (Alpine)	3.17	November 22, 2024
Alpine Linux (Alpine)	3.18	May 09 2025
Amazon Linux (AL1)	2012	31. Dezember 2021
CentOS Linux (CentOS)	7	30. Juni 2024
CentOS Linux (CentOS)	8	31. Dezember 2021
Debian-Server (Jessie)	8	30. Juni 2020
Debian-Server (Stretch)	9	30. Juni 2022

Betriebssystem	Version	Nicht mehr angeboten
Debian-Server (Buster)	10	30. Juni 2024
Fedora	33	30. November 2021
Fedora	34	7. Juni 2022
Fedora	35	13. Dezember 2022
Fedora	36	16. Mai 2023
Fedora	37	15. Dezember 2023
Fedora	38	21. Mai 2024
Fedora	39	26. November 2024
Fedora	40	13. Mai 2025
openSUSE Leap	15.2	1. Dezember 2021
OpenSUSE Leap	15.3	01. Dezember 2022
OpenSUSE Leap	15.4	December 7, 2023
OpenSUSE Leap	15.5	December 31, 2024
Oracle Linux (Oracle)	6	1. März 2021
Oracle Linux (Oracle)	7	31. Dezember 2024
Photon OS	2	2. Dezember 2021
Photon OS	3	1. März 2024
Red Hat Enterprise Linux (RHEL)	6	30. Juni 2020
Red Hat Enterprise Linux (RHEL)	7	30. Juni 2024

Betriebssystem	Version	Nicht mehr angeboten
SUSE Linux Enterprise Server (SLES)	12	30. Juni 2016
SUSE Linux Enterprise Server (SLES)	12.1	31. Mai 2017
SUSE Linux Enterprise Server (SLES)	12.2	31. März 2018
SUSE Linux Enterprise Server (SLES)	12.3	30. Juni 2019
SUSE Linux Enterprise Server (SLES)	12.4	30. Juni 2020
SUSE Linux Enterprise Server (SLES)	12,5	31. Oktober 2024
SUSE Linux Enterprise Server (SLES)	15	31. Dezember 2019
SUSE Linux Enterprise Server (SLES)	15,1	31. Januar 2021
SUSE Linux Enterprise Server (SLES)	15.2	31. Dezember 2021
SUSE Linux Enterprise Server (SLES)	15.3	31. Dezember 2022
SUSE Linux Enterprise Server (SLES)	15,4	31. Dezember 2023
SUSE Linux Enterprise Server (SLES)	15,5	31. Dezember 2024
Ubuntu (vertrauenswürdig)	12.04	28. April 2017

Betriebssystem	Version	Nicht mehr angeboten
Ubuntu (Vertrauenswürdig)	14.04	1. April 2024
Ubuntu (Groovy)	20,10	22. Juli 2021
Ubuntu (Hirsute)	21,04	20. Januar 2022
Ubuntu (Impish)	21.10	31. Juli 2022
Ubuntu (Kinetic)	22.10	July 20, 2023
Ubuntu (Lunar Lobster)	23.04	January 25, 2024
Ubuntu (Mantischer Minotauru s)	23,10	11. Juli 2024

Unterstützte Programmiersprachen

In diesem Abschnitt sind die Programmiersprachen aufgeführt, die Amazon Inspector unterstützt.

Unterstützte Programmiersprachen: EC2 Amazon-Scanning ohne Agenten

Amazon Inspector unterstützt derzeit die folgenden Programmiersprachen bei der Durchführung von agentenlosen Scans auf geeigneten EC2 Amazon-Instances. Weitere Informationen finden Sie unter Agentloses Scannen.

> **Note**
>
> Amazon Inspector sucht in Go und Rust nicht nach Schwachstellen in der Toolchain. Die Version des Programmiersprachen-Compilers, der zur Erstellung der Anwendung verwendet wurde, führt zu diesen Sicherheitslücken.

- C#
- Go
- Java
- JavaScript

- PHP
- Python
- Ruby
- Rust

Unterstützte Programmiersprachen: Amazon EC2 Deep Inspection

Amazon Inspector unterstützt derzeit die folgenden Programmiersprachen bei der Durchführung von Deep Inspection-Scans auf Amazon EC2 Linux-Instances. Weitere Informationen finden Sie unter Amazon Inspector Deep Inspection für Linux-basierte EC2 Amazon-Instances.

- Java(Archivformate .ear, .jar, .par und .war)
- JavaScript
- Python

Amazon Inspector verwendet Systems Manager Distributor, um das Plug-in für eine gründliche Inspektion Ihrer EC2 Amazon-Instance bereitzustellen.

> **Note**
>
> Die Tiefeninspektion wird für Bottlerocket-Betriebssysteme nicht unterstützt.

Um Deep Inspection Scans durchführen zu können, müssen Systems Manager Distributor und Amazon Inspector Ihr EC2 Amazon-Instance-Betriebssystem unterstützen. Informationen zu den unterstützten Betriebssystemen in Systems Manager Distributor finden Sie unter Unterstützte Paketplattformen und Architekturen im Systems Manager Manager-Benutzerhandbuch.

Unterstützte Programmiersprachen: Amazon ECR Scanning

Amazon Inspector unterstützt derzeit die folgenden Programmiersprachen beim Scannen von Container-Images in Amazon ECR-Repositorys:

> **Note**
>
> Amazon Inspector sucht nicht nach Schwachstellen in Rust der Toolchain. Die Version des Programmiersprachen-Compilers, der zur Erstellung der Anwendung verwendet wurde, führt zu diesen Sicherheitslücken.

- C#
- Go
- GoToolchain
- Java
- JavaJDK
- JavaScript
- PHP
- Python
- Ruby
- Rust

Unterstützte Laufzeiten

In diesem Abschnitt sind die Laufzeiten aufgeführt, die Amazon Inspector unterstützt.

Unterstützte Laufzeiten: Amazon Inspector Lambda Standard-Scanning

Das Standard-Scannen von Amazon Inspector Lambda unterstützt derzeit die folgenden Laufzeiten für die Programmiersprachen, die beim Scannen von Lambda-Funktionen auf Sicherheitslücken in Softwarepaketen von Drittanbietern verwendet werden können:

> **Note**
>
> Amazon Inspector sucht nicht nach Schwachstellen in Rust der Toolchain. Die Version des Programmiersprachen-Compilers, der zur Erstellung der Anwendung verwendet wurde, führt zu diesen Sicherheitslücken.

