



Guida per l'utente

Amazon Inspector



Amazon Inspector: Guida per l'utente

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

Table of Contents

Che cos'è Amazon Inspector?	1
Funzionalità	1
Accesso ad Amazon Inspector	3
Nozioni di base	5
Prima di attivare Amazon Inspector	5
Tutorial introduttivo: attivazione di Amazon Inspector	6
Scansioni automatiche	12
Panoramica dei tipi di scansione di Amazon Inspector	12
Attivazione di un tipo di scansione	14
Attivazione delle scansioni	15
Scansione delle istanze Amazon EC2	16
Agent-based scansione	16
Scansione senza agenti	22
Gestione della modalità di scansione	23
Esclusione delle istanze dalle scansioni di Amazon Inspector	24
Sistemi operativi supportati	25
Ispezione approfondita per istanze Linux	25
Scansione dell'istanza EC2 Windows	30
Scansione delle immagini dei contenitori Amazon ECR	33
Comportamenti di scansione per la scansione Amazon ECR	34
Mappatura delle immagini dei container ai container in esecuzione	35
Sistemi operativi e tipi di supporti supportati	37
Configurazione della durata della nuova scansione di Amazon ECR	38
Funzione di scansione Lambda	40
Comportamenti di scansione per la scansione della funzione Lambda	41
Runtime e funzioni supportati	42
Scansione standard Amazon Inspector Lambda	43
Scansione del codice Amazon Inspector Lambda	44
Disattivazione di un tipo di scansione	46
Disattivazione delle scansioni	47
Scansioni CIS	49
Requisiti delle istanze Amazon EC2 per le scansioni CIS di Amazon Inspector	50
Requisiti degli endpoint di Amazon Virtual Private Cloud per l'esecuzione di scansioni CIS su istanze private di Amazon EC2	51

Esecuzione di scansioni CIS	51
Considerazioni sulla gestione delle scansioni CIS di Amazon Inspector con AWS Organizations	52
Bucket Amazon S3 di proprietà di Amazon Inspector utilizzati per le scansioni CIS di Amazon Inspector	54
Creazione di una configurazione di scansione CIS	56
Visualizzazione dei risultati della scansione CIS	57
Modifica di una configurazione di scansione CIS	58
Scaricamento dei risultati di una scansione CIS	58
Codice di sicurezza di Amazon Inspector	60
Prerequisiti	60
Attivazione della sicurezza del codice	60
Creazione di una chiave di accesso gestita dal cliente AWS KMS	60
Creare un'integrazione	63
Creazione di un'integrazione per GitHub	64
Creazione di un'integrazione per GitLab Self Managed	66
Visualizzazione delle integrazioni	68
Visualizzazione degli archivi di codice	69
Eliminazione di un'integrazione	70
Creazione di una configurazione di scansione	70
Visualizzazione delle configurazioni di scansione	73
Modifica di una configurazione di scansione	74
Eliminazione di una configurazione di scansione	75
Esecuzione di una scansione su richiesta	75
Lingue supportate	76
Disattivazione di Code Security	77
Comprensione degli esiti	78
Tipi di esiti	79
Vulnerabilità del pacchetto	79
Vulnerabilità del codice	80
Raggiungibilità della rete	81
Visualizzazione dei risultati	82
Visualizzazione dei dettagli del risultato	83
Visualizzazione del punteggio di Amazon Inspector	86
Punteggio Amazon Inspector	87
Intelligenza sulla vulnerabilità	89

Comprendere dei livelli di gravità dei risultati	90
Gravità della vulnerabilità dei pacchetti software	90
Gravità della vulnerabilità del codice	91
Severità della raggiungibilità della rete	90
Gestione degli esiti	94
Filtro dei risultati	94
Creazione di filtri nella console Amazon Inspector	94
Eliminazione dei risultati	95
Creazione di una regola di soppressione	96
Visualizzazione dei risultati soppressi	97
Modifica di una regola di soppressione	97
Eliminazione di una regola di soppressione	97
Esportazione dei report sui risultati	98
Passaggio 1: verifica le tue autorizzazioni	99
Passaggio 2: configura un bucket S3	101
Fase 3: Configurare un AWS KMS key	104
Fase 4: Configurare ed esportare un rapporto sui risultati	107
Risoluzione degli errori	110
Automatizzazione delle risposte ai risultati con EventBridge	111
Schema degli eventi	112
Creazione di una EventBridge regola per notificarti i risultati di Amazon Inspector	114
EventBridge per ambienti con più account Amazon Inspector	118
Dashboard	119
Visualizzazione del pannello di controllo	119
Comprendere i componenti del dashboard	120
Ricerca nel database delle vulnerabilità	124
Ricerca nel database delle vulnerabilità	124
Comprendere i dettagli del CVE	125
Dettagli CVE	125
Intelligence sulle vulnerabilità	125
Riferimenti	125
Esportazione SBOMs	126
Formati Amazon Inspector	126
Filtri per SBOMs	131
Configura ed esporta SBOMs	132
EventBridge schema	135

Schema EventBridge di base Amazon per Amazon Inspector	135
Esempio di schema di eventi di ricerca di Amazon Inspector	136
Esempio di schema di eventi completo per la scansione iniziale di Amazon Inspector	148
Esempio di schema degli eventi di copertura di Amazon Inspector	151
Esempio di schema di attivazione automatica di Amazon Inspector	152
Scanner VM Amazon Inspector	153
Panoramica di	153
Vantaggi	153
Attivazione dello scanner VM Inspector	154
Requisiti	154
Abilitazione della scansione EC2 avanzata	154
Requisiti degli endpoint Amazon VPC per la scansione EC2 avanzata su istanze Amazon EC2 private	155
Domande frequenti	155
Third-party attribuzione del software	156
Installazione e configurazione manuali	156
Installazione e disinstallazione manuale di Inspector VM Scanner	157
Percorsi di installazione	161
Disinstallazione di Inspector VM Scanner	161
Esecuzione dello scanner VM Inspector	161
Visualizzazione dell'output	161
Visualizzazione dei log di	162
Visualizzazione dei log con systemd	162
Configurazione avanzata	163
Configurazione delle uscite locali	163
Configurazione dell'utilizzo delle risorse	164
Configurazione delle destinazioni di scansione	165
Gestione dell'esecuzione periodica	165
Plugin SSM	167
Il plug-in Amazon Inspector SSM per Linux	168
Il plug-in Amazon Inspector SSM per Windows	168
Generatore SBOM Amazon Inspector	171
Tipi di pacchetti supportati	171
Controlli di configurazione dell'immagine del contenitore supportati	172
Installazione di Sbmngen	172
Uso di Sbmngen	173

Genera un SBOM per un'immagine del contenitore e restituisci il risultato	174
Genera un SBOM da directory e archivi	175
Genera un SBOM da Go o Rust compilati file binari	175
Genera un SBOM dai volumi montati	176
Invia un SBOM ad Amazon Inspector per l'identificazione delle vulnerabilità	177
Utilizza scanner aggiuntivi per migliorare le capacità di rilevamento	179
Ottimizza le scansioni dei contenitori regolando la dimensione massima del file da scansionare	180
Disattiva l'indicatore di avanzamento	180
Autenticazione in registri privati con Sbomgen	181
Autenticazione tramite credenziali memorizzate nella cache (scelta consigliata)	181
Effettua l'autenticazione utilizzando il metodo interattivo	181
Effettua l'autenticazione utilizzando il metodo non interattivo	182
Esempi di risultati da Sbomgen	182
Versioni precedenti	185
Raccolta di sistemi operativi	198
Artefatti del sistema operativo supportati	199
Raccolta di pacchetti del sistema operativo basata su APK	200
Raccolta di pacchetti del sistema operativo basata su DPKG	201
Raccolta di pacchetti del sistema operativo basata su RPM	202
Raccolta di versioni del sistema operativo Windows	204
Raccolta di pacchetti di immagini Chainguard	204
Raccolta di pacchetti di immagini Distroless	205
Raccolta di pacchetti MiniMOS	207
Raccolta di dipendenze	207
Vai alla scansione delle dipendenze	208
Scansione delle dipendenze in Java	211
JavaScript scansione delle dipendenze	215
Scansione delle dipendenze.NET	222
Scansione delle dipendenze PHP	227
Scansione delle dipendenze in Python	230
Scansione delle dipendenze con Ruby	237
Scansione delle dipendenze da Rust	240
Scansione rapida delle dipendenze	243
Artefatti non supportati	246
Raccolta di ecosistemi	248

Ecosistemi supportati	248
7-Zip collezione ecosistemica	252
Amazon Q Developer raccolta di ecosistemi	253
Apache raccolta di ecosistemi	254
Atlassian raccolta di ecosistemi	260
Codice Claude collezione ecosistemica	262
Conda raccolta di ecosistemi	263
Curl raccolta ecosistemica	265
Cursore raccolta ecosistemica	267
Binari Docker raccolta di ecosistemi	268
Elasticsearch raccolta di ecosistemi	269
AI/ML Modelli generici collezione di ecosistemi	271
GitHub Copilota raccolta ecosistemica	272
Google raccolta di ecosistemi	273
HuggingFace raccolta ecosistemica	275
Java raccolta di ecosistemi	275
Jenkins raccolta di ecosistemi	277
CLI Kiro collezione di ecosistemi	279
MariaDB and MySQL raccolta di ecosistemi	280
Applicazioni Microsoft raccolta di ecosistemi	282
Microsoft SQL Server raccolta di ecosistemi	287
MongoDB raccolta di ecosistemi	288
Nginx raccolta di ecosistemi	289
Node.JS raccolta runtime	290
Ollama collezione ecosistemica	292
Collezionista di modelli Ollama collezione ecosistemica	293
Raccolta di ecosistemi OpenSSH	294
Collezione di ecosistemi OpenSSL	295
Collezione Oracle Database Server	296
PHP raccolta di ecosistemi	297
Redis raccolta di ecosistemi	298
Windsurf collezione ecosistemica	299
WordPress raccolta di ecosistemi	300
Zed collezione ecosistemica	303
Scansioni dei certificati SSL/TLS	304
Utilizzo delle scansioni dei Scomgen certificati	304

