Benutzer-Leitfaden

AWS-Sicherheitsagent

AWS-Sicherheitsagent: Benutzer-Leitfaden

Table of Contents

Einführung 1
Was ist AWS Security Agent? 1
Die wichtigsten Funktionen 1
Vorteile 3
Funktionen des AWS Security Agents 4
So funktioniert der AWS Security Agent 5
-Übersicht 5
Verstehen Sie die Ressourcenhierarchie und den Lebenszyklus 9
Was wird in Ihrer Organisation gemeinsam genutzt 9
Was ist pro Agent Space vorgesehen 10
Wie passen GitHub Repositorys in die Hierarchie 12
Hauptunterschiede zwischen den Sicherheitsfunktionen 13
Grundlegendes zu Ressourcenbeziehungen 14
Erste Schritte 16
Melde dich an für AWS 16
Wähle deinen Weg 16
Schnellstart: Führen Sie einen Penetrationstest durch 17
Voraussetzungen 18
Schritt 1: AWS Security Agent in der AWS-Konsole einrichten 18
Schritt 2: Aktivieren Sie Penetrationstests und verifizieren Sie Ihre Domain 19
Schritt 3: Quellcode Connect (optional) 20
Schritt 4: Erstellen Sie einen Penetrationstest und führen Sie ihn durch 20
Schritt 5: Überprüfen Sie die Ergebnisse des Penetrationstests 21
Schnellstart: Führen Sie eine Codeüberprüfung durch 22
Voraussetzungen 18
Schritt 1: AWS Security Agent in der AWS-Konsole einrichten 18
Schritt 2: Aktivieren und konfigurieren Sie die Codeüberprüfung 23
Schritt 3: Einen Code-Review erstellen und ausführen 25
Schritt 4: Überprüfe die Ergebnisse des Code-Reviews 26
Schnellstart: Ein Bedrohungsmodell ausführen 26
Schritt 1: AWS Security Agent in der AWS-Konsole einrichten 18
Schritt 2: Quellcode Connect 28
Schritt 3: Erstellen Sie ein Bedrohungsmodell und führen Sie es aus 28
Schritt 4: Bedrohungen überprüfen 29

Für Administratoren (Konsole) 31
Beispielkonfiguration 31
Agent Spaces einrichten und erstellen 31
AWS Security Agent einrichten 31
Erstellen Sie einen Agentenbereich 37
Integrationen Connect 39
So funktionieren Integrationen mit Agent Spaces 39
AWS Security Agent mit GitHub Repositorys Connect 42
Eine GitHub Integration entfernen 48
AWS Security Agent mit GitLab Repositorys Connect 50
Connect den AWS Security Agent mit GitLab Self-Managed 54
Eine GitLab Integration entfernen 57
AWS Security Agent mit GitHub Enterprise Server Connect 58
Eine GitHub Enterprise Server-Integration entfernen 62
Finde deine Atlassian-Installations-ID 63
AWS Security Agent mit Bitbucket-Repositorys Connect 64
Eine Bitbucket-Integration entfernen 68
AWS Security Agent mit Confluence Connect 69
Eine Confluence-Integration entfernen 73
Connect zur privat gehosteten Quellcodeverwaltung her 74
Agent mit privaten VPC-Ressourcen verbinden 79
Funktionen konfigurieren 83
Penetrationstest aktivieren 83
Pull-Request-Code-Review für GitHub Repositorys aktivieren 89
Codeüberprüfung aktivieren 90
Bedrohungsmodellierung aktivieren 97
Ermöglichen Sie es Benutzern, mit der Behebung der Ergebnisse von Penetrationstests und Code-Reviews zu beginnen 101
Stellen Sie Agentenressourcen aus einem S3-Bucket bereit 103
Aktivieren Sie eine Anwendungsdomäne für Penetrationstests 103
Verwaltete Zieldomänen, die für Penetrationstests verwendet werden 106
Sicherheitsanforderungen verwalten 108
Sicherheitsanforderungen verwalten 108
Von AWS verwaltete Pakete 115
Generieren Sie Sicherheitsanforderungen aus Dokumenten 117
Dokumente für die Generierung von Anforderungen vorbereiten 119

Benutzer und Zugriff verwalten ... 120
Benutzern Zugriff auf die AWS Security Agent-Web-App gewähren ... 121
Eine IAM-Rolle für AWS Security Agent erstellen ... 123
Kundenseitig verwaltete Schlüssel ... 129
Fehlerbehebung ... 155
Zugriff verweigert: Falscher GitHub Kontotyp ausgewählt oder falscher Organisationsname angegeben ... 155
Zugriff verweigert: Unzureichende Berechtigungen, um die GitHub App in der Organisation zu installieren ... 156
Der Agent kann während eines Penetrationstests keine Verbindung zum Endpunkt herstellen ... 157
Weitere Hilfe ... 157
Für Anwender und Entwickler (Web-App und IDE) ... 158
Melden Sie sich bei der AWS Security Agent Web App an ... 158
Zugriffsmethoden ... 158
Melden Sie sich mit IAM Identity Center (SSO) an ... 159
Melden Sie sich mit Administratorzugriff (IAM) an ... 160
Erstellen Sie eine Entwurfsprüfung ... 161
Voraussetzungen ... 18
Schritt 1: Beginnen Sie mit der Erstellung einer Entwurfsprüfung ... 162
Schritt 2: Benennen Sie Ihre Entwurfsprüfung ... 162
Schritt 3: Laden Sie Designdateien hoch ... 162
Schritt 4: Initiieren Sie die Entwurfsprüfung ... 163
Nächste Schritte ... 36
Überprüfen Sie die Ergebnisse einer Entwurfsprüfung ... 164
Erstellen Sie ein Bedrohungsmodell ... 168
Voraussetzungen ... 18
Wie AWS Security Agent ein Bedrohungsmodell ausführt ... 169
Rufen Sie die Seite mit den Bedrohungsmodellen auf ... 170
Erstellen Sie ein Bedrohungsmodell ... 170
Führen Sie ein Bedrohungsmodell aus ... 173
Überwachen Sie die Ausführung eines Bedrohungsmodells ... 173
Bearbeiten Sie ein Bedrohungsmodell ... 175
Löschen Sie ein Bedrohungsmodell ... 175
Nächste Schritte ... 36
Überprüfen Sie Bedrohungen anhand eines Bedrohungsmodells ... 176

Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests 182
Wie funktioniert die Codeüberprüfung bei Pull-Requests 183
Die Ergebnisse von Code-Reviews verstehen 183
Auf Sicherheitsfeststellungen reagieren 184
Filterung der Ergebnisse von Code-Reviews 185
Nächste Schritte 36
Einen Code-Review erstellen 187
Voraussetzungen 18
Rufen Sie die Seite mit den Code-Reviews auf 188
Erstellen Sie einen Code-Review 188
Führen Sie einen Code-Review durch 194
Überwachen Sie einen Code-Review-Lauf 194
Nächste Schritte 36
Ergebnisse eines Code-Reviews überprüfen 196
Korrigieren Sie die Ergebnisse der Codeüberprüfung 204
Führen Sie einen Differenzcodescan mit S3 durch 207
Wie funktionieren Differenzscans 207
Voraussetzungen 18
Schritt 1: Generieren Sie eine Diff-Datei 208
Schritt 2: Laden Sie den Diff auf S3 hoch 209
Schritt 3: Starten Sie einen Job zur Überprüfung des Differentialcodes 209
Schritt 4: Überwachen Sie den Scan 210
Schritt 5: Überprüfen Sie die Ergebnisse 211
Kontingente und -Einschränkungen 211
Nächste Schritte 36
Führen Sie Code-Sicherheitsscans von Ihrer IDE aus aus 212
So funktioniert die IDE-Integration 212
Voraussetzungen 18
Installieren Sie den MCP-Server 214
First-time einrichten 215
Führen Sie einen vollständigen Sicherheitsscan durch 216
Führen Sie einen Differenzscan aus 216
Führen Sie eine Überprüfung des Bedrohungsmodells durch 217
Überprüfen der Erkenntnisse 218
Wenden Sie automatische Korrekturen an 218
Automatische Scanvorschläge (Kiro) 219

Verfügbare Scantypen ... 219
Umgebungsvariablen ... 220
Fehlerbehebung ... 220
Kontingente und -Einschränkungen ... 211
Nächste Schritte ... 36
Erstellen Sie einen Penetrationstest ... 221
Voraussetzungen ... 18
Beginnen Sie mit der Erstellung eines Penetrationstests ... 222
Benennen Sie Ihren Penetrationstest ... 223
Umfang des Penetrationstests konfigurieren ... 223
Konfigurieren Sie die IAM-Rolle ... 226
Automatische Codekorrektur ... 205
VPC-Ressourcen konfigurieren (optional) ... 227
Konfigurieren Sie die Anmeldeinformationen für die Authentifizierung (optional) ... 229
Fügen Sie zusätzliche Ressourcen hinzu (optional) ... 231
Erstellen Sie den Penetrationstest ... 235
Überprüfen Sie die Ergebnisse eines Penetrationstests ... 236
Geben Sie Anmeldeinformationen für Penetrationstests an ... 248
Korrigieren Sie ein Ergebnis eines Penetrationstests ... 254
Sicherheit und Referenz ... 257
Sicherheit ... 257
Sicherheitsüberlegungen ... 258
Von AWS verwaltete Richtlinien ... 265
Datenschutz ... 269
Identity and Access Management ... 272
Vorfallreaktion ... 290
Compliance-Validierung ... 291
Ausfallsicherheit ... 293
Sicherheit der Infrastruktur ... 293
Schnittstellen-VPC-Endpunkte ... 294
Konfigurations- und Schwachstellenanalyse ... 299
Cross-service Prävention von verwirrten Stellvertretern ... 299
Bewährte Methoden für die Gewährleistung der Sicherheit ... 300
IAM-Aktionen und Ressourcenmigration ... 304
Protokollierung mit CloudTrail ... 309
Service Quotas ... 311

Kontingente für Operationen 311
Kontingente für Konfigurationen 312
Dokumentverlauf 312
..... cccxvii

Einführung

Erfahren Sie, was AWS Security Agent macht, wie er funktioniert und wie seine Ressourcen zusammenpassen.

Was ist AWS Security Agent?

AWS Security Agent ist ein Frontier Agent, der Ihre Anwendungen während des gesamten Entwicklungszyklus proaktiv schützt. Er führt automatisierte Sicherheitsüberprüfungen durch, die auf Ihre organisatorischen Anforderungen zugeschnitten sind, erstellt Bedrohungsmodelle, um zu ermitteln, wie Ihre Anwendungen angegriffen werden könnten, und führt bei Bedarf kontextsensitive Penetrationstests durch. Durch die kontinuierliche Überprüfung der Sicherheit vom Design bis zur Bereitstellung trägt der AWS Security Agent dazu bei, Sicherheitslücken in all Ihren Umgebungen frühzeitig zu verhindern.

Sicherheitsteams definieren die organisatorischen Sicherheitsanforderungen einmal in der AWS-Konsole: genehmigte Autorisierungsbibliotheken, Protokollierungsstandards und Datenzugriffsrichtlinien. AWS Security Agent setzt diese Sicherheitsanforderungen während der gesamten Entwicklung automatisch durch, bewertet Architekturdokumente und Code anhand Ihrer Standards und bietet spezifische Hinweise, wenn Verstöße festgestellt werden. Auf diese Weise wird eine konsistente Durchsetzung der Sicherheitsvorkehrungen bei Entwurfs- und Codeprüfungen in allen Teams gewährleistet.

Für die Validierung der Bereitstellung verwandelt der AWS Security Agent Penetrationstests von einem regelmäßigen Engpass in eine On-Demand-Funktion. Sicherheitsteams stellen Ziel-URLs, Authentifizierungsdetails, Quellcode und Anwendungsdokumentation zur Verfügung. AWS Security Agent entwickelt ein tiefes Anwendungsverständnis und führt ausgeklügelte Angriffsketten durch, um Sicherheitslücken zu entdecken und zu validieren, sodass Teams bei Bedarf Tests durchführen können.

Die wichtigsten Funktionen

AWS Security Agent bietet umfassende Sicherheitsfunktionen, die sich über den gesamten Entwicklungszyklus erstrecken.

Sicherheitsüberprüfung entwerfen

Der AWS Security Agent verschiebt die Sicherheit nach links, indem er in Echtzeit Sicherheitsfeedback zu Entwurfsdokumenten gibt und die Einhaltung der organisatorischen Sicherheitsanforderungen bewertet, bevor Code geschrieben wird. Sicherheitsteams laden Dokumente über die Webanwendung hoch und erhalten Anleitungen zur Problembehebung, um die Ergebnisse zu priorisieren, wodurch zeitaufwändige manuelle Überprüfungen schneller zu gezielten Analysen werden. Indem Sie Ihre Sicherheitsstandards proaktiv in jede Entwurfsprüfung integrieren, reduzieren Sie die Anzahl der Architekturüberarbeitungen in der Spätphase und können mit mehreren Entwicklungsteams Schritt halten.

Modellierung von Bedrohungen

AWS Security Agent erstellt Bedrohungsmodelle Ihrer Anwendungen, um zu ermitteln, wie sie angegriffen werden könnten. Stellen Sie technische Entwurfsdokumente (Dokumente zum Umfang) bereit, um zu definieren, worauf sich der Agent konzentriert, oder stellen Sie den Quellcode als Kontext bereit, damit der Agent Ihr bestehendes System verstehen kann, oder beides. AWS Security Agent generiert eine Systemübersicht, in der die Architektur, die Komponenten, die Vertrauensgrenzen, die Datenflüsse und die Sicherheitslage Ihrer Anwendung beschrieben werden, zusammen mit einer Reihe von Bedrohungen, die nach STRIDE-Kategorien (Spoofing, Manipulation, Ablehnung, Offenlegung von Informationen, Denial of Service, Erhöhung von Rechten) mit Schweregrad und umsetzbaren Empfehlungen klassifiziert sind. Bedrohungsmodelle sind wiederverwendbare Konfigurationen, die Sie erneut ausführen können, wenn sich Ihr Code und Ihr Design weiterentwickeln, sodass während des gesamten Entwicklungszyklus iterative Sicherheitsbewertungen möglich sind.

Überprüfung der Code-Sicherheit

AWS Security Agent schützt Anwendungen proaktiv mit zwei sich ergänzenden Ansätzen. Erstens können Sie in der Webanwendung Code-Reviews erstellen, um umfassende Scans Ihres vollständigen Quellcodes aus GitHub, GitLab, Bitbucket- oder GitHub Enterprise Server-Repositorys oder S3-Buckets durchzuführen, Sicherheitslücken zu identifizieren und die Einhaltung Ihrer organisatorischen Anforderungen in Ihrer gesamten Codebasis zu überprüfen. Zweitens können Sie die automatische Pull-Request-Analyse für verbundene Repositorys aktivieren, bei der der AWS Security Agent Codeänderungen überprüft und Sicherheitsergebnisse direkt als Pull Request- oder Merge-Request-Kommentare veröffentlicht. In beiden Fällen erhalten Entwickler Anleitungen zur Problembehebung, und der AWS Security Agent kann automatisch Pull-Anfragen generieren oder Anfragen mit Code-Fixes für identifizierte Sicherheitslücken zusammenführen. Dadurch wird

Sicherheitsexpertise in alle Repositorys integriert, wodurch sicherheitsbedingte Verzögerungen in der Entwicklungspipeline reduziert und die Evaluierung über alle Codebasen hinweg skaliert wird.

Penetrationstests

AWS Security Agent bietet Penetrationstests auf Abruf durch den Einsatz spezialisierter KI-Agenten, um Sicherheitslücken durch maßgeschneiderte mehrstufige Angriffsszenarien zu entdecken, zu validieren, zu melden und zu beheben. Der Agent versteht den Kontext Ihrer Anwendung, indem er Quellcode und Dokumentation analysiert, um Sicherheitslücken zu identifizieren und auszunutzen, die automatisierte Tools für Sicherheitsscans nicht finden können. Er dokumentiert die Ergebnisse anhand von Auswirkungsanalysen, reproduzierbare Angriffspfade und erstellt Pull-Requests mit sofort einsatzbereiten Code-Fixes. Dadurch werden regelmäßige Bewertungen in eine kontinuierliche Validierung umgewandelt, die für alle Anwendungen skaliert wird, anstatt sich nur auf kritische zu beschränken.

Vorteile

On-demand Barrierefreiheit

AWS Security Agent bietet sofortigen Zugriff auf Sicherheitsexpertise. Entwicklungsteams können bei Bedarf Entwurfsprüfungen, Bedrohungsmodelle, Codeanalysen und Penetrationstests durchführen, um dem Tempo moderner Entwicklungszyklen gerecht zu werden und proaktive Sicherheit in jeder Phase zu gewährleisten.

Überprüfen Sie die organisatorischen Anforderungen automatisch

Definieren Sie die Sicherheitsanforderungen Ihres Unternehmens einmal in der AWS-Konsole. AWS Security Agent validiert diese Anforderungen bei jeder Entwurfs- und Code-Sicherheitsprüfung automatisch für alle Anwendungen und stellt so sicher, dass die Teams Ihre spezifischen Anforderungen berücksichtigen und nicht generische Checklisten.

Skalieren Sie Ihre Sicherheitskompetenz

AWS Security Agent skaliert Sicherheitsüberprüfungen entsprechend der Entwicklungsgeschwindigkeit, indem er die Durchsetzung der organisatorischen Sicherheitsanforderungen automatisiert. Konfigurieren Sie Anforderungen zentral, führen Sie umfassende Prüfungen mit Ergebnisanalysen durch und verwalten Sie den Umfang von Penetrationstests in Ihrem gesamten Unternehmen über die AWS-Konsole.

Umsetzbare Korrekturen für bestätigte Sicherheitslücken

AWS Security Agent validiert Sicherheitsergebnisse, die bei Penetrationstests gefunden wurden, durch beweisbasierte Ausnutzung, bietet reproduzierbare Exploit-Pfade, umfassende Auswirkungsanalysen und erstellt Pull-Requests mit sofort einsatzbereiten Fixes in entwicklerfreundlicher Sprache. Dies hilft Teams, sich auf legitime Sicherheitsrisiken mit großer Wirkung zu konzentrieren, ohne Zeit mit Fehlalarmen zu verschwenden.

Unterstützung für Multi- und Hybrid-Clouds

AWS Security Agent ist in AWS-, On-Premise-, Hybrid-, Multicloud- und SaaS-Umgebungen einsetzbar und gewährleistet konsistente Sicherheitsrichtlinien und Tests, unabhängig von Ihrer Infrastruktur- oder Plattformauswahl.

Funktionen des AWS Security Agents

AWS Security Agent bietet während Ihres gesamten Entwicklungszyklus vier zentrale Sicherheitsfunktionen:

- Überprüfung der Entwurfssicherheit — Überprüft Entwurfs- und Architekturdokumente, um Sicherheitsrisiken zu identifizieren. Laden Sie Dokumente über die Webanwendung hoch, und der Service analysiert sie anhand der Sicherheitsanforderungen Ihres Unternehmens. AWS Security Agent bewertet die Einhaltung Ihrer definierten Sicherheitsanforderungen wie genehmigte Autorisierungsbibliotheken, Protokollierungsstandards und Datenzugriffsrichtlinien. Auf diese Weise können Sie unsichere Designs und Richtlinienverstöße frühzeitig im Entwicklungsprozess catch.
- Bedrohungsmodellierung — Erstellt ein Bedrohungsmodell für Ihre Anwendung, um zu ermitteln, wie sie angegriffen werden könnte. Stellen Sie Entwurfsdokumente als Umfangsdokumente bereit, um den Schwerpunkt der Analyse zu definieren, oder als Quellcode als Kontext, damit der Agent Ihr bestehendes System verstehen kann, oder beides. AWS Security Agent generiert eine Systemübersicht, in der die Architektur, die Komponenten, die Vertrauensgrenzen, die Datenflüsse und die Sicherheitslage Ihrer Anwendung sowie eine Reihe von Bedrohungen beschrieben werden, die nach STRIDE-Kategorien (Spoofing, Manipulation, Ablehnung, Offenlegung von Informationen, Denial of Service, Erhöhung von Rechten) mit Schweregraden und umsetzbaren Empfehlungen klassifiziert sind. Jede Bedrohung ist mit Beweisen in Ihrem Quellcode verknüpft.
- Überprüfung der Code-Sicherheit — Analysiert den Code in Ihren Repositorys und S3-Quellen, um Sicherheitslücken und Verstöße gegen organisatorische Sicherheitsanforderungen in allen Sprachen, Frameworks und Architekturen zu identifizieren. Erstellen Sie Code-Reviews in der

Webanwendung, um umfassende Scans Ihres vollständigen Quellcodes durchzuführen, oder aktivieren Sie Pull-Request-Kommentare, um Codeänderungen in automatisch zu überprüfen. GitHub Der AWS Security Agent bietet spezifische Anleitungen zur Problembehebung und kann automatisch Pull-Anfragen mit Code-Fixes für identifizierte Sicherheitslücken generieren. Dies gewährleistet die konsistente Durchsetzung Ihrer Sicherheitsrichtlinien in allen Entwicklungsteams.

- On-demand Penetrationstests — Erkennt, validiert, meldet und behebt Sicherheitslücken in Live-Webanwendungen und APIs mithilfe maßgeschneiderter mehrstufiger Angriffsszenarien. Konfigurieren Sie den Dienst so, dass er über die Webanwendung einen Pentest durchführt, indem Sie den Testumfang, die Authentifizierungsdetails und die Ressourcen angeben. AWS Security Agent entwickelt den Anwendungskontext aus dem bereitgestellten Quellcode und der Dokumentation und führt ausgeklügelte Angriffsketten aus, um ausnutzbare Sicherheitslücken zu identifizieren, die bei statischen Analysen und herkömmlichen Tools übersehen werden. Außerdem bietet er sofort einsatzbereite Code-Fixes und erstellt Pull-Anfragen direkt in Ihrem Code-Repository, sodass Sie Sicherheitslücken noch schneller beheben können.

So funktioniert der AWS Security Agent

AWS Security Agent arbeitet über drei Schnittstellen, um proaktive Anwendungssicherheit während des gesamten Entwicklungszyklus zu gewährleisten. Sicherheitskonfigurationen werden in der AWS-Managementkonsole verwaltet, Sicherheitsüberprüfungen werden in der Security Agent-Webanwendung durchgeführt und die automatische Behebung von Code- und Sicherheitsfeststellungen erfolgt direkt in Code-Repository-Plattformen wie GitHub.

-Übersicht

AWS Security Agent besteht aus drei Hauptkomponenten:

- AWS-Managementkonsole — Agent Spaces konfigurieren, Sicherheitsanforderungen definieren, Penetrationstests konfigurieren, Code-Repositorys verbinden und Benutzerzugriff verwalten
- Security Agent Web Application — Führen Sie Entwurfsprüfungen durch, erstellen und starten Sie Bedrohungsmodelle, führen Sie Penetrationstests durch, erstellen und führen Sie Codeprüfungen durch und überprüfen Sie die Sicherheitsergebnisse in den Ihnen zugewiesenen Agent Spaces
- Code-Repository-Integration — Erhalten Sie automatische Code-Reviews für Pull-Anfragen und Pull-Anfragen zur Behebung von Penetrationstests. Derzeit unterstützt AWS Security Agent die Verbindung zu GitHub.

Sie arbeiten mit dem AWS Security Agent in einer von drei Rollen, die Sie anhand der von Ihnen verwendeten Schnittstelle identifizieren können:

Rolle	Wo Sie arbeiten und was Sie tun
Administrator	Funktioniert in der AWS-Managementkonsole — richtet Agent Spaces ein, definiert Sicherheitsanforderungen, konfiguriert Funktionen und verwaltet den Benutzerzugriff.
Nutzer	Funktioniert in der Security Agent Webanwendung — führt Entwurfsp rüfungen, Bedrohungsmodelle, Penetrationstests und Codeprüfungen durch und überprüft die daraus resultierenden Ergebnisse.
Entwickler	Funktioniert in GitHub oder einer IDE (wie Kiro oder Claude Code) — empfängt automatisierte Sicherheitserkenntnisse bei Pull-Requests und führt Codescans durch, ohne die Entwicklungsumgebung zu verlassen.

Wir verwenden diese Rollennamen durchweg, um anzugeben, wer die einzelnen Aufgaben ausführt. Eine einzelne Person kann mehr als eine Rolle innehaben.

Konfiguration der Konsole

Administratoren konfigurieren den AWS Security Agent über die AWS-Managementkonsole.

Agent Spaces — Wenn Sie Ihren ersten Agent Space in der AWS-Managementkonsole erstellen, erstellt der AWS Security Agent die Webanwendung für Ihr Konto. Jeder Agent Space, den Sie erstellen, steht für eine bestimmte Anwendung oder ein Projekt, das Sie sichern möchten. In der Webanwendung wählen Benutzer aus, in welchem Agent Space sie arbeiten möchten, wenn sie Sicherheitsbewertungen durchführen.

Sicherheitsanforderungen — Definieren Sie die Sicherheitsanforderungen, die der AWS Security Agent bei Design- und Codeprüfungen bewertet, organisiert in Sicherheitsanforderungspaketen. Aktivieren Sie AWS-managed Pakete, die auf Industriestandards basieren, erstellen Sie benutzerdefinierte Pakete für die Richtlinien Ihres Unternehmens oder generieren Sie Anforderungen, indem Sie Ihre Sicherheitsdokumentation hochladen. Die Anforderungen gelten für alle Agent Spaces.

Konfiguration von Penetrationstests — Konfigurieren Sie die Funktionen für Penetrationstests für jeden Agent Space wie folgt:

- Überprüfung der Zieldomänen für Tests durch DNS- oder HTTP-Überprüfung
- Konfiguration des VPC-Zugriffs zum Testen privater Anwendungen
- Einrichtung der CloudWatch Protokollierung für Penetrationstestläufe
- Konfiguration von AWS Secrets Manager- oder Lambda-Funktionen für Testanmeldedaten
- Spezifizierung von S3-Buckets für zusätzlichen Anwendungskontext

Code-Review-Konfiguration — Konfigurieren Sie die Code-Review-Funktionen für jeden Agent Space wie folgt:

- GitHub Repositorys oder S3-Buckets verbinden, die Quellcode enthalten
- Auswahl der Einstellungen für die Codeüberprüfung (Sicherheitslücken, benutzerdefinierte Anforderungen oder beides)
- Aktivierung von Pull-Request-Kommentaren zur automatisierten Überprüfung von Codeänderungen in GitHub
- Einrichtung der CloudWatch Protokollierung für Code-Review-Läufe

Konfiguration der Bedrohungsmodellierung — Konfigurieren Sie die Funktionen zur Bedrohungsmodellierung für jeden Agent Space wie folgt:

- Verbindung von Quellcode-Repositorys oder S3-Buckets, die Quellcode enthalten
- Hinzufügen von Scope-Dokumenten (Feature-Design-Dokumenten), um die Analyse auf ein bestimmtes Feature zu konzentrieren
- Konfiguration der CloudWatch Protokollierung für Läufe von Bedrohungsmodellen
- Einrichtung einer Servicerolle für den AWS-Ressourcenzugriff

Integrationen — Connect GitHub Repositorys mit jedem Agent Space, um wichtige Funktionen zu aktivieren:

- Stellen Sie Anwendungskontext für genauere Penetrationstests bereit
- Aktivieren Sie die Codeüberprüfung für vollständige Repository-Scans und Pull-Request-Analysen
- Ermöglichen Sie die automatische Codekorrektur durch Pull-Anfragen zur Überprüfung des Codes und der Ergebnisse von Penetrationstests

Benutzerzugriff — Steuern Sie, wie Benutzer auf die Security Agent Webanwendung zugreifen. Wenn Sie IAM Identity Center (SSO) aktiviert haben, weisen Sie Benutzern in der AWS Security Agent-Konsole direkten SSO-Zugriff auf die Webanwendung zu. Wenn Sie IAM-only Access verwenden, können Benutzer mit AWS-Konsolenzugriff die Webanwendung über den Administratorzugriffslink in der Konsole für jeden Agent Space starten.

Aktivitäten der Webanwendung

Benutzer greifen auf die Security Agent Webanwendung zu, um Sicherheitsbewertungen in den ihnen zugewiesenen Agent Spaces durchzuführen.

Agent Space auswählen — Bei der Anmeldung bei der Webanwendung wählen Benutzer aus, in welchem Agent-Space sie arbeiten möchten. Benutzer können nur Agent Spaces sehen und darauf zugreifen, denen sie zugewiesen wurden.

Entwurfsprüfungen — Laden Sie Entwurfsdokumente und Architekturspezifikationen hoch, um sie anhand der Sicherheitsanforderungen des Unternehmens zu analysieren. Überprüfen Sie die Ergebnisse mit Anleitungen zur Problembehebung.

Bedrohungsmodelle — Erstellen Sie Bedrohungsmodelle und führen Sie sie aus, indem Sie Quellcode, technische Entwurfsdokumente (Scope Docs) oder beides bereitstellen. In den Scope-Dokumenten wird definiert, worauf der Agent seine Analyse konzentriert. Der Quellcode bietet den Kontext zu Ihrem bestehenden System. Bei jedem Durchlauf erhalten Sie eine Systemübersicht, in der die Architektur, die Vertrauensgrenzen, die Datenflüsse und die Sicherheitslage Ihrer Anwendung beschrieben werden, sowie eine Reihe von Bedrohungen, die nach STRIDE-Kategorien klassifiziert sind, mit Schweregraden und umsetzbaren Empfehlungen.

Code-Reviews — Erstellen und führen Sie Code-Reviews durch, die eine umfassende statische Analyse Ihres gesamten Quellcodes durchführen. Wählen Sie GitHub Repositorys oder S3-Quellen aus, konfigurieren Sie die Scaneinstellungen und überprüfen Sie detaillierte Sicherheitsergebnisse mit Anleitungen zur Problembehebung. AWS Security Agent identifiziert Sicherheitslücken und überprüft die Einhaltung der individuellen Sicherheitsanforderungen Ihres Unternehmens in Ihrer gesamten Codebasis.

Penetrationstests — Konfigurieren und führen Sie Penetrationstests durch, indem Sie Ziel-URLs, Authentifizierungsdetails und Dokumentation bereitstellen. AWS Security Agent führt autonome Tests durch, um ausnutzbare Sicherheitslücken in mehrstufigen Angriffsszenarien zu entdecken.

Ergebnisse überprüfen — Untersuchen Sie detaillierte Sicherheitsergebnisse aus Entwurfsprüfungen, Bedrohungsmodellen, Codeprüfungen und Penetrationstests, einschließlich Folgenanalysen, reproduzierbarer Angriffspfade und Anleitungen zur Problembehebung.

Validierung von Fixes — Re-run Sicherheitsbeurteilungen nach der Implementierung von Abhilfemaßnahmen, um zu überprüfen, ob Sicherheitslücken behoben wurden.

GitHub Integration

AWS Security Agent lässt sich direkt in die Workflows der Entwickler integrieren, GitHub um automatisiertes Sicherheitsfeedback bereitzustellen.

Kommentare zu Pull-Anfragen — Nachdem Administratoren die AWS Security Agent GitHub App installiert und die Codeüberprüfung mit Code-Review-Kommentaren für einen Agent Space aktiviert haben, analysiert AWS Security Agent automatisch Pull-Anfragen in verbundenen Repositorys. Entwickler erhalten in Form von Pull-Request-Kommentaren direkt Sicherheitserkenntnisse und Anleitungen zur Problembehebung, sodass Codeänderungen anhand der unternehmensinternen Sicherheitsanforderungen und häufig auftretenden Sicherheitslücken validiert werden.

Automatische Behebung — Wenn Benutzer die automatische Codekorrektur für eine Codeüberprüfung aktivieren, generiert der AWS Security Agent Fixes für identifizierte Sicherheitslücken und sendet Pull-Anfragen an die zugehörigen Repositorys. GitHub

Behebung von Penetrationstests — Wenn Administratoren die Suche nach Problembehebungen in der Konsole aktivieren, können Benutzer in der Webanwendung eine automatische Behebung der Ergebnisse von Penetrationstests anfordern. Der AWS Security Agent öffnet eine Pull-Anfrage an das zugehörige GitHub Repository mit Code-Fixes zur Behebung der Sicherheitsanfälligkeit.

Verstehen Sie die Ressourcenhierarchie und den Lebenszyklus

AWS Security Agent organisiert Ressourcen für Sicherheitstests in einer hierarchischen Struktur, die festlegt, was in Ihrer Organisation gemeinsam genutzt wird und welcher Umfang pro Anwendung gilt. Wenn Sie diese Struktur verstehen, können Sie AWS Security Agent effektiv konfigurieren und wissen, wo Sie verschiedene Ressourcen finden und verwalten können.

Was wird in Ihrer Organisation gemeinsam genutzt

Einige Ressourcen in AWS Security Agent werden einmalig auf Organisationsebene konfiguriert und gelten für alle Ihre Anwendungen und Agent Spaces. Diese Ressourcen auf Mandantenebene sorgen für Konsistenz und reduzieren den doppelten Konfigurationsaufwand.

Ressource	Darum geht es	Warum wird es geteilt
Sicherheitsanforderungen	Organisatorische Sicherheitsstandards, die definieren, was der AWS Security Agent bei Design- und Code-Reviews validiert	Ihre Sicherheitsrichtlinien gelten für alle Anwendungen. Definieren Sie sie einmal und der AWS Security Agent setzt sie überall durch.
GitHub Integrationen	Registrierte GitHub Organisationen oder Benutzerkonten, die autorisiert sind, sich mit dem AWS Security Agent zu verbinden	Registrieren Sie Ihre GitHub Organisation einmal und verbinden Sie dann bei Bedarf bestimmte Repositorys mit einem beliebigen Agent Space.
IAM Identity Center-Konfigurationen	SSO-Einstellungen, die steuern, wie Benutzer auf AWS Security Agent zugreifen	Das zentrale Identitätsmanagement gilt für alle Agent Spaces in Ihrer Organisation.

> **Important**
>
> Änderungen der Sicherheitsanforderungen wirken sich auf alle future Design- und Code-Reviews in allen Agent Spaces aus. Bestehende Bewertungen sind davon nicht betroffen.

Was ist pro Agent Space vorgesehen

Jeder Agent Space steht für eine bestimmte Anwendung oder ein bestimmtes Projekt, das Sie schützen möchten. Ressourcen auf Agent Space-Ebene sind auf diese spezifische Anwendung beschränkt, sodass verschiedene Teams unabhängig voneinander an ihren eigenen Konfigurationen und Bewertungen arbeiten können.

Ressource	Darum geht es	Warum ist der Umfang pro Anwendung festgelegt
Konfigurationen für Penetrationstests	Testen Sie Konfigurationen für bestimmte Funktionen, API-Endpu	Jede Anwendung hat eigene Ziele, Authentifizierungsmethoden und

Ressource	Darum geht es	Warum ist der Umfang pro Anwendung festgelegt
	nkte oder Funktionen in Ihrer Anwendung	Anwendungsgrenzen, die für diese Anwendung spezifisch sind.
Entwurfsp rüfungen	Individuelle architektonische Sicherheitsbeurteilungen von Entwurfsdokumenten	Jede Anwendung hat ihre eigenen Architektur- und Entwurfsdokumente, die unabhängig bewertet werden.
Bedrohung smodelle	Analysen zur Bedrohungsmodellie rung, die einen Systemüberblick bieten und Bedrohungen anhand von Quellcode, Entwurfsdokumenten oder beidem identifizieren	Jede Anwendung hat ihren eigenen Code und ihr eigenes Design und wird unabhängig voneinander als Bedrohung modelliert. Bedrohung smodelle sind wiederverwendbare Konfigurationen, die Sie erneut ausführen können, wenn sich Ihr Code und Ihr Design weiterent wickeln.
Integrationen	Quell- und Dokumentationsanbieter (GitHub, GitLab, Bitbucket, GitHub Enterprise Server und Confluenc e), die mit diesem Agent Space verbunden sind	Verschiedene Anwendungen basieren auf unterschiedlichen Quellen und Dokumentationen. Wenn sie auf Agent Space-Ebene miteinander verbunden werden, bleiben die Anwendungsgrenzen klar.
Einstellungen für die Codeüberp rüfung	Konfiguration der Funktionen zur Codeüberprüfung, einschließlich verbundener Quellen, Scaneinst ellungen und Aktivierung von PR-Kommentaren	Jede Anwendung hat ihre eigenen Repositorys und Sicherheitsüberprü fungen werden unabhängig voneinander konfiguriert.
Einstellungen zur Behebung von Penetrati onstests	Konfiguration, welche verbundenen Repositorys automatische Pull-Anfr agen zur Behebung der Ergebniss e von Penetrationstests erhalten können	Die Teams kontrollieren, wo der AWS Security Agent Codeänderungen einreichen kann, basierend auf dem Workflow ihrer Anwendung.

Ressource	Darum geht es	Warum ist der Umfang pro Anwendung festgelegt
Benutzerzuweisungen	Benutzer, die Zugriff auf diesen speziellen Agentenbereich haben	Teams sehen nur Sicherheitsbeurteilungen für Anwendungen, für die sie verantwortlich sind, sodass die Arbeit organisiert und konzentriert bleibt.

> **Tip**
>
> Wir empfehlen, einen Agentenbereich pro Anwendung oder Projekt einzurichten, um klare Grenzen zwischen den Teams zu wahren und Sicherheitsbeurteilungen effektiv zu organisieren.

Wie passen GitHub Repositorys in die Hierarchie

GitHub Repositorys werden in einem mehrstufigen Prozess integriert, der Unternehmensressourcen mit bestimmten Anwendungen verbindet:

1. Registrieren Sie sich auf Mandantenebene — Autorisieren Sie die AWS Security Agent GitHub App für Ihre GitHub Organisation oder Ihr Benutzerkonto einmalig
2. Auf Agent Space-Ebene Connect — Wählen Sie bestimmte Repositorys aus, um eine Verbindung zu jedem Agent Space herzustellen
3. Nutzung pro Repository konfigurieren — Aktivieren Sie spezifische Funktionen für jedes verbundene Repository:
   - Codeüberprüfung — Vollständiges Scannen des Quellcodes und automatisierte Pull-Request-Analyse
   - Kontext für Penetrationstests — Verständnis der Anwendung anhand des Quellcodes bei Penetrationstests
   - Automatische Codekorrektur — Automatisierte Pull-Requests mit Behebung von Sicherheitslücken für die Überprüfung von Code und die Ergebnisse von Penetrationstests

Ein einzelnes Repository kann mit mehreren Agent Spaces verbunden werden, wobei in jedem Bereich unterschiedliche Funktionen aktiviert sind.

Hauptunterschiede zwischen den Sicherheitsfunktionen

Jede Sicherheitsfunktion in AWS Security Agent folgt einem anderen Workflow-Modell, das darauf basiert, wie Sicherheitsteams sie verwenden.

Penetrationstests: Wiederverwendbare Konfigurationen mit unabhängigen Ausführungen

Penetrationstests verwenden ein Configuration-and-Run-Modell, das iterative Sicherheitstests unterstützt:

- Einmal erstellen, mehrfach ausführen — Definieren Sie eine Konfiguration für ein bestimmtes Ziel (API-Endpunkt, Funktionsbereich) mit Bereichsgrenzen, Authentifizierungs- und Testparametern
- Unabhängige Ausführungen — Führen Sie dieselbe Konfiguration mehrmals aus, um die Sicherheit zu verbessern. Jede Ausführung ist unabhängig und generiert neue Erkenntnisse

Dieses Modell unterstützt die kontinuierliche Sicherheitsvalidierung bei der Entwicklung und Implementierung von Verbesserungen.

One-off Entwurfsprüfungen: Bewertungen durch Klonen

Entwurfsprüfungen sind unabhängige Bewertungen, die keinem wiederverwendbaren Konfigurationsmodell folgen:

- Einmalige Bewertung — Bei jeder Entwurfsprüfung werden hochgeladene Dokumente einmal anhand der Sicherheitsanforderungen Ihres Unternehmens analysiert
- Kann nicht erneut ausgeführt werden — Entwurfsprüfungen können nicht wiederverwendet werden. Sie können dieselbe Bewertung nicht erneut ausführen
- Für Aktualisierungen klonen — Klonen Sie eine bestehende Entwurfsprüfung, um eine neue Prüfung zu erstellen, bei der die Originaldokumente vorinstalliert sind. So können Sie Dokumente aktualisieren und eine neue Analyse durchführen

Dieses Modell unterstützt architektonische Sicherheitsbeurteilungen zu einem bestimmten Zeitpunkt.

Code-Reviews: Wiederverwendbare Konfigurationen mit On-Demand-Scans und automatischer PR-Analyse

Code-Reviews bieten zwei Betriebsmodi zur Sicherung Ihres Quellcodes:

- Vollständige Code-Reviews (Webanwendung) — Erstellen Sie Code-Review-Konfigurationen, die GitHub Repositorys oder S3-Quellen auswählen, und führen Sie dann bei Bedarf umfassende Scans durch. Bei jedem Durchlauf wird Ihr vollständiger Quellcode statisch analysiert und Ergebnisse mit Anleitungen zur Problembehebung generiert. Sie können dieselbe Konfiguration für die Codeüberprüfung erneut ausführen, während sich Ihr Code weiterentwickelt.
- Pull-Request-Kommentare (GitHub) — Aktiviert die automatische Analyse für verbundene GitHub Repositorys. AWS Security Agent überprüft automatisch Pull-Requests, wenn sie als zur Überprüfung bereit markiert sind, und veröffentlicht Sicherheitsergebnisse als Kommentare direkt in GitHub.

Beide Modi verwenden Ihre konfigurierten Code-Review-Einstellungen (Sicherheitslücken, benutzerdefinierte Anforderungen oder beides) und unterstützen die automatische Codebehebung durch Pull-Requests.

Bedrohungsmodelle: Wiederverwendbare Konfigurationen mit On-Demand-Läufen

Bedrohungsmodelle verwenden ein Modell mit Konfiguration und Ausführung, das eine iterative Bewertung Ihrer Architektur unterstützt:

- Einmal erstellen, mehrfach ausführen — Definieren Sie ein Bedrohungsmodell, indem Sie Quellcode als Quellen auswählen, Entwurfsdokumente als Bereichsdokumente hochladen oder beides. Führen Sie es bei Bedarf aus und führen Sie es erneut aus, wenn sich Ihr Code und Ihr Design weiterentwickeln.
- Flexible Eingaben — Führen Sie ein Bedrohungsmodell nur für Quellcode, nur für Designdokumente oder für beides aus. In den Scope-Dokumenten wird definiert, worauf der Agent seine Analyse konzentriert. Der Quellcode bietet den Kontext zu Ihrem bestehenden System.
- Systemübersicht und Bedrohungen — Bei jedem Durchlauf erhalten Sie eine Systemübersicht, in der die Architektur, die Vertrauensgrenzen, die Datenflüsse und die Sicherheitslage Ihrer Anwendung beschrieben werden. Außerdem wird eine Reihe von Bedrohungen erstellt, die nach STRIDE-Kategorien mit Schweregrad, Nachweisen und umsetzbaren Empfehlungen klassifiziert sind.

Grundlegendes zu Ressourcenbeziehungen

Die Hierarchie bestimmt, wo Sie verschiedene Ressourcen konfigurieren und darauf zugreifen:

In der AWS-Managementkonsole:

- Konfigurieren Sie Ressourcen auf Mandantenebene (Sicherheitsanforderungen, GitHub Integrationen, IAM Identity Center)
- Agent Spaces erstellen und verwalten
- Konfigurieren Sie die Agent Space-Einstellungen (verbundene Repositorys, Aktivierung von Code-Reviews, Behebung von Penetrationstests)

In der Security Agent Webanwendung:

- Penetrationstest-Konfigurationen und Testausführungen erstellen und verwalten
- Erstellen und verwalten Sie Designprüfungen
- Erstellen, verwalten und führen Sie Codeprüfungen anhand verbundener Repositorys und S3-Quellen durch
- Erstellen, verwalten und führen Sie Bedrohungsmodelle anhand von Quellcode, Scope-Dokumenten oder beidem aus
- Sehen Sie sich Ergebnisse aus Penetrationstests, Codeprüfungen, Entwurfsprüfungen und Bedrohungsmodellen an

In GitHub:

- Ergebnisse der Überprüfung des Pull-Request-Codes als Pull-Request-Kommentare anzeigen
- Erhalten Sie automatische Pull-Requests zur Problembehebung für die Ergebnisse von Code-Reviews und Penetrationstests (sofern diese Option im Agent Space aktiviert ist)

> **Note**
>
> Die Ergebnisse der Pull-Request-Codeüberprüfung werden unter angezeigt. GitHub Die Ergebnisse der vollständigen Codeüberprüfung, des Penetrationstests und der Entwurfsprüfung werden in der Security Agent Webanwendung angezeigt.

Erste Schritte

Melden Sie sich an, finden Sie den richtigen Ausgangspunkt für Ihr Ziel und führen Sie Ihr erstes Assessment mit einem Schnellstart durch.

Melde dich an für ein AWS Konto

Um loszulegen AWS, benötigen Sie ein AWS Konto. Informationen zum Erstellen eines AWS Kontos finden Sie unter Erste Schritte mit einem AWS Konto im Referenzhandbuch zur AWS Kontoverwaltung.

Wähle deinen Weg

Finden Sie den richtigen Ausgangspunkt für das, was Sie mit AWS Security Agent tun möchten. Verwenden Sie die Tabelle unten, um Ihr Ziel einer Aufgabe zuzuordnen, und wechseln Sie dann direkt zu dieser Seite. Wenn Sie noch nicht mit AWS Security Agent vertraut sind, lesen Sie the section called “Was ist AWS Security Agent?” zuerst, um sich einen kurzen Überblick zu verschaffen.

Ich möchte...	Wer	Fangen Sie hier an
Verstehen Sie, was der AWS Security Agent tut, bevor Sie ihn einrichten	Alle	the section called “Was ist AWS Security Agent?”
Richten Sie den Service ein und erstellen Sie einen Agentenbereich	Admin.	the section called “AWS Security Agent einrichten”
Testen Sie meine aktive oder bereitgestellte Anwendung auf ausnutzbare Sicherheitslücken	Benutzer	the section called “Schnellstart: Führen Sie einen Penetrationstest durch”
Scanne meinen Quellcode auf Sicherheitslücken und Richtlinienverstöße	Benutzer	the section called “Schnellstart: Führen Sie eine Codeüberprüfung durch”
Modellieren Sie, wie meine Anwendung angegriffen werden könnte (STRIDE)	Benutzer	the section called “Schnellstart: Ein Bedrohungsmodell ausführen”

Ich möchte...	Wer	Fangen Sie hier an
Holen Sie sich Sicherheitsfeedback zu einem Design, bevor ich Code schreibe	Benutzer	the section called “Erstellen Sie eine Entwurfsprüfung”
Scanne Code, ohne meine IDE zu verlassen (Kiro oder Claude Code)	Developer	the section called “Führen Sie Code-Sicherheitsscans von Ihrer IDE aus aus”
Scanne vor dem Zusammenführen nur meine geänderten Zeilen	Developer	the section called “Führen Sie einen Differenzcodescan mit S3 durch”
Erhalte automatische Sicherhei tskommentare zu Pull Requests und Merge-Requests	Admin.	the section called “Codeüberprüfung aktivieren”
Überprüfe die Ergebnisse und entscheide, was zuerst behoben werden soll	Benutzer	Penetrationstest, Codeüberprüfung, Bedrohungsmodell, Entwurfsprüfung

> **Note**
>
> AWS Security Agent verwendet drei Rollen, die Sie anhand der Oberfläche identifizieren können, in der Sie arbeiten: Der Administrator arbeitet in der AWS-Managementkonsole (Einrichtung und Konfiguration), der Benutzer arbeitet in der Webanwendung (führt Bewertungen aus und überprüft die Ergebnisse) und der Entwickler arbeitet in GitHub oder einer IDE wie Kiro oder Claude Code. Eine einzelne Person kann mehr als eine Rolle innehaben. Vollständige Definitionen finden Sie unterthe section called “So funktioniert der AWS Security Agent”.

Schnellstart: Führen Sie einen Penetrationstest durch

Dieser Schnellstart führt Sie durch die Durchführung Ihres ersten Penetrationstests (Pentest) mit dem AWS Security Agent. Bei einem Penetrationstest wird Ihre bereitgestellte Anwendung anhand einer verifizierten Zieldomäne getestet und es werden Sicherheitsergebnisse zurückgegeben, die jeweils mit einem Schweregrad und Belegen versehen sind. Es behandelt eine öffentlich zugängliche

Anwendung. Informationen zum Testen einer Anwendung, die in einer privaten VPC gehostet wird, finden Sie unterthe section called “Penetrationstest aktivieren”.

> **Note**
>
> Sie benötigen Zugriff auf die AWS-Managementkonsole, um den AWS Security Agent einzurichten und den Testumfang zu definieren, sowie Zugriff auf die Webanwendung, um Penetrationstests zu erstellen und auszuführen.

Voraussetzungen

Bevor Sie beginnen, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Zugriff auf die AWS-Managementkonsole mit Berechtigungen zur Einrichtung des AWS Security Agents.
- Eine Live-Zieldomain, die die Anwendung hostet, die Sie testen möchten.
- Die Möglichkeit, einen DNS-Eintrag für diese Domain oder eine Route 53 53-Hosting-Zone in demselben AWS-Konto hinzuzufügen, sodass Sie die Inhaberschaft überprüfen können.
- (Optional) Ein Quellcode-Repository (GitHub GitLab, oder Bitbucket) für Ihre Anwendung, um dem Agenten mehr Kontext zu geben.

Schritt 1: AWS Security Agent in der AWS-Konsole einrichten

Wenn Sie AWS Security Agent noch nicht eingerichtet haben, schließen Sie die Ersteinrichtung ab:

1. Navigieren Sie in der AWS-Managementkonsole zu AWS Security Agent.
2. Wählen Sie AWS Security Agent einrichten aus.
3. Erstellen Sie einen Agentenbereich. Ein Agentenbereich kann von mehreren Benutzern verwendet werden und sollte für jede Anwendung, die Sie testen möchten, spezifisch sein. Geben Sie einen Namen und eine Beschreibung ein. Der Name sollte die Anwendung identifizieren, für die Sie einen Penetrationstest durchführen möchten.
4. Wählen Sie IAM-only unter Konfiguration des Benutzerzugriffs die Option Zugriff aus.
   - Dieser Schnellstart behandelt nicht die Aktivierung von Single Sign-On (SSO) mit IAM Identity Center, wodurch Benutzer direkt von der AWS-Konsole aus auf die Webanwendung zugreifen können.

- Damit Benutzer ohne Zugriff auf die AWS-Managementkonsole Penetrationstests starten können, aktivieren Sie die IAM Identity Center-Integration. Details hierzu finden Sie unter the section called “Benutzern Zugriff auf die AWS Security Agent-Web-App gewähren”.

5. Wählen Sie „AWS Security Agent einrichten“.

> **Note**
>
> Wenn Sie Setup wählen, erstellt AWS Security Agent Ihren Agent Space und richtet eine Webanwendung ein, in der Benutzer Penetrationstests, Codeprüfungen, Bedrohungsmodelle und Entwurfsprüfungen durchführen können.

Schritt 2: Aktivieren Sie Penetrationstests und verifizieren Sie Ihre Domain

> **Note**
>
> In der AWS-Konsole definieren Sie den Umfang dessen, was getestet werden kann. Benutzer führen dann in der Webanwendung spezifische Penetrationstests innerhalb dieses Bereichs durch.

1. Wählen Sie in der linken Seitenleiste Agent Spaces und dann den Agent Space aus, den Sie in Schritt 1 erstellt haben.
2. Wählen Sie die Registerkarte Penetrationstest und anschließend Penetrationstest einrichten, um den Assistenten zu öffnen.
3. Domain konfigurieren — Geben Sie die Zieldomain ein, die Sie testen möchten, und wählen Sie eine Bestätigungsmethode, einen DNS-TXT-Eintrag oder eine HTTP-Route aus. Die Domain muss live sein und die Anwendung hosten, die Sie testen möchten. Wählen Sie Weiter aus.
4. Domänen verifizieren — Überprüfen Sie die Inhaberschaft jeder Domain in der Tabelle Zieldomänen. AWS Security Agent führt Tests nur für verifizierte Domains durch. Eine detaillierte Anleitung und die genauen Werte, die kopiert werden müssen, finden Sie unterthe section called “Aktivieren Sie eine Anwendungsdomäne für Penetrationstests”.
   - 53 Domains im selben AWS-Konto weiterleiten — Wählen Sie die Domain aus und wählen Sie One-click Verifizierung. Der AWS Security Agent erstellt den DNS-Eintrag und schließt die Überprüfung für Sie ab.

- DNS-TXT-Eintrag (andere DNS-Anbieter) — Kopieren Sie in der Tabelle Zieldomänen den Eintragsnamen und den Datensatzwert, fügen Sie sie als TXT-Eintrag bei Ihrem DNS-Anbieter hinzu, wählen Sie dann die Domain aus und wählen Sie Verify. Die Übertragung von DNS-Änderungen kann einige Zeit in Anspruch nehmen, sodass die Überprüfung möglicherweise nicht sofort abgeschlossen wird.
- HTTP-Route — Erstellen Sie eine Datei `.well-known/aws/securityagent-domain-verification.json` auf Ihrem Webserver, fügen Sie das Verifizierungstoken im Format hinzu`{"tokens": ["<token>"]}`, wählen Sie dann die Domain aus und wählen Sie Verifizieren.

5. (Optional) Konfigurieren Sie zusätzliche Funktionen — Fügen Sie optionale Ressourcen wie CloudWatch Protokollgruppen und Anmeldeinformationen hinzu. Der Abschnitt Servicezugriff ist vorkonfiguriert: AWS Security Agent erstellt eine Servicerolle mit den erforderlichen Berechtigungen, sofern Sie keine bestehende IAM-Rolle wählen.

Schritt 3: Quellcode Connect (optional)

Durch die Verbindung mit einem Quellcode-Anbieter erhält der AWS Security Agent einen Überblick über Ihre Anwendung und verbessert die Reichweite von Penetrationstests. Dieser Schritt ist optional.

1. Wählen Sie auf der Registerkarte Penetrationstest im Banner Einen Quellcode-Anbieter für Penetrationstests Connect oben auf der Seite die Option Hinzufügen aus.
2. Registrieren Sie sich und stellen Sie eine Verbindung zu Ihrem Anbieter her. Wählen Sie dann die Repositorys aus, die für Penetrationstests zur Verfügung gestellt werden sollen.

Den vollständigen Integrationsablauf finden Sie unter AWS Security Agent Connect GitHub Repositorys verbinden oder im Verbindungsthema für Ihren Anbieter.

Schritt 4: Erstellen Sie einen Penetrationstest und führen Sie ihn durch

> **Note**
>
> Sie erstellen und führen Penetrationstests in der AWS Security Agent-Webanwendung durch. Wir empfehlen, Tests in einer Testumgebung durchzuführen, die der Produktion sehr ähnlich ist.

1. Starten Sie die Webanwendung: Wählen Sie die Registerkarte Web-App und dann Administratorzugriff.
2. Wählen Sie in der linken Seitenleiste Penetrationstests und dann Penetrationstest erstellen aus.
3. Details zum Penetrationstest — Richten Sie den Testumfang und die Protokollquelle ein:
   a. Geben Sie einen Pentest-Namen ein.
   b. Wählen Sie unter Ziel-URLs eine verifizierte Domain zum Testen aus oder geben Sie sie ein (z. B.`https://example.com`). Nur verifizierte Domains können getestet werden, und Subdomains werden automatisch abgedeckt.
   c. (Optional) Fügen Sie unter Zugängliche URLs Domänen hinzu, die der Agent für die Anmeldung und Navigation erreichen muss, aber nicht angreifen soll, z. B. Identitätsanbieter, CDNs oder APIs außerhalb Ihrer Zieldomain. Der AWS Security Agent kann diese Domänen erreichen, testet sie jedoch nicht. Nur Ihre Zieldomänen werden angegriffen.
   d. Überprüfen Sie die Netzwerkkonfigurationsregeln, um zu überprüfen, welche Endpunkte während des Tests Datenverkehr empfangen können und welche nicht.
   e. Wählen Sie unter Berechtigungen eine Servicerolle und optional eine CloudWatch Protokollgruppe aus. Wählen Sie Weiter aus.
4. (Optional) VPC-Ressourcen — Konfigurieren Sie eine VPC, wenn sich Ihr Ziel in einem privaten Netzwerk befindet, das nicht öffentlich erreichbar ist. Siehe the section called “Penetrationstest aktivieren”.
5. (Optional) Authentifizierungsressourcen — Geben Sie Anmeldeinformationen an, damit der Agent authentifizierte, nicht öffentliche Pfade erreichen kann. Wir empfehlen, diese hinzuzufügen, um die Abdeckung zu erweitern und Sicherheitslücken hinter einer Anmeldung aufzudecken.
6. (Optional) Zusätzliche Konfiguration — Fügen Sie Anwendungskontext wie Dateien, GitHub Repositorys oder S3-Quellen hinzu, um die Suchqualität zu verbessern. Sie können hier auch die automatische Codekorrektur aktivieren und andere Ausführungsoptionen festlegen.
7. Wählen Sie Erstellen und ausführen, um den Test jetzt zu starten, oder Pentest erstellen, um ihn zu speichern und später auszuführen.

Schritt 5: Überprüfen Sie die Ergebnisse des Penetrationstests

1. Die Durchführung eines Penetrationstests kann mehrere Stunden dauern. Je nach Größe und Komplexität Ihrer Anwendung sind die meisten innerhalb von 16 Stunden abgeschlossen.
2. Der Lauf beginnt mit einer Preflight-Phase, in der das Setup vor dem Teststart validiert wird: Sie richtet die Protokollierungsinfrastruktur ein, prüft, ob Ihre Zielendpunkte erreichbar sind, und

bereitet die Testumgebung vor. Wenn eine Preflight-Prüfung fehlschlägt (z. B. eine Zieldomäne, die nicht aufgelöst werden kann), wird der Lauf beendet, bevor der Test beginnt. Öffnen Sie die Registerkarte Preflight, um zu sehen, welche Prüfung fehlgeschlagen ist, lösen Sie das Problem und starten Sie einen neuen Testlauf.

3. Wenn der Lauf abgeschlossen ist, öffnen Sie ihn und überprüfen Sie die Registerkarten Testlauf — Übersicht, Protokolle und Ergebnisse.
4. Wählen Sie auf der Registerkarte Ergebnisse ein Ergebnis aus, um dessen Beschreibung, Schweregrad, Risikoart und Belege einzusehen.

Weitere Informationen finden Sie unter the section called “Erstellen Sie einen Penetrationstest” und the section called “Überprüfen Sie die Ergebnisse eines Penetrationstests”.

Schnellstart: Führen Sie eine Codeüberprüfung durch

Dieser Schnellstart führt Sie durch die Durchführung Ihres ersten Code-Reviews mit dem AWS Security Agent. AWS Security Agent durchsucht Ihre Quellcode-Repositorys auf Sicherheitslücken und die Einhaltung der Sicherheitsanforderungen Ihres Unternehmens.

> **Note**
>
> Sie benötigen Zugriff auf die AWS-Managementkonsole, um den AWS Security Agent einzurichten, und Zugriff auf die Webanwendung, um Codeprüfungen zu erstellen und auszuführen.

Voraussetzungen

Bevor Sie beginnen, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Zugriff auf die AWS-Managementkonsole mit Berechtigungen zur Einrichtung von AWS Security Agent.
- Ein Quellcode-Repository (GitHub, GitLab, oder Bitbucket) oder eine Amazon S3 S3-Quelle, die den Code enthält, den Sie überprüfen möchten.

Schritt 1: AWS Security Agent in der AWS-Konsole einrichten

Wenn Sie AWS Security Agent noch nicht eingerichtet haben, schließen Sie die Ersteinrichtung ab:

1. Navigieren Sie in der AWS-Managementkonsole zu AWS Security Agent.
2. Wählen Sie AWS Security Agent einrichten aus.
3. Erstellen Sie einen Agentenbereich. Ein Agentenbereich kann von mehreren Benutzern verwendet werden und sollte für jede Anwendung, die Sie testen möchten, spezifisch sein. Geben Sie einen Namen und eine Beschreibung für Ihren ersten Agentenbereich ein. Dieser Name wird Benutzern in der Webanwendung angezeigt. Der Name sollte die Anwendung identifizieren, deren Code Sie überprüfen möchten.
4. Wählen Sie IAM-only unter Konfiguration des Benutzerzugriffs die Option Zugriff aus.
   - Dieser Schnellstart behandelt nicht die Aktivierung von Single Sign-On (SSO) mit IAM Identity Center. Dadurch können Benutzer von der AWS-Konsole aus direkt auf die AWS Security Agent-Webanwendung zugreifen.
   - Um Benutzern ohne Zugriff auf die AWS-Managementkonsole die Durchführung von Codeprüfungen zu ermöglichen, aktivieren Sie die IAM Identity Center-Integration. Details hierzu finden Sie unter the section called “Benutzern Zugriff auf die AWS Security Agent-Web-App gewähren”.
5. Wählen Sie AWS Security Agent einrichten aus.

> **Note**
>
> Wenn Sie Setup wählen, erstellt AWS Security Agent Ihren Agent Space und richtet eine Webanwendung ein, in der Benutzer Penetrationstests, Codeprüfungen, Bedrohungsmodelle und Entwurfsprüfungen durchführen können.

Schritt 2: Aktivieren und konfigurieren Sie die Codeüberprüfung

> **Note**
>
> Wenn Sie bereits GitHub Repositorys oder S3-Buckets mit Ihrem Agent Space verbunden haben (z. B. durch die Einrichtung von Penetrationstests), ist die Codeüberprüfung bereits aktiviert. Sie können diesen Schritt überspringen und direkt zur Webanwendung wechseln.

Öffnen Sie den Einrichtungsassistenten für die Codeüberprüfung

1. Wählen Sie in der linken Seitenleiste Agent Spaces und dann Ihren Agent Space aus.

2. Wählen Sie in der Kopfzeile oder im Tab Codeüberprüfung die Option Codeüberprüfung aktivieren aus.

Integrationen, Repositorien und Buckets Connect

1. (Falls Sie noch keine GitHub Integration haben) Erstellen Sie eine Registrierung. GitHub Wenn Sie bereits eine haben, fahren Sie mit dem nächsten Schritt fort.
   a. Wählen Sie im Abschnitt Verbundene Integrationen die Option Hinzufügen und dann Neue Registrierung erstellen aus.
   b. Wählen Sie GitHubund wählen Sie Weiter.
   c. Wählen Sie Installieren und autorisieren und schließen Sie dann die Installation ab in GitHub:
      i. Wählen Sie den GitHub Benutzer oder die Organisation aus, der das Repository gehört, das Sie überprüfen möchten.
      ii. Wählen Sie Alle Repositorys oder Nur ausgewählte Repositorys aus.
      iii. Wählen Sie Installieren und autorisieren und schließen Sie die Authentifizierung ab. GitHub
   d. Zurück in der AWS-Managementkonsole geben Sie einen Registrierungsnamen ein und vergewissern Sie sich, dass der Kontotyp dem Ort entspricht, an dem Sie die GitHub App installiert haben.
   e. Wählen Sie Connect, um die Registrierung zu speichern.

   Den vollständigen GitHub Integrationsablauf finden Sie unter AWS Security Agent mit GitHub Repositorys Connect.
2. Connect GitHub Repositorys. Wähle im Abschnitt Verbundene Integrationen die Option Hinzufügen und dann deine GitHub Registrierung aus. Der zweistufige GitHubConnect-Assistent wird geöffnet:
   a. Wählen Sie unter GitHub Connect-Repositories die Repositorys aus, die Sie einbeziehen möchten, und klicken Sie auf Weiter.
   b. Stellen Sie unter Funktionen verwalten für jedes Repository die folgenden Optionen ein:
      - Kommentare zur Codeüberprüfung — Lassen Sie den AWS Security Agent Sicherheitsergebnisse als Kommentare zu Pull-Anfragen im Repository veröffentlichen.
      - Automatische Problembehebung — Lassen Sie Benutzer der AWS Security Agent-Webanwendung Pull-Requests anfordern, die die Ergebnisse korrigieren.
   c. Wählen Sie Speichern, um zum Einrichtungsassistenten zurückzukehren.

3. (Optional) Connect S3-Quellen an. Wählen Sie im Abschnitt S3-Buckets die Option S3-Ressource hinzufügen und geben Sie den S3-URI für einen Bucket ein, der Quellcode enthält, oder wählen Sie Durchsuchen, um einen auszuwählen.
4. Wählen Sie Ihre Einstellungen für die Codeüberprüfung aus. In der Standardeinstellung „Sicherheitsanforderungen und Ergebnisse von Sicherheitslücken“ wird der Code sowohl auf die Einhaltung der von Ihnen aktivierten Sicherheitsanforderungen als auch auf häufig auftretende Sicherheitslücken analysiert.
5. Wählen Sie Weiter aus.

Optionale Konfigurationen

1. Konfigurieren Sie optionale CloudWatch Protokollgruppen und den Dienstzugriff. Die Standarddienstrolle ist mit den erforderlichen Berechtigungen vorkonfiguriert.
2. Wählen Sie Speichern.

Schritt 3: Einen Code-Review erstellen und ausführen

> **Note**
>
> Sie erstellen und führen Code-Reviews nur in der AWS Security Agent-Webanwendung aus.

1. Wählen Sie die Registerkarte Web-App und dann Administratorzugriff, um die AWS Security Agent-Webanwendung zu starten.
2. Wählen Sie in der linken Seitenleiste Code Reviews aus.
3. Wähle Code-Review erstellen.
4. Konfigurieren Sie den Code-Review:
   a. Geben Sie einen Titel ein, der den Umfang dieser Überprüfung angibt (z. B. „billing-service-security-review“).
   b. Wählen Sie unter Quellen die GitHub Repositorys aus, die Sie in Schritt 2 mit Ihrem Agent Space verbunden haben, oder geben Sie die S3-Quellen ein, die Sie scannen möchten.
   c. Wählen Sie die Service-Rolle aus Ihren konfigurierten Rollen aus.
   d. (Optional) Wählen Sie Automatische Codekorrektur aktivieren, damit der AWS Security Agent automatisch Pull-Anfragen mit Korrekturen für alle Ergebnisse sendet.

5. Wählen Sie Codeüberprüfung erstellen aus.
6. Wählen Sie auf der Detailseite des Code-Reviews die Option Überprüfung starten aus.

Schritt 4: Überprüfe die Ergebnisse des Code-Reviews

1. Die Codeüberprüfung dauert in der Regel 30—60 Minuten, abhängig von der Größe Ihrer Codebasis.
2. Die Ausführung beginnt mit einer Preflight-Phase, in der das Setup vor dem Start des Scans validiert wird: Sie bestätigt, dass der AWS Security Agent Ihren Quellcode aus Ihrem Repository oder Ihrer S3-Quelle abrufen kann, und richtet die Scanumgebung ein. Wenn eine Preflight-Prüfung fehlschlägt (z. B. weil sie nicht auf Ihre Quelle zugreifen kann), wird der Lauf vor der statischen Analyse gestoppt. Öffnen Sie die Registerkarte Preflight, um zu sehen, welche Prüfung fehlgeschlagen ist, lösen Sie das Problem und starten Sie eine neue Ausführung.
3. Sobald Sie fertig sind, navigieren Sie zum abgeschlossenen Lauf und wählen Sie den Tab Ergebnisse aus.
4. Überprüfen Sie die Ergebnisse in der Listendetailansicht:
   a. Wählen Sie im linken Bereich ein Ergebnis aus, um dessen Details anzuzeigen.
   b. Lesen Sie die Abschnitte Beschreibung, Positionen des Codes, Beweise und Vorgeschlagene Problembehebung.
   c. Verwenden Sie den Korrekturcode, um eine Pull-Anfrage mit einem Fix zu generieren, oder überprüfen Sie die PRs zur automatischen Problembehebung, falls Sie diese Option aktiviert haben.

Weitere Informationen finden Sie unter the section called “Einen Code-Review erstellen” und the section called “Ergebnisse eines Code-Reviews überprüfen”.

Schnellstart: Ein Bedrohungsmodell ausführen

Dieser Schnellstart führt Sie durch die Ausführung Ihres ersten Bedrohungsmodells mit dem AWS Security Agent. Ein Bedrohungsmodell analysiert die Architektur Ihrer Anwendung und erstellt eine Systemübersicht (wie AWS Security Agent Ihr System versteht) und eine Reihe von Bedrohungen (wie sie angegriffen werden könnte, jeweils mit Schweregrad, STRIDE-Klassifizierung und Empfehlungen). Sie können ein Bedrohungsmodell anhand von Entwurfsdokumenten (Umfangsdokumenten) ausführen, um den Fokus zu definieren, oder anhand des Quellcodes (Quellen), um einen Kontext für Ihr vorhandenes System bereitzustellen, oder beides.

> **Note**
>
> Sie benötigen Zugriff auf die AWS-Managementkonsole, um den AWS Security Agent einzurichten, und Zugriff auf die Webanwendung, um Bedrohungsmodelle zu erstellen und auszuführen.

Schritt 1: AWS Security Agent in der AWS-Konsole einrichten

Wenn Sie AWS Security Agent noch nicht eingerichtet haben, schließen Sie die Ersteinrichtung ab:

1. Navigieren Sie in der AWS-Managementkonsole zu AWS Security Agent.
2. Wählen Sie AWS Security Agent einrichten aus.
3. Erstellen Sie einen Agentenbereich. Ein Agentenbereich kann von mehreren Benutzern verwendet werden und sollte für jede Anwendung, die Sie sichern möchten, spezifisch sein. Geben Sie einen Namen und eine Beschreibung für Ihren ersten Agentenbereich ein. Der Name sollte die Anwendung identifizieren, für die Sie ein Bedrohungsmodell erstellen möchten.
4. Wählen Sie IAM-only unter Konfiguration des Benutzerzugriffs die Option Zugriff aus.
   - Dieser Schnellstart behandelt nicht die Aktivierung von Single Sign-On (SSO) mit IAM Identity Center. Auf diese Weise können Benutzer direkt von der AWS-Konsole aus auf die AWS Security Agent-Webanwendung zugreifen.
   - Wenn Sie Benutzern ohne Zugriff auf die AWS-Managementkonsole die Möglichkeit geben möchten, Aufgaben wie das Starten eines Bedrohungsmodells auszuführen, sollten Sie die IAM Identity Center-Integration aktivieren.
5. Wählen Sie AWS Security Agent einrichten.

> **Note**
>
> Wenn Sie Setup wählen, erstellt AWS Security Agent Ihren Agent Space und richtet eine Webanwendung ein, in der Benutzer Penetrationstests, Codeprüfungen, Bedrohungsmodelle und Entwurfsprüfungen durchführen können.

Schritt 2: Quellcode Connect

> **Note**
>
> Wenn Sie bereits Repositorys oder S3-Buckets mit Ihrem Agent Space verbunden haben (z. B. durch die Einrichtung von Code-Reviews oder Penetrationstests), können Sie diesen Schritt überspringen und direkt zur Webanwendung wechseln. Sie können jederzeit ein Bedrohungsmodell auf hochgeladenen Scope-Dokumenten ausführen, ohne den Quellcode zu verbinden.

Connect Repositorys oder S3-Buckets, die den Quellcode enthalten, den der Agent als Kontext für das Verständnis Ihres vorhandenen Systems verwenden soll:

1. Wählen Sie in der linken Seitenleiste Agent Spaces und dann Ihren Agent Space aus.
2. Navigieren Sie zum Abschnitt Integrationen, um Ihren Quellcode-Anbieter zu verbinden.
3. Alternativ können Sie S3-Buckets verbinden, die Ihren Quellcode enthalten.

Den vollständigen Integrationsablauf finden Sie unter AWS Security Agent mit GitHub Repositorys Connect.

Schritt 3: Erstellen Sie ein Bedrohungsmodell und führen Sie es aus

> **Note**
>
> Sie erstellen und führen Bedrohungsmodelle in der AWS Security Agent-Webanwendung aus.

1. Starten Sie die Webanwendung über die Registerkarte Web-App in der AWS-Managementkonsole. Wenn Sie IAM Identity Center konfiguriert haben, melden Sie sich alternativ direkt an.
2. Wählen Sie in der linken Seitenleiste Bedrohungsmodelle aus.
3. Wählen Sie Bedrohungsmodell erstellen aus.
4. Konfigurieren Sie das Bedrohungsmodell:

a. Geben Sie einen Titel ein, der den Umfang dieses Bedrohungsmodells identifiziert (z. B. „Billing-Service“ oder „Checkout-API“).

b. Geben Sie mindestens eine Eingabe ein:

- Laden Sie unter Scope-Dokumente Designdokumente (DOC, DOCX, JPEG, MD, PDF, PNG oder TXT) hoch, geben Sie S3-URIs ein oder wählen Sie Confluence-Seiten aus.
- Wählen Sie unter Quellen Repositorys aus Ihren verbundenen Integrationen aus oder geben Sie S3-URIs mit Quellcode ein.

> **Tip**
>
> Geben Sie sowohl Quellen als auch Dokumente zum Umfang an, um das Bedrohungsmodell auf ein bestimmtes Design zu beschränken (z. B. ein Dokument zum Feature-Design), und geben Sie dem Agenten gleichzeitig Ihren Quellcode als Kontext, damit er Ihr bestehendes System verstehen kann.

c. Wählen Sie die Servicerolle aus Ihren konfigurierten Rollen aus.

d. (Optional) Wählen Sie eine Protokollgruppe für CloudWatch Protokolle aus.

5. Wählen Sie Bedrohungsmodell erstellen aus.
6. Wählen Sie auf der Detailseite des Bedrohungsmodells die Option Start run aus.

Schritt 4: Bedrohungen überprüfen

1. Der Lauf führt seine Analyseaufgaben durch. Sie können den Fortschritt auf der Registerkarte Preflight überwachen und die Aufgabendetails auf der Registerkarte Protokolle einsehen.
2. Sobald der Vorgang abgeschlossen ist, ändert sich der Ausführungsstatus in Abgeschlossen.
3. Wählen Sie die Registerkarte Übersicht, um die Systemübersicht und die Verteilung des Schweregrads zu überprüfen.
4. Wählen Sie die Registerkarte Bedrohungen, um die identifizierten Bedrohungen zu überprüfen:

a. Wählen Sie eine Bedrohung aus der Liste aus, um ihre Details im Seitenbereich anzuzeigen.

b. Überprüfen Sie die Felder Aussage, Schweregrad, STRIDE-Kategorien, Empfehlung, Nachweis und andere Felder.

c. Ändern Sie den Bedrohungsstatus auf „Gelöst“ oder „Abgewiesen“, während Sie die einzelnen Bedrohungen adressieren oder einschätzen.