- Go

  - go1.x
- Java
  - java8
  - java8.al2
  - java11
  - java17
  - java21
- .NET
  - .NET 6
  - .NET 8
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
  - nodejs22.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
  - python3.12
  - python3.13
- Ruby
  - ruby2.7
  - ruby3.2
  - ruby3.3

- Custom runtimes

- AL2
- AL2023

Unterstützte Laufzeiten: Amazon Inspector Lambda-Code-Scanning

Das Lambda-Codescanning von Amazon Inspector unterstützt derzeit die folgenden Laufzeiten für die Programmiersprachen, die beim Scannen von Lambda-Funktionen auf Sicherheitslücken im Code verwendet werden können:

- Java
  - java8
  - java8.al2
  - java11
  - java17
- .NET
  - .NET 6
  - .NET 8
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
  - python3.12
- Ruby
  - ruby2.7

- ruby3.2
- ruby3.3

Amazon Inspector deaktivieren

Sie können Amazon Inspector in der Amazon Inspector Inspector-Konsole oder mit der Amazon Inspector Inspector-API deaktivieren. Wenn Sie alle Scanarten für ein Konto deaktivieren, wird Amazon Inspector für dieses Konto automatisch deaktiviert.

Wenn Sie Amazon Inspector für ein Konto deaktivieren, werden alle Scanarten für dieses Konto deaktiviert. Darüber hinaus werden alle Amazon Inspector-Scaneinstellungen, einschließlich Filter, Unterdrückungsregeln und Ergebnisse, für das Konto gelöscht.

Wenn Sie Amazon Inspector EC2 Amazon-Scanning deaktivieren, löscht Amazon Inspector die folgenden SSM-Verknüpfungen:

- `InspectorDistributor-do-not-delete`
- `InspectorInventoryCollection-do-not-delete`
- `InvokeInspectorSsmPlugin-do-not-delete`. Darüber hinaus wird das Amazon Inspector SSM-Plugin, das über diese Verknüpfung installiert wurde, aus allen Ihren Windows Gastgeber. Weitere Informationen finden Sie unter Windows EC2 Instanz wird gescannt.

> **Note**
>
> Sobald Sie Amazon Inspector deaktivieren, fallen keine Servicegebühren mehr an. Sie können Amazon Inspector jedoch jederzeit reaktivieren.

Informationen zum Deaktivieren von Scantypen für verschiedene Ressourcen finden Sie unter Deaktivieren eines Scantyps.

Voraussetzungen

Beachten Sie je nach Kontotyp Folgendes:

- Wenn es sich bei Ihrem Konto um ein eigenständiges Amazon Inspector-Konto handelt, können Sie Amazon Inspector jederzeit deaktivieren.
- Wenn es sich bei Ihrem Konto um ein Mitgliedskonto in einer Umgebung mit mehreren Konten handelt, können Sie Amazon Inspector nicht deaktivieren. Sie müssen sich an den delegierten Administrator Ihrer Organisation wenden, um Amazon Inspector zu deaktivieren.

- Wenn Sie der delegierte Administrator für eine Organisation sind, müssen Sie alle Ihre Mitgliedskonten trennen, bevor Sie Amazon Inspector deaktivieren.

> **Note**
>
> Wenn Sie Amazon Inspector als delegierter Administrator deaktivieren, deaktivieren Sie die automatische Aktivierungsfunktion für Ihre Organisation.

Amazon Inspector deaktivieren

> **Note**
>
> Bevor Sie Amazon Inspector deaktivieren, sollten Sie erwägen, Ihre Ergebnisse zu exportieren.

Console

Um Amazon Inspector zu deaktivieren

1. Melden Sie sich mit Ihren Anmeldeinformationen an und öffnen Sie dann die Amazon Inspector Inspector-Konsole unter https://console.aws.amazon.com/inspector/v2/home.
2. Wählen Sie mithilfe der AWS-Region Auswahl in der oberen rechten Ecke der Seite die Region aus, in der Sie Amazon Inspector deaktivieren möchten.
3. Wählen Sie im Navigationsbereich Allgemeine Einstellungen aus.
4. Wählen Sie „Inspector deaktivieren“.
5. Wenn Sie zur Bestätigung aufgefordert werden, geben Sie in das Textfeld deaktivieren ein und wählen Sie dann Inspector deaktivieren.
6. (Empfohlen) Wiederholen Sie diese Schritte in jeder Region, für die Sie Amazon Inspector deaktivieren möchten.

API

Führen Sie den Vorgang „API deaktivieren“ aus. Geben Sie in der Anfrage das Konto an, das IDs Sie deaktivieren `resourceTypes` möchten, und `EC2, ECR, LAMBDA` für die Deaktivierung aller Scans, wodurch das Konto deaktiviert wird.

Amazon Inspector Inspector-Kontingente

In diesem Abschnitt sind die Amazon Inspector Inspector-Kontingente pro aufgeführt AWS-Region.

Ressource	Standard	Kommentare
Mitgliedskonten	10.000	Die maximale Anzahl von Mitgliedskonten, die einem delegierten Administratorkonto von Amazon Inspector zugeordnet sind. Das Limit basiert auf den Kontingenten für AWS Organizations.
Unterdrückungsregeln	500	Die maximale Anzahl an gespeicherten Unterdrückungsregeln pro AWS Konto und Region. Sie können keine Kontingenterhöhung beantragen.
Ergebnisse EC2 des Amazon-Netzwerks	10.000	Die maximale Anzahl von EC2 Amazon-Netzwerkergebnissen pro AWS Konto. Sie können keine Kontingenterhöhung beantragen.
CIS-Scan-Konfigurationen	500	Die maximale Anzahl von CIS-Scan-Konfigurationen. Sie können keine

Ressource	Standard	Kommentare
		Kontingenterhöhung beantragen.

Eine Liste der mit Amazon Inspector Classic verknüpften Kontingente finden Sie unter Amazon Inspector Classic-Servicekontingente in der Allgemeine AWS-Referenz. Eine Liste der damit AWS Organizations verbundenen Kontingente finden Sie unter AWS Organizations Servicekontingenten in der Allgemeine AWS-Referenz.

Regionen und Endpunkte

Dieses Thema enthält Tabellen mit Endpunkten für Amazon Inspector und Amazon Inspector Scan. Es enthält auch Tabellen, die zeigen, welche Amazon Inspector Inspector-Funktionen AWS-Regionen unterstützen. Informationen darüber, AWS-Regionen wo Amazon Inspector verfügbar ist, finden Sie unter Amazon Inspector Inspector-Endpunkt und Kontingente in der Allgemeine Amazon Web Services-Referenz.

Service-Endpunkte für Amazon Inspector

Die folgende Tabelle zeigt die Service-Endpunkte für Amazon Inspector. Die Namenskonvention für Amazon Inspector Inspector-Endpunkte lautet`inspector2.`*`Region`*`.amazonaws.com`.