Raccolta di licenze	307
Raccogli informazioni sulla licenza	307
Pacchetti supportati	308
Package URLs	315
Struttura PURL	315
Riferimenti alle versioni	317
Raccomandazioni	317
Java	318
JavaScript	318
Python	318
Utilizzo CycloneDX dei namespace	319
amazon:inspector:sbom_scannertassonomia dei namespace	319
amazon:inspector:sbom_generatortassonomia dello spazio dei nomi	321
Guida per sviluppatori di plugin	327
Panoramica di	327
Avvio rapido: creazione di nuovi plugin	328
Configurazione IDE	330
Struttura delle cartelle dei plugin	331
Plugin Discovery	333
Plugin di raccolta	337
Esecuzione dei plugin	341
Elenco degli scanner disponibili	342
Selezione dello scanner	343
Regole di collisione dei plugin	346
Debug	347
Documentazione di riferimento delle API	350
Gestione errori	350
Restrizioni Sandbox	351
Condivisione del codice tra plugin	352
Guida al test dei plugin	353
Quick Start	353
Layout delle directory	354
L'API testing.*	355
Esecuzione di test.	357
Testare gli helper dei plugin	358
Comportamento e invarianti	358

Limitazioni	359
Responsabilità degli sviluppatori	360
Riferimento all'API del plugin	360
Panoramica di	360
Plugin Lifecycle Globals	362
File I/O	366
FileHandle Metodi	372
Utilità binarie	373
Output del pacchetto	376
Hash dei componenti	378
Proprietà CyclonedX	380
Costanti di proprietà	382
Gruppi di scanner	384
Costanti del tipo di componente	385
Costanti dell'algoritmo Hash	386
Costanti della piattaforma	387
Informazioni sull'Artifact	388
Informazioni di sistema	389
Espressioni regolari	390
Analisi strutturata	392
Registro di sistema di Windows	393
Registrazione dei log	394
Funzioni di debug	395
API di test	395
Gestione errori	398
CI/CD integrazione	399
Integrazione con i plugin	399
Soluzioni supportate CI/CD	400
Integrazione personalizzata	400
Configura un account per l'integrazione CI/CD	401
Iscriviti a un Account AWS	402
Configura un ruolo IAM per CI/CD l'integrazione	402
Controlli dei file Dockerfile di Amazon Inspector	403
Utilizzo dei controlli Dockerfile Sbomgen	403
Controlli Dockerfile supportati	406
Creazione di un'integrazione personalizzata CI/CD	411

Passaggio 1. Configurazione Account AWS	412
Passaggio 2. Installazione Sborgen binary	412
Fase 3. Utilizzo Sborgen	412
Passaggio 4. Chiamata dell'API Amazon Inspector Scan	412
(Facoltativo) Fase 5. Genera e scansiona SBOM con un solo comando	413
Formati di output delle API	413
Operazioni GitHub	425
Plugin Jenkins	426
Passaggio 1. Configura un Account AWS	426
Passaggio 2. Installa il plugin Amazon Inspector Jenkins	426
(Facoltativo) Passaggio 3. Aggiungi le credenziali docker a Jenkins	427
(Facoltativo) Fase 4. Add (Aggiungi) AWS credenziali	427
Fase 5. Aggiungi il supporto CSS in un Jenkins script	428
Fase 6. Aggiungi Amazon Inspector Scan alla tua build	428
Fase 7. Visualizza il report sulla vulnerabilità di Amazon Inspector	433
Risoluzione dei problemi	434
Operazioni CodeCatalyst	436
Operazioni CodePipeline	436
TeamCity Plugin	436
GitLab componenti	437
Valutazione della copertura	438
Valutazione della copertura a livello di account	439
Valutazione della copertura delle istanze Amazon EC2	439
Valori di stato delle istanze Amazon EC2	440
Valutazione della copertura dei repository Amazon ECR	442
Valori dello stato di scansione del repository Amazon ECR	443
Valutazione della copertura delle immagini dei container Amazon ECR	444
Valori dello stato di scansione delle immagini dei contenitori Amazon ECR	445
Valutazione della copertura delle funzioni AWS Lambda	446
Le funzioni Lambda scansionano i valori dello stato	447
Gestione di più account	449
Informazioni sull'account amministratore delegato e sull'account membro	449
Modello di governance delle politiche organizzative	450
Azioni degli amministratori delegati	450
Azioni relative all'account dei membri	452
Designazione di un account amministratore	453

Considerazioni	453
Autorizzazioni necessarie per designare un amministratore delegato	454
Designazione di un amministratore delegato	455
Attivazione delle scansioni Amazon Inspector per gli account dei membri	456
Dissociazione degli account dei membri	460
Rimozione dell'amministratore delegato	461
Applicazione di tag alle risorse	464
Nozioni fondamentali sull'etichettatura	464
Aggiunta di tag	465
Aggiungere tag alle risorse di Amazon Inspector	465
Rimozione dei tag	466
Rimuovere i tag dalle risorse di Amazon Inspector	467
Utilizzo	468
Utilizzo della console di utilizzo	468
Scopri come Amazon Inspector calcola i costi di utilizzo	470
Informazioni sulla versione di prova gratuita di Amazon Inspector	470
Sicurezza	472
Protezione dei dati	473
Crittografia dei dati a riposo	474
Crittografia dei dati in transito	479
Identity and Access Management	479
Destinatari	480
Autenticazione con identità	480
Gestione dell'accesso tramite policy	481
Come funziona Amazon Inspector con IAM	483
Identity-based esempi di politiche	489
AWS politiche gestite	493
Uso di ruoli collegati ai servizi	510
Risoluzione dei problemi	519
Monitoraggio di Amazon Inspector	521
CloudTrail registri	522
Convalida della conformità	525
Resilienza	526
Sicurezza dell'infrastruttura	526
Risposta agli incidenti	526
Ricerca sulla sicurezza	527

Metodologia di ricerca	527
Riepilogo del rilevamento	528
Segnalazioni recenti di pacchetti dannosi	529
Risorse correlate	529
AWS PrivateLink	530
Considerazioni	530
Creazione di un endpoint di interfaccia	530
Integrazioni	532
Utilizzo di Amazon Inspector con AWS Organizations	532
Integrazione di Amazon Inspector con Amazon ECR	532
Integrazione di Amazon Inspector con Security Hub CSPM	533
Integrazione con Amazon ECR	533
Attivazione dell'integrazione	533
Utilizzo dell'integrazione con un ambiente multi-account	533
Integrazioni CSPM di Security Hub	534
Visualizzazione dei risultati di Amazon Inspector in AWS Security Hub CSPM	535
Attivazione e configurazione dell'integrazione di Amazon Inspector con Security Hub CSPM	538
Attivazione di Amazon Inspector da Security Hub (CSPM) utilizzando i criteri dell'organizzazione	538
Disattivazione del flusso di risultati da un'integrazione	539
Visualizzazione dei controlli di sicurezza per Amazon Inspector in Security Hub CSPM	539
Sistemi operativi e linguaggi di programmazione supportati	540
Sistemi operativi supportati	541
Sistemi operativi supportati: Amazon EC2 scanning	541
Sistemi operativi supportati: scansione Amazon ECR con Amazon Inspector	545
Sistemi operativi supportati: scansione CIS	548
Sistemi operativi supportati: Amazon Inspector Scan API	549
Sistemi operativi fuori produzione	551
Linguaggi di programmazione compatibili	556
Linguaggi di programmazione supportati: scansione senza agenti di Amazon EC2	556
Linguaggi di programmazione supportati: Amazon EC2 deep inspection	556
Linguaggi di programmazione supportati: Amazon ECR scanning	557
Runtime supportati	558
Runtime supportati: scansione standard di Amazon Inspector Lambda	558
Runtime supportati: scansione del codice Amazon Inspector Lambda	559

Disattivazione di Amazon Inspector	561
Disattivazione di Amazon Inspector gestita dalle politiche dell'organizzazione	562
Disattiva Amazon Inspector	563
Quote	564
Regioni ed endpoint	566
Endpoint di servizio per Amazon Inspector	566
Endpoint per l'API Amazon Inspector Scan	566
Region-specific disponibilità delle funzionalità	575
Cronologia dei documenti	580
Aggiornamenti dei prodotti Amazon Inspector	580
AWS Glossario	615
.....	dcxvi

Che cos'è Amazon Inspector?