Weitere Informationen finden Sie unter the section called “Erstellen Sie ein Bedrohungsmodell” und the section called “Überprüfen Sie Bedrohungen anhand eines Bedrohungsmodells”.

Für Administratoren (Konsole)

Als Administrator richten Sie AWS Security Agent in der AWS-Managementkonsole ein und konfigurieren Agent Spaces, auf die Benutzer über die AWS Security Agent-Webanwendung zugreifen. Jeder Agent Space steht für eine eigene Umgebung mit spezifischen Berechtigungen und Ressourcen.

Wir empfehlen, für jede Anwendung, die Sie testen möchten, einen eigenen Agent Space zu erstellen. Wenn Sie beispielsweise zwei interne Projekte haben — eine Abrechnungsanwendung und eine Anwendung zur Aufgabenverfolgung — sollten Sie zwei separate Agent Spaces erstellen.

Beispielkonfiguration

Stellen Sie sich vor, ein Administrator richtet einen Agent Space ein, um die Sicherheit einer internen Abrechnungsanwendung zu beurteilen. Der Administrator würde:

- Überprüfen Sie die Domain (z. B.`beta.billing.example.com`)
- Connect zur Codeüberprüfung her GitHub und aktivieren Sie sie
- Konfigurieren Sie den Netzwerkzugriff, indem Sie eine geeignete VPC, ein Subnetz und eine Sicherheitsgruppe für Penetrationstests zuweisen

Wenn Benutzer einen Penetrationstest oder eine Entwurfsprüfung einleiten, können sie aus diesen vorkonfigurierten Ressourcen auswählen und dabei innerhalb der von Ihnen definierten Richtlinien arbeiten und gleichzeitig die Flexibilität für ihre spezifischen Bewertungsanforderungen wahren.

Agent Spaces einrichten und erstellen

Richten Sie den AWS Security Agent für Ihre Organisation ein und erstellen Sie die Agent Spaces, die die Ressourcen und Bewertungen der einzelnen Anwendungen umfassen.

AWS Security Agent einrichten

Konfigurieren Sie den AWS Security Agent für Ihr Unternehmen, indem Sie Ihren ersten Agent Space einrichten und festlegen, wie Benutzer auf die Webanwendung zugreifen.

Diese Ersteinrichtung ermöglicht es Administratoren, Sicherheitsfunktionen in der AWS-Konsole zu konfigurieren, und bietet Benutzern über die Security Agent-Webanwendung Zugriff auf

Entwurfsprüfungen und Penetrationstests. Nach dieser einmaligen Einrichtung können Sie mit einem vereinfachten Verfahren weitere Agent Spaces erstellen.

-Übersicht

Agent Spaces verstehen

Ein Agent Space ist ein spezieller Arbeitsbereich zum Schutz einer bestimmten Anwendung oder eines bestimmten Projekts. Er enthält alle Sicherheitsüberprüfungen, optional verbundene GitHub Repositorys, Konfigurationen, Ergebnisse und Ergebnisse der Penetrationstests für diese Anwendung.

Agent Spaces hilft Ihnen bei der Organisation der Sicherheitsarbeit, indem die Sicherheitsbeurteilungen der einzelnen Anwendungen getrennt sind, sodass sich die Teams auf ihre spezifische Anwendung konzentrieren können. Wir empfehlen, einen Agent Space pro Anwendung oder Projekt einzurichten, um klare Grenzen zu wahren.

Sicherheitsanforderungen werden auf Organisationsebene definiert und gelten für alle Agent Spaces, wobei jeder Agent Space seine eigenen Entwurfsdokumente, Code-Repositorys, Penetrationstestkonfigurationen, Penetrationstestergebnisse und Sicherheitsfeststellungen verwaltet.

Was ist in einem Agent Space enthalten

Jeder Agent Space enthält:

- Optional verbundene GitHub Repositorys, die der Anwendung oder dem Projekt zugeordnet sind
- Frühere Entwurfsprüfungen für die Anwendung
- Anwendungsspezifische Konfigurationen und Grenzen für Penetrationstests
- Testergebnisse und Sicherheitsfeststellungen bei Penetrationstests

Was werden Sie bei der Einrichtung konfigurieren

Bei dieser erstmaligen Einrichtung treffen Sie organisatorische Entscheidungen, die für alle Agent Spaces gelten:

- Zugriffsmethode — Wählen Sie aus, wie Benutzer auf die Security Agent Webanwendung zugreifen sollen (IAM Identity Center SSO oder IAM-only Zugriff über die AWS-Konsole)
- Berechtigungen — Konfigurieren Sie die IAM-Rolle, die die Webanwendung für den Zugriff auf AWS-Services verwendet

- First Agent Space — Erstellen Sie Ihren ersten Agent Space mit einem Namen und einer Beschreibung

> **Note**
>
> Nach Abschluss dieser Einrichtung ist das Erstellen zusätzlicher Agent Spaces einfacher — geben Sie einfach einen Namen und eine Beschreibung ein. Einzelheiten the section called “Erstellen Sie einen Agentenbereich” zur Erstellung nachfolgender Agent Spaces finden Sie unter.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Berechtigungen zum Erstellen und Verwalten von AWS Security Agent-Ressourcen
- Verständnis der Anforderungen Ihres Unternehmens an das Identitätsmanagement
- (Optional) AWS-Konto mit Berechtigungen zum Erstellen von IAM-Rollen und zum Konfigurieren von IAM Identity Center (bei Verwendung von SSO-Zugriff)
- (Optional) Bestehende IAM-Rolle, wenn Sie eine benutzerdefinierte Berechtigungskonfiguration verwenden möchten

Schritt 1: Erstellen Sie Ihren ersten Agent Space

Definieren Sie die grundlegenden Eigenschaften Ihres ersten Agent Space, die den Benutzern in der Webanwendung angezeigt werden.

1. Wählen Sie auf der AWS Security Agent-Konsolenseite die Option Security Agent einrichten aus.
2. Geben Sie im Feld Agent Space-Name einen Namen für Ihren Agent Space ein.

> **Note**
>
> Der Agent Space-Name wird Benutzern in der Webanwendung angezeigt und hilft dabei, zu identifizieren, für welche Anwendung oder welches Projekt dieser Space steht.

3. (Optional) Geben Sie im Feld Beschreibung eine Beschreibung ein, anhand derer Sie den Zweck des Agent Space besser erkennen können.

> **Tip**
>
> Anhand der Beschreibung lässt sich der Zweck des Agent Space besser erkennen. Wir empfehlen, die spezifische Anwendung oder das Projekt zu beschreiben, das durch diesen Agent Space gesichert wird, z. B. „Webanwendung für das Kundenportal“ oder „Microservices für die Zahlungsabwicklung“ oder „Interne Analyseplattform“.

Schritt 2: Wählen Sie Ihre Zugriffsmethode

Wählen Sie aus, wie Benutzer auf die Security Agent Webanwendung zugreifen sollen. Sie haben die Wahl, ob Sie den SSO-Zugriff über das IAM Identity Center aktivieren oder den Zugriff über die AWS-Konsole bereitstellen möchten.

> **Note**
>
> Wenn Sie sich für IAM-only Zugriff entscheiden und später IAM Identity Center verwenden möchten, müssen Sie Ihr AWS Security Agent-Setup löschen und den Einrichtungsvorgang erneut abschließen.

1. Wählen Sie im Abschnitt Zugriffsmethode eine der folgenden Optionen aus:
   - IAM Identity Center (SSO) — Aktivieren Sie den SSO-Zugriff für Ihr Team über IAM Identity Center. Diese Option wird für Teams empfohlen, die eine zentrale Benutzerverwaltung benötigen und möchten, dass Benutzer direkt auf die Webanwendung zugreifen können, ohne die AWS-Konsole verwenden zu müssen.
   - IAM-only Zugriff — Stellen Sie den Zugriff über einen Administratorzugriffslink in der AWS-Konsole bereit. Diese Option ist einfacher einzurichten und eignet sich für Teams, die einen konsolenbasierten Zugriff bevorzugen oder keine SSO-Funktionen benötigen.
2. Wenn Sie IAM Identity Center (SSO) ausgewählt haben, fahren Sie mit Schritt 2a weiter unten fort.
3. Wenn Sie IAM-only Zugriff ausgewählt haben, fahren Sie mit Schritt 3 fort.

Schritt 2a: IAM Identity Center konfigurieren (nur SSO-Zugriff)

Führen Sie diese Schritte nur aus, wenn Sie IAM Identity Center (SSO) als Zugriffsmethode ausgewählt haben.

1. Überprüfen Sie im Abschnitt Connect to IAM Identity Center die AWS-Region.

> **Important**
>
> Ihre Anwendung muss in derselben Region konfiguriert sein, in der Sie IAM Identity Center aktivieren. In der angezeigten Region wird Ihr Agent Space erstellt. IAM Identity Center muss in derselben Region aktiviert sein, in der Sie Ihren Agent Space erstellen.

2. Wählen Sie im Abschnitt IAM Identity Center eine der folgenden Optionen aus:
   - Wählen Sie Kontoinstanz erstellen, um eine neue IAM Identity Center-Kontoinstanz zu erstellen
   - Wenn in derselben Region bereits eine Organisationsinstanz vorhanden ist, stellt der AWS Security Agent automatisch eine Verbindung zu ihr her

> **Note**
>
> Nachdem der AWS Security Agent mit dem IAM Identity Center verbunden ist, können Sie den Zugriff verwalten, indem Sie der Anwendung Benutzer in IAM Identity Center zuweisen.

3. Wenn Sie eine Verbindung zu Active Directory oder einem externen Identitätsanbieter herstellen, überprüfen Sie die Informationswarnung.

> **Important**
>
> Wenn Sie bereits Benutzer in Active Directory oder einer anderen Identitätsquelle verwalten und planen, diese Identitätsquelle mit dem IAM Identity Center zu verbinden, brechen Sie die Installation ab und rufen Sie zuerst die IAM Identity Center-Konsole auf. Wählen Sie die Identitätsquelle aus, mit der Sie eine Verbindung herstellen möchten, bevor Sie AWS Security Agent einrichten. Wenn Sie die Identitätsquellen zu einem späteren Zeitpunkt ändern, werden möglicherweise vorhandene Benutzerzuweisungen entfernt.

Schritt 3: Berechtigungen konfigurieren (optional)

Konfigurieren Sie die IAM-Rolle, die Ihre Security Agent-Webanwendung für den Zugriff auf AWS-Services, APIs und Konten verwendet.

1. Suchen Sie den Abschnitt Konfiguration der Berechtigungen — optional.

2. Wenn der Abschnitt ausgeblendet ist, wählen Sie den Abschnitt Konfiguration der Berechtigungen aus, um ihn zu erweitern.
3. Wählen Sie eine der folgenden Optionen aus:
   - Standardrolle erstellen — AWS Security Agent erstellt automatisch eine neue IAM-Rolle mit den erforderlichen Berechtigungen für die Webanwendung
   - Eine andere Rolle verwenden — Wählen Sie eine bestehende IAM-Rolle aus den verfügbaren Optionen aus
4. Lesen Sie die Beschreibung der Rolle:

> **Note**
>
> Für Ihre Webanwendung wird eine Standard-IAM-Rolle für den Zugriff auf andere AWS-Services, APIs und Konten erstellt. Der Rolle werden Berechtigungen erteilt, die für Sicherheitsüberprüfungen erforderlich sind.

Schritt 4: Schließen Sie die Einrichtung ab

Nachdem Sie alle erforderlichen Einstellungen konfiguriert haben, schließen Sie das Setup ab, um Ihren ersten Agent Space zu erstellen und die Security Agent Webanwendung einzurichten.

1. Überprüfen Sie alle Konfigurationsabschnitte, um die Richtigkeit sicherzustellen.
2. Wählen Sie Einrichten aus.
3. AWS Security Agent erstellt Ihren Agent Space, richtet die Webanwendung ein und konfiguriert die erforderlichen AWS-Ressourcen.

> **Note**
>
> Nach der Ersteinrichtung haben Sie die Möglichkeit, Funktionen wie Penetrationstests, Sicherheitsanforderungen und GitHub Integration zu konfigurieren.

Nächste Schritte

Nach der Einrichtung von AWS Security Agent:

- Definieren Sie die Sicherheitsanforderungen für Ihr Unternehmen

- Konfigurieren Sie Penetrationstestfunktionen wie Domainverifizierung und VPC-Zugriff für Ihren Agent Space
- Connect GitHub Repositorys für den Kontext von Codeüberprüfung und Penetrationstests
- (Wenn Sie IAM Identity Center verwenden) Weisen Sie Benutzer dem Agent Space in der AWS Security Agent-Konsole zu
- (Wenn Sie IAM-only Access verwenden) Starten Sie die Webanwendung über den Administratorzugriffslink in der AWS-Konsole
- Erstellen Sie zusätzliche Agent Spaces für andere Anwendungen (siehethe section called “Erstellen Sie einen Agentenbereich”)

Erstellen Sie einen Agentenbereich

Erstellen Sie Agent Spaces, um Anwendungen oder Projekte in Ihrer Organisation zu schützen. Jeder Agent Space bietet einen Arbeitsbereich für Sicherheitsbeurteilungen, Ergebnisse und Konfigurationen, die für diese Anwendung oder dieses Projekt spezifisch sind, und auf den mehrere Benutzer zugreifen können.

Bei diesem Verfahren wird davon ausgegangen, dass Sie die erste Einrichtung des AWS Security Agents bereits abgeschlossen haben. Wenn Sie den AWS Security Agent noch nicht eingerichtet haben, finden Sie weitere Informationen unterthe section called “AWS Security Agent einrichten”.

-Übersicht

Agent Spaces verstehen

Ein Agent Space ist ein spezieller Arbeitsbereich zum Schutz einer bestimmten Anwendung oder eines bestimmten Projekts. Er enthält alle Sicherheitsüberprüfungen, optional verbundene Code-Repositorys, Penetrationstest-Konfigurationen, Ergebnisse und Ergebnisse für diese Anwendung.

Agent Spaces hilft Ihnen bei der Organisation der Sicherheitsarbeit, indem die Sicherheitsbeurteilungen der einzelnen Anwendungen getrennt sind, sodass sich die Teams auf ihre spezifische Anwendung konzentrieren können. Wir empfehlen, einen Agent Space pro Anwendung oder Projekt einzurichten, um klare Grenzen zu wahren.

Eine umfassende Erläuterung der Agent Spaces und ihrer Integration in die Sicherheitsstruktur Ihres Unternehmens finden Sie unterthe section called “Verstehen Sie die Ressourcenhierarchie und den Lebenszyklus”.

Was ist in einem Agent Space enthalten

Jeder Agent Space enthält:

- Optional verbundene Code-Repositorys, die der Anwendung oder dem Projekt zugeordnet sind
- Frühere Entwurfsprüfungen für die Anwendung oder das Projekt
- Anwendungsspezifische Konfigurationen und Grenzen für Penetrationstests
- Testergebnisse und Sicherheitsfeststellungen bei Penetrationstests

Erstellen Sie einen Agentenbereich

Erstellen Sie einen neuen Agent Space für eine Anwendung oder ein Projekt, das Sie sichern möchten.

1. Navigieren Sie in der AWS Security Agent-Konsole zur Seite Agent Spaces.
2. Wählen Sie Create Agent Space aus.
3. Geben Sie im Feld Agent Space-Name einen Namen für Ihren Agent Space ein.

> **Note**
>
> Der Agent Space-Name wird Benutzern in der Webanwendung angezeigt und hilft dabei, zu identifizieren, für welche Anwendung oder welches Projekt dieser Space steht.

4. (Optional) Geben Sie im Feld Beschreibung eine Beschreibung ein, anhand derer Sie den Zweck des Agent Space besser erkennen können.

> **Tip**
>
> Anhand der Beschreibung lässt sich der Zweck des Agent Space besser erkennen. Wir empfehlen, die spezifische Anwendung oder das Projekt zu beschreiben, das durch diesen Agent Space gesichert wird, z. B. „Webanwendung für das Kundenportal“, „Mikrodienste für die Zahlungsabwicklung“ oder „Interne Analyseplattform“.

5. Wählen Sie Erstellen aus.

> **Note**
>
> AWS Security Agent erstellt Ihren Agent Space. Sie können jetzt Funktionen konfigurieren und Ressourcen verbinden, die für diese Anwendung spezifisch sind.

Nächste Schritte

Nachdem Sie Ihren Agent Space erstellt haben:

- Connect Quellcode-Repositorys (GitHub, GitLab, Bitbucket oder GitHub Enterprise Server) für Code-Reviews und Penetrationstests
- Connect Confluence für den Dokumentationskontext bei Designprüfungen und Penetrationstests
- Aktivieren Sie die Code-Review-Funktion für verbundene Repositorys (siehe) the section called “Pull-Request-Code-Review für GitHub Repositorys aktivieren”
- Konfigurieren Sie Funktionen für Penetrationstests, einschließlich Domänenverifizierung
- (Wenn Sie IAM Identity Center verwenden) Weisen Sie diesem Agent Space im Bereich Web App auf der Agent Space-Seite Benutzer zu. (siehethe section called “Benutzern Zugriff auf die AWS Security Agent-Web-App gewähren”)
- (Wenn Sie IAM-only Zugriff verwenden) Benutzer mit Konsolenzugriff können die Webanwendung über den Administratorzugriffslink für diesen Agent Space starten

Integrationen Connect

Connect und verwalte Quellcode-Anbieter (GitHub, GitLab, Bitbucket und GitHub Enterprise Server) und Dokumentationsanbieter (Confluence) sowie private Netzwerkkonnektivität, die deine Agent Spaces für Codeüberprüfung, Penetrationstests und Bedrohungsmodellierung verwenden.

So funktionieren Integrationen mit Agent Spaces

AWS Security Agent stellt eine Verbindung zu Quellcode- und Dokumentationsanbietern von Drittanbietern her, sodass der Agent Ihren Code und Ihre Dokumentation bei Sicherheitsbewertungen analysieren kann. Bevor Sie eine Verbindung zu einem bestimmten Anbieter herstellen, ist es hilfreich zu verstehen, wie Integrationen mit Agent Spaces und Funktionen zusammenhängen.

Einmal registrieren, überall wiederverwenden

Die Verbindung eines Anbieters mit dem AWS Security Agent umfasst zwei verschiedene Schritte, die auf verschiedenen Ebenen ablaufen:

1. Registrieren Sie die Integration (Kontoebene). Sie registrieren einen Anbieter einmalig auf der Seite Integrationen in der AWS Security Agent Management Console. Eine Registrierung stellt die autorisierte Verbindung zu einem Anbieterkonto dar, z. B. zu einer GitHub Organisation, einer GitLab Gruppe, einem Bitbucket-Workspace oder einer Confluence-Site. Registrierungen sind Ressourcen auf Kontoebene.
2. Verbinden Sie Ressourcen mit einem Agent Space (Agent Space-Ebene). Von einem Agent Space aus verbinden Sie Ressourcen aus einer registrierten Integration — Repositorys für Quellcode-Anbieter oder Seiten für Confluence — und konfigurieren, wie der Agent sie verwendet. Dieser Schritt ist für jeden Agent Space spezifisch.

Eine einzige Registrierung wird für jeden Agent Space in Ihrem Konto wiederverwendet. Wenn Sie bei der Einrichtung eines Agent Space einen Anbieter registriert haben, ist dieselbe Registrierung verfügbar, wenn Sie Ressourcen mit einem anderen Agent Space verbinden — Sie registrieren denselben Anbieter nicht erneut.

Eine Verbindung, die von allen Funktionen gemeinsam genutzt wird

Wenn Sie eine Ressource mit einem Agent Space verbinden, wird diese Ressource von allen Funktionen des Agenten gemeinsam genutzt. Sie verbinden ein Repository nicht für jede Funktion separat.

- Der Lesezugriff wird gewährt, indem die Ressource verbunden wird. Wenn Sie ein Repository verbinden, kann der AWS Security Agent es für vollständige Codescans (Code-Review), Penetrationstests, Bedrohungsmodellierung und Entwurfsprüfung lesen. Wenn Sie eine Confluence-Seite verbinden, kann der Agent sie aus Gründen der Dokumentation bei Entwurfsprüfungen, Bedrohungsmodellierung und Penetrationstests lesen.
- Schreibaktionen sind pro Repository freiwillig. Wenn Sie Quellcode-Repositorys mit einem Agent Space verbinden, können Sie zusätzlich Schreibaktionen für jedes Repository aktivieren:
  - Kommentare zur Codeüberprüfung — AWS Security Agent veröffentlicht Überprüfungsergebnisse als Kommentare zu Pull-Requests (oder Merge-Anfragen in GitLab).
  - Codebehebung — Der AWS Security Agent öffnet Pull-Anfragen (oder Merge-Anfragen) mit Korrekturen für entdeckte Sicherheitslücken.

Diese Schreibaktionen sind für öffentliche Repositorys nicht verfügbar. Read-only Die Analyse gilt weiterhin.

> **Note**
>
> Confluence ist eher ein Dokumentationsanbieter als ein Quellcode-Anbieter. AWS Security Agent liest verbundene Confluence-Seiten, um Kontext für Entwurfsprüfungen, Bedrohungsmodellierung und Penetrationstests bereitzustellen. Durch die Auswahl einer Seite wird Lesezugriff gewährt. Es gibt keine Möglichkeit, die Funktionen pro Seite umzuschalten.

Unterstützte Anbieter

AWS Security Agent unterstützt die folgenden Anbieter:

- Quellcode — GitHub (in der Cloud gehostetes GitHub und in der Cloud gehostetes GitHub Unternehmen), GitHub Enterprise Server (selbst gehostet), (in der Cloud gehostet), GitLab GitLab Self-Managed (selbst gehostet) und Bitbucket Cloud.
- Dokumentation — Confluence Cloud.

Für selbst gehostete Anbieter, die nicht über das öffentliche Internet erreichbar sind, kannst du den Traffic des Agenten über eine private Verbindung weiterleiten. Weitere Informationen finden Sie unter the section called “Connect zur privat gehosteten Quellcodeverwaltung her”.

Nächste Schritte

- Registrieren Sie einen Anbieter auf der Seite Integrationen. Weitere Informationen finden Sie im Connect-Thema für Ihren Anbieter, z. B. the section called “AWS Security Agent mit GitHub Repositorys Connect”
- Verbinden Sie Ressourcen mit einem Agent Space und konfigurieren Sie Funktionen. Siehe the section called “Codeüberprüfung aktivieren” und the section called “Penetrationstest aktivieren”.

AWS Security Agent mit GitHub Repositorys Connect

Connect Ihren AWS Security Agent mit GitHub Repositorys, um Funktionen zur Codeüberprüfung, Bedrohungsmodellierung, Penetrationstests und automatische Problembehebung zu aktivieren. AWS Security Agent unterstützt sowohl in der Cloud gehostete als auch in der Cloud GitHub gehostete GitHub Unternehmen. Bevor Sie beginnen, sollten Sie the section called “So funktionieren Integrationen mit Agent Spaces” sich darüber informieren, wie eine Registrierung in allen Agent Spaces wiederverwendet und funktionsübergreifend gemeinsam genutzt wird.

GitHub Die Integration dient mehreren Zwecken:

> **Note**
>
> Diese Seite behandelt in der Cloud gehostete GitHub (github.com) und in der Cloud GitHub gehostete Unternehmen. Informationen zum selbst gehosteten GitHub Enterprise Server finden Sie unter. the section called “AWS Security Agent mit GitHub Enterprise Server Connect”

- Codeüberprüfung — Analysieren Sie automatisch die Codeänderungen in den einzelnen Pull-Requests anhand der Sicherheitsanforderungen Ihres Unternehmens und führen Sie bei Bedarf vollständige Repository-Scans durch
- Bedrohungsmodellierung — Vermitteln Sie Anwendungsinformationen, indem Sie Quellcode, Datenflüsse und Architektur analysieren
- Kontext für Penetrationstests — Vermitteln Sie durch die Analyse des Quellcodes Anwendungsverständnis für Penetrationstests
- Automatisierte Problembehebung — Senden Sie Pull-Requests mit Fixes für Sicherheitslücken, die bei Sicherheitsbewertungen entdeckt wurden

GitHub Um eine Verbindung mit dem AWS Security Agent herzustellen, müssen Sie die AWS Security GitHub Agent-App für Ihre GitHub Organisation oder Ihr Benutzerkonto autorisieren und anschließend die Verbindung in der AWS-Konsole registrieren.

Wie funktioniert GitHub die Integration

Die Pull-Request-Analyse erfolgt innerhalb von GitHub. Nachdem Sie die GitHub App autorisiert, Repositorys verbunden und Kommentare zur Codeüberprüfung in der AWS-Managementkonsole aktiviert haben, scannt der AWS Security Agent automatisch die Änderungen in jeder neuen Pull-

Anfrage (ein Differenzscan nur des geänderten Codes) und veröffentlicht die Ergebnisse als Pull-Request-Kommentare.

Vollständige Codeprüfungen, bei denen die gesamte Codebasis eines Repositorys gescannt wird, werden in der AWS Security Agent-Webanwendung erstellt und ausgeführt, nicht in GitHub.

Penetrationstests und Bedrohungsmodellierung werden innerhalb der AWS Security Agent-Webanwendung initiiert. Benutzer wählen verbundene Repositorys aus, um den Anwendungskontext bereitzustellen, und geben Zieldomänen für Penetrationstests an. Wenn Sie die automatische Problembehebung aktivieren, können Benutzer den AWS Security Agent auffordern, Fehler zu korrigieren, indem sie Pull-Anfragen an verbundene Repositorys öffnen.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- GitHub Zugriff durch den Administrator der Organisation oder Zugriff auf den Inhaber GitHub des Benutzerkontos
- Berechtigungen zur Konfiguration von Integrationen für Ihren Agent Space in der AWS-Managementkonsole
- Verstehen Sie, welche Repositorys Sie für die Codeüberprüfung, Bedrohungsmodellierung und Penetrationstests verbinden möchten

Important

Eine GitHub App kann nur einmal für ein GitHub Konto oder eine GitHub Organisation installiert werden. Wenn Sie dieselbe GitHub Organisation mit dem AWS Security Agent verbinden müssen, müssen Sie dasselbe AWS-Konto verwenden, bei dem die Integration zuerst registriert wurde.

Note

Wenn Ihre GitHub Unternehmensorganisation IP-Allowlisting aktiviert hat, müssen Sie die zulässigen IP-Adressen in der GitHub App akzeptieren. Sie können sich auch dafür entscheiden, die IP-Adressen automatisch Ihrer Zulassungsliste hinzuzufügen. Weitere Informationen finden Sie in der GitHub Dokumentation unter Zulassen des Zugriffs durch GitHub Apps und Aktivieren zulässiger IP-Adressen.

Die folgenden IP-Adressen werden für den Zugriff auf Ihre GitHub Ressourcen verwendet:

- USA Ost (Nord-Virginia): (us-east-1)
  - 34.228.181.128
  - 44.219.176.187
  - 54.226.244.221
- USA West (Oregon): (us-west-2)
  - 34.212.16.133
  - 52.89.67.212
  - 54.187.135.61
- Asien-Pazifik (Mumbai): (ap-south-1)
  - 13.126.209.199
  - 13.234.6.24
  - 35.154.102.216
- Asien-Pazifik (Singapur): (ap-southeast-1)
  - 18.139.13.125
  - 47.130.240.215
  - 54.179.238.173
- Asien-Pazifik (Sydney): (ap-southeast-2)
  - 13.237.95.197
  - 13.238.84.102
  - 52.64.174.242
- Asien-Pazifik (Tokyo) (ap-northeast-1)
  - 13.192.12.233
  - 35.74.181.230
  - 57.183.50.158
- Europa (Frankfurt) (eu-central-1)
  - 18.158.110.140
  - 52.57.96.160
  - 52.59.55.56
- Europa (Irland) (eu-west-1)

- 34.251.85.24
- 52.30.157.157
- 52.51.192.222

- Südamerika (São Paulo) (sa-east-1)
  - 54.94.247.213
  - 54.207.222.14
  - 54.232.201.242

Autorisieren und registrieren Sie die AWS Security Agent App GitHub

Autorisieren Sie die AWS Security Agent GitHub App für den Zugriff auf Ihre GitHub Organisation oder Ihr Benutzerkonto und registrieren Sie dann die Verbindung in der AWS-Konsole.

> **Important**
>
> Führen Sie alle Schritte dieses Vorgangs aus, ohne Ihren Browser zu schließen oder zu einer anderen Seite zu navigieren. Wenn der Registrierungsprozess unterbrochen wird, müssen Sie möglicherweise die GitHub App deinstallieren und von vorne beginnen.

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie Add Integration (Integration hinzufügen) aus.
3. Wählen Sie GitHub.
4. Wählen Sie Weiter aus.
5. Wählen Sie Installieren und autorisieren aus.

   Sie werden weitergeleitet, GitHub um die Autorisierung abzuschließen. Stellen Sie sicher, dass Sie GitHub mit einem Konto angemeldet sind, das Administratorzugriff auf die Organisation oder das Benutzerkonto hat, mit dem Sie eine Verbindung herstellen möchten.
6. Wählen Sie unter das Konto oder die Organisation aus, in der Sie die AWS Security Agent GitHub App installieren möchten. GitHub
7. Wählen Sie aus, auf welche Repositorys der AWS Security Agent zugreifen kann:
   - Alle Repositorys — Gewähren Sie Zugriff auf alle aktuellen und future Repositorys in der Organisation oder im Benutzerkonto

- Nur Repositorys auswählen — Wählen Sie bestimmte Repositorys aus der Drop-down-Liste aus. Sie können mehrere Repositorys nacheinander auswählen.

> **Note**
>
> Sie können den Zugriff auf das Repository jederzeit ändern, indem Sie die GitHub App-Einstellungen in Ihrer GitHub Organisation oder in den Benutzerkontoeinstellungen aufrufen.

8. Wähle Installieren und autorisieren.
9. Sie werden zurück zur AWS-Managementkonsole geleitet, um die Registrierung abzuschließen.
10. Konfigurieren Sie im Abschnitt mit den Registrierungsdetails die folgenden Felder:
    a. Registrierungsname — Geben Sie einen aussagekräftigen Namen für diese GitHub Verbindung ein. Verwenden Sie einen Namen, der die GitHub Organisation oder das Benutzerkonto identifiziert, z. B. "Acme-Corp-Org" oder "Production-Repos“.
    b. Kontotyp — Wählen Sie eine der folgenden Optionen aus der Dropdownliste aus:
        - Organisation — Wenn Sie ein GitHub Organisationskonto verbunden haben
        - Benutzer — Wenn Sie ein persönliches GitHub Benutzerkonto verbunden haben
    c. Name der Organisation (wird nur angezeigt, wenn Sie Organisation ausgewählt haben) — Geben Sie den exakten Namen Ihrer GitHub Organisation ein, wie er in erscheint GitHub.
11. Wählen Sie Connect aus.
12. Sie sehen eine Bestätigungsmeldung und kehren zur Seite Integrationen zurück, auf der Ihre neue GitHub Verbindung mit ihrem Registrierungsnamen angezeigt wird. Um weitere GitHub Organisationen oder Benutzerkonten zu verbinden, wiederholen Sie diesen Vorgang, indem Sie erneut Integration hinzufügen wählen.

Problembehandlung bei der GitHub Integration

Wenn Sie während des GitHub Integrationsprozesses auf Probleme stoßen, verwenden Sie die folgenden Anleitungen, um häufig auftretende Probleme zu lösen.

Die Registrierung konnte nicht abgeschlossen werden

Wenn Sie den Registrierungsprozess nicht abschließen konnten (z. B. weil Ihr Browser geschlossen wurde, Sie die Registrierungsseite verlassen haben oder Sie auf eine Sitzungsunterbrechung

gestoßen sind), ist die AWS Security Agent GitHub App möglicherweise in Ihrer GitHub Organisation installiert, aber nicht in der AWS-Konsole registriert.

Symptome:

- Wenn Sie erneut versuchen, die GitHub App zu autorisieren, GitHub wird „Konfigurieren“ statt „Installieren“ angezeigt
- Sie können die Registrierung in der AWS-Konsole nicht abschließen
- Die Integration erscheint nicht in Ihrer Integrationsliste

Auflösung

1. Deinstallieren Sie die AWS Security Agent GitHub App von Ihrer GitHub Organisation oder Ihrem Benutzerkonto.
2. Kehren Sie zur AWS Security Agent-Konsole zurück und starten Sie den Integrationsprozess erneut von vorne.

Mehrere AWS-Konten versuchen, dieselbe GitHub Organisation zu integrieren

Eine GitHub App kann nur einmal für ein GitHub Konto oder eine GitHub Organisation installiert werden. Wenn Sie Repositorys einer GitHub Organisation verwenden müssen, die bereits mit einem anderen AWS Security Agent-Konto integriert ist, müssen Sie das AWS-Konto verwenden, bei dem die Integration zuerst registriert wurde.

Auflösung

- Identifizieren Sie, für welches AWS Security Agent-Konto die GitHub Integration registriert ist
- Verwenden Sie dieses AWS-Konto, um Agent Spaces zu erstellen und Repositorys zu verbinden
- Wenn Sie die Integration auf ein anderes AWS-Konto verschieben müssen, deinstallieren Sie die GitHub App zuerst vom ursprünglichen AWS-Konto und integrieren Sie sie dann in das neue Konto

Nächste Schritte

Nach dem Herstellen GitHub der Verbindung mit dem AWS Security Agent:

- Navigieren Sie zu dem Agent-Bereich, in dem Sie diese Repositorys verwenden möchten

- Wählen Sie „Codeüberprüfung aktivieren“ oder „Penetrationstests einrichten“, um bestimmte Repositorys mit Ihrem Agent Space zu verbinden und deren Verwendung zu konfigurieren
- Aktivieren Sie die automatische Problembehebung, damit der AWS Security Agent Pull-Requests mit Behebung von Sicherheitslücken einreichen kann
- Überprüfen Sie die GitHub App-Berechtigungen und den Repository-Zugriff in den Einstellungen Ihrer GitHub Organisation

Eine GitHub Integration entfernen

Entfernen Sie eine GitHub Integration, wenn Sie den AWS Security Agent nicht mehr benötigen, um von einer bestimmten GitHub Organisation oder einem bestimmten Benutzerkonto aus auf Repositorys zuzugreifen. Sie müssen zuerst die AWS Security Agent GitHub App von deinstallieren, GitHub bevor Sie die Integration in der AWS-Konsole entfernen.

Voraussetzungen für die Deinstallation

Bevor Sie eine GitHub Integration entfernen, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Sie haben überprüft, mit welchen Agent Spaces Repositorys über diese Integration verbunden sind
- Ich habe die Auswirkungen verstanden: Durch das Entfernen dieser Integration werden die Repository-Verbindungen für die Codeüberprüfung, den Kontext für Penetrationstests und die Behebung von Penetrationstests in allen Agent Spaces, die Repositorys aus dieser Integration verwenden, unterbrochen
- GitHub Administratorzugriff der Organisation oder Zugriff auf den Inhaber des Benutzerkontos, um die App zu deinstallieren GitHub

Schritt 1: Deinstallieren Sie die AWS Security Agent GitHub App von GitHub

Deinstallieren Sie zunächst die AWS Security Agent GitHub App von Ihrer GitHub Organisation oder Ihrem Benutzerkonto.

Für GitHub Organizations:

1. Gehe zu github.com und öffne deine Organisationsseite.
2. Wähle in der linken Seitenleiste Einstellungen.
3. Wählen Sie unter Code, Planung und Automatisierung die Option Installierte GitHub Apps aus.
4. Suchen Sie die AWS Security Agent app in der Liste.

5. Wählen Sie neben der AWS Security Agent-App die Option Konfigurieren aus.
6. Scrollen Sie zum Ende der Konfigurationsseite und wählen Sie Deinstallieren.
7. Bestätigen Sie die Deinstallation, wenn Sie dazu aufgefordert werden.

Für GitHub Benutzerkonten:

1. Gehe zu github.com.
2. Wähle dein Profilbild.
3. Wählen Sie Einstellungen aus.
4. Wählen Sie in der linken Seitenleiste Anwendungen aus.
5. Öffnen Sie den Tab Installierte GitHub Apps.
6. Suchen Sie die AWS Security Agent app in der Liste.
7. Wählen Sie neben der AWS Security Agent-App die Option Konfigurieren aus.
8. Scrollen Sie zum Ende der Konfigurationsseite und wählen Sie Deinstallieren.
9. Bestätigen Sie die Deinstallation, wenn Sie dazu aufgefordert werden.

Schritt 2: Entfernen Sie die Integration aus dem AWS Security Agent

Entfernen Sie nach der Deinstallation der GitHub App die Integrationsregistrierung aus der AWS-Konsole.

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Suchen Sie in der GitHub Integrationsliste nach der Integration, die Sie entfernen möchten.
3. Wählen Sie die Integration aus, indem Sie darauf klicken.
4. Wählen Sie Remove (Entfernen) aus.
5. Lesen Sie den Bestätigungsdialog, der Sie vor den Auswirkungen warnt:

> ⚠ Warning
>
> Das Entfernen dieser Integration wirkt sich auf alle Agent Spaces aus, mit denen Repositorys von dieser GitHub Organisation oder diesem Benutzerkonto aus verbunden sind. Das wird kaputt gehen:
>
> - Code-Review-Funktionalität für verbundene Repositorys
> - Kontext für Penetrationstests aus verbundenen Repositorien

- Funktionen zur Behebung von Penetrationstests für verbundene Repositorien

Stellen Sie sicher, dass Sie die AWS Security Agent GitHub App von deinstalliert haben, GitHub bevor Sie fortfahren.

6. Wenn Sie die GitHub App von noch nicht deinstalliert haben GitHub, erhalten Sie eine Warnung. Kehren Sie zu Schritt 1 zurück, um zuerst die Deinstallation abzuschließen.
7. Wenn Sie die GitHub App deinstalliert haben und sich der Auswirkungen bewusst sind, wählen Sie Deinstallation bestätigen.
8. Die Integration wurde aus Ihrer Integrationsliste entfernt. Alle Agent Spaces mit Repositorys aus dieser Integration haben keinen Zugriff mehr auf diese Repositorys für Codeprüfungen, Penetrationstests oder automatische Problembehebungen.

AWS Security Agent mit GitLab Repositorys Connect

Connect Ihren AWS Security Agent mit GitLab Cloud-Repositorys, um Funktionen zur Codeüberprüfung, Bedrohungsmodellierung, Penetrationstests und automatische Problembehebung zu aktivieren. Bevor Sie beginnen, sollten Sie the section called “So funktionieren Integrationen mit Agent Spaces” sich darüber informieren, wie eine Registrierung in allen Agent Spaces wiederverwendet und funktionsübergreifend gemeinsam genutzt wird.

GitLab Die Integration dient mehreren Zwecken:

- Codeüberprüfung — Analysieren Sie automatisch die Codeänderungen in jeder Merge-Anfrage anhand der Sicherheitsanforderungen Ihres Unternehmens und führen Sie bei Bedarf vollständige Repository-Scans durch
- Bedrohungsmodellierung — Vermitteln Sie Anwendungsinformationen, indem Sie Quellcode, Datenflüsse und Architektur analysieren
- Kontext für Penetrationstests — Vermitteln Sie durch die Analyse des Quellcodes Anwendungsverständnis für Penetrationstests
- Automatisierte Problembehebung — Reichen Sie Merge-Anfragen mit Korrekturen für Sicherheitslücken ein, die bei Sicherheitsbewertungen entdeckt wurden

Für GitLab die Verbindung mit dem AWS Security Agent muss ein GitLab Zugriffstoken mit den entsprechenden Berechtigungen bereitgestellt und die Verbindung anschließend in der AWS-Konsole registriert werden.

Wie funktioniert GitLab die Integration

Die Analyse von Anfragen zur Zusammenführung erfolgt innerhalb von GitLab. Nachdem Sie Ihr Zugriffstoken bereitgestellt, Projekte verbunden und Kommentare zur Codeüberprüfung in der AWS-Managementkonsole aktiviert haben, scannt der AWS Security Agent automatisch die Änderungen in jeder neuen Merge-Anfrage (ein Differentialscan nur des geänderten Codes) und veröffentlicht die Ergebnisse als Kommentare zur Merge-Anfrage.

Vollständige Codeprüfungen, bei denen die gesamte Codebasis eines Projekts gescannt wird, werden in der AWS Security Agent-Webanwendung erstellt und ausgeführt, nicht in GitLab.

Penetrationstests und Bedrohungsmodellierung werden innerhalb der AWS Security Agent-Webanwendung initiiert. Benutzer geben Zieldomänen an und wählen verbundene Repositorys aus, um den Anwendungskontext bereitzustellen. Wenn Sie die automatische Problembehebung aktivieren, können Benutzer den AWS Security Agent auffordern, Fehler zu korrigieren, indem sie Merge-Anfragen für verbundene Repositorys öffnen.

> **Note**
>
> Die automatische Behebung ist für öffentliche GitLab Repositorys nicht verfügbar, um zu verhindern, dass Sicherheitslücken offengelegt werden, bevor sie behoben wurden.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein GitLab.com Konto mit Maintainer- oder Inhaberzugriff auf die Projekte, die Sie verbinden möchten
- Ein GitLab Zugriffstoken mit den Bereichen, die für Ihren Verbindungstyp erforderlich sind:
  - Persönlich — Ein persönliches Zugriffstoken mit allen Leseberechtigungen und der `api` Erlaubnis.
  - Gruppe — Ein Gruppenzugriffstoken mit den `read_repository` Bereichen `read_api` und.
- Berechtigungen zur Konfiguration von Integrationen in der AWS Security Agent Management Console

> **Important**
>
> Legen Sie den Ablauf des Tokens auf maximal 365 Tage nach dem aktuellen Datum fest.

> **Note**
>
> GitLab Personal Access Tokens können für mehrere AWS-Konten verwendet werden. Im Gegensatz GitHub zu Atlassian-Integrationen gibt es keine Einschränkungen, wenn es darum geht, dasselbe GitLab Konto mit mehreren AWS Security Agent-Instanzen zu verbinden.

Registriere eine Verbindung GitLab

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie Add Integration (Integration hinzufügen) aus.
3. Wählen Sie GitLabund dann Weiter.
4. Wählen Sie unter Kontotyp auswählen eine der folgenden Optionen aus:
   - Persönlich — Connect dein individuelles GitLab Benutzerkonto.
   - Gruppe — Connect eine GitLab Gruppe, die mehrere Projekte enthält. Geben Sie Ihre Gruppen-ID ein.
5. Fügen Sie Ihr Zugriffstoken in das Feld GitLab Zugriffstoken ein.
6. Geben Sie im Feld Registrierungsname einen aussagekräftigen Namen für diese Verbindung ein, z. B. `Engineering-Team-GitLab` Gültige Zeichen sind Buchstaben, Zahlen, Punkte, Unterstriche und Bindestriche.
7. Wählen Sie Connect aus.

   Sie kehren zur Seite Integrationen zurück, auf der die neue Verbindung mit ihrem Registrierungsnamen angezeigt wird.

Fehler bei der Integration beheben GitLab

Wenn Sie während des GitLab Integrationsprozesses auf Probleme stoßen, verwenden Sie die folgenden Anleitungen, um häufig auftretende Probleme zu lösen.

Ungültiges oder abgelaufenes Token

Symptome

- Die Integration kann keine Verbindung herstellen
- Die zuvor funktionierende Integration funktioniert nicht mehr
- Fehlermeldung, die auf einen Authentifizierungsfehler hinweist

Auflösung

1. Navigieren Sie in GitLab zu Ihren Benutzereinstellungen und wählen Sie Zugriffstoken aus.
2. Stellen Sie sicher, dass Ihr Token nicht abgelaufen ist.
3. Stellen Sie sicher, dass das Token die für seinen Typ erforderlichen Gültigkeitsbereiche hat.
4. Wenn das Token abgelaufen ist, erstellen Sie ein neues Token und aktualisieren Sie die Integration in der AWS-Konsole.

Ratenbegrenzung

Symptome

- Zeitweise auftretende Fehler bei der Codeüberprüfung
- Analyse verzögerter Zusammenführungsanforderungen

Auflösung

- GitLab wendet Ratenbegrenzungen auf API-Anfragen an. Wenn Sie eine große Anzahl von Repositorys oder häufige Merge-Anfragen haben, können einige Anfragen gedrosselt werden.
- Warte, bis das Fenster für die Ratenbegrenzung zurückgesetzt wurde, oder kontaktiere den GitLab Support, um deine Ratenlimits zu erhöhen.

Nächste Schritte

Nach dem Herstellen GitLab der Verbindung mit dem AWS Security Agent:

- Navigieren Sie zu dem Agent-Bereich, in dem Sie diese Repositorys verwenden möchten

- Wählen Sie „Codeüberprüfung aktivieren“ oder „Penetrationstests einrichten“, um bestimmte Projekte mit Ihrem Agent Space zu verbinden und deren Nutzung zu konfigurieren
- Aktivieren Sie die Codekorrektur, damit der AWS Security Agent Merge-Anfragen mit Behebung von Sicherheitslücken einreichen kann
- Informationen zu privat gehosteten GitLab Instances finden Sie unter the section called “Connect den AWS Security Agent mit GitLab Self-Managed”

Connect den AWS Security Agent mit GitLab Self-Managed

Connect Ihren AWS Security Agent mit einer GitLab Self-Managed Instance, um Codeüberprüfung, Bedrohungsmodellierung, Penetrationstests und automatische Problembehebungsfunktionen für Repositorys zu aktivieren, die auf Ihrer eigenen Infrastruktur gehostet werden.

GitLab Self-Managed Die Integration funktioniert genauso wie GitLab Cloud (siehethe section called “AWS Security Agent mit GitLab Repositorys Connect”) mit zusätzlicher Konfiguration für die Netzwerkkonnektivität zu Ihrer privaten Instance. Bevor Sie beginnen, sollten Sie the section called “So funktionieren Integrationen mit Agent Spaces” sich darüber informieren, wie eine Registrierung in allen Agent Spaces wiederverwendet und funktionsübergreifend gemeinsam genutzt wird.

> **Note**
>
> GitLab Self-Managed wird über die GitLabIntegration registriert, nicht über einen separaten Integrationstyp. Im Registrierungsablauf wählen Sie „ GitLab Selbst gehosteten Endpunkt verwenden“ aus und geben Ihre Instanz-URL an. Der in der Cloud gehostete GitLab Flow wird unter beschrieben. the section called “AWS Security Agent mit GitLab Repositorys Connect”

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Eine GitLab Self-Managed Instanz, die entweder:
  - Öffentlich über das Internet zugänglich, ODER
  - Über eine private Verbindung zugänglich (siehethe section called “Connect zur privat gehosteten Quellcodeverwaltung her”)
- Ein GitLab Zugriffstoken mit den Bereichen, die für Ihren Verbindungstyp erforderlich sind:

  - Persönlich — Ein persönliches Zugriffstoken mit allen Leseberechtigungen und der `api` Erlaubnis.
  - Gruppe — Ein Gruppenzugriffstoken mit den `read_repository` Bereichen `read_api` und.
- Betreuer- oder Eigentümerzugriff auf die Projekte, mit denen Sie eine Verbindung herstellen möchten
- Ihre GitLab Instance muss HTTPS-Traffic mit einer TLS-Mindestversion von 1.2 bereitstellen

> **Note**
>
> Wenn Ihre GitLab Self-Managed Instance TLS-Zertifikate verwendet, die von einer privaten Zertifizierungsstelle ausgestellt wurden, können Sie den PEM-encoded öffentlichen Schlüssel des Zertifikats angeben, wenn Sie eine private Verbindung herstellen. Dadurch kann der AWS Security Agent der TLS-Verbindung zu Ihrer Instance vertrauen.

Registrieren Sie eine GitLab Self-Managed Verbindung

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie Add Integration (Integration hinzufügen) aus.
3. Wählen Sie GitLabund dann Weiter aus.
4. Wählen Sie unter Kontotyp auswählen die Option Persönlich oder Gruppe aus. Wenn Sie Gruppe auswählen, geben Sie Ihre Gruppen-ID ein.
5. Wählen Sie GitLab Selbst gehosteten Endpunkt verwenden aus.
6. Geben Sie im Feld GitLab selbst gehostete Endpunkt-URL beispielsweise die URL Ihrer Instanz ein. `https://gitlab.example.com`
7. Wenn Ihre Instance nicht öffentlich zugänglich ist, wählen Sie Über eine private Verbindung mit Endpunkt Connect und wählen Sie dann eine bestehende private Verbindung aus oder erstellen Sie eine neue. Siehe the section called “Connect zur privat gehosteten Quellcodeverwaltung her”.
8. Fügen Sie im Feld Zugriffstoken Ihr GitLab Zugriffstoken ein.
9. Geben Sie im Feld Registrierungsname einen aussagekräftigen Namen für diese Verbindung ein. Gültige Zeichen sind Buchstaben, Zahlen, Punkte, Unterstriche und Bindestriche.
10. Wählen Sie Connect aus.

Sie kehren zur Seite Integrationen zurück, auf der die neue Verbindung mit ihrem Registrierungsnamen angezeigt wird.

Private Konnektivität

Wenn Ihre GitLab Self-Managed Instance nicht öffentlich zugänglich ist, müssen Sie eine private Verbindung herstellen, bevor Sie die Integration registrieren. Eine the section called “Connect zur privat gehosteten Quellcodeverwaltung her” ausführliche Anleitung finden Sie unter.

> **Important**
>
> Service-managed Für private Verbindungen muss die GitLab Self-Managed Instance in demselben AWS-Konto ausgeführt werden, in dem der Agent Space erstellt wurde. Verwenden Sie für den kontoübergreifenden Zugriff eine selbstverwaltete private Verbindung, bei der Sie Ihre eigene VPC-Lattice-Ressourcenkonfiguration angeben.

GitLab Self-Managed Probleme bei der Integration beheben

Zusätzlich zu den unter beschriebenen Schritten zur Fehlerbehebung sind die folgenden Probleme spezifisch für selbstverwaltete Instanzen: the section called “AWS Security Agent mit GitLab Repositorys Connect”

Instanz nicht erreichbar

Symptome

- Die Verbindung schlägt mit einem Timeout oder einem Netzwerkfehler fehl
- Die Integration funktionierte zuvor, funktioniert aber nicht mehr

Auflösung

- Stellen Sie sicher, dass Ihre GitLab Instance läuft und darauf zugegriffen werden kann
- Wenn Sie eine private Verbindung verwenden, stellen Sie sicher, dass das VPC Lattice Resource Gateway fehlerfrei ist und die ENIs über Netzwerkkonnektivität zu Ihrer Instance verfügen
- Stellen Sie sicher, dass Sicherheitsgruppen Datenverkehr auf dem konfigurierten Port zulassen
- Stellen Sie sicher, dass das TLS-Zertifikat gültig und nicht abgelaufen ist

Fehler beim TLS-Zertifikat

Symptome

- Die Verbindung schlägt mit einem SSL/TLS Fehler fehl

Auflösung

- Stellen Sie sicher, dass Ihre Instance HTTPS mit TLS 1.2 oder höher bereitstellt
- Wenn Sie eine private Zertifizierungsstelle verwenden, stellen Sie sicher, dass der PEM-encoded öffentliche Schlüssel bei der Einrichtung der privaten Verbindung bereitgestellt wurde
- Stellen Sie sicher, dass das Zertifikat nicht abgelaufen ist

Nächste Schritte

Nach dem Herstellen GitLab Self-Managed der Verbindung mit dem AWS Security Agent:

- Navigieren Sie zu dem Agent-Bereich, in dem Sie diese Repositorys verwenden möchten
- Wählen Sie „Codeüberprüfung aktivieren“ oder „Penetrationstests einrichten“, um bestimmte Projekte miteinander zu verbinden
- Aktivieren Sie die Codekorrektur für Korrekturen, die auf Merge-Anfragen basieren

Eine GitLab Integration entfernen

Entfernen Sie eine GitLab Integration, wenn Sie den AWS Security Agent nicht mehr benötigen, um von einem bestimmten GitLab Konto oder einer bestimmten Gruppe aus auf Repositorys zuzugreifen.

Voraussetzungen für die Entfernung

Bevor Sie eine GitLab Integration entfernen, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Sie haben überprüft, mit welchen Agent Spaces Repositorys über diese Integration verbunden sind
- Ich habe die Auswirkungen verstanden: Durch das Entfernen dieser Integration werden die Repository-Verbindungen für die Codeüberprüfung, den Kontext für Penetrationstests, die Bedrohungsmodellierung und die automatische Problembehebung in allen Agent Spaces, die Repositorys aus dieser Integration verwenden, unterbrochen

Die Integration aus dem AWS Security Agent entfernen

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Suchen Sie die GitLab Integration, die Sie entfernen möchten.
3. Wählen Sie die Integration aus, indem Sie darauf klicken.
4. Wählen Sie Remove (Entfernen) aus.
5. Überprüfen Sie das Bestätigungsdialogfeld und wählen Sie Entfernen bestätigen.

> **Note**
>
> Nachdem Sie die Integration entfernt haben, möchten Sie möglicherweise auch das Personal Access Token widerrufen GitLab , um weiteren Zugriff zu verhindern. Navigieren Sie zu Ihren GitLab Benutzereinstellungen und löschen oder widerrufen Sie das Token.

Derselbe Prozess gilt für GitLab Self-Managed Integrationen.

AWS Security Agent mit GitHub Enterprise Server Connect

Connect Ihren AWS Security Agent mit einer GitHub Enterprise Server (GHES) -Instance, um Codeüberprüfung, Bedrohungsmodellierung, Penetrationstests und automatische Problembehebungsfunktionen für Repositorys zu ermöglichen, die auf Ihrer eigenen Infrastruktur gehostet werden.

GitHub Die Enterprise Server-Integration bietet dieselben Funktionen wie in der Cloud gehostet GitHub (siehe AWS Security Agent mit GitHub Repositorys Connect) mit zusätzlicher Konfiguration für die Netzwerkkonnektivität zu Ihrer selbst gehosteten Instance. Bevor Sie beginnen, sollten Sie the section called “So funktionieren Integrationen mit Agent Spaces” sich darüber informieren, wie eine Registrierung in allen Agent Spaces wiederverwendet und funktionsübergreifend gemeinsam genutzt wird.

> **Note**
>
> GitHub Enterprise Server wird über die GitHubIntegration registriert, nicht über einen separaten Integrationstyp. Im Registrierungsablauf wählen Sie GitHub Enterprise Server als Instanztyp. Der in der Cloud gehostete GitHub.com Flow wird unter beschriebenthe section called “AWS Security Agent mit GitHub Repositorys Connect”.

So funktioniert die GitHub Enterprise Server-Integration

Die Pull-Request-Analyse erfolgt innerhalb von GitHub Enterprise Server. Nachdem Sie die Verbindung autorisiert, Repositorys verbunden und Kommentare zur Codeüberprüfung in der AWS-Managementkonsole aktiviert haben, scannt der AWS Security Agent automatisch die Änderungen in jeder neuen Pull-Anfrage (ein Differenzscan nur des geänderten Codes) und veröffentlicht die Ergebnisse als Pull-Request-Kommentare.

Vollständige Code-Reviews, bei denen die gesamte Codebasis eines Repositorys gescannt wird, werden in der AWS Security Agent-Webanwendung erstellt und ausgeführt, nicht in GitHub Enterprise Server.

Penetrationstests und Bedrohungsmodellierung werden innerhalb der AWS Security Agent-Webanwendung initiiert. Benutzer geben Zieldomänen an und wählen verbundene Repositorys aus, um den Anwendungskontext bereitzustellen. Wenn Sie die automatische Problembehebung aktivieren, können Benutzer den AWS Security Agent auffordern, Fehler zu korrigieren, indem sie Pull-Anfragen an verbundene Repositorys öffnen.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Eine GitHub Enterprise Server-Instance, die entweder:
  - Öffentlich über das Internet zugänglich, ODER
  - Über eine private Verbindung zugänglich (siehethe section called “Connect zur privat gehosteten Quellcodeverwaltung her”)
- Zugriff durch Site-Administrator oder Organisationsadministrator auf Ihrer GHES-Instanz
- Ihre GHES-Instanz muss HTTPS-Verkehr mit einer TLS-Mindestversion von 1.2 bereitstellen
- Berechtigungen zur Konfiguration von Integrationen in der AWS Security Agent Management Console

> **Note**
>
> GitHub Enterprise Server-Integrationen können für mehrere AWS-Konten verwendet werden.

Registrieren Sie eine GitHub Enterprise Server-Verbindung

Die Registrierung einer GitHub Enterprise Server-Verbindung verwendet einen OAuth-based Autorisierungsablauf.

> **Important**
>
> Führen Sie alle Schritte dieses Vorgangs aus, ohne Ihren Browser zu schließen oder zu einer anderen Seite zu navigieren. Wenn der Registrierungsprozess unterbrochen wird, müssen Sie möglicherweise von vorne beginnen.

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie Add Integration (Integration hinzufügen) aus.
3. Wählen Sie GitHubund dann Weiter aus.
4. Wählen Sie unter Instanztyp die Option GitHub Enterprise Server aus.
5. Geben Sie im Feld GitHub Enterprise Server-URL beispielsweise die HTTPS-URL Ihrer Instanz ein`https://github.example.com`.
6. Wenn Ihre Instance nicht öffentlich zugänglich ist, wählen Sie Über eine private Verbindung mit Endpunkt Connect und wählen Sie dann eine bestehende private Verbindung aus oder erstellen Sie eine neue. Siehe the section called “Connect zur privat gehosteten Quellcodeverwaltung her”.
7. Konfigurieren Sie im Bereich Registrierungsdetails die folgenden Felder:
   a. Registrierungsname — Geben Sie einen aussagekräftigen Namen für diese Verbindung ein. Gültige Zeichen sind Buchstaben, Zahlen, Punkte, Unterstriche und Bindestriche.
   b. GitHub Kontotyp — Wählen Sie Organisation oder Benutzer aus.
   c. Name der Organisation (wird nur angezeigt, wenn Sie Organisation ausgewählt haben) — Geben Sie den genauen Namen Ihrer GitHub Enterprise Server-Organisation ein. Bei den Namen muss die Groß- und Kleinschreibung beachtet werden.
8. Wählen Sie Connect aus.

> **Note**
>
> Der AWS Security Agent leitet Sie von der Konsole weg, um die Autorisierung mit Ihrer GitHub Enterprise Server-Instance abzuschließen. Nach Abschluss der Autorisierung

kehren Sie zur Konsole zurück und die neue Verbindung wird auf der Integrationsseite angezeigt.

Private Konnektivität

Wenn Ihre GitHub Enterprise Server-Instanz nicht öffentlich zugänglich ist, müssen Sie vor der Registrierung der Integration eine private Verbindung herstellen. Eine the section called “Connect zur privat gehosteten Quellcodeverwaltung her” ausführliche Anleitung finden Sie unter.

Important

Service-managed Für private Verbindungen muss die GHES-Instance in demselben AWS-Konto ausgeführt werden, in dem der Agent Space erstellt wurde. Verwenden Sie für den kontoübergreifenden Zugriff eine selbstverwaltete private Verbindung.

Note

Wenn Ihre GHES-Instanz TLS-Zertifikate verwendet, die von einer privaten Zertifizierungsstelle ausgestellt wurden, geben Sie beim Erstellen der privaten Verbindung den PEM-encoded öffentlichen Schlüssel des Zertifikats an.

Problembehebung bei der GitHub Enterprise Server-Integration

Wenn Sie Probleme bei der Verbindung von AWS Security Agent mit GitHub Enterprise Server haben, verwenden Sie die folgenden Anleitungen, um häufig auftretende Probleme zu diagnostizieren und zu lösen.

Fehler bei der OAuth-Umleitung

Symptome

- Browserumleitungen schlagen während des Autorisierungsflusses fehl
- Nach der Autorisierung auf GHES wird eine Fehlerseite angezeigt

Auflösung

- Stellen Sie sicher, dass Ihre GHES-Instanz von Ihrem Browser aus zugänglich ist
- Stellen Sie sicher, dass die OAuth-Callback-URL korrekt konfiguriert ist
- Starten Sie den Integrationsprozess von Anfang an neu

Instanz nicht erreichbar

Symptome

- Die Verbindung schlägt mit einem Timeout oder einem Netzwerkfehler fehl

Auflösung

- Stellen Sie sicher, dass Ihre GHES-Instanz läuft und darauf zugegriffen werden kann
- Wenn Sie eine private Verbindung verwenden, überprüfen Sie die VPC-Lattice-Konnektivität
- Stellen Sie sicher, dass Sicherheitsgruppen Datenverkehr auf dem konfigurierten Port zulassen
- Stellen Sie sicher, dass das TLS-Zertifikat gültig ist (mindestens TLS 1.2)

Nächste Schritte

Nach dem Verbinden von GitHub Enterprise Server mit AWS Security Agent:

- Navigieren Sie zu dem Agent-Bereich, in dem Sie diese Repositorys verwenden möchten
- Wählen Sie „Codeüberprüfung aktivieren“ oder „Penetrationstests einrichten“, um bestimmte Repositorys zu verbinden
- Aktivieren Sie die Codekorrektur, damit der AWS Security Agent Pull-Anfragen mit Behebung von Sicherheitslücken senden kann

Eine GitHub Enterprise Server-Integration entfernen

Entfernen Sie eine GitHub Enterprise Server-Integration, wenn Sie den AWS Security Agent nicht mehr benötigen, um von einer bestimmten GHES-Instance aus auf Repositorys zuzugreifen.

Voraussetzungen für die Entfernung

Vor dem Entfernen einer GHES-Integration:

- Prüfen Sie, mit welchen Agent Spaces Repositorys über diese Integration verbunden sind
- Machen Sie sich mit den Auswirkungen vertraut: Durch das Entfernen werden die Codeprüfung, der Kontext der Penetrationstests, die Bedrohungsmodellierung und die automatische Problembehebung für alle verbundenen Repositorys beeinträchtigt

Entfernen Sie die Integration

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Suchen Sie die GitHub Enterprise Server-Integration, die Sie entfernen möchten.
3. Wählen Sie die Integration aus.
4. Wählen Sie Remove (Entfernen) aus.
5. Überprüfen Sie das Bestätigungsdialogfeld und wählen Sie Entfernen bestätigen.

> **Note**
>
> Nach dem Entfernen möchten Sie möglicherweise auch die OAuth-Anwendung auf Ihrer GHES-Instanz sperren. Navigieren Sie zu Ihren GHES-Organisationseinstellungen und entfernen Sie die autorisierte Anwendung.

Finde deine Atlassian-Installations-ID

Bevor du eine Bitbucket- oder Confluence-Integration in der AWS-Konsole registrierst, musst du die Installations-ID der AWS Security Agent Forge-App finden, die auf deiner Atlassian-Site installiert ist. Du fügst diese ID in den Registrierungsablauf ein.

Für Bitbucket

1. Navigiere in Bitbucket zu deinen Workspace-Einstellungen.
2. Wähle in der Seitenleiste Forge-Apps aus.
3. Suchen Sie Connect with AWS Security Agent in der Liste der installierten Apps.
4. Wählen Sie In die Zwischenablage kopieren, um die Installations-ID zu kopieren.

Für Confluence

1. Navigieren Sie in Confluence zur Confluence-Administration.
2. Wähle Apps in der Seitenleiste aus.
3. Suchen Sie Connect with AWS Security Agent in der Liste der installierten Apps.
4. Wählen Sie In die Zwischenablage kopieren, um die Installations-ID zu kopieren.

Sie benötigen diese Installations-ID, wenn Sie die Registrierung in der AWS Security Agent Management Console abschließen.

AWS Security Agent mit Bitbucket-Repositorys Connect

Connect deinen AWS Security Agent mit Bitbucket Cloud-Repositorys, um Funktionen zur Codeüberprüfung, Bedrohungsmodellierung, Penetrationstests und automatische Problembehebung zu nutzen. Bevor du anfängst, informiere dich darüberthe section called “So funktionieren Integrationen mit Agent Spaces”, wie eine Registrierung in allen Agent Spaces wiederverwendet und funktionsübergreifend gemeinsam genutzt wird.

Die Bitbucket-Integration dient mehreren Zwecken:

- Codeüberprüfung — Analysieren Sie automatisch die Codeänderungen in jedem Pull Request anhand der Sicherheitsanforderungen Ihres Unternehmens und führen Sie bei Bedarf vollständige Repository-Scans durch
- Bedrohungsmodellierung — Vermitteln Sie Anwendungsinformationen, indem Sie Quellcode, Datenflüsse und Architektur analysieren
- Kontext für Penetrationstests — Vermitteln Sie das Anwendungsverständnis für Penetrationstests
- Automatisierte Problembehebung — Senden Sie Pull-Requests mit Korrekturen für Sicherheitslücken, die bei Sicherheitsbewertungen entdeckt wurden

Um Bitbucket mit dem AWS Security Agent zu verbinden, musst du die App AWS Security Agent Forge auf deiner Atlassian-Site installieren und den OAuth-Autorisierungsablauf abschließen.

> **Note**
>
> AWS Security Agent unterstützt nur Bitbucket Cloud. Bitbucket Server und Bitbucket Data Center werden nicht unterstützt.

Wie funktioniert die Bitbucket-Integration

Die Pull-Request-Analyse erfolgt innerhalb von Bitbucket. Nachdem Sie die Forge-App installiert, Repositorys verbunden und Kommentare zur Codeüberprüfung in der AWS-Managementkonsole aktiviert haben, scannt der AWS Security Agent automatisch die Änderungen in jeder neuen Pull-Anfrage (ein Differenzscan nur des geänderten Codes) und veröffentlicht die Ergebnisse als Pull-Request-Kommentare.

Vollständige Code-Reviews, bei denen die gesamte Codebasis eines Repositorys gescannt wird, werden in der AWS Security Agent-Webanwendung erstellt und ausgeführt, nicht in Bitbucket.

Penetrationstests und Bedrohungsmodellierung werden innerhalb der AWS Security Agent-Webanwendung initiiert. Benutzer geben Zieldomänen an und wählen verbundene Repositorys aus, um den Anwendungskontext bereitzustellen.

> **Note**
>
> Automatische Problembehebungen sind für öffentliche Bitbucket-Repositorys nicht verfügbar, um zu verhindern, dass Sicherheitslücken offengelegt werden, bevor sie behoben wurden.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein Bitbucket Cloud-Workspace mit Administratorzugriff
- Ein Atlassian-Konto mit Site-Administratorrechten
- Berechtigungen zur Konfiguration von Integrationen in der AWS Security Agent Management Console

> **Important**
>
> Eine Atlassian-Site kann nur mit einem AWS-Konto pro Region verknüpft werden. Wenn du dieselbe Bitbucket-Site mit dem AWS Security Agent in einem anderen AWS-Konto in derselben Region verbinden musst, musst du zuerst die bestehende Integration entfernen.

> **Note**
>
> Für diese Integration gelten die Preise der Atlassian Forge-App. Weitere Informationen findest du in der Atlassian-Dokumentation unter Preise für die Forge-Plattform.

Registriere eine Bitbucket-Verbindung

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie Add Integration (Integration hinzufügen) aus.
3. Wählen Sie Bitbucket und dann Weiter aus.
4. Installiere die AWS Security Agent Forge-App in deinem Bitbucket-Workspace und befolge dabei die Anweisungen auf dem Bildschirm. Kopieren Sie nach der Installation die Installations-ID. Siehe the section called “Finde deine Atlassian-Installations-ID”.
5. Füge in das Feld Installations-ID die Installations-ID ein, die du aus Bitbucket kopiert hast.
6. Gib im Feld Bitbucket-Workspace beispielsweise deinen Bitbucket-Workspace-Slug ein. `acme-corp`
7. Klicken Sie auf Authorize.

   Du wirst zu Atlassian weitergeleitet, um den AWS Security Agent für den Zugriff auf deinen Bitbucket-Workspace zu autorisieren. Nach Abschluss der Autorisierung kehrst du zur Konsole zurück.
8. Geben Sie im Abschnitt Registrierungsdetails einen Registrierungsnamen für diese Verbindung ein. Gültige Zeichen sind Buchstaben, Zahlen, Punkte, Unterstriche und Bindestriche.
9. Wählen Sie Connect aus.

> **Note**
>
> Verzögerung bei der Bereitstellung nach der Verbindung
>
> Nachdem du Connect ausgewählt hast, kann die Bereitstellung auf Atlassian-Seite einige Minuten dauern. Während dieser Zeit meldet die Integration den Status Ausstehend. Wenn Sie versuchen, Repositorys auszuwählen oder die Codeüberprüfung zu aktivieren, bevor die Bereitstellung abgeschlossen ist, wird möglicherweise eine Meldung angezeigt, dass

die Installation noch nicht verfügbar ist. Warten Sie einige Minuten und versuchen Sie es dann erneut.

Fehlerbehebung bei der Bitbucket-Integration

Wenn du während des Bitbucket-Integrationsprozesses auf Probleme stößt, befolge die folgenden Anleitungen, um häufig auftretende Probleme zu lösen.

Die Registrierung konnte nicht abgeschlossen werden

Wenn der Registrierungsprozess unterbrochen wird (Browser geschlossen, Sitzungs-Timeout), ist die Forge-App möglicherweise auf deiner Atlassian-Website installiert, aber nicht in der AWS-Konsole registriert.

Auflösung

- Kehren Sie zur AWS Security Agent-Konsole zurück und starten Sie den Integrationsprozess neu.
- Die Forge-App bleibt installiert und muss nicht neu installiert werden.

Site ist bereits mit einem anderen AWS-Konto verbunden

Symptome

- Fehler beim Hinweis, dass die Atlassian-Site bereits mit einem anderen Konto verknüpft ist

Auflösung

- Eine Atlassian-Site kann nur mit einem AWS-Konto pro Region verbunden werden.
- Identifizieren Sie, welches AWS-Konto über die bestehende Integration verfügt, und verwenden Sie dieses Konto, oder entfernen Sie zuerst die bestehende Integration.

Die Installation ist nicht verfügbar

Symptome

- Wenn du Repositorys auswählst oder die Codeüberprüfung aktivierst, siehst du eine Meldung, dass sich die Bitbucket-Installation im Status befindet`PENDING_INSTALLATION`, nicht. `AVAILABLE`

Auflösung

- Die Installation wird weiterhin auf Atlassian-Seite bereitgestellt. Die Bereitstellung ist normalerweise innerhalb weniger Minuten abgeschlossen. Warten Sie ein paar Minuten und versuchen Sie es dann erneut.
- Wenn der Status nach einigen Minuten nicht verfügbar ist, entfernen Sie die Integration und registrieren Sie sie erneut. Bevor du dich erneut registrierst, deinstalliere die Forge-App aus deinem Bitbucket-Workspace. Eine Anleitung findest du unter Eine Bitbucket-Integration entfernen. Wenn du dich erneut registrierst, ohne die vorherige Forge-App zu deinstallieren, kann die Installation im Status „Ausstehend“ hängen bleiben.

Nächste Schritte

Nachdem Sie Bitbucket mit dem AWS Security Agent verbunden haben:

- Navigiere zu dem Agent-Bereich, in dem du diese Repositorys verwenden möchtest
- Wählen Sie Codeüberprüfung aktivieren oder Penetrationstests einrichten, um bestimmte Repositorys mit Ihrem Agent Space zu verbinden
- Aktivieren Sie die Codekorrektur, damit der AWS Security Agent Pull-Anfragen mit Behebung von Sicherheitslücken senden kann

Eine Bitbucket-Integration entfernen

Entferne eine Bitbucket-Integration, wenn du den AWS Security Agent nicht mehr benötigst, um von einem bestimmten Bitbucket-Workspace aus auf Repositorys zuzugreifen.

Voraussetzungen für die Entfernung

Bevor du eine Bitbucket-Integration entfernst:

- Überprüfe, mit welchen Agent Spaces Repositorys über diese Integration verbunden sind
- Machen Sie sich mit den Auswirkungen vertraut: Durch das Entfernen werden die Codeüberprüfung, der Kontext der Penetrationstests, die Bedrohungsmodellierung und die automatische Problembehebung für alle verbundenen Repositorys beeinträchtigt

Schritt 1: Entfernen Sie die Integration aus dem AWS Security Agent

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Suchen Sie die Bitbucket-Integration, die Sie entfernen möchten.
3. Wähle die Integration aus.
4. Wählen Sie Remove (Entfernen) aus.
5. Überprüfen Sie das Bestätigungsdialogfeld und wählen Sie Entfernen bestätigen.

Schritt 2: Deinstallieren Sie die Forge-App

Nachdem du die Integration aus dem AWS Security Agent entfernt hast, deinstalliere die Forge-App von deiner Atlassian-Site:

1. Navigiere in Bitbucket zu den Workspace-Einstellungen.
2. Wähle Forge-Apps aus.
3. Suchen Sie Connect with AWS Security Agent.
4. Wählen Sie Deinstallieren.

> **Important**
>
> Die Forge-App wird nicht automatisch deinstalliert
>
> Durch das Entfernen der Integration in der AWS Security Agent-Konsole wird die Forge-App nicht von deiner Atlassian-Site deinstalliert. Wenn du denselben Bitbucket-Workspace erneut registrieren möchtest, musst du zuerst die Forge-App deinstallieren. Andernfalls kann die neue Installation im Status „Ausstehend“ hängen bleiben.

AWS Security Agent mit Confluence Connect

Connect Sie Ihren AWS Security Agent mit Confluence Cloud, um den Dokumentationskontext für Sicherheitsbewertungen bereitzustellen. Im Gegensatz zu Codeanbietern dient Confluence als Dokumentationsquelle, die Bedrohungsmodelle, Architekturdokumente, API-Spezifikationen und andere Materialien bereitstellt, die die Qualität von Sicherheitsüberprüfungen verbessern. Bevor

Sie beginnen, sollten Sie the section called “So funktionieren Integrationen mit Agent Spaces” sich darüber informieren, wie eine Registrierung in Agent Spaces wiederverwendet wird.