Name der Region	Region	Endpunkt	Protokoll
Kanada West (Calgary)	ca-west-1	inspector2.ca-west-1.amazonaws.com inspector2.ca-west-1.api.aws.com	HTTPS
USA Ost (Nord-Virginia)	us-east-1	inspector2.us-east-1.amazonaws.com inspector2.us-east-1.api.aws.com inspector2-fips.us-east-1.amazonaws.com	HTTPS
USA Ost (Ohio)	us-east-2	inspector2.us-east-2.amazonaws.com inspector2.us-east-2.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
		inspector2-fips.us-east-2.amazonaws.com	
USA West (Nordkalifornien)	us-west-1	inspector2.us-west-1.amazonaws.com inspector2.us-west-1.api.aws.com inspector2-fips.us-west-1.amazonaws.com	HTTPS
USA West (Oregon)	us-west-2	inspector2.us-west-2.amazonaws.com inspector2.us-west-2.api.aws.com inspector2-fips.us-west-2.amazonaws.com	HTTPS
Afrika (Kapstadt)	af-south-1	inspector2.af-south-1.amazonaws.com inspector2.af-south-1.api.aws.com	HTTPS
Asien-Pazifik (Hongkong)	ap-east-1	inspector2.ap-east-1.amazonaws.com inspector2.ap-east-1.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Sydney)	ap-southeast-2	inspector2.ap-southeast-2.amazonaws.com inspector2.ap-southeast-2.api.aws.com	HTTPS
Asien-Pazifik (Jakarta)	ap-southeast-3	inspector2.ap-southeast-3.amazonaws.com inspector2.ap-southeast-3.api.aws.com	HTTPS
Asien-Pazifik (Melbourne)	ap-southeast-4	inspector2.ap-southeast-4.amazonaws.com inspector2.ap-southeast-4.api.aws.com	HTTPS
Asien-Pazifik (Malaysia)	ap-southeast-5	inspector2.ap-southeast-5.amazonaws.com inspector2.ap-southeast-5.api.aws.com	HTTPS
Asien-Pazifik (Thailand)	ap-southeast-7	inspector2.ap-southeast-7.amazonaws.com inspector2.ap-southeast-7.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Mumbai)	ap-south-1	inspector2.ap-south-1.amazonaws.com inspector2.ap-south-1.api.aws.com	HTTPS
Asien-Pazifik (Hyderabad)	ap-south-2	inspector2.ap-south-2.amazonaws.com inspector2.ap-south-2.api.aws.com	HTTPS
Asia Pacific (Osaka)	ap-northeast-3	inspector2.ap-northeast-3.amazonaws.com inspector2.ap-northeast-3.api.aws.com	HTTPS
Asien-Pazifik (Seoul)	ap-northeast-2	inspector2.ap-northeast-2.amazonaws.com inspector2.ap-northeast-2.api.aws.com	HTTPS
Asien-Pazifik (Singapur)	ap-southeast-1	inspector2.ap-southeast-1.amazonaws.com inspector2.ap-southeast-1.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Sydney)	ap-southeast-2	inspector2.ap-southeast-2.amazonaws.com inspector2.ap-southeast-2.api.aws.com	HTTPS
Asien-Pazifik (Tokio)	ap-northeast-1	inspector2.ap-northeast-1.amazonaws.com inspector2.ap-northeast-1.api.aws.com	HTTPS
Kanada (Zentral)	ca-central-1	inspector2.ca-central-1.amazonaws.com inspector2.ca-central-1.api.aws.com	HTTPS
Europa (Frankfurt)	eu-central-1	inspector2.eu-central-1.amazonaws.com inspector2.eu-central-1.api.aws.com	HTTPS
Europa (Irland)	eu-west-1	inspector2.eu-west-1.amazonaws.com inspector2.eu-west-1.api.aws.com	HTTPS
Europa (London)	eu-west-2	inspector2.eu-west-2.amazonaws.com inspector2.eu-west-2.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Europa (Mailand)	eu-south-1	inspector2.eu-south-1.amazonaws.com inspector2.eu-south-1.api.aws.com	HTTPS
Europa (Spain)	region-eu-south-2	Inspektor 2. region-eu-south-2.amazonaws.com Inspektor 2. region-eu-south-2.api.aws.com	HTTPS
Europa (Paris)	eu-west-3	inspector2.eu-west-3.amazonaws.com inspector2.eu-west-3.api.aws.com	HTTPS
Europa (Stockholm)	eu-north-1	inspector2.eu-north-1.amazonaws.com inspector2.eu-north-1.api.aws.com	HTTPS
Europa (Zürich)	eu-central-2	inspector2.eu-central-2.amazonaws.com inspector2.eu-central-2.api.aws.com	HTTPS
Israel (Tel Aviv)	il-central-1	inspector2.il-central-1.amazonaws.com inspector2.il-central-1.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Naher Osten (VAE)	me-central-1	inspector2.me-central-1.amazonaws.com inspector2.me-central-1.api.aws.com	HTTPS
Naher Osten (Bahrain)	me-south-1	inspector2.me-south-1.amazonaws.com inspector2.me-south-1.api.aws.com	HTTPS
Mexiko (Zentral)	mx-central-1	inspector2.mx-central-1.amazonaws.com inspector2.mx-central-1.api.aws.com	HTTPS
Südamerika (São Paulo)	sa-east-1	inspector2.sa-east-1.amazonaws.com inspector2.sa-east-1.api.aws.com	HTTPS
AWS GovCloud (US-Ost)	us-gov-east-1	Inspektor 2. us-gov-east-1.amazonaws.com Inspektor 2. us-gov-east-1.api.aws.com Inspektor2-fips. us-gov-east-1.amazonaws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
AWS GovCloud (US-West)	us-gov-west-1	Inspektor 2. us-gov-west-1.amazonaws.com Inspektor 2. us-gov-west-1.api.aws.com Inspektor2-fips. us-gov-west-1.amazonaws.com	HTTPS

Endpunkte für die Amazon Inspector Scan API

Die folgende Tabelle zeigt die regionalen Endpunkte, die beim Aufrufen der Amazon Inspector Scan API verwendet werden können. Wenn Sie die API verwenden, müssen Sie den Endpunkt und die entsprechende Region für die AWS Region angeben, in der Sie derzeit authentifiziert sind.