Amazon Inspector è un servizio di gestione delle vulnerabilità che rileva automaticamente i carichi di lavoro e li analizza continuamente per individuare vulnerabilità del software ed esposizione involontaria alla rete. [Amazon Inspector rileva e analizza le istanze Amazon EC2, le immagini dei container in Amazon ECR e le funzioni Lambda.](#) Quando Amazon Inspector rileva una vulnerabilità del software o un'esposizione involontaria della rete, crea [un risultato](#), ovvero un rapporto dettagliato sul problema. Puoi [gestire i risultati](#) nella console o nell'API di Amazon Inspector.

Note

Quando invia una richiesta di supporto, Amazon Inspector potrebbe accedere ed elaborare i risultati pertinenti nel luogo in Regione AWS cui sono archiviati (ma all'interno della stessa area geografica) per risolvere il problema.

Argomenti

- [Caratteristiche di Amazon Inspector](#)
- [Accesso ad Amazon Inspector](#)

Caratteristiche di Amazon Inspector

Gestione centralizzata di più account Amazon Inspector

Se AWS l'ambiente dispone di più account, è possibile gestire centralmente l'ambiente tramite un singolo account utilizzando AWS Organizations. Utilizzando questo approccio, puoi designare un account come account amministratore delegato per Amazon Inspector.

Amazon Inspector può essere attivato per l'intera organizzazione con un solo clic. Inoltre, puoi automatizzare l'attivazione del servizio per i futuri membri ogni volta che entrano a far parte della tua organizzazione. L'account amministratore delegato di Amazon Inspector può gestire i risultati, i dati e determinate impostazioni per i membri dell'organizzazione. Ciò include la visualizzazione dei dettagli aggregati dei risultati per tutti gli account dei membri, l'attivazione o la disattivazione delle scansioni per gli account dei membri e la revisione delle risorse scansionate all'interno dell'organizzazione.

AWS

Scansiona continuamente il tuo ambiente per individuare vulnerabilità ed esposizione della rete

Con Amazon Inspector, non è necessario pianificare o configurare manualmente le scansioni di valutazione. Amazon Inspector rileva e avvia automaticamente [la scansione](#) delle risorse idonee. Amazon Inspector continua a valutare l'ambiente durante l'intero ciclo di vita delle risorse effettuando una nuova scansione automatica delle risorse in risposta a modifiche che potrebbero introdurre una nuova vulnerabilità, ad esempio: installazione di un nuovo pacchetto in un'istanza EC2, installazione di una patch e quando viene pubblicata una nuova vulnerabilità ed esposizione comune (CVE) che ha un impatto sulla risorsa. A differenza dei tradizionali software di scansione di sicurezza, Amazon Inspector ha un impatto minimo sulle prestazioni della tua flotta.

Quando vengono identificate vulnerabilità o percorsi di rete aperti, Amazon Inspector produce [un](#) risultato che puoi esaminare. La scoperta include dettagli completi sulla vulnerabilità, sulla risorsa interessata e raccomandazioni per la correzione. Se correggi in modo appropriato un risultato, Amazon Inspector rileva automaticamente il problema e lo chiude.

Valuta accuratamente le vulnerabilità con il punteggio di rischio di Amazon Inspector

Poiché Amazon Inspector raccoglie informazioni sull'ambiente tramite scansioni, fornisce punteggi di gravità specificamente adattati al tuo ambiente. Amazon Inspector esamina i parametri di sicurezza che compongono il punteggio di base del [National Vulnerability Database \(NVD\) per una vulnerabilità](#) e li regola in base all'ambiente di elaborazione. Ad esempio, il servizio può ridurre il punteggio Amazon Inspector di un risultato per un'istanza Amazon EC2 se la vulnerabilità è sfruttabile sulla rete ma dall'istanza non è disponibile alcun percorso di rete aperto verso Internet. Questo punteggio è in formato CVSS ed è una modifica del punteggio di base del [Common Vulnerability Scoring System \(CVSS\)](#) fornito da NVD.

Identifica i risultati ad alto impatto con la dashboard di Amazon Inspector

La [dashboard di Amazon Inspector](#) offre una visione di alto livello dei risultati provenienti da tutto l'ambiente. Dalla dashboard, puoi accedere ai dettagli granulari di un risultato. La dashboard contiene informazioni semplificate sulla copertura delle scansioni nell'ambiente in uso, sui risultati più critici e sulle risorse con il maggior numero di risultati. Il pannello di correzione basata sul rischio nella dashboard di Amazon Inspector presenta i risultati che riguardano il maggior numero di istanze e immagini. Questo pannello semplifica l'identificazione dei risultati con il maggiore impatto sull'ambiente, l'analisi dei dettagli dei risultati e l'esame delle soluzioni suggerite.

Gestisci i risultati utilizzando visualizzazioni personalizzabili

Oltre alla dashboard, la console Amazon Inspector offre una visualizzazione dei risultati. Questa pagina elenca tutti i risultati relativi al tuo ambiente e fornisce i dettagli dei singoli risultati. È possibile

visualizzare i risultati raggruppati per categoria o tipo di vulnerabilità. In ogni visualizzazione, puoi personalizzare ulteriormente i risultati utilizzando i filtri. Puoi anche utilizzare i filtri per creare regole di soppressione che nascondono i risultati indesiderati dalle tue visualizzazioni.

È possibile utilizzare filtri e regole di soppressione per generare report sui risultati che mostrano tutti i risultati o una selezione personalizzata di risultati. I report possono essere generati in formato CSV o JSON.

Monitora ed elabora i risultati con altri servizi e sistemi

Per supportare l'integrazione con altri servizi e sistemi, Amazon Inspector [pubblica i risultati su Amazon EventBridge come eventi di ricerca](#). EventBridge è un servizio di bus eventi senza server in grado di indirizzare i dati dei risultati verso destinazioni come AWS Lambda funzioni e argomenti di Amazon Simple Notification Service (Amazon SNS). Con EventBridge, puoi monitorare ed elaborare i risultati quasi in tempo reale come parte dei flussi di lavoro di sicurezza e conformità esistenti.

Se l'hai attivato [AWS Security Hub CSPM](#), Amazon Inspector [pubblicherà anche i risultati su Security Hub CSPM](#). Security Hub CSPM è un servizio che fornisce una visione completa del tuo livello di sicurezza in tutto AWS l'ambiente e ti aiuta a controllare il tuo ambiente rispetto agli standard e alle migliori pratiche del settore della sicurezza. Con Security Hub CSPM, puoi monitorare ed elaborare più facilmente i tuoi risultati come parte di un'analisi più ampia del livello di sicurezza della tua organizzazione. AWS

Accesso ad Amazon Inspector

Amazon Inspector è disponibile nella maggior parte dei casi. Regioni AWS Per un elenco delle regioni in cui Amazon Inspector è attualmente disponibile, consulta gli [endpoint e le quote di Amazon Inspector](#) nell'Amazon Web Services General Reference. Per ulteriori informazioni Regioni AWS, consulta [Managing Regioni AWS](#) in Amazon Web Services General Reference. In ogni regione, puoi lavorare con Amazon Inspector nei seguenti modi.

AWS Console di gestione

Console di gestione AWS È un'interfaccia basata su browser che è possibile utilizzare per creare e gestire AWS risorse. Come parte di tale console, la console Amazon Inspector fornisce l'accesso al tuo account e alle tue risorse Amazon Inspector. Puoi eseguire attività di Amazon Inspector dalla console Amazon Inspector.

AWS strumenti da riga di comando

Con gli strumenti da riga di AWS comando, puoi emettere comandi dalla riga di comando del tuo sistema per eseguire attività di Amazon Inspector. L'utilizzo della riga di comando può essere più rapido e conveniente rispetto all'utilizzo della console. Gli strumenti a riga di comando sono inoltre utili per creare script che eseguono le attività di .

AWS fornisce due set di strumenti da riga di comando: the AWS Command Line Interface (AWS CLI) e the AWS Strumenti per PowerShell. Per informazioni sull'installazione e l'utilizzo di AWS CLI, consultate la [Guida per l'utente dell'interfaccia a riga di AWS comando](#). Per informazioni sull'installazione e l'utilizzo degli strumenti per PowerShell, consultate la [Guida per AWS Strumenti per PowerShell l'utente](#).

AWS SDKs

AWS fornisce SDKs che consistono in librerie e codice di esempio per vari linguaggi e piattaforme di programmazione, tra cui Java, Go, Python, C++ e .NET. SDKs Forniscono un accesso comodo e programmatico ad Amazon Inspector e ad altri. Servizi AWS Gestiscono anche attività come la firma crittografica delle richieste, la gestione degli errori e il tentativo automatico delle richieste. Per informazioni sull'installazione e l'utilizzo di AWS SDKs, consulta [Tools to Build on AWS](#).

API REST di Amazon Inspector

L'API REST di Amazon Inspector ti offre un accesso completo e programmatico al tuo account e alle tue risorse Amazon Inspector. Con questa API, puoi inviare richieste HTTPS direttamente ad Amazon Inspector. Tuttavia, a differenza degli strumenti da riga di AWS comando e SDKs, l'uso di questa API richiede che l'applicazione gestisca dettagli di basso livello, come la generazione di un hash per firmare una richiesta.

Guida introduttiva ad Amazon Inspector

Questa sezione fornisce informazioni da prendere in considerazione prima di attivare Amazon Inspector e un tutorial introduttivo che descrive come attivare Amazon Inspector e visualizzare i risultati [nella](#) console Amazon Inspector e con l'API Amazon Inspector.