Die Confluence-Integration dient mehreren Zwecken:

- Kontext der Entwurfsprüfung — Stellen Sie Architekturdokumente und Entwurfsspezifikationen für die Überprüfung von Sicherheitsentwürfen bereit
- Bedrohungsmodellierung — Stellen Sie bestehende Bedrohungsmodelle und Systemdokumentationen für die Bedrohungsanalyse bereit
- Kontext für Penetrationstests — Stellen Sie Anwendungsdokumentation bereit, um ein tieferes Verständnis während der Penetrationstests zu gewährleisten

Um Confluence mit dem AWS Security Agent zu verbinden, müssen Sie die AWS Security Agent Forge-App auf Ihrer Atlassian-Site installieren und den OAuth-Autorisierungsablauf abschließen.

> **Note**
>
> AWS Security Agent unterstützt nur Confluence Cloud. Confluence Data Center und Confluence Server werden nicht unterstützt.

Wie funktioniert die Confluence-Integration

Confluence ist eher ein Dokumentationsanbieter als ein Quellcode-Anbieter. Nachdem Sie die Forge-App installiert und Bereiche oder Seiten in der AWS-Managementkonsole verbunden haben, kann der AWS Security Agent auf Ihre Confluence-Inhalte zugreifen, um bei Sicherheitsbewertungen Kontext bereitzustellen.

Der AWS Security Agent liest Seiteninhalte, um Ihre Anwendungsarchitektur, Sicherheitsanforderungen und Designentscheidungen zu verstehen. Dieser Kontext verbessert die Qualität und Relevanz der Sicherheitsergebnisse bei Entwurfs-, Code- und Penetrationstests.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Eine Confluence Cloud-Site mit Administratorzugriff
- Ein Atlassian-Konto mit Site-Administratorrechten

- Berechtigungen zur Konfiguration von Integrationen in der AWS Security Agent Management Console

> **Important**
>
> Eine Atlassian-Site kann nur mit einem AWS-Konto pro Region verknüpft werden. Wenn Sie dieselbe Confluence-Site mit dem AWS Security Agent in einem anderen AWS-Konto in derselben Region verbinden müssen, müssen Sie zuerst die bestehende Integration entfernen.

> **Note**
>
> Für diese Integration gelten die Preise der Atlassian Forge-App. Weitere Informationen findest du in der Atlassian-Dokumentation unter Preise für die Forge-Plattform.

Registriere eine Confluence-Verbindung

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie Add Integration (Integration hinzufügen) aus.
3. Wählen Sie Confluence und dann Weiter aus.
4. Installiere die AWS Security Agent Forge-App auf deiner Atlassian-Site und befolge dabei die Anweisungen auf dem Bildschirm. Kopiere nach der Installation die Installations-ID. Siehe the section called “Finde deine Atlassian-Installations-ID”.
5. Geben Sie im Feld Confluence-Site-URL beispielsweise Ihre Site-URL ein. `https://acme.atlassian.net`
6. Fügen Sie in das Feld Installations-ID die Installations-ID ein, die Sie aus Confluence kopiert haben.
7. Klicken Sie auf Authorize.

   Du wirst zu Atlassian weitergeleitet, um AWS Security Agent für den Zugriff auf deine Confluence-Site zu autorisieren. Nach Abschluss der Autorisierung kehren Sie zur Konsole zurück.
8. Geben Sie im Abschnitt Registrierungsdetails einen Registrierungsnamen für diese Verbindung ein. Gültige Zeichen sind Buchstaben, Zahlen, Punkte, Unterstriche und Bindestriche.

9. Wählen Sie Connect aus.

Wählen Sie Seiten für einen Agent Space aus

Nachdem Sie die Confluence-Integration registriert haben, verbinden Sie bestimmte Seiten mit einem Agent Space. Die Auswahl einer Seite gewährt dem AWS Security Agent Lesezugriff (Abruf) auf den Inhalt dieser Seite. Es gibt keine seitenspezifischen Funktionsoptionen — der Agent liest jede verbundene Seite. Im Überprüfungsschritt des Verbindungsassistenten können Sie alle Seiten entfernen, auf die der Agent nicht zugreifen soll.

Beheben Sie Fehler bei der Confluence-Integration

Wenn Sie während des Confluence-Integrationsprozesses auf Probleme stoßen, verwenden Sie die folgenden Anleitungen, um häufig auftretende Probleme zu lösen.

Die Registrierung konnte nicht abgeschlossen werden

Wenn der Registrierungsprozess unterbrochen wird, ist die Forge-App möglicherweise auf deiner Atlassian-Website installiert, aber nicht in der AWS-Konsole registriert.

Auflösung

- Kehren Sie zur AWS Security Agent-Konsole zurück und starten Sie den Integrationsprozess neu.
- Die Forge-App bleibt installiert und muss nicht neu installiert werden.

Die Forge-App wurde von Atlassian deinstalliert

Wenn die AWS Security Agent Forge-App von deiner Atlassian-Site deinstalliert wird, während die Integration noch in AWS Security Agent existiert:

Symptome

- Die Integration wird in der AWS-Konsole angezeigt, kann aber nicht auf Confluence-Inhalte zugreifen
- Fehler beim Versuch, Seiten aufzulisten oder zu lesen

Auflösung

- Installiere die Forge-App vom Atlassian Marketplace neu

- Wenn das Problem weiterhin besteht, entfernen Sie die Integration in der AWS-Konsole und registrieren Sie sich erneut

Site ist bereits mit einem anderen AWS-Konto verbunden

Auflösung

- Eine Atlassian-Site kann nur mit einem AWS-Konto pro Region verbunden werden.
- Identifizieren Sie, welches AWS-Konto über die bestehende Integration verfügt, und verwenden Sie dieses Konto, oder entfernen Sie zuerst die bestehende Integration.

Nächste Schritte

Nachdem Sie Confluence mit dem AWS Security Agent verbunden haben:

- Navigieren Sie zu dem Agent-Bereich, in dem Sie diese Dokumentation verwenden möchten
- Wählen Sie bestimmte Seiten aus, die als Kontext für Entwurfsprüfungen und Penetrationstests verwendet werden sollen
- Laden Sie bei Bedarf zusätzliche Dokumentation über S3 hoch (siehethe section called “Stellen Sie Agentenressourcen aus einem S3-Bucket bereit”)

Eine Confluence-Integration entfernen

Entfernen Sie eine Confluence-Integration, wenn Sie den AWS Security Agent nicht mehr benötigen, um auf Dokumentation von einer bestimmten Confluence-Site zuzugreifen.

Voraussetzungen für die Entfernung

Bevor Sie eine Confluence-Integration entfernen:

- Prüfen Sie, mit welchen Agent Spaces Seiten aus dieser Integration verbunden sind
- Verstehen Sie die Auswirkungen: Durch das Entfernen wird der Dokumentationskontext für Entwurfsprüfungen, Penetrationstests und Bedrohungsmodellierung für alle Agent Spaces, die Inhalte aus dieser Integration verwenden, unterbrochen

Schritt 1: Entfernen Sie die Integration aus dem AWS Security Agent

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Suchen Sie die Confluence-Integration, die Sie entfernen möchten.
3. Wählen Sie die Integration aus.
4. Wählen Sie Remove (Entfernen) aus.
5. Überprüfen Sie das Bestätigungsdialogfeld und wählen Sie Entfernen bestätigen.

Schritt 2: Deinstallieren Sie die Forge-App (optional)

Nachdem du die Integration aus dem AWS Security Agent entfernt hast, kannst du optional die Forge-App von deiner Atlassian-Site deinstallieren:

1. Navigiere in Confluence zur Confluence-Administration.
2. Wählen Sie Apps aus.
3. Suchen Sie Connect with AWS Security Agent.
4. Wählen Sie Deinstallieren.

Connect zur privat gehosteten Quellcodeverwaltung her

> **Important**
>
> Private Connections befindet sich in der Vorschauversion für AWS Security Agent und kann sich ändern.

AWS Security Agent kann eine Verbindung zu Quellcodeverwaltungssystemen herstellen, die in privaten Netzwerken ausgeführt werden, z. B. GitLab Self-Managed Instances und GitHub Enterprise Server. Private Verbindungen verwenden Amazon VPC Lattice, um eine sichere Konnektivität zwischen dem AWS Security Agent und Ihrer privaten Infrastruktur herzustellen, ohne Ihre Systeme dem öffentlichen Internet auszusetzen.

Wie funktionieren private Verbindungen

Eine private Verbindung erstellt einen sicheren Netzwerkpfad zwischen AWS Security Agent und einer Zielressource in Ihrer VPC. Unter der Haube verwendet AWS Security Agent Amazon VPC

Lattice, um diese Konnektivität herzustellen. VPC Lattice ist ein AWS-Anwendungsnetzwerkservice für die Verbindung, Sicherung und Überwachung des Datenverkehrs zwischen Services über VPCs und Konten hinweg, ohne dass Sie die zugrunde liegende Netzwerkinfrastruktur verwalten müssen.

Wenn Sie eine private Verbindung herstellen:

1. Sie stellen die VPC, Subnetze und (optional) Sicherheitsgruppen bereit, die Netzwerkkonnektivität zu Ihrem Zieldienst haben.
2. AWS Security Agent erstellt ein vom Service verwaltetes Ressourcen-Gateway und stellt Elastic Network Interfaces (ENIs) in den von Ihnen angegebenen Subnetzen bereit.
3. Der Agent verwendet das Ressourcen-Gateway, um den Datenverkehr über den privaten Netzwerkpfad an die IP-Adresse oder den DNS-Namen Ihres Zieldienstes weiterzuleiten.

Service-managed im Vergleich zu selbstverwalteten privaten Verbindungen

AWS Security Agent unterstützt zwei Modi für private Verbindungen:

Service-managed(empfohlen für einfache Setups):

- AWS Security Agent erstellt und verwaltet das VPC Lattice Resource Gateway für Sie.
- Der Zielservice muss in demselben AWS-Konto ausgeführt werden, in dem der Agent Space erstellt wurde.
- Kann nicht mehrere AWS-Konten umfassen.

Self-managed(für erweiterte oder kontoübergreifende Konfigurationen):

- Sie erstellen und verwalten Ihre eigene VPC Lattice-Ressourcenkonfiguration.
- Sie geben den Amazon-Ressourcennamen (ARN) Ihrer bestehenden Ressourcenkonfiguration an.
- Unterstützt den kontoübergreifenden Zugriff (der Kunde verwaltet die kontoübergreifenden Netzwerke).

Voraussetzungen

Bevor Sie eine private Verbindung herstellen, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Ein aktiver Agent Space

- Ein privat erreichbarer Zieldienst (GitLab Self-Managed oder GitHub Enterprise Server) unter einer bekannten privaten IP-Adresse oder einem bekannten DNS-Namen
- Der Zieldienst muss HTTPS-Verkehr mit einer TLS-Mindestversion von 1.2 bereitstellen
- Subnetze in Ihrer VPC (1—20 Subnetze). Wir empfehlen, Subnetze in mehreren Availability Zones auszuwählen, um eine hohe Verfügbarkeit zu gewährleisten.
- (Optional) Sicherheitsgruppen zur Steuerung des Datenverkehrs zu den ENIs (bis zu 5)

> **Note**
>
> Die folgenden Availability Zones werden von VPC Lattice nicht unterstützt:`use1-az3`,`usw1-az2`,`apne1-az3`,`apne2-az2`,, `euc1-az2euw1-az4`,`cac1-az3`. `ilc1-az2`

> **Note**
>
> Private Verbindungen sind Ressourcen auf Kontoebene. Nachdem Sie eine private Verbindung erstellt haben, können Sie sie für mehrere Integrationen und Agent Spaces wiederverwenden, die denselben Host erreichen müssen.

> **Note**
>
> Wenn Sie eine private Verbindung für einen Anbieter auswählen, der die OAuth-Authentifizierung verwendet, gilt die private Verbindung sowohl für den Provider-Endpunkt als auch für den Token-Exchange-Endpunkt. Stellen Sie sicher, dass die private Verbindung mit einer Hostadresse konfiguriert ist, die den Datenverkehr an beide Endpunkte weiterleiten kann.

Erstellen Sie eine private Verbindung

1. Navigieren Sie in der AWS Security Agent Management Console zu Integrationen.
2. Wählen Sie die Registerkarte Private Verbindungen.
3. Wählen Sie Private Verbindung erstellen.

4. Geben Sie unter Verbindungsdetails einen Namen ein. Namen können Kleinbuchstaben, Zahlen und Bindestriche enthalten und müssen mit einem Buchstaben oder einer Zahl beginnen und enden.
5. Wählen Sie im Abschnitt Verbindungsdetails aus, wie der AWS Security Agent eine Verbindung zu Ihrem Zielservice herstellt:
   - Damit der AWS Security Agent die Verbindung erstellen und verwalten kann, lassen Sie die Option Vorhandene Ressourcenkonfiguration verwenden deaktiviert und füllen Sie dann die Abschnitte Ressourcenstandort, Zugriffskontrolle und Servicezieldetails aus.
   - Um eine bereits erstellte VPC-Lattice-Ressourcenkonfiguration zu durchlaufen, wählen Sie Bestehende Ressourcenkonfiguration verwenden aus und füllen Sie dann den Abschnitt Bestehende Ressourcenkonfiguration aus. Die vom Service verwalteten Abschnitte gelten nicht.
6. (Service-managed) Unter Ressourcenstandort:
   a. Wählen Sie die VPC aus, in der sich Ihre Ressource befindet.
   b. Wählen Sie ein oder mehrere Subnetze aus. Wir empfehlen, Subnetze in mindestens zwei Availability Zones auszuwählen.
   c. (Optional) Wählen Sie einen IP-Adresstyp (IPv4, IPv6 oder Dual-Stack).
7. (Service-managed) Unter Zugriffskontrolle:
   a. Wählen Sie die Sicherheitsgruppen aus, die Ihren verbundenen Ressourcen zugeordnet sind.
   b. (Optional) Geben Sie unter Erweiterte Konfiguration die Portbereiche ein, z. B. bis zu 11 durch Kommas getrennte TCP-Ports oder -Bereiche. `443, 8080-8090`
8. (Service-managed) Unter Details zum Serviceziel:
   a. Geben Sie im Feld Hostadresse den DNS-Hostnamen oder die IP-Adresse Ihres Zieldienstes ein.
   b. Wählen Sie für die DNS-Auflösung Öffentlich für eine öffentlich auflösbare Hostadresse oder In VPC (privates DNS) für eine Adresse, die nur innerhalb der VPC aufgelöst werden kann.
   c. (Optional) Geben Sie im Feld Öffentlicher Schlüssel des Zertifikats den PEM-encoded öffentlichen Schlüssel ein, wenn Ihr Ziel ein selbstsigniertes Zertifikat oder eine private Zertifizierungsstelle verwendet. Dadurch kann der AWS Security Agent der TLS-Verbindung vertrauen.
9. (Self-managed) Wählen Sie unter Bestehende Ressourcenkonfiguration für Ressourcenkonfiguration eine VPC-Lattice-Ressourcenkonfiguration aus der Liste aus, oder geben Sie eine Ressourcenkonfigurations-ID (beginnend mit`rcfg-`) oder deren ARN ein. Die Liste zeigt die Ressourcenkonfigurationen in der aktuellen Region.

10. Wählen Sie Create Connection (Verbindung erstellen) aus.

    Der Verbindungsstatus ändert sich in Create in progress. Dies kann bis zu 10 Minuten dauern. Wenn der Vorgang abgeschlossen ist, ändert sich der Status in Verfügbar.

Verwenden Sie eine private Verbindung mit einer Integration

Wenn Sie eine GitLab Self-Managed oder GitHub Enterprise Server-Integration registrieren, wählen Sie Über eine private Verbindung mit Endpunkt Connect und wählen Sie dann Ihre Verbindung aus der Liste Private Verbindungen aus. Der AWS Security Agent leitet den Datenverkehr über diese Verbindung an den Anbieter weiter. In der Liste werden nur Verbindungen mit dem Status Verfügbar angezeigt.

Sie können bei der Registrierung auch Private Verbindung erstellen auswählen. AWS Security Agent behält Ihren Registrierungsentwurf bei, während Sie die Verbindung herstellen, und kehrt dann zum Registrierungsablauf zurück.

Sicherheit

- Keine öffentliche Internetpräsenz — Der gesamte Datenverkehr verbleibt im AWS-Netzwerk.
- Customer-controlled Sicherheitsgruppen — Sie verwalten die Regeln für eingehenden und ausgehenden Datenverkehr.
- Service-linked Rolle mit den geringsten Rechten — AWS Security Agent verwendet eine serviceverknüpfte Rolle, die auf Ressourcen beschränkt ist, die mit markiert sind. `AWSSecurityAgentManaged`

> **Note**
>
> Wenn Ihre Organisation über Service Control Policies (SCPs) verfügt, die VPC Lattice API-Aktionen einschränken, wird das vom Service verwaltete Ressourcen-Gateway über eine serviceverknüpfte Rolle erstellt. Stellen Sie sicher, dass Ihre SCPs die erforderlichen Aktionen für die serviceverknüpfte Rolle zulassen.

Überprüfen Sie eine private Verbindung

Nachdem die private Verbindung den Status Verfügbar erreicht hat, überprüfen Sie die Konnektivität:

- Stellen Sie sicher, dass Ihr Zieldienst läuft und Verbindungen auf dem erwarteten Port akzeptiert
- Stellen Sie sicher, dass Sicherheitsgruppen, die an die ENIs angeschlossen sind, ausgehenden Datenverkehr auf dem Zielport zulassen
- Stellen Sie sicher, dass die Sicherheitsgruppe Ihres Dienstes eingehenden Datenverkehr von VPC-Lattice-Datenebenen-IPs innerhalb Ihres VPC-CIDR-Bereichs zulässt.
- Vergewissern Sie sich, dass Subnetz-Routing-Tabellen den Datenverkehr zwischen den ENI-Subnetzen und Ihrem Service zulassen

Löschen Sie eine private Verbindung

1. Navigieren Sie in der AWS Security Agent-Konsole zu Integrationen.
2. Wählen Sie die Registerkarte Private Verbindungen.
3. Wählen Sie die private Verbindung aus, die Sie löschen möchten.
4. Wählen Sie Löschen aus.

> **Important**
>
> Sie können keine private Verbindung löschen, die derzeit von einer Integration verwendet wird. Entfernen Sie zuerst die Integration und löschen Sie dann die private Verbindung.

Nächste Schritte

- the section called “Connect den AWS Security Agent mit GitLab Self-Managed”- Connect eine private GitLab Instanz
- the section called “AWS Security Agent mit GitHub Enterprise Server Connect”- Connect einen privaten GitHub Unternehmensserver

Agent mit privaten VPC-Ressourcen verbinden

Wenn die Anwendung, für die Sie einen Penetrationstest durchführen möchten, nicht im öffentlichen Internet verfügbar ist, müssen Sie dem AWS Security Agent eine VPC-Konfiguration zur Verfügung stellen. Der AWS Security Agent verwendet diese VPC-Konfiguration, einschließlich einer VPC, eines Subnetzes und von Sicherheitsgruppen, für den Zugriff auf die Anwendung.

> **Note**
>
> Beim Testen von Endpunkten in einer privaten VPC sind nur Endpoints zulässig, die IP-Adressen in bekannten privaten IP-Bereichen auflösen (weitere Informationen finden Sie unter VPC CIDR-Blöcke). Die folgenden IPv4- und IPv6-Bereiche sind zulässig:
>
> ```
> 10.0.0.0/8
> 172.16.0.0/12
> 192.168.0.0/16
> fd00::/8
> ```

> **Note**
>
> Wenn eine Verbindung zu einem Subnetz hergestellt wird, erstellt der AWS Security Agent ein ENI (Elastic Network Interface) in dem für den Penetrationstest konfigurierten Subnetz. Dieser ENI ist keine öffentliche IP-Adresse zugeordnet, was bedeutet, dass sie nicht mit VPC-Internet-Gateways in öffentlichen Subnetzen kommunizieren kann. Wenn Ihr Penetrationstest einen offenen Internetzugang erfordert, verwenden Sie stattdessen ein privates Subnetz mit einem zugehörigen VPC-NAT-Gateway

Sie gewähren dem AWS Security Agent über die AWS-Managementkonsole allgemeinen Zugriff auf eine VPC. In der Security Agent Web-App wählen Benutzer die spezifische Konfiguration für einen Penetrationstest aus.

So fügen Sie eine VPC im Agent Space hinzu

1. Navigieren Sie zur Agent Space-Übersichtsseite
2. Wählen Sie Aktionen und dann Penetrationstest-Konfiguration bearbeiten
3. Geben Sie unter der Überschrift VPC die VPC, Subnetze und Sicherheitsgruppen an

Sie können bis zu 5 VPCs hinzufügen.

Erforderliche Berechtigungen für die Agent Space-Dienstrolle

Um einen Penetrationstest mit einer VPC durchzuführen, muss Ihre Agent Space-Dienstrolle die folgenden Berechtigungen beinhalten:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CreateNetworkInterface",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs",
        "ec2:ModifyNetworkInterfaceAttribute",
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "ec2:CreateNetworkInterfacePermission",
      "Resource": "arn:aws:ec2:{{region}}:{{accountId}}:network-interface/*",
      "Condition": {
        "ArnEquals": {
          "ec2:Subnet": [
            "arn:aws:ec2:{{region}}:{{accountId}}:subnet/[[subnetIds]]"
          ]
        }
      }
    }
  ]
}
```

Um eine bestimmte VPC-Konfiguration für einen Penetrationstest in der Security Agent Web-App auszuwählen

1. Navigieren Sie zur Übersichtsseite der Penetrationstests
2. Wählen Sie den Penetrationstest aus, für den Sie die VPC-Konfiguration hinzufügen müssen, und wählen Sie dann Pentest-Details ändern aus.
3. Wählen Sie Weiter, um zum Bereich VPC-Ressourcen zu gelangen
4. Wählen Sie die VPC -, Subnetz - und Sicherheitsgruppen aus

5. Wählen Sie Weiter, um zum letzten Abschnitt zu gelangen und den Penetrationstest zu speichern

> **Note**
>
> Cross-account Penetrationstests werden derzeit für VPC-Ressourcen (Subnetze und Sicherheitsgruppen) unterstützt, die mit AWS Resource Access Manager gemeinsam genutzt werden. Secrets Manager Manager-Geheimnisse und Lambda-Funktionen, die für Authentifizierungsdaten verwendet werden, müssen in demselben AWS-Konto wie Ihr AWS Security Agent-Setup konfiguriert sein.

Durchführen eines Penetrationstests für VPC-Ressourcen in einem anderen AWS-Konto

Mit AWS Resource Access Manager können Sie Penetrationstests für VPC-Ressourcen durchführen, die mit Ihrem Konto geteilt werden. Beide Konten müssen Teil derselben AWS-Organisation sein.

1. (Optional) Aktivieren Sie die automatische gemeinsame Nutzung von Ressourcen für Ihre AWS-Organisation

```
aws ram enable-sharing-with-aws-organization
```

2. Verwenden Sie die Anmeldeinformationen des AWS-Kontos, dem die VPC-Ressourcen gehören, und geben Sie Subnetz- und Sicherheitsgruppenressourcen für das Konto des Penetrationstest-Besitzers frei

```
aws ram create-resource-share \
    --name SharePentestResources \
    --resource-arns <subnet ARN> <security group ARN> \
    --principals <penetration test owner account ID>
```

3. Navigieren Sie zur Agent Space-Übersichtsseite
4. Wählen Sie Penetrationstest und suchen Sie nach dem Namen der Servicerolle
5. Stellen Sie sicher, dass die IAM-Rolle Zugriff auf die gemeinsam genutzten VPC-Ressourcen gewährt

6. Wählen Sie Aktionen und dann Penetrationstest-Konfiguration bearbeiten
7. Geben Sie unter der Überschrift VPC die gemeinsam genutzte VPC, die Subnetze und die Sicherheitsgruppen an und speichern Sie die aktualisierte Konfiguration.
8. Navigieren Sie in der AWS Security Agent-Web-App zur Übersichtsseite für Penetrationstests.
9. Wählen Sie den Penetrationstest aus, für den Sie die VPC-Konfiguration hinzufügen müssen, und wählen Sie dann Pentest-Details ändern aus.
10. Aktualisieren Sie den Penetrationstest, um die gemeinsam genutzten VPC-Ressourcen zu verwenden

Funktionen konfigurieren

Aktivieren und konfigurieren Sie Penetrationstests, Codeüberprüfung und Bedrohungsmodellierung für Ihre Agent Spaces, einschließlich Problembehebung, S3-Quellen und Zieldomänen.

Penetrationstest aktivieren

Konfigurieren Sie AWS Security Agent für die Ausführung autonomer Penetrationstests für Ihre Anwendungen. Dieses Setup ermöglicht es dem AWS Security Agent, auf Ihre AWS-Ressourcen zuzugreifen, den Domainbesitz zu überprüfen und umfassende Sicherheitstests durchzuführen, um ausnutzbare Sicherheitslücken in Ihren Webanwendungen und APIs zu identifizieren.

Durch die Aktivierung von Penetrationstests kann der AWS Security Agent Ihre Anwendungssicherheit kontinuierlich überprüfen, ohne dass es zu Verzögerungen bei manuellen Tests kommt. Durch die Konfiguration der erforderlichen AWS-Integrationen stellen Sie sicher, dass der AWS Security Agent sowohl öffentliche als auch private Anwendungen testen, Ergebnisse protokollieren und auf Anmeldeinformationen für authentifizierte Tests zugreifen kann. CloudWatch

In diesem Verfahren konfigurieren Sie Zieldomänen, richten optional VPC, CloudWatch Protokollierung, Speicherung von Anmeldeinformationen und Lambda-Funktionen für Tests ein, konfigurieren S3-Integrationen für die Bereitstellung von zusätzlichem Kontext und richten den Servicezugriff über IAM-Rollen ein.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- AWS-Konto mit Administratorberechtigungen zum Erstellen von IAM-Rollen
- Möglichkeit zur Überprüfung des Domainbesitzes (Änderung von DNS- oder HTTP-Einträgen)

- Zu testende Zieldomänen oder -anwendungen
- (Optional) VPC-Konfigurationsdetails beim Testen privater Anwendungen
- (Optional) S3-Bucket, wenn zusätzliche Artefakte für den AWS Security Agent bereitgestellt werden

Schritt 1: Domain konfigurieren

Geben Sie im ersten Schritt des Assistenten die Zieldomänen an, die Sie testen möchten, und geben Sie an, wie der AWS Security Agent die Inhaberschaft überprüfen soll.

1. Geben Sie im Abschnitt Zieldomänen Ihre Domain in das Feld Domain ein.
2. Wählen Sie eine Bestätigungsmethode aus:
   - DNS_TXT — Beweisen Sie den Domainbesitz, indem Sie der DNS-Konfiguration Ihrer Domain einen TXT-Eintrag hinzufügen.
   - HTTP_ROUTE — Beweisen Sie den Domainbesitz, indem Sie eine Bestätigungsdatei unter einer bestimmten URL auf Ihrer Domain hosten.
   - PRIVATE_VPC — Nur für private VPC-Penetrationstests nutzbar. Überprüft, ob die Domain zu einer IP in einem privaten CIDR-Bereich aufgelöst wird
   - Weitere Informationen finden Sie unter the section called “Aktivieren Sie eine Anwendungsdomäne für Penetrationstests”.
3. Wählen Sie Weitere Domäne hinzufügen, um weitere Domänen hinzuzufügen (bis zu 5 insgesamt).
4. Wählen Sie Weiter, um mit der Domainverifizierung fortzufahren.

Schritt 2: Domains verifizieren

Überprüfen Sie im zweiten Schritt des Assistenten die Inhaberschaft aller von Ihnen konfigurierten Domänen. Der AWS Security Agent benötigt eine nachgewiesene Inhaberschaft, bevor er Penetrationstests durchführen kann.

1. Überprüfen Sie die in der Tabelle Zieldomänen aufgelisteten Domänen.
2. Wählen Sie jede Domain aus und lösen Sie die Überprüfung anhand der von Ihnen ausgewählten Methode aus:
   - Route-53-Domains (dasselbe AWS-Konto): Wählen Sie One-click Verifizierung. Der AWS Security Agent erstellt automatisch den DNS-Eintrag und schließt die Überprüfung ab.

- DNS-TXT (andere DNS-Anbieter): Kopieren Sie das Verifizierungstoken, fügen Sie den TXT-Eintrag bei Ihrem DNS-Registrar hinzu, wählen Sie dann die Domain aus und wählen Sie Verify.
- HTTP-Route: Platzieren Sie das Bestätigungstoken am erforderlichen Routenpfad auf Ihrem Webserver, wählen Sie dann die Domain aus und wählen Sie Verifizieren. Details hierzu finden Sie unter the section called “Aktivieren Sie eine Anwendungsdomäne für Penetrationstests”.

3. Wählen Sie Weiter, um mit der optionalen Konfiguration fortzufahren.

> **Note**
>
> Sub-domains Wenn Sie die DNS-TXT- oder HTTP-Routenüberprüfung verwenden, ist für eine verifizierte Domain keine individuelle Überprüfung erforderlich. Für die private VPC-Überprüfung muss der Domainname des Zielendpunkts mit dem vollständigen Domainnamen übereinstimmen, der für die Überprüfung konfiguriert wurde.

> **Note**
>
> Bei privaten Domains innerhalb einer VPC können Sie fortfahren, auch wenn der Status der Domainverifizierung NICHT ERREICHBAR ist. Der AWS Security Agent versucht zu Beginn jedes Pentest-Laufs, die Domain für private Endpunkte zu verifizieren.

Schritt 3: (Optional) Zusätzliche Funktionen konfigurieren

Im dritten Schritt des Assistenten können Sie optionale AWS-Ressourcen konfigurieren, um den Umfang Ihrer Penetrationstests zu erweitern. Alle Abschnitte in diesem Schritt sind optional und standardmäßig ausgeblendet.

(Optional) VPC-Einstellungen konfigurieren

Wenn Sie private Zieldomänen testen möchten, die in einer VPC gehostet werden, konfigurieren Sie die VPC-Einstellungen für AWS Security Agent. Dieser Abschnitt ist optional und standardmäßig ausgeblendet.

1. Erweitern Sie den Abschnitt VPCs.
2. Wählen Sie in der VPC-Dropdownliste die VPC aus, die Ihre privaten Zieldomänen hostet.

3. Wählen Sie in der Dropdownliste Subnetz die VPC-Subnetze aus, die der AWS Security Agent verwenden soll:

> **Tip**
>
> Wählen Sie für eine hohe Verfügbarkeit mehrere Subnetze aus mehreren Availability Zones aus. Stellen Sie sicher, dass Ihre Subnetze ein NAT-Gateway für ausgehende Konnektivität enthalten.

4. Wählen Sie in der Dropdownliste Sicherheitsgruppe die VPC-Sicherheitsgruppen aus, die der AWS Security Agent verwenden soll:

> **Important**
>
> Stellen Sie sicher, dass Ihre Sicherheitsgruppen ausgehende Verbindungen zulassen, damit der AWS Security Agent Penetrationstests durchführen kann.

(Optional) Konfigurieren Sie CloudWatch die Protokollierung

Konfigurieren Sie CloudWatch die Konfiguration so, dass Protokolle Ihrer Penetrationstestläufe gespeichert werden. Dieser Abschnitt ist optional und standardmäßig ausgeblendet.

1. Erweitern Sie den Abschnitt mit den CloudWatch Protokollen.
2. Wählen Sie in der Dropdownliste Protokollgruppen die vorhandenen CloudWatch Protokollgruppen aus
3. Wenn Sie keine Protokollgruppe auswählen, erstellt der AWS Security Agent eine Protokollgruppe `/aws/securityagent/<agent name>/<pentest id>` mit dem Namen der entsprechenden Berechtigungen.

> **Note**
>
> Stellen Sie sicher, dass Ihre IAM-Rolle über Schreibberechtigungen für die ausgewählte CloudWatch Protokollgruppe verfügt.

(Optional) Konfigurieren Sie Secrets für Testanmeldedaten

Wenn für Ihre Anwendung Authentifizierungsdaten zum Testen erforderlich sind, kann der AWS Security Agent diese sicher von AWS Secrets Manager abrufen. Dieser Abschnitt ist optional und standardmäßig ausgeblendet.

1. Erweitern Sie den Abschnitt Secrets.
2. Wählen Sie die AWS Secrets Manager Manager-Geheimnisse aus, die Anmeldeinformationen für Ihre Anwendung enthalten.
3. Bei der Konfiguration eines Penetrationstests in der Webanwendung können Sie für authentifizierte Tests auf diese Geheimnisse zurückgreifen.

> **Important**
>
> Anmeldeinformationen werden verschlüsselt und in AWS Secrets Manager gespeichert. Stellen Sie sicher, dass Ihre IAM-Rolle über Zugriffsberechtigungen für Secrets Manager for AWS Security Agent verfügt, um diese Anmeldeinformationen beim Testen zu verwenden.

(Optional) Lambda-Funktionen für Testanmeldedaten konfigurieren

Konfigurieren Sie Lambda-Funktionen, die beim Testen Anmeldeinformationen für Ihre Anwendung bereitstellen können. Dieser Abschnitt ist optional und standardmäßig ausgeblendet.

1. Erweitern Sie den Abschnitt Lambda-Funktionen.
2. Wählen Sie die Lambda-Funktionen aus, die Authentifizierungsdaten für Ihre Anwendung bereitstellen können.
3. AWS Security Agent ruft diese Funktionen bei Penetrationstests auf, um Anmeldeinformationen dynamisch abzurufen.

> **Note**
>
> Stellen Sie sicher, dass Ihre IAM-Rolle über Berechtigungen zum Aufrufen der angegebenen Lambda-Funktionen verfügt. Lambda-Funktionen sollten Anmeldeinformationen im erwarteten Format zurückgeben, damit der AWS Security Agent sie beim Testen verwenden kann.

(Optional) Konfigurieren Sie den S3-Bucket

Geben Sie S3-Bucket-Details an, wenn Sie Dokumente oder Artefakte hochladen möchten, um sie als Eingabe für den AWS Security Agent bereitzustellen. Dieser Abschnitt ist optional und standardmäßig ausgeblendet.

1. Erweitern Sie den Abschnitt S3-Buckets.
2. Geben Sie im Feld Bucket Ihren S3-Bucket-Namen ein oder suchen Sie danach.

> **Note**
>
> Sie können GitHub Repositorys auch später verbinden oder Dateien direkt in die Webanwendung hochladen. Durch die bereitgestellten Informationen kann eine umfassende Berichterstattung gewährleistet, Fehlalarme reduziert und umsetzbare Ergebnisse erzielt werden.

(Optional) Konfigurieren Sie den Servicezugriff

AWS Security Agent benötigt eine IAM-Rolle für den Zugriff auf Ihre AWS-Ressourcen (VPC, CloudWatch Protokollgruppen, Secrets Manager, Lambda-Funktionen usw.) für Penetrationstests. Dieser Abschnitt ist optional und standardmäßig ausgeblendet.

1. Erweitern Sie den Abschnitt Servicezugriff.
2. Standardmäßig verwendet der AWS Security Agent eine Standard-IAM-Rolle mit den erforderlichen Berechtigungen für Penetrationstests.
3. Um die IAM-Rolle anzupassen, wählen Sie eine der folgenden Optionen aus:
   a. Standardrolle erstellen — AWS Security Agent erstellt automatisch eine neue IAM-Rolle mit den erforderlichen Berechtigungen
   b. Eine bestehende Servicerolle verwenden — Wählen Sie eine bestehende IAM-Rolle aus dem Drop-down-Menü aus
4. Wenn Sie eine bestehende Rolle verwenden:
   a. Wählen Sie das Drop-down-Menü unter Bestehende Rolle auswählen
   b. Wählen Sie Ihre IAM-Rolle aus der Liste aus
   c. Wählen Sie das Aktualisierungssymbol, um die Liste bei Bedarf zu aktualisieren

> **Note**
>
> Die Standard-IAM-Rolle umfasst Berechtigungen für den Zugriff auf VPC-Ressourcen, CloudWatch Protokollgruppen, Secrets Manager, Lambda-Funktionen und andere Dienste, die für Penetrationstests erforderlich sind. Es wird empfohlen, die Standard-IAM-Rolle zu verwenden, sofern Sie keine speziellen Sicherheitsanforderungen haben.

Schritt 4: Speichern und aktivieren Sie Penetrationstests

Nachdem Sie alle erforderlichen Einstellungen konfiguriert haben, aktivieren Sie Penetrationstests für Ihren AWS Security Agent.

1. Überprüfen Sie alle Konfigurationsabschnitte, um die Richtigkeit sicherzustellen.
2. Wählen Sie Speichern.
3. AWS Security Agent validiert Ihre Konfiguration und erstellt die erforderlichen AWS-Ressourcen.

Nächste Schritte

Nach der Aktivierung der Penetrationstests:

- Konfigurieren Sie die Bereiche für Penetrationstests in der AWS Security Agent-Webanwendung
- Richten Sie Benachrichtigungseinstellungen für Testergebnisse ein
- Überprüfen Sie die Ergebnisse des Penetrationstests und reagieren Sie darauf, sobald sie entdeckt werden
- (Optional) Konfigurieren Sie zusätzliche Repositorys für die Behebung von Ergebnissen

Weitere Informationen zum Ausführen und Verwalten von Penetrationstests finden Sie in der Dokumentation zur AWS Security Agent-Webanwendung.

Pull-Request-Code-Review für GitHub Repositorys aktivieren

Die Überprüfung des Pull-Request-Codes ist jetzt als Teil des Code-Review-Einrichtungsassistenten konfiguriert. Wenn Sie GitHub Repositorys mit Ihrem Agent Space verbinden, können Sie Pull-Request-Kommentare für einzelne Repositorys aktivieren, damit der AWS Security Agent automatisch Pull-Anfragen analysiert und Sicherheitsfeststellungen veröffentlicht.

Vollständige Anweisungen zur Einrichtung finden Sie unter. the section called “Codeüberprüfung aktivieren”

Informationen darüber, wie Ergebnisse von Pull-Requests angezeigt werden GitHub und wie Sie darauf reagieren können, finden Sie unterthe section called “Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests”.

Codeüberprüfung aktivieren

Konfigurieren Sie Ihren Agent Space so, dass er die Codeüberprüfung ermöglicht, indem Sie Quellcode aus GitHub Repositorys oder S3-Buckets verbinden. Code Review analysiert Ihren Quellcode auf Sicherheitslücken und die Einhaltung der individuellen Sicherheitsanforderungen Ihres Unternehmens.

Das Einrichten von Code-Review-Konfigurationen ist ein Space-wide Agentenvorgang. Die Integrationen und S3-Buckets, die Sie verbinden, werden von allen Funktionen gemeinsam genutzt, einschließlich Codeüberprüfung und Penetrationstests.

Nach Abschluss der Installation können Benutzer in der AWS Security Agent-Webanwendung Codeprüfungen erstellen und ausführen, um Repositorys und S3-Quellen auf Sicherheitsprobleme zu durchsuchen.

> **Note**
>
> Wenn Sie bereits GitHub Repositorys mit Ihrem Agent Space verbunden haben (z. B. durch die Einrichtung von Penetrationstests), ist die Codeüberprüfung bereits aktiviert. Sie können diese Einrichtung überspringen und direkt zur Webanwendung wechseln, um Code-Reviews zu erstellen. Siehe the section called “Einen Code-Review erstellen”.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein in der AWS-Managementkonsole erstellter Agent Space (siehethe section called “Erstellen Sie einen Agentenbereich”)
- Mindestens eine der folgenden Quellcode-Eingaben:
  - Ein GitHub Organisations- oder Benutzerkonto, auf dem die AWS Security Agent GitHub App installiert ist (siehethe section called “AWS Security Agent mit GitHub Repositorys Connect”)

  - Ein S3-Bucket mit Quellcode, den Sie überprüfen möchten
- Berechtigungen zur Konfiguration von Integrationen für Ihren Agent Space
- (Optional) In einem Sicherheitsanforderungspaket ist mindestens eine Sicherheitsanforderung aktiviert, wenn Sie die Validierung von Sicherheitsanforderungen verwenden möchten (siehethe section called “Sicherheitsanforderungen verwalten”)

Rufen Sie den Einrichtungsassistenten für die Codeüberprüfung auf

Navigieren Sie zur Einrichtung des Code-Reviews für Ihren Agent Space.

1. Wählen Sie in der AWS Security Agent-Konsole Ihren Agent Space aus.
2. Wählen Sie an einem der folgenden Orte die Option Codeüberprüfung aktivieren aus:
   - Die Code-Review-Karte
   - Wählen Sie auf der Registerkarte Codeüberprüfung die Option Codeüberprüfung aktivieren

> **Tip**
>
> Wenn Sie möchten, dass der AWS Security Agent Feedback zur Codeüberprüfung automatisch bearbeitet, aktivieren Sie auch die Codekorrektur. Details dazu finden Sie unter the section called “Ermöglichen Sie es Benutzern, mit der Behebung der Ergebnisse von Penetrationstests und Code-Reviews zu beginnen”.

Sie werden zum Einrichtungsassistenten für die Codeüberprüfung von Konfigurationen weitergeleitet.

Schritt 1: Integrationen, Repos und Buckets Connect

Im ersten Schritt des Assistenten verbinden Sie Ihre Quellcode-Eingaben und konfigurieren Sie die Einstellungen für die Codeüberprüfung. Sie müssen mindestens ein GitHub Repository oder einen S3-Bucket hinzufügen, um fortzufahren.

> **Important**
>
> Die hier konfigurierten Integrationen und S3-Buckets werden in Ihrem Agent Space gemeinsam genutzt. Die Änderungen betreffen sowohl die Funktionen zur Codeüberprüfung als auch für Penetrationstests.

GitHub Repositorys Connect

Fügen Sie GitHub Repositorys von Ihren autorisierten GitHub Organisationen oder Benutzerkonten hinzu. Wenn Sie Hinzufügen wählen, wird der zweistufige GitHubConnect-Assistent geöffnet, in dem Sie zuerst Repositorys auswählen und dann konfigurieren, welche Aktionen der AWS Security Agent für jedes Repositorium ausführen kann.

1. Wählen Sie im Abschnitt Verbundene Integrationen die Option Hinzufügen aus.
2. Wählen Sie die GitHub Registrierung aus, die die Repositorys enthält, die Sie überprüfen möchten.
3. Aktivieren Sie im Schritt GitHub Repositories verbinden das Kontrollkästchen für jedes Repository, das Sie verbinden möchten.
4. Wählen Sie Weiter, um zu Funktionen verwalten zu wechseln.

> **Note**
>
> Auf verbundene Repositorys wird zur Codeanalyse bei der Codeüberprüfung und bei Penetrationstests schreibgeschützt zugegriffen. Im nächsten Schritt konfigurieren Sie Schreibaktionen wie das Veröffentlichen von Bewertungskommentaren und das Öffnen von Korrektur-Pull-Requests.

> **Note**
>
> Wenn Sie noch keine GitHub Integration registriert haben, wählen Sie Einstellungen, um zur Seite Integrationen zu gelangen, auf der Sie die AWS Security Agent GitHub App autorisieren können. Weitere Informationen finden Sie unter the section called “AWS Security Agent mit GitHub Repositorys Connect”.

Konfigurieren Sie Repository-Funktionen GitHub

Wählen Sie im Schritt Funktionen verwalten des GitHubConnect-Assistenten aus, was der AWS Security Agent in den einzelnen Repositorys tun kann. Kommentare zur Codeüberprüfung und zur Codekorrektur werden für jedes Repository unabhängig voneinander festgelegt.

1. Aktivieren Sie für jedes Repository die Option Kommentare zur Codeüberprüfung, damit der AWS Security Agent Sicherheitsergebnisse als Kommentare zu Pull-Requests veröffentlicht.

2. Schalten Sie für jedes Repository die Codekorrektur ein, damit der AWS Security Agent die Ergebnisse korrigieren kann. Wenn diese Option aktiviert ist, können Web-App-Benutzer Pull-Requests anfordern, die Ergebnisse korrigieren, und die Autoren von Pull-Requests können Kommentare abgeben, `@AWS-Security-Agent fix all findings.` damit der Agent auf seine Bewertungskommentare eingeht.
3. Wählen Sie Speichern, um Ihre Auswahl zu übernehmen und zum Einrichtungsassistenten für die Codeüberprüfung zurückzukehren.

Wenn Code-Review-Kommentare für ein Repository aktiviert sind:

- AWS Security Agent analysiert automatisch Pull-Anfragen, wenn sie als „Bereit zur Überprüfung“ gekennzeichnet sind. Pull-Request-Entwürfe werden nicht analysiert.
- Sicherheitsergebnisse werden als Bewertungskommentare direkt auf der Pull-Anfrage veröffentlicht und enthalten spezifische Hinweise zur Problembehebung.
- Die Analyse verwendet Ihre konfigurierten Einstellungen für die Codeüberprüfung (Sicherheitslücken, benutzerdefinierte Anforderungen oder beides).

> **Note**
>
> Kommentare zu Pull-Requests sind nur für private GitHub Repositorys verfügbar.

Wenn die Codekorrektur für ein Repository aktiviert ist, können Web-App-Benutzer mit der Behebung sowohl für die Ergebnisse der Codeüberprüfung als auch für die Ergebnisse des Penetrationstests in diesem Repository beginnen, und der AWS Security Agent stellt jeden Fix als Pull-Anfrage bereit. Weitere Informationen finden Sie unter the section called “Ermöglichen Sie es Benutzern, mit der Behebung der Ergebnisse von Penetrationstests und Code-Reviews zu beginnen”.

Weitere Informationen darüber, wie die Ergebnisse von Pull-Anfragen angezeigt werden GitHub und wie darauf reagiert werden kann, finden Sie unter. the section called “Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests”

Fügen Sie S3-Buckets hinzu

Fügen Sie S3-Buckets hinzu, die Quellcode oder kontextuelle Ressourcen für die Codeüberprüfung enthalten.

1. Wählen Sie im Abschnitt S3-Buckets die Option S3-Ressource hinzufügen aus.
2. Geben Sie den S3-URI für den Bucket oder das Präfix ein, das Ihren Quellcode enthält.
3. Wählen Sie Hinzufügen aus.

> **Note**
>
> Sie können bis zu 10 S3-Ressourcen hinzufügen. S3-Buckets werden für verschiedene Funktionen wie Codeüberprüfung und Penetrationstests gemeinsam genutzt.

> **Tip**
>
> Sie können S3-Buckets hinzufügen, die Quellcode, Konfigurationsdateien, Infrastructure-as-Code-Vorlagen oder andere Artefakte enthalten, die der AWS Security Agent auf Sicherheitsprobleme analysieren soll.

Konfigurieren Sie die Einstellungen für die Codeüberprüfung

Konfigurieren Sie die Arten von Sicherheitsproblemen, die der AWS Security Agent bei Codeprüfungen analysiert. Diese Einstellung gilt für alle Repositorys und Quellen, für die die Codeüberprüfung in diesem Agent Space aktiviert ist.

1. Wählen Sie im Abschnitt Einstellungen für die Codeüberprüfung eine der folgenden Optionen aus:
   - Überprüfung der Sicherheitsanforderungen — Überprüfen Sie, ob der Code den Sicherheitsanforderungen entspricht, die Sie in Ihren Sicherheitsanforderungspaketen aktiviert haben.
   - Ergebnisse von Sicherheitslücken — Identifizieren Sie häufig auftretende Sicherheitslücken im Code.
   - Sicherheitsanforderungen und Erkenntnisse zu Sicherheitslücken — Analysieren Sie den Code sowohl auf die Einhaltung der von Ihnen aktivierten Sicherheitsanforderungen als auch auf allgemeine Sicherheitslücken. Dies ist die Standardeinstellung.

> **Note**
>
> Wenn die Validierung von Sicherheitsanforderungen aktiviert ist, überprüft der AWS Security Agent den Code anhand der Sicherheitsanforderungen, die Sie in Ihren Sicherheitsanforderungspaketen aktiviert haben. Wenn Sie die Validierung von Sicherheitsanforderungen auswählen, aber nicht mindestens eine Sicherheitsanforderung aktiviert haben, identifiziert der AWS Security Agent keine anforderungsbasierten Ergebnisse. Weitere Informationen zu Sicherheitsanforderungen finden Sie unter the section called “Sicherheitsanforderungen verwalten”.

1. Wählen Sie Weiter, um mit den optionalen Konfigurationen fortzufahren.

Schritt 2: Optionale Konfigurationen

Im zweiten Schritt des Assistenten konfigurieren Sie optionale Einstellungen für CloudWatch Protokollierung und Servicezugriff für Ihre Codeüberprüfungsumgebung.

(Optional) Konfigurieren Sie CloudWatch Protokolle

Konfigurieren Sie CloudWatch Protokollgruppen, um das Anwendungsverhalten während der Codeüberprüfung zu erfassen und zu analysieren.

1. Erweitern Sie den Abschnitt CloudWatch Protokolle.
2. Wählen Sie in der Dropdownliste Protokollgruppen eine oder mehrere bestehende CloudWatch Protokollgruppen aus Ihrem AWS-Konto aus.

> **Note**
>
> Wenn Sie keine Protokollgruppe auswählen, erstellt der AWS Security Agent automatisch eine Standard-Protokollgruppe zum Speichern von Codeüberprüfungs-Ausführungsprotokollen.

(Optional) Konfigurieren Sie den Servicezugriff

Konfigurieren Sie die IAM-Servicerolle, die der AWS Security Agent für den Zugriff auf Ihre AWS-Ressourcen wie S3-Buckets und CloudWatch Logs für die Codeüberprüfung verwendet.

1. Erweitern Sie den Abschnitt Servicezugriff.
2. Wählen Sie eine der folgenden Optionen aus:
   - Standardrolle erstellen — AWS Security Agent erstellt automatisch eine neue IAM-Rolle mit den erforderlichen Berechtigungen für die Codeüberprüfung.
   - Eine bestehende Servicerolle verwenden — Wählen Sie eine bestehende IAM-Rolle aus dem Drop-down-Menü aus.
3. Wenn Sie die Standardrolle verwenden, geben Sie einen Namen für die Servicerolle ein. Der Name muss für alle Rollen im Konto eindeutig sein, alphanumerische Zeichen und `+=,.@-_` Zeichen enthalten und darf keine Leerzeichen enthalten.

> **Note**
>
> Der Name der Servicerolle hat eine maximale Länge von 64 Zeichen. Eine Servicerolle wird automatisch erstellt, wenn Sie keine vorhandene Rolle auswählen.

1. Wählen Sie Speichern, um die Einrichtung abzuschließen.

Nach der Einrichtung

Nach Abschluss des Einrichtungsassistenten für die Codeüberprüfung:

- Auf der Karte zur Codeüberprüfung auf Ihrer Agent Space-Seite wird der Status Bereit angezeigt.
- Benutzer können die Webanwendung starten und Code-Reviews erstellen, um verbundene Repositorys und S3-Quellen zu scannen.
- Sie können Ihre Konfiguration jederzeit ändern, indem Sie auf der Registerkarte Codeüberprüfung die Option Konfiguration bearbeiten auswählen.

Bearbeiten Sie die Code-Review-Konfiguration

So ändern Sie Ihre Code-Review-Konfiguration nach der Ersteinrichtung:

1. Navigieren Sie in der AWS Security Agent-Konsole zu Ihrem Agent Space.
2. Wählen Sie die Registerkarte Codeüberprüfung aus.
3. Wählen Sie Edit configuration (Konfiguration bearbeiten) aus.

4. Aktualisieren Sie Ihre Integrationen, S3-Buckets, Code-Review-Einstellungen, CloudWatch Protokolle oder den Servicezugriff nach Bedarf.
5. Wählen Sie Speichern.

Nächste Schritte

Nach der Einrichtung von Code-Review-Konfigurationen:

- Starten Sie die Webanwendung, um Code-Reviews zu erstellen und auszuführen (siehethe section called “Einen Code-Review erstellen”)
- Connect zusätzliche GitHub Repositorys oder S3-Buckets, wenn Ihre Codebasis wächst
- Konfigurieren Sie Sicherheitsanforderungspakete für die organisationsspezifische Validierung (siehe) the section called “Sicherheitsanforderungen verwalten”
- Sehen Sie sich an, wie die Ergebnisse von Pull-Requests angezeigt werden in GitHub (siehe) the section called “Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests”

Bedrohungsmodellierung aktivieren

Konfigurieren Sie Ihren Agent Space so, dass er die Bedrohungsmodellierung ermöglicht, indem Sie Quellcode-Repositorys verbinden und AWS-Ressourcen konfigurieren. Die Bedrohungsmodellierung analysiert die Architektur Ihrer Anwendung und identifiziert Sicherheitsbedrohungen anhand von Quellcode, Entwurfsdokumenten oder beidem.

Das Einrichten von Konfigurationen zur Bedrohungsmodellierung ist ein Space-wide Agentenvorgang. Die Integrationen und S3-Buckets, die Sie verbinden, werden von allen Funktionen gemeinsam genutzt, einschließlich Bedrohungsmodellierung, Codeüberprüfung und Penetrationstests.

Nach Abschluss der Einrichtung können Benutzer Bedrohungsmodelle in der AWS Security Agent-Webanwendung erstellen und ausführen.

> **Note**
>
> Wenn Sie bereits Repositorys oder S3-Buckets mit Ihrem Agent Space verbunden haben (z. B. durch die Einrichtung von Codeüberprüfungen oder Penetrationstests), ist die Bedrohungsmodellierung möglicherweise bereits aktiviert. Sie können direkt zur

Webanwendung wechseln, um ein Bedrohungsmodell zu erstellen. Siehe the section called “Erstellen Sie ein Bedrohungsmodell”.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein in der AWS-Managementkonsole erstellter Agent Space (siehethe section called “Erstellen Sie einen Agentenbereich”)
- Berechtigungen zur Konfiguration von Integrationen für Ihren Agent Space
- (Optional) Eine GitHub, GitLab, oder Bitbucket-Organisation mit installierter AWS Security Agent-App (siehe AWS Security Agent mit Repositorys Connect, AWS Security Agent mit GitHub Repositorys Connect oder AWS Security Agent mit GitLab Bitbucket-Repositorys verbinden)

Rufen Sie den Einrichtungsassistenten zur Bedrohungsmodellierung auf

Navigieren Sie zur Konfiguration der Bedrohungsmodellierung für Ihren Agent Space.

1. Wählen Sie in der AWS Security Agent-Konsole Ihren Agent Space aus.
2. Wählen Sie auf der Karte mit dem Bedrohungsmodell oder auf der Registerkarte Bedrohungsmodell die Option Bedrohungsmodell konfigurieren aus.

Sie werden zum Assistenten zum Konfigurieren des Bedrohungsmodells weitergeleitet, der aus zwei optionalen Schritten besteht.

Schritt 1: Quellcode-Repositorys Connect (optional)

Connect die GitHub GitLab, oder Bitbucket-Repositorys, für die du die Bedrohungsmodellierung aktivieren möchtest. Die Bedrohungsmodelle selbst werden in der Webanwendung erstellt und angezeigt.

Important

Die hier konfigurierten Integrationen werden in Ihrem gesamten Agent Space gemeinsam genutzt. Die Änderungen betreffen die Funktionen zur Bedrohungsmodellierung, Codeüberprüfung und Penetrationstests.

Repositorys Connect

1. Wählen Sie im Abschnitt Verbundene Integrationen die Option Hinzufügen aus.
2. Wählen Sie die Registrierung aus, die die Repositorys enthält, die Sie verwenden möchten.
3. Aktivieren Sie das Kontrollkästchen für jedes Repository, das Sie verbinden möchten.
4. Wählen Sie Speichern, um Ihre Auswahl zu übernehmen.

> **Note**
>
> Wenn Sie noch keine Integration registriert haben, wählen Sie Einstellungen, um zur Seite Integrationen zu gelangen, auf der Sie die AWS Security Agent-App autorisieren können. Weitere Informationen finden Sie unter AWS Security Agent Connect GitHub Repositorys Connect, AWS Security Agent mit GitLab Repositorys Connect oder AWS Security Agent mit Bitbucket-Repositorys verbinden.

1. Wählen Sie Weiter, um mit den optionalen Konfigurationen fortzufahren.

Schritt 2: Optionale Konfigurationen

Konfigurieren Sie S3-Buckets, CloudWatch Protokollierung und Einstellungen für den Dienstzugriff für Ihre Umgebung zur Bedrohungsmodellierung. Diese Einstellungen werden mit anderen Funktionen in Ihrem Agent Space gemeinsam genutzt.

S3-Buckets (optional)

Fügen Sie S3-Buckets hinzu, die den Quellcode enthalten, den der Agent bei der Bedrohungsmodellierung als Kontext verwenden soll.

1. Wählen Sie im Abschnitt S3-Buckets die Option S3-Ressource hinzufügen aus.
2. Geben Sie den S3-URI für den Bucket oder das Präfix ein.
3. Wählen Sie Hinzufügen aus.

> **Note**
>
> Sie können bis zu 10 S3-Ressourcen hinzufügen.

CloudWatch Protokolle (optional)

Konfigurieren Sie CloudWatch Protokollgruppen, um das Anwendungsverhalten bei der Ausführung von Bedrohungsmodellen zu erfassen und zu analysieren.

1. Wählen Sie im Abschnitt CloudWatch Protokolle eine oder mehrere bestehende CloudWatch Protokollgruppen aus Ihrem AWS-Konto aus.

Zugriff auf Services

Konfigurieren Sie die IAM-Servicerolle, die der AWS Security Agent für den Zugriff auf Ihre AWS-Ressourcen wie S3-Buckets und CloudWatch Logs für die Bedrohungsmodellierung verwendet. Eine Servicerolle ist erforderlich, um die Bedrohungsmodellierung zu aktivieren.

1. Wählen Sie im Abschnitt Dienstzugriff eine bestehende IAM-Dienstrolle aus der Dropdownliste aus, oder lassen Sie das Feld leer, damit automatisch eine Servicerolle erstellt wird.

> **Note**
>
> Eine Servicerolle wird automatisch erstellt, wenn Sie keine vorhandene Rolle auswählen.

1. Wählen Sie Speichern, um die Konfiguration abzuschließen.

Nach der Einrichtung

Nach Abschluss der Konfiguration der Bedrohungsmodellierung:

- Auf der Karte mit dem Bedrohungsmodell auf Ihrer Agent Space-Seite wird der Status Bereit angezeigt.
- Benutzer können die Webanwendung starten und Bedrohungsmodelle aus dem verbundenen Quellcode, hochgeladenen Scope-Dokumenten, Confluence-Seiten oder einer Kombination dieser Eingaben erstellen.

Nächste Schritte

Nach der Aktivierung der Bedrohungsmodellierung:

- Starten Sie die Webanwendung, um Bedrohungsmodelle zu erstellen und auszuführen (siehe Bedrohungsmodell erstellen)
- Connect zusätzliche Repositorys oder S3-Buckets, wenn Ihre Codebasis wächst
- Connect Confluence, um Seiten als Bereichsdokumente auswählen zu können (siehe AWS Security Agent mit Confluence Connect)

Ermöglichen Sie es Benutzern, mit der Behebung der Ergebnisse von Penetrationstests und Code-Reviews zu beginnen

In der AWS-Managementkonsole können Sie die automatische Problembehebung aktivieren, sodass Benutzer der AWS Security Agent-Web-App Korrekturen für ein bestimmtes Problem anfordern können. Der AWS Security Agent stellt jeden Fix als GitHub Pull-Anfrage im betroffenen Repository bereit.

Sie aktivieren die Problembehebung pro Repository von einem Agent Space aus. Über die Registerkarten Penetrationstest und Codeüberprüfung wird derselbe Repository-Konfigurationsassistent geöffnet, sodass Sie beide Registerkarten verwenden können, um die Einstellung Automatische Problembehebung aktiviert zu verwalten. Die Korrektur bezieht sich auf Ergebnisse aus beiden Funktionen, sodass Sie sie nur einmal pro Repository aktivieren müssen.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

1. Penetrationstests (siehethe section called “Penetrationstest aktivieren”) oder Codeüberprüfung (siehethe section called “Codeüberprüfung aktivieren”) aktiviert
2. Die AWS Security Agent GitHub App für Ihr GitHub Unternehmen installiert und autorisiert (siehethe section called “AWS Security Agent mit GitHub Repositorys Connect”)

Öffnen Sie den Repository-Konfigurationsassistenten

Wählen Sie den Einstiegspunkt, der der Einrichtung Ihrer Repositorys entspricht.

Auf der Registerkarte Penetrationstest

Verwenden Sie diesen Pfad, um GitHub Repositorys für Penetrationstests hinzuzufügen und die Problembehebung in demselben Durchgang zu konfigurieren.

1. Navigieren Sie zur Agent-Space-Übersichtsseite.
2. Wählen Sie die Registerkarte Penetrationstest.
3. Wählen Sie eine GitHub Registrierung aus, der Ihre Repositorys gehören:
   a. Wenn Sie dem Agent Space keine GitHub Registrierung zugeordnet haben, wählen Sie im Informationsfeld Connect GitHub to AWS Security Agent die Option Hinzufügen aus, um eine Registrierung auszuwählen.
   b. Wenn bereits eine oder mehrere GitHub Registrierungen verknüpft sind, wählen Sie im Abschnitt Verbundene Integrationen die Option Hinzufügen aus, um eine weitere auszuwählen.
4. Wählen Sie Weiter, um GitHub Repositorys auszuwählen.
5. Wählen Sie Weiter, um die Repository-Funktionen zu konfigurieren.

Auf der Registerkarte „Codeüberprüfung“

Verwenden Sie diesen Pfad, wenn Ihre Repositorys bereits für die Codeüberprüfung verbunden sind.

1. Navigieren Sie zur Übersichtsseite von Agent Space.
2. Wählen Sie die Registerkarte Codeüberprüfung.
3. Wählen Sie in der GitHub Repository-Tabelle Bearbeiten aus, um den Repository-Konfigurationsassistenten zu öffnen.

Aktivieren Sie die automatische Problembehebung und speichern Sie

Nachdem Sie die Repository-Funktionen erreicht haben, gehen Sie durch einen der oben genannten Pfade:

1. Markieren Sie in der Spalte Automatische Wiederherstellung aktiviert jedes Repository, das Sie korrigieren möchten, als Aktiviert.
2. Wählen Sie Connect, um die Konfiguration zu speichern.

Benutzer der Web-App können jetzt mit der Behebung von Ergebnissen in diesen Repositorys beginnen, und der AWS Security Agent öffnet Pull-Requests mit den vorgeschlagenen Fixes.

Stellen Sie Agentenressourcen aus einem S3-Bucket bereit

Sie können dem AWS Security Agent Zugriff auf Ressourcen in einem S3-Bucket gewähren, z. B. API-Dokumente, Dokumente zum Bedrohungsmodell und Quellcode, die Penetrationstests unterstützen.

Sie gewähren dem AWS Security Agent Lesezugriff auf einen S3-Bucket in der AWS-Managementkonsole. In der Security Agent Web-App wählen Benutzer einzelne Ressourcen aus S3 nach Bedarf aus.

1. Navigieren Sie zur Übersichtsseite von Agent Space
2. Wählen Sie Aktionen und dann Penetrationstest-Konfiguration bearbeiten
3. Definieren Sie im Abschnitt S3-Buckets die S3-URI, die den S3-Bucket enthält. Sie können mehrere S3-Buckets hinzufügen.

Aktivieren Sie eine Anwendungsdomäne für Penetrationstests

Bevor Sie einen Penetrationstest für eine Anwendung ausführen können, müssen Sie eine Zieldomäne hinzufügen und die Inhaberschaft verifizieren. AWS Security Agent führt nur Penetrationstests für verifizierte Domains durch.

> **Note**
>
> Sie müssen keine zusätzlichen Domains validieren, die Ihre Anwendung möglicherweise verwendet. Sie müssen nur die Domain validieren, für die Sie aktiv Penetrationstests durchführen werden.

Schritt 1: Fügen Sie eine Zieldomain hinzu

Um eine Zieldomain hinzuzufügen, navigieren Sie auf der Agent Space-Übersichtsseite zur Registerkarte Penetrationstest. Je nachdem, ob Sie bereits Penetrationstests konfiguriert haben, verwenden Sie eine der folgenden Methoden:

- First-time Setup: Wählen Sie Penetrationstest einrichten, um den Penetrationstest-Assistenten zu öffnen. Geben Sie im ersten Schritt die Domain ein und wählen Sie eine Bestätigungsmethode aus.

- Eine Domain zu einer bestehenden Konfiguration hinzufügen: Wählen Sie in der Tabelle Zieldomänen die Option Domain hinzufügen aus. Geben Sie die Domain ein und wählen Sie im Modal eine Bestätigungsmethode aus.

Sie können eine Basisdomain oder eine Subdomain hinzufügen, z. B. `example.com` oder`billing.example.com`. AWS schlägt vor, eine Subdomain zu verwenden, für die Sie die Erlaubnis haben, `TXT` Datensätze zu erstellen.

> **Note**
>
> Wenn Sie eine Domain mithilfe der DNS-TXT- oder HTTP-Routenverifizierung verifizieren, werden die Unterdomänen dieser Domain automatisch abgedeckt. Mit der Verifizierung `example.com` können Sie beispielsweise testen `api.example.com` oder `billing.example.com` ohne zusätzliche Überprüfung. Für die private VPC-Überprüfung muss der Domainname des Zielendpunkts mit dem vollständigen Domainnamen übereinstimmen, der für die Überprüfung konfiguriert wurde.

Wählen Sie eine der folgenden Überprüfungsmethoden:

- DNS-TXT-Eintrag: Beweisen Sie den Domainbesitz, indem Sie bei Ihrem DNS-Anbieter einen DNS-TXT-Eintrag erstellen.
- HTTP-Route: Beweisen Sie den Domainbesitz, indem Sie auf Ihrem Webserver eine Route erstellen, die ein eindeutiges Token enthält, das vom AWS Security Agent bereitgestellt wird.
- Private VPC: Nur für private VPC-Penetrationstests nutzbar. Überprüft, ob die Domain zu einer IP in einem privaten CIDR-Bereich aufgelöst wird. Der konfigurierte Domänenname muss mit dem vollständigen Domänennamen aller zugehörigen Zielendpunkte übereinstimmen

Schritt 2: Überprüfen Sie die Inhaberschaft der Domain

Nachdem Sie die Domain hinzugefügt haben, überprüfen Sie die Inhaberschaft mit der von Ihnen ausgewählten Methode. Sie können die Überprüfung jederzeit in der Tabelle Zieldomänen auslösen, indem Sie die Domain auswählen und Verifizieren wählen.

Überprüfen Sie dies mithilfe eines DNS-TXT-Eintrags

AWS Security Agent generiert einen TXT-DNS-Eintragswert. Sie müssen diesen Eintrag bei Ihrem DNS-Anbieter hinzufügen, um die Inhaberschaft nachzuweisen.

Wenn die Domain in Route 53 registriert ist (dasselbe AWS-Konto):

Der AWS Security Agent kann den DNS-Eintrag automatisch erstellen. Wählen Sie die Domain aus der Tabelle Target Domains aus und wählen Sie One-click Verifizierung. Der AWS Security Agent erstellt den DNS-Validierungsdatensatz und schließt die Überprüfung automatisch ab.

Wenn die Domain bei einem anderen DNS-Anbieter registriert ist:

1. Kopieren Sie den vom AWS Security Agent bereitgestellten TXT-Datensatzwert.
2. Fügen Sie den TXT-Eintrag bei Ihrem DNS-Registrar hinzu.
3. Kehren Sie zur Tabelle mit den Zieldomänen zurück, wählen Sie die Domain aus und klicken Sie auf Überprüfen.

Überprüfen Sie mithilfe der HTTP-Routenvalidierung

Diese Methode beweist den Domainbesitz, indem ein eindeutiges Token auf Ihrem Webserver platziert wird. Nur Domaininhaber oder autorisierte Webadministratoren können Routen auf einem Webserver erstellen, wodurch der Besitz nachgewiesen wird.

1. Erstellen Sie eine Datei im folgenden Pfad auf Ihrem Webserver:

```
.well-known/aws/securityagent-domain-verification.json
```

2. Platzieren Sie das vom AWS Security Agent bereitgestellte Token in diesem Format in der Datei:

```
{
  "tokens": ["<insert-token>"]
}
```

3. Kehren Sie zur Tabelle mit den Zieldomänen zurück, wählen Sie die Domain aus und wählen Sie Verify aus. Der AWS Security Agent sendet eine HTTPS-GET-Anfrage an die Bestätigungs-URL und validiert das Token.
4. Wenn die Domain im öffentlichen Internet zugänglich ist, stellen Sie sicher, dass Ihre Domain über ein gültiges SSL-Zertifikat verfügt, bevor Sie die Überprüfung durchführen.

> **Note**
>
> Wenn Ihre Domain in mehreren Agentenbereichen registriert ist und Sie die HTTP-Routenvalidierung verwenden, können Sie die Token für beide Agentenbereiche im selben `tokens` Array platzieren.

Umgehen Sie die Überprüfung des Domainbesitzes

Kunden, die berechtigt sind, Penetrationstests an einem Endpunkt durchzuführen, aber die Eigentumsverifizierung nicht abschließen können, können bei uns eine manuelle Überprüfung beantragen. Bitte öffnen Sie eine Kundendienstanfrage, um diese Anfrage zu stellen. Ihre Anfrage muss Ihren geschäftlichen Anwendungsfall und Ihre Begründung für die manuelle Überprüfung der Inhaberschaft enthalten (d. h., warum Sie berechtigt sind, Penetrationstests am Endpunkt durchzuführen).

Verwaltete Zieldomänen, die für Penetrationstests verwendet werden

In der AWS-Managementkonsole können Sie Zieldomänen hinzufügen und verwalten, die von Agent-Spaces für Penetrationstests genutzt werden. Diese Zieldomänen müssen verifiziert werden, bevor sie in einem Penetrationstest verwendet werden können. Weitere Informationen zur Überprüfung von Zieldomänen finden Sie unter the section called “Penetrationstest aktivieren”

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

1. Der Penetrationstest wurde aktiviert (siehethe section called “Penetrationstest aktivieren”)

Ressourcen der Zieldomäne verwalten

- Navigieren Sie zur Übersichtsseite der Zieldomänen.
- Sie sollten alle mit Ihrem Konto verknüpften Zieldomänenressourcen sehen
  - Wenn Sie keine mit Ihrem Konto verknüpften Zieldomänen sehen, folgen Sie den Schritten unter the section called “Penetrationstest aktivieren” So erstellen und verifizieren Sie eine Zieldomain
- Zieldomänen können zwischen Agent-Spaces und Share-Bestätigungsstatus wiederverwendet werden

  - Um eine bestehende Zieldomäne zu einem Agentenbereich hinzuzufügen, navigieren Sie zur Registerkarte Penetrationstest des Agentenbereichs. Wählen Sie Domain hinzufügen und wählen Sie im Feld Domainname unter Aus verfügbaren, zuvor registrierten Domains auswählen die gewünschte Domain aus
  - Zieldomänen müssen einem Agentbereich zugeordnet werden, bevor sie in einem Penetrationstest verwendet werden können
- Durch das Entfernen einer Domäne aus einem Agentenbereich wird die Domäne nicht gelöscht. Die zugehörige Zieldomäne kann auf der Übersichtsseite der Zieldomänen dauerhaft gelöscht werden

Überprüfen Sie die Zieldomänen

Damit eine Zieldomain in einem Penetrationstest verwendet werden kann, muss sie zunächst mit einer der folgenden Methoden verifiziert werden:

- Route-53-Domains (dasselbe AWS-Konto): Wählen Sie One-click Verifizierung. Der AWS Security Agent erstellt automatisch den DNS-Eintrag und schließt die Überprüfung ab.
- DNS-TXT (andere DNS-Anbieter): Kopieren Sie das Verifizierungstoken, fügen Sie den TXT-Eintrag bei Ihrem DNS-Registrar hinzu, wählen Sie dann die Domain aus und wählen Sie Verify.
- HTTP-Route: Platzieren Sie das Bestätigungstoken am erforderlichen Routenpfad auf Ihrem Webserver, wählen Sie dann die Domain aus und wählen Sie Verifizieren. Details hierzu finden Sie unter the section called “Aktivieren Sie eine Anwendungsdomäne für Penetrationstests”.
- Private VPC: Überprüft, ob die IP der Zieldomain in einen privaten CIDR-Bereich fällt (eine Liste der privaten CIDR-Bereiche finden Sie unterthe section called “Agent mit privaten VPC-Ressourcen verbinden”). Der Domainname des Penetrationstest-Zielendpunkts muss mit dem vollständigen Domainnamen übereinstimmen, der für die Überprüfung konfiguriert wurde. Nur für private VPC-Penetrationstests nutzbar

> **Note**
>
> Bei der DNS-TXT-, HTTP-Route- und Route 53-Überprüfung werden Subdomänen einer verifizierten Domain automatisch abgedeckt und erfordern keine separate Überprüfung. Für die private VPC-Überprüfung muss jede Domain einzeln verifiziert werden.

Sicherheitsanforderungen verwalten

Definieren Sie die Sicherheitsanforderungen, die AWS Security Agent bei Design- und Codeprüfungen anhand von AWS-managed Paketen, benutzerdefinierten Paketen oder Anforderungen, die aus Ihrer eigenen Dokumentation importiert wurden, bewertet.

Sicherheitsanforderungen verwalten

Organisieren Sie die Sicherheitsanforderungen, die der AWS Security Agent zur Analyse Ihrer Anwendungen verwendet, in Sicherheitsanforderungspaketen. Ein Paket ist eine Sammlung von Sicherheitsanforderungen. Sie können AWS verwaltete Pakete aktivieren, Ihre eigenen benutzerdefinierten Pakete erstellen und Anforderungen für ein benutzerdefiniertes Paket generieren, indem Sie Ihre Sicherheitsdokumentation hochladen.

-Übersicht

Eine Sicherheitsanforderung definiert einen Sicherheitsstandard oder eine Sicherheitsrichtlinie, anhand derer der AWS Security Agent Ihre Anwendung bei Design- und Code-Reviews bewertet. Wenn ein Design oder ein Code eine Anforderung nicht erfüllt, meldet der AWS Security Agent ein Ergebnis. Jede Sicherheitsanforderung gehört zu einem Sicherheitsanforderungspaket. Sicherheitsanforderungen werden von allen Agent Spaces gemeinsam genutzt und bei Entwurfs- und Code-Reviews bewertet.

AWS Der Security Agent bietet zwei Arten von Paketen, die auf separaten Registerkarten angezeigt werden:

- Pakete mit verwalteten Sicherheitsanforderungen AWS— bereitgestellte Pakete mit Sicherheitsanforderungen, die auf Industriestandards und bewährten Methoden basieren. Sie können diese Pakete sofort aktivieren, um sie anhand gängiger Sicherheitsrichtlinien zu bewerten, ohne dass eine benutzerdefinierte Konfiguration erforderlich ist. Die Anforderungen in einem verwalteten Paket sind schreibgeschützt. Sie können eine AWS verwaltete Anforderung als Vorlage für eine benutzerdefinierte Anforderung verwenden. Eine Liste der verfügbaren Managed Packs finden Sie unter AWS Managed Security Requirement Packs.
- Benutzerdefinierte Sicherheitsanforderungspakete — Pakete, die Sie erstellen, um die spezifischen Richtlinien und Standards Ihres Unternehmens durchzusetzen. Sie erstellen die Anforderungen in einem benutzerdefinierten Paket von Grund auf neu, passen AWS die verwalteten Anforderungen als Ausgangspunkt an oder generieren sie, indem Sie Ihre Sicherheitsdokumentation hochladen.

> **Note**
>
> Die Compliance-Framework-Pakete sollen dabei helfen, Sicherheitsanforderungen zu identifizieren, die für die Einhaltung bestimmter Frameworks relevant sein können. Sie bewerten weder Ihre Einhaltung noch garantieren sie, dass Sie ein Audit bestehen werden.

Sie aktivieren und deaktivieren Packs als Einheit. Wenn ein Pack aktiviert ist, bewertet der AWS Security Agent Ihre Anwendungen bei Design- und Code-Reviews anhand der Anforderungen in diesem Paket.

> **Note**
>
> Die Aktivierung oder Deaktivierung eines Pakets gilt für neue Design- und Code-Reviews. Bestehende Bewertungen sind nicht betroffen.

Ein verwaltetes Paket aktivieren oder deaktivieren

Aktivieren Sie ein AWS-managed Pack, um Ihre Anwendungen anhand einer Reihe von AWS-verwalteten Sicherheitsanforderungen zu bewerten. Deaktivieren Sie es, wenn Sie es nicht mehr benötigen.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Managed Security Requirements Packs aus.
4. Wählen Sie das Paket aus, das Sie aktivieren oder deaktivieren möchten.
5. Führen Sie eine der folgenden Aktionen aus:
   a. Um das Pack zu aktivieren, wählen Sie Paket aktivieren.
   b. Um das Pack zu deaktivieren, wählen Sie Disable Pack.

> **Tip**
>
> Wählen Sie ein Paket aus, um die darin enthaltenen Sicherheitsanforderungen einzusehen. Wählen Sie eine Anforderung aus, um ihre vollständige Definition, einschließlich ihrer Anwendbarkeit, Konformitätskriterien und Anleitungen zur Problembehebung, einzusehen.

Erstellen Sie ein benutzerdefiniertes Paket

Erstellen Sie ein benutzerdefiniertes Paket, um Sicherheitsanforderungen zu gruppieren, die für Ihr Unternehmen spezifisch sind. Nachdem Sie das Paket erstellt haben, fügen Sie ihm manuell Anforderungen hinzu, passen Sie eine AWS verwaltete Anforderung an oder laden Sie Dokumente hoch, um Anforderungen zu generieren.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Pakete mit Sicherheitsanforderungen.
4. Wählen Sie Sicherheitsanforderungspaket erstellen aus.
5. Geben Sie einen Namen für das Sicherheitsanforderungspaket und optional eine Beschreibung ein.
6. Wählen Sie Sicherheitsanforderungspaket erstellen aus.

> **Note**
>
> Nachdem Sie ein Paket erstellt haben, können Sie Quelldokumente hinzufügen oder hochladen, um die Sicherheitsanforderungen für Ihr Paket zu generieren.

Fügen Sie manuell eine Sicherheitsanforderung hinzu

Fügen Sie einem benutzerdefinierten Paket eine Sicherheitsanforderung hinzu, um einen Sicherheitsstandard zu definieren, den der AWS Security Agent durchsetzt.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Pakete mit Sicherheitsanforderungen und wählen Sie dann das Paket aus, zu dem Sie eine Anforderung hinzufügen möchten.
4. Wählen Sie Anforderung manuell hinzufügen aus.
5. Konfigurieren Sie die folgenden Felder:
   a. Name der Sicherheitsanforderung — Geben Sie einen aussagekräftigen Namen ein, der beispielsweise die Sicherheitskontrolle identifiziert `enforce-encryption-at-rest` (maximal 80 Zeichen).

b. Beschreibung — Geben Sie eine kurze Beschreibung dessen an, was mit dieser Sicherheitsanforderung geprüft wird (maximal 500 Zeichen).

c. Anwendbarkeit — Beschreiben Sie die Szenarien, Systemtypen oder Bedingungen, unter denen diese Sicherheitsanforderung bewertet werden sollte. Geben Sie die Szenarien an, in denen die Anforderung als NOT_APPLICABLE gekennzeichnet werden sollte (maximal 10.000 Zeichen).

d. Konformitätskriterien — Definieren Sie, was bei dieser Sicherheitsanforderung zwischen Einhaltung und Nichteinhaltung besteht. Geben Sie die Indikatoren, Beispiele und technischen Details an, nach denen der AWS Security Agent bei der Bewertung der Einhaltung der Vorschriften sucht (maximal 10.000 Zeichen).

e. Hinweise zur Problembehebung (optional) — Stellen Sie Anleitungen zur Behebung von Verstößen bereit, einschließlich Links zu internen Dokumentationen oder Standards Ihres Unternehmens (maximal 10.000 Zeichen).

6. Führen Sie eine der folgenden Aktionen aus:

a. Um die Anforderung zu erstellen, ohne sie zu aktivieren, wählen Sie Sicherheitsanforderung erstellen.

b. Um die Anforderung zu erstellen und zu aktivieren, wählen Sie Sicherheitsanforderung erstellen und aktivieren.

> **Note**
>
> Eine Anforderung wird bei Sicherheitsüberprüfungen nur dann bewertet, wenn das Paket, das sie enthält, aktiviert ist. Um zu kontrollieren, ob ein Pack bewertet wird, aktivieren oder deaktivieren Sie das Pack.

Passen Sie eine an AWS-verwaltete Sicherheitsanforderung

Erstellen Sie eine benutzerdefinierte Sicherheitsanforderung, die eine AWS verwaltete Anforderung als Ausgangspunkt verwendet. AWS Der Security Agent füllt die Anforderungsdetails aus der verwalteten Anforderung vorab aus, und Sie bearbeiten sie so, dass sie zu Ihrer Organisation passen.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Sicherheitsanforderungspakete und dann das benutzerdefinierte Paket aus, zu dem die Anforderung hinzugefügt werden soll.

4. Wählen Sie Benutzerdefinierte Sicherheitsanforderung erstellen aus.
5. Um das Formular vorab auszufüllen, suchen Sie im Abschnitt Eine AWS verwaltete Sicherheitsanforderung anpassen nach einer -verwalteten Anforderung, die Sie als AWS Vorlage verwenden möchten, und wählen Sie sie aus.
6. Bearbeiten Sie alle Felder entsprechend den Anforderungen Ihrer Organisation.
7. Führen Sie eine der folgenden Aktionen aus:
   a. Um die Anforderung zu erstellen, ohne sie zu aktivieren, wählen Sie Sicherheitsanforderung erstellen.
   b. Um die Anforderung zu erstellen und sofort zu aktivieren, wählen Sie Sicherheitsanforderung erstellen und aktivieren.

> **Note**
>
> Durch das Anpassen einer AWS verwalteten Anforderung wird eine unabhängige benutzerdefinierte Sicherheitsanforderung erstellt. Spätere Änderungen an der AWS-verwalteten Anforderung haben keine Auswirkungen auf Ihre benutzerdefinierte Version.

Generieren Sie Anforderungen, indem Sie Dokumente hochladen

Generieren Sie die Sicherheitsanforderungen für ein benutzerdefiniertes Paket anhand Ihrer vorhandenen Sicherheitsdokumentation, anstatt jede Anforderung von Hand zu schreiben. AWS Der Security Agent liest die von Ihnen hochgeladenen Dokumente, identifiziert den sicherheitsrelevanten Inhalt und generiert strukturierte Anforderungen, die Sie überprüfen und bearbeiten können. Das vollständige Verfahren finden Sie unter Generieren von Sicherheitsanforderungen anhand von Dokumenten. Hinweise dazu, was in den von Ihnen hochgeladenen Dokumenten enthalten sein muss, finden Sie unter Dokumente für die Generierung von Anforderungen vorbereiten.

> **Important**
>
> Durch das Hochladen neuer Quelldokumente werden alle Anforderungen für das Paket neu generiert. Alle vorhandenen Anforderungen, auch solche, die Sie manuell hinzugefügt haben, werden ersetzt.

Bearbeiten Sie eine benutzerdefinierte Sicherheitsanforderung

Ändern Sie eine benutzerdefinierte Sicherheitsanforderung, um ihre Definition, Kriterien oder Hinweise zur Problembehebung zu aktualisieren. Sie können die Anforderungen in einem Paket mit AWS verwaltetem System nicht bearbeiten.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Pakete mit Sicherheitsanforderungen und wählen Sie dann das Paket aus, das die Anforderung enthält.
4. Wählen Sie die Anforderung aus, die Sie bearbeiten möchten, und klicken Sie dann auf Benutzerdefinierte Sicherheitsanforderung bearbeiten.
5. Aktualisieren Sie die Anforderungsfelder nach Bedarf. Der Name der Sicherheitsanforderung kann nach der Erstellung nicht geändert werden.
6. Wählen Sie Sicherheitsanforderung aktualisieren aus.

> **Note**
>
> Änderungen an einer Sicherheitsanforderung gelten für neue Design- und Code-Reviews. Bestehende Bewertungen sind nicht betroffen.

Aktiviere oder deaktiviere ein Pack

Aktivieren oder deaktivieren Sie ein Sicherheitsanforderungspaket, um zu steuern, ob der AWS Security Agent die darin enthaltenen Anforderungen auswertet. Sie aktivieren und deaktivieren Packs als Einheit.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte für den Pakettyp und dann das Paket aus.
4. Führen Sie eine der folgenden Aktionen aus:
   a. Um das Paket zu aktivieren, wählen Sie Paket aktivieren.
   b. Um das Pack zu deaktivieren, wählen Sie Disable Pack.

> **Note**
>
> Wenn ein Paket aktiviert ist, werden die in dem Paket enthaltenen Anforderungen im Rahmen von Sicherheitsüberprüfungen bewertet. Wenn ein Pack deaktiviert ist, werden seine Anforderungen nicht bewertet.

Löschen Sie eine benutzerdefinierte Sicherheitsanforderung

Löschen Sie eine benutzerdefinierte Sicherheitsanforderung, wenn Sie nicht mehr möchten, dass der AWS Security Agent sie auswertet.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Pakete mit Sicherheitsanforderungen und wählen Sie dann das Paket aus, das die Anforderung enthält.
4. Wählen Sie eine oder mehrere Sicherheitsanforderungen aus und klicken Sie dann auf Sicherheitsanforderung löschen.
5. Bestätigen Sie das Löschen.

> **Note**
>
> Wenn Sie Sicherheitsanforderungen löschen, werden sie bei future Überprüfungen nicht mehr bewertet. Die Anforderungen sind in früheren Bewertungen weiterhin als schreibgeschützte Ergebnisse sichtbar.

Bewährte Methoden für die Definition von Sicherheitsanforderungen

Wenn Sie eine Sicherheitsanforderung erstellen, befolgen Sie diese Richtlinien, damit der AWS Security Agent Ihre Anwendungen genau bewerten und umsetzbare Ergebnisse liefern kann.

Name der Sicherheitsanforderung — Verwenden Sie einen eindeutigen, spezifischen Namen, der die Sicherheitskontrolle identifiziert. Vermeiden Sie allgemeine Begriffe, die den Zweck der Anforderung nicht vermitteln.

Beschreibung — Erläutern Sie, was die Kontrolle überprüft und welches Risiko dadurch gemindert wird. Dies hilft Benutzern zu verstehen, warum die Anforderung wichtig ist.

Anwendbarkeit — Beschreiben Sie die Workloads, Systemtypen oder Bedingungen, unter denen die Anforderung bewertet werden sollte. Geben Sie an, wann die Anforderung als NOT_APPLICABLE gekennzeichnet werden sollte, und geben Sie spezifische Szenarien an, um Fehlalarme zu vermeiden. Formulierungen wie „Dieses Steuerelement gilt für ALLE Workloads, die...“ und „Als NOT_ANWENDBAR markieren, wenn...“ setzen klare Grenzen für den Anwendungsbereich.

Compliance-Kriterien — Gliedern Sie diese Kriterien in zwei Teile: Was zeigt die Einhaltung der Vorschriften und was deutet auf eine Nichteinhaltung hin? Seien Sie bei den technischen Indikatoren spezifisch und beziehen Sie Sonderfälle mit ein. Beginnen Sie mit „Ein Design ist konform, wenn es nachweist...“, gefolgt von technischen Details, dann „Ein Design ist nicht konform, wenn es...“ mit den Mustern, die auf einen Verstoß hinweisen.

Anleitung zur Problembehebung — Stellen Sie Anleitungen mit technischen Details und Konfigurationsbeispielen bereit. Fügen Sie Links zur internen Dokumentation Ihres Unternehmens hinzu, damit Entwickler über die Ressourcen verfügen, die sie zur Behebung von Verstößen benötigen.

Nächste Schritte

Nachdem Sie Ihre Sicherheitsanforderungspakete konfiguriert haben:

- Erstellen Sie eine Design- oder Codeüberprüfung, um Ihre Anwendung anhand der von Ihnen aktivierten Packs zu bewerten.
- Ermöglichen Sie Penetrationstests, um Ihre Design- und Code-Reviews zu ergänzen.
- Gewähren Sie Benutzern Zugriff auf die AWS Security Agent Web-App.

Von AWS verwaltete Sicherheitsanforderungspakete

Ein AWS verwaltetes Sicherheitsanforderungspaket ist eine Sammlung von Sicherheitsanforderungen, die auf der Grundlage von Industriestandards und bewährten Verfahren AWS erstellt und verwaltet werden. Sie ermöglichen es einem Managed Pack, Ihre Anwendungen im Rahmen von Design- und Code-Reviews anhand der Anforderungen zu bewerten, ohne selbst Anforderungen schreiben zu müssen.

Die Sicherheitsanforderungen in einem Managed Pack sind schreibgeschützt. Sie können sie nicht ändern. Sie können eine AWS verwaltete Anforderung als Vorlage verwenden, um eine

benutzerdefinierte Anforderung zu erstellen, und die Kopie dann so bearbeiten, dass sie zu Ihrer Organisation passt. Weitere Informationen finden Sie unter Sicherheitsanforderungen verwalten.

AWS aktualisiert verwaltete Pakete im Laufe der Zeit. Wenn eine Anforderung in einem verwalteten Paket AWS hinzugefügt oder geändert wird, gilt die Änderung für jedes Konto, für das das Paket aktiviert ist.

> **Note**
>
> Die Compliance-Framework-Pakete sollen dabei helfen, Sicherheitsanforderungen zu identifizieren, die für die Einhaltung bestimmter Frameworks relevant sein können. Sie bewerten weder Ihre Einhaltung noch garantieren sie, dass Sie ein Audit bestehen werden.

AWS-verwaltetes Paket: ASA-Basispaket

Eine Reihe grundlegender Sicherheitsanforderungen, die für die meisten Workloads gelten. Dieses Paket behandelt allgemeine Sicherheitsprobleme bei Anwendungen wie Authentifizierung und Autorisierung, Datenschutz, Protokollierung und Eingabevalidierung. Aktivieren Sie dieses Paket, um eine Sicherheitsgrundlage für Ihre Design- und Code-Reviews zu schaffen.

AWS-verwaltetes Paket: Well-Architected AWS-Paket

Sicherheitsanforderungen, die sich aus der Sicherheitssäule des AWS Well-Architected Frameworks ergeben. In diesem Paket wird Ihre Anwendung anhand von AWS Richtlinien zum Schutz von Daten, zur Verwaltung von Identitäten und Berechtigungen sowie zur Erkennung und Reaktion auf Sicherheitsereignisse bewertet. Weitere Informationen zum Framework finden Sie unter Security Pillar — AWS Well-Architected Framework.

AWS-verwaltetes Paket: NIST CSF-Paket

Aus dem NIST Cybersecurity Framework (CSF) abgeleitete Sicherheitsanforderungen. Dieses Paket bewertet Ihre Anwendung anhand von Kontrollbereichen, die sich aus dem Framework ergeben, wie Identitäts- und Zugriffsmanagement, Datensicherheit und Schutztechnologie. Aktivieren Sie dieses Paket, um Ihre Design- und Codeprüfungen an NIST CSF anzupassen.

AWS-verwaltetes Paket: PCI DSS-Paket

Aus dem Payment Card Industry Data Security Standard (PCI DSS) abgeleitete Sicherheitsanforderungen. In diesem Paket wird Ihre Anwendung anhand von Kontrollbereichen

bewertet, die für den Umgang mit Karteninhaberdaten relevant sind, z. B. Zugriffskontrolle, Verschlüsselung von Daten bei der Übertragung und Speicherung sowie Protokollierung. Aktivieren Sie dieses Paket, um Ihre Design- und Codeprüfungen auf PCI DSS abzustimmen.

Generieren Sie Sicherheitsanforderungen aus Dokumenten

Generieren Sie die Sicherheitsanforderungen für ein benutzerdefiniertes Paket, indem Sie Ihre vorhandene Sicherheitsdokumentation hochladen. AWS Der Security Agent liest die von Ihnen hochgeladenen Dokumente, identifiziert den sicherheitsrelevanten Inhalt und generiert strukturierte Sicherheitsanforderungen mit Anwendbarkeit, Compliance-Kriterien und Anleitungen zur Problembehebung. Sie können die generierten Anforderungen überprüfen und bearbeiten, bevor sie für Überprüfungen verwendet werden.

Verwenden Sie dies, anstatt jede Anforderung von Hand zu schreiben, wenn Sie bereits Sicherheitsrichtlinien, Standards oder Richtlinien dokumentiert haben. Hinweise dazu, was in den von Ihnen hochgeladenen Dokumenten enthalten sein sollte, finden Sie unter Dokumente für die Generierung von Anforderungen vorbereiten.

Unterstützte Datei-Formate

Sie können die folgenden Dateiformate hochladen:

- DOKUMENT
- DOCX
- MD
- PDF
- TXT

Anforderungen generieren

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Pakete mit Sicherheitsanforderungen aus.
4. Erstellen Sie ein Paket oder wählen Sie ein vorhandenes benutzerdefiniertes Paket aus, für das Sie Anforderungen generieren möchten.
5. Wählen Sie Dateien auswählen oder ziehen Sie Ihre Dokumente per Drag & Drop in den Upload-Bereich.

6. Wählen Sie Anforderungen generieren.
7. Warten Sie, bis der AWS Security Agent die Dokumente verarbeitet hat. Die Generierung läuft im Hintergrund und kann bei großen Dateien einige Minuten dauern. Während der Generierung können Sie keinen weiteren Upload für das Paket starten.
8. Überprüfen Sie die generierten Sicherheitsanforderungen. Bearbeiten, löschen oder fügen Sie Anforderungen nach Bedarf hinzu. Aktivieren Sie das Pack, wenn der AWS Security Agent seine Anforderungen bei Sicherheitsüberprüfungen bewerten soll.

> **Note**
>
> AWS Der Security Agent generiert Anforderungen auf Workload-Ebene und konzentriert sich dabei auf die sicherheitsrelevanten Inhalte in Ihren Dokumenten. Die Anzahl der generierten Anforderungen hängt von den von Ihnen bereitgestellten Inhalten ab. Überprüfen Sie die generierten Anforderungen, um sicherzustellen, dass sie Ihre Absicht widerspiegeln.

Ersetzen Sie die Anforderungen durch einen neuen Upload

Durch das Hochladen neuer Quelldokumente werden die Anforderungen für ein Paket neu generiert.

> **Important**
>
> Wenn Sie neue Quelldokumente in ein Paket hochladen, generiert der AWS Security Agent alle Anforderungen für dieses Paket neu. Alle vorhandenen Anforderungen, einschließlich der manuell hinzugefügten, werden ersetzt. Um Ihre aktuellen Anforderungen beizubehalten, laden Sie keine neuen Dokumente hoch.

1. Navigieren Sie in der AWS Konsole zu AWS Security Agent.
2. Wählen Sie im Navigationsbereich Sicherheitsanforderungen aus.
3. Wählen Sie die Registerkarte Benutzerdefinierte Pakete mit Sicherheitsanforderungen und wählen Sie dann das Paket aus.
4. Starten Sie einen neuen Upload und bestätigen Sie, dass das Hochladen neuer Quelldokumente die bestehenden Anforderungen ersetzt.
5. Wählen Sie Ihre Dateien aus und klicken Sie dann auf Anforderungen generieren.

Nächste Schritte

- Überprüfen und aktivieren Sie die generierten Anforderungen. Siehe Sicherheitsanforderungen verwalten.
- Erstellen Sie ein Design- oder Code-Review, um Ihre Anwendung mit der Masse zu vergleichen.

Dokumente für die Generierung von Anforderungen vorbereiten

Wenn Sie Sicherheitsanforderungen anhand von Dokumenten generieren, liest der AWS Security Agent Ihre Dokumente und erstellt anhand der gefundenen sicherheitsrelevanten Inhalte strukturierte Anforderungen. Sie müssen Ihre Dokumente nicht auf eine bestimmte Art formatieren oder vorab die Anwendbarkeit, die Compliance-Kriterien und Anleitungen zur Problembehebung verfassen. AWS Der Security Agent leitet diese aus dem von Ihnen bereitgestellten Inhalt ab. Laden Sie die Sicherheitsdokumentation hoch, die Sie bereits haben und die in Ihren eigenen Worten verfasst wurde.

Auf dieser Seite wird beschrieben, was enthalten sein muss, damit der AWS Security Agent genaue und nützliche Anforderungen generiert. Informationen zum Upload-Verfahren und zu den Dateibeschränkungen finden Sie unter Generieren von Sicherheitsanforderungen anhand von Dokumenten.

Was sollte enthalten sein

AWS Der Security Agent sucht nach Inhalten, die Ihre Sicherheitserwartungen beschreiben. Dokumente, die die folgenden Themen behandeln, stellen die strengsten Anforderungen dar:

- Zugriffskontrolle und Autorisierung
- Authentifizierung
- Datenschutz und Verschlüsselung
- Netzwerksicherheit
- Protokollierung und Prüfung
- Vorfallreaktion
- Sicherheitslücken- und Patch-Management
- Compliance-Verpflichtungen, die für Ihre Workloads gelten

Zu guten Quelldokumenten gehören Sicherheitsrichtlinien, technische Standards, Kontrollkataloge, Architekturrichtlinien und Standards für sichere Codierung.

Schreiben Sie auf Workload-Ebene

AWS Der Security Agent generiert Anforderungen, die für einen einzelnen Workload oder eine einzelne Anwendung gelten, und zwar in demselben Umfang, den er bei Entwurfs- und Code-Reviews bewertet. Inhalt, der beschreibt, wie ein sicherer Workload erstellt und betrieben wird, generiert Anforderungen. Organization-wide Governance-Themen wie die Strategie für mehrere Konten, die Verwaltung von Root-Benutzern und die Einrichtung der Protokollierung auf Kontoebene fallen nicht in diesen Bereich und stellen keine Workload-Anforderungen dar.

Beschreiben Sie das Ergebnis, nicht eine einzelne Implementierung

Geben Sie das Sicherheitsergebnis an, das Sie erwarten, und nicht ein einzelnes vorgeschriebenes Produkt oder eine einzelne Konfiguration. AWS Der Security Agent generiert Anforderungen, die sich auf die benötigten Sicherheitsfunktionen konzentrieren und mehr als eine gültige Implementierung ermöglichen. Beispielsweise generiert „Daten im Ruhezustand sind verschlüsselt“ eine klarere, umfassendere Anforderung als eine Aussage, die einen bestimmten Dienst oder eine bestimmte Einstellung benennt.

Geben Sie genau an, wie gut aussieht

Je konkreter Ihre Dokumente das erwartete Verhalten beschreiben, desto besser kann AWS Security Agent Compliance-Kriterien definieren. Beschreiben Sie, wo möglich, was ein konformes Design zeigt und was auf einen Verstoß hindeutet. Vage oder rein zielgerichtete Aussagen führen zu weniger und schwächeren Anforderungen als konkrete, auslastungsrelevante Aussagen.

Überprüfe, was du zurückbekommst

Jede generierte Anforderung umfasst einen Namen, eine Anwendbarkeit, Konformitätskriterien und Anleitungen zur Problembehebung, die der AWS Security Agent aus Ihren Dokumenten abgeleitet hat. Überprüfen Sie die generierten Anforderungen, bearbeiten Sie alle, die noch verfeinert werden müssen, und aktivieren Sie die Anforderungen, die der AWS Security Agent auswerten soll. Weitere Informationen finden Sie unter Sicherheitsanforderungen verwalten.

Benutzer und Zugriff verwalten

Steuern Sie, wer auf jeden Agent Space zugreifen kann, und konfigurieren Sie die IAM-Rollen und Verschlüsselungsschlüssel, die AWS Security Agent verwendet.

Benutzern Zugriff auf die AWS Security Agent-Web-App gewähren

AWS Security Agent bietet Benutzern zwei Methoden für den Zugriff auf die Webanwendung, je nachdem, wie Sie Ihren Agent Space während der Einrichtung konfiguriert haben.

Übersicht über die Zugriffsmethoden

IAM Identity Center (SSO) — Wenn Sie IAM Identity Center bei der Erstellung Ihres Agent Space aktiviert haben, können Benutzer direkt über SSO auf die Webanwendung zugreifen. Sie weisen dem Agent Space über die AWS Security Agent-Konsole Benutzer zu, und Benutzer melden sich mit ihren Identity Center-Anmeldeinformationen an.

Administratorzugriff — Benutzer mit AWS-Konsolenzugriff können die Webanwendung über einen Administratorzugriffslink auf der Agent Space-Übersichtsseite in der AWS-Managementkonsole starten.

Gewähren Sie Zugriff mit IAM Identity Center (SSO)

Wenn Sie Ihren Agent Space mit IAM Identity Center konfiguriert haben, können Sie dem Agent Space mithilfe der AWS Security Agent-Konsole Benutzer zuweisen.

Benutzer über die AWS Security Agent-Konsole zuweisen

1. Navigieren Sie in der AWS Security Agent Management Console zu Ihrem Agent Space.
2. Wählen Sie die Registerkarte Web-App aus.
3. Wählen Sie in der Tabelle Benutzer die Option Benutzer hinzufügen aus.
4. Wählen Sie vorhandene Benutzer aus IAM Identity Center aus oder erstellen Sie neue Benutzer.
5. Bestätigen Sie die Benutzerzuweisungen.

> **Tip**
>
> Benutzer, die dem Agent Space zugewiesen sind, können auf die Webanwendung zugreifen, indem sie sich mit ihren SSO-Anmeldeinformationen über das IAM Identity Center anmelden.

Greifen Sie mit SSO auf die Webanwendung zu

Nachdem Benutzer dem Agent Space zugewiesen wurden:

1. Benutzer navigieren zur URL der Webanwendung für den Agent Space.

> **Tip**
>
> Suchen Sie die Web-App-URL auf der Agent-Space-Detailseite in der AWS Security Agent-Konsole, indem Sie Web-App-URL kopieren auswählen. Benutzer sollten diese URL mit einem Lesezeichen versehen, damit sie leicht darauf zugreifen können.

2. Benutzer melden sich mit ihren SSO-Anmeldeinformationen an.
3. Nach der Authentifizierung können Benutzer den Agent Space auswählen und mit der Durchführung von Sicherheitsbewertungen beginnen.

Zugriff mit IAM-only Zugriff gewähren

Wenn Sie Ihren Agent Space mit IAM-only Zugriff konfiguriert haben, können Benutzer mit AWS-Konsolenzugriff die Webanwendung über einen Administratorzugriffslink starten.

Verwenden Sie den Admin-Zugangslink

1. Melden Sie sich bei der AWS Security Agent-Konsole an.
2. Navigieren Sie zu dem Agent-Bereich, auf den Sie zugreifen möchten.
3. Wählen Sie auf der Agent Space-Landingpage auf der Registerkarte Web-App die Schaltfläche Admin-Zugriff, um die Webanwendung zu starten.
4. Die Webanwendung wird auf einer neuen Registerkarte geöffnet, wobei der Benutzer automatisch authentifiziert wird.

> **Note**
>
> Der Administratorzugriffslink ist nur für Benutzer verfügbar, die bereits mit den entsprechenden AWS Security Agent-Berechtigungen an der AWS-Konsole authentifiziert sind. Für diese Methode ist keine IAM Identity Center-Konfiguration erforderlich.

Nächste Schritte

Nachdem Sie Benutzern Zugriff auf die Webanwendung gewährt haben:

- Benutzer können Konfigurationen und Durchläufe von Penetrationstests erstellen und verwalten
- Benutzer können Entwurfsprüfungen erstellen und verwalten
- Benutzer können Sicherheitsfeststellungen und Anleitungen zur Problembehebung einsehen
- Konfigurieren Sie die Benachrichtigungseinstellungen für Sicherheitslücken

Eine IAM-Rolle für AWS Security Agent erstellen

AWS Security Agent verwendet IAM-Rollen auf drei Arten:

1. Anwendungsrolle: Wird beim Erstellen der AWS Security Agent-Anwendung verwendet. In Anwendungsfällen mit IAM Identity Center und Admin-Zugriffslinks übernimmt der Service diese Rolle, um WebApp Benutzern Berechtigungen zur Interaktion mit AWS Security Agent-APIs zu gewähren.
2. Rolle des Penetrationstest-Service: Wird beim Erstellen von Agent-Spaces als Liste verfügbarer Rollen angegeben. Später wählen WebApp Benutzer bei der Erstellung eines Penetrationstests eine dieser Rollen aus. Der AWS Security Agent Service übernimmt diese Rolle, um während des Tests auf Ihre AWS-Ressourcen zuzugreifen.
3. Rolle des Akteurs: Wird verwendet, um Anfragen an Ihre Ziel-Webanwendung (z. B. AWS API Gateway Gateway-APIs) zu authentifizieren und zu autorisieren. Diese Rollen werden bei der Erstellung des Agentenbereichs bereitgestellt. Der AWS-Sicherheitsagent übernimmt Akteurrollen, um mit Ihrer Zielanwendung zu interagieren.

Rolle der Anwendung

Die Anwendungsrolle wird verwendet, wenn Sie Ihre AWS Security Agent-Anwendung im Service erstellen. Bei Authentifizierungsszenarien für IAM Identity Center und Administratorzugriffslinks übernimmt der AWS Security Agent-Service diese Rolle, um WebApp Benutzern die erforderlichen Berechtigungen für die Interaktion mit den AWS Security Agent-APIs zu gewähren.

Erforderliche Berechtigungen

Diese Rolle benötigt Berechtigungen für:

- API-Operationen für den AWS Security Agent aufrufen
- Lesen und Schreiben von Anwendungskonfigurationsdaten
- Auf Informationen zur Benutzersitzung zugreifen

- Authentifizierungstoken für WebApp Benutzer verwalten

Vertrauensrichtlinie

Die Vertrauensrichtlinie muss es dem AWS Security Agent-Service ermöglichen, diese Rolle zu übernehmen:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Berechtigungsrichtlinie

Die Rolle sollte Berechtigungen für Folgendes beinhalten:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "securityagent:GetApplication",
        "securityagent:UpdateApplication",
        "securityagent:ListAgentInstances",
        "securityagent:CreatePentestSession"
      ],
      "Resource": "arn:aws:securityagent:*:*:application/*"
    }
  ]
}
```

> **Note**
>
> Passen Sie die Berechtigungen an Ihre spezifischen Anwendungsanforderungen und das Prinzip der geringsten Rechte an.

Rolle des Penetrationstest-Dienstes

Die Penetrationstest-Servicerolle wird bei der Erstellung von Agentenbereichen als Liste verfügbarer Rollen angegeben. Wenn WebApp Benutzer einen Penetrationstest erstellen, wählen sie eine dieser Rollen aus. Der AWS Security Agent Service übernimmt dann diese Rolle, um auf Ihre AWS-Ressourcen zuzugreifen und diese zu testen.

Erforderliche Berechtigungen

Diese Rolle benötigt Berechtigungen, um während Penetrationstests auf Ihre AWS-Ressourcen zuzugreifen und diese zu analysieren:

- VPC-Konfigurationen und Netzwerktopologie lesen und beschreiben
- Untersuchen Sie EC2-Instances, Sicherheitsgruppen und Netzwerk-ACLs
- Analysieren Sie IAM-Richtlinien und Ressourcenberechtigungen
- CloudWatch Logs und Metriken lesen
- Zugriff auf AWS-Servicekonfigurationen, die für Sicherheitstests relevant sind

Vertrauensrichtlinie

Die Vertrauensrichtlinie muss es dem AWS Security Agent-Service ermöglichen, diese Rolle für Penetrationstests zu übernehmen:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
```

```
        "StringEquals": {
          "sts:ExternalId": "your-external-id"
        }
      }
    }
  ]
}
```

> **Note**
>
> Verwenden Sie eine externe ID für zusätzliche Sicherheit, wenn Sie kontoübergreifenden Zugriff oder Servicezugriff zulassen.

Berechtigungsrichtlinie

Die Rolle sollte Lesezugriff auf Ihre AWS-Ressourcen beinhalten. Erwägen Sie die Verwendung dieser verwalteten Richtlinien:

- `SecurityAudit`- Von AWS verwaltete Richtlinie für Sicherheitsüberprüfungen
- `ViewOnlyAccess`- Read-only Zugriff auf die meisten AWS-Services

Oder erstellen Sie eine benutzerdefinierte Richtlinie mit bestimmten Berechtigungen:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:Describe*",
        "vpc:Describe*",
        "iam:Get*",
        "iam:List*",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams",
        "cloudwatch:Describe*",
        "cloudwatch:Get*",
        "cloudwatch:List*",
        "s3:GetObject",
        "s3:ListBucket"
```

```
      ],
      "Resource": "*"
    }
  ]
}
```

> **Important**
>
> Erteilen Sie nur die Mindestberechtigungen, die für Ihren Penetrationstestbereich erforderlich sind. Überprüfen und passen Sie die Berechtigungen an, je nachdem, welche AWS-Services Sie in Sicherheitstests einbeziehen möchten.

Rolle des Schauspielers

Die Rolle „Akteur“ wird verwendet, um Anfragen an Ihre Ziel-Webanwendung während Penetrationstests zu authentifizieren und zu autorisieren. Diese Rollen werden bei der Erstellung des Agentenbereichs bereitgestellt, und der AWS Security Agent übernimmt sie für die Interaktion mit Ihren Zielanwendungsendpunkten (wie AWS API Gateway Gateway-APIs, Lambda-Funktions-URLs oder anderen AWS-hosted Anwendungen).

Erforderliche Berechtigungen

Diese Rolle benötigt Berechtigungen für:

- API-Gateway-Endpunkte aufrufen
- Lambda-Funktionen ausführen
- Greifen Sie auf anwendungsspezifische AWS-Ressourcen zu
- Authentifizieren Sie sich mit den Authentifizierungsmechanismen Ihrer Zielanwendung
- Führen Sie HTTP-Operationen für Ihre Anwendungsendpunkte durch

Vertrauensrichtlinie

Die Vertrauensrichtlinie muss es dem AWS Security Agent Agent Service ermöglichen, diese Rolle zu übernehmen:

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "your-external-id"
        }
      }
    }
  ]
}
```

Berechtigungsrichtlinie

Die Berechtigungen hängen von Ihrer Zielanwendungsarchitektur ab. Hier sind Beispiele für gängige Szenarien:

Für API-Gateway-Anwendungen

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "execute-api:Invoke"
      ],
      "Resource": "arn:aws:execute-api:us-east-1:*:your-api-id/*"
    }
  ]
}
```

Für Lambda-Funktions-URLs

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

```
      "Action": [
        "lambda:InvokeFunctionUrl",
        "lambda:InvokeFunction"
      ],
      "Resource": "arn:aws:lambda:us-east-1:*:function:your-function-name"
    }
  ]
}
```

Für Application Load Balancer mit Cognito-Authentifizierung

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cognito-idp:InitiateAuth",
        "cognito-idp:RespondToAuthChallenge"
      ],
      "Resource": "arn:aws:cognito-idp:us-east-1:*:userpool/your-user-pool-id"
    }
  ]
}
```

> **Important**
>
> Konfigurieren Sie Actor Role-Berechtigungen so, dass sie den Authentifizierungs- und Autorisierungsanforderungen Ihrer Zielanwendung entsprechen. Die Rolle sollte dieselbe Zugriffsebene haben wie die Benutzer oder Dienste, die der Security Agent bei Penetrationstests simuliert.

Vom Kunden verwaltete Schlüssel für AWS Security Agent

Standardmäßig verschlüsselt der AWS Security Agent alle Daten im Ruhezustand mit AWS verwalteten Verschlüsselungsschlüsseln. Sie können optional einen vom Kunden verwalteten Schlüssel des AWS Key Management Service (AWS KMS) verwenden, um Ihre Daten zu verschlüsseln, sodass Sie die volle Kontrolle über die Verschlüsselungsschlüssel haben, die Ihre Ressourcen schützen.

AWS Der Security Agent unterstützt vom Kunden verwaltete Schlüssel auf Ressourcenebene. Wenn Sie eine Ressource der obersten Ebene, wie z. B. einen Agent Space oder eine Integration, erstellen, können Sie einen KMS-Schlüssel angeben, um alle Daten zu verschlüsseln, die zu dieser Ressource und ihren Unterressourcen gehören. Wenn Sie beispielsweise bei der Erstellung eines Agent Space einen vom Kunden verwalteten Schlüssel angeben, werden alle mit diesem Agent Space verknüpften Daten verschlüsselt, einschließlich Agent Space-Konfigurationen, Penetrationstestkonfigurationen, Jobs und Ausführungsdetails, Sicherheitserkenntnisse, entdeckte Endpunkte und Screenshots. Sie können auch AWS Ressourcen bereitstellen, die bereits mit Ihrem eigenen, vom Kunden verwalteten Schlüssel verschlüsselt sind, z. B. S3-Buckets, CloudWatch Log-Protokollgruppen oder Secrets Manager Manager-Geheimnisse. Informationen zu den erforderlichen KMS-Berechtigungen finden Sie unterthe section called “Erforderliche KMS-Berechtigungen”.

So funktionieren vom Kunden verwaltete Schlüssel

AWS Der Security Agent verwendet den hierarchischen AWS KMS-Schlüsselbund, um Ihre Daten mit vom Kunden verwalteten Schlüsseln zu verschlüsseln. Wenn Sie bei der Ressourcenerstellung einen KMS-Schlüssel angeben, erstellt der Service einen durch Ihren KMS-Schlüssel geschützten Zweigschlüssel und speichert ihn in einem DynamoDB-based Amazon-Schlüsselspeicher.
Für jeden Verschlüsselungsvorgang leitet der hierarchische Schlüsselbund einen eindeutigen Wrapping-Schlüssel aus dem aktiven Branch-Schlüssel ab, der für jede Anfrage einen eindeutigen Datenverschlüsselungsschlüssel verschlüsselt.

Der hierarchische Schlüsselbund bietet die folgenden Vorteile:

- Per-resource Verschlüsselungsbereich — Jede Ressource der obersten Ebene hat ihren eigenen Zweigschlüssel, sodass Daten für verschiedene Ressourcen unter separaten Schlüsseln verschlüsselt werden.
- Automatische Schlüsselrotation — Der AWS Security Agent wechselt die Zweigschlüssel regelmäßig. Nach der Rotation werden neue Daten mit der neuen Version des Zweigschlüssels verschlüsselt, während ältere Versionen beibehalten werden, um zuvor verschlüsselte Daten zu entschlüsseln.

Verschlüsselungskontext

AWS Der Security Agent verwendet bei allen kryptografischen Vorgängen mit Ihrem KMS-Schlüssel den Verschlüsselungskontext. Der Verschlüsselungskontext besteht aus einer Reihe nicht geheimer Schlüssel-Wert-Paare, die zusätzliche authentifizierte Daten für den Verschlüsselungsvorgang bereitstellen.

Der Schlüssel für den Verschlüsselungskontext folgt dem Format`aws:securityagent:_<resource-type>_`, in dem der Typ der zu verschlüsselnden Ressource *`<resource-type>`* steht (z. B. oder). `agent-space` `integration` Der Wert ist der Amazon-Ressourcenname (ARN) der Ressource.

> **Note**
>
> Bei Integrationen enthält der Verschlüsselungskontextschlüssel das `aws-crypto-ec:` Präfix, aus dem sich das Format `aws-crypto-ec:aws:securityagent:integration` ergibt.

Sie können den Verschlüsselungskontext verwenden, um Ihre KMS-Schlüsselrichtlinie auf bestimmte Ressourcentypen zu beschränken. Weitere Informationen finden Sie unter the section called “Erforderliche KMS-Berechtigungen”.

Default encryption (Standardverschlüsselung)

Wenn Sie eine Ressource erstellen, ohne einen vom Kunden verwalteten Schlüssel anzugeben, verschlüsselt der AWS Security Agent die Ressource mit AWS verwalteten Verschlüsselungsschlüsseln, die von den zugrunde liegenden Speicherdiensten (Amazon DynamoDB und Amazon S3) bereitgestellt werden. Für die Standardverschlüsselung ist keine zusätzliche Konfiguration erforderlich.

Standard-KMS-Schlüssel für Anwendungen

Sie können einen Standard-KMS-Schlüssel auf Anwendungsebene festlegen. Wenn ein Standard-KMS-Schlüssel konfiguriert ist, verwendet der AWS Security Agent ihn als Fallback für neue Agent Spaces und Integrationen, wenn Sie bei der Ressourcenerstellung nicht explizit einen KMS-Schlüssel angeben.

Um einen Standard-KMS-Schlüssel festzulegen, geben Sie den `defaultKmsKeyId` Parameter an, wenn Sie Ihre Anwendung mit der AWS CLI oder dem SDK erstellen oder aktualisieren. Die Konsole fordert Sie bei der Einrichtung der Anwendung auf, einen Standard-KMS-Schlüssel anzugeben.

Wenn Sie bei der Ressourcenerstellung explizit einen KMS-Schlüssel angeben, hat dieser Vorrang vor der Standardeinstellung auf Anwendungsebene.

Voraussetzungen

Bevor Sie einen vom Kunden verwalteten Schlüssel konfigurieren, müssen Sie die folgenden Voraussetzungen erfüllen:

- Erstellen Sie einen KMS-Schlüssel für die symmetrische Verschlüsselung in AWS KMS. Der Schlüssel muss die folgenden Anforderungen erfüllen:
  - Schlüsseltyp: Symmetrisch
  - Schlüsselverwendung: Verschlüsseln und Entschlüsseln
  - Schlüsselspezifikation: `SYMMETRIC_DEFAULT`
  - Schlüsselstatus: Aktiviert

  Anweisungen finden Sie unter Schlüssel erstellen im AWS Key Management Service Developer Guide.
- Konfigurieren Sie die KMS-Schlüsselrichtlinie und die IAM-Richtlinien, um dem AWS Security Agent die erforderlichen Berechtigungen zu gewähren. Details hierzu finden Sie unter the section called “Erforderliche KMS-Berechtigungen”.

Erforderliche KMS-Berechtigungen

AWS Der Security Agent benötigt in zwei Szenarien KMS-Berechtigungen:

- Daten innerhalb des AWS Security Agents verschlüsseln — Wenn Sie einen vom Kunden verwalteten Schlüssel für einen Agent Space oder eine Integration angeben, benötigt der Service Berechtigungen, um diesen Schlüssel für kryptografische Operationen mit Ihren Daten zu verwenden.
- CMK-encrypted AWS Ressourcen verwenden — Wenn Sie AWS Ressourcen bereitstellen, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind (z. B. S3-Buckets, CloudWatch Log-Protokollgruppen oder Secrets Manager Manager-Geheimnisse), benötigt der Service Berechtigungen, um diese Schlüssel für den Zugriff auf die verschlüsselten Ressourcen zu verwenden.

AWS Der Security Agent greift je nach Vorgang mit unterschiedlichen Identitäten auf Ihren KMS-Schlüssel zu:

- AWS Operationen der Management-Konsole — Wenn Administratoren Ressourcen in der AWS Management-Konsole erstellen oder verwalten (z. B. einen Agent Space erstellen oder eine Anwendung aktualisieren), verwendet der Dienst die Administratorrolle, um KMS aufzurufen AWS .
- Webanwendungsvorgänge — Wenn IAM Identity Center-Benutzer über die AWS Security Agent-Webanwendung auf Daten zugreifen (z. B. um Penetrationstestergebnisse anzuzeigen oder einen Penetrationstest zu starten), verwendet der Dienst die während der Installation des AWS Security Agents erstellte Anwendungsrolle, um AWS KMS aufzurufen.
- Zugreifen auf vom Kunden bereitgestellte Ressourcen — Wenn der Service während eines Penetrationstests auf vom Kunden bereitgestellte AWS Ressourcen zugreift (z. B. S3-Buckets, CloudWatch Log-Protokollgruppen und Secrets Manager), verwendet er die Penetrationstest-Servicerolle, um KMS aufzurufen. AWS
- Asynchrone Workflows — Für Hintergrundvorgänge, die außerhalb einer Benutzersitzung ausgeführt werden (z. B. Ausführung von Penetrationstests, Verarbeitung von Codeüberprüfungen und Zweigschlüsselrotation), verwendet der Dienst seinen eigenen Dienstprinzipal (`securityagent.amazonaws.com`), um KMS in Ihrem Namen aufzurufen AWS .

In den folgenden Abschnitten werden die erforderlichen Berechtigungen für jede Identität beschrieben.

Schlüsselrichtlinie

Ihre KMS-Schlüsselrichtlinie muss dem AWS Security Agent die Erlaubnis erteilen, den Schlüssel für kryptografische Operationen zu verwenden. Die erforderlichen wichtigen Richtlinienanweisungen hängen davon ab, welche Ressourcentypen Sie verschlüsseln möchten und welche CMK-encrypted AWS Ressourcen Sie für den Dienst bereitstellen.

Wichtige Richtlinie für Agent Spaces

Die folgende wichtige Richtlinie gewährt dem AWS Security Agent die erforderlichen Berechtigungen zum Verschlüsseln und Entschlüsseln von Agent Space-Daten, einschließlich Penetrationstestergebnissen und Screenshots.

Ersetzen Sie die folgenden Platzhalterwerte in der Richtlinie:

- *`111122223333`* — Ihre AWS Konto-ID
- *`MyRole`* — Die IAM-Rolle, mit der Sie den AWS Security Agent in der Konsole verwalten
- *`MyApplicationRole`* — Die Anwendungsrolle, die während der Installation des AWS Security Agents erstellt wurde

- *us-east-1* — Die AWS Region, in der Sie den AWS Security Agent verwenden

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKeyMetadataValidationForApplicationAndAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowUseOfHierarchicalKeyringForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
 "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
```

```
    },
    {
      "Sid": "AllowSynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyApplicationRole"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
"arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
"arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        }
      }
```

```
    },
    {
      "Sid": "AllowAsynchronousDataAccessForCodeRemediation",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*"
    }
  ]
}
```

> **Important**
>
> Um Zweigschlüssel rotieren zu können, muss Ihre KMS-Schlüsselrichtlinie dem AWS Security Agent Service Principal (`securityagent.amazonaws.com`) `kms:ReEncryptFrom` Berechtigungen und gewährenkms:GenerateDataKeyWithoutPlaintext. Andernfalls schlägt die Rotation von Zweigschlüsseln fehl. `kms:ReEncryptTo` Weitere Informationen zu den erforderlichen Berechtigungen finden Sie unter Rotation eines Zweigschlüssels.

Wichtige Richtlinie für Integrationen

Die folgende wichtige Richtlinie gewährt dem AWS Security Agent die erforderlichen Berechtigungen zum Verschlüsseln und Entschlüsseln von Integrationsdaten, einschließlich der Ergebnisse der Codeüberprüfung.

Ersetzen Sie die folgenden Platzhalterwerte in der Richtlinie:

- *111122223333* — Ihre AWS Konto-ID
- *MyRole* — Die IAM-Rolle, mit der Sie den AWS Security Agent in der Konsole verwalten
- *us-east-1* — Die AWS Region, in der Sie den AWS Security Agent verwenden

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```
    {
      "Sid": "AllowKeyAccessValidationForIntegrations",
      "Effect": "Allow",
      "Principal": {
          "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws-crypto-ec:aws:securityagent:integration":
"arn:aws:securityagent:us-east-1:111122223333:integration/*"
        },
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowKeyMetadataValidationForIntegrations",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "AllowAsynchronousDataAccessForIntegrations",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:ReEncryptTo",
```

```
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        },
        "StringLike": {
          "kms:EncryptionContext:aws-crypto-ec:aws:securityagent:integration":
 "arn:aws:securityagent:us-east-1:111122223333:integration/*"
        }
      }
    }
  ]
}
```

> **Note**
>
> Wenn Sie denselben KMS-Schlüssel für mehrere Ressourcentypen verwenden möchten, können Sie die wichtigsten Richtlinienerklärungen des Agent Space, der Integration und des Security Requirement Pack zu einer einzigen Schlüsselrichtlinie zusammenfassen.

Wichtige Richtlinie für Sicherheitsanforderungspakete

Die folgende wichtige Richtlinie gewährt dem AWS Security Agent die erforderlichen Berechtigungen zum Verschlüsseln und Entschlüsseln von Security Requirement Pack-Daten. Sicherheitsanforderungspakete verwenden keine Webanwendungsrolle. Die Richtlinie verwendet zwei Zugriffspfade:

- Synchroner Pfad — Wenn Sie die API aufrufen, verwendet der Dienst Ihre weitergeleiteten Anmeldeinformationen, um AWS KMS in Ihrem Namen aufzurufen (über die `kms:ViaService` Bedingung).
- Asynchroner Pfad — Bei asynchronen Vorgängen, bei denen die Pakete verwendet werden können, verwendet der Dienst seinen eigenen Dienstprinzipal, um KMS direkt aufzurufen AWS .

Ersetzen Sie die folgenden Werte in der Richtlinie:

- *111122223333* — Ihre AWS Konto-ID

- *MyRole* — Die IAM-Rolle, die Sie zum Aufrufen von Security Requirement Pack-Vorgängen verwenden
- *us-east-1* — Die AWS Region, in der Sie den Security Agent verwenden AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKeyMetadataValidation",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowUseOfHierarchicalKeyringForSecurityPacks",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyRole"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        },
        "StringLike": {
```

```
          "kms:EncryptionContext:aws:securityagent:security-requirement-pack":
 "arn:aws:securityagent:us-east-1:111122223333:security-requirement-pack/*"
        }
      }
    },
    {
      "Sid": "AllowAsynchronousDataAccessForSecurityPacks",
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt",
        "kms:ReEncryptTo",
        "kms:ReEncryptFrom"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:security-requirement-pack":
 "arn:aws:securityagent:us-east-1:111122223333:security-requirement-pack/*"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:securityagent:us-east-1:111122223333:security-
requirement-pack/*"
        }
      }
    }
  ]
}
```

Die Richtlinie enthält drei Aussagen:

- AllowKeyMetadataValidation— Genehmigungen, `kms:DescribeKey` damit der AWS Security Agent überprüfen kann, ob Ihr vom Kunden verwalteter Schlüssel existiert, aktiviert ist und symmetrische Verschlüsselung verwendet, wenn Sie bei der Ressourcenerstellung einen CMK angeben. Verwendet die `kms:ViaService` Bedingung, um sicherzustellen, dass der Anruf über den Service erfolgt.
- AllowUseOfHierarchicalKeyringForSecurityPacks— Erteilt `kms:GenerateDataKeyWithoutPlaintext` (neue Zweigschlüssel erstellen) `kms:ReEncryptTo` und `kms:ReEncryptFrom` (bestehende Zweigschlüssel rotieren) und

`kms:Decrypt` (bestehende Zweigschlüssel abrufen) für hierarchische Schlüsselbundoperationen. Diese Operationen verwenden Ihre weitergeleiteten Anmeldeinformationen über den synchronen Pfad und sind aufgrund der Verschlüsselungskontextbedingung auf die Ressourcen des Sicherheitsanforderungspakets beschränkt.

- AllowAsynchronousDataAccessForSecurityPacks— Gewährt dem AWS Security Agent Service Principal `kms:GenerateDataKeyWithoutPlaintext` `kms:Decrypt`kms:ReEncryptTo,, und `kms:ReEncryptFrom` für asynchrone Operationen, wenn keine Anruferanmeldedaten verfügbar sind.

Im Gegensatz zur Schlüsselrichtlinie von Agent Spaces beinhaltet die Richtlinie für das Security Requirement Pack keinen Prinzipal für die Rolle einer Webanwendung. Der Zugriff auf die Security Requirement Packs erfolgt über die AWS Management-Konsole mit Ihrer Administratorrolle, nicht über die AWS Security Agent-Webanwendung. Alle synchronen Operationen verwenden die einzige Anruferrolle (via`kms:ViaService`).

> **Note**
>
> Wenn Sie denselben KMS-Schlüssel sowohl für Agent Spaces als auch für Security Requirement Packs verwenden, können Sie die wichtigsten Richtlinienangaben aus beiden Abschnitten zu einer einzigen Schlüsselrichtlinie zusammenfassen.

Wichtige Richtlinie für CMK-encrypted AWS Ressourcen

Wenn Sie dem AWS Security Agent AWS Ressourcen zur Verfügung stellen, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind, müssen Sie die Schlüsselrichtlinie für den KMS-Schlüssel jeder Ressource aktualisieren, um die erforderlichen Berechtigungen zu gewähren. Dies gilt für die folgenden Ressourcen:

- S3-Buckets — Wenn Sie Lernressourcen aus einem S3-Bucket bereitstellen, der mit einem vom Kunden verwalteten Schlüssel (SSE-KMS) verschlüsselt ist, muss die Schlüsselrichtlinie es der Rolle des Penetrationstest-Dienstes ermöglichen, Objekte zu entschlüsseln.
- Secrets Manager Manager-Geheimnisse — Wenn Ihre Penetrationstest-Anmeldeinformationen in Secrets Manager Manager-Geheimnissen gespeichert sind, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind, muss die Schlüsselrichtlinie es der Penetrationstest-Servicerolle ermöglichen, diese Geheimnisse zu entschlüsseln. Wenn die Webanwendung in Ihrem Namen

Geheimnisse erstellt, muss die Schlüsselrichtlinie es der Anwendungsrolle auch ermöglichen, diese Geheimnisse zu verschlüsseln.

- CloudWatch Log-Protokollgruppen — Wenn die CloudWatch Logs-Protokollgruppe, die zum Speichern von Ausführungsprotokollen für Penetrationstests verwendet wird, mit einem vom Kunden verwalteten Schlüssel verschlüsselt ist, muss die Schlüsselrichtlinie CloudWatch Logs ermöglichen, den KMS-Schlüssel über die Service-Rolle zu validieren und dem CloudWatch Logs-Dienstprinzipal die Ausführung kryptografischer Operationen zu ermöglichen.

> **Important**
>
> AWS Security Agent kann in Ihrem Namen auch CloudWatch Logs-Protokollgruppen und Secrets Manager-Geheimnisse erstellen. Fügen Sie bei der Konfiguration Ihrer KMS-Schlüsselrichtlinie auch die entsprechenden Anweisungen für diese vom Dienst erstellten Ressourcen hinzu.

Die folgenden wichtigen Richtlinienerklärungen gewähren die erforderlichen Berechtigungen für CMK-encrypted Ressourcen. Schließen Sie nur die Aussagen ein, die für Ihre Konfiguration gelten. Wenn eine Ressource denselben KMS-Schlüssel verwendet, den Sie für Ihren Agent Space angegeben haben, fügen Sie diese Anweisungen der Richtlinie für diesen Schlüssel hinzu. Wenn eine Ressource einen anderen KMS-Schlüssel verwendet, fügen Sie diese Anweisungen stattdessen der Richtlinie dieses Schlüssels hinzu.

Ersetzen Sie die folgenden Platzhalterwerte in der Richtlinie:

- *`111122223333`* — Ihre AWS Konto-ID
- *`MyApplicationRole`* — Die Anwendungsrolle, die während der Installation des AWS Security Agents erstellt wurde
- *`MyPenTestServiceRole`* — Die Rolle des Penetrationstest-Dienstes
- *`us-east-1`* — Die AWS Region, in der Sie den AWS Security Agent verwenden

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKmsKeyAccessForCreatingSecrets",
```

```
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyApplicationRole"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyDecryptionForS3Objects",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "s3.*.amazonaws.com",
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::YOUR-BUCKET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyDecryptionForSecretsManagerSecrets",
      "Effect": "Allow",
      "Principal": {
```

```
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:YOUR-SECRET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyValidationForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyPenTestServiceRole"
      },
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "logs.*.amazonaws.com"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyAccessForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Principal": {
        "Service": "logs.us-east-1.amazonaws.com"
      },
      "Action": [
        "kms:Encrypt",
```

```
        "kms:Decrypt",
        "kms:GenerateDataKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:logs:arn": "arn:aws:logs:us-
east-1:111122223333:log-group:YOUR-LOG-GROUP-NAME*"
        }
      }
    }
  ]
}
```

Note

- Die `AllowKmsKeyDecryptionForS3Objects` Erklärung ist nur erforderlich, wenn Sie Lernressourcen aus S3-Buckets bereitstellen, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind. Weitere Informationen zur Konfiguration SSE-KMS für S3 finden Sie unter Daten schützen mit SSE-KMS.
- Die `AllowKmsKeyDecryptionForSecretsManagerSecrets` Erklärung ist nur erforderlich, wenn Ihre Anmeldeinformationen für den Penetrationstest in Secrets Manager gespeichert sind, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind. Die Servicerolle benötigt Zugriff, um Geheimnisse während des Penetrationstests zu entschlüsseln. Weitere Informationen zur geheimen Verschlüsselung finden Sie unter Geheime Verschlüsselung und Entschlüsselung in Secrets Manager.
- `AllowKmsKeyValidationForCloudWatchLogsLogGroups`In der Anweisung wird die Dienstrolle `kms:DescribeKey` zugewiesen, sodass CloudWatch Logs den KMS-Schlüssel anhand der Dienstrolle validieren kann, wenn er einer Protokollgruppe zugeordnet wird.
- Die `AllowKmsKeyAccessForCreatingSecrets` Anweisung ist erforderlich, wenn die Webanwendung Secrets Manager in Ihrem Namen mithilfe eines vom Kunden verwalteten Schlüssels erstellt. Die Anwendungsrolle benötigt `kms:GenerateDataKey` und `kms:Decrypt` muss das Geheimnis mit Ihrem KMS-Schlüssel verschlüsseln.
- Die `AllowKmsKeyAccessForCloudWatchLogsLogGroups` Anweisung gewährt dem CloudWatch Logs-Dienstprinzipal (`logs.`*`us-east-1`*`.amazonaws.com`) die

Berechtigungen, die er zum Verschlüsseln und Entschlüsseln von Protokolldaten benötigt. Diese Anweisung ist auch erforderlich, wenn der AWS Security Agent in Ihrem Namen eine Protokollgruppe erstellt, obwohl Sie keine vorhandene angeben. Weitere Informationen finden Sie unter Verschlüsseln von Protokolldaten in CloudWatch Logs using AWS KMS.

- Wenn mehrere Ressourcen denselben KMS-Schlüssel verwenden, können Sie die Anweisungen zu einer einzigen Schlüsselrichtlinie zusammenfassen.

Rolle des Administrators

Für die Administratorrolle (die IAM-Rolle, mit der Sie den AWS Security Agent in der Konsole verwalten) ist keine zusätzliche IAM-Richtlinie erforderlich.

Rolle „Anwendung“

Fügen Sie der bei der Installation des AWS Security Agents angegebenen Anwendungsrolle zusätzlich zur KMS-Schlüsselrichtlinie die folgende IAM-Richtlinie hinzu. Diese Richtlinie gewährt der Rolle Berechtigungen zur Verwendung Ihrer vom Kunden verwalteten Schlüssel zum Verschlüsseln und Entschlüsseln von Agent Space-Daten und zum Erstellen von Geheimnissen in AWS Secrets Manager.

Ersetzen Sie die folgenden Platzhalterwerte in der Richtlinie:

- *111122223333* — Ihre AWS Konto-ID
- *us-east-1* — Die AWS Region, in der Sie AWS Security Agent verwenden
- Der KMS-Schlüssel ARNs in `Resource` — Die ARNs Ihrer KMS-Schlüssel

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSynchronousDataAccessForAgentSpaces",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
```

```
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890cd"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333",
          "kms:ViaService": "securityagent.us-east-1.amazonaws.com"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:securityagent:agent-space":
 "arn:aws:securityagent:us-east-1:111122223333:agent-space/*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyAccessForCreatingSecrets",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "arn:aws:kms:us-east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890cd"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:*"
        }
      }
    }
  ]
}
```

> **Note**
>
> Die Anwendungsrolle ist eine IAM-Rolle, die Sie angeben oder die die Konsole während der Installation des AWS Security Agents erstellt. Die Webanwendung übernimmt diese Rolle,

um Ausführungsprotokolle für Penetrationstests abzurufen und Secrets Manager Manager-Geheimnisse in Ihrem Namen zu erstellen.

- Die `AllowKmsKeyAccessForCreatingSecrets` Anweisung ist erforderlich, wenn Sie Authentifizierungsressourcen für Penetrationstests konfigurieren und Anmeldeinformationen direkt eingeben möchten, anstatt ein vorhandenes Geheimnis anzugeben. Die Webanwendung erstellt in Ihrem Namen ein Geheimnis, und die Anwendungsrolle benötigt die in dieser Anweisung angegebenen Berechtigungen, um das Geheimnis mit dem vom Kunden verwalteten Schlüssel zu verschlüsseln, der für Ihren Agentbereich angegeben ist. Aktualisieren Sie die `Resource` ARNs in dieser Erklärung so, dass sie mit dem KMS-Schlüssel übereinstimmen, der für Ihren Agent Space verwendet wird.

Service – -Rolle

Während des Penetrationstests übernimmt der AWS Security Agent die Rolle des Penetrationstest-Dienstes für den Zugriff auf Ihre AWS Ressourcen. Wenn eine dieser Ressourcen mit einem vom Kunden verwalteten Schlüssel verschlüsselt ist, müssen Sie der Servicerolle Berechtigungen zur Verwendung der entsprechenden KMS-Schlüssel erteilen. Dies gilt für die folgenden Ressourcen:

- S3-Buckets — Wenn Sie Lernressourcen (wie API-Dokumente, Bedrohungsmodelle oder Quellcode) aus einem mit einem vom Kunden verwalteten Schlüssel verschlüsselten S3-Bucket bereitstellen, benötigt die Servicerolle Berechtigungen zum Entschlüsseln von Objekten in diesem Bucket. Weitere Informationen zur Konfiguration SSE-KMS für S3 finden Sie unter Daten schützen mit. SSE-KMS
- Secrets Manager Manager-Geheimnisse — Wenn Ihre Anmeldeinformationen für Penetrationstests in Secrets Manager Manager-Geheimnissen gespeichert sind, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind, benötigt die Servicerolle Berechtigungen, um diese Geheimnisse zu entschlüsseln. Weitere Informationen zur geheimen Verschlüsselung finden Sie unter Geheime Verschlüsselung und Entschlüsselung in Secrets Manager.
- CloudWatch Log-Protokollgruppen — Wenn die Protokollgruppe CloudWatch Logs, die zum Speichern von Protokollen zur Ausführung von Penetrationstests verwendet wird, mit einem vom Kunden verwalteten Schlüssel verschlüsselt ist (einschließlich Protokollgruppen, die vom AWS Security Agent in Ihrem Namen erstellt wurden), benötigt die Service-Rolle die `kms:DescribeKey` Erlaubnis, den Schlüssel zu validieren. Der CloudWatch Logs-Dienstprinzipal kümmert sich um die eigentliche Verschlüsselung und Entschlüsselung der Protokolldaten. Diese Berechtigungen

werden über die Schlüsselrichtlinie gewährt (siehethe section called “Schlüsselrichtlinie”). Weitere Informationen zum Verschlüsseln von Protokolldaten finden Sie unter Verschlüsseln von Protokolldaten in CloudWatch Logs using KMS. AWS

> **Important**
>
> Wenn Sie für die Verschlüsselung dieser Ressourcen einen anderen KMS-Schlüssel verwenden als den, den Sie für Ihren Agent Space angegeben haben, müssen Sie der Servicerolle Berechtigungen für jeden KMS-Schlüssel gewähren, der eine Ressource schützt, auf die die Servicerolle bei Penetrationstests zugreift.

Fügen Sie der Penetrationstest-Servicerolle die folgende IAM-Richtlinie hinzu. Geben Sie nur die Aussagen an, die für Ihre Konfiguration gelten.