Die Namenskonvention für Amazon Inspector Scan-Endpunkte lautet`inspector-scan.`*`region`*`.amazonaws.com`. Wenn Sie beispielsweise authentifiziert sind, würden Sie den Endpunkt verwenden`us-west-2`, `inspector-scan.us-west-2.amazonaws.com` um die API aufzurufen. `inspector-scan`

Name der Region	Region	Endpunkt	Protokoll
Kanada West (Calgary)	ca-west-1	inspector-scan.ca-west-1.amazonaws.com inspector-scan.ca-west-1.api.aws.com	HTTPS
USA Ost (Ohio)	us-east-2	inspector-scan.us-east-2.amazonaws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
		inspector-scan.us-east-2.api.aws.com inspector-scan-fips.us-east-2.amazonaws.com	
USA Ost (Nord-Virginia)	us-east-1	inspector-scan.us-east-1.amazonaws.com inspector-scan.us-east-1.api.aws.com inspector-scan-fips.us-east-1.amazonaws.com	HTTPS
USA West (Nordkalifornien)	us-west-1	inspector-scan.us-west-1.amazonaws.com inspector-scan.us-west-1.api.aws.com inspector-scan-fips.us-west-1.amazonaws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
USA West (Oregon)	us-west-2	inspector-scan.us-west-2.amazonaws.com inspector-scan.us-west-2.api.aws.com inspector-scan-fips.us-west-2.amazonaws.com	HTTPS
Afrika (Kapstadt)	af-south-1	inspector-scan.af-south-1.amazonaws.com inspector-scan.af-south-1.api.aws.com	HTTPS
Asien-Pazifik (Hongkong)	ap-east-1	inspector-scan.ap-east-1.amazonaws.com inspector-scan.ap-east-1.api.aws.com	HTTPS
Asien-Pazifik (Jakarta)	ap-southeast-3	inspector-scan.ap-southeast-3.amazonaws.com inspector-scan.ap-southeast-3.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Mumbai)	ap-south-1	inspector-scan.ap-south-1.amazonaws.com inspector-scan.ap-south-1.api.aws.com	HTTPS
Asia Pacific (Osaka)	ap-northeast-3	inspector-scan.ap-northeast-3.amazonaws.com inspector-scan.ap-northeast-3.api.aws.com	HTTPS
Asien-Pazifik (Seoul)	ap-northeast-2	inspector-scan.ap-northeast-2.amazonaws.com inspector-scan.ap-northeast-2.api.aws.com	HTTPS
Asien-Pazifik (Hyderabad)	ap-south-2	inspector-scan.ap-south-2.amazonaws.com inspector-scan.ap-south-2.api.aws.com	HTTPS
Asien-Pazifik (Singapur)	ap-southeast-1	inspector-scan.ap-southeast-1.amazonaws.com inspector-scan.ap-southeast-1.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Sydney)	ap-southeast-2	inspector-scan.ap-southeast-2.amazonaws.com inspector-scan.ap-southeast-2.api.aws.com	HTTPS
Asien-Pazifik (Melbourne)	ap-southeast-4	inspector-scan.ap-southeast-4.amazonaws.com inspector-scan.ap-southeast-4.api.aws.com	HTTPS
Asien-Pazifik (Malaysia)	ap-southeast-5	inspector-scan.ap-southeast-5.amazonaws.com inspector-scan.ap-southeast-5.api.aws.com	HTTPS
Asien-Pazifik (Thailand)	ap-southeast-7	inspector-scan.ap-southeast-7.amazonaws.com inspector-scan.ap-southeast-7.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Asien-Pazifik (Tokio)	ap-northeast-1	inspector-scan.ap-northeast-1.amazonaws.com inspector-scan.ap-northeast-1.api.aws.com	HTTPS
Kanada (Zentral)	ca-central-1	inspector-scan.ca-central-1.amazonaws.com inspector-scan.ca-central-1.api.aws.com	HTTPS
Europa (Frankfurt)	eu-central-1	inspector-scan.eu-central-1.amazonaws.com inspector-scan.eu-central-1.api.aws.com	HTTPS
Europa (Irland)	eu-west-1	inspector-scan.eu-west-1.amazonaws.com inspector-scan.eu-west-1.api.aws.com	HTTPS
Europa (London)	eu-west-2	inspector-scan.eu-west-2.amazonaws.com inspector-scan.eu-west-2.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Europa (Mailand)	eu-south-1	inspector-scan.eu-south-1.amazonaws.com inspector-scan.eu-south-1.api.aws.com	HTTPS
Europa (Spanien)	eu-south-2	inspector-scan.eu-south-2.amazonaws.com inspector-scan.eu-south-2.api.aws.com	HTTPS
Europa (Paris)	eu-west-3	inspector-scan.eu-west-3.amazonaws.com inspector-scan.eu-west-3.api.aws.com	HTTPS
Europa (Stockholm)	eu-north-1	inspector-scan.eu-north-1.amazonaws.com inspector-scan.eu-north-1.api.aws.com	HTTPS
Europa (Zürich)	eu-central-2	inspector-scan.eu-central-2.amazonaws.com inspector-scan.eu-central-2.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
Naher Osten (VAE)	me-central-1	inspector-scan.me-central-1.amazonaws.com inspector-scan.me-central-1.api.aws.com	HTTPS
Naher Osten (Bahrain)	me-south-1	inspector-scan.me-south-1.amazonaws.com inspector-scan.me-south-1.api.aws.com	HTTPS
Mexiko (Zentral)	mx-central-1	inspector-scan.mx-central-1.amazonaws.com inspector-scan.mx-central-1.api.aws.com	HTTPS
Israel (Tel Aviv)	il-central-1	inspector-scan.il-central-1.amazonaws.com inspector-scan.il-central-1.api.aws.com	HTTPS
Südamerika (São Paulo)	sa-east-1	inspector-scan.sa-east-1.amazonaws.com inspector-scan.sa-east-1.api.aws.com	HTTPS

Name der Region	Region	Endpunkt	Protokoll
AWS GovCloud (US-Ost)	us-gov-east-1	Inspektor-Scan. us-gov-east-1.amazonaws.com Inspektor-Scan. us-gov-east-1.api.aws.com inspector-scan-fips. us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud (US-West)	us-gov-west-1	Inspektor-Scan. us-gov-west-1.amazonaws.com Inspektor-Scan. us-gov-west-1.api.aws.com inspector-scan-fips. us-gov-west-1.amazonaws.com	HTTPS

Verfügbarkeit regionsspezifischer Feature

In diesem Abschnitt wird die Verfügbarkeit der Amazon Inspector Inspector-Funktionen von beschrieben AWS-Region.

Agentenloses EC2 Scannen für Amazon-Regionen EC2

Die folgende Tabelle zeigt AWS-Regionen , wo agentenloses Scannen für Amazon derzeit verfügbar EC2 ist.