Argomenti

- [Prima di attivare Amazon Inspector](#)
- [Tutorial introduttivo: attivazione di Amazon Inspector](#)

Prima di attivare Amazon Inspector

Prima di attivare Amazon Inspector, considera quanto segue:

Amazon Inspector è un servizio regionale

I tuoi dati vengono archiviati nel Regione AWS punto in cui attivi Amazon Inspector. Ripeti i passaggi indicati nella prima parte del [tutorial introduttivo](#) per tutti i Regioni AWS casi in cui intendi utilizzare Amazon Inspector.

Amazon Inspector crea i ruoli collegati ai servizi `AWSServiceRoleForAmazonInspector` e `AWSServiceRoleForAmazonInspector2Agentless`

Un [ruolo collegato a un servizio è un ruolo](#) in AWS Identity and Access Management (IAM) collegato a un servizio. AWS [AWSServiceRoleForAmazonInspector2](#) e [AWSServiceRoleForAmazonInspector2Agentless](#) consentono ad Amazon Inspector di accedere ai Servizi AWS requisiti necessari per eseguire le valutazioni di sicurezza.

Le identità IAM con autorizzazioni di amministratore possono abilitare Amazon Inspector

[Proteggi le tue credenziali creando utenti con IAM o AWS IAM Identity Center](#) Questo ti aiuta ad assicurarti che gli utenti dispongano solo delle autorizzazioni necessarie per gestire Amazon Inspector. Per ulteriori informazioni, consulta la [policy AWS gestita](#): `AmazonInspectorFullAccess`

La scansione ibrida viene abilitata automaticamente

La scansione ibrida include la scansione [basata su agenti e la scansione senza agenti](#). Per impostazione predefinita, Amazon Inspector utilizza questi metodi di scansione su tutte le istanze

Amazon EC2 idonee. Per ulteriori informazioni, consulta [Scansione EC2 delle istanze Amazon con Amazon Inspector](#).

La scansione Amazon ECR e la scansione della funzione Lambda non richiedono l'agente SSM

La scansione basata su agenti utilizza [l'agente SSM per raccogliere l'inventario del](#) software. La scansione senza agenti utilizza le istantanee di Amazon EBS per raccogliere l'inventario del software.

Note

Per impostazione predefinita, l'agente SSM è già installato nelle EC2 istanze Amazon basate su Amazon Machine Images. Tuttavia, in alcuni casi potrebbe essere necessario attivare l'agente SSM manualmente. Per ulteriori informazioni, consulta [Lavorare con l'agente SSM nella Guida](#) per l'AWS Systems Manager utente.

I costi mensili si basano sui carichi di lavoro scansionati

Per ulteriori informazioni, consulta [Prezzi di Amazon Inspector](#).

Abilitazione di più account con AWS Organizations

Per le organizzazioni che lo utilizzano [AWS Organizations](#), Amazon Inspector supporta sia la gestione degli amministratori delegati che l'abilitazione basata su policy organizzative. Le politiche organizzative forniscono una governance centralizzata con l'attivazione automatica di nuovi account. Per istruzioni dettagliate su entrambi gli approcci, vedere. [Tutorial introduttivo: attivazione di Amazon Inspector](#)

Tutorial introduttivo: attivazione di Amazon Inspector

Questo argomento descrive come attivare Amazon Inspector per un ambiente di account autonomo (account membro) e un ambiente multi-account (account amministratore delegato). Quando attivi Amazon Inspector, inizia automaticamente a rilevare i carichi di lavoro e a scansionarli per individuare vulnerabilità del software ed esposizione involontaria della rete.

Standalone account environment

La procedura seguente descrive come attivare Amazon Inspector nella console per un account membro. Per attivare Amazon Inspector in modo programmatico, `inspector2-enablement-with-cli`

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Seleziona Inizia.
3. Scegli Attivate Amazon Inspector.

Quando attivi Amazon Inspector per un account indipendente, [tutti i tipi di scansione](#) vengono attivati per impostazione predefinita. Per informazioni sugli account membro, consulta [Comprendere l'account amministratore delegato e gli account membro in Amazon Inspector](#).

Multi-account (with AWS Organizations policy)

AWS Organizations le policy forniscono una governance centralizzata per abilitare Amazon Inspector in tutta l'organizzazione. Quando utilizzi una politica aziendale, l'abilitazione di Amazon Inspector viene gestita automaticamente per tutti gli account coperti dalla policy e gli account dei membri non possono modificare la scansione gestita dalla policy utilizzando l'API Amazon Inspector.

Prerequisiti

- Il tuo account deve far parte di un'organizzazione. AWS Organizations
- È necessario disporre delle autorizzazioni per creare e gestire le politiche dell'organizzazione in AWS Organizations.
- L'accesso affidabile per Amazon Inspector deve essere abilitato in AWS Organizations Per istruzioni, consulta [Enabling trusted access for Amazon Inspector nella Guida](#) per l'AWS Organizations utente.
- I ruoli collegati al servizio Amazon Inspector devono esistere nell'account di gestione. Per crearli, abilita Amazon Inspector nell'account di gestione o esegui i seguenti comandi dall'account di gestione:
 - `aws iam create-service-linked-role --aws-service-name inspector2.amazonaws.com`
 - `aws iam create-service-linked-role --aws-service-name agentless.inspector2.amazonaws.com`
- È necessario designare un amministratore delegato di Amazon Inspector.

Note

Senza i ruoli di account di gestione e amministratore delegato collegati al servizio di Amazon Inspector, le politiche dell'organizzazione applicheranno l'abilitazione di Amazon Inspector, ma gli account dei membri non saranno associati all'organizzazione Amazon Inspector per i risultati centralizzati e la gestione degli account.

Per abilitare Amazon Inspector utilizzando le politiche AWS Organizations

1. Designare un amministratore delegato per Amazon Inspector prima di creare politiche organizzative per garantire che gli account dei membri siano associati all'organizzazione Amazon Inspector per una visibilità centralizzata dei risultati. Accedi all'account di AWS Organizations gestione, apri la console Amazon Inspector su <https://console.aws.amazon.com/inspector/v2/home> e segui i passaggi indicati. [Designazione di un amministratore delegato per l'organizzazione AWS](#)

Note

Ti consigliamo vivamente di mantenere uguali l'ID dell'account amministratore delegato di AWS Organizations Amazon Inspector e l'ID dell'account amministratore delegato designato da Amazon Inspector. Se l'ID dell'account amministratore AWS Organizations delegato è diverso dall'ID dell'account amministratore delegato di Amazon Inspector, Amazon Inspector dà la priorità all'ID dell'account designato da Amazon Inspector. Quando l'amministratore delegato di Amazon Inspector non è impostato ma l'amministratore AWS Organizations delegato è impostato e l'account di gestione ha i ruoli collegati al servizio Amazon Inspector, Amazon Inspector assegna automaticamente l'ID dell'account amministratore delegato come amministratore AWS Organizations delegato di Amazon Inspector.

2. Nella console Amazon Inspector, accedi alle impostazioni generali dall'account di gestione. In Politica di delega, scegli Allega dichiarazione. Nella finestra di dialogo Allega dichiarazione sulla politica, esamina la politica, seleziona Riconosco di aver esaminato la politica e di aver compreso le autorizzazioni che concede, quindi scegli Allega dichiarazione.

⚠ Important

L'account di gestione deve disporre delle seguenti autorizzazioni per allegare la dichiarazione sulla politica di delega:

- Autorizzazioni Amazon Inspector dalla policy gestita 2 [AmazonInspector_v2 FullAccess](#)
- AWS Organizations `organizations:PutResourcePolicy` autorizzazione dalla politica gestita [AWSOrganizationsFullAccess](#)

Se manca l'`organizations:PutResourcePolicy` autorizzazione, l'operazione ha esito negativo e viene visualizzato l'errore: `Failed to attach statement to the delegation policy.`

3. Successivamente, crea una policy Amazon Inspector AWS Organizations . Dal pannello di navigazione, scegli Gestione, quindi scegli Configurazioni.
4. Configura la politica di gestione delle vulnerabilità. Fornisci dettagli con nome e descrizione (facoltativo) per la politica.
5. Nella pagina Configure Inspector, nella sezione Dettagli, immettere un nome e una descrizione per la politica. Nella selezione delle capacità, effettuate una delle seguenti operazioni:
 - Scegliete Configura e abilitate tutte le funzionalità (consigliato). Ciò attiva tutte le funzionalità di Inspector, tra cui EC2, ECR, Lambda standard, Lambda code scan e Code Security.
 - Scegli Selezione sottoinsieme di funzionalità. Seleziona qualsiasi funzionalità del tipo di scansione che deve essere attivata.
6. Nella sezione Selezione dell'account, seleziona una delle seguenti opzioni:
 - Scegli Tutte le unità organizzative e gli account se desideri applicare la configurazione a tutte le unità organizzative e gli account.
 - Scegli Unità organizzative e account specifici se desideri applicare la configurazione a unità organizzative e account specifici. Se scegli questa opzione, utilizza la barra di ricerca o l'albero della struttura organizzativa per specificare le unità organizzative e gli account a cui verrà applicata la politica.

- Scegli Nessuna unità organizzativa o account se non desideri applicare la configurazione a nessuna unità organizzativa o account.
7. Nella sezione Regioni, scegli Abilita tutte le regioni, Disabilita tutte le regioni o Specificare le regioni.
- Se scegli Abilita tutte le regioni, puoi determinare se abilitare automaticamente le nuove regioni.
 - Se scegli Disabilita tutte le regioni, puoi determinare se disabilitare automaticamente le nuove regioni.
 - Se scegli Specificare le regioni, devi scegliere quali regioni vuoi abilitare e disabilitare.