Ersetzen Sie die folgenden Platzhalterwerte in der Richtlinie:

- *111122223333* — Ihre AWS Konto-ID
- *us-east-1* — Die AWS Region, in der Sie AWS Security Agent verwenden
- Der KMS-Schlüssel ARNs in `Resource` — Die ARNs der vom Kunden verwalteten Schlüssel, die zur Verschlüsselung der einzelnen Ressourcen verwendet wurden

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowKmsAccessForEncryptedS3Buckets",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-S3-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
```

```
          "kms:ViaService": "s3.*.amazonaws.com",
          "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::YOUR-BUCKET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsAccessForEncryptedSecrets",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-SECRETS-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "secretsmanager.*.amazonaws.com",
          "kms:EncryptionContext:SecretARN": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:YOUR-SECRET-NAME*"
        }
      }
    },
    {
      "Sid": "AllowKmsKeyValidationForCloudWatchLogsLogGroups",
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/EXAMPLE-CLOUDWATCH-LOGS-KEY-ID"
      ],
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:ViaService": "logs.*.amazonaws.com"
        }
      }
    }
  ]
```

```
}
```

> **Note**
>
> - Die `AllowKmsAccessForEncryptedS3Buckets` Erklärung ist nur erforderlich, wenn Sie Lernressourcen aus S3-Buckets bereitstellen, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind. Aktualisieren `Resource` Sie den ARN so, dass er mit dem KMS-Schlüssel übereinstimmt, der zur Verschlüsselung Ihres S3-Buckets verwendet wurde, und aktualisieren Sie den `kms:EncryptionContext:aws:s3:arn` Wert so, dass er Ihrem Bucket-Namen entspricht.
> - Die `AllowKmsAccessForEncryptedSecrets` Erklärung ist nur erforderlich, wenn Ihre Anmeldeinformationen für den Penetrationstest in Secrets Manager gespeichert sind, die mit einem vom Kunden verwalteten Schlüssel verschlüsselt sind. Aktualisieren `Resource` Sie den ARN so, dass er mit dem KMS-Schlüssel übereinstimmt, der zum Verschlüsseln Ihrer Geheimnisse verwendet wurde, und aktualisieren Sie den `kms:EncryptionContext:SecretARN` Wert so, dass er Ihrem geheimen Namen entspricht.
> - Die `AllowKmsKeyValidationForCloudWatchLogsLogGroups` Erklärung ist nur erforderlich, wenn Ihre CloudWatch Logs-Protokollgruppe mit einem vom Kunden verwalteten Schlüssel verschlüsselt ist, einschließlich Protokollgruppen, die vom AWS Security Agent in Ihrem Namen erstellt wurden. Aktualisieren Sie den `Resource` ARN so, dass er mit dem KMS-Schlüssel übereinstimmt, der zum Verschlüsseln Ihrer Protokollgruppe verwendet wurde.
> - Wenn mehrere Ressourcen denselben KMS-Schlüssel verwenden, können Sie die Anweisungen kombinieren und den ARN für den gemeinsamen Schlüssel einmal im `Resource` Feld auflisten.

Erstellen Sie eine Ressource mit einem vom Kunden verwalteten Schlüssel

Sie können bei der Einrichtung des AWS Security Agents oder bei der Erstellung einzelner Ressourcen einen vom Kunden verwalteten Schlüssel angeben. Der KMS-Schlüssel verschlüsselt alle Daten, die zu dieser Ressource und ihren Unterressourcen gehören.

Legen Sie während der Installation einen Standard-KMS-Schlüssel fest (Konsole)

Sie können bei der Erstinstallation des AWS Security Agents einen Standard-KMS-Schlüssel konfigurieren. Dieser Schlüssel wird standardmäßig für neue Agent Spaces und Integrationen verwendet, sofern Sie keinen anderen Schlüssel angeben.

1. Erweitern Sie auf der Seite AWS Security Agent einrichten den Abschnitt Verschlüsselung — optional.
2. Wählen Sie Verschlüsselungseinstellungen anpassen (erweitert) aus.
3. Wählen Sie unter Wählen Sie einen AWS-KMS-Schlüssel aus Ihrem Konto aus, oder geben Sie einen KMS-Schlüssel-ARN ein.
4. Schließen Sie die verbleibenden Einrichtungsschritte ab und wählen Sie Set up AWS Security Agent.

AWS Der Security Agent validiert den KMS-Schlüssel, um zu bestätigen, dass er existiert, aktiviert ist und symmetrische Verschlüsselung verwendet. Schlägt die Überprüfung fehl, wird die Installation nicht fortgesetzt und Sie erhalten eine Fehlermeldung.

Erstellen Sie einen Agentenbereich mit einem vom Kunden verwalteten Schlüssel (Konsole)

1. Navigieren Sie in der AWS Security Agent-Konsole zur Seite Agent Spaces.
2. Wählen Sie Create Agent Space.
3. Geben Sie einen Namen und optional eine Beschreibung für Ihren Agent Space ein.
4. Erweitern Sie den Abschnitt Advanced (Erweitert).
5. Wählen Sie unter Datenverschlüsselung eine der folgenden Optionen aus:
   - Standardschlüssel der Anwendung verwenden — Verwendet den Standard-KMS-Schlüssel, der während der Installation des AWS Security Agents konfiguriert wurde. Diese Option ist standardmäßig ausgewählt, wenn ein Standardschlüssel konfiguriert ist.
   - Einen anderen Schlüssel verwenden — Geben Sie einen anderen KMS-Schlüssel für diesen Agent Space an. Wählen Sie Verschlüsselungseinstellungen anpassen (erweitert) und wählen Sie dann für Wählen Sie einen AWS-KMS-Schlüssel aus Ihrem Konto aus oder geben Sie einen ARN ein.
6. Wählen Sie Erstellen aus.

Erstellen Sie eine Integration mit einem vom Kunden verwalteten Schlüssel (Konsole)

1. Navigieren Sie in der AWS Security Agent-Konsole zur Seite Integrationen.
2. Wählen Sie Integration hinzufügen und wählen Sie Ihren Anbieter aus (z. B. GitHub).
3. Schließen Sie Schritt 1 ab: Installieren und autorisieren Sie die Anbieteranwendung.
4. Geben Sie in Schritt 2: Registrierungsdetails einen Registrierungsnamen ein und konfigurieren Sie die Anbietereinstellungen.
5. Wählen Sie im Abschnitt Datenverschlüsselung die Option Verschlüsselungseinstellungen anpassen (erweitert) aus.
6. Wählen Sie unter Wählen Sie einen AWS-KMS-Schlüssel aus Ihrem Konto aus, oder geben Sie einen KMS-Schlüssel-ARN ein.
7. Wählen Sie Connect aus.

Erstellen Sie eine Ressource mit einem vom Kunden verwalteten Schlüssel (AWS CLI oder SDK)

So geben Sie einen vom Kunden verwalteten Schlüssel mithilfe der AWS CLI oder des SDK an:

- Geben Sie den `defaultKmsKeyId` Parameter an, wenn Sie aufrufen`CreateApplication`, um einen Standard-KMS-Schlüssel für Ihre Anwendung festzulegen. Dieser Schlüssel wird als Fallback verwendet, wenn Agent Spaces oder Integrationen ohne expliziten KMS-Schlüssel erstellt werden.
- Schließen Sie den `kmsKeyId` Parameter ein, wenn Sie eine `CreateAgentSpace` bestimmte Ressource aufrufen oder verschlüsseln `CreateIntegration` möchten. Dies hat Vorrang vor der Standardeinstellung auf Anwendungsebene.

Einzelheiten zu den Parametern finden Sie in der AWS Security Agent API-Referenz.

Wenn Sie keinen KMS-Schlüssel angeben, verwendet der AWS Security Agent den Standard-KMS-Schlüssel auf Anwendungsebene, sofern einer konfiguriert ist. Andernfalls wird die Ressource mit AWS-verwalteten Schlüsseln verschlüsselt.

Schlüsselrotation

AWS Der Security Agent wechselt die Zweigschlüssel automatisch in regelmäßigen Abständen. Durch die Rotation von Zweigschlüsseln wird eine neue aktive Version des Zweigschlüssels erstellt, wobei alle vorherigen Versionen beibehalten werden. Nach der Rotation:

- Neue Daten werden mit der neuen Version des Zweigschlüssels verschlüsselt.

- Zuvor verschlüsselte Daten können auch mit der älteren Version des Zweigschlüssels entschlüsselt werden.
- Eine erneute Verschlüsselung der Daten ist nicht erforderlich.

Die Zweigschlüsselrotation ist von der KMS-Schlüsselrotation getrennt. Sie können die automatische KMS-Schlüsselrotation für Ihren vom Kunden verwalteten Schlüssel unabhängig aktivieren. Weitere Informationen finden Sie unter Rotation von KMS-Schlüsseln im AWS Key Management Service Developer Guide.

Nach der Rotation von Zweigschlüsseln gibt es einen kurzen Zeitraum, in dem zwischengespeicherte Kopien des vorherigen Zweigschlüssels weiterhin für neue Verschlüsselungsvorgänge verwendet werden können. Dieses Fenster wird durch die Gültigkeitsdauer (TTL) des internen Caches bestimmt und hat keinen Einfluss auf die Sicherheit Ihrer Daten.

Überlegungen

Beachten Sie Folgendes, wenn Sie vom Kunden verwaltete Schlüssel mit dem AWS Security Agent verwenden:

- Vom Kunden verwaltete Schlüssel gelten nur für neue Ressourcen. Bestehende Ressourcen, die vor der Konfiguration eines vom Kunden verwalteten Schlüssels erstellt wurden, verwenden weiterhin die vom AWS Kunden verwaltete Verschlüsselung. Es findet keine Migration vorhandener Daten statt.
- Vom Kunden verwaltete Schlüssel werden bei der Erstellung angegeben. Sie geben den KMS-Schlüssel an, wenn Sie eine Ressource erstellen. Sie können den KMS-Schlüssel für eine vorhandene Ressource nicht hinzufügen oder ändern.
- Es werden mehrere KMS-Schlüssel unterstützt. Sie können verschiedene KMS-Schlüssel für verschiedene Ressourcen verwenden. Sie können beispielsweise einen Schlüssel für die Produktion von Agent Spaces und einen anderen Schlüssel für die Entwicklung von Agent Spaces verwenden.
- Durch das Deaktivieren oder Löschen eines KMS-Schlüssels kann auf Daten nicht zugegriffen werden. Wenn Sie einen KMS-Schlüssel deaktivieren oder das Löschen eines KMS-Schlüssels planen, kann der AWS Security Agent die mit diesem Schlüssel verschlüsselten Daten nicht entschlüsseln. Auf die betroffenen Ressourcen kann erst zugegriffen werden, wenn der Schlüssel wieder aktiviert wird. Durch das Löschen eines KMS-Schlüssels wird der Zugriff auf alle mit diesem Schlüssel verschlüsselten Daten dauerhaft verhindert.

- Wichtige Richtlinienberechtigungen sind erforderlich. Wenn Sie die erforderlichen Berechtigungen aus Ihrer KMS-Schlüsselrichtlinie entfernen, kann der AWS Security Agent nicht auf verschlüsselte Daten zugreifen. Stellen Sie sicher, dass die Schlüsselrichtlinie Berechtigungen für die Administratorrolle (zur Verwaltung des Dienstes in der AWS Konsole), die Anwendungsrolle (die von der Webanwendung verwendet wird), die Penetrationstest-Servicerolle (die von den Agenten zur Durchführung von Penetrationstests übernommen wird) und den AWS Security Agent Service Principal gewährt, wie unter beschriebenthe section called “Erforderliche KMS-Berechtigungen”.
- AWS Es gelten KMS-Kontingente. Kryptografische Operationen mit Ihrem KMS-Schlüssel werden auf Ihre KMS-Anforderungskontingente angerechnet. AWS Bei normaler Verwendung minimiert die hierarchische Schlüsselbundarchitektur KMS-Aufrufe durch Zwischenspeicherung von Zweigschlüsseln. Weitere Informationen finden Sie unter Kontingente anfordern im AWS Key Management Service Developer Guide.

Fehlerbehebung

Finden Sie Lösungen für häufig auftretende Fehler bei der Verwendung von AWS Security Agent.

Zugriff verweigert: Falscher GitHub Kontotyp ausgewählt oder falscher Organisationsname angegeben

- Sie haben die AWS Security Agent-Anwendung in der gewünschten GitHub Organisation installiert, aber fälschlicherweise `User` anstelle von `GitHub Account Type` `Organization`
- Sie haben die AWS Security Agent-Anwendung in der gewünschten GitHub Organisation installiert und den Wert korrekt `GitHub Account Type` auf gesetzt, das `Organization Name` Feld `Organization` jedoch leer gelassen oder einen falschen Organisationsnamen eingegeben, der nicht mit der Organisation übereinstimmt, in der Sie die Anwendung installiert haben.

Lösung:

1. Gehen Sie zu der App GitHub und deinstallieren Sie sie von der Organisation. Weitere Informationen finden Sie unter the section called “Schritt 1: Deinstallieren Sie die AWS Security Agent GitHub App von GitHub”.
2. Gehen Sie zurück zur Integrationsseite und starten Sie den Integrationsprozess neu, indem Sie auf die App klickenAdd `Integrations`, sie installieren und erneut für die gewünschte GitHub Organisation autorisieren.

3. Wählen Sie `Organization` aus der Drop-down-Liste von `GitHub account type`
4. Stellen `Organization Name` Sie sicher, dass Ihre Eingabe EXAKT mit der übereinstimmt, in der Sie die Anwendung installiert haben.
5. Wählen Sie Connect, um Ihre GitHub Organisationsintegration zu erstellen.

Zugriff verweigert: Unzureichende Berechtigungen, um die GitHub App in der Organisation zu installieren

Wenn Sie versuchen, die AWS Security Agent-Anwendung in der gewünschten GitHub Organisation zu installieren, werden auf der Schaltfläche auf der Installationsseite zwei verschiedene Meldungen angezeigt.

- Eine Organisation `Member` wird Folgendes sehen `Authorize & Request`
- Eine Organisation `Owner` wird es sehen `Install & Authorize`

Sie können anhand der folgenden Schritte überprüfen, ob Sie `Owner` Teil der GitHub Organisation sind `Member` oder der Organisation angehören.

1. Gehe zu github.com
2. Wähle dein Profil auf der Website
3. Navigieren Sie `Organizations` zum Drop-down-Menü und wählen Sie es aus
4. Suchen Sie in der Liste der Organisationen nach der Organisation, in der Sie den AWS Security Agent installieren möchten. Dort wird `Owner` neben dem Namen der Organisation angegeben, ob Sie ein `Member` oder ein sind.

Mögliche Lösungen:

- Lassen Sie Ihre Installationsanfrage von einem Eigentümer genehmigen, BEVOR Sie versuchen, die Integration zu erstellen
- Bitten Sie einen Eigentümer, Ihre Rolle in der GitHub Organisation von a `Member` auf an zu aktualisieren `Owner` und den Integrationsprozess erneut zu starten

Der Agent kann während eines Penetrationstests keine Verbindung zum Endpunkt herstellen

Wenn der Penetrationstest-Agent keine Aufrufe an die konfigurierte Ziel-URL tätigen kann oder den Zielendpunkt nicht erfolgreich durchsucht:

- Wenn Ihr Endpunkt Domains außerhalb der konfigurierten Ziel-URL aufruft, überprüfen Sie, ob die zusätzlichen Domänen in Ihrer Pentest-Konfiguration als Accessible URLs hinzugefügt wurden
- Penetrationstests sind derzeit nur für HTTP/HTTPS Endgeräte verfügbar, die Datenverkehr über die Ports 80 oder 443 bedienen
- Wenn Sie eine WAF konfiguriert haben, stellen Sie sicher, dass Ihre WAF den Verkehr mit Penetrationstests nicht blockiert. Sie können Penetrationstest-Traffic nach `User-Agent` Header zulassen, wobei dieser Header standardmäßig auf gesetzt ist `securityagent`

Weitere Hilfe

Falls nach dem Ausführen dieser Schritte zur Fehlerbehebung weiterhin Probleme auftreten:

- Überprüfen Sie die Servicestatusseite von AWS Security Agent
- Wenden Sie sich an den AWS-Support, wenn Sie Hilfe bei komplexen Konfigurationsproblemen benötigen

Für Anwender und Entwickler (Web-App und IDE)

Führen Sie Designprüfungen, Bedrohungsmodelle, Codeprüfungen und Penetrationstests in der Webanwendung, einer IDE oder einem verbundenen Quellenanbieter durch und überprüfen und korrigieren Sie anschließend die Ergebnisse.

Melden Sie sich bei der AWS Security Agent Web App an

Sie sollten sich bei der AWS Security Agent Web App anmelden, um Aufgaben wie die folgenden zu erledigen:

- Führen Sie eine Entwurfsprüfung durch
- Führen Sie einen Penetrationstest durch

Zugriffsmethoden

AWS Security Agent bietet verschiedene Methoden für den Zugriff auf die Webanwendung, je nachdem, wie Ihr Unternehmen den Zugriff bei der Ersteinrichtung konfiguriert hat und ob Sie Zugriff auf die AWS-Konsole haben.

IAM Identity Center (SSO) — Wenn Ihre Organisation IAM Identity Center aktiviert hat, können Sie sich mit Ihren SSO-Anmeldeinformationen anmelden. Sie müssen mindestens einem Agent Space in IAM Identity Center zugewiesen sein, bevor Sie auf die Webanwendung zugreifen können.

Administratorzugriff (IAM) — Wenn Sie Zugriff auf die AWS-Konsole mit den entsprechenden Berechtigungen haben, können Sie die Webanwendung über den Administratorzugriffslink auf einer beliebigen Agent Space-Seite starten. Diese Methode ermöglicht die Authentifizierung über Ihre bestehende Konsolensitzung.

> **Note**
>
> Die primäre Zugriffsmethode, die Ihr Unternehmen verwendet, wurde bei der ersten Einrichtung des AWS Security Agents festgelegt. Informationen zur Konfiguration des Benutzerzugriffs und der Benutzerzuweisungen finden Sie unterthe section called “Benutzern Zugriff auf die AWS Security Agent-Web-App gewähren”.

Melden Sie sich mit IAM Identity Center (SSO) an

Wenn Ihr Unternehmen den IAM Identity Center-Zugriff konfiguriert hat, können Sie sich mit Ihren SSO-Anmeldeinformationen über mehrere Zugangspunkte bei der Webanwendung anmelden.

Voraussetzungen

- Ihnen wurde mindestens ein Agent Space im IAM Identity Center zugewiesen
- Sie haben Ihre IAM Identity Center SSO-Anmeldeinformationen

Greifen Sie auf die Webanwendung zu

Wählen Sie je nach Bedarf eine der folgenden Methoden:

Um alle Agent Spaces anzuzeigen, denen Sie zugewiesen sind:

1. Navigieren Sie in der AWS Security Agent-Konsole zu Einstellungen.
2. Suchen Sie die Webanwendungsdomäne und kopieren Sie die URL.

> **Tip**
>
> Setzen Sie ein Lesezeichen für diese URL, um den Zugriff zu erleichtern. Dies ist der universelle Einstiegspunkt, über den Sie alle Agent Spaces einsehen können, denen Sie zugewiesen sind.

3. Navigieren Sie zur URL der Webanwendung.
4. Geben Sie Ihre IAM Identity Center-Anmeldeinformationen ein, wenn Sie dazu aufgefordert werden.
5. Nach der Authentifizierung wird eine Liste aller Agent Spaces angezeigt, denen Sie zugewiesen sind.
6. Wählen Sie den Agent Space aus, in dem Sie arbeiten möchten.

Um direkt auf einen bestimmten Agent Space zuzugreifen (erfordert AWS-Konsolenzugriff):

1. Melden Sie sich bei der AWS-Managementkonsole an.
2. Navigieren Sie zur AWS Security Agent-Konsole.

3. Navigieren Sie zu dem Agent-Bereich, auf den Sie zugreifen möchten.
4. Wählen Sie eine der folgenden Optionen:
   - Schaltfläche „Webanwendung starten“ auf der Agent Space-Übersichtsseite
   - Schaltfläche „Web-App starten“ im Abschnitt „Codeüberprüfung“
   - Schaltfläche „Web-App starten“ im Bereich „Penetrationstests“
5. Geben Sie Ihre IAM Identity Center-Anmeldeinformationen ein, wenn Sie dazu aufgefordert werden.
6. Nach der Authentifizierung werden Sie direkt zu diesem Agent Space in der Webanwendung weitergeleitet.

> **Note**
>
> Wenn Sie keinen Zugriff auf die AWS-Konsole haben, verwenden Sie die Webanwendungsdomäne auf der Seite Einstellungen, um auf alle Ihnen zugewiesenen Agent Spaces zuzugreifen.

Melden Sie sich mit Administratorzugriff (IAM) an

Wenn Sie Zugriff auf die AWS-Konsole mit den entsprechenden AWS Security Agent-Berechtigungen haben, können Sie die Webanwendung über den Administratorzugriffslink mit automatischer Authentifizierung starten.

Voraussetzungen

- Sie sind bei der AWS-Managementkonsole angemeldet
- Sie verfügen über die entsprechenden Berechtigungen für den Zugriff auf AWS Security Agent-Ressourcen

Greifen Sie auf die Webanwendung zu

Wählen Sie eine der folgenden Methoden:

So greifen Sie auf einen bestimmten Agent Space zu:

1. Melden Sie sich bei der AWS-Managementkonsole an.

2. Navigieren Sie zur AWS Security Agent-Konsole.
3. Navigieren Sie zu dem Agent-Bereich, auf den Sie zugreifen möchten.
4. Wählen Sie auf der Übersichtsseite von Agent Space die Schaltfläche Admin-Zugriff.
5. Die Webanwendung wird auf einer neuen Registerkarte mit automatischer Authentifizierung für diesen Agent Space geöffnet.

> **Note**
>
> Die Schaltfläche für den Administratorzugriff steht Benutzern mit AWS-Konsolenzugriff immer zur Verfügung, unabhängig davon, ob Ihre Organisation IAM Identity Center als primäre Zugriffsmethode verwendet.

Erstellen Sie eine Entwurfsprüfung

Beurteilen Sie Ihre Entwurfsdokumente anhand der Sicherheitsanforderungen Ihres Unternehmens, indem Sie Dateien hochladen, damit der AWS Security Agent sie überprüfen kann. Entwurfsprüfungen helfen dabei, Sicherheitsprobleme früh im Entwicklungszyklus zu erkennen, sodass Sie architektonische Probleme dann lösen können, wenn sie am kostengünstigsten zu lösen sind.

AWS Security Agent analysiert Ihre Entwurfsdokumente anhand der Sicherheitsanforderungen Ihres Unternehmens und liefert detaillierte Sicherheitsergebnisse, um die Sicherheitslage vor Beginn der Implementierung zu verbessern.

In diesem Verfahren erstellen Sie eine Entwurfsprüfung, indem Sie Designdateien zur Analyse hochladen.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Zugriff auf die AWS Security Agent-Webanwendung
- Entwerfen Sie Dokumente, die zum Hochladen bereit sind (DOC, DOCX, JPEG, MD, PDF, PNG und TXT)
- Jede Datei muss 2 MB oder kleiner sein, insgesamt also 6 MB für alle Dateien

- Verstehen Sie, welche Sicherheitsanforderungen für Ihr Unternehmen aktiviert sind

Schritt 1: Beginnen Sie mit der Erstellung einer Entwurfsprüfung

Navigieren Sie in der Agent Web App zur Seite zur Erstellung von Design Reviews.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Navigieren Sie zum Bereich Design Reviews.
3. Wählen Sie „Design Review erstellen“.

> **Tip**
>
> Sie können die aktivierten Sicherheitsanforderungen Ihres Unternehmens einsehen, indem Sie zur Seite mit den Sicherheitsanforderungen in der AWS Security Agent-Konsole navigieren. Wählen Sie eine aktivierte Anforderung aus, um deren Details einzusehen. Diese Anforderungen werden zur Analyse Ihrer Konstruktionsdateien verwendet.

Schritt 2: Benennen Sie Ihre Entwurfsprüfung

Geben Sie einen aussagekräftigen Namen an, anhand dessen Zweck und Umfang dieser Entwurfsprüfung besser erkennbar sind.

1. Suchen Sie im Abschnitt Name der Entwurfsprüfung das Feld Name.
2. Geben Sie einen aussagekräftigen Namen für Ihre Entwurfsprüfung ein.

> **Note**
>
> Der Name sollte das Projekt, die Funktion oder die Komponente, die überprüft wird, eindeutig identifizieren. Maximal 80 Zeichen.

Schritt 3: Laden Sie Designdateien hoch

Laden Sie die Entwurfsdokumente hoch, die der AWS Security Agent auf Einhaltung der Sicherheitsbestimmungen analysieren soll.

1. Überprüfen Sie im Abschnitt Zu überprüfende Dateien die Dateianforderungen:

> **Important**
>
> Pro Entwurfsprüfung können maximal 5 Dateien hochgeladen werden. Jede Datei muss 2 MB oder kleiner sein, insgesamt also 6 MB für alle Dateien. Unterstützte Formate: DOC, DOCX, JPEG, MD, PDF, PNG und TXT.

2. Laden Sie Ihre Dateien mit einer der folgenden Methoden hoch:
   a. Drag & Drop — Ziehe Dateien direkt in den Datei-Dropzone-Bereich
   b. Durchsuchen — Verwenden Sie die Schaltfläche „Dateien auswählen“, um Dateien auf Ihrem Computer zu suchen und auszuwählen
3. Stellen Sie sicher, dass alle erforderlichen Dateien hochgeladen wurden.

> **Tip**
>
> Um optimale Ergebnisse zu erzielen, fügen Sie Architekturdiagramme, Entwurfsspezifikationen und technische Unterlagen bei, in denen die sicherheitsrelevanten Komponenten und Datenflüsse Ihres Systems beschrieben werden.

Schritt 4: Initiieren Sie die Entwurfsprüfung

Nachdem Sie alle erforderlichen Informationen konfiguriert haben, starten Sie die Sicherheitsanalyse Ihrer Konstruktionsdokumente.

1. Überprüfen Sie alle hochgeladenen Dateien und Einstellungen, um die Richtigkeit sicherzustellen.
2. Wählen Sie Entwurfsprüfung starten.
3. AWS Security Agent analysiert Ihre Entwurfsdokumente anhand der aktivierten Sicherheitsanforderungen.

> **Note**
>
> Die Entwurfsprüfung ist in der Regel innerhalb von Minuten abgeschlossen, abhängig von der Anzahl und Größe der hochgeladenen Dateien. Sie erhalten Sicherheitsergebnisse, die auf den Sicherheitsanforderungen Ihres Unternehmens basieren.

Nächste Schritte

Nachdem Sie mit Ihrer Entwurfsprüfung begonnen haben:

- Überwachen Sie den Fortschritt der Überprüfung in der Agent Web App
- Überprüfen Sie die Sicherheitsergebnisse
- Teilen Sie die Ergebnisse mit Ihrem Entwicklungsteam
- Gehen Sie in Ihrem Design auf identifizierte Sicherheitslücken ein
- Aktualisieren Sie die Entwurfsdokumente und reichen Sie sie bei Bedarf erneut ein

Überprüfen Sie die Ergebnisse einer Entwurfsprüfung

Die Ergebnisse der Überprüfung helfen Ihnen zu verstehen, welche Sicherheitsanforderungen erfüllt sind, welche Aufmerksamkeit erfordern und welche Maßnahmen Sie ergreifen müssen, um die Sicherheitslage Ihres Entwurfs vor Beginn der Implementierung zu verbessern.

In diesem Verfahren erfahren Sie, wie Sie auf die Ergebnisse der Entwurfsprüfung zugreifen, diese filtern und interpretieren können, um Sicherheitslücken effektiv zu behandeln.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Eine abgeschlossene Entwurfsprüfung
- Zugriff auf die AWS Security Agent-Webanwendung
- Vertrautheit mit den aktivierten Sicherheitsanforderungen Ihres Unternehmens

Schritt 1: Greifen Sie auf die Entwurfsprüfung zu

Rufen Sie Ihre Entwurfsprüfung auf, um die Ergebnisse und die Zusammenfassung einzusehen.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Navigieren Sie zum Bereich Design Reviews.
3. Wählen Sie die Entwurfsprüfung, die Sie prüfen möchten, aus der Liste aus.

> **Tip**
>
> Auf der Detailseite der Entwurfsprüfung wird eine Zusammenfassung des Prüfungsstatus, des Abschlussdatums und der Anzahl der überprüften Dateien angezeigt.

Schritt 2: Überprüfen Sie die Zusammenfassung der Ergebnisse

Sehen Sie sich die allgemeine Zusammenfassung an, um sich ein Bild von der allgemeinen Sicherheitslage Ihres Entwurfs zu machen.

1. Suchen Sie den Abschnitt Zusammenfassung.
2. Überprüfen Sie die Anzahl für jede Kategorie des Konformitätsstatus: Konform Non-compliant, Ungenügende Daten und Nicht zutreffend.

> **Note**
>
> Die Zusammenfassung enthält Zählungen für jeden Statustyp, sodass Sie schnell einschätzen können, wie viele Ergebnisse Ihrer Aufmerksamkeit bedürfen. Ausführliche Erläuterungen zu den einzelnen Status finden Sie in Schritt 4.

Schritt 3: Ergebnisse filtern und navigieren

Verwenden Sie die Filter- und Suchfunktionen, um sich auf bestimmte Ergebnisse oder den Compliance-Status zu konzentrieren.

1. Suchen Sie im Abschnitt Ergebnisse überprüfen nach den Filtersteuerelementen.
2. Um nach Status zu filtern:
   a. Wählen Sie das Status-Dropdown-Menü.
   b. Wählen Sie einen bestimmten Compliance-Status aus, um nur Ergebnisse mit diesem Status anzuzeigen.

3. Um nach bestimmten Sicherheitsanforderungen zu suchen:
   a. Geben Sie Stichwörter in das Suchfeld ein.
   b. Die Ergebnisse werden während der Eingabe automatisch aktualisiert.
4. Verwenden Sie die Steuerelemente für die Seitennummerierung, um durch mehrere Ergebnisseiten zu navigieren.

> **Tip**
>
> Filtern Sie zunächst nach Non-compliantStatus, um Ergebnisse, die sofortige Aufmerksamkeit erfordern, in Ihrem Design zu priorisieren.

Schritt 4: Verstehen Sie den Compliance-Status

Für jedes Ergebnis wird ein Konformitätsstatus angezeigt, der angibt, wie Ihr Design eine bestimmte Sicherheitsanforderung erfüllt:

- Konformität — Ihr Design erfüllt die Sicherheitsanforderungen auf der Grundlage der Analyse
- Non-compliant— Ihr Entwurf verstößt gegen die Sicherheitsanforderung oder erfüllt sie nur unzureichend
- Unzureichende Daten — Den hochgeladenen Dateien fehlen genügend Informationen, um die Einhaltung der Vorschriften zu überprüfen
- Nicht zutreffend — Die Sicherheitsanforderung gilt nicht für Ihr Systemdesign

> **Important**
>
> Status „Konzentrieren Sie sich auf Daten“ Non-compliantund „Ungenügend“, da diese Maßnahmen erfordern. Korrigieren Sie fehlerhafte Ergebnisse, indem Sie Ihren Entwurf aktualisieren, und beheben Sie unzureichende Datenergebnisse, indem Sie zusätzliche Konstruktionsunterlagen hochladen.

Schritt 5: Details zu den Ergebnissen anzeigen

Wählen Sie einzelne Ergebnisse aus, um eine detaillierte Begründung und Hinweise zur Problembehebung einzusehen.

1. Wählen Sie in der Tabelle mit den Ergebnissen einen Namen für die Sicherheitsanforderung aus.
2. Überprüfen Sie die Details zu den Ergebnissen, darunter:
   - Die spezifische Sicherheitsanforderung wird derzeit bewertet
   - Eine Bemerkung, in der erklärt wird, warum das Ergebnis den Status „Konformität“ erhalten hat, einschließlich genauer Angaben darüber, was fehlt oder nicht konform ist
   - Empfohlene Leitlinien zur Behebung des Fehlers
   - Links zur internen Dokumentation oder zu den Standards Ihrer Organisation für die Sicherheitsanforderungen

> **Note**
>
> Der Kommentar erklärt die Argumentation des AWS Security Agents mit spezifischen Details. Bei unzureichenden Datenergebnissen wird in dem Kommentar angegeben, welche Informationen fehlen, z. B. „In den Entwurfsdokumenten werden Authentifizierungsmechanismen nicht erwähnt“ oder „Es wurden keine Informationen zur Datenverschlüsselung im Ruhezustand gefunden“.

Schritt 6: Gehen Sie auf die Ergebnisse ein

Ergreifen Sie Maßnahmen auf Ergebnisse, die Ihrer Aufmerksamkeit bedürfen, um die Sicherheitslage Ihres Entwurfs zu verbessern.

Für Non-compliantErgebnisse:

1. Lesen Sie die empfohlenen Leitlinien zur Problembehebung.
2. Überprüfen Sie alle verlinkten internen Dokumentationen auf zusätzlichen Kontext.
3. Aktualisieren Sie Ihre Konstruktionsdokumente, um die Sicherheitsanforderungen zu erfüllen.
4. Dokumentieren Sie die von Ihnen vorgenommenen Änderungen zum future Nachschlagen.

Bei unzureichenden Datenergebnissen:

1. Lesen Sie den Kommentar sorgfältig durch, um zu verstehen, welche spezifischen Informationen fehlen.
2. Erstellen oder aktualisieren Sie Konstruktionsdokumente mit den fehlenden Details.

3. Bereiten Sie die aktualisierten Dateien für die erneute Einreichung vor.

Nächste Schritte

Nachdem Sie Ihre Entwurfsergebnisse überprüft haben:

- Laden Sie den Ergebnisbericht als CSV-Datei herunter, um ihn mit Ihrem Team zu teilen
- Aktualisieren Sie die Konstruktionsdokumente, um fehlerhafte Ergebnisse zu beheben
- Erstellen Sie zusätzliche Dokumentation für unzureichende Datenergebnisse
- Teilen Sie die Ergebnisse Ihrem Entwicklungsteam zur Diskussion mit
- Klonen Sie diese Entwurfsprüfung, um eine neue Prüfung zu erstellen, bei der die Originaldokumente vorinstalliert sind. So können Sie den Namen aktualisieren und die Analyse erneut ausführen, um die Verbesserungen zu überprüfen
- Fahren Sie mit der Implementierung von Entwürfen fort, die die Compliance-Anforderungen erfüllen

Weitere Informationen zur Verwaltung von Sicherheitsanforderungen finden Sie unterthe section called “Sicherheitsanforderungen verwalten”.

Weitere Informationen zum Erstellen von Entwurfsprüfungen finden Sie unterthe section called “Erstellen Sie eine Entwurfsprüfung”.

Erstellen Sie ein Bedrohungsmodell

Erstellen Sie Bedrohungsmodelle in der AWS Security Agent-Webanwendung, um die Architektur Ihrer Anwendung zu analysieren und Sicherheitsbedrohungen zu identifizieren. Ein Bedrohungsmodell liefert zwei Hauptergebnisse: eine Systemübersicht, die beschreibt, wie Ihr System aufgebaut ist und sich verhält, und eine Reihe von Bedrohungen, die beschreiben, wie das System angegriffen werden könnte, jeweils mit einem Schweregrad, einer STRIDE-Klassifizierung und umsetzbaren Empfehlungen.

Ein Bedrohungsmodell akzeptiert zwei Arten von Eingaben, und Sie können eine oder beide angeben:

- Dokumente zum Umfang — Dokumente zum technischen Design, API-Spezifikationen oder Architekturdokumentationen, die definieren, worauf sich das Bedrohungsmodell konzentriert. Sofern bereitgestellt, beschränkt der Agent seine Analyse auf den in diesen Dokumenten

beschriebenen Kontext. Sie können Dateien hochladen (DOC, DOCX, JPEG, MD, PDF, PNG, TXT), S3-URIs bereitstellen oder Confluence-Seiten über eine Integration verbinden.

- Quellen — Quellcode aus verbundenen Repositorys (GitHub, GitHub Enterprise Server, GitLab GitLab Self-Managed, oder Bitbucket) oder von S3-Standorten. Der AWS Security Agent liest den Quellcode, um Ihr bestehendes System zu verstehen. In Kombination mit den Scope-Dokumenten bietet der Quellcode einen Kontext zur aktuellen Implementierung.

In diesem Verfahren erstellen Sie ein Bedrohungsmodell, indem Sie dessen Eingaben und Berechtigungen konfigurieren, und starten dann eine Ausführung.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Zugriff auf die AWS Security Agent-Webanwendung
- Mindestens eine der folgenden Optionen:
  - Ein verbundenes Repository (GitHub, GitHub Enterprise Server, GitLab GitLab Self-Managed, oder Bitbucket), das als Quelle verwendet werden soll (siehethe section called “Bedrohungsmodellierung aktivieren”)
  - Ein S3-Bucket, der Quellcode enthält
  - Entwerfen Sie Dokumente, die als Scope-Dokumente oder als Confluence-Integration hochgeladen werden können

> **Tip**
>
> Wenn Sie bereits Repositorys oder S3-Buckets mit Ihrem Agent Space verbunden haben (z. B. durch die Einrichtung von Codeüberprüfungen oder Penetrationstests), können Sie sofort ein Bedrohungsmodell erstellen — es ist keine zusätzliche Einrichtung erforderlich.

Wie AWS Security Agent ein Bedrohungsmodell ausführt

Die von Ihnen bereitgestellten Eingaben bestimmen, wie der AWS Security Agent das Bedrohungsmodell ausführt. Es gibt drei Szenarien.

Eingaben, die Sie bereitstellen	So führt AWS Security Agent das Bedrohungsmodell aus
Nur Quellen (Quellcode)	AWS Security Agent analysiert den Quellcode, um die Struktur Ihrer Anwendung zu verstehen, klassifiziert Komponenten, extrahiert Datenflüsse, erstellt ein Bedrohungsmodell und identifiziert Bedrohungen auf der Grundlage der tatsächlichen Implementierung des Systems.
Nur Dokumente zum Geltungsbereich (Entwurfsdokumente)	AWS Security Agent führt ein Bedrohungsmodell aus, das sich auf das in Ihren Dokumenten beschriebene Design konzentriert. Es identifiziert Bedrohungen auf der Grundlage der Architektur, der Datenflüsse und der Komponenten, die in den Umfangsdokumenten beschrieben sind. Dies ist nützlich, um ein Bedrohungsmodell zu modellieren, bevor Code existiert.
Sowohl Quellen als auch Dokumente zum Geltungsbereich	AWS Security Agent ordnet das Bedrohungsmodell dem in den Geltungsbereichsdokumenten beschriebenen Kontext zu (z. B. einem Entwurfsdokument für eine neue Funktion). Es verwendet den Quellcode, um Ihr bestehendes System zu verstehen, und modelliert dann das in den Scope-Dokumenten beschriebene Design — unabhängig davon, ob dieses Design bereits implementiert ist oder nicht.

Rufen Sie die Seite mit den Bedrohungsmodellen auf

Navigieren Sie in der Webanwendung zum Bereich Bedrohungsmodellierung.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Wählen Sie in der linken Seitenleiste Bedrohungsmodelle aus.
3. Sie sehen eine Liste vorhandener Bedrohungsmodelle mit ihrem Titel, dem Status der letzten Ausführung und der Anzahl der Bedrohungen nach Schweregrad.

Erstellen Sie ein Bedrohungsmodell

Richten Sie ein neues Bedrohungsmodell ein, indem Sie dessen Eingaben und Berechtigungen konfigurieren.

1. Wählen Sie auf der Seite Bedrohungsmodelle die Option Bedrohungsmodell erstellen aus.

Konfigurieren Sie die Details zum Bedrohungsmodell

Geben Sie einen Titel für Ihr Bedrohungsmodell ein.

1. Geben Sie im Feld Titel einen aussagekräftigen Namen für Ihr Bedrohungsmodell ein.

> **Tip**
>
> Verwenden Sie einen Namen, der die Anwendung oder den Dienst identifiziert, der modelliert wird. Zum Beispiel „Billing-Service“ oder „Checkout-API“.

Dokumente zum Geltungsbereich hinzufügen

Stellen Sie die technischen Entwurfsdokumente bereit, die definieren, worauf sich das Bedrohungsmodell konzentriert, z. B. Architekturdokumentation, API-Spezifikationen oder Designvorschläge. Wenn Dokumente zum Umfang bereitgestellt werden, beschränkt der Agent seine Analyse auf den in diesen Dokumenten beschriebenen Kontext. Scope-Dokumente sind optional, wenn Sie Quellen angeben.

1. Fügen Sie im Abschnitt Scope-Dokumente Dokumente mit einer oder mehreren der folgenden Methoden hinzu:
   - Dateien hochladen — Laden Sie Designdokumente direkt hoch (DOC, DOCX, JPEG, MD, PDF, PNG oder TXT).
   - S3-URIs — Geben Sie S3-URIs ein, die auf Dokumente verweisen, die in verbundenen S3-Buckets gespeichert sind.
   - Confluence-Seiten — Wenn Sie eine Confluence-Integration mit Ihrem Agent Space verbunden haben, wählen Sie Seiten aus Ihrem Confluence-Workspace aus.

> **Tip**
>
> Um optimale Ergebnisse zu erzielen, fügen Sie Architekturdiagramme, API-Spezifikationen und technische Unterlagen hinzu, die die Komponenten, Datenflüsse und Vertrauensgrenzen Ihres Systems beschreiben.

Fügen Sie Quellen hinzu

Wählen Sie den Quellcode aus, den AWS Security Agent verwendet, um Ihr bestehendes System zu verstehen. In Kombination mit den Scope-Dokumenten bietet der Quellcode einen Kontext zur aktuellen Implementierung — der Agent verwendet ihn, um zu verstehen, was bereits existiert. Quellen sind optional, wenn Sie Scope-Dokumente bereitstellen.

1. Fügen Sie im Abschnitt Quellen Quellcode mit einer oder mehreren der folgenden Methoden hinzu:

Integrierte Repositorys

Wähle aus Repositorys, die mit deinem Agent Space (GitHub, GitHub Enterprise Server, GitLab GitLab Self-Managed, oder Bitbucket) verbunden sind.

1. Markiere das Kästchen neben jedem Repository, das du als Quelle einbeziehen möchtest.
2. Verwenden Sie das Suchfeld, um bestimmte Repositorys anhand des Namens zu finden.

> **Note**
>
> Hier werden nur Repositorys angezeigt, die mit Ihrem Agent Space verbunden sind. Um weitere Repositorys hinzuzufügen, bitten Sie Ihren Administrator, die Agent Space-Konfiguration zu aktualisieren.

S3-Quellen

Fügen Sie S3-URIs hinzu, die auf Quellcode verweisen, der in verbundenen S3-Buckets gespeichert ist.

1. Geben Sie den S3-URI für jeden Quellcode-Speicherort ein, den Sie einbeziehen möchten.

AWS-Ressourcen konfigurieren

Wählen Sie die IAM-Servicerolle und die optionale CloudWatch Protokollgruppe für dieses Bedrohungsmodell aus.

1. Wählen Sie in der Dropdownliste Servicerolle die IAM-Rolle aus Ihren konfigurierten Servicerollen aus.

> **Note**
>
> Die Servicerolle muss über Berechtigungen verfügen, um auf Ihren Quellcode in S3 zuzugreifen und in Protokolle zu CloudWatch schreiben. Servicerollen werden während der Einrichtung von Agent Space in der AWS-Managementkonsole konfiguriert.

2. (Optional) Wählen Sie in der Dropdownliste Protokollgruppe eine CloudWatch Protokollgruppe aus, in der Ausführungsprotokolle für das Bedrohungsmodell gespeichert werden sollen.

Erstellen Sie das Bedrohungsmodell

1. Überprüfen Sie die Sicherheitskonfiguration.
2. Wählen Sie Bedrohungsmodell erstellen aus.

Sie werden auf die Detailseite des Bedrohungsmodells weitergeleitet, auf der Sie einen Testlauf starten können.

Führen Sie ein Bedrohungsmodell aus

Nachdem Sie ein Bedrohungsmodell erstellt haben, starten Sie einen Rechenlauf, um mit der Analyse zu beginnen.

1. Wählen Sie auf der Detailseite des Bedrohungsmodells die Option Start run aus.
2. AWS Security Agent beginnt mit der Analyse Ihrer Quellen und Dokumente zum Geltungsbereich.

Sie können einen Rechenlauf auch von der Seite mit der Liste der Bedrohungsmodelle aus starten, indem Sie neben dem Bedrohungsmodell, das Sie ausführen möchten, die Option Ausführung starten auswählen.

Überwachen Sie die Ausführung eines Bedrohungsmodells

Verfolgen Sie den Fortschritt Ihres Bedrohungsmodells bei der Ausführung.

Run status (Ausführungsstatus)

In der Kopfzeile der Run-Seite wird eine Statusanzeige angezeigt:

- In Bearbeitung — Der Threat Model Agent wird ausgeführt.
- Stoppen — Es wurde ein Abbruch angefordert. Der Agent beendet gerade seine aktuelle Aufgabe, bevor er den Vorgang beendet.
- Abgeschlossen — Die Ausführung wurde erfolgreich abgeschlossen.
- Fehlgeschlagen — Bei der Ausführung ist ein Fehler aufgetreten. Eine Fehlermeldung mit Einzelheiten wird angezeigt. Starten Sie einen neuen Lauf, nachdem Sie das Problem behoben haben.
- Gestoppt — Der Lauf wurde vom Benutzer abgebrochen. Alle Bedrohungen, die vor dem Stopp generiert wurden, bleiben erhalten.

Seiten-Tabs ausführen

Navigieren Sie auf der Seite mit den Ausführungsdetails zwischen den Tabs:

- Überblick — Sehen Sie sich die Zusammenfassung der Ausführung (Job-ID, Startzeit, Status, Dauer) mit einem Kreisdiagramm mit Schweregrad an, in dem die Bedrohungen nach Schweregrad (Kritisch, Hoch, Mittel, Niedrig) aufgeschlüsselt sind. Unter der Zusammenfassung sehen Sie ein Balkendiagramm mit STRIDE-Bedrohungskategorien, das zeigt, wie viele Bedrohungen in die einzelnen STRIDE-Kategorien fallen. Sehen Sie sich nach Abschluss der Ausführung die vom Agenten erstellte Systemübersicht an.
- Konfiguration — Sehen Sie sich die Quellen, Dokumente und Berechtigungen (Servicerolle, CloudWatch Protokollgruppe) an, die für diesen Lauf verwendet wurden.
- Preflight — Zeigt den Status der Preflight-Prüfungen an, die vor Beginn der Bedrohungsanalyse ausgeführt werden, einschließlich der Einrichtung der Protokollierungsinfrastruktur, der Validierung des S3-Quellzugriffs (falls zutreffend) und der Einrichtung der Testumgebung. Ein Fortschrittsbalken zeigt an, wie viele Prüfungen abgeschlossen wurden.
- Protokolle — Zeigt eine filterbare Liste der Aufgaben an, die der Agent während der Ausführung ausgeführt hat. Wählen Sie eine beliebige Aufgabe aus, um ihre detaillierte Protokollausgabe in einem Seitenbereich anzuzeigen.
- Bedrohungen — Sehen Sie sich die Bedrohungen an, die während des Testlaufs erkannt wurden (siehethe section called “Überprüfen Sie Bedrohungen anhand eines Bedrohungsmodells”).

Verlauf ausführen

Jedes Bedrohungsmodell führt eine Historie aller Läufe. Gehen Sie auf der Detailseite des Bedrohungsmodells wie folgt vor:

- In der Ausführungstabelle sind alle Läufe mit Startzeit, Status, Dauer, Anzahl der Bedrohungen und ID aufgeführt.
- Wählen Sie den Link zur Startzeit eines beliebigen Laufs, um die vollständigen Details zu sehen.

> **Tip**
>
> Re-run dasselbe Bedrohungsmodell, nachdem Sie Ihren Quellcode aktualisiert oder Ihre Scope-Dokumente überarbeitet haben, um zu sehen, wie sich die Systemübersicht und die Bedrohungen im Laufe der Zeit ändern.

Bearbeiten Sie ein Bedrohungsmodell

So ändern Sie die Konfiguration Ihres Bedrohungsmodells:

1. Wählen Sie auf der Detailseite des Bedrohungsmodells die Option Bearbeiten aus.
2. Aktualisieren Sie den Titel, die Quellen, die Dokumente zum Geltungsbereich oder die AWS-Ressourcen nach Bedarf.
3. Wählen Sie Änderungen speichern aus.

> **Note**
>
> Die Bearbeitung eines Bedrohungsmodells hat keine Auswirkungen auf zuvor abgeschlossene Läufe. Diese behalten den Konfigurations-Snapshot bei, der zum Zeitpunkt der Ausführung verwendet wurde.

Löschen Sie ein Bedrohungsmodell

Gehen Sie wie folgt vor, um ein Bedrohungsmodell und alle zugehörigen Varianten und Bedrohungen zu löschen:

1. Öffnen Sie auf der Detailseite des Bedrohungsmodells das Aktionsmenü und wählen Sie Löschen.
2. Bestätigen Sie das Löschen.

> **Important**
>
> Wenn gerade ein Rechenlauf läuft, müssen Sie ihn beenden, bevor Sie das Bedrohungsmodell löschen können.

Nächste Schritte

Nach der Ausführung eines Bedrohungsmodells:

- Sehen Sie sich die Systemübersicht und die Bedrohungen an (siehethe section called “Überprüfen Sie Bedrohungen anhand eines Bedrohungsmodells”)
- Re-run das Bedrohungsmodell, nachdem Sie Änderungen vorgenommen haben, um sicherzustellen, dass die Bedrohungen behoben wurden
- Passen Sie Ihre Quellen und den Umfang der Dokumente an, wenn sich Ihre Anwendung weiterentwickelt

Überprüfen Sie Bedrohungen anhand eines Bedrohungsmodells

Sehen Sie sich nach Abschluss eines Bedrohungsmodells die Systemübersicht und die Bedrohungen an, um zu verstehen, wie Ihre Anwendung angegriffen werden könnte und was dagegen zu tun ist. Die Systemübersicht ist ein umfassendes Dokument, in dem die Architektur, die Vertrauensgrenzen, die Datenflüsse und die Sicherheitslage Ihrer Anwendung beschrieben werden. Zu jeder Bedrohung gehören eine Erklärung, der Schweregrad, die STRIDE-Klassifizierung, die betroffenen Ressourcen und eine Empfehlung zur Bekämpfung der Bedrohung.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein abgeschlossener Test des Bedrohungsmodells
- Zugriff auf die AWS Security Agent-Webanwendung

Schritt 1: Greifen Sie auf die Ausführung des Bedrohungsmodells zu

Navigieren Sie zu Ihrem abgeschlossenen Bedrohungsmodell-Lauf.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Wählen Sie in der linken Seitenleiste Bedrohungsmodelle aus.
3. Wählen Sie das Bedrohungsmodell aus, das Sie untersuchen möchten.
4. Wählen Sie in der Ausführungstabelle den abgeschlossenen Lauf aus, indem Sie auf den Link zur Startzeit klicken.

Schritt 2: Überprüfen Sie die Systemübersicht

Die Systemübersicht ist eine umfassende Beschreibung, wie AWS Security Agent Ihr System versteht. Es handelt sich um ein strukturiertes Dokument, das Folgendes beinhalten kann:

- Zweck — Was das System tut und wem es dient.
- Funktionen — Die wichtigsten Funktionen, die das System bietet.
- Entwurfsabsicht — Die Entwurfsänderung oder die Funktion, für die ein Bedrohungsmodell modelliert wird (sofern Dokumente zum Umfang bereitgestellt werden).
- Architektur — Aufbau des Systems, einschließlich Bereitstellungsmustern und Kommunikationsprotokollen.
- Komponenten — Eine Tabelle mit Systemkomponenten mit ihrem Zweck und ihren wichtigsten Interaktionen.
- Vertrauensgrenzen — Wo sich der Sicherheitskontext ändert, einschließlich der Schutzmaßnahmen, die an jedem Grenzübergang bestehen.
- Datenflüsse — Detaillierte Beschreibungen der Art und Weise, wie Daten durch das System fließen, einschließlich Protokollen, Anmeldeinformationen und Schutzmaßnahmen bei jedem Schritt.
- Sicherheitslage — Aktuelle Authentifizierungs-, Verschlüsselungs- und Zugriffskontrollmechanismen.
- Vertrauliche Ressourcen — Daten und Anmeldeinformationen, die geschützt werden müssen, mit ihrer Klassifizierung und ihren Gefahrenquellen.
- Wichtige Annahmen — Security-relevant Annahmen, die der Agent über das System getroffen hat.

Um die Systemübersicht zu überprüfen:

1. Wählen Sie die Registerkarte „Übersicht“.
2. Sehen Sie sich den Abschnitt „Zusammenfassung der Ausführung“ an, in dem die Job-ID, die Startzeit, der Status und die Dauer angezeigt werden.
3. Sehen Sie sich das Diagramm Schweregrad an, das eine Aufschlüsselung der Bedrohungen nach Schweregrad (Kritisch, Hoch, Mittel, Niedrig) mit Anzahl und Prozentwerten enthält.
4. Sehen Sie sich das Diagramm mit den Bedrohungskategorien an, das zeigt, wie viele Bedrohungen in die einzelnen STRIDE-Kategorien fallen (Spoofing, Manipulation, Ablehnung, Offenlegung von Informationen, Denial-of-Service, Erhöhung von Rechten).
5. Lesen Sie im Abschnitt Systemübersicht die vollständige Analyse des Agenten.

> **Tip**
>
> Wenn die Systemübersicht Ihr System nicht genau wiedergibt, verfeinern Sie Ihre Eingaben — fügen Sie relevante Repositorys als Quellen hinzu oder laden Sie umfassendere Dokumente zum Umfang hoch — und führen Sie das Bedrohungsmodell erneut aus.

Schritt 3: Überprüfen Sie die Bedrohungen

Rufen Sie die Registerkarte Bedrohungen auf, um sich alle Bedrohungen anzusehen, die während des Testlaufs erkannt wurden.

1. Wählen Sie die Registerkarte Bedrohungen aus.
2. Bedrohungen werden als Liste angezeigt, wobei auf jeder Karte der Titel, der Schweregrad, der Status und der Zeitpunkt der letzten Aktualisierung aufgeführt sind. Sie können Bedrohungen nach Schweregrad und Status filtern oder nach Titel suchen.
3. Wählen Sie eine Bedrohung aus der Liste aus, um die vollständigen Details im rechten Bereich anzuzeigen.

Schweregrad der Bedrohung

Jeder Bedrohung wird ein Schweregrad zugewiesen:

- Kritisch — Sofortiges Handeln erforderlich; eine Ausnutzung kann zu einer vollständigen Gefährdung des Systems führen.

- Hoch — Erfordert sofortiges Handeln; eine Ausnutzung kann zu erheblichen Sicherheitsrisiken führen.
- Mittel — Sollte innerhalb eines angemessenen Zeitrahmens behoben werden; trägt zum allgemeinen Sicherheitsrisiko bei.
- Niedrig — Kann im Rahmen einer regelmäßigen Wartung behoben werden; minimales unmittelbares Risiko.

Einzelheiten zur Bedrohung

Wählen Sie eine Bedrohung aus, um ihre Details im rechten Bereich anzuzeigen. Die Details sind in die folgenden Abschnitte unterteilt:

Metadatenzeile:

- Schweregrad — Die vom Agenten zugewiesene Risikostufe (Kritisch, Hoch, Mittel oder Niedrig).
- STRIDE-Kategorien — Die Bedrohungsklassifizierung: Spoofing, Manipulation, Ablehnung, Offenlegung von Informationen, Denial-of-Service oder Erhöhung von Rechten.
- Erstellt am — Als die Bedrohung identifiziert wurde.
- Letzte Aktualisierung — Datum, an dem die Bedrohung zuletzt geändert wurde.

Abschnitt Beschreibung:

- Erklärung — Eine Beschreibung der Bedrohung in natürlicher Sprache: was die Bedrohungsquelle tun kann, welche Auswirkungen sie hat und welche Bedingungen sie ermöglichen.
- Quelle — Der Akteur oder der Ursprung der Bedrohung (z. B. „authentifizierter Benutzer“ oder „externer Angreifer“).
- Aktion — Was die Bedrohungsquelle tun kann (z. B. „SQL-Abfragen in den Suchparameter einfügen“).
- Auswirkung — Die direkte Folge der Bedrohungsaktion (z. B. „unbefugter Zugriff auf Kundendaten“).

Abschnitt „Referenzen“:

- Betroffene Anlagen — Bestimmte Vermögenswerte, die von der Bedrohung betroffen sind (z. B. „Kundenzahlungsdaten“ oder „DynamoDB-Tabelle“).

- Voraussetzungen — Bedingungen, die erfüllt sein müssen, damit die Bedrohung ausgenutzt werden kann.
- Betroffene Ziele — Betroffene Sicherheitsziele: Vertraulichkeit, Integrität, Verfügbarkeit, Autorisierung, Authentifizierung oder. Non-repudiation
- Beweise — Quellcode-Dateipfade, die die Bedrohung unterstützen und auf bestimmte Dateien in Ihrem Repository verweisen.
- Anchor — Eine Codereferenz, die die Bedrohung mit einem bestimmten Knoten in Ihrem Quellcode verknüpft.

Abschnitt mit Empfehlungen:

- Empfehlung — Umsetzbare Leitlinien zur Bekämpfung der Bedrohung.

Schritt 4: Manuelles Erstellen Sie eine Bedrohung

Sie können Bedrohungen hinzufügen, die der Agent nicht identifiziert hat, z. B. Bedrohungen, die bei der manuellen Überprüfung entdeckt wurden, oder Bedrohungen aus externen Quellen.

1. Wählen Sie nach Abschluss eines Testlaufs auf der Registerkarte Bedrohungen die Option Bedrohung erstellen aus.
2. Geben Sie die Bedrohungsdetails ein:
   - Erklärung — Eine Beschreibung der Bedrohung in natürlicher Sprache.
   - Schweregrad — Wählen Sie Kritisch, Hoch, Mittel oder Niedrig aus.
   - Quelle — Der Akteur oder Ursprung der Bedrohung.
   - Voraussetzungen — Bedingungen, die erforderlich sind, damit die Bedrohung ausgenutzt werden kann.
   - Aktion — Was die Bedrohungsquelle tun kann.
   - Auswirkung — Die direkte Folge der Bedrohungsaktion.
   - Betroffene Vermögenswerte — Bestimmte betroffene Vermögenswerte (durch Kommas getrennt).
   - Betroffene Sicherheitsziele — Wählen Sie zwischen Vertraulichkeit, Integrität, Verfügbarkeit, Autorisierung, Authentifizierung oder. Non-repudiation
   - STRIDE-Kategorien — Wählen Sie die entsprechenden Kategorien aus.
   - Empfehlung — Leitlinien zur Bekämpfung der Bedrohung.

3. Wählen Sie Erstellen aus.

Die manuell erstellte Bedrohung wird in der Bedrohungsliste zusammen mit den vom Agenten generierten Bedrohungen angezeigt.

Schritt 5: Bedrohungen bearbeiten und nach Bedrohungen suchen

Während Sie Bedrohungen überprüfen, können Sie deren Details bearbeiten und ihren Status aktualisieren, um den Fortschritt zu verfolgen.

1. Wählen Sie eine Bedrohung aus der Liste aus.
2. Wählen Sie das Bearbeitungssymbol im Bereich mit den Bedrohungsdetails, um das Bearbeitungsformular zu öffnen.
3. Sie können die folgenden Felder ändern:
   - Status — Verfolgen Sie den Bedrohungszyklus:
     - Offen — Die Bedrohung wurde erkannt und erfordert Aufmerksamkeit (Standardeinstellung).
     - Gelöst — Sie haben das Problem behoben.
     - Abgewiesen — Sie haben die Bedrohung überprüft und festgestellt, dass sie nicht zutrifft.
   - Schweregrad — Passen Sie den Schweregrad an, falls die Einschätzung des Agenten nicht Ihrem Kontext entspricht.
   - Erklärung — Verfeinern Sie die Beschreibung der Bedrohung.
   - Quelle, Voraussetzungen, Aktion, Auswirkung — Aktualisieren Sie die Bedrohungsdetails auf der Grundlage Ihrer Fachkenntnisse.
   - Betroffene Ressourcen — Fügen Sie betroffene Ressourcen hinzu oder entfernen Sie sie (durch Kommas getrennt).
   - Betroffene Sicherheitsziele — Wählen Sie die betroffenen Sicherheitsziele aus (Vertraulichkeit, Integrität, Verfügbarkeit, Autorisierung, Authentifizierung,). Non-repudiation
4. Klicken Sie auf Speichern, um Ihre Änderungen zu übernehmen.

Schritt 6: Laden Sie einen Bericht herunter

Nach Abschluss eines Laufs können Sie einen PDF-Bericht herunterladen, der die Systemübersicht und alle identifizierten Bedrohungen zusammenfasst.

1. Wählen Sie auf der Seite mit abgeschlossener Ausführung die Option Bericht erstellen aus.

2. Die PDF-Datei wird auf Ihren Computer heruntergeladen.

Schritt 7: Überprüfen Sie die Preflight-Checks und -Protokolle

Gehen Sie wie folgt vor, wenn Sie untersuchen möchten, wie der Agent zu seinen Schlussfolgerungen gekommen ist, oder wenn Sie einen Teilfehler debuggen möchten:

1. Wählen Sie die Registerkarte Preflight, um den Status der Preflight-Prüfungen anzuzeigen, die vor Beginn der Bedrohungsanalyse ausgeführt werden. Jede Prüfung zeigt ihren Status an (abgeschlossen, in Bearbeitung oder ausstehend), und ein Fortschrittsbalken gibt an, wie viele Prüfungen abgeschlossen wurden. Sie können eine abgeschlossene Prüfung erweitern, um ihre detaillierte Protokollausgabe anzuzeigen.
2. Wählen Sie die Registerkarte Protokolle, um eine filterbare Liste der Aufgaben anzuzeigen, die der Agent während der Ausführung ausgeführt hat. Wählen Sie eine beliebige Aufgabe aus der Liste aus, um ihre detaillierte Protokollausgabe im Seitenbereich anzuzeigen.

Nächste Schritte

Nachdem Sie die Ergebnisse Ihres Bedrohungsmodells überprüft haben:

- Gehen Sie zuerst auf der Grundlage der Empfehlungen des Agenten vor Bedrohungen mit hohem Schweregrad
- Aktualisieren Sie den Bedrohungsstatus, während Sie Fixes implementieren
- Führen Sie ein neues Bedrohungsmodell durch, um zu überprüfen, ob Ihre Änderungen die identifizierten Bedrohungen adressieren
- Passen Sie Ihre Quellen und den Umfang der Dokumente an, wenn sich Ihre Anwendung weiterentwickelt (siehethe section called “Erstellen Sie ein Bedrohungsmodell”)

Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests

Nachdem Sie die Codeüberprüfung für Pull-Anfragen für Ihre Repositorys aktiviert haben, analysiert der AWS Security Agent automatisch Pull-Anfragen und veröffentlicht Sicherheitsergebnisse direkt an Ihren Quellcodekontrollanbieter. Auf diese Weise können Entwickler Sicherheitsprobleme innerhalb ihres normalen Workflows lösen, ohne den Pull-Request verlassen zu müssen.

> **Note**
>
> Diese Seite bezieht sich auf GitHub Pull-Requests, GitLab Merge-Requests und Bitbucket-Pull-Requests. Die Erfahrung ist bei allen Anbietern ähnlich.

Wie funktioniert die Codeüberprüfung bei Pull-Requests

Wenn Sie eine Pull-Anfrage (oder eine Merge-Anfrage in GitLab) in einem Repository mit aktiviertem Code-Review einreichen, beginnt der AWS Security Agent automatisch mit der Analyse.

1. Auslöser für die Pull-Request-Analyse — Die Codeüberprüfung wird ausgelöst, wenn eine Pull-Anfrage in Repositorys, in denen Sie die Code-Review-Funktion aktiviert haben, als „Bereit zur Überprüfung“ markiert ist. Pull-Request-Entwürfe werden nicht analysiert.
2. Bestätigung der Analyse — Wenn der AWS Security Agent mit der Analyse Ihrer Pull-Anfrage beginnt, wird ein erster Kommentar veröffentlicht: „AWS Security Agent analysiert Ihren Code...“ Dadurch wissen Sie, dass die Analyse gestartet wurde und noch läuft.
3. Abschluss der Überprüfung — Nach Abschluss der Analyse veröffentlicht der AWS Security Agent eine Überprüfung Ihrer Pull-Anfrage mit den Ergebnissen. Alle Sicherheitsergebnisse werden in einer einzigen Überprüfung zusammengefasst, um Ihre Pull-Anfrage zu organisieren und die Anzahl der Benachrichtigungen zu minimieren.

Die Ergebnisse von Code-Reviews verstehen

AWS Security Agent liefert unterschiedliche Arten von Ergebnissen, je nachdem, was er bei der Analyse findet.

Wenn Sicherheitsprobleme gefunden werden

Wenn der AWS Security Agent Sicherheitsprobleme in Ihren Codeänderungen feststellt, veröffentlicht er eine Überprüfung, die Folgendes beinhaltet:

- Zusammenfassung — Ein allgemeiner Überblick über alle Sicherheitsfeststellungen, in dem die Arten der identifizierten Probleme und ihre möglichen Auswirkungen beschrieben werden
- Einzelne Ergebnisse — Die detaillierten Sicherheitsergebnisse werden in Form von Thread-Kommentaren unter der Hauptüberprüfung angezeigt, wobei jedes Ergebnis Folgendes beinhaltet:
  - Beschreibung des Sicherheitsproblems

- Stelle in Ihrem Code, an der das Problem gefunden wurde
- Anleitung zur Problembehebung, in der erklärt wird, wie das Problem behoben werden kann
- Relevanter Kontext, der auf Ihren Einstellungen für die Codeüberprüfung basiert (Verstöße gegen Sicherheitsanforderungen, allgemeine Sicherheitslücken oder beides)

> **Note**
>
> Die Art der analysierten Sicherheitsprobleme hängt von Ihren Einstellungen für die Codeüberprüfung ab. Wenn Sie die Überprüfung von Sicherheitsanforderungen konfiguriert haben, beziehen sich die Ergebnisse auf die individuellen Sicherheitsanforderungen Ihres Unternehmens. Wenn Sie die Ergebnisse von Sicherheitslücken konfiguriert haben, werden anhand der Ergebnisse häufig auftretende Sicherheitslücken identifiziert. Weitere Informationen zu den Einstellungen für die Codeüberprüfung finden Sie unterthe section called “Pull-Request-Code-Review für GitHub Repositorys aktivieren”.

Wenn keine Sicherheitsprobleme gefunden wurden

Wenn der AWS Security Agent die Analyse abschließt und keine Sicherheitsprobleme in Ihren Codeänderungen feststellt, veröffentlicht er einen Kommentar: „Es wurden keine Probleme identifiziert.“ Dadurch wird bestätigt, dass die Überprüfung erfolgreich abgeschlossen wurde und Ihre Codeänderungen aufgrund Ihrer konfigurierten Einstellungen für die Codeüberprüfung keine Sicherheitsfeststellungen ausgelöst haben.

Auf Sicherheitsfeststellungen reagieren

Nachdem Sie die vom AWS Security Agent veröffentlichten Sicherheitsergebnisse überprüft haben, können Sie direkt bei Ihrem Quellcodekontrollanbieter Maßnahmen ergreifen.

- Behebung von Ergebnissen — Aktualisieren Sie Ihren Code auf der Grundlage der in den Ergebnissen enthaltenen Hinweise zur Problembehebung und fügen Sie dann neue Commits zur Pull-Anfrage hinzu. Der AWS Security Agent analysiert den aktualisierten Code.
- Konversationen lösen — Nachdem Sie eine Sicherheitslücke behoben haben, markieren Sie die Konversation als gelöst, um Ihren Fortschritt zu verfolgen.

> **Tip**
>
> Jedes Ergebnis beinhaltet spezifische Anleitungen zur Behebung, die auf das identifizierte Sicherheitsproblem zugeschnitten sind. Lesen Sie sich diese Leitlinien sorgfältig durch, um das Sicherheitsrisiko zu verstehen und zu erfahren, wie es wirksam angegangen werden kann.

Filterung der Ergebnisse von Code-Reviews

Sie können anpassen, wie der AWS Security Agent Ihren Code analysiert, indem Sie Ihrem Repository eine `filtering.md` Datei hinzufügen. Mit dieser Datei können Sie Fehlalarme reduzieren, indem Sie Kontext zu Ihrer Codebasis bereitstellen und Dateien oder Ordner von der Analyse ausschließen.

Die Filterdatei wird erstellt

Erstellen Sie eine Datei mit dem Namen `filtering.md` im `.awssecurityagent` Verzeichnis im Stammverzeichnis Ihres Repositorys:

```
.awssecurityagent/filtering.md
```

AWS Security Agent liest diese Datei bei der Analyse von Pull-Anfragen aus dem Hauptzweig Ihres Repositorys (z. B. `main` oder`mainline`).

Dateistruktur

Die `filtering.md` Datei verwendet die Standard-Markdown-Formatierung mit bestimmten Abschnitten, die der AWS Security Agent erkennt. Die Datei muss eine `Code Review` Überschrift enthalten, gefolgt von einem oder beiden der folgenden Abschnitte: `IgnorePatterns` und `ContextHints` (kein Leerzeichen).

Das folgende Beispiel zeigt die vollständige Struktur einer `filtering.md` Datei:

```
# filtering.md

## Code Review

### IgnorePatterns
```

```
**/*.md

/myapp/src/**/*.snap

/myapp/config/README

### ContextHints

- The backend is a trusted system and won't return non-standard protocols.
- URL is generated from server with presigned token, so no SSRF security
 vulnerabilities.
- AppSec has verified that we are allowed to use cache with an eviction policy.
```

Ignoriere Muster

`IgnorePatterns`In diesem Abschnitt werden Dateien und Ordner angegeben, die der AWS Security Agent bei der Codeüberprüfung überspringen sollte. Wird verwendet`glob patterns`, um zu definieren, welche Pfade von der Analyse ausgeschlossen werden sollen.

Anforderungen an das Format:

- Jedes Muster muss sich in einer eigenen Zeile befinden.
- Trennen Sie jedes Muster durch eine Leerzeile dazwischen. Dadurch wird sichergestellt, dass die Datei korrekt gerendert wird, wenn sie in unseren GitHub Tools zur Codeüberprüfung angezeigt wird.
- Muster folgen dem Standard-Glob-Format. `**/*.md`Entspricht beispielsweise allen Markdown-Dateien und `/myapp/src/**/*.snap` entspricht allen `.snap` Dateien innerhalb des `/myapp/src/` Ordners im Stammverzeichnis.
- In diesem Abschnitt werden bis zu 1000 Ignoriermuster unterstützt.

Hinweise zum Kontext

Dieser `ContextHints` Abschnitt bietet zusätzlichen Kontext zu Ihrer Codebasis, der dem AWS Security Agent hilft, genauere Einschätzungen vorzunehmen. Verwenden Sie Kontexthinweise, um Architekturentscheidungen, Sicherheitsausnahmen oder andere Informationen zu erläutern, die sich auf die Interpretation der Ergebnisse auswirken könnten.

Anforderungen an das Format:

- Jeder Hinweis muss mit einem Gedankenstrich (`-`) gefolgt von einem Leerzeichen beginnen.

- Schreiben Sie jeden Hinweis als eine einzelne Zeile Freiformtext, die auf 500 Zeichen begrenzt ist.
- Jeder Hinweis sollte einen bestimmten Kontext zu Ihrer Codebasis beschreiben.
- In diesem Abschnitt werden bis zu 20 Kontexthinweise unterstützt.

Kontexthinweise werden angewendet, nachdem der AWS Security Agent seine erste Analyse abgeschlossen hat, und helfen so dabei, Ergebnisse herauszufiltern, die nicht auf Ihren speziellen Anwendungsfall zutreffen.

Nächste Schritte

Nach der Überprüfung der Ergebnisse zur Codesicherheit:

- Aktualisieren Sie Ihren Code auf der Grundlage von Anleitungen zur Problembehebung
- Pushen Sie neue Commits, um eine erneute Analyse Ihrer Änderungen auszulösen
- Passen Sie bei Bedarf die Einstellungen für die Codeüberprüfung an (siehe) the section called “Pull-Request-Code-Review für GitHub Repositorys aktivieren”
- Informieren Sie sich über die Sicherheitsanforderungen Ihres Unternehmens, um die Validierungskriterien zu verstehen
- Ziehen Sie Penetrationstests für eine umfassende Sicherheitsvalidierung der bereitgestellten Anwendungen in Betracht

Einen Code-Review erstellen

Erstellen Sie Code-Reviews in der AWS Security Agent-Webanwendung, um Ihre Quellcode-Repositorys und S3-Quellen auf Sicherheitslücken zu durchsuchen. Code-Reviews führen umfassende statische Analysen für Ihre gesamte Codebasis durch, identifizieren Sicherheitsprobleme und bieten Anleitungen zur Behebung.

Im Gegensatz zur auf Pull-Requests basierenden Codeüberprüfung, bei der einzelne Codeänderungen analysiert werden (siehethe section called “Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests”), wird bei Codeprüfungen auf Abruf Ihr vollständiger Quellcode gescannt, um Sicherheitslücken zu identifizieren und die Einhaltung der Sicherheitsanforderungen Ihres Unternehmens zu überprüfen.

In diesem Verfahren erstellen Sie eine Codeüberprüfung, indem Sie Quellcodeeingaben auswählen, Berechtigungen konfigurieren und die Überprüfung ausführen.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Zugriff auf die AWS Security Agent-Webanwendung
- Mindestens ein verbundenes GitHub Repository oder S3-Bucket in Ihrem Agent Space

> **Tip**
>
> Wenn Sie bereits GitHub Repositorys mit Ihrem Agent Space verbunden haben, ist Code Review sofort einsatzbereit — es ist keine zusätzliche Einrichtung erforderlich. Wählen Sie auf der Karte zur Codeüberprüfung auf Ihrer Agent Space-Seite die Option In Web-App starten aus, oder starten Sie die Webanwendung direkt.
> Wenn Sie zusätzliche Quellen verbinden oder S3-Buckets konfigurieren müssen, finden Sie weitere Informationen unterthe section called “Codeüberprüfung aktivieren”.

Rufen Sie die Seite mit den Code-Reviews auf

Navigieren Sie in der Webanwendung zum Bereich Code-Reviews.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Wählen Sie in der linken Seitenleiste Code Reviews aus.
3. Sie sehen eine Liste vorhandener Code-Reviews mit ihren Quellinformationen, dem Status der letzten Ausführung und einer Zusammenfassung der Ergebnisse.

Erstellen Sie einen Code-Review

Richten Sie einen neuen Code-Review ein, indem Sie dessen Quellcode-Eingaben und Berechtigungen konfigurieren.

1. Wählen Sie auf der Seite Code-Reviews die Option Code-Review erstellen aus.

Konfigurieren Sie die Details zur Codeüberprüfung

Geben Sie einen Titel ein und wählen Sie den Quellcode aus, den Sie überprüfen möchten.

1. Geben Sie im Feld Titel einen aussagekräftigen Namen für Ihre Code-Überprüfung ein.

> **Tip**
>
> Verwenden Sie einen Namen, der die Anwendung, das Repository oder den Umfang der Überprüfung identifiziert. Zum Beispiel „billing-service-security-review“ oder „infrastructure-code-audit“.

2. Wählen Sie im Bereich Quellen die Quellcode-Eingaben für diese Überprüfung aus. Wählen Sie aus zwei Tabs:

GitHub Repositorien

Wählen Sie aus Repositorys, die mit Ihrem Agent Space verbunden sind.

1. Wählen Sie den Tab GitHub Repositorys.
2. Aktivieren Sie in der Tabelle Integrierte Repositorys das Kontrollkästchen neben jedem Repository, das Sie in die Überprüfung einbeziehen möchten.
3. Verwenden Sie das Suchfeld, um bestimmte Repositorys anhand des Namens zu finden.

> **Note**
>
> Hier werden nur Repositorys angezeigt, die über die Code-Review-Konfiguration mit Ihrem Agent Space verbunden sind. Um weitere Repositorys hinzuzufügen, wählen Sie in Ihrer Admin-Konsole die Option Verwalten oder bitten Sie Ihren Administrator, die Agent Space-Konfiguration zu aktualisieren.

S3-Quellen

Wählen Sie ZIP-Dateien aus den S3-Buckets aus, die mit Ihrem Agent Space verbunden sind. Ihr Agent Space-Administrator konfiguriert, welche S3-Buckets verfügbar sind. Jede ZIP-Datei, die in einem dieser Buckets gespeichert ist, kann als Quelle für eine Codeüberprüfung verwendet werden.

1. Wählen Sie den Tab S3-Quellen.
2. Geben Sie die S3-URI jeder ZIP-Datei ein, die Sie in die Überprüfung aufnehmen möchten. Sie können bis zu 30 S3-Quellen hinzufügen.

> **Note**
>
> S3-Quellen müssen ZIP-Dateien sein, die in S3-Buckets gespeichert sind, die mit Ihrem Agent Space verbunden sind. Informationen zur Bereitstellung zusätzlicher Buckets finden Sie unter. the section called “Codeüberprüfung aktivieren”

Konfigurieren von -Berechtigungen

Wählen Sie die IAM-Servicerolle und die optionale CloudWatch Protokollgruppe für diesen Code-Review aus.

1. Suchen Sie im Abschnitt „Berechtigungen“ nach der Dropdownliste „Servicerolle“.
2. Wählen Sie die IAM-Rolle aus Ihren konfigurierten Servicerollen aus.

   > **Note**
   >
   > Die Servicerolle muss über Berechtigungen verfügen, um auf Ihren Quellcode in S3 zuzugreifen und in CloudWatch Protokolle sowie alle anderen AWS-Ressourcen zu schreiben, die für die Codeüberprüfung benötigt werden. Servicerollen werden bei der Einrichtung der Codeüberprüfung in der AWS-Managementkonsole konfiguriert.

3. (Optional) Wählen Sie in der Dropdownliste „CloudWatch Protokollgruppe“ eine Protokollgruppe aus, in der die Ausführungsprotokolle für die Codeüberprüfung gespeichert werden sollen.

   > **Note**
   >
   > Wenn Sie keine Protokollgruppe auswählen, erstellt der AWS Security Agent eine Standard-Protokollgruppe zum Speichern von Code-Review-Protokollen.

Konfigurieren Sie die automatische Codekorrektur

Aktivieren Sie die automatische Problembehebung, damit der AWS Security Agent Codekorrekturen für alle Ergebnisse generiert, sobald die Überprüfung abgeschlossen ist.

1. Aktivieren Sie im Abschnitt Automatische Codekorrektur das Kontrollkästchen Automatische Codekorrektur aktivieren.

Wie der AWS Security Agent das Update bereitstellt, hängt von der Quelle ab:

- Private GitHub Repositorys — Der AWS Security Agent sendet eine Pull-Anfrage mit dem Fix an das Repository.
- Öffentliche GitHub Repositorys — Um zu verhindern, dass die Sicherheitsanfälligkeit offengelegt wird, bevor sie behoben ist, öffnet der AWS Security Agent keine Pull-Anfrage. Stattdessen fügt er dem Ergebnis einen Vorschlag für einen Unterschied bei, den Sie von der Webanwendung herunterladen und privat anwenden können.
- S3-Quellen — Codekorrektur ist nicht verfügbar. Überprüfen Sie die Details zu den Ergebnissen und wenden Sie die Korrekturen manuell an.

> **Important**
>
> An private Repositorys übermittelte Pull-Requests zur Problembehebung sind für alle Benutzer mit Lesezugriff sichtbar. Überprüfe die Änderungen vor dem Zusammenführen. Die automatische Codekorrektur ist nur verfügbar, wenn GitHub Repositorys als Quelle ausgewählt wurden.

> **Note**
>
> Wenn diese Option deaktiviert ist, können Sie die Codekorrektur für einzelne GitHub-sourced Ergebnisse auch nach Abschluss der Überprüfung manuell auslösen.

Konfigurieren Sie die simulierte Validierung

Aktivieren Sie die simulierte Validierung, um dynamisch zu bestätigen, ob entdeckte Sicherheitslücken in einer laufenden Anwendung ausgenutzt werden können.

1. Aktivieren Sie im Abschnitt Simulierte Validierung das Kontrollkästchen Simulierte Validierung aktivieren.

Wenn diese Option aktiviert ist, stellt der AWS Security Agent nach Abschluss der statischen Analyse eine simulierte Umgebung bereit. Er integriert Ihren Quellcode, startet Dienste innerhalb der Umgebung und versucht, während des Scans gefundene Sicherheitslücken auszunutzen.

Die Ergebnisse werden dann mit einem Bestätigungsstatus aktualisiert, der angibt, ob die Sicherheitsanfälligkeit erfolgreich ausgenutzt wurde.

> **Note**
>
> Die simulierte Validierung ist derzeit nur für eigenständige Docker-Anwendungen verfügbar. Wenn mehrere Repositorys als Quellen ausgewählt wurden, ist die simulierte Validierung nicht verfügbar.

> **Important**
>
> Die simulierte Validierung verlängert die Verarbeitungszeit Ihres Code-Review-Laufs. Der Validierungsschritt stellt eine Umgebung bereit und führt Ausnutzungsversuche durch. Dieser Vorgang dauert in der Regel 1—3 Stunden, abhängig von der Anzahl der Ergebnisse und der Komplexität der Anwendung.

Zulässige Netzwerkziele

In der simulierten Umgebung ist der ausgehende Netzwerkzugriff auf eine vordefinierte Zulassungsliste beschränkt. Ihre Anwendung kann während der Validierung die folgenden Domänen erreichen:

In der folgenden Tabelle sind die zulässigen Domänen und ihre Zwecke aufgeführt.

Domain	Zweck
`.amazonaws.com` , `.aws.amazon.com`	AWS-Services
`.public.ecr.aws`	Amazon ECR Public
`.docker.com` , `.docker.io`	Docker Hub
`.github.com` , `.githubusercontent.com`	GitHub

Domain	Zweck
`.gitlab.com`	GitLab
`.npmjs.com` , `.npmjs.org`	npm-Registrierung
`.pypi.org` , `.pypi.python.org` , `.pythonhosted.org`	Python-Paketindex
`.crates.io` , `.rustup.rs`	Rust-Pakete
`.maven.org` , `.gradle.org`	Java/Gradle Pakete
`.nuget.org`	.NET-Pakete
`.rubygems.org` , `.ruby-lang.org`	Ruby-Pakete
`.golang.org` , `.pkg.go.dev` , `.goproxy.io`	Go-Pakete
`.nodejs.org` , `.yarnpkg.com`	Node.js
`.alpinelinux.org` , `.debian.org` , `.ubuntu.com` , `.centos.org` , `.fedoraproject.org`	Repositorien für Linux-Distributionen
`.cloudfront.net`	CloudFront Distributionen
`.google.com` , `.googleapis.com`	Google-APIs
`.microsoft.com` , `.visualstudio.com`	Microsoft-Dienste

Domain	Zweck
`.sourceforge.net` , `.bitbucket.org`	Quell-Hosting

> **Note**
>
> Wenn Ihre Anwendung Netzwerkzugriff auf Domänen benötigt, die nicht in dieser Liste aufgeführt sind, wird die Verbindung durch die Netzwerk-Firewall blockiert. Anwendungen, die von externen APIs oder Diensten abhängig sind, die oben nicht aufgeführt sind, werden in der simulierten Umgebung möglicherweise nicht korrekt gestartet.

Erstellen Sie den Code-Review

1. Überprüfen Sie Ihre Konfiguration, um die Richtigkeit sicherzustellen.
2. Wählen Sie Codeüberprüfung erstellen aus.

Sie werden auf die Detailseite für die Codeüberprüfung weitergeleitet, auf der Sie eine Überprüfung starten können.

Führen Sie einen Code-Review durch

Nachdem Sie einen Code-Review erstellt haben, starten Sie einen Rechenlauf, um mit der Analyse zu beginnen.

1. Wählen Sie auf der Detailseite des Code-Reviews die Option Überprüfung starten aus.
2. AWS Security Agent beginnt mit der Analyse Ihres Quellcodes.

Sie können eine Überprüfung auch auf der Seite mit der Liste der Codeüberprüfungen starten, indem Sie neben der Codeüberprüfung, die Sie durchführen möchten, die Option Überprüfung starten auswählen.

Überwachen Sie einen Code-Review-Lauf

Verfolgen Sie den Fortschritt Ihrer Codeüberprüfung während der Ausführung.

Überprüfen Sie die Ausführungsphasen

Ein Codeüberprüfungslauf durchläuft die folgenden Phasen, die als Fortschrittsanzeige angezeigt werden:

1. Preflight — Der AWS Security Agent validiert den Zugriff auf Ihren Quellcode und richtet die Testumgebung ein. Die Preflight-Prüfungen beinhalten:
   - Einrichtung der Serviceinfrastruktur
   - Validierung des S3-Quellzugriffs
   - Testumgebung einrichten
2. Statische Analyse — AWS Security Agent scannt Ihren Quellcode auf Sicherheitslücken und Verstöße gegen Anforderungen.
3. Simulierte Validierung (optional) — Wenn die simulierte Validierung aktiviert ist, stellt der AWS Security Agent eine simulierte Umgebung bereit und stellt Ihre Anwendung bereit. Anschließend versucht er, die bei der statischen Analyse entdeckten Sicherheitslücken auszunutzen. In diesem Schritt wird bestätigt, ob die Ergebnisse in der Ziellaufzeit ausgenutzt werden können. Diese Phase wird nur angezeigt, wenn Sie bei der Erstellung des Code-Reviews die simulierte Validierung aktivieren.
4. Finalisieren — AWS Security Agent kompiliert die Ergebnisse und generiert eine Zusammenfassung der Ergebnisse.

Ausführungsdetails anzeigen

Navigieren Sie auf der Detailseite des Laufs zwischen den Tabs, um den Fortschritt zu überwachen:

- Ausführung zur Codeüberprüfung — Zeigt die Zusammenfassung der Ausführung an, einschließlich der Lauf-ID, der Erstellungszeit, des Status, der Dauer, der Stunden der Aufgaben, der Aufschlüsselung nach Schweregrad und des Diagramms mit den Risikoarten.
- Preflight — Sehen Sie sich den Fortschritt und den Status der einzelnen Validierungsschritte der Preflight-Prüfung an.
- Code-Review-Protokolle — Sehen Sie sich die Aufgaben an, die der AWS Security Agent während der Überprüfung identifiziert und ausgeführt hat, mit detaillierten Aufgabenprotokollen für jeden Schritt.
- Simulierte Validierung — Wenn die simulierte Validierung aktiviert ist, können Sie sich den Bereitstellungsstatus, die Validierungsaufgaben für einzelne Ergebnisse und deren Ergebnisse anzeigen lassen.

- Ergebnisse — Sehen Sie sich die Sicherheitsergebnisse nach Abschluss der Überprüfung an (siehe). the section called “Ergebnisse eines Code-Reviews überprüfen”

Verlauf ausführen

Bei jeder Codeüberprüfung wird ein Verlauf aller Läufe gespeichert. Auf der Detailseite zur Codeüberprüfung:

- Im Abschnitt Letzte Ausführung wird die letzte Ausführung mit Startzeit, Status, Dauer und ID angezeigt.
- In der Tabelle Alle Läufe werden alle vorherigen Läufe mit Startzeit, Status, Dauer, Zusammenfassung der Ergebnisse und ID aufgeführt.
- Wählen Sie Ausführung überwachen, um die Details des letzten aktiven Laufs anzuzeigen.
- Wählen Sie den Link zur Startzeit eines beliebigen Laufs, um die vollständigen Details zu sehen.

Nächste Schritte

Nach der Durchführung eines Code-Reviews:

- Überprüfen Sie die Sicherheitsergebnisse und die entsprechenden Hinweise zur Behebung dieser Probleme (siehethe section called “Ergebnisse eines Code-Reviews überprüfen”)
- Korrigieren Sie die Ergebnisse mithilfe automatisierter Pull-Requests oder manueller Fixes (siehe) the section called “Korrigieren Sie die Ergebnisse der Codeüberprüfung”
- Führen Sie nach der Implementierung der Korrekturen weitere Überprüfungen durch, um die Behebung zu überprüfen
- Passen Sie Ihre Konfiguration oder Quellen für die Codeüberprüfung an, wenn sich Ihre Codebasis weiterentwickelt

Ergebnisse eines Code-Reviews überprüfen

Lesen Sie nach Abschluss einer Codeüberprüfung die Zusammenfassung der Ausführung und die Sicherheitsergebnisse, um mehr über Sicherheitslücken in Ihrem Quellcode zu erfahren. Jedes Ergebnis enthält eine Beschreibung, eine Bewertung des Schweregrads, die Position des Codes, eine Begründung für das Risiko und Lösungsvorschläge, die Ihnen helfen, Sicherheitsprobleme zu priorisieren und zu beheben.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein abgeschlossener oder laufender Code-Review-Lauf
- Zugriff auf die AWS Security Agent-Webanwendung

Schritt 1: Greifen Sie auf den Codeüberprüfungslauf zu

Navigieren Sie zu Ihrem abgeschlossenen Code-Review-Lauf, um die Zusammenfassung und die Ergebnisse einzusehen.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Wählen Sie in der linken Seitenleiste Code Reviews aus.
3. Wählen Sie den Code-Review aus, den Sie untersuchen möchten.
4. Wählen Sie in der Tabelle Alle Läufe den abgeschlossenen Lauf aus, indem Sie auf den Link mit der Startzeit klicken. Wählen Sie alternativ Ausführung überwachen aus, um den letzten Lauf anzuzeigen.

Schritt 2: Überwachen Sie den Fortschritt des Laufs

Verfolgen Sie den Fortschritt Ihres Codeüberprüfungslaufs mithilfe der Schrittanzeige.

1. Suchen Sie den horizontalen Schrittindikator unter dem Seitenkopf.
2. Überprüfen Sie den Status jeder Phase:
   - Preflight — Überprüft den Zugriff auf Ihren Quellcode und richtet die Scanumgebung ein. Zu den Prüfungen gehören die Einrichtung der Serviceinfrastruktur, die Validierung des S3-Quellzugriffs und die Einrichtung der Testumgebung, zu der auch GitHub Zugriffsprüfungen gehören.
   - Statische Analyse — Scannt Ihren Quellcode auf Sicherheitslücken und Verstöße gegen Anforderungen.
   - Simulierte Validierung (optional) — Wenn diese Option aktiviert ist, wird eine simulierte Umgebung bereitgestellt und versucht, entdeckte Sicherheitslücken in der laufenden Anwendung auszunutzen.
   - Finalisieren — Stellt die Ergebnisse zusammen und generiert eine Zusammenfassung der Ergebnisse.

> **Note**
>
> Für jeden Schritt wird eine Statusanzeige angezeigt (Abgeschlossen oder In Bearbeitung). Der Lauf ist abgeschlossen, wenn für alle Phasen Completed (Abgeschlossen) angezeigt wird. Der simulierte Validierungsschritt wird nur angezeigt, wenn Sie die simulierte Validierung bei der Erstellung des Code-Reviews aktivieren.

Schritt 3: Überprüfen Sie die Zusammenfassung der Ausführung

Navigieren Sie zur Registerkarte „Ausführung der Codeüberprüfung“, um sich die allgemeinen Ergebnisse anzusehen.

1. Wählen Sie die Registerkarte Ausführen der Codeüberprüfung aus.
2. Der Abschnitt mit der Zusammenfassung der Ausführung enthält Informationen zum Status und zur Dauer der Ausführung sowie weitere allgemeine Informationen. Es bietet auch ein Dashboard mit Sicherheitsergebnissen, die nach Schweregrad und Risikotypen kategorisiert sind.
3. Die Anwendungsübersicht von AWS Security Agent bietet eine Zusammenfassung des Code-Review-Scans, wenn der Lauf abgeschlossen ist

Schritt 4: Überprüfen Sie die Preflight-Ergebnisse

Gehen Sie zur Registerkarte Preflight, um zu überprüfen, ob alle Quellzugriffsprüfungen bestanden wurden.

1. Wählen Sie die Registerkarte Preflight aus.
2. Überprüfen Sie die Preflight-Fortschrittsanzeige, die die Anzahl der abgeschlossenen Prüfungen anzeigt.
3. Vergewissern Sie sich, dass für jede Prüfung ein Erfolgsstatus angezeigt wird:
   - Einrichtung der Serviceinfrastruktur — Bestätigt, dass die Testumgebung bereit ist
   - Überprüfung des S3-Quellzugriffs — Bestätigt den Zugriff auf den S3-Quellcode (falls zutreffend)
   - Testumgebung einrichten — Bestätigt, dass die Analyseumgebung konfiguriert ist

> **Note**
>
> Wenn eine Preflight-Prüfung fehlschlägt, wird die Ausführung nicht mit der statischen Analyse fortgeführt. Überprüfen Sie die Prüfdetails, um Zugriffsprobleme zu identifizieren und zu beheben, und starten Sie dann einen neuen Lauf.

Schritt 5: Überprüfen Sie die Code-Review-Protokolle

Navigieren Sie zur Registerkarte Code Review Logs, um sich die Aufgaben anzusehen, die der AWS Security Agent während der Analyse ausgeführt hat.

1. Wählen Sie die Registerkarte Code-Review-Protokolle aus.
2. Durchsuchen Sie die Liste der Aufgaben im linken Bereich.
3. Wählen Sie eine Aufgabe aus, um ihr detailliertes Protokoll im rechten Bereich anzuzeigen.

> **Tip**
>
> Verwenden Sie das Suchfeld, um Aufgaben nach Namen zu filtern. Die Protokolle geben Aufschluss darüber, was der AWS Security Agent untersucht hat und wie er zu seinen Ergebnissen gekommen ist.

Schritt 6: Navigieren Sie zu den Ergebnissen

Wählen Sie die Registerkarte Ergebnisse, um alle Sicherheitsergebnisse des Laufs anzuzeigen.

> **Note**
>
> Standard-Konfidenzfilter
>
> Standardmäßig werden Ihnen nur Ergebnisse mit hoher Agentensicherheit angezeigt. Um auch Ergebnisse mit mittlerer oder niedriger Agentensicherheit und falsch positive Ergebnisse anzuzeigen, deaktivieren Sie die Option Unbestätigte Ergebnisse ausblenden.

1. Wählen Sie die Registerkarte Ergebnisse aus.

2. Die Ergebnisse werden in einer geteilten Ansicht mit der Ergebnisliste auf der linken Seite und den Details des ausgewählten Ergebnisses auf der rechten Seite angezeigt.
3. Sehen Sie sich die Informationen an, die auf den einzelnen Ergebniskarten angezeigt werden:
   - Fundtitel — Ein aussagekräftiger Name, der das Sicherheitsproblem zusammenfasst
   - Schweregradkennzeichen — Color-coded Schweregradindikator:
     - Kritisch (rot) — Sofortiges Handeln erforderlich; eine Ausnutzung kann zu einer Beeinträchtigung des Systems führen
     - Hoch (rot) — Sofortiges Handeln ist erforderlich; eine Ausnutzung kann zu erheblichen Sicherheitsrisiken führen
     - Mittel (orange) — Sollte innerhalb eines angemessenen Zeitrahmens behoben werden; trägt zum allgemeinen Sicherheitsrisiko bei
     - Niedrig (gelb) — Kann im Rahmen regelmäßiger Wartungsarbeiten behoben werden; minimales unmittelbares Risiko
   - Letzte Aktualisierung — Zeitstempel, zu dem der Befund zuletzt aktualisiert wurde

Schritt 7: Überprüfen Sie die Details zu den Ergebnissen

Wählen Sie einzelne Ergebnisse aus, um umfassende Informationen zu den einzelnen Sicherheitslücken zu erhalten.

1. Wählen Sie im linken Bereich ein Ergebnis aus, um die zugehörigen Details im rechten Bereich anzuzeigen.
2. Sehen Sie sich die verfügbaren Aktionen an:
   - Befund lösen — Markieren Sie den Befund als gelöst, nachdem Sie ihn behoben haben
   - Code korrigieren — Generieren Sie eine Pull-Anfrage mit einem Fix (für GitHub Quellen verfügbar)
3. Geben Sie Feedback mithilfe von War dieses Ergebnis korrekt? — Wählen Sie Ja oder Nein, um die Genauigkeit future Analysen zu verbessern.
4. Sehen Sie sich die wichtigsten Attribute im Abschnitt „Übersicht“ an:
   - Vertrauen der Agenten — Das Vertrauensniveau, das AWS Security Agent in dieses Ergebnis eingeht
   - Schweregrad — Die Risikostufe mit einem farblich markierten Schild
   - Risikotyp — Die Kategorie des Sicherheitsrisikos

- Validierungsstatus — Wenn die simulierte Validierung aktiviert ist, wird angezeigt, ob das Ergebnis in einer laufenden Umgebung als ausnutzbar bestätigt wurde:
  - Bestätigt — Die Sicherheitsanfälligkeit wurde in der simulierten Umgebung erfolgreich ausgenutzt
  - Nicht reproduziert — Die Sicherheitsanfälligkeit konnte in der Ziellaufzeit nicht ausgenutzt werden
  - Überprüfung fehlgeschlagen — Der Validierungsversuch war aufgrund eines Fehlers ergebnislos
  - Nicht validiert — Für dieses Ergebnis wurde keine simulierte Validierung durchgeführt. Dies tritt auf, wenn die Validierung nicht aktiviert ist oder der Validierungsschritt das Timeout überschritten hat, bevor dieses Ergebnis erreicht wurde.
- Gelöst — Ob das Ergebnis behoben wurde () Yes/No
- Letzte Aktualisierung — Zeitstempel der letzten Aktualisierung

5. Erweitern Sie den Abschnitt Beschreibung, um Folgendes zu lesen:
   - Eine ausführliche Erläuterung des Sicherheitsproblems
   - Wie könnte die Sicherheitsanfälligkeit ausgenutzt werden
   - Die möglichen Auswirkungen auf Ihre Anwendung
   - Schritte zur Überprüfung, die der Mitarbeiter durchgeführt hat, um das Ergebnis zu bestätigen
6. Erweitern Sie den Abschnitt Codepositionen, um Folgendes anzuzeigen:
   - Spezifische Dateipfade, in denen das Problem identifiziert wurde
   - Eine kurze Beschreibung dessen, was an den einzelnen Speicherorten gefunden wurde
   - Liniennummernschilder, die auf den entsprechenden Code verweisen
7. Erweitern Sie den Abschnitt Beweise, um Folgendes zu überprüfen:
   - Der spezifische Code, die Konfiguration oder die Beobachtungen, die das Ergebnis stützen
   - Eine kurze Erklärung, warum jedes Element die Sicherheitsanfälligkeit aufzeigt
   - Die betroffenen Dateien und Zeilennummern für jedes Beweisstück
8. Erweitern Sie den Abschnitt Vorgeschlagene Lösung, um Folgendes anzuzeigen:
   - Die Änderungen, die der AWS Security Agent zur Behebung des Fehlers empfiehlt
   - Beispielcode oder Konfiguration für jede empfohlene Änderung

> **Tip**
>
> Verwenden Sie die Codepunkte, um schnell zu den betroffenen Dateien in Ihrem Repository zu navigieren. Jeder Speicherort enthält ausreichend Kontext, um das Problem zu verstehen, ohne die gesamte Datei lesen zu müssen.

Schritt 8: Priorisieren Sie die Ergebnisse und gehen Sie darauf ein

Überprüfen Sie die Ergebnisse und ergreifen Sie entsprechende Maßnahmen, um Sicherheitslücken zu beheben und den Sicherheitsstatus Ihrer Anwendung zu verbessern.

Für Ergebnisse mit hohem Schweregrad und hoher Zuverlässigkeit der Agenten:

1. Lesen Sie sich die Abschnitte Beschreibung und Codepositionen sorgfältig durch.
2. Verwenden Sie den Korrekturcode, um eine Pull-Anfrage mit einem Fix zu generieren, oder überprüfen Sie die PRs zur automatischen Problembehebung, falls Sie diese Option aktiviert haben.
3. Planen Sie eine nachfolgende Codeüberprüfung, um sicherzustellen, dass der Fix wirksam ist.

Bei Befunden mit mittlerem Schweregrad und hoher Zuverlässigkeit der Agenten gilt Folgendes:

1. Priorisieren Sie auf der Grundlage Ihrer Risikobereitschaft und Ihres Geschäftskontextes.
2. Nehmen Sie Problembehebungsaufgaben in Ihre reguläre Planung von Entwicklungs-Sprints auf.
3. Überlegen Sie, ob mehrere Befunde mit mittlerem Schweregrad zusammen ein höheres Risiko darstellen.

Für Ergebnisse mit mittlerem oder niedrigem Konfidenzgrad:

1. Überprüfen Sie die Abschnitte Beschreibung und Quellcodes, um die Annahmen und Bedingungen zu überprüfen, unter denen die Sicherheitsanfälligkeit auftritt.
2. Wenn die Sicherheitsanfälligkeit gültig ist, sollten Sie Prioritäten auf der Grundlage von Schweregrad und Risikotoleranz festlegen und Abhilfemaßnahmen in Betracht ziehen.

Schritt 9: Verfolgen Sie den Fortschritt der Problembehebung

Verwenden Sie die Oberfläche für die Ergebnisse und führen Sie Wiederholungen durch, um nachzuverfolgen, welche Sicherheitslücken behoben wurden.

1. Verwenden Sie bei der Implementierung von Problembehebungen die Option Resolve Finding, um die Ergebnisse als behoben zu kennzeichnen.
2. Führen Sie eine neue Codeüberprüfung durch, um sicherzustellen, dass Fixes die Ergebnisse beheben und keine neuen Probleme verursachen.
3. Sehen Sie sich die Tabelle „Alle Durchläufe“ auf der Detailseite der Codeüberprüfung an, um die Ergebnisse der einzelnen Durchläufe im Laufe der Zeit zu vergleichen.

> **Tip**
>
> Starten Sie nach dem Zusammenführen von Pull-Requests zur Problembehebung eine neue Ausführung desselben Code-Reviews, um sicherzustellen, dass die Sicherheitslücken behoben wurden. Vergleichen Sie die Anzahl der Ergebnisse zwischen den einzelnen Durchläufen, um Ihren Fortschritt zu verfolgen.

Nächste Schritte

Nachdem Sie die Ergebnisse Ihres Code-Reviews überprüft haben:

- Priorisieren Sie Ergebnisse mit hohem Schweregrad mit hoher Zuverlässigkeit, um sie sofort beheben zu können
- Verwenden Sie den Problembehebungscode, um automatische Problembehebungen zu generieren (siehe GitHub-sourced) the section called “Korrigieren Sie die Ergebnisse der Codeüberprüfung”
- Führen Sie nach der Implementierung von Fixes weitere Code-Reviews durch, um die Behebung zu überprüfen
- Passen Sie die Quellen oder Einstellungen für die Codeüberprüfung an, wenn sich Ihre Codebasis weiterentwickelt (siehe) the section called “Codeüberprüfung aktivieren”

Korrigieren Sie die Ergebnisse der Codeüberprüfung

Nachdem Sie die Sicherheitsergebnisse einer Codeüberprüfung überprüft haben, können Sie den AWS Security Agent verwenden, um Codekorrekturen für Ergebnisse in GitHub Repository-Quellen zu generieren. Für private Repositorys öffnet der AWS Security Agent eine Pull-Anfrage mit dem vorgeschlagenen Fix. Bei öffentlichen Repositorys fügt der AWS Security Agent dem Ergebnis einen herunterladbaren Diff bei, den Sie lokal anwenden können. Ergebnisse aus S3-Quellen müssen manuell berichtigt werden.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein abgeschlossener Code-Review-Lauf mit Ergebnissen
- GitHub Repositorys, die als Quelle für den Code-Review angeschlossen sind
- Zugriff auf die AWS Security Agent-Webanwendung
- Vertrautheit mit der Architektur und den Sicherheitsanforderungen Ihrer Anwendung

So funktioniert die Codekorrektur

Wenn Sie die Codekorrektur für ein Ergebnis auslösen, analysiert der AWS Security Agent das Ergebnis und seine Codespeicherorte und generiert dann eine Codekorrektur. Wie der Fix bereitgestellt wird, hängt von der Quelle ab:

- Private GitHub Repositorys — AWS Security Agent sendet eine Pull-Anfrage an das entsprechende Repository. Die Pull-Anfrage enthält eine Beschreibung des Sicherheitsproblems und der vorgenommenen Änderungen.
- Öffentliche GitHub Repositorys — Der AWS Security Agent fügt dem Ergebnis eine vorgeschlagene Abweichung bei, sodass die Sicherheitsanfälligkeit nicht veröffentlicht wird, bevor sie behoben ist. Sie können den Unterschied aus der Webanwendung herunterladen und ihn lokal mit anwenden. `git apply /path/to/code_remediation_changes.diff`
- S3-Quellen — Codekorrektur ist nicht verfügbar. Verwenden Sie die Beschreibung des Ergebnisses, die Codepositionen und die Risikoüberlegung, um Korrekturen manuell anzuwenden.

> **Important**
>
> Vom AWS Security Agent erstellte Pull-Anfragen sind für alle Benutzer sichtbar, die Lesezugriff auf das Repository haben. Überprüfen Sie die Änderungen vor dem Zusammenführen, um sicherzustellen, dass sie den Anforderungen Ihrer Anwendung entsprechen.

Automatische Codekorrektur

Wenn Sie bei der Erstellung der Codeüberprüfung die automatische Codekorrektur aktiviert haben, generiert AWS Security Agent Korrekturen für alle in Frage kommenden Ergebnisse, sobald die Überprüfung abgeschlossen ist. Sie müssen keine weiteren Maßnahmen ergreifen — Pull-Requests werden in Ihren verbundenen privaten GitHub Repositorys angezeigt, und nach Abschluss der Ausführung werden Unterschiede zu Ergebnissen aus öffentlichen GitHub Repositorys angezeigt.

Manuelle Codekorrektur

Wenn die automatische Codekorrektur deaktiviert ist, können Sie die Korrektur für einzelne Ergebnisse aus Quellen auslösen. GitHub

1. Navigieren Sie zur Registerkarte „Ergebnisse“ einer abgeschlossenen Codeüberprüfung.
2. Wählen Sie das Ergebnis aus, das Sie korrigieren möchten.
3. Wählen Sie im Bereich mit den Ergebnisdetails die Option Code korrigieren aus.
4. Der AWS Security Agent generiert einen Codefix und sendet entweder eine Pull-Anfrage an das Repository oder hängt je nach Sichtbarkeit des Repositorys einen herunterladbaren Diff an den Befund an.

Überprüfen Sie die Pull-Anfragen zur Behebung

Nachdem der AWS Security Agent eine Pull-Anfrage zur Behebung an ein privates GitHub Repository gesendet hat, gehen Sie wie folgt vor:

1. Navigieren Sie zu Ihrem Repository GitHub .
2. Suchen Sie die vom AWS Security Agent erstellte Pull-Anfrage.
3. Überprüfen Sie die Änderungen, einschließlich:

- Die Beschreibung, in der die Sicherheitslücke und die Problembehebung erläutert werden
- Der Code wurde geändert, um die Sicherheitsanfälligkeit zu beheben
- Jeder relevante Kontext zum Sanierungsansatz

4. Führen Sie den Pull-Request zusammen, falls der Fix angemessen ist, oder schließen Sie ihn und implementieren Sie eine alternative Lösung.

> **Tip**
>
> Starten Sie nach dem Zusammenführen der Korrektur-Pull-Requests einen neuen Code-Review-Lauf, um sicherzustellen, dass die Fixes die Ergebnisse beheben und keine neuen Sicherheitsprobleme mit sich bringen.

Wenden Sie die Abhilfemaßnahmen an

Wenden Sie bei Ergebnissen aus öffentlichen GitHub Repositorien den herunterladbaren Diff lokal an.

1. Laden Sie im Bereich mit den Suchergebnisdetails den Vergleich aus dem Abschnitt Codekorrektur herunter.
2. Führen `git apply /path/to/code_remediation_changes.diff` Sie in Ihrem lokalen Klon des Repositorys den Befehl aus.
3. Überprüfen Sie die vorgenommenen Änderungen, testen Sie sie anhand Ihrer Anwendung und übernehmen Sie sie im Rahmen Ihres normalen Entwicklungsworkflows.

Einschränkungen

- Die Codekorrektur ist nur für Ergebnisse aus GitHub Repository-Quellen verfügbar. Ergebnisse aus S3-Quellen müssen manuell berichtigt werden.
- AWS Security Agent generiert Korrekturen auf der Grundlage seiner Analyse der Sicherheitsanfälligkeit. Überprüfen Sie alle Änderungen, bevor Sie sie zusammenführen oder übernehmen, um sicherzustellen, dass sie für Ihre Anwendung geeignet sind.
- Bei einigen komplexen Ergebnissen ist möglicherweise manuelles Eingreifen erforderlich, das über die automatische Korrektur hinausgeht.

Nächste Schritte

Nach der Behebung der Ergebnisse:

- Überprüfe Pull-Requests in deinen GitHub Repositorys und füge sie zusammen, oder übertrage die angewendeten Diffs über deinen normalen Workflow
- Führe eine neue Codeüberprüfung durch, um die Korrekturen zu überprüfen und nach verbleibenden Problemen zu suchen
- Beheben Sie die Ergebnisse in der Webanwendung, nachdem Sie die Behebung bestätigt haben
- Passen Sie die Quellen oder Einstellungen für die Codeüberprüfung nach Bedarf an (siehethe section called “Codeüberprüfung aktivieren”)

Führen Sie einen Differenzcodescan mit S3 durch

Führen Sie einen differentiellen Codescan (Diff) durch, um nur die geänderten Zeilen in Ihrem Quellcode zu analysieren, anstatt einen vollständigen Repository-Scan durchzuführen. Differenzscans sind schneller als vollständige Scans und liefern Ergebnisse, die auf bestimmte Codeänderungen abzielen. Sie eignen sich daher ideal für die Validierung vor der Zusammenführung in Entwicklungsworkflows.

Im Gegensatz zur auf Pull-Requests basierenden Codeüberprüfung, die automatisch durch Ereignisse eines Drittanbieters für die Quellcodeverwaltung ausgelöst wird (siehethe section called “Überprüfen Sie die Ergebnisse zur Codesicherheit in Pull-Requests”), werden differenzielle Scans programmgesteuert über die AWS Security Agent API oder das SDK initiiert. Sie laden eine einheitliche Diff-Datei auf S3 hoch und verweisen darauf, wenn Sie einen Code-Review-Job starten.

Wie funktionieren Differenzscans

Ein Differenzscan analysiert Ihre Codeänderungen im gesamten Kontext Ihres Repositorys. Wenn Sie eine Ressource zur Codeüberprüfung mit Ihrem auf S3 hochgeladenen Quellcode erstellen, verwendet der Differenzscan das gesamte Repository als Kontext, wobei die Ergebnisse speziell auf die geänderten Zeilen in Ihrem Diff konzentriert werden. Auf diese Weise kann der Scan Sicherheitsprobleme identifizieren, die sich aus der Interaktion Ihrer Änderungen mit vorhandenem Code ergeben — wie z. B. fehlerhafte Authentifizierungsabläufe, unsichere Datenverarbeitung zwischen Modulen oder Änderungen, die bestehende Sicherheitslücken aufdecken.

Der Scanvorgang:. Sie laden eine einheitliche Diff-Datei (die Ausgabe von`git diff`) in einen S3-Bucket hoch, der mit Ihrem Agent Space verbunden ist. Sie rufen die `StartCodeReviewJob`

API mit einem `diffSource` Parameter auf, der auf den S3-Speicherort Ihres Diffs verweist. Der AWS Security Agent analysiert nur den geänderten Code im Vergleich auf Sicherheitslücken und die Einhaltung der Sicherheitsanforderungen Ihres Unternehmens. Die Ergebnisse werden anhand der geänderten Zeilen mit Schweregradeinstufungen, Codepositionen und Anleitungen zur Problembehebung zusammengefasst.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein Agentenbereich mit aktivierter Codeüberprüfung (siehe) the section called “Codeüberprüfung aktivieren”
- Eine Code-Review-Ressource, die bereits für das Ziel-Repository erstellt wurde, wobei der vollständige Quellcode in den S3-Bucket hochgeladen wurde, der mit Ihrem Agent Space verbunden ist. Das vollständige Repository bietet Kontext, der eine tiefere Analyse der Interaktion Ihrer Änderungen mit vorhandenem Code ermöglicht. Anweisungen the section called “Einen Code-Review erstellen” zum Erstellen eines Code-Reviews und zum Hochladen von Quellcode finden Sie unter.
- Ein S3-Bucket, der mit Ihrem Agent Space verbunden ist
- IAM-Berechtigungen zum Hochladen in den S3-Bucket und zum Aufrufen `securityagent:StartCodeReviewJob`
- Eine einheitliche Diff-Datei, die aus Ihren Codeänderungen generiert wurde (z. B. der Ausgabe von`git diff main..feature-branch`)

> **Tip**
>
> Um möglichst genaue Ergebnisse zu erzielen, stellen Sie sicher, dass Ihre Code-Review-Ressource die neueste Version Ihres vollständigen Quellcodes auf S3 hochgeladen hat. Der Diff-Scan verwendet dieses vollständige Repository als Kontext, um Ihre Anwendungsarchitektur, Datenflüsse und bestehende Sicherheitskontrollen bei der Analyse der geänderten Zeilen zu verstehen.

Schritt 1: Generieren Sie eine Diff-Datei

Generieren Sie ein einheitliches Diff, das die Codeänderungen darstellt, die Sie scannen möchten.

```
git diff main..feature-branch > changes.diff
```

Alternativ können Sie einen Diff für stufenweise vorgenommene Änderungen generieren:

```
git diff --cached > changes.diff
```

> **Tip**
>
> Die Diff-Datei sollte im einheitlichen Standarddiff-Format vorliegen. Der AWS Security Agent analysiert die Dateipfade und geänderten Zeilen aus den Diff-Headern, um den Umfang seiner Analyse zu ermitteln.

Schritt 2: Laden Sie den Diff auf S3 hoch

Laden Sie die Diff-Datei in einen S3-Bucket hoch, der mit Ihrem Agent Space verbunden ist.

```
aws s3 cp changes.diff s3://my-security-agent-bucket/diffs/changes.diff
```

> **Important**
>
> Bei dem S3-Bucket muss es sich um einen Bucket handeln, der bereits mit Ihrem Agent Space verbunden ist. Die Ihrem Agent Space zugeordnete IAM-Servicerolle muss Lesezugriff auf diesen Standort haben. Anweisungen the section called “Codeüberprüfung aktivieren” zum Verbinden von S3-Buckets finden Sie unter.

Schritt 3: Starten Sie einen Job zur Überprüfung des Differentialcodes

Rufen Sie die `StartCodeReviewJob` API mit dem `diffSource` Parameter auf, um einen Differenzscan zu initiieren.

Verwenden der AWS CLI

```
aws securityagent start-code-review-job \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --diff-source '{"s3Uri": "s3://my-security-agent-bucket/diffs/changes.diff"}'
```

Verwenden des AWS-SDK (Python)

```
import boto3

client = boto3.client('securityagent')

response = client.start_code_review_job(
    agentSpaceId='your-agent-space-id',
    codeReviewId='your-code-review-id',
    diffSource={
        's3Uri': 's3://my-security-agent-bucket/diffs/changes.diff'
    }
)

print(f"Job started: {response['codeReviewJobId']}")
```

Verwenden des AWS-SDK (JavaScript/TypeScript)

```
import { SecurityAgentClient, StartCodeReviewJobCommand } from '@aws-sdk/client-
securityagent';

const client = new SecurityAgentClient({ region: 'us-east-1' });

const response = await client.send(new StartCodeReviewJobCommand({
  agentSpaceId: 'your-agent-space-id',
  codeReviewId: 'your-code-review-id',
  diffSource: {
    s3Uri: 's3://my-security-agent-bucket/diffs/changes.diff',
  },
}));

console.log(`Job started: ${response.codeReviewJobId}`);
```

Die API kehrt sofort mit einem `codeReviewJobId` und `status` von zurück`IN_PROGRESS`.

Schritt 4: Überwachen Sie den Scan

Fragen Sie den Status des Code-Review-Jobs ab, bis er abgeschlossen ist.

```
aws securityagent get-code-review-job \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
```

```
  --code-review-job-id "the-job-id"
```

Der Job durchläuft dieselben Phasen wie eine vollständige Codeüberprüfung: Preflight → Statische Analyse → Finalisieren. Differenzscans werden in der Regel schneller abgeschlossen als vollständige Scans, da sie weniger Codezeilen analysieren.

Schritt 5: Überprüfen Sie die Ergebnisse

Nachdem der Job abgeschlossen ist, listen Sie die Ergebnisse für den Code-Review-Job auf.

```
aws securityagent list-findings \
  --agent-space-id "your-agent-space-id" \
  --code-review-id "your-code-review-id" \
  --code-review-job-id "the-job-id"
```

Jedes Ergebnis beinhaltet:

- Eine Beschreibung des Sicherheitsproblems
- Der Schweregrad (Kritisch, Hoch, Mittel oder Niedrig)
- Code-Positionen, die auf bestimmte Zeilen im Diff verweisen
- Risikoüberlegung zur Erläuterung der möglichen Auswirkungen
- Anleitung zur Problembehebung

Weitere Informationen dazu, wie Sie die Ergebnisse verstehen und entsprechend handeln können, finden Sie unterthe section called “Ergebnisse eines Code-Reviews überprüfen”.

Kontingente und -Einschränkungen

- Maximale Anzahl geänderter Dateien in einem Diff: 3.000 Dateien oder 5 MB Gesamtgröße des Diffs
- Die Ergebnisse sind auf 30 pro Scan begrenzt, priorisiert nach Schweregrad (Kritisch > Hoch > Mittel > Niedrig)

Nächste Schritte

Nach der Ausführung eines Differenzscans:

- Überprüfen Sie die Ergebnisse und wenden Sie die Hinweise zur Problembehebung an (siehethe section called “Ergebnisse eines Code-Reviews überprüfen”)
- Aktivieren Sie Kommentare zu Pull-Requests für die automatische Codeüberprüfung bei jeder Pull-Anfrage (siehethe section called “Pull-Request-Code-Review für GitHub Repositorys aktivieren”)
- Führen Sie regelmäßig eine vollständige Codeüberprüfung durch, um Probleme zu erkennen, die nicht auf einzelne Änderungen zurückzuführen sind (siehethe section called “Einen Code-Review erstellen”)
- Verwenden Sie die IDE-Integration, um Differenzscans direkt von Ihrer Entwicklungsumgebung aus auszuführen (siehethe section called “Führen Sie Code-Sicherheitsscans von Ihrer IDE aus aus”)

Führen Sie Code-Sicherheitsscans von Ihrer IDE aus aus

Führen Sie mithilfe von Kiro oder Claude Code die Code-Sicherheitsscans von AWS Security Agent direkt von Ihrer IDE aus aus. Mit der IDE-Integration können Sie Ihren lokalen Quellcode auf Sicherheitslücken scannen, Differenzscans nur für geänderten Code ausführen und Entwurfsdokumente anhand von Bedrohungsmodellen überprüfen — und das alles, ohne Ihre Entwicklungsumgebung verlassen zu müssen. Die Ergebnisse werden zusammen mit Ihrem Code zusammen mit Anleitungen zur Problembehebung angezeigt, und Sie können automatische Korrekturen direkt von der IDE aus anwenden.

So funktioniert die IDE-Integration

Die IDE-Integration verwendet den AWS Security Agent MCP (Model Context Protocol) -Server, um Ihre IDE mit dem AWS Security Agent-Service zu verbinden:

1. Der MCP-Server packt Ihren Quellcode in ein ZIP-Archiv.
2. Das Archiv wird in einen S3-Bucket hochgeladen, den der Server automatisch in Ihrem AWS-Konto bereitstellt.
3. Der Server ruft die AWS Security Agent API auf, um einen Scan zu starten.
4. AWS Security Agent analysiert den Code auf Sicherheitslücken und die Einhaltung der Sicherheitsanforderungen Ihres Unternehmens.
5. Die Ergebnisse werden mit Codepositionen, Beschreibungen, Schweregradbewertungen und Vorschlägen zur Problembehebung an die IDE zurückgesendet.

Der MCP-Server kümmert sich um die gesamte Orchestrierung — Bereitstellung von Agentenspeicher, S3-Bucket-Erstellung, IAM-Rolleneinrichtung, Codepaketierung und Scanabfrage — sodass Sie nur lokal konfigurierte AWS-Anmeldeinformationen benötigen.

> **Note**
>
> Die einzige Voraussetzung sind AWS-Anmeldeinformationen, die in Ihrer lokalen Umgebung konfiguriert sind (z. `aws configure` B. über AWS SSO oder Umgebungsvariablen). Der MCP-Server stellt den Agent Space, die IAM-Dienstrolle und den S3-Bucket bei der ersten Verwendung automatisch bereit.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Lokal konfigurierte AWS-Anmeldeinformationen mit den folgenden Berechtigungen:
  - `iam:CreateRole`, `iam:PutRolePolicy` (für die einmalige Einrichtung der Servicerolle)
  - `s3:CreateBucket`, `s3:PutObject`, `s3:PutPublicAccessBlock`, `s3:PutLifecycleConfiguration`
  - `securityagent:CreateAgentSpace`, `securityagent:UpdateAgentSpace`, `securityagent:ListAgentSpaces`, `securityagent:BatchGetAgentSpaces`
  - `securityagent:CreateCodeReview`, `securityagent:StartCodeReviewJob`, `securityagent:BatchGetCodeReviewJobs`
  - `securityagent:ListFindings`, `securityagent:BatchGetFindings`
  - `securityagent:StartCodeRemediation`(für automatisierte Korrekturen)
  - `sts:GetCallerIdentity`
- uv installiert (Python-Paketläufer)
- Python 3.10 oder höher
- Eine der folgenden IDEs:
  - Kiro (installieren Sie den AWS Security Agent Power)
  - Claude Code (installieren Sie das AWS Security Agent-Plugin)

Installieren Sie den MCP-Server

Kiro

Installieren Sie den AWS Security Agent Power vom Kiro Marketplace. The Power bündelt die MCP-Serverkonfiguration, Steuerungsanweisungen und Lifecycle-Hooks für automatische Sicherheitsscans.

Alternativ können Sie den MCP-Server manuell in Ihrem Projekt konfigurieren: `.kiro/mcp.json`

```
{
  "mcpServers": {
    "security-agent": {
      "command": "uvx",
      "args": ["awslabs.security-agent-mcp-server@latest"],
      "env": {
        "AWS_PROFILE": "default",
        "AWS_REGION": "us-east-1",
        "FASTMCP_LOG_LEVEL": "ERROR"
      }
    }
  }
}
```

Claude Code

Installieren Sie das AWS Security Agent-Plug-In, das Fähigkeiten zur Einrichtung, vollständigen Scans, Vergleichsscans, Überprüfung von Bedrohungsmodellen und Problembehebung bietet.

Konfigurieren Sie den MCP-Server in Ihrem Projekt: `.mcp.json`

```
{
  "mcpServers": {
    "awslabs.security-agent-mcp-server": {
      "command": "uvx",
      "args": ["awslabs.security-agent-mcp-server@latest"],
      "env": {
        "AWS_PROFILE": "default",
        "AWS_REGION": "us-east-1",
        "FASTMCP_LOG_LEVEL": "ERROR"
      }
```

```
    }
  }
}
```

> **Tip**
>
> Sie können den MCP-Server auch über Docker ausführen, wenn Sie Python nicht lokal installieren möchten. Informationen zur Docker-Konfiguration finden Sie in der MCP-Serverdokumentation.

First-time einrichten

Bei der ersten Verwendung stellt der MCP-Server automatisch die erforderlichen AWS-Ressourcen bereit:

Ressource	Namenskonvention	Zweck
Agentenraum	User-chosen oder automatisch erstellt	Container für Scans, Bewertungen und Pentests
IAM-Dienstrolle	`SecurityAgentScanRole`	Wird davon ausgegangen, den hochgeladenen Code `securityagent.amazonaws.com` zu lesen
S3-Bucket	`security-agent-scans-{account}-{region}`	Speichert komprimierten Quellcode (30-tägiger Lebenszyklus mit automatischem Ablauf)

Um das Setup explizit auszulösen, fragen Sie Ihre IDE:

```
Set up the security agent
```

Das Setup überprüft Ihre AWS-Anmeldeinformationen, erstellt oder verwendet einen Agent Space, stellt die IAM-Rolle und den S3-Bucket bereit und bestätigt die Bereitschaft. Wenn Sie bereits über

Agent Spaces aus der AWS-Managementkonsole verfügen, werden diese im Setup angezeigt und Sie werden gefragt, welcher verwendet werden soll.

> **Note**
>
> Das Setup wird automatisch bei Ihrem ersten Scan ausgeführt, sofern es nicht bereits konfiguriert ist. Sie müssen es nicht separat ausführen.

Führen Sie einen vollständigen Sicherheitsscan durch

Scannen Sie Ihr gesamtes Projekt auf Sicherheitslücken:

```
Scan this project for security issues
```

Die IDE:

1. Archiviert Ihren Quellcode (mit Ausnahme `.git` von Build-Artefakten und anderen unwichtigen Verzeichnissen) `node_modules`
2. Lädt das Archiv auf S3 hoch
3. Startet einen vollständigen Code-Review-Job
4. Umfragen müssen abgeschlossen werden (in der Regel etwa eine Stunde)
5. Stellt die Ergebnisse nach Schweregrad gruppiert dar

Fortschritt des Scans

Vollständige Scans dauern je nach Größe der Codebasis in der Regel etwa 1 Stunde. Die IDE überprüft den Fortschritt alle 5 Minuten und benachrichtigt Sie, wenn die Ergebnisse vorliegen. Sie können jederzeit nach dem Status fragen:

```
How's the scan going?
```

Führen Sie einen Differenzscan aus

Scannen Sie für schnelleres Feedback während der Entwicklung nur den Code, der sich seit einer Git-Referenz geändert hat:

```
Scan my changes against main for security issues
```

Wenn Sie keine Basisreferenz angeben, werden beim Scan standardmäßig Ihre noch nicht festgeschriebenen Änderungen mit denen verglichen. `HEAD` Sie können explizit eine andere Basis angeben:

```
Diff scan my uncommitted changes
```

Differenzscans laden sowohl den vollständigen Repository-Kontext als auch den Git-Diff-Patch hoch und führen dann eine Analyse durch, die sich nur auf die geänderten Zeilen konzentriert. Die Ergebnisse liegen in der Regel innerhalb von 5—15 Minuten vor.

Weitere Informationen zur S3-Diff-Scan-API finden Sie unter. the section called “Führen Sie einen Differenzcodescan mit S3 durch”

Führen Sie eine Überprüfung des Bedrohungsmodells durch

Analysieren Sie mithilfe der STRIDE-Methode Entwurfsdokumente auf Änderungen des Sicherheitsstatus:

```
Run a threat model review on my spec
```

Die IDE identifiziert Ihre `requirements.md` `design.md` Dateien (in der Regel unter`.kiro/specs/`), lädt sie zusammen mit Ihrem Quellcode hoch und führt eine Analyse des Bedrohungsmodells durch. Die Ergebnisse identifizieren:

- Sicherheitsbedrohungen, kategorisiert nach STRIDE (Spoofing, Manipulation, Ablehnung, Offenlegung von Informationen, Denial-of-Service, Erhöhung von Rechten)
- Schweregradeinstufungen für jede Bedrohung
- Auswirkungen auf bestimmte Ressourcen in Ihrer Codebasis
- Empfehlungen zur Schadensbegrenzung

> **Tip**
>
> Führen Sie eine Überprüfung des Bedrohungsmodells durch, bevor Sie Implementierungsaufgaben anhand einer Spezifikation generieren. Dadurch werden Probleme beim Sicherheitsdesign erkannt, bevor der Code geschrieben wird.

Überprüfen der Erkenntnisse

Wenn ein Scan abgeschlossen ist, werden die Ergebnisse in Ihrer IDE nach Schweregrad gruppiert angezeigt:

```
# CRITICAL: SQL Injection in user-service.ts
   File: src/api/user-service.ts:45
   User input flows directly into SQL query without parameterization

# HIGH: Hardcoded credentials in config.ts
   File: src/config/database.ts:12
   Database password stored in source code
```

Außerdem wird ein detaillierter Bericht `.security-agent/findings-{scan_id}.md` mit vollständigen Informationen zu jedem Ergebnis erstellt, einschließlich Risikotyp, Vertrauensbewertung, Codepositionen und Abhilfemaßnahmen.

So zeigen Sie die Ergebnisse eines früheren Scans an:

```
Show my security findings
```

Wenden Sie automatische Korrekturen an

Wenden Sie nach der Überprüfung der Ergebnisse die automatische Behebung an:

```
Fix the security findings
```

Die IDE arbeitet die Ergebnisse nach Schweregrad von oben nach unten ab (Kritisch → Hoch → Mittel → Niedrig) und wendet Codekorrekturen an, wobei die für jedes Ergebnis geltenden Anleitungen zur Problembehebung verwendet werden. Nachdem alle Fixes angewendet wurden, führt sie automatisch einen Verifizierungs-Vergleichsscan durch, um zu bestätigen, dass die Probleme behoben wurden.

Sie können auch einzelne Ergebnisse korrigieren:

```
Fix the SQL injection in user-service.ts
```

> **Important**
>
> Überprüfe immer die automatisierten Korrekturen, bevor du sie festlegst. Stellen Sie sicher, dass die Korrekturen keine bestehenden Funktionen beeinträchtigen oder zu Regressionen führen.

Automatische Scanvorschläge (Kiro)

Wenn Sie Kiro Power verwenden, kann der AWS Security Agent automatisch Diff-Scans vorschlagen, nachdem Sie sicherheitsrelevante Codeänderungen vorgenommen haben. Der Power installiert einen Hook, der jede abgeschlossene Codierungsrunde bewertet und einen Scan vorschlägt, wenn sich Änderungen auf Folgendes auswirken:

- Authentifizierungs- oder Autorisierungslogik
- Kryptografie oder Umgang mit Geheimnissen
- Überprüfung der Eingabe oder Datenbereinigung
- Netzwerkanfragen oder externe Integrationen
- Sicherheitskonfiguration (CORS, Header, Sitzungsverwaltung)

Sie können automatische Vorschläge jederzeit deaktivieren, indem Sie sie löschen. `.kiro/hooks/security-diff-scan-suggester.kiro.hook`

Verfügbare Scantypen

Scan-Typ	Dauer	Anwendungsfall	Befehl
Vollständiger Scan	Etwa 1 Stunde	Umfassende Überprüfung der gesamten Codebasis	„Mein Projekt auf Sicherheitsprobleme scannen“
Diff-Scan	5—15 Minuten	Nur geänderter Code (Pre-Commit, Pre-PR)	„Meine Änderungen scannen“ oder „Diff Scan gegen Main“

Scan-Typ	Dauer	Anwendungsfall	Befehl
Bedrohungsmodell	5—30 Minuten	Designdokumente (requirements.md, design.md)	„Führen Sie eine Überprüfung des Bedrohungsmodells anhand meiner Spezifikation durch“

Umgebungsvariablen

Variable	Description	Standard
`AWS_REGION`	AWS-Region für SecurityAgent API-Aufrufe	`us-east-1`
`AWS_PROFILE`	Name des AWS-Anmeldeinformationsprofils	Standardprofil
`FASTMCP_LOG_LEVEL`	Protokollebene des MCP-Servers (DEBUG, INFO, WARNING, ERROR)	`WARNING`

Fehlerbehebung

„Nicht konfiguriert. Führen Sie zuerst das Setup aus.“

Ihr `.security-agent/config.json` fehlt oder der Agent Space ist nicht mehr vorhanden. Bitten Sie die IDE, das Setup auszuführen:

```
Set up the security agent
```

Der Scan schlägt fehl mit AccessDenied

Stellen Sie sicher, dass Ihre AWS-Anmeldeinformationen über die erforderlichen IAM-Berechtigungen verfügen, die im Abschnitt Voraussetzungen aufgeführt sind. Die am häufigsten fehlende Berechtigung ist `iam:CreateRole` (wird nur für die erstmalige Einrichtung benötigt).

Der Scan läuft ab oder dauert zu lange

- Vollständige Scans großer Codebasen können mehr als 1 Stunde dauern
- Verwenden Sie Differenzscans für die iterative Entwicklung — sie sind innerhalb von Minuten abgeschlossen
- Stellen Sie sicher, dass Ihr Quellarchiv keine unnötigen Verzeichnisse enthält (der MCP-Server schließt gängige Muster automatisch aus)

Leerer Unterschied — keine Änderungen beim Scannen

Wenn Sie einen Diff-Scan ohne noch nicht festgeschriebene Änderungen ausführen, meldet der MCP-Server „Keine Änderungen im Vergleich zu HEAD“ und startet keinen Scan. Übernehmen oder speichern Sie Ihre Änderungen zuerst, oder geben Sie eine andere Basisreferenz an.

Kontingente und -Einschränkungen

- Maximale Größe des Quellarchivs: 2 GB
- S3-Bucket-Lebenszyklus: Hochgeladener Code wird nach 30 Tagen automatisch gelöscht

Nächste Schritte

Nachdem Sie Ihren ersten IDE-Sicherheitsscan ausgeführt haben:

- Aktivieren Sie Kommentare zur Überprüfung von Pull-Request-Codes für die automatische GitHub Integration (siehethe section called “Pull-Request-Code-Review für GitHub Repositorys aktivieren”)
- Konfigurieren Sie die Sicherheitsanforderungen für die organisationsspezifische Richtlinienvalidierung (siehe) the section called “Sicherheitsanforderungen verwalten”
- Führen Sie regelmäßig vollständige Scans durch catch um Probleme in Ihrer gesamten Codebasis zu erkennen (siehethe section called “Einen Code-Review erstellen”)
- Prüfen Sie vor der Implementierung neue Funktionsspezifikationen anhand von Bedrohungsmodellen

Erstellen Sie einen Penetrationstest

Richten Sie automatisierte Penetrationstests für Ihre Webanwendungen ein, indem Sie den Testumfang, die Zieldomänen und den AWS-Ressourcenzugriff konfigurieren. Penetrationstests

helfen dabei, Sicherheitslücken in laufenden Anwendungen zu identifizieren, indem reale Angriffsszenarien gegen Ihre verifizierten Domains simuliert werden.

AWS Security Agent führt auf der Grundlage des konfigurierten Bereichs und der Berechtigungen umfassende Sicherheitstests für Ihre Webanwendungen durch und liefert detaillierte Erkenntnisse über ausnutzbare Sicherheitslücken, bevor Angreifer sie entdecken können.

In diesem Verfahren erstellen Sie einen Penetrationstest, indem Sie die Testdetails konfigurieren, den Testumfang definieren und die erforderlichen Berechtigungen einrichten.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Zugriff auf die AWS Security Agent-Webanwendung
- Mindestens eine verifizierte Domain zum Testen
- IAM-Rolle mit entsprechenden Berechtigungen für AWS Security Agent
- Verständnis der Architektur und der kritischen Pfade Ihrer Anwendung

Beginnen Sie mit der Erstellung eines Penetrationstests

Navigieren Sie in der Agent Web App zur Seite zur Erstellung von Penetrationstests.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Navigieren Sie zum Abschnitt Penetrationstests.
3. Wählen Sie Penetrationstest erstellen.

> **Tip**
>
> Nur verifizierte Domains können in Penetrationstests einbezogen werden. Bitten Sie Ihren Administrator, die Domain in der AWS-Managementkonsole zu verifizieren. Siehe the section called “Aktivieren Sie eine Anwendungsdomäne für Penetrationstests”.

Benennen Sie Ihren Penetrationstest

Geben Sie einen aussagekräftigen Namen an, anhand dessen Zweck und Umfang dieses Penetrationstests ermittelt werden können.

1. Geben Sie im Feld Name des Penetrationstests einen aussagekräftigen Namen für Ihren Penetrationstest ein.

   Example

   Der Name sollte die Anwendung, Umgebung oder Komponente, die getestet wird, eindeutig identifizieren. Maximal 100 Zeichen.

Umfang des Penetrationstests konfigurieren

Definieren Sie, welche Domänen und URL-Pfade getestet werden sollen, und konfigurieren Sie optionale Ausnahmen, um die Testgrenzen zu kontrollieren.

Fügen Sie Zieldomänen hinzu

Geben Sie die verifizierten Domains an, die aktiv auf Sicherheitslücken getestet werden.

1. Suchen Sie im Abschnitt Umfang des Penetrationstests nach Ziel-URLs.
2. Erweitern Sie den Abschnitt Verifizierte Domains, um die verfügbaren Domains anzuzeigen.
3. Geben Sie im Feld Ziel-URL eine Zieldomänen-URL ein.

   > **Important**
   >
   > Nur verifizierte Domains können getestet werden. Die URL muss sich unter einer Domain befinden, die Sie zuvor in AWS Security Agent verifiziert haben. Sub-domains für eine verifizierte Domain ist keine separate Überprüfung erforderlich.

4. Um mehrere Zieldomänen hinzuzufügen:
   a. Wählen Sie Domain hinzufügen.
   b. Geben Sie jede weitere Domain-URL ein.
5. Um eine Zieldomain zu entfernen, wählen Sie neben der Domain-URL die Option Entfernen aus.

> **Tip**
>
> Um optimale Ergebnisse zu erzielen, sollten Sie alle Domänen einbeziehen, die Teil des Benutzerflusses Ihrer Anwendung sind, einschließlich Subdomänen für APIs, Authentifizierungsdienste und Inhaltsbereitstellung. Sub-domains für eine verifizierte übergeordnete Domain ist keine separate Überprüfung erforderlich.

Risikoarten ausschließen (optional)

Wählen Sie bestimmte Risikokategorien aus, die von den Tests ausgeschlossen werden sollen, wenn sie für Ihre Anwendung nicht zutreffen.

1. Suchen Sie das Feld Risikotypen ausschließen.
2. Wählen Sie das Drop-down-Menü, um die verfügbaren Risikotypen anzuzeigen.
3. Wählen Sie einen oder mehrere Risikotypen aus, die vom Penetrationstest ausgeschlossen werden sollen.

> **Note**
>
> Der Ausschluss von Risikoarten schränkt den Umfang der Tests ein. Schließen Sie nur Risikotypen aus, die für Ihre Anwendung nicht relevant sind oder die Sie separat testen möchten.

Fügen Sie URL-Pfade hinzu, die außerhalb des Geltungsbereichs liegen (optional)

Geben Sie URL-Pfade an, die während des Penetrationstests nicht getestet werden sollen. AWS Security Agent schließt den angegebenen Pfad und alle darunter verschachtelten Pfade aus. Wenn Sie beispielsweise eine URL hinzufügen`https://example.com/admin`, die außerhalb des Gültigkeitsbereichs liegt, `https://example.com/admin/tools` liegt sie ebenfalls außerhalb des Gültigkeitsbereichs.

1. Suchen Sie den Abschnitt URLs. Out-of-scope
2. Geben Sie im Eingabefeld Out-of-scope URLs einen URL-Pfad ein, der ausgeschlossen werden soll (z. B. `/admin/delete` oder`/api/reset`).

> **Warning**
>
> Out-of-scope Pfade werden nicht auf Sicherheitslücken getestet. Stellen Sie sicher, dass Sie nur Pfade ausschließen, auf die während des Tests nicht zugegriffen werden sollte, z. B. bei destruktiven Vorgängen oder sensiblen Verwaltungsfunktionen.

3. So fügen Sie mehrere Pfade hinzu, die außerhalb des Gültigkeitsbereichs liegen:
   a. Wählen Sie URL hinzufügen.
   b. Geben Sie jeden weiteren Pfad ein.
4. Um einen Pfad zu entfernen, wählen Sie neben dem Pfad die Option Entfernen aus.

Fügen Sie zugängliche Domänen hinzu (optional)

Geben Sie Domänen an, die für den Test erforderlich sind, aber keine Ziele für Schwachstellentests sind.

1. Suchen Sie den Abschnitt Zugängliche URLs.
2. Geben Sie im Eingabefeld Barrierefreie URLs eine Domain ein, auf die während des Tests zugegriffen werden soll.

> **Note**
>
> Fügen Sie zugängliche Domains für Dienste von Drittanbietern (wie Okta, Auth0, Stripe) hinzu, die sich außerhalb Ihrer Zieldomain befinden. Dies ist erforderlich, damit der AWS Security Agent während des Tests auf diese URLs für die Anmeldung und Navigation zugreifen kann. AWS Security Agent führt KEINE Penetrationstests für diese Domains durch — sie werden ausschließlich für Zugriffszwecke verwendet. Zugängliche Domains erfordern keine Überprüfung der Inhaberschaft, auch wenn sie zu einer anderen Domain als Ihrer Zieldomain gehören. Nur Zieldomains erfordern eine verifizierte Inhaberschaft.

3. Um mehrere zugängliche Domains hinzuzufügen:
   a. Wählen Sie URL hinzufügen.
   b. Geben Sie jede weitere Domain ein.
4. Um eine Domain zu entfernen, wählen Sie neben der Domain die Option Entfernen aus.

Fügen Sie benutzerdefinierte HTTP-Header hinzu (optional)

Geben Sie benutzerdefinierte HTTP-Header an, die allen Anfragen des AWS Security Agent während der Penetrationstests hinzugefügt werden.

1. Suchen Sie den Abschnitt Benutzerdefinierte HTTP-Header.
2. Geben Sie im Eingabefeld Benutzerdefinierte HTTP-Header einen Header-Namen und einen Header-Wert ein, die ausgehenden Anfragen zugeordnet werden.

> **Note**
>
> Standardmäßig fügt der AWS Security Agent einen benutzerdefinierten Header für User-Agentset to securityagent hinzu, sofern kein anderer benutzerdefinierter User-AgentHeader-Wert angegeben ist.

3. Um mehrere benutzerdefinierte Header hinzuzufügen:
   a. Wählen Sie Add header.
   b. Geben Sie jeden zusätzlichen benutzerdefinierten Header ein.
4. Um eine benutzerdefinierte Kopfzeile zu entfernen, wählen Sie neben der Kopfzeile die Option Entfernen aus.

Konfigurieren Sie die IAM-Rolle

Wählen Sie die vorkonfigurierte Servicerolle für diesen Penetrationstest aus. AWS Security Agent verwendet ein Space-based Agent-Berechtigungsmodell, bei dem Administratoren bei der Einrichtung Ihres Agent Space IAM-Rollen konfigurieren. Sie wählen aus Rollen aus, die bereits konfiguriert und einsatzbereit sind.

1. Suchen Sie im Abschnitt Berechtigungen nach der Dropdownliste Servicerollen.
2. Wählen Sie die IAM-Rolle aus, die dem AWS Security Agent Zugriff auf die erforderlichen AWS-Ressourcen gewährt.

> **Important**
>
> Die ausgewählte IAM-Rolle muss über Berechtigungen für den Zugriff auf VPC-Ressourcen, CloudWatch Protokolle und alle anderen AWS-Services verfügen, die für

den Penetrationstest benötigt werden. Stellen Sie sicher, dass die Rolle die richtige Vertrauensbeziehung mit dem AWS Security Agent hat.

3. Suchen Sie das Drop-down-Menü für die CloudWatch Protokollgruppe.
4. Wählen Sie die Protokollgruppe aus, in der die Protokolle der Penetrationstests gespeichert werden sollen. (optional)

> **Note**
>
> Die ausgewählte CloudWatch Protokollgruppe speichert detaillierte Protokolle der Ausführung des Penetrationstests, einschließlich der gestellten Anfragen, der eingegangenen Antworten und der entdeckten Sicherheitslücken.
> Wenn Sie keine Protokollgruppe auswählen, wird automatisch eine neue CloudWatch Protokollgruppe mit dem `/aws/securityagent` Präfix zum Speichern der Penetrationstest-Protokolle erstellt.

Automatische Codekorrektur

Markieren Sie das Kontrollkästchen Automatische Korrektur aktivieren.

> **Important**
>
> Um Sicherheitslücken in Ihren Quellcode-Repositorys zu korrigieren, kann der AWS Security Agent Pull-Anfragen an Ihre Repositorys senden. Die Pull-Requests sind möglicherweise für alle Benutzer sichtbar, die Lesezugriff auf die Repositorys haben.

VPC-Ressourcen konfigurieren (optional)

Wenn Ihre Zieldomänen privat sind und in einer VPC gehostet werden, konfigurieren Sie die VPC-Einstellungen, in denen der AWS Security Agent Penetrationstests durchführen soll. Dieser Schritt ist nur für Anwendungen erforderlich, die nicht öffentlich zugänglich sind.

> **Note**
>
> Überspringen Sie diesen Schritt, wenn Ihre Zieldomänen öffentlich zugänglich sind. Die VPC-Konfiguration ist nur zum Testen von privaten Anwendungen erforderlich, die in einer Amazon Virtual Private Cloud gehostet werden.

Wählen Sie die VPC, Subnetze und Sicherheitsgruppen für die Penetrationstestumgebung aus.

1. Suchen Sie im Abschnitt VPC nach der Dropdownliste VPC-ID.
2. Wählen Sie die VPC aus, in der Ihre Zieldomänen gehostet werden.

> **Important**
>
> Die ausgewählte VPC muss die Zieldomänen enthalten, die Sie in Schritt 3 angegeben haben. Stellen Sie sicher, dass die VPC über eine angemessene Routing- und Netzwerkkonfiguration verfügt, damit der AWS Security Agent auf Ihre Anwendungen zugreifen kann.

3. Suchen Sie das Drop-down-Menü Subnetze.
4. Wählen Sie ein oder mehrere Subnetze aus, in denen der Penetrationstest ausgeführt werden soll.

> **Note**
>
> Wählen Sie Subnetze aus, die Netzwerkzugriff auf Ihre Zielanwendungen haben. Der Penetrationstest wird von Ressourcen aus ausgeführt, die in diesen Subnetzen bereitgestellt werden.

5. Suchen Sie das Drop-down-Menü Sicherheitsgruppe.
6. Wählen Sie die Sicherheitsgruppe aus, die den Netzwerkzugriff für den Penetrationstest steuert.

> **Important**
>
> Die ausgewählte Sicherheitsgruppe muss ausgehenden Datenverkehr zu Ihren Zieldomänen und allen zugänglichen Domänen zulassen. Stellen Sie sicher, dass die Sicherheitsgruppenregeln den erforderlichen Netzwerkzugriff für umfassende Tests zulassen.

Konfigurieren Sie die Anmeldeinformationen für die Authentifizierung (optional)

Wenn für Ihre Zieldomänen eine Authentifizierung erforderlich ist, geben Sie Anmeldeinformationen ein, damit der AWS Security Agent während der Penetrationstests auf geschützte Bereiche Ihrer Anwendung zugreifen kann. Dieser Schritt ist nur für Anwendungen erforderlich, für die eine Benutzerauthentifizierung erforderlich ist.

> **Note**
>
> Überspringen Sie diesen Schritt, wenn für Ihre Zieldomänen keine Authentifizierung erforderlich ist oder wenn alle Bereiche, die Sie testen möchten, öffentlich zugänglich sind. Konfigurieren Sie Anmeldeinformationen nur, wenn Sie den AWS Security Agent zum Testen authentifizierter Bereiche Ihrer Anwendung benötigen.

Hinzufügen von Anmeldeinformationen

Geben Sie die Authentifizierungsdaten ein, die der AWS Security Agent für den Zugriff auf Ihre Anwendung verwendet.

1. Wählen Sie im Abschnitt Credential #1 eine Eingabemethode für Anmeldeinformationen aus:
   - Anmeldeinformationen eingeben — Geben Sie Ihre Anmeldeinformationen direkt in AWS Security Agent ein.
   - Erweiterte Einstellung — Verwenden Sie für vertrauliche Anmeldeinformationen erweiterte Optionen wie AWS Secrets Manager oder AWS Lambda Lambda-Funktionen. Details dazu finden Sie unter the section called “Geben Sie Anmeldeinformationen für Penetrationstests an”.

     > **Tip**
     >
     > Für Produktionsumgebungen oder vertrauliche Anmeldeinformationen empfehlen wir, die Option für erweiterte Einstellungen zu verwenden, um auf sichere Weise auf Anmeldeinformationen zu verweisen, die in AWS Secrets Manager oder Systems Manager Parameter Store gespeichert sind.

Geben Sie die Anmeldedaten ein

Geben Sie den Benutzernamen und das Passwort für das authentifizierte Konto ein.

1. Geben Sie im Feld Benutzername den Benutzernamen für die Authentifizierung ein.
2. Geben Sie im Feld Passwort das Passwort für die Authentifizierung ein.

> **Important**
>
> Stellen Sie sicher, dass die von Ihnen angegebenen Anmeldeinformationen über angemessene Zugriffsebenen für die Bereiche verfügen, die Sie testen möchten. Die Anmeldeinformationen sollten der Zugriffsebene eines typischen Benutzers entsprechen und nicht den Administratorrechten.

Wählen Sie die Zugriffsdomäne

Geben Sie an, welche Zieldomäne diese Anmeldeinformationen für die Authentifizierung verwenden soll.

1. Wählen Sie in der Dropdownliste Zugriffsdomäne die Domäne aus, in der diese Anmeldeinformationen verwendet werden sollen.

> **Note**
>
> Wenn Sie über mehrere Zieldomänen verfügen, für die unterschiedliche Anmeldeinformationen erforderlich sind, können Sie weitere Anmeldeinformationssätze hinzufügen, indem Sie nach Abschluss dieser Konfiguration der Anmeldeinformationen auf Weitere Anmeldeinformationen hinzufügen klicken.

Konfigurieren Sie die Anmeldeaufforderung für den Agenten (optional)

Stellen Sie Anweisungen bereit, um den AWS Security Agent durch den Authentifizierungsprozess Ihrer Anwendung zu führen.

1. Erweitern Sie den Abschnitt zur Agent-Anmeldeaufforderung, wenn Ihr Authentifizierungsablauf spezielle Anweisungen erfordert.

2. Geben Sie Anweisungen ein, die beschreiben, wie Sie die bereitgestellten Anmeldeinformationen im Anmeldeablauf Ihrer Anwendung verwenden können.

> **Note**
>
> Die Anmeldeaufforderung des Agenten teilt dem Agenten mit, wie er Ihre Anmeldeinformationen auf Ihre Anwendung anwenden soll. Dies ist nützlich für komplexe Authentifizierungsabläufe, mehrstufige Anmeldeprozesse oder Anwendungen mit nicht standardmäßigen Anmeldeverfahren. Fügen Sie schrittweise Anweisungen hinzu wie „Navigieren Sie zu /login, geben Sie den Benutzernamen in das Feld „E-Mail“ ein, geben Sie das Passwort ein und wählen Sie „Anmelden“.

Fügen Sie mehrere Anmeldeinformationen hinzu (optional)

Wenn für Ihre Anwendung mehrere Gruppen von Anmeldeinformationen erforderlich sind oder wenn für verschiedene Domänen eine separate Authentifizierung erforderlich ist, fügen Sie zusätzliche Anmeldeinformationssätze hinzu.

1. Nachdem Sie die erste Konfiguration der Anmeldeinformationen abgeschlossen haben, wählen Sie Weitere Anmeldeinformationen hinzufügen aus.
2. Wiederholen Sie die Schritte zur Konfiguration der Anmeldeinformationen für jeden weiteren Anmeldeinformationssatz.
3. Um einen Satz von Anmeldeinformationen zu entfernen, wählen Sie neben dem Anmeldeinformationsheader die Option Entfernen aus.

> **Tip**
>
> Konfigurieren Sie mehrere Anmeldeinformationen, wenn Sie verschiedene Benutzerrollen testen, auf mehrere authentifizierte Domänen zugreifen oder die rollenbasierten Zugriffskontrollen in Ihrer Anwendung überprüfen möchten.

Fügen Sie zusätzliche Ressourcen hinzu (optional)

Stellen Sie zusätzliche Ressourcen bereit, um den AWS Security Agent bei der Durchführung gründlicherer und genauerer Penetrationstests zu unterstützen. Zusätzliche Ressourcen können

Architekturdiagramme, API-Dokumentation, Konfigurationsdateien, GitHub Repositorys oder S3-hosted Materialien umfassen, die den Kontext Ihrer Anwendung vermitteln.

> **Note**
>
> Zusätzliche Ressourcen sind optional, werden aber empfohlen. Durch die Bereitstellung umfassender Informationen zu Ihrer Anwendung können Sie eine gründliche Testabdeckung sicherstellen, Fehlalarme reduzieren und aussagekräftigere Ergebnisse erzielen.

Fügen Sie dem Penetrationstest Ressourcen hinzu

Wählen Sie vorhandene Ressourcen aus oder laden Sie neue Dateien hoch, die als Leitfaden für den Penetrationstest dienen.

1. Im Bereich Verbundene Ressourcen können Sie:
   - Wählen Sie Aus verfügbaren auswählen, um aus Ressourcen auszuwählen, die bereits mit dem AWS Security Agent verbunden sind (wie GitHub Repositorys oder S3-Buckets).
   - Wählen Sie Hochladen, um neue Dateien direkt von Ihrem lokalen System hinzuzufügen.

> **Tip**
>
> Zu den nützlichen Ressourcen gehören API-Dokumentation, Architekturdiagramme, OpenAPI/Swagger Spezifikationen, Konfigurationsdateien, Authentifizierungsflussdiagramme und alle anderen Materialien, die die Struktur und das Verhalten Ihrer Anwendung beschreiben.

Wählen Sie aus den verfügbaren Ressourcen

Wählen Sie aus Ressourcen, die bereits in AWS Security Agent integriert sind.

1. Wählen Sie Aus vorhandenen Ressourcen auswählen.
2. Durchsuchen Sie die Liste der verfügbaren Ressourcen aus verbundenen Quellen wie:
   - GitHub Repositorys, auf der Registerkarte GitHub Repositorien
   - S3-Buckets
   - Zuvor hochgeladene Dateien

- Repositorien für Dokumentation

3. Wählen Sie die Ressourcen aus, die Sie in den Penetrationstest einbeziehen möchten.
4. Wählen Sie Zum Penetrationstest hinzufügen, um die ausgewählten Ressourcen anzuhängen.

Example

Wir empfehlen, relevante GitHub Repositorys auszuwählen und zu Ihrem Pentest hinzuzufügen, damit der AWS Security Agent ein Verständnis für Ihren Anwendungskontext entwickeln und implementierungsfertige Code-Fixes durch Pull-Requests generieren kann (sofern aktiviert)

> **Note**
>
> Ressourcen, die aus verfügbaren Quellen ausgewählt wurden, bleiben mit ihrem ursprünglichen Standort synchronisiert. Wenn Sie ein GitHub Repository oder eine S3-Datei aktualisieren, verwendet der Penetrationstest die aktualisierte Version.

> **Note**
>
> Wenn Sie Ihrem Pentest eine private VPC zugeordnet und ein GitHub Repository konfiguriert haben, stellen Sie sicher, dass GitHub es über Ihre private VPC zugänglich ist, um Ressourcen abzurufen. GitHub In den meisten Fällen müssen Sie entweder sicherstellen, dass ausgehender Datenverkehr über das VPC NAT Gateway standardmäßig zulässig ist, oder Sie müssen spezifische Regeln konfigurieren, um ausgehenden Datenverkehr für GitHub IPs zuzulassen (siehe GitHub Meta-API-Endpunkt).

Laden Sie neue Ressourcen hoch

Laden Sie Dateien direkt von Ihrem lokalen System hoch oder stellen Sie dem AWS Security Agent Klartextinhalte zur Verfügung.

1. Klicken Sie auf Upload.
2. Wählen Sie eine der folgenden Eingabemethoden:
   - Lokale Dateien hochladen — Wählen Sie eine oder mehrere Dateien aus Ihrem lokalen System aus.

- Klartext einfügen — Geben Sie Textinhalt direkt in das Eingabefeld ein oder fügen Sie ihn ein. Klicken Sie auf Upload.

3. Wählen Sie dann Hinzufügen, um den Upload abzuschließen.
4. Die hochgeladenen Ressourcen werden in der Tabelle Verbundene Ressourcen angezeigt.

> **Tip**
>
> Verwenden Sie die Option Klartext, wenn Sie schnell API-Endpunktlisten, URL-Muster, Testanweisungen oder andere textbasierte Informationen bereitstellen möchten, ohne eine separate Datei zu erstellen.

> **Important**
>
> Stellen Sie sicher, dass hochgeladene Dateien und eingefügte Inhalte keine vertraulichen Informationen wie Produktionsanmeldedaten, private Schlüssel oder personenbezogene Daten (PII) enthalten. Verwenden Sie bereinigte Versionen von Konfigurationsdateien und Dokumentation.

Bestehende Ressourcen Connect

Bestehende Ressourcen können aus dem stammen, was Sie zuvor auf AWS Security Agent hochgeladen haben, aus Ihrem S3-Bucket und Ihren integrierten GitHub Repositorys. Wählen Sie „Aus vorhandenen Ressourcen auswählen“, um sie auszuwählen.

Verbundene Ressourcen verwalten

Überprüfen, organisieren und entfernen Sie die mit dem Penetrationstest verbundenen Ressourcen.

In der Tabelle Verbundene Ressourcen werden alle im Penetrationstest enthaltenen Ressourcen mit den folgenden Informationen angezeigt:

- Name — Der Dateiname oder die Ressourcen-ID
- Typ — Die Ressourcenkategorie (Hochgeladene Dateien, S3-Ressourcen, GitHub Repositorys usw.)

Um Ressourcen zu verwalten:

1. Wählen Sie mithilfe der Checkboxen eine oder mehrere Ressourcen aus.
2. Wählen Sie Aus dem Penetrationstest entfernen, um die ausgewählten Ressourcen zu trennen.

> **Note**
>
> Sie können die Tabelle nach Name oder Typ sortieren, indem Sie auf die Spaltenüberschriften klicken. Dies hilft bei der Organisation von Ressourcen, wenn Sie mit vielen Dateien arbeiten.

Erstellen Sie den Penetrationstest

Finalisieren und starten Sie Ihre Penetrationstest-Konfiguration.

Nachdem Sie alle Einstellungen konfiguriert haben, können Sie den Penetrationstest erstellen.

1. Überprüfen Sie alle Konfigurationsabschnitte, um die Richtigkeit sicherzustellen.
2. Wählen Sie eine der folgenden Optionen:
   - Wählen Sie Create penetration, um die Konfiguration zu speichern, ohne sie sofort auszuführen.
   - Wählen Sie Create and execute, um die Konfiguration zu speichern und sofort den Penetrationstest zu starten.
   - Wählen Sie Abbrechen, um die Penetrationstest-Konfiguration zu verwerfen.

> **Important**
>
> Bevor Sie einen Penetrationstest durchführen, stellen Sie sicher, dass:
>
> - Alle Zieldomänen sind korrekt verifiziert und zugänglich
> - IAM-Rollen verfügen über die entsprechenden Berechtigungen
> - Out-of-scope Die Pfade sind ordnungsgemäß konfiguriert, um zerstörerische Testvorgänge zu verhindern
> - Sie sind autorisiert, Sicherheitstests für alle Zieldomänen durchzuführen

> **Note**
>
> Nach dem Start des Penetrationstests können Sie seinen Fortschritt im Abschnitt Penetrationstestläufe überwachen. Es kann mehrere Stunden dauern, bis der Test abgeschlossen ist. Je nach Umfang und Komplexität Ihrer Anwendung sind die meisten innerhalb von 16 Stunden abgeschlossen.

Überprüfen Sie die Ergebnisse eines Penetrationstests

Überwachen Sie die Pentest-Ausführung in Echtzeit auf der Seite Penetration Test Logs, nachdem der AWS Security Agent einen Pentest gestartet hat. Der AWS Security Agent protokolliert jede Aktion während des Pentests. Lesen Sie nach Abschluss der Pentest-Zusammenfassung, die einen Überblick über die Anwendung, die Abdeckung mit identifizierten Endpunkten und eine Risikobewertung der Sicherheitsergebnisse enthält.

Evaluieren Sie die Sicherheitsergebnisse, um Sicherheitslücken in Anwendungen zu beheben. Jedes Ergebnis enthält eine Folgenabschätzung, eine Bewertung des Schweregrads, unterstützende Nachweise und Details zur Pull-Anfrage zur Behebung (sofern die automatische Codekorrektur aktiviert ist).

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein abgeschlossener oder laufender Penetrationstest
- Zugriff auf die AWS Security Agent-Webanwendung

Schritt 1: Greifen Sie auf den Penetrationstestlauf zu

Navigieren Sie zu Ihrem Penetrationstestlauf, um eine Übersicht, Protokolle und Ergebnisseiten einzusehen.

1. Melden Sie sich bei der AWS Security Agent-Webanwendung an.
2. Navigieren Sie zum Abschnitt Penetrationstests.
3. Wählen Sie den Penetrationstestlauf, den Sie untersuchen möchten, aus der Liste aus.

> **Tip**
>
> Auf der Detailseite für den Penetrationstest wird eine Zusammenfassung des Teststatus, des Abschlussdatums und der Anzahl der identifizierten Ergebnisse angezeigt.

Schritt 2: Überwachen Sie den Testfortschritt

Verfolgen Sie den Fortschritt Ihres Penetrationstestlaufs anhand des Schrittindikators.

1. Suchen Sie den horizontalen Schrittindikator unter dem Seitenkopf.
2. Überprüfen Sie den Status jeder Testphase:
   - Preflight — Ersteinrichtung und Konnektivitätsprüfungen
   - Statische Analyse — Code- und Konfigurationsanalyse
   - Pentests — Laufzeittests und Schwachstellenscans
   - Abschluss — Endgültige Validierung und Erstellung von Berichten

> **Note**
>
> Für jeden Schritt wird eine Statusanzeige angezeigt (Abgeschlossen, In Bearbeitung oder Ausstehend). Die Ergebnisse werden während des gesamten Testprozesses entdeckt und validiert, wobei nach Abschluss jeder Phase neue Sicherheitslücken auftreten.

Schritt 3: Navigieren Sie zur Registerkarte mit der Übersicht über den Durchlauf des Penetrationstests

1. Der Abschnitt mit der Zusammenfassung des Testlaufs enthält Informationen zum Teststatus, zur Dauer und zu weiteren wichtigen Informationen. Es bietet auch ein Dashboard mit Sicherheitsergebnissen, die nach Schweregrad und Risikotypen kategorisiert sind
2. Die Anwendungsübersicht von AWS Security Agent bietet eine Zusammenfassung des durchgeführten Penetrationstests.
3. Vom AWS Security Agent entdeckte Endpunkte bietet eine Liste aller Endpunkte, die der AWS-Sicherheitsagent während des Pentestlaufs entdeckt und getestet hat.

Schritt 4: Navigieren Sie zur Registerkarte „Penetrationstest-Protokolle“

Greifen Sie auf detaillierte Protokolle aller Aktionen zu, die der AWS Security Agent während des Pentests ausgeführt hat.

1. Die Aktionen sind nach Aktionstyp und Risikotypen kategorisiert.
2. Wählen Sie eine bestimmte Aktion aus, um detaillierte Protokolle anzuzeigen:
   - Zusammenfassung der Tests — High-level Zusammenfassung der Aktionen und Ergebnisse des Agenten
   - Penetrationstest-Protokolle — Detaillierte Protokolle aller Testaktivitäten

> **Note**
>
> Validator-Aktionen liefern Protokolle, die die Ergebnisse in jeder Kategorie validieren

> **Note**
>
> Auf der Registerkarte Ergebnisse wird eine geteilte Ansicht mit der Ergebnisliste auf der linken Seite und ausgewählten Ergebnisdetails auf der rechten Seite angezeigt.

Schritt 5: Navigieren Sie zu den Ergebnissen

Jedes Ergebnis in der Liste enthält wichtige Informationen, anhand derer Sie schnell einschätzen können, wie wichtig es ist.

> **Note**
>
> Standard-Konfidenzfilter
>
> Standardmäßig werden nur Ergebnisse mit hoher Agentensicherheit angezeigt. Um auch Ergebnisse mit mittlerer oder niedriger Agentensicherheit und falsch positive Ergebnisse anzuzeigen, deaktivieren Sie die Option Unbestätigte Ergebnisse ausblenden.

Überprüfen Sie die Informationen, die auf den einzelnen Ergebniskarten angezeigt werden:

- Fundname — Der Titel und die Kennung der Sicherheitslücke
- Sicherheitsabzeichen — Zeigt an, ob der Agent mit dem Ergebnis vertraut ist (Hoch, Mittel oder Niedrig)
- Ausweis „Schweregrad“ — Zeigt die Risikostufe mit farblicher Kennzeichnung an:
  - Kritisch (rot) — Sofortiges Handeln erforderlich; Ausnutzung kann zu Systemgefährdungen führen
  - Hoch (rot) — Sofortiges Handeln ist erforderlich; eine Ausnutzung kann zu erheblichen Sicherheitsrisiken führen
  - Mittel (orange) — Sollte innerhalb eines angemessenen Zeitrahmens behoben werden; trägt zum allgemeinen Sicherheitsrisiko bei
  - Niedrig (gelb) — Kann im Rahmen regelmäßiger Wartungsarbeiten behoben werden; minimales unmittelbares Risiko
  - Informativ (blau) — Zu Informationszwecken; minimales bis kein unmittelbares Risiko
- Zeitstempel der letzten Aktualisierung — Zeigt an, wann das Ergebnis zuletzt geändert oder validiert wurde
- Vorschau der Beschreibung — Kurze Zusammenfassung der Sicherheitsanfälligkeit

> **Important**
>
> Priorisieren Sie die Ergebnisse mit den Hinweisen „Kritisch“ oder „Hoch“ und der Vertrauensstufe „Hoch“, da es sich dabei um validierte Sicherheitslücken handelt, die sofort behoben werden müssen.

Schritt 6: Überprüfen Sie die Details zu den Ergebnissen

Wählen Sie einzelne Ergebnisse aus, um umfassende Informationen zu den einzelnen Sicherheitslücken zu erhalten.

1. Wählen Sie im linken Bereich einen Namen für ein Ergebnis aus, um dessen Details im rechten Bereich anzuzeigen.
2. Überprüfen Sie den Validierungsstatus:

> **Note**
>
> Wenn bei einem Ergebnis die Meldung Unbekannt „Dieses Ergebnis wurde noch nicht vom AWS Security Agent bestätigt“ angezeigt wird, bedeutet dies, dass die Schwachstellenerkennung immer noch bestätigt wird. Diese Ergebnisse erfordern möglicherweise eine manuelle Überprüfung.

3. Überprüfen Sie die oben angezeigten Schlüsselattribute:
   - Vertrauen der Agenten — Das Vertrauensniveau, das AWS Security Agent in dieses Ergebnis eingeht
   - Schweregrad — Die Risikostufe mit einem farblich markierten Schild
   - Logs finden — Wählen Sie „Aktionen und Logs verfolgen“, um detaillierte Ausführungsprotokolle und Nachweise einzusehen
   - Risikotyp — Die Kategorie oder Art des Sicherheitsrisikos (z. B. Umgehung der Authentifizierung, SQL-Injection)
4. Erweitern Sie den Abschnitt Beschreibung um folgenden Text:
   - Eine ausführliche Erklärung der Sicherheitsanfälligkeit
   - Wie funktioniert die Sicherheitsanfälligkeit
   - Warum stellt sie ein Sicherheitsrisiko dar
   - Die möglichen Auswirkungen auf Ihre Anwendung
5. Erweitern Sie den Abschnitt Risk Reasoning, um mehr über die Berechnung des Schweregrads zu erfahren:
   - Aufschlüsselung der CVSS-Metriken (Common Vulnerability Scoring System)
   - Attack Vector (AV) — Wie die Sicherheitslücke ausgenutzt werden kann
   - Angriffskomplexität (AC) — Wie schwierig ist der Exploit
   - Erforderliche Rechte (PR) — Welche Zugriffsebene ist erforderlich
   - Benutzerinteraktion (UI) — Ob eine Benutzeraktion erforderlich ist
   - Umfang (S) — Ob sich die Sicherheitsanfälligkeit auf andere Komponenten auswirkt
   - Auswirkungen auf Vertraulichkeit, Integrität und Verfügbarkeit
6. Erweitern Sie den Abschnitt Schritte zur Reproduktion, um Folgendes anzuzeigen:
   - Detaillierte technische Schritte zur Wiederherstellung der Sicherheitslücke
   - Beispiele für Anfragen und Antworten

- Spezifische Parameter oder Bedingungen, die das Problem auslösen

7. Der Abschnitt Überprüfungsskript bietet eine ausführbare Möglichkeit, das Ergebnis zu reproduzieren. Erweitern Sie diesen Abschnitt (sofern verfügbar), um Folgendes anzuzeigen:
   - Anweisungen — So richten Sie das Überprüfungsskript ein und führen es aus
   - Umgebungsvariablen — Listet die erforderlichen Variablen auf, die Sie konfigurieren müssen, bevor Sie das Skript ausführen. Vertrauliche Werte werden aus Sicherheitsgründen geschwärzt.
   - Skript herunterladen — Wählen Sie diese Option, um das ausführbare Überprüfungsskript herunterzuladen

> **Note**
>
> Überprüfungsskripte sind nur für bestätigte Sicherheitslücken verfügbar. Der Agent muss erfolgreich ein ausführbares Reproduktionsskript generiert und validiert haben.

> **Note**
>
> Verifizierungsskripte werden mithilfe generativer KI generiert. Überprüfen Sie das Skript vor der Ausführung und führen Sie es nur auf Systemen aus, zu deren Testen Sie berechtigt sind. Anleitungen zum verantwortungsvollen Testen von AI-generated Skripten finden Sie in der AWS-Richtlinie für verantwortungsvolle KI.

> **Tip**
>
> Das Überprüfungsskript bietet eine ausführbare Möglichkeit, das Ergebnis unabhängig zu reproduzieren. Stellen Sie die erforderlichen Umgebungsvariablen mit Ihren eigenen Anmeldeinformationen ein und führen Sie dann das Skript auf Ihrem Zielsystem aus, um zu überprüfen, ob die Sicherheitsanfälligkeit besteht.

> **Tip**
>
> Verwenden Sie den Link „Aktionen und Protokolle verfolgen“, um auf das vollständige Beweispaket zuzugreifen, einschließlich HTTP-Anfragen, Antworten und Ausnutzungsversuchen, die die Sicherheitsanfälligkeit belegen.

Ergebnisse bearbeiten

Sie können jedes Ergebnis bearbeiten, um die Details zu korrigieren oder die Einschätzung des Agenten zu verfeinern. Die folgenden Felder können bearbeitet werden:

- name — Der Titel des Befundes
- Beschreibung — Detaillierte Beschreibung der Sicherheitslücke
- Status — Aktueller Status des Befundes
- riskType — Art des identifizierten Sicherheitsrisikos
- RiskLevel — Schweregrad (KRITISCH, HOCH, MITTEL, NIEDRIG, INFORMATIV)
- RiskScore — Numerische Risikobewertung
- Argumentation — Begründung für die zugewiesene Risikobewertung
- AttackScript — Proof-of-concept Code, der die Sicherheitsanfälligkeit demonstriert
- CustomerNote — Optionaler Hinweis, in dem Sie Ihre Gründe für die Bearbeitung erläutern

Um ein Ergebnis zu bearbeiten:

1. Navigieren Sie zur Detailseite des Ergebnisses.
2. Wählen Sie das Bearbeitungssymbol, um eines der Felder in der vorherigen Liste zu ändern.
3. Speichern Sie Ihre Änderungen.

> **Note**
>
> Ihre Änderungen werden sofort gespeichert und spiegeln sich in den Ergebnissen wider. Wenn die Personalisierung von Ergebnissen aktiviert ist, werden alle editierbaren Felder, die Sie ändern, verwendet, um die Einstellungen des Agenten für future Läufe zu verfeinern.

Sehen Sie sich die ursprüngliche Agentenversion an

Wenn Sie ein Ergebnis bearbeiten, wird in der Kopfzeile der Ergebnisdetails eine Versionsauswahl angezeigt. Auf diese Weise können Sie sich die ursprüngliche Bewertung durch den Agenten ansehen:

1. Suchen Sie in der Kopfzeile der Suchdetails nach der Versionsauswahl.
2. Wählen Sie Original aus, um das Ergebnis so anzuzeigen, wie es ursprünglich vom Agenten erstellt wurde.
3. Wählen Sie Neueste aus, um zur aktuellen Version zurückzukehren, in der Ihre Änderungen übernommen wurden.

Überprüfen Sie die personalisierten Ergebnisse

Wenn die Personalisierung von Ergebnissen aktiviert ist, lernt AWS Security Agent aus Ihren Änderungen an den Ergebnissen. Der Agent wendet diese Einstellungen auf ähnliche Ergebnisse in future Penetrationstestläufen an. Wenn ein Ergebnis auf der Grundlage Ihrer vorherigen Änderungen angepasst wurde, wird im Bereich „Details“ der Abschnitt „Personalisierungsänderungen“ angezeigt. Informationen zur Aktivierung dieser Funktion finden Sie unter Personalisierung von Ergebnissen aktivieren oder deaktivieren.

1. Suchen Sie im Bereich mit den Ergebnisdetails den Abschnitt Personalisierungsänderungen.

> **Note**
>
> Der Abschnitt Personalisierungsänderungen wird nur bei Ergebnissen angezeigt, die AWS Security Agent an Ihre vorherigen Änderungen angepasst hat. Ergebnisse, die keiner erlernten Präferenz entsprechen, werden genau so angezeigt, wie der Agent sie erstellt hat, also ohne diesen Abschnitt.

2. Lesen Sie in der Zusammenfassung, was sich geändert hat und warum. In der Zusammenfassung werden alle angepassten Attribute mit ihrem ursprünglichen, vom Agenten generierten Wert, dem personalisierten Wert und der Begründung aufgeführt. Beispiel:

Aufgrund Ihrer vorherigen Änderungen wurden die folgenden Änderungen an den Ergebnissen vorgenommen, die ursprünglich vom System generiert wurden: Die Risikostufe wurde von MITTEL auf KRITISCH geändert; die Risikobewertung wurde von 5,3 auf 9,5 geändert. Begründung: Der

Schweregrad wurde erhöht, um die vollständige Offenlegung des Anwendungsquellcodes über öffentlich zugängliche Quellzuordnungen widerzuspiegeln.

1. Lesen Sie die Zusammenfassung, um zu verstehen, wie das Ergebnis angepasst wurde, bevor Sie entscheiden, wie Sie darauf reagieren möchten.
2. Um ein personalisiertes Ergebnis außer Kraft zu setzen, bearbeiten Sie es erneut wie jedes andere Ergebnis (ändern Sie beispielsweise den Schweregrad oder die Risikobewertung). Ihre letzte Änderung hat immer Vorrang: AWS Security Agent lernt daraus und wendet die aktualisierte Einstellung bei der nächsten Ausführung an, wodurch die frühere Regel ersetzt wird.

> **Note**
>
> Personalisierungsänderungen passen Suchattribute wie RiskLevel, RiskScore, Name, Beschreibung, Argumentation und AttackScript an die Standards an, die AWS Security Agent aus Ihren Änderungen gelernt hat. Jedes Feld, das Sie zuvor bearbeitet haben, kann in future Durchläufen aufgrund ähnlicher Ergebnisse angepasst werden.

Wie Personalisierung aus Ihren Änderungen lernt

Wenn die Personalisierung von Ergebnissen aktiviert ist, lernt der AWS Security Agent aus allen Änderungen, die Sie an den Ergebnissen vornehmen, und wendet ähnliche Anpassungen auf future Läufe an. Jedes Feld, das Sie bearbeiten (wie im vorherigen Abschnitt Ergebnisse bearbeiten aufgeführt), wird verwendet, um die erlernten Einstellungen des Agenten zu verfeinern.

> **Note**
>
> Im Statusfeld wird nur die Markierung eines Befundes als FALSE_POSITIV als erlernbare Präferenz behandelt. Änderungen an anderen Statuswerten lösen kein Lernen aus.

Beim nächsten Pentest-Lauf bewertet der Agent die neuen Ergebnisse anhand Ihrer erlernten Einstellungen und nimmt gegebenenfalls Anpassungen vor. Für jedes Ergebnis, das angepasst wurde, wird ein Abschnitt mit Änderungen an der Personalisierung angezeigt, in dem erklärt wird, was geändert wurde.

> **Note**
>
> Die Personalisierung lernt nur aus den Änderungen, die im letzten abgeschlossenen Pentest-Lauf vorgenommen wurden. Die Funktion muss zu Beginn des Pentests aktiviert sein, damit das Lernen stattfinden kann. Wenn Sie dasselbe Ergebnis zu einem späteren Zeitpunkt bearbeiten, hat die letzte Änderung Vorrang. Der Agent aktualisiert seine Einstellungen entsprechend Ihrer letzten Entscheidung.

Aktivieren oder deaktivieren Sie die Personalisierung von Ergebnissen

Die Personalisierung von Ergebnissen ist standardmäßig aktiviert. Um es zu deaktivieren oder erneut zu aktivieren:

1. Navigieren Sie zu Ihrer Pentest-Konfiguration.
2. Suchen Sie den Schalter für die Personalisierung von Ergebnissen.
3. Um die Personalisierung von Ergebnissen zu aktivieren, schalten Sie den Schalter ein. Um ihn zu deaktivieren, schalten Sie ihn aus.

Wenn diese Option deaktiviert ist, werden die Ergebnisse in ihrem ursprünglichen, vom Agenten erstellten Zustand angezeigt, ohne dass eine Personalisierung angewendet wurde. Zuvor erlernte Einstellungen bleiben erhalten und werden erneut angewendet, wenn die Funktion erneut aktiviert wird.

Schritt 7: Interpretieren Sie die CVSS-Metriken

Wenn Sie die CVSS-Metriken verstehen, können Sie den tatsächlichen Schweregrad einschätzen und Korrekturmaßnahmen priorisieren.

Achten Sie bei der Lektüre des Abschnitts „Argumentation“ auf die folgenden wichtigen Kennzahlen:

- Angriffsvektor (Network/Adjacent/Local/Physical) — Gibt an, wie der Angriff aus der Ferne ausgeführt werden kann
- Angriffskomplexität (Low/High) — Zeigt an, ob für die Ausnutzung spezielle Bedingungen erforderlich sind
- Erforderliche Rechte (None/Low/High) — Identifiziert, welche Zugriffsebene ein Angreifer benötigt
- Benutzerinteraktion (None/Required) — Stellt fest, ob der Exploit die Beteiligung des Benutzers erfordert

- Confidentiality/Integrity/Availability Auswirkung (None/Low/High) — Misst die Auswirkungen auf die Sicherheit Ihres Systems

> **Important**
>
> Ergebnisse mit Netzwerkangriffsvektor, geringer Komplexität und hoher confidentiality/integrity Auswirkung stellen die gefährlichsten Sicherheitslücken dar, die sofortige Maßnahmen erfordern.

Schritt 8: Priorisieren Sie die Ergebnisse und gehen Sie darauf ein

Ergreifen Sie auf der Grundlage der Ergebnisse Maßnahmen, um Sicherheitslücken zu beheben und die Sicherheitslage Ihrer Anwendung zu verbessern.

Für kritische Ergebnisse und Ergebnisse mit hohem Schweregrad mit hoher Sicherheit:

1. Lesen Sie die Abschnitte „Beschreibung“ und „Schritte zur Reproduktion“ sorgfältig durch.
2. Greifen Sie über den Link „Aktionen und Protokolle verfolgen“ auf die detaillierten Protokolle zu, um vollständige Beweise zu sammeln.
3. Greifen Sie mit einer der folgenden Methoden auf sofort einsatzbereite Code-Fixes zu:
   - Für automatische Problembehebung: Verwenden Sie den Pull-Request-Link im Bereich Problembehebung
   - Für manuelle Anfragen: Wähle auf der Ergebnisseite „Behebungscode“, um eine Pull-Anfrage anzufordern. Voraussetzungen:
   - Der Administrator muss die Codekorrektur für GitHub Repositorys in der AWS Security Agent-Konsole aktivieren.
   - Repositorys müssen in Ihrer Pentest-Konfiguration enthalten sein
4. Planen Sie einen nachfolgenden Penetrationstest ein, um zu überprüfen, ob das Update wirksam ist.

Für Befunde mit mittlerem und niedrigem Schweregrad:

1. Priorisieren Sie auf der Grundlage Ihrer Risikotoleranz und Ihres Geschäftskontextes.
2. Nehmen Sie Problembehebungsaufgaben in Ihre reguläre Planung von Entwicklungs-Sprints auf.

3. Überlegen Sie, ob mehrere Befunde mit geringem Schweregrad zusammen ein höheres Risiko darstellen.
4. Dokumentieren Sie alle akzeptierten Risiken mit entsprechender Begründung.

> **Important**
>
> Ignorieren Sie Ergebnisse mit geringem Schweregrad nicht. Oft können mehrere Sicherheitslücken mit geringem Schweregrad zu schwerwiegenderen Exploits verkettet werden, insbesondere in Kombination mit Social Engineering oder physischem Zugriff.

Schritt 9: Verfolgen Sie den Fortschritt der Problembehebung

Verwenden Sie die Oberfläche mit Ergebnissen, um nachzuverfolgen, welche Sicherheitslücken behoben wurden und bei welchen weitere Maßnahmen erforderlich sind.

1. Während Sie an der Problembehebung arbeiten, schauen Sie sich den Abschnitt „Schritte zur Reproduktion“ an, um Ihre Korrekturen zu überprüfen.
2. Dokumentieren Sie Ihren Behebungsansatz für jedes Ergebnis für future Referenz- und Compliance-Audits.

> **Tip**
>
> Führen Sie ein Behebungsprotokoll, das jedes Ergebnis seiner Lösung zuordnet, einschließlich der Codeänderungen, Konfigurationsupdates oder Architekturentscheidungen, die zur Behebung der Sicherheitsanfälligkeit getroffen wurden.

Nächste Schritte

Nachdem Sie die Ergebnisse Ihres Penetrationstests überprüft haben:

- Priorisieren Sie kritische und schwerwiegende Ergebnisse mit hoher Zuverlässigkeit, um eine sofortige Behebung zu ermöglichen
- Laden Sie Verifizierungsskripts herunter, überprüfen Sie sie und führen Sie sie in Ihrer Testumgebung aus, um die Skripts zu testen und Sicherheitslücken zu identifizieren

- Erstellen Sie in Ihrem Issue-Management-System Tracking-Tickets mit Links zur Suche nach Details und Nachweisen
- Implementieren Sie Korrekturen und Sicherheitskontrollen, um identifizierte Sicherheitslücken zu beheben
- Überwachen Sie den Fortschrittsindikator des Penetrationstestlaufs auf neu entdeckte Sicherheitslücken
- Planen Sie einen nachfolgenden Penetrationstest ein, um sicherzustellen, dass die Sicherheitslücken ordnungsgemäß behoben wurden
- Aktualisieren Sie Ihren Testprozess für die Anwendungssicherheit und Ihr Bedrohungsmodell auf der Grundlage der Ergebnisse
- Sehen Sie sich die CVSS-Metriken an, um den allgemeinen Sicherheitsstatus Ihrer Anwendung zu verstehen

Weitere Informationen zur Durchführung von Penetrationstests finden Sie unterthe section called “Erstellen Sie einen Penetrationstest”.

Weitere Informationen zum Lebenszyklus des Security Agents finden Sie unterthe section called “Verstehen Sie die Ressourcenhierarchie und den Lebenszyklus”.

Geben Sie Anmeldeinformationen für Penetrationstests an

Geben Sie Anmeldeinformationen ein, damit der AWS Security Agent authentifizierte Bereiche Ihrer Webanwendungen testen kann. Ohne Anmeldeinformationen kann der Agent nur öffentlich zugängliche Seiten und APIs testen.

Konfigurieren Sie die Anmeldeinformationen für die Authentifizierung

1. Suchen Sie im Workflow zur Erstellung von Penetrationstests den Abschnitt Authentifizierungsdaten — optional.
2. Wählen Sie im Abschnitt Credential #1 Ihre Eingabemethode für Anmeldeinformationen aus:
   - Anmeldeinformationen eingeben — Geben Sie die Anmeldeinformationen direkt ein. Am besten für Entwicklungs- und Testumgebungen geeignet.
   - Erweiterte Einstellung — Verwenden Sie die Verwaltung von AWS-native Anmeldeinformationen. Empfohlen für Produktionsumgebungen und sensible Anmeldeinformationen.

Erweiterte Optionen

Wenn Sie die Option Erweiterte Einstellung auswählen, können Sie aus drei Strategien für Anmeldeinformationen wählen:

- Übernahme der IAM-Rolle — Für Anwendungen, die die AWS Cognito- oder IAM-Authentifizierung verwenden
- AWS Secrets Manager — Für die sichere Speicherung von Anmeldeinformationen mit Verschlüsselung und Rotation
- Lambda-Funktion — Für die dynamische Generierung von Anmeldeinformationen oder komplexe Authentifizierungsabläufe

Geben Sie die Anmeldeinformationen direkt ein

1. Wählen Sie Anmeldeinformationen eingeben aus.
2. Geben Sie den Benutzernamen und das Passwort ein.
3. Wählen Sie in der Dropdownliste Zugriffs-URL die URL aus, unter der diese Anmeldeinformationen verwendet werden sollen. Dies muss aus der Liste der Zielendpunkte ausgewählt werden.
4. (Optional) Geben Sie im Feld 2FA — optional ein TOTP-Geheimnis für Anwendungen ein, die eine Zwei-Faktor-Authentifizierung erfordern. Führen Sie dazu einen der folgenden Schritte aus:
   - Geben Sie den TOTP-Schlüssel direkt ein (z. B.`JBSWY3DPEHPK3PXP`) oder geben Sie den vollständigen `otpauth://totp/` URI ein (z. B.). `otpauth://totp/Example:user@example.com?secret=JBSWY3DPEHPK3PXP&issuer=Example`
   - Wählen Sie das Upload-Symbol, um ein QR-Code-Bild von der Einrichtungsseite Ihrer Authenticator-App hochzuladen. Der QR-Code wird lokal gescannt und die TOTP-URI wird automatisch extrahiert.

     Wenn ein TOTP-Geheimnis angegeben wird, generiert der Agent automatisch neue Einmalcodes und gibt sie ein, wenn bei der Anmeldung eine 2FA-Aufforderung erkannt wird.
5. (Optional) Erweitern Sie die Agent Space-Anmeldeaufforderung, um spezifische Anmeldeanweisungen bereitzustellen, wenn Ihre Anwendung über einen komplexen Authentifizierungsablauf verfügt.

> **Important**
>
> Verwenden Sie Testkonten mit repräsentativem Zugriff anstelle von persönlichen Konten oder Administratorkonten.

Verwenden Sie erweiterte Einstellungen

Wenn Sie eine erweiterte Anmeldeinformationsstrategie (Secrets Manager-, Lambda- oder IAM-Rolle) auswählen, ruft AWS Security Agent Ihre Anmeldeinformationen mithilfe der Servicerolle direkt von der konfigurierten AWS-Ressource ab. Verwenden Sie die Agent Space-Anmeldeaufforderung, um dem Agenten Anweisungen zu geben, wie Sie diese Anmeldeinformationen auf Ihre Anwendung anwenden können — zum Beispiel, zu welcher Anmelde-URL Sie navigieren und welche Formularfelder Sie ausfüllen müssen.

> **Note**
>
> Secrets Manager Manager-Geheimnisse und Lambda-Funktionen müssen sich in demselben AWS-Konto befinden wie Ihr AWS Security Agent-Setup. Cross-account Anmeldeinformationen werden derzeit nicht unterstützt.

1. Wählen Sie Erweiterte Einstellung aus.
2. Wählen Sie in der Dropdownliste Strategie für den Benutzerzugriff eine der folgenden Optionen aus:

Wählen Sie die verfügbare IAM-Rolle aus, die der Agent annehmen soll

Verwenden Sie diese Option für Anwendungen, die AWS Cognito, API Gateway mit IAM-Authentifizierung oder andere AWS-native Authentifizierungssysteme verwenden. Die IAM-Rolle muss über eine Vertrauensbeziehung verfügen, die es dem AWS Security Agent ermöglicht, diese zu übernehmen, sowie über Berechtigungen für den Zugriff auf das Authentifizierungssystem Ihrer Anwendung.

Wählen Sie statische Anmeldeinformationen aus dem verbundenen AWS Secrets Manager aus

Verwenden Sie diese Option, um Anmeldeinformationen mit Verschlüsselung, Rotation und Zugriffsprüfung sicher von AWS Secrets Manager abzurufen.

Die IAM-Rolle muss über `secretsmanager:DescribeSecret` Berechtigungen `secretsmanager:GetSecretValue` verfügen.

Der Agent ruft den geheimen Wert direkt von Secrets Manager ab. Verwenden Sie die Agent Space-Anmeldeaufforderung, um dem Agenten mitzuteilen, wie er diese Anmeldeinformationen für den Anmeldeablauf Ihrer Anwendung verwenden soll — beispielsweise, zu welcher URL er navigieren und welche Formularfelder ausgefüllt werden müssen. Sie können ein beliebiges Format zum Speichern Ihres Geheimnisses verwenden, da der Agent das Format anhand der Anweisungen interpretiert, die Sie in der Anmeldeaufforderung angeben.

Wenn der Agent beispielsweise ein username/password Anmeldeformular unter einreichen sollhttps://example.com/login, können Sie Ihr Geheimnis als JSON mit `username` und `password` -Feldern formatieren. Wenn für die Anwendung TOTP-based 2FA erforderlich ist, fügen Sie ein `totpSecret` Feld hinzu, das entweder direkt den TOTP-Schlüssel oder eine vollständige `otpauth://totp/` URI enthält:

```
{
  "username": "test-user",
  "password": "secure-password-here",
  "totpSecret": "JBSWY3DPEHPK3PXP"
}
```

Konfigurieren Sie dann die Authentifizierungsanweisungen:. Legen Sie die Zugriffs-URL auf fest `https://example.com` (oder eine andere URL, die aus der Liste der Zielendpunkte ausgewählt wurde). Geben Sie Folgendes in die Agent Space-Anmeldeaufforderung ein: „Navigieren Sie zu dem angegebenen Benutzernamen https://example.com/login und Passwort und geben Sie ihn in das Formular ein.“

Ein weiteres Beispiel: Wenn Sie stattdessen einen API-Schlüssel in einem HTTP-Header angeben möchten, können Sie ihn als Klartext speichern:

```
"api-key-here"
```

Konfigurieren Sie dann die Authentifizierungsanweisungen:. Geben Sie Folgendes in die Agent Space-Anmeldeaufforderung ein: „Setzen Sie den X-API-Key Header für alle Anfragen auf den bereitgestellten API-Schlüssel.“

> **Important**
>
> Nur TOTP-based 2FA wird unterstützt. SMS, E-Mail, Push-Benachrichtigungen, Hardwareschlüssel und OAuth-Authentifizierung werden nicht unterstützt.

Wählen Sie die verfügbare Lambda-Funktion aus, um Anmeldeinformationen dynamisch abzurufen

Verwenden Sie diese Option für komplexe Authentifizierungssysteme, die dynamische Generierung von Anmeldeinformationen oder die Integration mit externen Identitätsanbietern.

Die IAM-Rolle muss über `lambda:InvokeFunction` Berechtigungen verfügen und die Funktion muss innerhalb von 30 Sekunden abgeschlossen sein.

Wenn der Penetrationstest ausgeführt wird, ruft der AWS Security Agent Ihre AWS Lambda Lambda-Funktion synchron auf und übergibt ein Ereignis, das den Penetrationstest und die spezifischen Anmeldeinformationen identifiziert, die aufgelöst werden:

```
{
  "pentest_arn": "arn:aws:securityagent:us-west-2:111122223333:pentest/pt-
abcd1234-5678-90ab-cdef-EXAMPLE11111",
  "actor_identifier": "Credential1"
}
```

- `pentest_arn`— Der Amazon-Ressourcenname (ARN) des Penetrationstests, für den die Anmeldeinformationen aufgelöst werden. Verwenden Sie ihn, um die Anfrage innerhalb Ihrer Funktion zu spezifizieren, zu autorisieren oder zu prüfen.
- `actor_identifier`— Der Name der Anmeldeinformationen, den Sie diesen Anmeldeinformationen in der Konsole zugewiesen haben (z. B.). `Credential1` Verwenden Sie ihn, um die richtigen Anmeldeinformationen zurückzugeben, wenn eine einzelne Funktion mehrere Anmeldeinformationen verwendet.

Der Agent verwendet die Ausgabe der Funktion direkt als Anmeldeinformationen. Verwenden Sie die Agent Space-Anmeldeaufforderung, um dem Agenten mitzuteilen, wie er diese Anmeldeinformationen auf Ihre Anwendung anwenden soll, genauso wie Sie es mit AWS Secrets Manager tun würden. Beispiele the section called “Wählen Sie statische Anmeldeinformationen aus dem verbundenen AWS Secrets Manager aus” zur Formatierung der Ausgabe Ihrer Lambda-Funktion und zu unterstützten Authentifizierungstypen finden Sie unter.

Konfigurieren Sie mehrere Anmeldeinformationen

So testen Sie verschiedene Benutzerrollen oder Authentifizierungssysteme:

1. Wählen Sie Weitere Anmeldeinformationen hinzufügen aus.
2. Konfigurieren Sie die zusätzlichen Anmeldeinformationen mit einer der Eingabemethoden.
3. Um Anmeldeinformationen zu entfernen, wählen Sie im Abschnitt Anmeldeinformationen die Option Entfernen aus.

Optimierung der Anmeldung

Die Anmeldungsoptimierung ermöglicht es dem AWS Security Agent, Navigationsmuster aus früheren Pentest-Läufen zu lernen und sie auf nachfolgende Läufe anzuwenden. Zu diesen Mustern gehören Anmeldeabläufe und mehrstufige Authentifizierungsworkflows. Dies reduziert die Zeit, die der Agent damit verbringt, erneut herauszufinden, wie er sich authentifizieren kann, was zu schnelleren Scans und einer konsistenteren Testabdeckung führt.

So funktioniert die Login-Optimierung

Beim ersten Pentest-Lauf erkennt der Agent, wie er mithilfe der von Ihnen angegebenen Anmeldeinformationen im Anmeldeablauf Ihrer Anwendung navigieren kann. Es zeichnet die erfolgreichen Navigationsschritte auf und generiert eine Skill-Datei, die den erlernten Anmeldeablauf erfasst.

Bei nachfolgenden Läufen liest der Agent die gespeicherte Skill-Datei und wendet das erlernte Navigationsmuster direkt an, wodurch die Erkennungsphase übersprungen wird. Das bedeutet Folgendes:

- Schnellere Durchführung authentifizierter Tests — der Agent wendet bewährte Navigationsmuster sofort an, anstatt alles von Grund auf neu zu untersuchen
- Konsistenteres Verhalten — der Agent folgt demselben bewährten Weg, anstatt nach Alternativen zu suchen

> **Note**
>
> Die Anmeldeoptimierung lernt während der ersten Anmeldesitzung innerhalb eines Laufs. Nachfolgende Anmeldesitzungen im selben Lauf profitieren von dem, was in der

ersten Sitzung gelernt wurde. Bei future Läufen profitieren alle Anmeldesitzungen von der gespeicherten Fähigkeit.

Aktivieren oder deaktivieren Sie die Login-Optimierung

Die Login-Optimierung ist standardmäßig aktiviert. Um es zu deaktivieren oder erneut zu aktivieren:

1. Navigieren Sie in der Pentest-Konfiguration zum Abschnitt Authentifizierungsdaten — optional.
2. Suchen Sie den Schalter „Login-Optimierung“.
3. Um die Login-Optimierung zu aktivieren, schalten Sie den Schalter ein. Um ihn zu deaktivieren, schalten Sie ihn aus.

Wenn diese Option deaktiviert ist, erkennt der Agent den Anmeldefluss bei jedem Lauf von Grund auf neu. Zuvor erlernte Navigationsfähigkeiten bleiben erhalten und werden erneut angewendet, wenn die Funktion erneut aktiviert wird.

> **Tip**
>
> Wenn sich der Anmeldeablauf Ihrer Anwendung erheblich ändert (z. B. eine neu gestaltete Anmeldeseite oder ein neuer Authentifizierungsschritt), passt sich der Agent automatisch an, indem er seine erlernten Fähigkeiten beim nächsten Lauf aktualisiert. Sie müssen die Optimierung nicht manuell zurücksetzen.

Korrigieren Sie ein Ergebnis eines Penetrationstests

Wenn Sie sich die Ergebnisse eines Penetrationstests ansehen, können Sie den AWS Security Agent auffordern, einen Fehler zu korrigieren. Für private GitHub Repositorys öffnet der AWS Security Agent eine Pull-Anfrage mit dem vorgeschlagenen Fix. Für öffentliche Repositorys ist die Problembehebung als herunterladbare Diff-Datei verfügbar, die Sie lokal anwenden können.

Sie müssen die Suche nach Problembehebungen in der AWS-Managementkonsole aktivieren. (Siehe the section called “Ermöglichen Sie es Benutzern, mit der Behebung der Ergebnisse von Penetrationstests und Code-Reviews zu beginnen”). Benutzer können in der AWS Security Agent Web App mit der Behebung eines bestimmten Fehlers beginnen.

Voraussetzungen

Stellen Sie vor Beginn sicher, dass Sie über Folgendes verfügen:

- Ein abgeschlossener oder laufender Penetrationstest
- Zugriff auf die AWS Security Agent-Webanwendung
- Vertrautheit mit der Architektur und den Sicherheitsanforderungen Ihrer Anwendung

Schritt 1: Automatische Problembehebung aktivieren oder deaktivieren

Sie können Optionen zur Codekorrektur konfigurieren, wenn Sie einen Penetrationstest erstellen oder ändern. Wenn Sie die automatische Wiederherstellung aktivieren, versucht der AWS Security Agent automatisch, die zugehörigen GitHub Repositorys zu korrigieren, wenn der Agent während des Pentests ein Ergebnis bestätigt. Sie können die Codekorrektur auch manuell starten. Aktivieren oder deaktivieren Sie in der Ansicht zur Bearbeitung der Details des Penetrationstests im Abschnitt Automatische Codekorrektur die Codekorrektur.

Schritt 2: Wählen Sie Repositorys für die Codekorrektur aus

1. Wählen Sie Weiter bis zum letzten Schritt Zusätzliche Lernressourcen.
2. Wählen Sie „Aus Ressourcen auswählen“.
3. Wählen Sie GitHub Repositorys aus.
4. Wählen Sie die Repositorys aus, die Sie für die Codekorrektur verwenden möchten.
5. Speichern Sie den Penetrationstest.
6. Sie können die erfolgreich verknüpften Repositorien auf der Registerkarte Lernressourcen für Penetrationstests einsehen.

Schritt 3: Starten Sie einen Penetrationstest und sehen Sie sich die Ergebnisse an

Führen Sie den Penetrationstest durch, um die Ergebnisse zu ermitteln. Weitere Informationen finden Sie unter the section called “Überprüfen Sie die Ergebnisse eines Penetrationstests”.

Schritt 4: Starten Sie die Codekorrektur und sehen Sie sich diese an

1. Navigieren Sie zum Ergebnis.
2. Wenn Sie die automatische Codekorrektur aktiviert haben, wird eine Codekorrektur gestartet, sobald der AWS Security Agent einen Befund bestätigt.

3. Wenn Sie eine Codekorrektur manuell starten möchten, wählen Sie die Schaltfläche Code korrigieren.
4. Im Abschnitt „Codekorrektur“ des Ergebnisses können Sie den Status der Codekorrektur und Links zu den Pull Requests einsehen. Wenn das GitHub Repository öffentlich ist, ist die Codekorrektur als herunterladbare Datei statt als Pull-Request verfügbar. Sie können ausführen`git apply /path/to/code_remediation_changes.diff`, um die Änderung lokal auf Ihr Repository anzuwenden.

Sicherheit und Referenz

Sicherheitsrichtlinien, Servicekontingenten und der Dokumentverlauf für AWS Security Agent.

Sicherheit im AWS Security Agent

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der der AWS Security Agent in der AWS Cloud ausgeführt wird. Dazu gehören die Service-Infrastruktur, KI-Modelle und Penetrationstest-Agenten. Third-party Auditoren testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheit im Rahmen der AWS Compliance-Programme.
- Sicherheit in der Cloud – Ihre Verantwortlichkeit umfasst die folgenden Bereiche.
  - Verwaltung des Zugriffs auf AWS Security Agent mithilfe von IAM-Richtlinien und -Berechtigungen
  - Schutz der Inhalte, die Sie für den Service bereitstellen, einschließlich Designdokumente, Code-Repositorys und Anwendungs-URLs für Penetrationstests
  - Konfiguration der überwachten Repositorys und Anwendungen
  - Überprüfung der vom Dienst bereitgestellten Sicherheitserkenntnisse und entsprechende Maßnahmen
  - Sicherung Ihrer Anwendungen auf der Grundlage der bereitgestellten Anleitungen zur Problembehebung
  - Die Vertraulichkeit Ihrer Daten, die Anforderungen Ihres Unternehmens und geltende Gesetze und Vorschriften

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von AWS Security Agent anwenden können.

Sicherheitsüberlegungen für AWS Security Agent und KI-gestützte Penetrationstests

AWS Security Agent ist ein Frontier Agent, der Ihre Anwendungen während des gesamten Entwicklungszyklus in all Ihren Umgebungen proaktiv schützt. Er führt automatisierte Sicherheitsüberprüfungen durch, die auf Ihre Anforderungen zugeschnitten sind, wobei Sicherheitsteams zentral Standards definieren, die bei Überprüfungen automatisch validiert werden. Der Security Agent führt bei Bedarf Penetrationstests durch, die auf Ihre Anwendung zugeschnitten sind. Dabei werden verifizierte Sicherheitsrisiken erkannt und gemeldet. Dieser Ansatz skaliert das Sicherheits-Know-How für alle Ihre Anwendungen, um sie an die Geschwindigkeit der Entwicklung anzupassen und gleichzeitig einen umfassenden Sicherheitsschutz zu bieten. Durch die Integration von Sicherheit vom Design bis zur Bereitstellung trägt er dazu bei, Sicherheitslücken frühzeitig und in großem Umfang zu verhindern.

Sicherheitsteams definieren die organisatorischen Sicherheitsanforderungen einmal in der AWS-Konsole: genehmigte Autorisierungsbibliotheken, Protokollierungsstandards und Datenzugriffsrichtlinien. AWS Security Agent setzt diese Sicherheitsanforderungen während der gesamten Entwicklung automatisch durch, bewertet Architekturdokumente und Code anhand Ihrer Standards und bietet spezifische Hinweise, wenn Verstöße festgestellt werden. Auf diese Weise wird eine konsistente Durchsetzung der Sicherheitsvorkehrungen in allen Teams gewährleistet und die Überprüfungen werden an die Geschwindigkeit der Entwicklung angepasst.

Für die Validierung der Bereitstellung verwandelt der AWS Security Agent Penetrationstests von einem regelmäßigen Engpass in eine On-Demand-Funktion. Sicherheitsteams stellen Ziel-URLs, Authentifizierungsdetails, Quellcode und Dokumentation zur Verfügung. AWS Security Agent entwickelt ein tiefes Anwendungsverständnis und führt ausgeklügelte Angriffsketten durch, um Sicherheitslücken zu entdecken und zu validieren, sodass Teams bei Bedarf Tests durchführen können.

Die wichtigsten Funktionen

AWS Security Agent bietet umfassende Sicherheitsfunktionen, die sich über den gesamten Entwicklungszyklus erstrecken.

Sicherheitsüberprüfung entwerfen

Der AWS Security Agent bietet auf Anfrage Sicherheitsfeedback zu Designdokumenten und bewertet die Einhaltung der organisatorischen Sicherheitsanforderungen, bevor der Code geschrieben wird. Sicherheitsteams laden Entwurfsdokumente über die Webanwendung hoch, wo der Agent sie anhand

Ihrer Sicherheitsanforderungen analysiert und Ergebnisse mit Anleitungen zur Problembehebung präsentiert. Dadurch werden stundenlange manuelle Überprüfungen schneller zu gezielten Analysen, sodass die Teams Sicherheitsbedenken dann angehen können, wenn die Behebung am effizientesten ist.

Überprüfung der Codesicherheit

AWS Security Agent analysiert Pull-Anfragen oder hochgeladenen Code auf organisatorische Sicherheitsanforderungen und allgemeine Sicherheitsprobleme wie fehlende Eingabevalidierung und SQL-Injection-Risiken. Der Agent bietet Anleitungen zur Problembehebung direkt in Ihrer Code-Repository-Plattform. Sicherheitsteams konfigurieren, welche Repositorys überwacht werden sollen, und skalieren die Evaluierung auf alle Codebasen, während sie gleichzeitig den Überblick über kritische Probleme behalten.

On-demand Penetrationstests

AWS Security Agent bietet Penetrationstests auf Abruf, mit denen validierte Sicherheitslücken anhand maßgeschneiderter mehrstufiger Angriffsszenarien entdeckt und gemeldet werden. AWS Security Agent setzt spezialisierte KI-Agenten ein, die anhand der bereitgestellten Dokumentation und Anmeldeinformationen den Anwendungskontext entwickeln und dann ausgeklügelte Angriffsketten ausführen, um komplexe Sicherheitslücken zu identifizieren, die herkömmlichen Tools entgehen. Er dokumentiert die Ergebnisse mit Auswirkungsanalysen, reproduzierbaren Angriffspfaden und sofort einsatzbereiten Codekorrekturen. Dadurch werden Penetrationstests von Wochen auf Stunden beschleunigt und die Validierung für Ihr gesamtes Anwendungsportfolio skaliert.

Häufig gestellte Fragen

Sicherheitskontrolle &

Wie authentifiziert und verwaltet der AWS Security Agent den Zugriff auf Systeme?

Penetrationstests sind die einzige Funktion in AWS Security Agent, die sich zur Laufzeit beim System eines Benutzers authentifizieren kann. Der AWS Security Agent akzeptiert Anmeldeinformationen in Form eines statischen Benutzernamens und Kennworts (gespeichert in Secrets Manager) oder eines Anbieters von Anmeldeinformationen (als Lambda-Funktion) als Konfiguration, bevor der Pen-Test gestartet wird. Diese Anmeldeinformationen werden verwendet, um die normale Funktionalität des Benutzers während des gesamten system/application Lebenszyklus des Pen-Tests auszuüben. Wir empfehlen Benutzern, für Pentesting-Zwecke neue Anmeldeinformationen mit entsprechend eingeschränkten Berechtigungen zu erstellen.

Können Benutzer den Umfang und die Tiefe der Tests kontrollieren, um unbeabsichtigte Auswirkungen auf das System zu verhindern?

Mit dem AWS Security Agent können Kunden eine bestimmte Kategorie von Sicherheitslücken auswählen, die auf einem Endpunkt untersucht werden sollen. Benutzer können URLs angeben, die außerhalb des Gültigkeitsbereichs liegen, um zu verhindern, dass der AWS Security Agent Penetrationstests für diese Ziele durchführt. https://docs.aws.amazon.com/securityagent/latest/userguide/perform-penetration-test.html

Kann der AWS Security Agent selbst ein Sicherheitsrisiko darstellen?

Der AWS Security Agent wird angewiesen, Sicherheitsrisiken zu erkennen, dabei aber bewusst nur minimale Nutzlasten zu verwenden (z. B. die SQL-Version zu extrahieren, anstatt eine Tabelle zu löschen, wenn ein SQL-Injection-Angriff entdeckt wird). AWS Security Agent ist außerdem auf deterministische Schutzmaßnahmen beschränkt, um riskantes Verhalten wie eine übermäßige Belastung der Zielanwendung zu verhindern. Obwohl Schutzmaßnahmen vorhanden sind, kann es dennoch zu unbeabsichtigten oder nicht offensichtlichen Interaktionen mit der Geschäftslogik kommen. Daher empfehlen wir immer, Penetrationstests in einer Vorproduktionsumgebung durchzuführen.

Welche Daten sammelt der AWS Security Agent und wo werden sie gespeichert?

Mit dem AWS Security Agent können Benutzer Artefakte hochladen, um einen Kontext zu ihrer getesteten Anwendung bereitzustellen. Weitere Informationen zum Datenschutz finden Sie unterthe section called “Datenschutz”. AWS Security Agent wählt automatisch die optimale Region innerhalb Ihrer Region für die Verarbeitung Ihrer Inferenzanfragen aus. Dies maximiert die verfügbaren Rechenressourcen und die Modellverfügbarkeit und sorgt für ein optimales Kundenerlebnis. Ihre Daten werden nur in der Region gespeichert, aus der die Anfrage stammt. Eingabeaufforderungen und Ausgabeergebnisse können jedoch außerhalb dieser Region verarbeitet werden. Alle Daten werden bei der Übertragung über das sichere Netzwerk von Amazon verschlüsselt. Weitere Informationen finden Sie unter Regionsübergreifende Inferenz.

Welche Kontrollen stehen zur Verfügung, um unautorisierte Tests an einem Endpunkt zu blockieren?

Endpunkte, die als Ziel-URLs für Pentests angegeben sind, erfordern eine DNS- oder HTTP-Validierung als Eigentumsnachweis. Der AWS Security Agent fordert den Kunden auf, einen TXT-Eintrag zum DNS des Endpunkts hinzuzufügen oder eine HTTP-Route, die eine Bestätigungszeichenfolge zurückgibt, als Eigentumsnachweis verfügbar zu machen. Erst nach dem Nachweis des Besitzes kann der Benutzer mit einem Pentest fortfahren. Anfragen an URLs außerhalb des Ziels und an URLs, auf die zugegriffen werden kann, werden vom Netzwerk blockiert.

Die Kunden sind dafür verantwortlich, sicherzustellen, dass sie über die erforderliche Autorisierung verfügen, um alle Systeme zu testen, die von ihren Penetrationstests betroffen sein könnten. Jegliche Nutzung von AWS Security Agent muss der AWS-Richtlinie zur akzeptablen Nutzung (https://aws.amazon.com/aup/) entsprechen.

Wie blockieren und melden Benutzer jeglichen Missbrauch mithilfe von AWS Security Agent?

AWS Security Agent überwacht kontinuierlich Anfragen und versucht, auf URLs zuzugreifen, die sich außerhalb der Ziel-URLs befinden. Wenn ein Missbrauch festgestellt wird, z. B. der Versuch, den AWS Security Agent zu verwenden, um unbefugte Tests an einem Endpunkt eines Drittanbieters durchzuführen, werden alle laufenden Pentests im Konto beendet. Kunden können sich an den AWS-Support oder ihr AWS-Kundenbetreuungsteam wenden, um Hilfe zu erhalten.

Kann der AWS Security Agent den Pen-Testing-Workflow ersetzen?

AWS Security Agent ist kein professioneller Penetrationstest-Service, und wir empfehlen Benutzern, AWS Security Agent in ihren Sicherheitsüberprüfungs-Workflow zu integrieren. AWS Security Agent kann während der Entwicklungsphase des Softwarelebenszyklus Zugriff auf Penetrationstests auf Abruf bieten, wenn die Zusammenarbeit mit Pentesting-Experten zu früh oder unpraktisch wäre oder zu häufig neu bewertet werden müsste. Sicherheitsexperten können die Ergebnisse von AWS Security Agent überprüfen, um sie zu validieren, zu erläutern oder um neue Erkenntnisse zu erweitern (falls vorhanden).

Können Benutzer die rollenbasierte Zugriffskontrolle (RBAC) für verschiedene Teammitglieder einrichten?

Ja. AWS Security Agent ist in das AWS IAM Identity Center integriert, sodass Administratoren Teammitglieder verwalten können, die Zugriff auf die AWS Security Agent-Webanwendung haben, mit der Benutzer Entwurfsprüfungen und Pentests erstellen, verwalten und anzeigen können.

Testmöglichkeiten

Welche Arten von Sicherheitslücken kann der AWS Security Agent erkennen?

Der AWS Security Agent erkennt Sicherheitslücken in den OWASP Top 10 für Webanwendungen. AWS Security Agent bietet spezifische Risikotypen, die Sie in die unten beschriebenen Tests einbeziehen oder ausschließen können. Die Ergebnisse können sich innerhalb dieser Risikokategorien oder aufgrund neuer Erkenntnisse ergeben, die anhand von Hinweisen aus einer Kombination dieser Risikokategorien entdeckt wurden.

- Beliebiger Datei-Upload
  - Durch das Hochladen beliebiger Dateien wird bestätigt, dass das Programm in der Lage sein sollte, gefälschte und bösartige Dateien abzuwehren, um die Anwendung und die Benutzer zu schützen
- Code-Injektion
  - Code Injection ist der allgemeine Begriff für Angriffsarten, die darin bestehen, Code einzuschleusen, der dann interpreted/executed von der Anwendung ausgeführt wird
- Befehlseinschleusung
  - Die Befehlsinjektion ist ein Angriff, bei dem das Ziel darin besteht, beliebige Befehle auf dem Host-Betriebssystem über eine anfällige Anwendung auszuführen
- Cross-Site Scripting (XSS)
  - Cross-Site Scripting-Angriffe (XSS) sind eine Art von Injektion, bei der bösartige Skripts in ansonsten harmlose und vertrauenswürdige Websites eingeschleust werden
- Unsichere direkte Objektreferenz
  - Unsichere Direct Object References (IDOR) treten auf, wenn eine Anwendung auf der Grundlage von Benutzereingaben direkten Zugriff auf Objekte ermöglicht
- Sicherheitslücken in JSON Web Token
  - JWTs sind eine häufige Quelle von Sicherheitslücken, sowohl in Bezug auf die Art und Weise, wie sie in Anwendungen implementiert werden, als auch in den zugrunde liegenden Bibliotheken
- Inklusion lokaler Dateien
  - Die Sicherheitsanfälligkeit beim Einfügen von Dateien ermöglicht es einem Angreifer, eine Datei einzufügen, wobei er in der Regel einen in der Zielanwendung implementierten Mechanismus zur „dynamischen Dateieinbindung“ ausnutzt
- Durchquerung von Pfaden
  - Der Path Traversal-Angriff (auch als Directory Traversal bezeichnet) zielt darauf ab, auf Dateien und Verzeichnisse zuzugreifen, die außerhalb des Webstammordners gespeichert sind
- Eskalation von Rechten
  - Eine Rechteeskalation tritt auf, wenn ein Benutzer Zugriff auf mehr Ressourcen oder Funktionen erhält, als ihm normalerweise erlaubt sind, und eine solche Erhöhung oder Änderung hätte durch die Anwendung verhindert werden müssen
- Server-Side Forgery anfordern (SSRF)
  - Server-Side Request Forgery (SSRF) tritt auf, wenn der Angreifer Funktionen auf dem Server missbrauchen kann, um interne Ressourcen zu lesen oder zu aktualisieren

- Server-Side Template-Injektion
  - Sicherheitslücken durch serverseitige Template-Injection (SSTI) treten auf, wenn Benutzereingaben auf unsichere Weise in eine Vorlage eingebettet werden und zur Remotecodeausführung auf dem Server führen
- SQL Injection
  - Ein SQL-Injection-Angriff besteht aus dem Einfügen oder „Injizieren“ einer SQL-Abfrage über die Eingabedaten vom Client in die Anwendung
- Externe XML-Entität
  - Ein Angriff auf eine externe XML-Entität ist ein Angriff auf eine Anwendung, die XML-Eingaben analysiert. Dieser Angriff tritt auf, wenn XML-Eingaben, die einen Verweis auf eine externe Entität enthalten, von einem schwach konfigurierten XML-Parser verarbeitet werden

Welche Authentifizierungsmethoden unterstützt der AWS Security Agent?

AWS Security Agent unterstützt gängige Authentifizierungsmethoden, einschließlich OAuth und JWT. Weitere Informationen finden Sie in der -Dokumentation.

Wie geht der AWS Security Agent mit der Ratenbegrenzung und der Verhinderung von Denial of Service (DOS) um?

Der AWS Security Agent verfügt über Schutzmaßnahmen, um zu verhindern, dass er getestete Endgeräte, einschließlich DOS, unterbricht oder herunterfährt. Er verfügt über interne Geschwindigkeitskontrollen, um unerwartete Verkehrsmuster zu erkennen und zu behandeln.

Kann der AWS Security Agent sowohl REST- als auch GraphQL-APIs testen?

Ja, der AWS Security Agent kann API-Endpunkte testen. Wir empfehlen unseren Kunden, API-Dokumentation als zusätzliche Lernressourcen zur Verfügung zu stellen, damit der AWS Security Agent einen besseren Überblick über die Form und Funktionalität der einzelnen getesteten APIs erhält.

Wie können Benutzer überprüfen, ob der AWS Security Agent alle kritischen Anwendungslogik und Endpunkte abgedeckt hat?

Der AWS Security Agent führt zunächst eine umfassende Untersuchung der Zielanwendung (en) durch und versucht, sie normal auszuführen, bevor es zu Exploits kommt. Auf diese Weise kann er sich während der Laufzeit ein fundiertes Verständnis der Anwendung aneignen und kritische Anwendungslogik und Endpunkte erkennen. Aufgrund seines stochastischen Charakters kann nicht

garantiert werden, dass der AWS Security Agent alle kritischen Anwendungen und Endpunkte für eine Zielanwendung erkennt und testet. Die AWS Security Agent-Webanwendung bietet Einblick in alle erkannten Endpunkte und die in den Penetrationstestprotokollen durchgeführten Aktionen.

Genauigkeit und Zuverlässigkeit&.

Wie validiert der AWS Security Agent die Ergebnisse vor der Berichterstattung?

AWS Security Agent verwendet deterministische Validatoren, um das gemeldete Ergebnis zu validieren. Bei Risikoarten, bei denen es nicht möglich ist, deterministische Validatoren zu verwenden, wiederholt der AWS Security Agent die Ermittlungsschritte selbstständig, um Vertrauen in die Gültigkeit des Ergebnisses zu gewinnen. AWS Security Agent meldet nur die Ergebnisse mit hoher oder mittlerer Zuverlässigkeit und versteckt die nicht verifizierten Ergebnisse standardmäßig.

Kann sich AWS Security Agent an die benutzerdefinierte Anwendungslogik anpassen?

AWS Security Agent akzeptiert optional Quellcode, Bedrohungsmodell, Entwurfsdokumente und API-Dokumentation als zusätzliche Lernressourcen, um benutzergesteuerten Kontext zu der Zielanwendung zu erhalten, die im Lebenszyklus eines Pentests verwendet wird.

Können Benutzer die Testmethodik von AWS Security Agent vor der Ausführung überprüfen?

Derzeit gibt es keine Möglichkeit, eine Vorschau der Vorgehensweise von AWS Security Agent anzuzeigen. Der AWS Security Agent-Plan ist dynamischer Natur und basiert auf der Erkundung der Zielanwendung. Kunden können den AWS Security Agent während seiner Erkundung in Echtzeit überwachen, indem sie die Protokolle der Penetrationstests beobachten. Wenn die Protokolle einen ungültigen oder unerwünschten Verlauf anzeigen, können Kunden den laufenden Pentest-Lauf beenden.

Integration und Bereitstellung &

Lässt sich AWS Security Agent in Sicherheitstools (SIEM, Vulnerability Management) oder CI/CD Pipelines integrieren?

AWS Security Agent lässt sich nicht in bestehende Sicherheitstools oder CI/CD Pipelines integrieren.

Wie geht AWS Security Agent mit umgebungsspezifischen Konfigurationen um?

Der AWS Security Agent kann so konfiguriert werden, dass er mit bestimmten IAM-Rollen, innerhalb von VPCs, mit kundenspezifischen anwendungsrelevanten Anmeldeinformationen und mit Github-Quell-Repositorys als Quellcode-Referenz für die Zielanwendung ausgeführt wird.

Kann AWS Security Agent in Air-Gapped- oder isolierten Umgebungen ausgeführt werden?

Der AWS Security Agent kann so konfiguriert werden, dass er Konnektivität zu VPCs bietet, auch zu solchen, die keinen ausgehenden Internetzugang haben.

Können mehrere Teammitglieder gleichzeitig Tests durchführen?

AWS Security Agent unterstützt 5 gleichzeitige Pentest-Läufe pro Konto, unabhängig davon, wer den Test startet. Kunden können maximal 100 Agent Spaces und 1.000 Pentest-Projekte erstellen.

Operative Auswirkungen

Wie wirkt sich das auf die Leistung der getesteten Systeme aus?

Der AWS Security Agent verfügt über Schutzmaßnahmen, um zu verhindern, dass er die zu testenden Endgeräte unterbricht oder herunterfährt. Dazu gehören Geschwindigkeitskontrollen für die Anzahl der Anrufe, die der AWS Security Agent an einen Endpunkt tätigen kann. Das System oder der zu testende Endpunkt sollte aufgrund der Pen-Testaktivität mit einem gewissen Anstieg des Datenverkehrs und der Auslösung potenzieller Überwachungswarnungen rechnen. Wir empfehlen, AWS Security Agent oder andere Pen-Testing-Aktivitäten nur in einer Vorproduktionsumgebung auszuführen.

Können Benutzer den AWS Security Agent planen oder drosseln?

AWS Security Agent verfügt weder über öffentliche APIs noch über die Möglichkeit, die Pen-Testläufe zu planen. AWS Security Agent bietet auch keine Parallelitätskontrolle für Anfragen an den Zielendpunkt, wenn der Pen-Testlauf gestartet wird. Wenn der AWS Security Agent Probleme mit Zielendpunkten verursacht, können Kunden die laufenden Pentests beenden.

Was ist die typische Dauer einer vollständigen Sicherheitsbeurteilung?

Die Laufzeit jedes Pentests hängt von der Breite der Zielanwendung und den Risikotypen ab, die für die Bewertung konfiguriert wurden. Die meisten Pentest-Läufe sind innerhalb von 16 Stunden abgeschlossen.

Von AWS verwaltete Richtlinien für AWS-Sicherheitsagenten

Bei einer von AWS verwalteten Richtlinie handelt es sich um eine eigenständige Richtlinie, die von AWS erstellt und verwaltet wird. Von AWS verwaltete Richtlinien sind so konzipiert, dass sie Berechtigungen für viele gängige Anwendungsfälle bereitstellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass von AWS verwaltete Richtlinien möglicherweise keine Berechtigungen mit den geringsten Rechten für Ihre speziellen Anwendungsfälle gewähren, da sie für alle AWS-Kunden verfügbar sind. 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 den verwalteten AWS-Richtlinien definierten Berechtigungen nicht ändern. Wenn AWS die in einer von AWS verwalteten Richtlinie definierten Berechtigungen aktualisiert, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. Es ist sehr wahrscheinlich, dass AWS eine von AWS verwaltete Richtlinie aktualisiert, wenn ein neuer AWS-Service eingeführt wird oder neue API-Operationen für bestehende Services verfügbar werden.

Weitere Informationen finden Sie unter Von AWS verwaltete Richtlinien im IAM-Benutzerhandbuch.

Von AWS verwaltete Richtlinie: SecurityAgentWebAppAPIPolicy

Erteilt Berechtigungen zur Interaktion mit der Security Agent Web Application API. Diese Richtlinie ermöglicht es Benutzern, automatisierte Sicherheits-Penetrationstests zu konfigurieren und auszuführen, Testausführungen zu verwalten, Sicherheitsergebnisse einzusehen und auf Security Agent-Ressourcen zuzugreifen. Diese Richtlinie bezieht sich auf den alten Agent-Instance-Ressourcentyp und auf bestimmte ältere IAM-Aktionen.

Details zu Berechtigungen

Diese Richtlinie gewährt Berechtigungen zur Interaktion mit dem Security Testing Control-Dienst (securityagent: *) für:

- Pentest Management: Penetrationstests und deren Ausführung erstellen, aktualisieren, löschen und auflisten
- Sicherheitsergebnisse: Sehen Sie sich Sicherheitsergebnisse aus abgeschlossenen Tests an, beschreiben und aktualisieren Sie sie, einschließlich verwandter Inhalte und Metadaten.
- Aufgabenverwaltung: Aufgaben zur Codeüberprüfung und Dokumentationsprüfung auflisten und abrufen
- Ressourcenerkennung: Listet Agenteninstanzen, Artefakte, Integrationen und entdeckte Endpunkte auf und zeigt sie an
- Testausführung: Starten und beenden Sie Pentest-Ausführungen mit Funktionen zur Echtzeitüberwachung

- Codekorrektur: Starten Sie die automatische Codekorrektur im Hinblick auf Sicherheitslücken

Die neueste Version des JSON-Richtliniendokuments finden Sie SecurityAgentWebAppAPIPolicyim AWS Managed Policy Reference Guide.

Von AWS verwaltete Richtlinie: AWSSecurityAgentWebAppPolicy

Erteilt Berechtigungen zur Interaktion mit der Security Agent Web Application API. Diese Richtlinie ermöglicht es Benutzern, automatisierte Sicherheits-Penetrationstests zu konfigurieren und auszuführen, Testausführungen zu verwalten, Sicherheitsergebnisse einzusehen und auf Security Agent-Ressourcen zuzugreifen.

Details zu Berechtigungen

Diese Richtlinie gewährt Berechtigungen zur Interaktion mit dem Security Testing Control-Dienst (securityagent: *) für:

- Pentest Management: Penetrationstests und ihre Aufgaben erstellen, aktualisieren, löschen und auflisten
- Sicherheitsergebnisse: Sehen Sie sich Sicherheitsergebnisse aus abgeschlossenen Tests an, beschreiben und aktualisieren Sie sie, einschließlich zugehöriger Inhalte und Metadaten.
- Aufgabenverwaltung: Aufgaben zur Codeüberprüfung und Entwurfsprüfung auflisten und abrufen
- Ressourcenerkennung: Listet Agentenbereiche, Artefakte, Integrationen und entdeckte Endpunkte auf und zeigt sie an
- Testausführung: Starten und beenden Sie Pentest-Jobs mit Funktionen zur Echtzeitüberwachung
- Codekorrektur: Starten Sie die automatische Codekorrektur im Hinblick auf Sicherheitslücken

Die neueste Version des JSON-Richtliniendokuments finden Sie AWSSecurityAgentWebAppPolicyim AWS Managed Policy Reference Guide.

Aktualisierungen der verwalteten AWS-Richtlinien durch AWS Security Agents

Sehen Sie sich Details zu Aktualisierungen der von AWS verwalteten Richtlinien für AWS-Sicherheitsagenten an, seit dieser Service begonnen hat, diese Änderungen zu verfolgen.

Um Benachrichtigungen über alle Änderungen an den Quelldateien dieser spezifischen Dokumentationsseite zu erhalten, können Sie die folgende URL mit einem RSS-Reader abonnieren:

Änderungen	Beschreibung	Date
Berechtigungen zu AWSSecurityAgentWebAppPolicy hinzugefügt.	Hinzugefügt `TargetDomain` und `DesignReviewFeedback` Ressourcenberechtigungen für die neuen Ressourcentypen.	31. März 2026
Eine neue verwaltete Richtlinie AWSSecurityAgentWebAppPolicywurde hinzugefügt.	Es wurde eine verwaltete Richtlinie `AWSSecurityAgentWebAppPolicy` für den neuen AgentSpace Ressourcentyp und die Änderung des Namens der IAM-Aktion hinzugefügt.	9. Februar 2026
Berechtigungen zu SecurityAgentWebAppAPIPolicy hinzugefügt.	Es wurde hinzugefügt`securityagent:StartCodeRemediation` , um Benutzern die Möglichkeit zu geben, automatische Codekorrekturen für Sicherheitslücken zu starten.	20. Januar 2026
Berechtigungen zu SecurityAgentWebAppAPIPolicy hinzugefügt.	Es wurde hinzugefügt`securityagent:BatchGetSecurityTestContentMetadata` , um Benutzern das Ansehen von Bildern in der Konsole zu ermöglichen.	5. Dezember 2025
Die AWS-Sicherheitsagenten haben begonnen, Änderungen nachzuverfolgen.	Die AWS-Sicherheitsagenten haben damit begonnen, Änderungen an ihren von AWS verwalteten Richtlinien nachzuverfolgen.	2. Dezember 2025

Datenschutz im AWS Security Agent

Das AWS-Modell der gemeinsamen Verantwortung gilt für den Datenschutz in AWS Security Agent. Wie in diesem Modell beschrieben, ist AWS verantwortlich für den Schutz der globalen Infrastruktur, auf der die gesamte AWS Cloud ausgeführt wird. 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 AWS-Services verantwortlich, die Sie verwenden. Informationen zum Datenschutz in Europa finden Sie im Blogbeitrag AWS Shared Responsibility Model und GDPR auf dem AWS-Sicherheitsblog. Aus Datenschutzgründen empfehlen wir Ihnen, die AWS-Kontoanmeldedaten zu schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einzurichten. 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 SSL/TLS für die Kommunikation mit AWS-Ressourcen verwendet. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API- und Benutzeraktivitätsprotokollierung mit AWS ein CloudTrail. Informationen zur Verwendung von CloudTrail Trails zur Erfassung von AWS-Aktivitäten finden Sie unter Arbeiten mit CloudTrail Pfaden im CloudTrail AWS-Benutzerhandbuch.
- Verwenden Sie AWS-Verschlüsselungslösungen zusätzlich zu allen standardmäßigen Sicherheitskontrollen innerhalb von 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 auf 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 dem AWS Security Agent oder anderen AWS-Services über die Konsole, die API, die AWS-CLI oder die AWS-SDKs arbeiten. 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.

Verschlüsselung im Ruhezustand

AWS Security Agent verschlüsselt standardmäßig alle Daten im Ruhezustand mit AWS-managed Verschlüsselungsschlüsseln. Dies umfasst:

- Designdokumente und Code — Alle Designdokumente, Code-Repositorys und Anwendungsartefakte, die Sie für Sicherheitsüberprüfungen bereitstellen, werden verschlüsselt AES-256 .
- Sicherheitsergebnisse — Alle Sicherheitsergebnisse, Schwachstellenberichte und Empfehlungen zur Behebung von Sicherheitslücken werden im Ruhezustand verschlüsselt.
- Konfigurationsdaten — Sicherheitsanforderungen, benutzerdefinierte Richtlinien und Dienstkonfigurationen sind verschlüsselt.
- Audit-Logs — Alle Dienstaktivitätsprotokolle und Audit-Trails sind verschlüsselt.

AWS Security Agent verwendet den AWS Key Management Service (AWS KMS) zur Verwaltung von Verschlüsselungsschlüsseln. Sie können optional einen vom Kunden verwalteten Schlüssel verwenden, um Ihre Daten zu verschlüsseln, sodass Sie die volle Kontrolle über die Verschlüsselungsschlüssel haben, die Ihre Ressourcen schützen. Weitere Informationen finden Sie unter the section called “Kundenseitig verwaltete Schlüssel”.

Verschlüsselung während der Übertragung

AWS Security Agent verschlüsselt alle Daten während der Übertragung mit Transport Layer Security (TLS) 1.2 oder höher. Dies gilt in folgenden Fällen:

- API-Kommunikation — Alle API-Aufrufe zwischen Ihren Anwendungen und dem AWS Security Agent verwenden HTTPS mit TLS-Verschlüsselung.
- Konsolenzugriff — Auf die AWS Security Agent-Konsole wird über HTTPS zugegriffen.
- Repository-Verbindungen — Verbindungen zu GitHub und anderen Code-Repositorys verwenden verschlüsselte Protokolle.
- Agentenkommunikation — Die gesamte Kommunikation zwischen dem AWS Security Agent Service und den Penetrationstest-Agenten erfolgt über verschlüsselte Kanäle.

Schlüsselverwaltung

AWS Security Agent verwendet den AWS Key Management Service (AWS KMS) zur Verwaltung von Verschlüsselungsschlüsseln. Standardmäßig werden Daten mit AWS-managed Schlüsseln verschlüsselt. Sie können optional einen vom Kunden verwalteten Schlüssel angeben, wenn Sie Ressourcen wie Agent Spaces und Integrationen erstellen. Weitere Informationen finden Sie unter the section called “Kundenseitig verwaltete Schlüssel”.

Richtlinie für den Datenverkehr zwischen Netzwerken

AWS Security Agent verwendet das öffentliche Internet, um mit Cloud-gehosteten Quellcodekontrollanbietern (GitHub, GitLab, Bitbucket) und Confluence Cloud zu kommunizieren.

Für selbst gehostete Anbieter (GitLab Self-Managed GitHub Enterprise Server) können Sie private Verbindungen mit Amazon VPC Lattice konfigurieren, um den gesamten Datenverkehr innerhalb des AWS-Netzwerks zu halten. Weitere Informationen finden Sie unter the section called “Connect zur privat gehosteten Quellcodeverwaltung her”.

In der Standardkonfiguration verwendet AWS Security Agent das öffentliche Internet, um Ihre App für Penetrationstests zu erreichen. Sie können optional Penetrationstests so konfigurieren, dass eine VPC für den Zugriff auf Ihre Anwendung verwendet wird. Weitere Informationen finden Sie unter the section called “Agent mit privaten VPC-Ressourcen verbinden”.

Cross-Region Datenverarbeitung

AWS Security Agent verwendet regionsübergreifende Inferenz, um die verfügbaren Rechenressourcen und die Modellverfügbarkeit zu optimieren. Je nach Region, aus der die Anfrage stammt, verarbeiten wir möglicherweise Eingabeaufforderungen und Ausgabeergebnisse in einer anderen Region.

- In USA Ost (Nord-Virginia) —`us-east-1`, USA West (Oregon) —`us-west-2`, Asien-Pazifik (Sydney) —`ap-southeast-2`, Asien-Pazifik (Tokio) —`ap-northeast-1`, Europa (Frankfurt) — `eu-central-1` und Europa (Irland) — `eu-west-1` verwendet AWS Security Agent geografische regionsübergreifende Inferenz. Bei den meisten Funktionen erfolgt die Datenverarbeitung innerhalb der geografischen Grenzen (z. B. USA, EU, Australien oder Japan), aus der die Anfrage stammt. Im Rahmen von Code Remediation werden Anfragen aus Australien und Japan in der Europäischen Union bearbeitet. Einzelheiten zum funktionsspezifischen Routing finden Sie in der Tabelle mit regionsübergreifenden Inferenzen.

- Im asiatisch-pazifischen Raum (Mumbai) —`ap-south-1`, im asiatisch-pazifischen Raum (Singapur) — `ap-southeast-1` und in Südamerika (São Paulo) — `sa-east-1` verwendet AWS Security Agent globale regionsübergreifende Inferenz. Wir können Eingabeaufforderungen und Ausgabeergebnisse in jeder kommerziellen AWS-Region verarbeiten.

In allen Fällen bleiben Ihre Daten nur in der Region gespeichert, aus der die Anfrage stammt. Alle Daten, die während regionsübergreifender Operationen übertragen werden, verbleiben im AWS-Netzwerk und werden nicht über das öffentliche Internet übertragen. Wir verschlüsseln Daten bei der Übertragung zwischen AWS-Regionen.

Cross-Region Inferenz ist immer aktiviert und kann nicht deaktiviert werden. Einzelheiten dazu, welche Regionen Anfragen für die einzelnen Funktionen verarbeiten, finden Sie im Abschnitt Regionalübergreifende Inferenz unter. the section called “Bewährte Methoden für die Gewährleistung der Sicherheit”

Löschen von Daten

Wenn Sie Daten aus dem AWS Security Agent löschen:

- Die Daten sind zum Löschen markiert und über den Service nicht mehr zugänglich.
- Die Daten werden innerhalb von 30 Tagen aus allen AWS Security Agent-Systemen gelöscht.

Um Ihre Daten zu löschen

1. Navigieren Sie in der AWS-Konsole zu AWS Security Agent.
2. Wählen Sie die Daten aus, die Sie löschen möchten (Sicherheitsüberprüfungen, Ergebnisse oder benutzerdefinierte Anforderungen).
3. Wählen Sie Löschen und bestätigen Sie den Löschvorgang.

Identitäts- und Zugriffsmanagement für AWS Security Agent

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAM Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um die Ressourcen des AWS Security Agents zu nutzen. IAM ist eine AWS-Service , die Sie ohne zusätzliche Kosten verwenden können.

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie in AWS Security Agent ausführen.

Servicebenutzer — Wenn Sie den AWS Security Agent-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 des AWS Security Agents 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 auf eine Funktion in AWS Security Agent nicht zugreifen können, finden Sie weitere Informationen unterthe section called “Fehlerbehebung bei Identität und Zugriff auf den AWS Security Agent”.

Service-Administrator — Wenn Sie in Ihrem Unternehmen für die Ressourcen des AWS Security Agents verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AWS Security Agent. Es ist Ihre Aufgabe, zu bestimmen, auf welche Funktionen und Ressourcen des AWS Security Agents Ihre Servicebenutzer zugreifen sollen. Anschließend müssen Sie Anfragen an Ihren IAM Administrator senden, um die Berechtigungen Ihrer Servicebenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die grundlegenden Konzepte von zu verstehen IAM. Weitere Informationen darüber, wie Ihr Unternehmen AWS Security Agent nutzen IAM kann, finden Sie unterthe section called “So funktioniert AWS Security Agent 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 AWS Security Agent zu verwalten. Beispiele für identitätsbasierte Richtlinien für den AWS Security Agent, die Sie in verwenden können IAM, finden Sie unter. the section called “Beispiele für identitätsbasierte Richtlinien für AWS Security Agent”

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als Root-Benutzer des AWS-Kontos authentifiziert (angemeldet AWS) sein, an oder IAM-Benutzer, indem Sie eine IAM Rolle übernehmen. AWS empfiehlt, eine IAM-Rolle zu verwenden und die direkte Verwendung des Root-Benutzers zu vermeiden.

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)

Nutzer, die Single-Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als föderierte Identität anmelden, hat Ihr Administrator zuvor einen Identitätsverbund mithilfe von Rollen eingerichtet. IAM Wenn Sie AWS mithilfe eines Verbunds darauf zugreifen, 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 unter So melden Sie sich bei Ihrem AWS-Konto an im Sign-In AWS-Benutzerhandbuch. AWS

Wenn Sie AWS programmgesteuert zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mit Ihren Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode, um Anfragen selbst zu signieren, finden Sie unter Signaturprozess für Signature Version 4 in der Allgemeinen AWS-Referenz.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise auch zusätzliche Sicherheitsinformationen angeben. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. Weitere Informationen finden Sie unter Multi-factor Authentifizierung im AWS IAM Identity Center (Nachfolger von AWS Single Sign-On) User Guide und Using Multi-Factor Authentication (MFA) in AWS im IAM-Benutzerhandbuch.

Root-Benutzer des AWS-Kontos

Wenn Sie zum ersten Mal eine erstellen AWS-Konto, beginnen Sie mit einer Single-Sign-In-Identität, die vollständigen Zugriff auf alle AWS-Services Ressourcen im Konto hat. Diese Identität wird als AWS-Konto-Stammbenutzer bezeichnet. Um auf den Stammbenutzer zuzugreifen, müssen Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, die zur Erstellung des Kontos verwendet wurden. Wir raten ausdrücklich davon ab, den Root-Benutzer für alltägliche Aufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen und verwenden Sie diese, um 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 im Referenzhandbuch zur Kontoverwaltung unter Aufgaben, für die Root-Benutzeranmeldedaten erforderlich sind.

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 (Nachfolger von AWS Single Sign-On).

IAM-Benutzer und Gruppen

Eine IAM-Benutzerist eine Identität innerhalb von Ihnen AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. Wir empfehlen, sich nach Möglichkeit auf temporäre Zugangsdaten zu verlassen IAM-Benutzer , anstatt solche mit langfristigen Zugangsdaten wie Passwörtern und Zugangsschlüsseln zu erstellen. Wenn Sie jedoch spezielle Anwendungsfälle haben, für die langfristige Anmeldeinformationen erforderlich sind, empfehlen wir IAM-Benutzer, dass Sie die Zugriffsschlüssel 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 angibt IAM-Benutzer. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen erleichtern die Verwaltung von Berechtigungen für große Benutzergruppen. Sie könnten beispielsweise eine Gruppe mit dem Namen iamAdmins einrichten und dieser Gruppe Berechtigungen zur Verwaltung von Ressourcen erteilen. IAM

Benutzer sind nicht dasselbe wie 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 Wann sollte eine Rolle IAM-Benutzer (statt einer Rolle) erstellt werden? im IAM-Benutzerhandbuch.

IAM Rollen

Eine IAM Rolle ist eine Identität innerhalb von Ihnen AWS-Konto , für die bestimmte Berechtigungen gelten. Sie ähnelt einer IAM-Benutzer, ist aber keiner bestimmten Person zugeordnet. Sie können

vorübergehend eine IAM Rolle in der übernehmen, AWS Management Console indem Sie die Rollen 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 zur Verwendung von Rollen finden Sie unter IAM Rollen verwenden im IAM-Benutzerhandbuch.

IAM Rollen mit temporären Anmeldeinformationen sind in den folgenden Situationen nützlich:

- 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 im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Um zu steuern, 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 (Nachfolger von AWS Single Sign-On).
- Temporäre IAM-Benutzer Berechtigungen — Ein IAM-Benutzer kann eine IAM Rolle übernehmen, um vorübergehend verschiedene Berechtigungen für eine bestimmte Aufgabe zu übernehmen.
- Cross-account Zugriff — Sie können eine IAM Rolle verwenden, um jemandem (einem vertrauenswürdigen Principal) 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 zum Unterschied zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie im IAM-Benutzerhandbuch unter Unterschiede zwischen IAM Rollen und ressourcenbasierten Richtlinien.
- Cross-service Zugriff — Einige verwenden Funktionen in anderen. AWS-Services AWS-Services Wenn Sie beispielsweise in einem Dienst einen Anruf tätigen, ist es üblich, dass dieser Dienst Anwendungen ausführt Amazon EC2 oder Objekte darin speichert Amazon S3. Ein Service kann dies mithilfe der Berechtigungen des aufrufenden Prinzipals, einer Servicerolle oder einer serviceverknüpften Rolle tun.
  - Hauptberechtigungen — Wenn Sie eine IAM-Benutzer OR-Rolle verwenden, um Aktionen in auszuführen AWS, gelten Sie als Principal. Richtlinien erteilen einem Prinzipal-Berechtigungen. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service auslösen. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen.

  - Servicerolle — Eine Servicerolle ist eine IAM Rolle, die ein Dienst übernimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM Administrator kann eine Servicerolle von innen heraus erstellen, ändern und löschen IAM. Weitere Informationen finden Sie unter Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service im IAM-Benutzerhandbuch.
  - Service-linked Rolle — Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer AWS-Service verknüpft ist. Der Dienst kann die Rolle übernehmen, eine Aktion in Ihrem Namen auszuführen. Service-linked Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM Administrator kann die Berechtigungen für dienstbezogene Rollen anzeigen, aber nicht bearbeiten.
- Anwendungen, die auf einer Instanz ausgeführt werden Amazon EC2 — Sie können eine IAM Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer Amazon EC2 Instanz ausgeführt werden und AWS API-Anfragen stellen AWS CLI . Dies ist dem Speichern von Zugriffsschlüsseln innerhalb der Amazon EC2 Instanz vorzuziehen. Um einer Amazon 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 Amazon EC2 Instanz ausgeführt werden, temporäre Anmeldeinformationen abzurufen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter Verwenden einer IAM Rolle zum Erteilen von Berechtigungen für Anwendungen, die auf Amazon EC2 Instances ausgeführt werden.

Informationen zur Verwendung von IAM Rollen finden Sie im IAM-Benutzerhandbuch unter Wann IAM sollte eine Rolle (anstelle eines Benutzers) erstellt werden?.

Verwalten des Zugriffs mit Richtlinien

Sie steuern 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.

Jede IAM Entität (Benutzer oder Rolle) beginnt ohne Berechtigungen. Standardmäßig können Benutzer nichts tun, nicht einmal ihr eigenes Passwort ändern. Um einem Benutzer die Berechtigung für eine Aktion zu erteilen, muss ein Administrator einem Benutzer eine Berechtigungsrichtlinie zuweisen. Alternativ kann der Administrator den Benutzer zu einer Gruppe hinzufügen, die über die gewünschten Berechtigungen verfügt. Wenn ein Administrator einer Gruppe Berechtigungen erteilt, erhalten alle Benutzer in dieser Gruppe diese Berechtigungen.

IAM Richtlinien definieren Berechtigungen für eine Aktion, unabhängig von der Methode, mit der Sie den Vorgang ausführen. 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.

Identity-based Richtlinien

Identity-based Richtlinien sind Richtliniendokumente für JSON-Berechtigungen, die Sie an eine Identität, z. B. eine IAM-Benutzer Rolle oder Gruppe, anhängen können. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen einer identitätsbasierten Richtlinie finden Sie unter IAM Richtlinien erstellen im IAM-Benutzerhandbuch.

Identity-based Richtlinien können weiter in Inline-Richtlinien oder verwaltete Richtlinien unterteilt werden. Eingebundene 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 eingebundenen Richtlinie wählen, finden Sie unter Auswahl zwischen verwalteten und eingebundenen Richtlinien im IAM-Benutzerhandbuch.

Resource-based Richtlinien

Resource-based Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource wie einen Amazon S3 Bucket anhängen. Dienstadministratoren können diese Richtlinien verwenden, um zu definieren, welche Aktionen ein bestimmter Prinzipal (Kontomitglied, Benutzer oder Rolle) auf dieser Ressource ausführen kann und unter welchen Bedingungen. Resource-based Richtlinien sind Inline-Richtlinien. Es gibt keine verwalteten ressourcenbasierten Richtlinien.

Zugriffssteuerungslisten (ACLs)

Zugriffssteuerungslisten (ACL) sind eine Art von Richtlinie, die steuert, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) Berechtigungen für den Zugriff auf eine Ressource haben. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat. Amazon S3 AWS WAF, und Amazon VPC sind Beispiele für Dienste, die ACLs unterstützen. Weitere Informationen zu ACLs finden Sie unter Zugriffssteuerungsliste (ACL) – Übersicht im Entwicklerhandbuch zu Amazon Simple Storage Service.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger verbreitete Richtlinientypen. Mit diesen Richtlinientypen können Sie die maximalen Berechtigungen festlegen, die Ihnen durch die gängigeren Richtlinientypen gewährt werden.

- Berechtigungsgrenzen — Eine Berechtigungsgrenze ist eine erweiterte Funktion, mit der Sie die maximalen Berechtigungen festlegen, die eine identitätsbasierte Richtlinie einer IAM Entität (IAM-Benutzer oder Rolle) gewähren kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen stellen die Schnittmenge zwischen den identitätsbasierten Richtlinien der Entität und ihren Berechtigungsgrenzen dar. Resource-based Richtlinien, die den Benutzer oder die Rolle in dem `Principal` Feld angeben, sind nicht durch die Berechtigungsgrenze begrenzt. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen zu Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM Entitäten im IAM-Benutzerhandbuch.
- Service Control Policies (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 innerhalb einer Organisation alle Features aktivieren, können Sie Service-Kontrollrichtlinien (SCPs) auf alle oder einzelne Ihrer Konten anwenden. SCPs schränken Berechtigungen für Entitäten in Mitgliedskonten einschließlich des jeweiligen Root-Benutzer des AWS-Kontos ein. Weitere Informationen zu Organisationen und SCPs finden Sie unter Funktionsweise von SCPs 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 die Schnittmenge der identitätsbasierten Richtlinien des Benutzers oder der Rolle 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 für eine Anfrage mehrere Arten von Richtlinien gelten, sind die daraus resultierenden Berechtigungen schwieriger zu verstehen. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie unter Bewertungslogik für Richtlinien im IAM-Benutzerhandbuch.

So funktioniert AWS Security Agent mit IAM

Bevor Sie IAM den Zugriff auf AWS Security Agent verwalten, sollten Sie sich darüber informieren, welche IAM Funktionen mit AWS Security Agent verwendet werden können.

IAM-Feature	Unterstützung für AWS Security Agent
the section called “Identity-based Richtlinien für AWS Security Agent”	Ja
the section called “Resource-based Richtlinien innerhalb von AWS Security Agent”	Nein
the section called “Richtlinienaktionen für AWS Security Agent”	Ja
the section called “Richtlinienressourcen für AWS Security Agent”	Teilweise
the section called “Schlüssel für Richtlini enbedingungen für AWS Security Agent”	Ja
the section called “Zugriffskontrolllisten (ACLs) im AWS Security Agent”	Nein
the section called “Attribute-based Zugriffsk ontrolle (ABAC) mit AWS Security Agent”	Nein

IAM-Feature	Unterstützung für AWS Security Agent
the section called “Temporäre Anmeldeinformationen mit AWS Security Agent verwenden”	Ja
the section called “Zugriffssitzungen für AWS Security Agent weiterleiten”	Ja
the section called “Servicerollen für AWS Security Agent”	Nein
the section called “Service-linked Rollen für AWS Security Agent”	Ja

Einen umfassenden Überblick darüber, wie AWS Security Agent und andere AWS-Services Unternehmen AWS-Services damit arbeiten IAM, finden Sie IAM im IAM-Benutzerhandbuch.

Identity-based Richtlinien für AWS Security Agent

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identity-based Richtlinien sind Richtliniendokumente für JSON-Berechtigungen, die Sie an eine Identität anhängen können, z. B. an einen IAM-Benutzer, eine Benutzergruppe oder eine Rolle. 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 IAM identitätsbasierten Richtlinien können Sie zulässige oder verweigerte Aktionen und Ressourcen sowie die Bedingungen angeben, unter denen Aktionen zulässig oder verweigert werden. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, der er zugeordnet ist. Weitere Informationen zu allen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie im IAM-Benutzerhandbuch unter Referenz zu IAM JSON-Richtlinienelementen.

Identity-based Richtlinienbeispiele für AWS Security Agent

Beispiele für identitätsbasierte Richtlinien von AWS Security Agent finden Sie unter. the section called “Beispiele für identitätsbasierte Richtlinien für AWS Security Agent”

Resource-based Richtlinien innerhalb von AWS Security Agent

Unterstützt ressourcenbasierte Richtlinien: Nein

Resource-based Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anhängen. 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 AWS-Services gehören.

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 Principal und die Ressource in unterschiedlichen AWS-Konten befinden, muss ein IAM-Administrator des vertrauenswürdigen Kontos auch der Prinzipalentität (Benutzer oder Rolle) die Erlaubnis erteilen, auf die Ressource zuzugreifen. 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 Kontenübergreifender Ressourcenzugriff in IAM im IAM-Benutzerhandbuch.

Richtlinienaktionen für AWS Security Agent

Unterstützt Aktionen Ja

Administratoren können mithilfe von AWS-JSON-Richtlinien festlegen, wer zum Zugriff auf was berechtigt ist. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das `Action` Element einer IAM identitätsbasierten Richtlinie beschreibt die spezifischen Aktionen, die durch die Richtlinie zugelassen oder verweigert werden. Richtlinienaktionen haben normalerweise

denselben Namen wie der zugehörige AWS API-Vorgang. Die Aktion wird in einer Richtlinie verwendet, um Berechtigungen zur Durchführung der zugehörigen Aktion zu gewähren.

Richtlinienaktionen in AWS Security Agent verwenden das folgende Präfix vor der Aktion:`securityagent:`. Um beispielsweise jemandem die Erlaubnis zu erteilen, eine Umgebung mit dem AWS Security Agent `CreateEnvironment` API-Vorgang zu erstellen, nehmen Sie die `securityagent:CreateEnvironment` Aktion in seine Richtlinie auf. Richtlinienanweisungen müssen entweder ein – `Action`oder ein `NotAction`-Element enthalten. Der AWS Security Agent definiert seine eigenen Aktionen, die Aufgaben beschreiben, die Sie mit diesem Service ausführen können.

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie wie folgt durch Kommata:

```
"Action": [
      "securityagent:action1",
      "securityagent:action2"
```

Sie können auch Platzhalter verwenden, um mehrere Aktionen anzugeben. Beispielsweise können Sie alle Aktionen festlegen, die mit dem Wort `List` beginnen, einschließlich der folgenden Aktion:

```
"Action": "securityagent:List*"
```

Richtlinienressourcen für AWS Security Agent

Unterstützt Richtlinienressourcen: teilweise

Administratoren können mithilfe von AWS-JSON-Richtlinien festlegen, wer zum Zugriff auf was berechtigt ist. 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": "*"
```

Einige AWS-Security-Agent-API-Aktionen unterstützen mehrere Ressourcen. Beispielsweise können beim Aufrufen der `ListEnvironments` API-Aktion mehrere Umgebungen referenziert werden. Um mehrere Ressourcen in einer einzigen Anweisung anzugeben, trennen Sie die ARNs durch Kommata voneinander.

```
"Resource": [
      "EXAMPLE-RESOURCE-1",
      "EXAMPLE-RESOURCE-2"
```

Die AWS Security Agent-Umgebungsressource hat beispielsweise den folgenden ARN:

```
arn:${Partition}:securityagent:${Region}:${Account}:environment/${EnvironmentId}
```

Verwenden Sie zur Angabe der Umgebungen `my-environment-1` und `my-environment-2` in Ihrer Erklärung die folgenden Beispiel-ARNs:

```
"Resource": [
        "arn:aws:securityagent:us-east-1:123456789012:environment/my-environment-1",
        "arn:aws:securityagent:us-east-1:123456789012:environment/my-environment-2"
```

Um alle Umgebungen anzugeben, die zu einem bestimmten Konto gehören, verwenden Sie den Platzhalter (*):

```
"Resource": "arn:aws:securityagent:us-east-1:123456789012:environment/*"
```

Schlüssel für Richtlinienbedingungen für AWS Security Agent

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS-JSON-Richtlinien festlegen, wer zum Zugriff auf was berechtigt ist. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Mit dem `Condition` Element (oder `Condition` Block) können Sie Bedingungen angeben, unter denen eine Anweisung gültig 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 sein, bevor die Berechtigungen für die Anweisung erteilt werden.

Sie können bei der Angabe von Bedingungen auch Platzhaltervariablen verwenden. Sie können beispielsweise nur dann eine IAM-Benutzer Zugriffsberechtigung für eine Ressource erteilen, wenn sie mit ihrem IAM-Benutzer Namen gekennzeichnet ist. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter IAM Richtlinienelemente: Variablen und Tags.

AWS Security Agent definiert seinen eigenen Satz von Bedingungsschlüsseln und unterstützt auch die Verwendung einiger globaler Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontext-Schlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

Zugriffskontrolllisten (ACLs) im AWS Security Agent

Unterstützt ACLs: Nein

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Attribute-based Zugriffskontrolle (ABAC) mit AWS Security Agent

Unterstützt ABAC (Tags in Richtlinien): Nein

Temporäre Anmeldeinformationen mit AWS Security Agent verwenden

Unterstützt temporäre Anmeldeinformationen: Ja

Einige AWS-Services funktionieren nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, einschließlich der Frage, welche AWS-Services mit temporären Anmeldeinformationen funktionieren, finden Sie im IAM-Benutzerhandbuch unter AWS-Services, die mit IAM funktionieren.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen Methode als einem Benutzernamen und einem Passwort bei der AWS-Managementkonsole anmelden. Wenn Sie beispielsweise über den Single Sign-On (SSO) -Link Ihres Unternehmens auf AWS zugreifen, erstellt dieser Prozess automatisch temporäre Anmeldeinformationen. 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.

Sie können mithilfe der AWS-CLI oder der AWS-API manuell temporäre Anmeldeinformationen erstellen. Sie können diese temporären Anmeldeinformationen dann für den Zugriff auf AWS verwenden. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter Temporäre Sicherheitsanmeldeinformationen in IAM.

Zugriffssitzungen für AWS Security Agent weiterleiten

Unterstützt Forward Access Sessions (FAS): Ja

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle verwenden, um Aktionen in AWS auszuführen, 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 AWS-Service aufruft, in Kombination mit dem anfordernden AWS-Service, um Anfragen an nachgelagerte Services zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Service 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-Anforderungen finden Sie unter Zugriffssitzungen weiterleiten.

Servicerollen für AWS Security Agent

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.

Service-linked Rollen für AWS Security Agent

Unterstützt serviceverknüpfte Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einem AWS-Service verknüpft ist. Der Service kann die Rolle übernehmen, eine Aktion in Ihrem Namen auszuführen. Service-linked Rollen erscheinen in Ihrem AWS-Konto und gehören dem Service. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Beispiele für identitätsbasierte Richtlinien für AWS Security Agent

Standardmäßig sind Rollen nicht berechtigt, IAM-Benutzer AWS Security Agent-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben mit der AWS Management Console AWS CLI, oder AWS API ausführen. Ein IAM Administrator muss IAM Richtlinien erstellen, die Benutzern und Rollen die Erlaubnis gewähren, bestimmte API-Operationen mit den angegebenen Ressourcen auszuführen, die sie benötigen. Der Administrator muss diese Richtlinien dann den Gruppen IAM-Benutzer oder Gruppen zuordnen, für die diese Berechtigungen erforderlich sind.

Informationen zum Erstellen einer identitätsbasierten IAM-Richtlinie mithilfe dieser Beispieldokumente zu JSON-Richtlinien finden Sie unter Erstellen von Richtlinien mit dem JSON-Editor im IAM-Benutzerhandbuch.

Best Practices für Richtlinien

Identity-based Richtlinien legen fest, ob jemand AWS Security Agent-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Wenn Sie identitätsbasierte Richtlinien erstellen oder bearbeiten, befolgen Sie diese Richtlinien und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien und Umstellung auf Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um damit zu beginnen, Ihren Benutzern und Workloads 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 Managed Policies oder AWS Managed Policies for Job Functions im IAM-Benutzerhandbuch.
- Berechtigungen mit den geringsten Rechten anwenden — Wenn Sie Berechtigungen mit IAM Richtlinien festlegen, gewähren Sie nur die Berechtigungen, die für die Ausführung 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 IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen IAM im IAM-Benutzerhandbuch.
- Verwenden Sie Bedingungen in IAM Richtlinien, um den Zugriff weiter einzuschränken — Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen einzuschränken. 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 im Rahmen einer bestimmten Aktion verwendet werden AWS-Service, z. CloudFormation B. Weitere Informationen finden Sie unter IAM JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.

- Wird verwendet IAM Access Analyzer , um Ihre IAM Richtlinien zu validieren, um sichere und funktionale Berechtigungen zu gewährleisten — IAM Access Analyzer validiert neue und bestehende Richtlinien, sodass die Richtlinien der IAM Richtliniensprache (JSON) und den IAM Best Practices entsprechen. IAM Access Analyzer bietet mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen, um Sie bei der Erstellung sicherer und funktionaler Richtlinien zu unterstützen. Weitere Informationen finden Sie unter IAM Access Analyzer Richtlinienvalidierung im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das Root-Benutzer in Ihrem Konto erfordert IAM-Benutzer , 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 Konfiguration des MFA-protected API-Zugriffs im IAM-Benutzerhandbuch.

Verwenden der AWS Security Agent-Konsole

Um auf die AWS Security Agent-Konsole zugreifen zu können, muss ein IAM-Principal über einen Mindestsatz an Berechtigungen verfügen. Diese Berechtigungen müssen es dem Principal ermöglichen, Details zu den AWS Security Agent-Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die restriktiver als die mindestens erforderlichen Berechtigungen ist, funktioniert die Konsole für Prinzipals, denen diese Richtlinie zugewiesen ist, nicht wie vorgesehen.

Um sicherzustellen, dass Ihre IAM-Prinzipale weiterhin die AWS Security Agent-Konsole verwenden können, erstellen Sie eine Richtlinie mit Ihrem eigenen eindeutigen Namen. Hängen Sie die Richtlinie an die Prinzipale an. Weitere Informationen finden Sie unter Hinzufügen von Berechtigungen zu einem Benutzer imIAM-Benutzerhandbuch:

Einzelheiten zu den Richtlinien finden Sie unter. the section called “Von AWS verwaltete Richtlinie: SecurityAgentWebAppAPIPolicy”

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 Sie ausführen möchten.

Fehlerbehebung bei Identität und Zugriff auf den AWS Security Agent

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit AWS Security Agent und auftreten können IAM.

AccessDeniedException

Wenn Sie `AccessDeniedException` beim Aufrufen eines AWS-API-Vorgangs eine Meldung erhalten, verfügen die von Ihnen verwendeten IAM-Prinzipalanmeldedaten nicht über die erforderlichen Berechtigungen, um diesen Aufruf durchzuführen.

```
An error occurred (AccessDeniedException) when calling the CreateEnvironment operation:
User: arn:aws:iam::111122223333:user/user_name is not authorized to perform:
securityagent:CreateEnvironment on resource:
 arn:aws:securityagent:region:111122223333:environment/my-env
```

In der vorherigen Beispielnachricht ist der Benutzer nicht berechtigt, den AWS Security Agent `CreateEnvironment` API-Vorgang aufzurufen. Informationen zum Erteilen von AWS Security Agent-Administratorberechtigungen für einen IAM-Prinzipal finden Sie unterthe section called “Beispiele für identitätsbasierte Richtlinien für AWS Security Agent”.

Weitere allgemeine Informationen zu IAM finden Sie unter Steuern des Zugriffs auf AWS-Ressourcen mithilfe von Richtlinien im IAM-Benutzerhandbuch.

Ich möchte Personen außerhalb meines AWS-Konto um auf meine AWS Security Agent-Ressourcen zuzugreifen

Sie können eine Rolle erstellen, mit der Benutzer anderer Konten oder Personen außerhalb Ihrer Organisation auf Ihre Ressourcen zugreifen können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Im Fall von Diensten, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (Access Control Lists, ACLs) verwenden, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob AWS Security Agent diese Funktionen unterstützt, finden Sie unterthe section called “So funktioniert AWS Security Agent mit IAM”.
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie IAM-Benutzer im IAM-Benutzerhandbuch unter Zugriff auf ein anderes AWS-Konto , das Sie besitzen.

- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren AWS-Konten, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs auf Ressourcen, die AWS-Konten Eigentum Dritter sind.
- 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 im IAM-Benutzerhandbuch unter Unterschiede zwischen IAM Rollen und ressourcenbasierten Richtlinien.

Vorfallreaktion

AWS Security Agent ist ein proaktiver Sicherheitstest-Service, der darauf ausgelegt ist, Sicherheitslücken zu identifizieren und zu verhindern, bevor sie ausgenutzt werden können. Der Service konzentriert sich eher auf die präventive Sicherheitsvalidierung durch Entwurfsprüfungen, Codeanalysen und Penetrationstests als auf die reaktive Erkennung oder Reaktion auf Vorfälle.

Erkennung von Vorfällen

AWS Security Agent bietet keine Funktionen zur Erkennung von Vorfällen. Der Service fungiert als proaktives Sicherheitstest-Tool, das Anwendungen während der Entwicklung und vor der Bereitstellung auf Sicherheitslücken überprüft. Verwenden Sie für die Überwachung der Laufzeitsicherheit und die Erkennung von Vorfällen Dienste wie Amazon GuardDuty, AWS Security Hub oder Amazon CloudWatch.

Warnung bei Vorfällen

AWS Security Agent generiert keine Echtzeitwarnungen für Sicherheitsvorfälle. Der Service stellt nach Abschluss von Entwurfsprüfungen, Codeanalysen oder Penetrationstests Sicherheitsergebnisse über die AWS-Konsole bereit. Bei diesen Ergebnissen handelt es sich eher um potenzielle Sicherheitslücken, die während der Tests entdeckt wurden, als um aktive Sicherheitsvorfälle.

Behebung von Vorfällen

AWS Security Agent bietet keine automatische Behebung von Vorfällen. Der Service identifiziert Sicherheitslücken und bietet Anleitungen zur Behebung, darunter:

- Detaillierte Beschreibungen der identifizierten Sicherheitslücken

- Reproduzierbare Exploit-Pfade für validierte Ergebnisse
- Spezifische Codekorrekturen und Anleitungen zur Implementierung
- Folgenabschätzung für entdeckte Probleme

Entwicklungs- und Sicherheitsteams nutzen diese Anleitung, um Sicherheitslücken manuell zu beheben, bevor sie Produktionsumgebungen erreichen.

Unterstützung von Aktivitäten zur Reaktion auf Vorfälle

AWS Security Agent ist zwar nicht für die Reaktion auf Vorfälle konzipiert, Sicherheitsteams können den Service jedoch zur Unterstützung von Aktivitäten nach einem Vorfall nutzen:

Validierung von Sicherheitslücken

: Verwenden Sie nach einem Sicherheitsvorfall den AWS Security Agent, um zu testen, ob ähnliche Sicherheitslücken in anderen Anwendungen oder Umgebungen bestehen.

Bewertung des Sicherheitszustands

: Führen Sie Penetrationstests durch, um die im Rahmen der Behebung von Sicherheitsvorfällen implementierten Sicherheitsverbesserungen zu validieren.

Ursachenanalyse

: Verwenden Sie Funktionen zur Überprüfung der Code-Sicherheit, um festzustellen, wie ähnliche Sicherheitslücken in anderen Codebasen bestehen könnten.

Konformitätsvalidierung für AWS Security Agent

Um zu erfahren, ob ein AWS-Service in den Geltungsbereich bestimmter Compliance-Programme fällt, sehen Sie sich AWS-Services unter Umfang nach Compliance-Programmen an und wählen Sie das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter AWS-Compliance-Programme.

Die Auditberichte von Drittanbietern lassen sich mit AWS Artifact herunterladen. Weitere Informationen finden Sie unter Herunterladen von Berichten in AWS Artifact.

Ihre Compliance-Verantwortung bei der Nutzung von AWS-Services hängt von der Sensibilität Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen bereit, um Sie bei der Compliance zu 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 an Arbeitsmappen und Leitfäden könnte für Ihre Branche und Ihren Standort gelten.
- AWS-Leitfäden zur Einhaltung von Vorschriften für Kunden — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. Die Leitfäden fassen die bewährten Methoden zur Sicherung von AWS-Services zusammen und ordnen die Leitlinien den Sicherheitskontrollen in verschiedenen Frameworks (einschließlich National Institute of Standards and Technology (NIST), Payment Card Industry Security Standards Council (PCI) und International Organization for Standardization (ISO)) zu.
- Auswertung von Ressourcen mit Regeln im AWS Config-Entwicklerhandbuch – Der AWS Config-Service bewertet, wie gut Ihre Ressourcenkonfigurationen mit internen Praktiken, Branchenrichtlinien und Vorschriften übereinstimmen.
- AWS Security Hub — Dieser AWS-Service bietet einen umfassenden Überblick über Ihren Sicherheitsstatus innerhalb von AWS. Security Hub verwendet Sicherheitskontrollen, um Ihre AWS-Ressourcen zu bewerten und Ihre Einhaltung der Sicherheitsstandards und Best Practices der Sicherheitsbranche zu überprüfen. Die Liste der unterstützten Services und Kontrollen finden Sie in der Security-Hub-Steuerelementreferenz.
- Amazon GuardDuty — Dieser AWS-Service erkennt potenzielle Bedrohungen für Ihre AWS-Konten, Workloads, Container und Daten, indem er Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen wie PCI DSS zu erfüllen, indem wir die in bestimmten Compliance-Frameworks vorgeschriebenen Anforderungen zur Erkennung von Eindringlingen erfüllen.
- AWS Audit Manager – Dieser AWS-Service hilft Ihnen, Ihre AWS-Nutzung kontinuierlich zu überprüfen, um zu vereinfachen, wie Sie mit Risiken umgehen und Compliance sowie Branchenstandards einhalten.

Resilienz im AWS Security Agent

> **Note**
>
> AWS Security Agent ist in den folgenden Regionen verfügbar: USA Ost (Nord-Virginia) —`us-east-1`, USA West (Oregon) —`us-west-2`, Asien-Pazifik (Mumbai) —`ap-south-1`, Asien-Pazifik (Singapur) —`ap-southeast-1`, Asien-Pazifik (Sydney) —`ap-southeast-2`, Asien-Pazifik (Tokio) — `ap-northeast-1eu-central-1`, Europa (Frankfurt) —`eu-west-1`, Europa (Irland) — und Südamerika (São Paulo) —`sa-east-1`. Es verwendet regionsübergreifende Inferenz. Im asiatisch-pazifischen Raum (Mumbai), im asiatisch-pazifischen Raum (Singapur) und in Südamerika (São Paulo) verwendet AWS Security Agent globale regionsübergreifende Inferenz, die Inferenzanfragen an jede kommerzielle AWS-Region weiterleitet. In allen anderen Regionen werden geografische regionsübergreifende Inferenzen verwendet, wodurch die Datenverarbeitung innerhalb bestimmter geografischer Grenzen gehalten wird. AWS Security Agent ist ein Tool, das bei der Entwicklung Ihrer Anwendung verwendet wird und nicht als kritische oder kundenorientierte Infrastruktur eingesetzt werden sollte.

Im Zentrum der globalen AWS-Infrastruktur stehen die AWS-Regionen und Availability Zones. AWS-Regionen stellen mehrere physisch getrennte und isolierte Availability Zones bereit, die über hoch redundante Netzwerke mit niedriger Latenz und hohen Durchsätzen 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.

Weitere Informationen zu AWS-Regionen und Availability Zones finden Sie unter Weltweite AWS-Infrastruktur.

Infrastruktursicherheit in AWS Security Agent

Als verwalteter Service ist der AWS Security Agent durch die globale Netzwerksicherheit von AWS geschützt. Informationen zu AWS-Sicherheitsservices und zum Schutz der Infrastruktur durch AWS finden Sie unter AWS Cloud Security. Informationen zum Entwerfen Ihrer AWS-Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Sicherheit im Well-Architected AWS-Framework.

Netzwerkisolierung

AWS Security Agent ist ein vollständig verwalteter Service, auf den über die AWS-Konsole und die AWS Security Agent-Webanwendung zugegriffen werden kann. Der Zugriff auf den Service wird über AWS Identity and Access Management (IAM) oder AWS IAM Identity Center gesteuert, das in Ihren Identitätsanbieter integriert werden kann.

AWS Security Agent unterstützt Schnittstellen-VPC-Endpunkte, die von AWS betrieben werden PrivateLink, sodass Sie von Ihrer VPC aus privat auf die Service-APIs zugreifen können, ohne das öffentliche Internet zu nutzen. Weitere Informationen finden Sie unter the section called “Schnittstellen-VPC-Endpunkte”.

AWS Security Agent benötigt Internetzugang, um Penetrationstests an Zielanwendungen durchzuführen und den Betrieb auf der Kontrollebene durchzuführen. Der Service verwendet die AWS-managed Infrastruktur zum Testen und stellt keine EC2-Instances oder andere Rechenressourcen in Ihrem Konto bereit. Wenn Sie den VPC-Zugriff für Penetrationstests konfigurieren, erstellt der Security Agent eine ENI in Ihrem Subnetz, aber diese ENI hat keine öffentliche IP-Adresse. Weitere Informationen finden Sie unter the section called “Agent mit privaten VPC-Ressourcen verbinden”.

Multi-tenancy und Ressourcenisolierung

AWS Security Agent ist ein Service für mehrere Mandanten. Sicherheitsüberprüfungen, Ergebnisse und Kundendaten werden für einzelne AWS-Konten isoliert und im Ruhezustand verschlüsselt. AWS wendet Standardkontrollen zur Isolierung der Infrastruktur an, um sicherzustellen, dass die Sicherheitstests eines Kunden die Leistung oder Vertraulichkeit eines anderen Kunden nicht beeinträchtigen.

AWS Security Agent und VPC-Schnittstellen-Endpunkte (AWS PrivateLink)

Sie können AWS PrivateLink damit eine private Verbindung zwischen Ihrer VPC und dem AWS Security Agent herstellen. Sie können auf den Security Agent zugreifen, als wäre er in Ihrer VPC, ohne ein Internet-Gateway, ein NAT-Gerät, eine VPN-Verbindung oder eine Direct Connect-Verbindung verwenden zu müssen. Instances in Ihrer VPC benötigen keine öffentlichen IP-Adressen, um auf den Security Agent 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 Traffic dienen, der für den Security Agent bestimmt ist.

Weitere Informationen finden Sie AWS PrivateLink im Handbuch unter Access AWS Services through. AWS PrivateLink

Überlegungen zum Security Agent

Bevor Sie einen Schnittstellen-Endpunkt für den Security Agent einrichten, lesen Sie die Überlegungen im AWS PrivateLink Handbuch.

Der Security Agent unterstützt Aufrufe all seiner API-Aktionen von Ihrer VPC aus, einschließlich Operationen zur Verwaltung von Agentenbereichen, Penetrationstests und Sicherheitsergebnissen.

Sie können bei der Erstellung eines Endpunkts IPv4, IPv6 oder Dual-Stack wählen.

VPC-Endpunktrichtlinien werden für Security Agent unterstützt. Standardmäßig ist der vollständige Zugriff auf den Security Agent über den Schnittstellenendpunkt zulässig. Sie können den Zugriff kontrollieren, indem Sie eine Endpunkt-Richtlinie an den Schnittstellenendpunkt anhängen oder den Endpunkt-Netzwerkschnittstellen eine Sicherheitsgruppe zuordnen.

Alle API-Aufrufe an den Security Agent werden mit TLS 1.2 oder höher verschlüsselt, einschließlich Aufrufe, die über VPC-Endpunkte getätigt werden. Weitere Informationen zum Datenschutz in finden Sie unter the section called “Datenschutz”. API-Aufrufe des Security Agents werden protokolliert. AWS CloudTrail Weitere Informationen finden Sie unter the section called “Beispiel: Einträge in der AWS Security Agent-Protokolldatei”.

Erstellen Sie einen Schnittstellen-Endpunkt für den Security Agent

Sie können einen Schnittstellenendpunkt für Security Agent entweder mit der Amazon VPC-Konsole oder der AWS Befehlszeilenschnittstelle (AWS CLI) erstellen. Weitere Informationen finden Sie im AWS PrivateLink Handbuch unter Einen Schnittstellen-Endpunkt erstellen.

Erstellen Sie einen Schnittstellenendpunkt für den Security Agent mit dem folgenden Dienstnamen:

```
com.amazonaws.<region>.securityagent
```

Führen Sie den folgenden Befehl aus, um den Schnittstellenendpunkt mit der AWS CLI zu erstellen. Ersetzen Sie die Region, die VPC-ID, die Subnetz-IDs und die Sicherheitsgruppen-ID durch Ihre eigenen Werte.

```
aws ec2 create-vpc-endpoint \
    --vpc-id vpc-1a2b3c4d \
    --vpc-endpoint-type Interface \
    --service-name com.amazonaws.<region>.securityagent \
    --subnet-ids subnet-1a2b3c4d subnet-5e6f7a8b \
    --security-group-ids sg-1a2b3c4d \
    --private-dns-enabled
```

Important

Private DNS muss für den Schnittstellenendpunkt aktiviert sein. Der Security Agent verlangt, dass Sie den standardmäßigen regionalen DNS-Namen verwenden (z. B.`securityagent.us-east-1.api.aws`), um API-Anfragen über den Endpunkt zu stellen. Das direkte Aufrufen der endpunktspezifischen VPCE-URL wird nicht unterstützt.

Um privates DNS zu verwenden, müssen Sie `enableDnsSupport` sowohl die `enableDnsHostnames` Attribute als auch `true` für Ihre VPC auf festlegen. Weitere Infomationen finden Sie unter DNS-Attribute für Ihre VPC im Amazon VPC-Benutzerhandbuch.

Konfigurieren Sie Sicherheitsgruppen für den Schnittstellenendpunkt

Wenn Sie einen Schnittstellenendpunkt erstellen, können Sie den Endpunkt-Netzwerkschnittstellen Sicherheitsgruppen zuordnen, um den Datenverkehr zum Security Agent zu kontrollieren. Erstellen Sie eine Sicherheitsgruppe, die eingehenden HTTPS-Verkehr (Port 443) von den Ressourcen in Ihrer VPC zulässt, die mit dem Security Agent kommunizieren müssen.

Die Sicherheitsgruppe sollte die folgende Regel für eingehenden Datenverkehr enthalten:

- Typ: HTTPS
- Protokoll: TCP
- Portbereich: 443
- Quelle: Ihr VPC-CIDR-Block (zum Beispiel`10.0.0.0/16`) oder die Sicherheitsgruppen-IDs der Ressourcen, die Zugriff auf den Security Agent benötigen

Weitere Informationen finden Sie im Handbuch unter Sicherheitsgruppen. AWS PrivateLink

Erstellen einer Endpunktrichtlinie für Ihren Schnittstellen-Endpunkt

Eine Endpunktrichtlinie ist eine IAM-Ressource, die Sie an einen Schnittstellen-Endpunkt anfügen können. Die standardmäßige Endpunktrichtlinie ermöglicht vollen Zugriff auf den Security Agent über den Schnittstellenendpunkt. Um den Zugriff auf den Security Agent von Ihrer VPC aus zu kontrollieren, fügen Sie dem Schnittstellen-Endpunkt eine benutzerdefinierte Endpunkt-Richtlinie hinzu.

Eine Endpunktrichtlinie gibt die folgenden Informationen an:

- Die Principals, die Aktionen ausführen können (AWS Konten, IAM-Benutzer und IAM-Rollen).
- Aktionen, die ausgeführt werden können
- Die Ressourcen, auf denen die Aktionen ausgeführt werden können.

Weitere Informationen finden Sie im Handbuch unter Steuern des Zugriffs auf Dienste mithilfe von Endpunktrichtlinien. AWS PrivateLink

Beispiel: VPC-Endpunktrichtlinie für AWS Security Agent-Aktionen

Im Folgenden finden Sie ein Beispiel für eine Endpunktrichtlinie für AWS Security Agent. Wenn diese Richtlinie an einen Endpunkt angehängt ist, gewährt sie allen Prinzipalen auf allen Ressourcen Zugriff auf die aufgelisteten AWS Security Agent-Aktionen.

```
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": [
            "securityagent:CreatePentest",
            "securityagent:ListPentests",
            "securityagent:BatchGetPentests",
            "securityagent:StartPentestExecution",
            "securityagent:StopPentestExecution",
            "securityagent:ListFindings",
            "securityagent:BatchGetFindings"
         ],
         "Resource": "*"
      }
   ]
```

```
}
```

Beispiel: VPC-Endpunktrichtlinie, die jeglichen Zugriff von einem bestimmten Konto aus verweigert AWS

Die folgende VPC-Endpunktrichtlinie verweigert dem AWS Konto `123456789012` jeglichen Zugriff auf Ressourcen, die den Endpunkt verwenden. Die Richtlinie erlaubt alle Aktionen von anderen Konten.

```
{
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "securityagent:*",
         "Resource": "*"
      },
      {
         "Effect": "Deny",
         "Principal": {
            "AWS": "123456789012"
         },
         "Action": "securityagent:*",
         "Resource": "*"
      }
   ]
}
```

Fehlerbehebung

Beim Aufrufen der Security Agent API wird die Verbindung unterbrochen

- Stellen Sie sicher, dass der VPC-Endpunktstatus wie folgt lautet`available`:

  ```
  aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1a2b3c4d \
      --query "VpcEndpoints[].State"
  ```

- Vergewissern Sie sich, dass die dem Endpunkt zugeordnete Sicherheitsgruppe eingehenden TCP-Verkehr auf Port 443 von Ihrer VPC CIDR oder Quellsicherheitsgruppe zulässt.
- Stellen Sie sicher, dass Ihre VPC-Routing-Tabellen den Security Agent-Verkehr nicht an ein Internet-Gateway oder NAT-Gateway statt an den Endpunkt weiterleiten.

Die DNS-Auflösung schlägt fehl für **`securityagent.<region>.api.aws`**

- Stellen Sie sicher, dass privates DNS auf dem Endpunkt aktiviert ist:

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1a2b3c4d \
    --query "VpcEndpoints[].PrivateDnsEnabled"
```

- Vergewissern Sie sich, dass `enableDnsHostnames` beide `enableDnsSupport` und `true` auf Ihrer VPC auf eingestellt sind:

```
aws ec2 describe-vpc-attribute --vpc-id vpc-1a2b3c4d --attribute enableDnsSupport
aws ec2 describe-vpc-attribute --vpc-id vpc-1a2b3c4d --attribute enableDnsHostnames
```

`Access denied`Fehler beim Aufrufen der Security Agent-APIs

- Überprüfen Sie die VPC-Endpunktrichtlinie, um sicherzustellen, dass sie die Aktionen, die Sie ausführen möchten, zulässt.
- Stellen Sie sicher, dass Ihre IAM-Identitätsrichtlinie die erforderlichen `securityagent:` Berechtigungen gewährt.
- Wenn Sie eine benutzerdefinierte Endpunktrichtlinie verwenden, stellen Sie sicher, dass sowohl die Endpunktrichtlinie als auch die IAM-Richtlinie die Anfrage zulassen.

Konfiguration und Schwachstellenanalyse in AWS Security Agent

Konfiguration und IT-Steuerung unterliegen der übergreifenden Verantwortlichkeit von AWS und Ihnen, unserem Kunden. Weitere Informationen finden Sie unter AWS Modell der übergreifenden Verantwortlichkeit.

Cross-service verwirrter Stellvertreter, Prävention

Das Problem des verwirrten Stellvertreters ist ein Sicherheitsproblem, bei dem eine Entität, die keine Berechtigung zur Durchführung einer Aktion hat, eine privilegiertere Entität zur Durchführung der Aktion zwingen kann. In AWS kann ein serviceübergreifender Identitätswechsel zum Problem des verwirrten Stellvertreters führen. Cross-service Ein Identitätswechsel kann auftreten, wenn ein Service (der anrufende Service) einen anderen Service (den aufgerufenen Service) aufruft. Der Aufruf-Service kann so manipuliert werden, dass er seine Berechtigungen verwendet, um auf die Ressourcen eines anderen Kunden zu reagieren, auf die er sonst nicht zugreifen dürfte.

Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Service Principals schützen können, denen Zugriff auf Ressourcen in Ihrem Konto gewährt wurde. Weitere Informationen finden Sie unter Cross-service Confused Deputy Prevention im AWS Identity and Access Management-Benutzerhandbuch.

Bewährte Sicherheitsmethoden für AWS Security Agent

AWS Security Agent bietet eine Reihe von Sicherheitsfunktionen, die Sie bei der Entwicklung und Implementierung Ihrer eigenen Sicherheitsrichtlinien berücksichtigen sollten. Die folgenden bewährten Methoden sind allgemeine Richtlinien und keine vollständige Sicherheitslösung. Da diese bewährten Methoden für Ihre Umgebung möglicherweise nicht angemessen oder ausreichend sind, sollten Sie sie als hilfreiche Überlegungen und nicht als bindend ansehen.

Verwenden Sie Umgebungen außerhalb der Produktion für Penetrationstests

AWS Security Agent verwendet eine umfassende Suite von Penetrationstest-Tools aus der Kali-Linux-Distribution. Diese Tools dienen zur Identifizierung von Sicherheitslücken und können Aktionen ausführen, die den Anwendungsstatus, die Daten oder die Systemkonfigurationen ändern.

Bewährtes Verfahren: Führen Sie Penetrationstests in Umgebungen durch, die nicht zur Produktion verwendet werden, die Ihrem Produktionssetup entsprechen. Diese Testumgebungen sollten:

- Sie dürfen keine Live-Kundendaten oder sensible Produktionsinformationen enthalten
- Seien Sie von den Produktionssystemen isoliert
- Verfügen Sie über ähnliche Konfigurationen und Sicherheitskontrollen wie in der Produktion
- Verwenden Sie keine Anmeldeinformationen für den Zugriff auf Produktionssysteme

Tests in Produktionsumgebungen können zu folgenden Ergebnissen führen:

- Änderung oder Löschung von Daten
- Betriebsunterbrechungen oder Leistungseinbußen
- Unbeabsichtigte Zustandsänderungen
- Auslösung von Sicherheitswarnungen oder Verfahren zur Reaktion auf Vorfälle

Bestätigen Sie die AI-generated Sicherheitsfeststellungen

AWS Security Agent führt Sicherheitsanalysen mithilfe von KI-Agenten durch. Aufgrund des nicht deterministischen Charakters von KI-Systemen können Penetrationstestläufe bei verschiedenen Ausführungen zu unterschiedlichen Ergebnissen führen.

Bewährtes Verfahren: Überprüfen Sie die Sicherheitsfeststellungen, bevor Sie Abhilfemaßnahmen ergreifen:

- Überprüfen Sie die vom AWS Security Agent generierten Überprüfungsskripten für jedes Ergebnis.
- Führen Sie Verifizierungsskripten in Ihrer Testumgebung aus, um die Sicherheitslücke zu bestätigen
- Erwägen Sie die Durchführung mehrerer Penetrationstests, um eine umfassende Abdeckung sicherzustellen
- Beurteilen Sie den Schweregrad und die Ausnutzbarkeit der Ergebnisse anhand professioneller Sicherheitsbeurteilungen

Möglicherweise handelt es sich nicht bei allen identifizierten Problemen um Sicherheitslücken, die in Ihrem spezifischen Einsatzkontext ausgenutzt werden können.

Überprüfen und testen Sie den generierten Problembehebungscode

AWS Security Agent kann Codekorrekturen und Sicherheitsverbesserungen für identifizierte Sicherheitslücken generieren. Diese AI-generated Fixes müssen vor der Bereitstellung überprüft werden.

Bewährtes Verfahren: Überprüfen Sie alle generierten Codeänderungen:

- Untersuchen Sie die vorgeschlagenen Korrekturen auf Vollständigkeit und Richtigkeit
- Testen Sie die Korrekturen in Umgebungen außerhalb der Produktionsumgebung gründlich
- Stellen Sie sicher, dass Fixes nicht zu neuen Sicherheitslücken führen oder Funktionen beeinträchtigen
- Verwenden Sie den AWS Security Agent, um nach der Installation von Fixes erneut zu testen, oder führen Sie die bereitgestellten Überprüfungsskripte aus
- Folgen Sie den Codeüberprüfungs- und Genehmigungsprozessen Ihrer Organisation

Zugriff auf das Code-Repository

AWS Security Agent kann Sicherheitsleitfäden zu Codeänderungen durch Pull-Request-Kommentare und Code-Review-Integration bereitstellen. Um vertrauliche Sicherheitsinformationen zu schützen, unterliegt diese Funktion bestimmten Einschränkungen.

Einschränkung: Die Richtlinien zur Codesicherheit sind nur auf private Repositorys beschränkt. Dadurch wird sichergestellt, dass:

- Sicherheitsergebnisse bleiben für Ihr Unternehmen vertraulich
- Potenzielle Sicherheitslücken werden vor der Behebung nicht öffentlich bekannt gegeben
- In den generierten Empfehlungen zur Behebung von Sicherheitslücken werden keine Details zur Ausnutzung

AWS Security Agent bietet keine Hinweise zur Codesicherheit für öffentliche Repositorys. Es wird sich nicht zu öffentlichen Repositorien oder Open-Source-Projekten äußern, bei denen Sicherheitsergebnisse öffentlich sichtbar wären.

Mit den Penetrationstests von AWS Security Agent können private und öffentliche Repositorys, die Sie für den Pentest konfiguriert haben, überprüft und behoben werden. Wenn das Repository öffentlich ist, wird der Behebungscode als herunterladbare Diff-Datei statt als Pull-Anfrage bereitgestellt.

Zugängliche URLs

Zugängliche URLs geben zusätzliche Endpunkte an, auf die die Penetrationstestumgebung während des Tests zugreifen kann. Diese sind erforderlich, wenn Ihre Anwendung von externen Diensten wie externen Authentifizierungsanbietern oder CDNs abhängig ist. Alle zum Testen erforderlichen Netzwerkabhängigkeiten müssen entweder als Ziel-URLs oder als zugängliche URLs angegeben werden. Das Netzwerk blockiert den Zugriff auf nicht spezifizierte Endpunkte.

Auswirkungen auf die Sicherheit: Der AWS Security Agent ist nicht angewiesen, Sicherheitstests für URLs durchzuführen, auf die zugegriffen werden kann. Durch die Angabe zugänglicher URLs signalisieren Sie, dass Sie diesen Abhängigkeiten vertrauen. Penetrationstestdaten, einschließlich Anmeldeinformationen, können während des Tests an diese zugänglichen URL-Endpunkte übertragen werden.

Regionsübergreifende Inferenz

AWS Security Agent wählt automatisch die optimale Region für die Verarbeitung Ihrer Inferenzanfragen aus. Dies maximiert die verfügbaren Rechenressourcen und die Modellverfügbarkeit und sorgt für ein optimales Kundenerlebnis. Ihre Daten bleiben nur in der Region gespeichert, aus der die Anfrage stammt. Eingabeaufforderungen und Ausgabeergebnisse können jedoch außerhalb dieser Region verarbeitet werden. Wir übertragen alle Daten verschlüsselt über das AWS-Netzwerk.

AWS Security Agent verwendet je nach Region zwei Arten von regionsübergreifenden Inferenzen:

- Regionsübergreifende geografische Inferenz — Die Datenverarbeitung bleibt für die meisten Features innerhalb bestimmter geografischer Grenzen (z. B. USA, EU, Australien oder Japan). Im Rahmen von Code Remediation werden Anfragen aus Australien und Japan in der Europäischen Union bearbeitet. Wird in den USA Ost (Nord-Virginia) —`us-east-1`, USA West (Oregon) —`us-west-2`, Asien-Pazifik (Sydney) —`ap-southeast-2`, Asien-Pazifik (Tokio) —`ap-northeast-1`, Europa (Frankfurt) — `eu-central-1` und Europa (Irland) — verwendet`eu-west-1`.
- Globale regionsübergreifende Inferenz — Leitet Inferenzanfragen an jede kommerzielle AWS-Region weiter, optimiert die verfügbaren Ressourcen und ermöglicht einen höheren Modelldurchsatz. Wird im asiatisch-pazifischen Raum (Mumbai) —`ap-south-1`, im asiatisch-pazifischen Raum (Singapur) — `ap-southeast-1` und in Südamerika (São Paulo) — verwendet`sa-east-1`.

Für Regionen, die globale regionsübergreifende Inferenz verwenden, können Eingabeaufforderungen und Ausgabeergebnisse in jeder kommerziellen AWS-Region verarbeitet werden. Alle Daten, die während regionsübergreifender Operationen übertragen werden, verbleiben im AWS-Netzwerk und werden nicht über das öffentliche Internet übertragen. Wir verschlüsseln Daten bei der Übertragung zwischen AWS-Regionen.

In der folgenden Tabelle wird anhand der Region, aus der die Anfrage stammt, und der verwendeten Funktion beschrieben, wo Ihre Inferenzanfragen verarbeitet werden.

Herkunft der Anfrage	Alle Funktionen außer Code Remediation	Korrektur von Code
Vereinigte Staaten — USA Ost (Nord-Virginia) —`us-east-1`	Vereinigte Staaten	Vereinigte Staaten

Herkunft der Anfrage	Alle Funktionen außer Code Remediation	Korrektur von Code
, USA West (Oregon) — `us-west-2`		
Europäische Union — Europa (Irland) —`eu-west-1` , Europa (Frankfurt) — `eu-central-1`	Europäische Union	Europäische Union
Australien — Asien-Pazifik (Sydney) — `ap-southeast-2`	Australien	Europäische Union
Japan — Asien-Pazifik (Tokio) — `ap-northeast-1`	Japan	Europäische Union
Südamerika — Südamerika (São Paulo) — `sa-east-1`	Jede kommerzielle AWS-Region	Jede kommerzielle AWS-Region
Indien — Asien-Pazifik (Mumbai) — `ap-south-1`	Jede kommerzielle AWS-Region	Jede kommerzielle AWS-Region
Südostasien — Asien-Pazifik (Singapur) — `ap-southeast-1`	Jede kommerzielle AWS-Region	Jede kommerzielle AWS-Region

Cross-Region Inferenz ist immer aktiviert und kann nicht deaktiviert werden. Cross-Region Die Inferenz wird nicht durch Kundenrichtlinien in Service Control Policies (SCPs) oder AWS Control Tower beeinflusst, die Kundeninhalte auf bestimmte Regionen beschränken. Weitere Informationen darüber, wie der AWS Security Agent Ihre Daten bei der regionsübergreifenden Verarbeitung schützt, finden Sie unter Cross-Region Datenverarbeitung.

IAM-Aktionen und Ressourcenmigration

AWS Security Agent ist ein Frontier Agent, der Ihre Anwendungen während des gesamten Entwicklungszyklus in all Ihren Umgebungen proaktiv schützt. Wenn Sie vor dem 9. Februar 2026 bei

AWS Security Agent angemeldet sind, werden Sie von den bevorstehenden Änderungen am 9. März 2026 an Ihren bestehenden Agent-Instance-Ressourcen und den IAM-Aktionen von AWS Security Agent betroffen sein. In Vorbereitung auf die Veröffentlichung des öffentlichen API/SDK Supports wird die Agent Instance-Ressource in Agent Space umbenannt, und bestimmte IAM-Aktionen werden umbenannt. Diese Änderungen wirken sich auf alle IAM-Rollen für Anwendungs- oder Agenteninstanzen aus, die Sie vor dem 9. März 2026 erstellt haben. Um Authentifizierungsprobleme nach dem 9. März 2026 zu vermeiden, müssen Sie die Schritte unter Vorbereitung der Migration befolgen.

> **Note**
>
> Wenn Sie nach dem 9. Februar 2026 neue Agent-Instanzen erstellen, wird die neue Agent-Instanz als Agent Space erstellt und es sind keine Migrationsschritte erforderlich.

Geplante Änderungen

AWS Security Agent benennt die Agent-Instance-Ressource in Agent Space: `arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*` umbenannt in `umarn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*`. Darüber hinaus werden die folgenden IAM-Aktionen umbenannt:

- `securityagent:ListAgentInstances`umbenannt in `securityagent:ListAgentSpaces`
- `securityagent:ListControls`umbenannt in `securityagent:ListSecurityRequirements`
- `securityagent:BatchGetAgentInstances`umbenannt in `securityagent:BatchGetAgentSpaces`
- `securityagent:BatchGetSecurityTestContentMetadata`umbenannt in `securityagent:BatchGetPentestJobContentMetadata`
- `securityagent:BatchGetTasks`umbenannt in `securityagent:BatchGetPentestJobTasks`
- `securityagent:CreateDocumentReview`umbenannt in `securityagent:CreateDesignReview`
- `securityagent:GetDocumentReview`umbenannt in `securityagent:GetDesignReview`
- `securityagent:GetDocumentReviewArtifact`umbenannt in `securityagent:GetDesignReviewArtifact`

- `securityagent:ListDocumentReviews`umbenannt in `securityagent:ListDesignReviews`
- `securityagent:ListDocumentReviewComments`umbenannt in `securityagent:ListDesignReviewComments`
- `securityagent:ListTasks`umbenannt in `securityagent:ListPentestJobTasks`
- `securityagent:StartPentestExecution`umbenannt in `securityagent:StartPentestJob`
- `securityagent:StopPentestExecution`umbenannt in `securityagent:StopPentestJob`
- `securityagent:DeleteDocumentReview`umbenannt in `securityagent:DeleteDesignReview`

Vorbereitung der Migration

Um Probleme nach dem 9. März 2026 zu vermeiden und gleichzeitig den AWS Security Agent vor dem 9. März 2026 weiter zu verwenden, müssen Sie roles/policies bis zum 9. März 2026 sowohl den neuen als auch den alten Ressourcen und IAM-Aktionen in Ihrem IAM vertrauen. Die folgenden Anweisungen bieten eine Anleitung für die Migration zu den neuen Ressourcen- und Aktionsformaten:

1. Melden Sie sich bei Ihrem AWS-Konto an und navigieren Sie zur AWS Security Agent-Konsole
2. Wählen Sie im linken Bereich Einstellungen und wählen Sie die Rolle unter Servicerolle
3. Wählen Sie in der IAM-Konsole für die zugehörige Rolle die Optionen Berechtigungen hinzufügen und Richtlinien anhängen
4. Wählen Sie AWSSecurityAgentWebAppPolicyund wählen Sie Berechtigungen hinzufügen
   a. Wichtiger Hinweis: Stellen Sie sicher, dass Sie AWSSecurityAgentWebAppPolicyes als neue Richtlinie ausgewählt haben und nicht SecurityAgentWebAppAPIPolicy
5. Vergewissern Sie sich, dass für Ihre IAM-Rolle jetzt beide Richtlinien gelten AWSSecurityAgentWebAppPolicyund dass sie SecurityAgentWebAppAPIPolicyunter Berechtigungen aufgeführt sind
6. Wählen Sie in derselben IAM-Rollenkonsole Vertrauensbeziehungen und dann Vertrauensrichtlinie bearbeiten aus
7. Aktualisieren Sie Ihre Vertrauensrichtlinie auf das folgende Format und ersetzen Sie {{accountId}} durch Ihre AWS-Konto-ID

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:application/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:SetContext",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ForAllValues:ArnEquals": {
          "sts:RequestContextProviders": "arn:aws:iam::aws:contextProvider/
IdentityCenter"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:application/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
```

```
      }
    }
  ]
}
```

8. Navigieren Sie zurück zur AWS Security Agent-Konsole. Wählen Sie im linken Bereich Agent Spaces
9. Führen Sie für jeden Agent Space, bei dem Penetrationstests aktiviert sind, die folgenden Schritte durch
   a. Navigieren Sie zum Agent Space und wählen Sie Penetrationstest
   b. Wählen Sie unter Dienstzugriff die Rolle unter Name der Servicerolle aus
   c. Wählen Sie in der IAM-Konsole für die zugehörige Rolle die Option Vertrauensbeziehungen und dann Vertrauensrichtlinie bearbeiten aus
   d. Aktualisieren Sie Ihre Vertrauensrichtlinie auf das folgende Format und ersetzen Sie {{accountId}} durch Ihre AWS-Konto-ID

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "securityagent.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "{{accountId}}"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-space/*",
            "arn:aws:securityagent:us-east-1:{{accountId}}:agent-instance/*"
          ]
        }
      }
    }
  ]
}
```

Protokollierung AWS Security Agent API-Aufrufe mit AWS CloudTrail

AWS Security Agent ist in einen Dienst integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Dienst im AWS Security Agent ausgeführt wurden. CloudTrail erfasst alle API-Aufrufe für den AWS Security Agent als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der AWS Security Agent-Konsole und Code-Aufrufe an die AWS Security Agent 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 AWS Security Agent. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von CloudTrail gesammelten Informationen können Sie die Anfrage an den AWS Security Agent, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Informationen ermitteln.

Weitere Informationen CloudTrail dazu finden Sie im AWS CloudTrail Benutzerhandbuch.

AWS Informationen zum Security Agent in CloudTrail

CloudTrail ist für Ihr AWS Konto aktiviert, wenn Sie das Konto erstellen. Wenn im AWS Security Agent eine Aktivität auftritt, wird diese Aktivität zusammen mit anderen AWS Dienstereignissen in der CloudTrail Ereignishistorie in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem AWS Konto anzeigen, suchen und herunterladen. Weitere Informationen finden Sie unter Ereignisse mit dem CloudTrail Ereignisverlauf anzeigen.

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS Konto, einschließlich der Ereignisse für AWS Security Agent, 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 erstellen, gilt der Trail standardmäßig für alle AWS Regionen. Der Trail protokolliert Ereignisse aus allen Regionen der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie hier:

- Übersicht zum Erstellen eines Trails
- CloudTrail unterstützte Dienste und Integrationen
- Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen und Empfangen von CloudTrail Protokolldateien von mehreren Konten

Alle Aktionen des AWS Security Agents werden von protokolliert CloudTrail. Aufrufe von, und `ListFindings` Aktionen generieren beispielsweise Einträge in den CloudTrail Protokolldateien. `CreatePentest` `BatchGetPentestJobs`

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 Root- oder AWS Identity and Access Management (IAM) - Benutzeranmeldedaten gestellt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde.

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

Beispiel: AWS Einträge in der Security Agent-Protokolldatei

Verstehen AWS Einträge in der Security Agent-Protokolldatei

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 und enthält 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.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die `CreatePentest` Aktion demonstriert:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/Alice",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "Alice"
    },
```

```
    "eventTime": "2025-01-15T10:30:00Z",
    "eventSource": "securityagent.amazonaws.com",
    "eventName": "CreatePentest",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.12",
    "userAgent": "aws-cli/2.13.0",
    "requestParameters": {
        "pentestName": "WebApp-Security-Test",
        "targetUrl": "https://example.com",
        "testScope": "OWASP-Top-10"
    },
    "responseElements": {
        "pentestId": "pt-1234567890abcdef0",
        "pentestArn": "arn:aws:securityagent:us-east-1:123456789012:pentest/
pt-1234567890abcdef0"
    },
    "requestID": "a1b2c3d4-e5f6-7890-abcd-ef1234567890",
    "eventID": "12345678-1234-1234-1234-123456789012",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```


Service Quotas

AWS Security Agent verfügt über Kontingente, die die Anzahl der Ressourcen, die Sie erstellen können, und die Geschwindigkeit, mit der Sie Operationen ausführen können, einschränken. Als anpassbar markierte Kontingente können erhöht werden, indem Sie eine Anfrage über den AWS-Support einreichen. Weitere Informationen finden Sie unter Erstellung von Supportfällen und Fallmanagement.

Kontingente für Operationen

Betriebskontingente beschränken die monatliche Nutzung von Funktionen für Sicherheitstests und -überprüfungen, um die Verwaltung der Servicekapazität zu unterstützen.

Ressource	Scope	Kontingent	Einstellbar
Bewertungen von Entwürfen	Pro Monat pro Konto pro Region	200	Ja
PR-Code-Bewertungen	Pro Monat pro Konto pro Region	1.000	Ja

Kontingente für Konfigurationen

Konfigurationskontingente begrenzen die Anzahl der Ressourcen und Einstellungen, die Sie in Ihrer AWS Security Agent-Umgebung konfigurieren können.

Ressource	Scope	Kontingent	Einstellbar
Agent-Spaces	Pro Konto pro Region	100	Ja
Integrationen	Pro Konto pro Region	20	Nein
Integrierte Ressourcen pro Integration	Pro Integration	50	Nein
Kundenspezifische Sicherheitsanforderungen	Pro Konto pro Region	20	Nein
Pentest-Projekte	Pro Konto pro Region	1.000	Ja
Gleichzeitige Pentest-Läufe	Pro Konto pro Region	5	Ja
Projekte zur Überprüfung des Codes	Pro Konto pro Region	1.000	Ja
Gleichzeitige Codeüberprüfung wird ausgeführt	Pro Konto pro Region	5	Ja

Dokumentverlauf

In der folgenden Tabelle werden einige der wichtigsten Updates und neuen Funktionen für das AWS Security Agents User Guide beschrieben.

Änderung	Beschreibung	Datum
Private Verbindungen (Vorschau)	Dokumentation für private Verbindungen hinzugefügt. Diese neue Funktion ermöglicht es dem AWS Security Agent, mithilfe von Amazon VPC Lattice eine Verbindung zu Quellcode verwaltungssystemen herzustellen, die in privaten Netzwerken laufen, ohne Ihre Systeme dem öffentlichen Internet auszusetzen.	16. Juni 2026
Bedrohungsmodellierung (Vorschau)	Dokumentation zur Bedrohungsmodellierung hinzugefügt. Diese neue Funktion erstellt ein Bedrohungsmodell Ihrer Anwendung auf der Grundlage von Entwurfsdokumenten (Scope Docs), Quellcode (Sources) oder beidem. Scope-Dokumente sind Feature-Design-Dokumente, die den Schwerpunkt der Analyse definieren, während der Quellcode den Kontext zu Ihrem bestehenden System bereitstellt. Wenn Sie keine Dokumente zum Umfang bereitstellen, generiert der Agent allein anhand des Quellcodes ein Bedrohungsmodell. Bei jedem Durchlauf erhalten Sie eine Systemübersicht und	8. Juni 2026

	eine Reihe von Bedrohungen, die nach STRIDE-Kategorien klassifiziert sind, mit Schweregradeinstufungen und Empfehlungen.	
Vollständige Überprüfung des Repository-Codes (Vorschau)	Dokumentation für die vollständige Überprüfung des Repository-Codes hinzugefügt. Diese neue Funktion führt eine kontextsensitive Sicherheitsanalyse Ihrer gesamten Codebasis durch und generiert für die Ergebnisse eine Codekorrektur.	12. Mai 2026
Die Verfügbarkeit der Region wurde aktualisiert, um Abhilfemaßnahmen zu finden	Die regionale Verfügbarkeit für Penetrationstests, bei denen Abhilfemaßnahmen in der AWS Security Agent Web App gefunden wurden, wurde aktualisiert.	16. April 2026
Die Verfügbarkeit in der Region wurde aktualisiert, um die Suche nach Problemlösungen zu ermöglichen	Die regionale Verfügbarkeit für die Aktivierung von Penetrationstests zur Behebung von Abhilfemaßnahmen in der AWS-Managementkonsole wurde aktualisiert.	16. April 2026

Unterstützung für vom Kunden verwaltete Schlüssel	Es wurde eine Dokumentation für die Unterstützung von Customer Managed Key (CMK) hinzugefügt. Sie können jetzt bei der Erstellung von Agent Spaces und Integrationen einen vom Kunden verwalteten KMS-Schlüssel angeben, um Ihre Daten mit Schlüsseln zu verschlüsseln, die Sie kontrollieren.	31. März 2026
AWS verwaltete Richtlinienaktualisierungen	AWSSecurityAgentWebAppPolicy Verwaltete Richtlinien für die neuen Typen TargetDomain und DesignReviewFeedback Ressourcentypen wurden hinzugefügt.	31. März 2026
AWS verwaltete Richtlinienaktualisierungen	AWSSecurityAgentWebAppPolicy Es wurde eine verwaltete Richtlinie für den neuen AgentSpace Ressourcentyp und die Änderung des Namens der IAM-Aktion hinzugefügt.	9. Februar 2026
AWS verwaltete Richtlinienaktualisierungen	Aktualisiert SecurityAgentWebAppAPIPolicy , sodass Kunden Designrezensionen löschen können.	28. Januar 2026

AWS verwaltete Richtlinienaktualisierungen	Das Update wurde aktualisiert SecurityAgentWebApPAPIPolicy , sodass Kunden mit der automatisierten Codekorrektur aufgrund von Sicherheitslücken beginnen können.	20. Januar 2026
AWS verwaltete Richtlinienaktualisierungen	Es wurde aktualisiert SecurityAgentWebApPAPIPolicy , sodass Kunden nun Bilder in der Konsole ansehen können.	5. Dezember 2025
Erste Version von AWS Security Agents	Erste Dokumentation für den Servicestart	2. Dezember 2025

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.