Name der Region	Regionscode
USA Ost (Nord-Virginia)	us-east-1
USA Ost (Ohio)	us-east-2
USA West (Nordkalifornien)	us-west-1
USA West (Oregon)	us-west-2
Afrika (Kapstadt)	af-south-1
Asien-Pazifik (Hongkong)	ap-east-1
Asien-Pazifik (Tokio)	ap-northeast-1
Asien-Pazifik (Seoul)	ap-northeast-2
Asia Pacific (Osaka)	ap-northeast-3
Asien-Pazifik (Mumbai)	ap-south-1
Asien-Pazifik (Hyderabad)	ap-south-2
Asien-Pazifik (Singapur)	ap-southeast-1
Asien-Pazifik (Sydney)	ap-southeast-2
Asien-Pazifik (Jakarta)	ap-southeast-3
Asien-Pazifik (Melbourne)	ap-southeast-4
Asien-Pazifik (Malaysia)	ap-southeast-5
Asien-Pazifik (Thailand)	ap-southeast-7
Kanada (Zentral)	ca-central-1
Kanada West (Calgary)	ca-west-1
Europa (Stockholm)	eu-north-1

Name der Region	Regionscode
Europa (Frankfurt)	eu-central-1
Europa (Zürich)	eu-central-2
Europa (Irland)	eu-west-1
Europa (London)	eu-west-2
Europa (Paris)	eu-west-3
Europa (Mailand)	eu-south-1
Europa (Spanien)	eu-south-2
Israel (Tel Aviv)	il-central-1
Naher Osten (VAE)	me-central-1
Naher Osten (Bahrain)	me-south-1
Mexiko (Zentral)	mx-central-1
Südamerika (São Paulo)	sa-east-1
AWS GovCloud (US-Ost)	us-gov-east-1
AWS GovCloud (US-West)	us-gov-west-1

Lambda-Code-Scanning-Regionen

Die folgende Tabelle zeigt, AWS-Regionen wo Lambda-Code-Scanning derzeit verfügbar ist.

Name der Region	Regionscode
USA Ost (Nord-Virginia)	us-east-1
USA West (Oregon)	us-west-2
USA Ost (Ohio)	us-east-2

Name der Region	Regionscode
Asien-Pazifik (Sydney)	ap-southeast-2
Asien-Pazifik (Tokio)	ap-northeast-1
Europa (Frankfurt)	eu-central-1
Europa (Irland)	eu-west-1
Europa (London)	eu-west-2
Europa (Stockholm)	eu-north-1
Asien-Pazifik (Singapur)	ap-southeast-1

Important

Wenn Sie versuchen, Lambda-Code-Scanning mit der Amazon Inspector Enable API in einer AWS-Region Umgebung zu aktivieren, in der Lambda-Code-Scans nicht verfügbar sind, wird Ihnen die folgende Fehlermeldung „Zugriff verweigert“ angezeigt:

```
An error occurred (AccessDeniedException) when calling the Enable operation:
 Lambda code scanning is not supported in unsupported-AWS-Region
```

Amazon Inspector Code — Sicherheitsregionen

Die folgende Tabelle zeigt AWS-Regionen , wo Amazon Inspector Code Security derzeit verfügbar ist.

Name der Region	Regionscode
USA Ost (Nord-Virginia)	us-east-1
USA West (Oregon)	us-west-2
USA Ost (Ohio)	us-east-2

Name der Region	Regionscode
Asien-Pazifik (Sydney)	ap-southeast-2
Asien-Pazifik (Tokio)	ap-northeast-1
Europa (Frankfurt)	eu-central-1
Europa (Irland)	eu-west-1
Europa (London)	eu-west-2
Europa (Stockholm)	eu-north-1
Asien-Pazifik (Singapur)	ap-southeast-1

AWS GovCloud (US) Regionen

Die neuesten Informationen finden Sie unter Amazon Inspector im AWS GovCloud (US) Benutzerhandbuch.

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in den einzelnen Versionen des Amazon Inspector Inspector-Benutzerhandbuchs ab November 2021 beschrieben. Um Benachrichtigungen über Aktualisierungen der Dokumentation zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Neue Richtlinie	Amazon Inspector fügt eine neue verwaltete Richtlini e hinzu, die vollen Zugriff auf Amazon Inspector und Zugriff auf andere verwandte Services bietet. Weitere Informationen finden Sie unter AWS Verwaltete Richtlinien für Amazon Inspector.	3. Juli 2025
Aktualisierte Funktionalität	Amazon Inspector ist jetzt in neuer Version erhältlic h AWS-Regionen. Weitere Informationen finden Sie unter Regionen und Endpunkte.	1. Juli 2025
Aktualisierte Funktionalität	Amazon Inspector aktualisiert seine Aufbewahrungsfrist für abgeschlossene Ergebniss e. Amazon Inspector entfernt Ergebnisse nach 3 Tagen, wenn zugehörige Ressourcen gelöscht oder beendet wurden oder nicht mehr für das Scannen in Frage kommen. Weitere Informationen finden Sie unter Grundlegendes zu	25. Juni 2025

	den Ergebnissen von Amazon Inspector.	
Aktualisierte Funktionalität	Amazon Inspector aktualisiert seine unterstützten Betriebssysteme für EC2 Amazon-Scanning und Amazon ECR-Scannen. Amazon EC2 Scanning unterstützt jetzt Fedora Version 42 und Ubuntu Version 25.04. Amazon ECR Scanning unterstützt jetzt Alpine Version 3.22, Fedora Version 42 und Ubuntu Version 25.04. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	18. Juni 2025
Neues Feature	Amazon Inspector scannt jetzt den Quellcode von Erstanbieteranwendungen, Abhängigkeiten von Drittanbieteranwendungen und Infrastructure as Code auf Sicherheitslücken. Weitere Informationen finden Sie unter Amazon Inspector Code Security.	17. Juni 2025

Update für das Plugin	Amazon Inspector wird auf ein Szenario aufmerksam gemacht, in dem das Amazon Inspector SSM-Plugin möglicherweise eine Sicherheitslücke für `CVE-2025-0913` und `CVE-2025-4673` entdeckt. Es wurde bestätigt, dass das Amazon Inspector SSM-Plugin davon nicht betroffen ist CVEs, und es wurde ein Update bereitgestellt, um diese Erkennung zu beheben.	13. Juni 2025
Neues Feature	Amazon Inspector kann jetzt aktiv verwendete Container-Images und den Zeitpunkt der letzten Verwendung von Container-Images in einem Cluster anzeigen. Weitere Informationen finden Sie unter Zuordnen von Container-Images zu laufenden Containern.	16. Mai 2025
Neues Feature	Amazon Inspector kann jetzt aktiv verwendete Container-Images und den Zeitpunkt der letzten Verwendung von Container-Images in einem Cluster anzeigen. Weitere Informationen finden Sie unter Zuordnen von Container-Images zu laufenden Containern.	16. Mai 2025