(Facoltativo) Per le impostazioni avanzate, fate riferimento alla guida di AWS Organizations.

(Facoltativo) Per i tag Resource, aggiungete i tag come coppie chiave-valore per identificare facilmente la configurazione.

8. Scegli Avanti, rivedi le modifiche, quindi scegli Applica. I tuoi account target sono configurati in base alla politica. Lo stato di configurazione della politica viene visualizzato nella parte superiore della pagina Politiche. Ogni funzionalità indica se è stata configurata o se si verificano errori di distribuzione. Per eventuali errori, scegli il link relativo al messaggio di errore per visualizzare maggiori dettagli. Per visualizzare la politica efficace a livello di account, puoi consultare la scheda Organizzazione nella pagina Configurazioni, dove puoi scegliere un account.

Quando Amazon Inspector è abilitato tramite le politiche dell'organizzazione, gli account coperti dalla politica non possono disabilitare i tipi di scansione gestiti dalle policy tramite l'API o la console di Amazon Inspector. Per informazioni dettagliate su ciò che gli amministratori delegati e gli account dei membri possono e non possono fare in base alle politiche dell'organizzazione, consulta [Gestione di più account in Amazon Inspector con AWS Organizations](#)

Multi-account (without AWS Organizations policy)

 Note

È necessario utilizzare l'account AWS Organizations di gestione per completare questa procedura. Solo l'account AWS Organizations di gestione può designare un amministratore delegato. Potrebbero essere necessarie autorizzazioni per designare un

amministratore delegato. Per ulteriori informazioni, consulta [Autorizzazioni necessarie per designare un amministratore delegato](#).

Quando attivi Amazon Inspector per la prima volta, Amazon Inspector crea il `AWSServiceRoleForAmazonInspector` ruolo collegato al servizio per l'account. Per informazioni su come Amazon Inspector utilizza i ruoli collegati ai servizi, consulta [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#)

Per designare un amministratore delegato per Amazon Inspector

1. [Accedi all'account di AWS Organizations gestione, quindi apri la console Amazon Inspector su `https://console.aws.amazon.com/inspector/v2/home`.](https://console.aws.amazon.com/inspector/v2/home)
2. Scegli Avvia.
3. In Amministratore delegato, inserisci l'ID a 12 cifre dell'ID Account AWS che desideri designare come amministratore delegato.
4. Scegli Delegato, quindi scegli nuovamente Delegato.
5. (Facoltativo) Se desideri attivare Amazon Inspector per l'account di AWS Organizations gestione, scegli Attiva Amazon Inspector in Autorizzazioni di servizio.

Quando si designa un amministratore delegato, per impostazione predefinita vengono attivati [tutti i tipi di scansione](#) per l'account. Per informazioni sull'account amministratore delegato, consulta [Comprendere l'account amministratore delegato e gli account dei membri in Amazon Inspector](#).

Tipi di scansione automatizzati in Amazon Inspector

Amazon Inspector utilizza un motore di scansione appositamente progettato che monitora le risorse alla ricerca di vulnerabilità software utilizzabili e di esposizione involontaria della rete. [Quando Amazon Inspector rileva una vulnerabilità del software o un'esposizione involontaria della rete, crea una scoperta](#). Quando attivi Amazon Inspector per la prima volta, il tuo account viene registrato automaticamente a [tutti i tipi di scansione, tra cui la scansione](#) Amazon Amazon EC2, Amazon ECR Scanning e la scansione standard Lambda.

Note

La scansione del codice Lambda è un livello opzionale di scansione delle funzioni Lambda che puoi attivare in qualsiasi momento.

Argomenti

- [Panoramica dei tipi di scansione di Amazon Inspector](#)
- [Attivazione di un tipo di scansione](#)
- [Scansione delle istanze Amazon EC2 con Amazon Inspector](#)
- [Scansione delle immagini dei container Amazon Elastic Container Registry con Amazon Inspector](#)
- [Scansione AWS Lambda funzioni con Amazon Inspector](#)
- [Disattivazione di un tipo di scansione in Amazon Inspector](#)

Panoramica dei tipi di scansione di Amazon Inspector

Amazon Inspector offre diversi tipi di scansione, che si concentrano su tipi di risorse specifici nel tuo AWS ambiente.

Scansione Amazon EC2

Quando attivi la scansione di Amazon EC2, Amazon Inspector analizza le istanze EC2 alla ricerca di vulnerabilità ed esposizioni comuni (CVE), problemi di esposizione della rete, problemi di raggiungibilità della rete, vulnerabilità del sistema operativo e del pacchetto del linguaggio di programmazione. Amazon Inspector esegue le scansioni utilizzando l'agente SSM installato sull'istanza o tramite istantanee delle istanze di Amazon EBS. Per ulteriori informazioni, consulta

[Scansione delle istanze Amazon EC2 con Amazon Inspector](#). Per impostazione predefinita, quando attivi la scansione di Amazon EC2, abiliti automaticamente la modalità di scansione ibrida. Per ulteriori informazioni, consulta la sezione Scansione [senza agenti](#).

Scansione Amazon ECR

Quando attivi la scansione Amazon ECR, Amazon Inspector converte tutti i repository del tuo registro privato da repository di container per la scansione di base a repository di container a scansione avanzata. Puoi configurare questa impostazione con regole di inclusione per eseguire la scansione solo in modalità push o per scansionare repository selezionati. Amazon Inspector analizza solo le immagini dei contenitori ECR che sono attivi (il `imageStatus` campo è `ACTIVE`) in ECR. Amazon Inspector scansiona tutte le immagini inviate o passate ad active (`lastActivatedAt`) in ECR negli ultimi 30 giorni o recuperate negli ultimi 90 giorni. Amazon Inspector continua a monitorare le immagini per 90 giorni per impostazione predefinita. Puoi modificare questa impostazione in qualsiasi momento. Per ulteriori informazioni, consulta [Scansione delle immagini dei container Amazon Elastic Container Registry con Amazon Inspector](#).

Scansione standard Lambda

Quando attivi la scansione standard Lambda, Amazon Inspector rileva tutte le funzioni Lambda nel tuo account e le analizza immediatamente per individuare eventuali vulnerabilità. Amazon Inspector analizza nuove funzioni e livelli Lambda quando vengono distribuiti. Amazon Inspector li analizza nuovamente quando vengono aggiornati o quando vengono pubblicati nuovi CVE. Per ulteriori informazioni, scansione, consulta. [Scansione AWS Lambda funzioni con Amazon Inspector](#)

Scansione standard Lambda + scansione del codice Lambda

Quando attivi la scansione del codice Lambda, Amazon Inspector rileva le funzioni e i livelli Lambda nel tuo account e li analizza per individuare eventuali vulnerabilità del codice. Questo tipo di scansione valuta le dipendenze dei pacchetti applicativi utilizzati in una funzione Lambda per CVE. Quando si attiva questo tipo di scansione, si attiva anche la scansione standard Lambda. Per ulteriori informazioni, consulta [Scansione AWS Lambda funzioni con Amazon Inspector](#).

Codice di sicurezza per Amazon Inspector

Questo tipo di scansione sfrutta il motore di scansione Amazon Q Developer per scansionare il codice di applicazioni di prime parti, le dipendenze delle applicazioni di terze parti e Infrastructure as Code per individuare eventuali vulnerabilità. Per ulteriori informazioni, consulta [Code Security for Amazon Inspector](#).

Attivazione di un tipo di scansione

È possibile attivare un tipo di scansione in qualsiasi momento. Quando attivi un tipo di scansione, Amazon Inspector inizia a scansionare le risorse idonee per quel tipo di scansione.

[Scansione Amazon EC2](#)

Questo tipo di scansione estrae i metadati da un'istanza Amazon EC2 prima di confrontarli con le regole raccolte dagli avvisi di sicurezza. Quando attivi questo tipo di scansione, Amazon Inspector analizza tutte le istanze Amazon EC2 idonee del tuo account alla ricerca di vulnerabilità dei pacchetti e problemi di raggiungibilità della rete. Dopo aver attivato questo tipo di scansione, puoi visualizzare quante istanze vengono scansionate nella scheda Istanze.

[Scansione Amazon ECR](#)

Questo tipo di scansione analizza le immagini e gli archivi dei container in Amazon ECR. Quando attivi questo tipo di scansione, modifichi l'impostazione di configurazione della scansione per il registro privato dalla scansione di base alla scansione avanzata. Dopo aver attivato la scansione Amazon ECR, puoi visualizzare quante immagini e repository vengono scansionati nelle schede Immagini container e Repository container.

[Scansione standard Lambda](#) + [Scansione del codice Lambda](#)

La scansione Lambda standard è il tipo di scansione Lambda predefinito. Quando attivi la scansione standard Lambda, tutte le funzioni Lambda vengono analizzate alla ricerca di vulnerabilità del software, purché siano state richiamate o aggiornate negli ultimi 90 giorni. Dopo aver attivato la scansione standard Lambda, puoi visualizzare quante funzioni Lambda vengono scansionate nella scheda Funzioni Lambda.

La scansione del codice Lambda analizza il codice dell'applicazione personalizzato in una funzione Lambda. Quando attivi la scansione del codice Lambda, tutte le funzioni Lambda verranno analizzate alla ricerca di vulnerabilità del codice, purché siano state richiamate o aggiornate negli ultimi 90 giorni. Dopo aver attivato la scansione standard Lambda, puoi visualizzare quante funzioni Lambda vengono scansionate per individuare eventuali vulnerabilità del codice nella scheda Funzioni Lambda.

Note

Se desideri attivare la scansione del codice Lambda, devi prima attivare la scansione standard Lambda.

Codice di sicurezza di Amazon Inspector

Questo tipo di scansione analizza il codice applicativo di prime parti, le dipendenze delle applicazioni di terze parti e Infrastructure as Code alla ricerca di vulnerabilità. Quando attivi Code Security, Amazon Inspector inizia a scansionare i tuoi repository di codice alla ricerca di vulnerabilità del codice in base alle configurazioni di scansione. Dopo aver attivato Amazon Inspector Code Security, puoi visualizzare quanti repository di codice vengono scansionati nella scheda Code repository.

Attivazione delle scansioni

La procedura seguente descrive come attivare un tipo di scansione in Amazon Inspector.

Note

Se sei l'amministratore delegato di un' AWS organizzazione, puoi abilitare i tipi di scansione di Amazon Inspector per più account in più regioni utilizzando uno script di shell. Per ulteriori informazioni, consulta [inspector2-enablement-with-cli on](#). GitHub Altrimenti, completa i seguenti passaggi dopo aver effettuato l'accesso come amministratore delegato di Amazon Inspector.

Console

Per attivare le scansioni

1. Apri la console Amazon Inspector all'indirizzo. <https://console.aws.amazon.com/inspector/v2/home>
2. Utilizzando il Regione AWS selettore nell'angolo superiore destro della pagina, seleziona la regione in cui desideri attivare un nuovo tipo di scansione.
3. Nel riquadro di navigazione, scegli Gestione account.
4. Nella pagina Gestione dell'account, seleziona gli account per i quali desideri attivare un tipo di scansione.
5. Scegli Attiva e seleziona il tipo di scansione che desideri attivare.
6. (Consigliato) Ripeti questi passaggi Regione AWS per ognuno dei quali desideri attivare quel tipo di scansione.

API

Esegui l'operazione [Enable](#) API. Nella richiesta, fornisci gli ID dell'account per cui stai attivando le scansioni, il token di idempotenza e uno o più di EC2, ECRLAMBDA, o LAMBDA_CODE resourceTypes per attivare scansioni di quel tipo.

Scansione delle istanze Amazon EC2 con Amazon Inspector

Amazon Inspector La scansione di Amazon EC2 estrae i metadati dall'istanza EC2 prima di confrontarli con le regole raccolte dagli avvisi di sicurezza. [Amazon Inspector analizza le istanze alla ricerca di vulnerabilità dei pacchetti e problemi di raggiungibilità della rete per produrre risultati.](#) Amazon Inspector esegue scansioni di raggiungibilità della rete una volta ogni 12 ore e scansioni di vulnerabilità dei pacchetti con una cadenza variabile che dipende dal metodo di scansione associato all'istanza EC2.

[Le scansioni delle vulnerabilità dei pacchetti possono essere eseguite utilizzando un metodo di scansione basato su agenti o senza agente.](#) Entrambi questi metodi di scansione determinano come e quando Amazon Inspector raccoglie l'inventario software da un'istanza EC2 per le scansioni delle vulnerabilità dei pacchetti. Agent-based la scansione raccoglie l'inventario software dalle istanze utilizzando l'agente Amazon EC2 Systems Manager (SSM), mentre la scansione senza agente raccoglie l'inventario software utilizzando le istantanee di Amazon EBS.

Amazon Inspector utilizza i metodi di scansione attivati per il tuo account. Quando attivi Amazon Inspector per la prima volta, il tuo account viene automaticamente registrato alla scansione ibrida, che utilizza entrambi i metodi di scansione. Tuttavia, puoi [modificare questa impostazione](#) in qualsiasi momento. Per informazioni su come attivare un tipo di scansione, vedere [Attivazione di un tipo di scansione](#). Questa sezione fornisce informazioni sulla scansione di Amazon EC2.

Agent-based scansione

Scansione EC2 migliorata

La scansione EC2 avanzata viene eseguita utilizzando [Scanner VM Amazon Inspector](#). Questo scanner viene installato e aggiornato utilizzando le associazioni SSM. I clienti possono effettuare l'attivazione accedendo alla console Amazon Inspector e visitando la pagina Impostazioni > Impostazioni di scansione. Scegli Avvia aggiornamento per iniziare la scansione EC2 avanzata.

1. Amazon Inspector crea associazioni SSM nel tuo account per raccogliere l'inventario dalle tue istanze. Queste associazioni installano plug-in su singole istanze per raccogliere l'inventario.

2. Utilizzando strumenti di sistema come Systemd e Scheduled Tasks, Inspector VM Scanner estrae l'inventario dei pacchetti da un'istanza e comunica tali informazioni ad Amazon Inspector.
3. Amazon Inspector valuta l'inventario estratto e genera risultati per eventuali vulnerabilità rilevate.

Scansione standard

Agent-based le scansioni vengono eseguite continuamente utilizzando il plug-in Amazon Inspector SSM su tutte le istanze idonee. Per le scansioni basate su agenti, Amazon Inspector utilizza le associazioni SSM e i plug-in installati tramite queste associazioni per raccogliere l'inventario software dalle tue istanze. [Oltre alle scansioni delle vulnerabilità dei pacchetti dei sistemi operativi, la scansione basata su agenti di Amazon Inspector può anche rilevare le vulnerabilità dei pacchetti per i pacchetti in linguaggi di programmazione delle applicazioni tramite l'ispezione approfondita di Amazon Inspector.](#)

Il seguente processo spiega come Amazon Inspector utilizza SSM per raccogliere l'inventario ed eseguire scansioni basate su agenti:

1. Amazon Inspector crea associazioni SSM nel tuo account per raccogliere l'inventario dalle tue istanze. Queste associazioni installano plug-in su singole istanze per raccogliere l'inventario.
2. Utilizzando SSM, Amazon Inspector estrae l'inventario dei pacchetti da un'istanza.
3. Amazon Inspector valuta l'inventario estratto e genera risultati per eventuali vulnerabilità rilevate.

Note

Per la scansione basata su agenti, l'istanza Amazon EC2 deve essere gestita da SSM all'interno dello stesso Account AWS

Requisiti degli endpoint Amazon VPC per la scansione EC2 avanzata su istanze Amazon EC2 private

Puoi eseguire Enhanced EC2 Scanning su istanze Amazon EC2 su una rete Amazon. Tuttavia, se desideri eseguire Enhanced EC2 Scanning su istanze Amazon EC2 private, devi creare endpoint Amazon VPC. Sono necessari i seguenti endpoint:

- `com.amazonaws.region.ec2messages`
- `com.amazonaws.region.inspector2-telemetry`

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

region Dov'è il codice regionale applicabile Regione AWS.

Per ulteriori informazioni, consulta [Migliorare la sicurezza delle istanze Amazon EC2 utilizzando gli endpoint Amazon VPC per Systems Manager nella Guida per l'utente](#).AWS Systems Manager

Note

Al momento, alcuni Regioni AWS non supportano l'endpoint.

`com.amazonaws.region.inspector2-telemetry`

Istanze idonee

Amazon Inspector utilizzerà il metodo basato su agenti per scansionare un'istanza se soddisfa le seguenti condizioni:

- L'istanza ha un sistema operativo supportato. Per un elenco dei sistemi operativi supportati, consulta la colonna di supporto alla Agent-based scansione di [the section called “Sistemi operativi supportati: Amazon EC2 scanning”](#).
- L'istanza non è esclusa dalle scansioni tramite i tag di esclusione di Amazon Inspector EC2.

Agent-based comportamenti di scansione

Quando si utilizza il metodo di scansione basato su agenti, Amazon Inspector avvia nuove scansioni di vulnerabilità delle istanze EC2 nelle seguenti situazioni:

- Quando avvii una nuova istanza EC2.
- Quando installi un nuovo software su un'istanza EC2 esistente (Linux e Mac).
- Quando Amazon Inspector aggiunge un nuovo elemento CVE (Common Vulnerabilities and Exposures) al suo database e tale CVE è rilevante per la tua istanza EC2 (Linux e Mac).

Amazon Inspector aggiorna il campo Ultima scansione per un'istanza EC2 quando viene completata una scansione iniziale. Dopodiché, il campo Ultima scansione viene aggiornato quando Amazon

Inspector valuta l'inventario SSM (ogni 30 minuti per impostazione predefinita) o quando un'istanza viene nuovamente scansionata perché al database Amazon Inspector è stato aggiunto un nuovo CVE che ha un impatto su quell'istanza.

Puoi verificare quando un'istanza EC2 è stata analizzata l'ultima volta per individuare eventuali vulnerabilità dalla scheda Istanze nella pagina di gestione dell'account o utilizzando il comando.

[ListCoverage](#)

Configurazione dell'agente SSM

Affinché Amazon Inspector rilevi le vulnerabilità del software per un'istanza Amazon EC2 utilizzando il metodo di scansione basato su agenti, l'istanza deve essere un'istanza gestita in Amazon [EC2 Systems Manager](#) (SSM). Un'istanza gestita da SSM ha l'agente SSM installato e in esecuzione e SSM dispone dell'autorizzazione per gestire l'istanza. Se stai già utilizzando SSM per gestire le tue istanze, non sono necessari altri passaggi per le scansioni basate su agenti.