Updates für unterstützte Betriebssysteme	Amazon Inspector fügt Unterstützung hinzu BusyBox für Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	13. Mai 2025
Aktualisierte Richtlinie	Amazon Inspector fügt der mit dem Service verknüpften Rolle namens `AmazonInspector2ServiceRolePolicy` eine neue Berechtigung hinzu. Mit dieser Berechtigung können Sie IP-Adressen und Internet-Gateways beschreiben. Weitere Informationen finden Sie unter AWS Verwaltete Richtlinien für Amazon Inspector.	29. April 2025
Update für das Plugin	Amazon Inspector wird auf ein Szenario aufmerksam gemacht, in dem das Amazon Inspector SSM-Plugin möglicherweise eine Schwachstellensuche für `CVE-2025-22871` generiert. Es wurde bestätigt, dass das Amazon Inspector SSM-Plugin davon nicht betroffen ist CVEs, und es wurde ein Update bereitgestellt, um diese Erkennung zu beheben.	21. April 2025

Update für das Plugin	Amazon Inspector wird auf ein Szenario aufmerksam gemacht, in dem das Amazon Inspector SSM-Plugin eine Sicherheitslücke für `CVE-2020-8911` `CVE-2020-8912` , und `CVE-2024-45337` finden könnte. Es wurde bestätigt, dass Amazon Inspector davon nicht betroffen ist, CVEs und es wurde ein Update bereitgestellt, um diese Erkennung zu beheben.	18. April 2025
Aktualisierungen des Kapitels Amazon Inspector SBOM Generator	Amazon Inspector aktualisiert die Version von Amazon Inspector SBOM Generator. Weitere Informationen finden Sie unter Frühere Versionen des Amazon Inspector SBOM Generators.	16. April 2025
Aktualisierungen des Kapitels Amazon Inspector SBOM Generator	Amazon Inspector fügt dem Kapitel Amazon Inspector SBOM Generator ein neues Thema hinzu. In diesem Thema wird beschrieben, wie Lizenzinformationen in einer Softwareliste Sbomgen nachverfolgt werden. Weitere Informationen finden Sie unter Lizenzsammlung für Amazon Inspector SBOM Generator.	16. April 2025

Aktualisierungen der verwaltet en Richtlinien	Amazon Inspector fügt Berechtigungen hinzu, die nur Lesezugriff auf Amazon ECS- und Amazon EKS-Aktio nen ermöglichen. Weitere Informationen finden Sie unter Servicebezogene Rollenber echtigungen für Amazon Inspector.	25. März 2025
Updates für unterstützte Betriebssysteme	Amazon Inspector unterstüt zt im SUSE Linux Enterpris e Server 12.5 Rahmen des Scannens für Amazon EC2 und Amazon ECR nicht mehr. Weitere Informationen finden Sie unter Unterstüt zte Betriebssysteme und Programmiersprachen für Amazon Inspector.	21. März 2025
Updates für unterstützte Betriebssysteme	Amazon Inspector fügt Unterstützung für Chainguar d und Wolfi zum Amazon ECR-Scannen hinzu. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	21. März 2025
Aktualisierungen des Inhaltsve rzeichnisses	Amazon Inspector fügt ein Kapitel über das Taggen von Amazon Inspector Inspector -Ressourcen hinzu. Weitere Informationen finden Sie unter Ressourcen zum Taggen von Amazon Inspector.	25. Februar 2025

Aktualisierungen des Inhaltsverzeichnisses	Amazon Inspector fügt dem Kapitel Amazon Inspector SBOM Generator ein neues Thema hinzu. Weitere Informationen finden Sie in der umfassenden Betriebssystemsammlung von Amazon Inspector SBOM Generator.	28. Januar 2025
Aktualisierte Funktionalität	Amazon Inspector erweitert nodejs202.x seine Liste der unterstützten Laufzeiten für Lambda-Standardscans um und. python3.13 Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	24. Januar 2025
Aktualisierte Funktionalität	Amazon Inspector entfernt Oracle Linux (Oracle) 7 und SUSE Linux Enterprise Server (SLES) 15.5 aus seiner Liste der unterstützten Betriebssysteme für Amazon EC2 und Amazon ECR. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	31. Dezember 2024

Aktualisierte Funktionalität	Amazon Inspector erweitert seine Liste der unterstützten Betriebssysteme für Amazon EC2 und Amazon ECR um Ubuntu 24.10. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	12. Dezember 2024
Aktualisierungen des Inhaltsverzeichnisses	Amazon Inspector fügt dem Kapitel Amazon Inspector SBOM Generator neue Themen hinzu. Weitere Informationen finden Sie unter Amazon Inspector SBOM Generator.	9. Dezember 2024
Aktualisierte Funktionalität	Amazon Inspector aktualisiert die `amazon:inspector:sbom_generator` Tabelle, um Namespaces hinzuzufügen und zu entfernen. Weitere Informationen finden Sie unter Verwenden von CyclonedX-Namespaces mit Amazon Inspector.	9. Dezember 2024
Aktualisierte Funktionalität	Amazon Inspector aktualisiert seine CI/CD-Integrationsfunktion zur Unterstützung von Scanaktionen mit. CodePipeline Weitere Informationen finden Sie unter Amazon Inspector Scan-Aktionen verwenden mit CodePipeline.	26. November 2024

Aktualisierungen des Inhaltsverzeichnisses	Amazon Inspector organisiert das Inhaltsverzeichnis neu und enthält nun ein Kapitel für den Amazon Inspector SBOM Generator. Weitere Informationen finden Sie unter Amazon Inspector SBOM Generator.	22. November 2024
Aktualisierte Funktionalität	Amazon Inspector entfernt Fedora 39 von seiner Liste der unterstützten Betriebssysteme für Amazon EC2 und Amazon ECR. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	22. November 2024
Aktualisierte Funktionalität	Amazon Inspector entfernt Alpine 3.17 aus seiner Liste der unterstützten Betriebssysteme für Amazon ECR. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	22. November 2024
Aktualisierte Funktionalität	Amazon Inspector fügt Sbomgen Versionen zu früheren Versionen des Amazon Inspector SBOM Generators hinzu.	19. November 2024

Aktualisierte Funktionalität	Amazon Inspector wird AL2 als unterstützte Laufzeit hinzugefügt. Weitere Informationen finden Sie unter Unterstützte Betriebssysteme und Programmiersprachen für Amazon Inspector.	26. August 2024
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Erklärung hinzugefügt. Die neue Anweisung ermöglicht es Amazon Inspector, Funktions-Tags zurückzugeben AWS Lambda.	31. Juli 2024
Aktualisierte Funktionalität	Amazon Inspector veröffentlicht neue Sicherheitskontrollen. Weitere Informationen finden Sie unter Amazon Inspector-Steuerelemente im AWS Security Hub Benutzerhandbuch.	11. Juli 2024
Aktualisierte Funktionalität	Der Amazon Inspector SBOM Generator scannt jetzt Dockerfiles und Docker-Container-Images auf Fehlkonfigurationen, die zu Sicherheitslücken führen können. Weitere Informationen finden Sie unter Amazon Inspector Dockerfile-Checks.	10. Juni 2024