L'agente SSM è installato per impostazione predefinita sulle istanze EC2 create da alcune Amazon Machine Images (AMI). Per ulteriori informazioni, consulta Informazioni [su SSM Agent](#) nella Guida per l'utente. AWS Systems Manager Tuttavia, anche se è installato, potrebbe essere necessario attivare l'agente SSM manualmente e concedere l'autorizzazione SSM per gestire l'istanza.

La procedura seguente descrive come configurare un'istanza Amazon EC2 come istanza gestita utilizzando un profilo di istanza IAM. La procedura fornisce anche collegamenti a informazioni più dettagliate nella Guida per l'AWS Systems Manager utente.

[AmazonSSMManagedInstanceCore](#) è la politica consigliata da utilizzare quando si collega un profilo di istanza. Questa policy dispone di tutte le autorizzazioni necessarie per la scansione di Amazon Inspector EC2.

Note

Puoi anche automatizzare la gestione SSM di tutte le tue istanze EC2, senza l'uso di profili di istanza IAM, utilizzando SSM Default Host Management Configuration. Per ulteriori informazioni, consulta la pagina [Configurazione di gestione host predefinita](#).

Per configurare SSM per un'istanza Amazon EC2

1. Se non è già installato dal fornitore del sistema operativo, installa l'agente SSM. Per ulteriori informazioni, consulta [Working with SSM Agent](#).

2. Utilizzare il AWS CLI per verificare che l'agente SSM sia in esecuzione. Per ulteriori informazioni, consulta [Verifica dello stato dell'agente SSM e avvio dell'agente](#).
3. Concedi l'autorizzazione a SSM per gestire la tua istanza. Puoi concedere l'autorizzazione creando un profilo di istanza IAM e collegandolo alla tua istanza. Ti consigliamo di utilizzare la [AmazonSSMManagedInstanceCore](#) policy, poiché questa policy ha le autorizzazioni per SSM Distributor, SSM Inventory e SSM State manager, di cui Amazon Inspector ha bisogno per le scansioni. Per istruzioni su come creare un profilo di istanza con queste autorizzazioni e collegarlo a un'istanza, vedere [Configurare le autorizzazioni dell'istanza per Systems Manager](#).
4. (Facoltativo) Attiva gli aggiornamenti automatici per l'agente SSM. Per ulteriori informazioni, consulta [Automazione degli aggiornamenti all'agente SSM](#).
5. (Facoltativo) Configura Systems Manager per utilizzare un endpoint Amazon Virtual Private Cloud (Amazon VPC). Per ulteriori informazioni, consulta [Creazione di endpoint Amazon VPC](#).

Risorse SSM create per la scansione

Amazon Inspector richiede diverse risorse SSM nel tuo account per eseguire le scansioni di Amazon EC2. Le seguenti risorse vengono create quando attivi per la prima volta la scansione di Amazon Inspector EC2:

Note

Se una di queste risorse SSM viene eliminata mentre la scansione Amazon Inspector Amazon EC2 è attivata per il tuo account, Amazon Inspector tenterà di ricrearla all'intervallo di scansione successivo.

InspectorInventoryCollection-do-not-delete

Si tratta di un'associazione Systems Manager State Manager (SSM) che Amazon Inspector utilizza per raccogliere l'inventario delle applicazioni software dalle istanze Amazon EC2. Se il tuo account dispone già di un'associazione SSM per la raccolta dell'inventario InstanceIds*, Amazon Inspector la utilizzerà invece di crearne una propria.

InspectorResourceDataSync-do-not-delete

Si tratta di una sincronizzazione dei dati delle risorse che Amazon Inspector utilizza per inviare i dati di inventario raccolti dalle istanze Amazon EC2 a un bucket Amazon S3 di proprietà di

Amazon Inspector. Per ulteriori informazioni, consulta [Configurazione della sincronizzazione dei dati delle risorse](#) per l'inventario nella Guida per l'utente AWS Systems Manager

InspectorVmScannerDistributor-do-not-delete

Si tratta di un'associazione SSM che Amazon Inspector utilizza per installarla e aggiornarla [Scanner VM Amazon Inspector](#) sulle istanze Amazon EC2.

InspectorDistributor-do-not-delete

Si tratta di un'associazione SSM utilizzata da Amazon Inspector per la scansione delle istanze di Windows. Questa associazione installa il plug-in Amazon Inspector SSM sulle tue istanze Windows. Se il file del plug-in viene eliminato inavvertitamente, questa associazione lo reinstallerà all'intervallo di associazione successivo.

InvokeInspectorSsmPlugin-do-not-delete

Si tratta di un'associazione SSM utilizzata da Amazon Inspector per la scansione delle istanze di Windows. Questa associazione consente ad Amazon Inspector di avviare scansioni utilizzando il plug-in, inoltre puoi utilizzarlo per impostare intervalli personalizzati per le scansioni delle istanze di Windows. Per ulteriori informazioni, consulta [Impostazione di pianificazioni personalizzate per Windows scansioni delle istanze](#).

InspectorLinuxDistributor-do-not-delete

Si tratta di un'associazione SSM utilizzata da Amazon Inspector per l'ispezione approfondita di Amazon EC2 Linux. Questa associazione installa il plug-in Amazon Inspector SSM sulle tue istanze Linux.

InvokeInspectorLinuxSsmPlugin-do-not-delete

Si tratta di un'associazione SSM utilizzata da Amazon Inspector per l'ispezione approfondita di Amazon EC2 Linux. Questa associazione consente ad Amazon Inspector di avviare scansioni utilizzando il plug-in.

Note

Quando disattivi la scansione o l'ispezione approfondita di Amazon Inspector Amazon EC2, la risorsa SSM `InvokeInspectorLinuxSsmPlugin-do-not-delete` non viene più richiamata.

Scansione senza agenti

Amazon Inspector utilizza il metodo di scansione senza agenti su istanze idonee quando l'account è in modalità di scansione ibrida. La modalità di scansione ibrida include scansioni basate su agenti e senza agenti e viene abilitata automaticamente quando si attiva la scansione Amazon EC2.

Per le scansioni senza agenti, Amazon Inspector utilizza le istantanee EBS per raccogliere un inventario software dalle tue istanze. La scansione senza agente analizza le istanze alla ricerca di vulnerabilità del sistema operativo e dei pacchetti del linguaggio di programmazione delle applicazioni.

Note

Durante la scansione delle istanze Linux alla ricerca delle vulnerabilità dei pacchetti del linguaggio di programmazione delle applicazioni, il metodo agentless analizza tutti i percorsi disponibili, mentre la scansione basata su agenti analizza solo i percorsi predefiniti e i percorsi aggiuntivi specificati come parte di [Ispezione approfondita di Amazon Inspector per istanze Amazon Linux-based EC2](#). Ciò può comportare che la stessa istanza abbia risultati diversi a seconda che venga scansionata utilizzando il metodo basato su agenti o il metodo senza agenti.

Il seguente processo spiega come Amazon Inspector utilizza gli snapshot EBS per raccogliere l'inventario ed eseguire scansioni senza agenti:

1. Amazon Inspector crea uno snapshot EBS di tutti i volumi collegati all'istanza. Mentre Amazon Inspector lo utilizza, lo snapshot viene archiviato nel tuo account e contrassegnato InspectorScan come chiave di tag e un ID di scansione univoco come valore del tag.
2. Amazon Inspector recupera i dati dagli snapshot utilizzando le [API dirette di EBS e li valuta](#) per individuare eventuali vulnerabilità. I risultati vengono generati per tutte le vulnerabilità rilevate.
3. Amazon Inspector elimina gli snapshot EBS creati nel tuo account.

Istanze idonee

Amazon Inspector utilizzerà il metodo agentless per scansionare un'istanza se soddisfa le seguenti condizioni:

- L'istanza ha un sistema operativo supportato. Per ulteriori informazioni, consulta la colonna > supporto alla Agent-based scansione di [the section called “Sistemi operativi supportati: Amazon EC2 scanning”](#).
- Lo stato dell'istanza è pari a Unmanaged EC2 instanceStale inventory, oNo inventory.
- L'istanza è supportata da Amazon EBS e ha uno dei seguenti formati di file system:
 - ext3
 - ext4
 - xfs
- L'istanza non è esclusa dalle scansioni tramite i tag di esclusione di Amazon EC2.
- Il numero di volumi collegati all'istanza è inferiore a 8 e hanno una dimensione combinata inferiore o uguale a 1200 GB.

Comportamenti di scansione senza agenti

Quando il tuo account è configurato per la scansione ibrida, Amazon Inspector esegue scansioni senza agente su istanze idonee ogni 24 ore. Amazon Inspector rileva e analizza le nuove istanze idonee ogni ora, incluse nuove istanze senza agenti SSM o istanze preesistenti con stato modificato in. SSM_UNMANAGED

Amazon Inspector aggiorna il campo Ultima scansione per un'istanza Amazon EC2 ogni volta che esegue la scansione degli snapshot estratti da un'istanza dopo una scansione senza agente.

Puoi verificare quando un'istanza EC2 è stata analizzata l'ultima volta per individuare eventuali vulnerabilità dalla scheda Istanze nella pagina di gestione dell'account o utilizzando il comando.