Aktualisierte Funktionalität	Amazon Inspector aktualisiert seine CI/CD-Integrationsfunktion, um CodeCatalyst Aktionen zu unterstützen, sodass Sie Amazon Inspector Inspector-Schwachstellenscans zu Ihren CodeCatalyst Workflows hinzufügen können. Weitere Informationen finden Sie unter Aktionen verwenden CodeCatalyst .	7. Juni 2024
Aktualisierte Funktionalität	Amazon Inspector bietet eine Option zum Herunterladen einer CSV-Datei mit CIS-Scanergebnissen. Weitere Informationen finden Sie unter CIS-Scanergebnisse anzeigen und herunterladen in Center for Internet Security (CIS) - Scans für EC2 Amazon-Instances.	3. Mai 2024
Aktualisierte Funktionalität	Amazon Inspector aktualisiert seine CI/CD-Integrationsfunktion zur UnterstützungGitHub Actions, sodass Sie Amazon Inspector Inspector-Schwachstellenscans zu Ihren GitHub Workflows hinzufügen können. Weitere Informationen finden Sie unter Amazon Inspector verwenden mit GitHub Actions.	29. April 2024

Aktualisierte Funktionalität	Amazon Inspector aktualisiert die verwaltete Richtlinie `AmazonInspector2FullAccess`, sodass die serviceverknüpfte Rolle `AWSServiceRoleForAmazonInspector2Agentless` erstellt wird. Auf diese Weise können Benutzer agentenbasiertes Scannen und agentenloses Scannen durchführen, wenn sie Amazon Inspector aktivieren.	24. April 2024
Aktualisierte Funktionalität	Amazon Inspector aktualisiert die Aufbewahrungsfrist für abgeschlossene Ergebnisse von 30 Tagen auf 7 Tage. Weitere Informationen finden Sie unter Grundlegendes zu den Ergebnissen in Amazon Inspector.	12. Februar 2024
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Erklärung hinzugefügt. Die neue Anweisung ermöglicht es Amazon Inspector, CIS-Scans für Ihre Instance zu starten.	23. Januar 2024

Neue Richtlinie	Amazon Inspector hat eine neue Richtlinie, AmazonInspector2ManagedCisPolicyPolicy, hinzugefügt, die Sie als Teil eines Instance-Profils verwenden können, um CIS-Scans auf einer Instance zuzulassen.	23. Januar 2024
Neue Funktion	Amazon Inspector aktualisiert jetzt die Dauer des ECR-Rescans von Container-Images, wenn Sie sie abrufen. Informationen zum Ändern der Dauer des erneuten Scans auf der Grundlage von Push- oder Pull-Daten finden Sie unter Konfiguration der ECR-Rescan-Dauer.	23. Januar 2024
Neue Funktion	Amazon Inspector kann jetzt Center for Internet Security (CIS) -Scans auf EC2 Instances ausführen. Weitere Informationen finden Sie unter Amazon Inspector CIS-Scans.	23. Januar 2024
Neue Funktion	Amazon Inspector kann jetzt Container-Bilder in Ihren CI/CD Pipelines scannen. Weitere Informationen finden Sie unter CI/CD-Integration mit Amazon Inspector.	30. November 2023

Neue Richtlinie	Amazon Inspector hat eine neue Richtlinie hinzugefügt, die es Amazon Inspector ermöglicht, Amazon EBS-Snapshots von Ihrer EC2 Instance für agentenloses Scannen zu scannen. Weitere Informationen zu dieser Richtlinie finden Sie unter Agentloses Scannen.	8. November 2023
Neue Funktion	Amazon Inspector unterstützt jetzt das Scannen unterstützter EC2 Linux-Amazon-Instances ohne SSM-Agenten durch agentenloses Scannen. Weitere Informationen finden Sie unter Agentloses Scannen.	8. November 2023
Neue unterstützte Ressourcen	Amazon Inspector unterstützt jetzt das Scannen von EC2 macOS-Amazon-Instances. Siehe Unterstützte Betriebssysteme: Amazon EC2 sucht nach unterstützten macOS-Versionen.	05. Oktober 2023
Neue Regionen	Amazon Inspector ist jetzt in Asien-Pazifik (Jakarta), Afrika (Kapstadt), Asien-Pazifik (Osaka) und Europa (Zürich) verfügbar.	29. September 2023
Neues Feature	Sie können jetzt EC2 Instances mithilfe von Ausschluss-Tags von Amazon Inspector-Scans ausschließen.	14. September 2023

Neues Feature	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, Netzwerkkonfigurationen von EC2 Amazon-Instances zu scannen, die Teil der Elastic Load Balancing Balancing-Zielgruppen sind.	31. August 2023
Neues Feature	Amazon Inspector bietet jetzt Informationen zu Sicherheitslücken für gefundene Sicherheitslücken in Paketen.	31. Juli 2023
Aktualisierte Funktionalität	Amazon Inspector hat neue Berechtigungen hinzugefügt, die es Benutzern mit Lesezugriff ermöglichen, Software Bill of Materials (SBOM) für ihre Ressourcen zu exportieren.	29. Juni 2023
Neues Feature	Sie können jetzt SBOM für Ressourcen exportieren, die von Amazon Inspector gescannt werden.	13. Juni 2023

Neues Feature	Das Scannen von Lambda-Code ist jetzt allgemein verfügbar. Es wurden neue Funktionen hinzugefügt, mit denen Sie Code verschlüsseln können, der in Ihren Ergebnissen beim Lambda-Code-Scannen identifiziert wurde. Darüber hinaus bietet das Lambda-Code-Scannen jetzt Vorschläge zur Behebung von Neuschreibungen Ihres Codes.	13. Juni 2023
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ReadOnlyAccessRichtlinie eine neue Erklärung hinzugefügt. Mit den neuen Anweisungen können Benutzer mit Lesezugriff Details zum Status und zu den Ergebnissen des Lambda-Code-Scans für ihr Konto abrufen.	2. Mai 2023
Neues Feature	Amazon Inspector hat die Suche nach Sicherheitslücken in der Datenbank hinzugefügt, mit der Sie überprüfen können, ob Amazon Inspector ein bestimmtes CVE abdeckt.	1. Mai 2023

Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie neue Berechtigungen hinzugefügt, die es Amazon Inspector ermöglichen, AWS CloudTrail serviceverknüpfte Kanäle in Ihrem Konto zu erstellen, wenn Sie Lambda-Scanning aktivieren. Auf diese Weise kann Amazon Inspector CloudTrail Ereignisse in Ihrem Konto überwachen.	30. April 2023
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2FullAccessRichtlinie eine neue Erklärung hinzugefügt. Die neue Erklärung ermöglicht es Benutzern, Details zu den beim Lambda-Code-Scannen gefundenen Sicherheitslücken abzurufen.	17. April 2023
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Erklärung hinzugefügt. Die neue Erklärung ermöglicht es Amazon Inspector, Informationen über die benutzerdefinierten Pfade, die Sie für Amazon EC2 Deep Inspection definiert haben, an Amazon EC2 Systems Manager zu senden.	17. April 2023

Neues Feature	Amazon Inspector bietet zusätzliche Unterstützung für EC2 Linux-Instances in Form von Amazon Inspector Deep Inspection, die Ihre Instances auf Paketschwachstellen in Programmiersprachenpaketen für Anwendungen scannt.	17. April 2023
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Erklärung hinzugefügt. Die neuen Anweisungen ermöglichen es Amazon Inspector, Scans des Entwicklercodes in AWS Lambda Funktionen anzufordern und Scandaten von Amazon CodeGuru Security zu empfangen. Darüber hinaus hat Amazon Inspector Berechtigungen zur Überprüfung von IAM-Richtlinien hinzugefügt. Amazon Inspector verwendet diese Informationen, um Lambda-Funktionen auf Code-Schwachstellen zu überprüfen.	28. Februar 2023
Neues Feature	Amazon Inspector bietet zusätzliche Unterstützung für Lambda-Funktionen in Form von Lambda-Code-Scans, die den Entwicklercode Ihrer Lambda-Funktionen auf Sicherheitslücken scannen.	28. Februar 2023

Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Erklärung hinzugefügt. Die neue Anweisung ermöglicht es Amazon Inspector, Informationen CloudWatch darüber abzurufen, wann eine AWS Lambda Funktion zuletzt aufgerufen wurde. verwendet diese Informationen, um Scans auf die Lambda-Funktionen in Ihrer Umgebung zu konzentrieren, die in den letzten 90 Tagen aktiv waren.	20. Februar 2023
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Erklärung hinzugefügt. Die neue Erklärung ermöglicht es Amazon Inspector, Informationen über Ihre AWS Lambda Funktionen abzurufen. Amazon Inspector verwendet diese Informationen, um Ihre Lambda-Funktionen auf Sicherheitslücken zu überprüfen.	28. November 2022
Neues Feature	Amazon Inspector bietet Unterstützung für AWS Lambda Scanfunktionen.	28. November 2022

Aktualisierter Inhalt	Es wurden Verfahren, Richtlinienbeispiele und Tipps für den Export von Ergebnisberichten aus Amazon Inspector in einen Amazon Simple Storage Service (Amazon S3) -Bucket hinzugefügt.	14. Oktober 2022
Neuer Inhalt	Es wurden Informationen zur Bewertung der Amazon Inspector Inspector-Abdeckung Ihrer AWS Umgebung mithilfe der Amazon Inspector Inspector-Konsole hinzugefügt. Die Informationen umfassen Beschreibungen der Statuswerte für einzelne Ressourcen in Ihrer Umgebung.	7. Oktober 2022

Neues Feature	Amazon Inspector bietet jetzt zusätzliche Informationen zur Behebung von Sicherheitslücken in Paketen. Den Suchdetails wurden neue Felder hinzugefügt. Die neuen Felder geben Aufschluss darüber, ob ein Update im Rahmen eines Paket-Updates verfügbar ist. Wenn ein Update verfügbar ist, werden im Abschnitt „Vorgeschlagene Abhilfemaßnahmen“ eines Ergebnisses die Befehle angezeigt, die Sie ausführen können, um das Problem zu beheben.	02. September 2022
Aktualisierte Funktionalität	Amazon Inspector hat der AmazonInspector2ServiceRolePolicyRichtlinie eine neue Aktion hinzugefügt. Die neue Aktion ermöglicht es Amazon Inspector, SSM-Zuordnungsausführungen zu beschreiben. Amazon Inspector hat außerdem zusätzlichen Ressourcenbereich hinzugefügt, damit Amazon Inspector SSM-Verknüpfungen mit `AmazonInspector2` eigenen SSM-Dokumenten erstellen, aktualisieren, löschen und starten kann.	31. August 2022

Neues Feature	Amazon Inspector unterstützt jetzt Scans für Windows Instances. Amazon Inspector kann jetzt SSM-verwaltete Instances scannen, auf denen unterstützte Windows Betriebssysteme ausgeführt werden. Scans von Windows Hosts werden vom Amazon Inspector SSM-Plugin durchgeführt, das über neue SSM-Verknüpfungen installiert und aufgerufen wird, die automatisch von Amazon Inspector erstellt werden.	31. August 2022
Aktualisierte Funktionalität	Amazon Inspector hat den Ressourcenbereich der AmazonInspector2ServiceRolePolicyRichtlinie aktualisiert, sodass Amazon Inspector Softwareinventar in anderen AWS Partitionen erfassen kann.	12. August 2022
Aktualisierte Funktionalität	In der AmazonInspector2ServiceRolePolicyRichtlinie hat Amazon Inspector den Ressourcenbereich der Aktionen neu strukturiert, sodass Amazon Inspector SSM-Verknüpfungen erstellen, löschen und aktualisieren kann.	10. August 2022

Neues Feature	Amazon Inspector unterstützt jetzt die Änderung Ihrer Einstellung für die Dauer Ihres automatisierten ECR-Rescans. Die Einstellung für die Dauer des automatischen erneuten Scans in Amazon ECR bestimmt, wie lange Amazon Inspector kontinuierlich Bilder überwacht, die in Repositorys übertragen werden. Wenn ein Bild älter als die Scandauer ist, scannt Amazon Inspector das Bild nicht mehr und schließt alle vorhandenen Ergebnisse dafür. Bei allen neuen Konten wird die Dauer des automatischen ECR-Wiederholungsscans automatisch auf „Lebenszeit“ gesetzt. Zuvor erstellte Konten hatten eine Dauer von 30 Tagen für den automatischen ECR-Rescan, aber Sie können jetzt zwischen einer Dauer von 30 Tagen, 180 Tagen oder lebenslangen Scans wählen.	25. Juni 2022

Neue Funktionalität	Amazon Inspector hat eine neue AWS verwaltete Richtlinie hinzugefügt, die den AmazonInspector2ReadOnlyAccess schreibgeschützten Zugriff auf die Funktionen von Amazon Inspector ermöglicht.	21. Januar 2022
Allgemeine Verfügbarkeit	Dies ist die erste öffentliche Version des Amazon Inspector Inspector-Benutzerhandbuchs.	29. November 2021

AWS Glossar

Die neueste AWS Terminologie finden Sie im AWS Glossar in der AWS-Glossar Referenz.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.