[ListCoverage](#)

Gestione della modalità di scansione

La modalità di scansione EC2 determina i metodi di scansione che Amazon Inspector utilizzerà per eseguire le scansioni EC2 nel tuo account. Puoi visualizzare la modalità di scansione del tuo account dalla pagina delle impostazioni di scansione EC2 in Impostazioni generali. Gli account autonomi o gli amministratori delegati di Amazon Inspector possono modificare la modalità di scansione. Quando imposti la modalità di scansione come amministratore delegato di Amazon Inspector, tale modalità di scansione viene impostata per tutti gli account membri della tua organizzazione. Amazon Inspector offre le seguenti modalità di scansione:

Agent-based scansione: in questa modalità di scansione, Amazon Inspector utilizzerà esclusivamente il metodo di scansione basato su agenti per la scansione delle vulnerabilità dei pacchetti. Questa modalità di scansione analizza solo le istanze gestite tramite SSM presenti nel tuo account, ma ha il vantaggio di fornire scansioni continue in risposta a nuovi CVE o modifiche alle istanze. Agent-based la scansione fornisce anche un'ispezione approfondita di Amazon Inspector per le istanze idonee. Questa è la modalità di scansione predefinita per gli account appena attivati.

Scansione ibrida: in questa modalità di scansione, Amazon Inspector utilizza una combinazione di metodi basati su agenti e senza agenti per individuare le vulnerabilità dei pacchetti. Per le istanze EC2 idonee su cui è installato e configurato l'agente SSM, Amazon Inspector utilizza il metodo basato su agenti. Per le istanze idonee che non sono gestite tramite SSM, Amazon Inspector utilizzerà il metodo agentless per le istanze idonee. EBS-backed

Per modificare la modalità di scansione

1. Accedi utilizzando le tue credenziali, quindi apri la console Amazon Inspector all'indirizzo. <https://console.aws.amazon.com/inspector/v2/home>
2. Utilizzando il Regione AWS selettore nell'angolo in alto a destra della pagina, seleziona la regione in cui desideri modificare la modalità di scansione EC2.
3. Dal pannello di navigazione laterale, in Impostazioni generali, seleziona Impostazioni di scansione EC2.
4. In Modalità di scansione, seleziona Modifica.
5. Scegli una modalità di scansione, quindi seleziona Salva modifiche.

Esclusione delle istanze dalle scansioni di Amazon Inspector

Puoi escludere Windows istanze Linux e dalle scansioni di Amazon Inspector etichettando queste istanze con la chiave. `InspectorEc2Exclusion` La chiave tag non fa distinzione tra maiuscole e minuscole. L'inclusione di un valore di tag è facoltativa. Per informazioni sull'aggiunta di tag, consulta [Etichettare le risorse Amazon EC2](#).

Quando tagghi un'istanza per l'esclusione dalle scansioni di Amazon Inspector, Amazon Inspector contrassegna l'istanza come esclusa e non crea risultati per essa. Tuttavia, il plug-in Amazon Inspector SSM continuerà a essere richiamato. Per evitare che il plug-in venga richiamato, devi [consentire l'accesso ai tag](#) nei metadati dell'istanza.

Note

Non ti viene addebitato alcun costo per le istanze escluse.

Inoltre, puoi escludere un'istanza dalle scansioni senza agente contrassegnando con il tag la AWS KMS chiave utilizzata per crittografare quel volume. InspectorEc2Exclusion [Per ulteriori informazioni, consulta Etichettatura delle chiavi.](#)

Sistemi operativi supportati

Amazon Inspector analizza le istanze Mac, Windows e Linux supportate alla ricerca di vulnerabilità nei pacchetti del sistema operativo. Per le istanze Linux, Amazon Inspector può produrre risultati per i pacchetti di linguaggi di programmazione delle applicazioni che utilizzano. [Ispezione approfondita di Amazon Inspector per istanze Amazon Linux-based EC2](#) Per le istanze Mac e Windows vengono scansionati solo i pacchetti del sistema operativo.

Per informazioni sui sistemi operativi supportati, inclusi i sistemi operativi che possono essere scansionati senza un agente SSM, consulta. [Valori di stato delle istanze Amazon EC2](#)

Ispezione approfondita di Amazon Inspector per istanze Amazon Linux-based EC2

Note

Questa pagina si applica ai clienti che non hanno optato per Enhanced EC2 Scanning. I clienti che hanno optato per Enhanced EC2 Scanning non necessitano di alcuna configurazione di policy aggiuntiva.

Amazon Inspector amplia la copertura di scansione di Amazon EC2 per includere l'ispezione approfondita. Con un'ispezione approfondita, Amazon Inspector rileva le vulnerabilità dei pacchetti di linguaggi di programmazione delle applicazioni nelle istanze Amazon Linux-based EC2. Amazon Inspector analizza i percorsi predefiniti per individuare le librerie di pacchetti del linguaggio di programmazione. Tuttavia, puoi [configurare percorsi personalizzati oltre ai percorsi](#) che Amazon Inspector analizza per impostazione predefinita.

Note

Requisiti `ssm:PutInventory` e `ssm:GetParameter` autorizzazioni per un'ispezione approfondita. Se sull'istanza è configurato un profilo di istanza IAM, Amazon Inspector utilizza tale profilo e ignora il ruolo DHMC. Il profilo dell'istanza deve includere queste autorizzazioni. Se non è impostato alcun profilo di istanza, Amazon Inspector utilizza il ruolo [Default Host Management Configuration](#) configurato, che deve includere queste autorizzazioni.

Per eseguire scansioni di ispezione approfondita per le istanze Linux-based Amazon EC2, Amazon Inspector utilizza i dati raccolti con il plug-in Amazon Inspector SSM. Per gestire il plug-in Amazon Inspector SSM ed eseguire un'ispezione approfondita per Linux, Amazon Inspector crea automaticamente l'associazione SSM nel tuo account. `InvokeInspectorLinuxSsmPlugin-donot-delete` Amazon Inspector raccoglie l'inventario aggiornato delle applicazioni dalle tue istanze Linux-based Amazon EC2 ogni 6 ore.

Note

L'ispezione approfondita non è supportata per Windows e le nostre istanze Mac.

Questa sezione descrive come gestire l'ispezione approfondita di Amazon Inspector per le istanze Amazon EC2, incluso come impostare percorsi personalizzati per la scansione di Amazon Inspector.

Argomenti

- [Accesso o disattivazione dell'ispezione approfondita](#)
- [Percorsi personalizzati per l'ispezione approfondita di Amazon Inspector](#)
- [Pianificazioni personalizzate per l'ispezione approfondita di Amazon Inspector](#)
- [Linguaggi di programmazione compatibili](#)

Accesso o disattivazione dell'ispezione approfondita**Note**

Per gli account che attivano Amazon Inspector dopo il 17 aprile 2023, l'ispezione approfondita viene attivata automaticamente come parte della scansione di Amazon EC2.

Per gestire l'ispezione approfondita

1. Accedi utilizzando le tue credenziali, quindi apri la console Amazon Inspector all'indirizzo <https://console.aws.amazon.com/inspector/v2/home>
2. Dal pannello di navigazione, scegli Impostazioni generali, quindi scegli Impostazioni di scansione Amazon EC2.
3. Nell'ambito dell'ispezione approfondita dell'istanza Amazon EC2, puoi [impostare percorsi personalizzati per la tua organizzazione o per il tuo account](#).

Puoi controllare lo stato di attivazione in modo programmatico per un singolo account con l'API. [GetEc2DeepInspectionConfiguration](#) Puoi controllare lo stato di attivazione a livello di codice per più account con l'API. [BatchGetMemberEc2DeepInspectionStatus](#)

Se hai attivato Amazon Inspector prima del 17 aprile 2023, puoi attivare l'ispezione approfondita tramite il banner della console o l'[UpdateEc2DeepInspectionConfiguration](#) API. Se sei l'amministratore delegato di un'organizzazione in Amazon Inspector, puoi utilizzare l'API per attivare [BatchUpdateMemberEc2DeepInspectionStatus](#) l'ispezione approfondita per te e per i tuoi account membro.

Puoi disattivare l'ispezione approfondita tramite l'API. [UpdateEc2DeepInspectionConfiguration](#) Gli account dei membri di un'organizzazione non possono disattivare l'ispezione approfondita. Invece, l'account membro deve essere disattivato dall'amministratore delegato utilizzando l'API. [BatchUpdateMemberEc2DeepInspectionStatus](#)

Percorsi personalizzati per l'ispezione approfondita di Amazon Inspector

Puoi impostare percorsi personalizzati per la scansione di Amazon Inspector durante l'ispezione approfondita delle tue istanze Linux Amazon EC2. Quando imposti un percorso personalizzato, Amazon Inspector analizza i pacchetti in quella directory e in tutte le sue sottodirectory.

Tutti gli account possono definire fino a 5 percorsi personalizzati. L'amministratore delegato di un'organizzazione può definire 10 percorsi personalizzati.

Amazon Inspector analizza tutti i percorsi personalizzati oltre ai seguenti percorsi predefiniti, che Amazon Inspector analizza per tutti gli account:

- /usr/lib
- /usr/lib64

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

Note

I percorsi personalizzati devono essere percorsi locali. Amazon Inspector non esegue la scansione di percorsi di rete mappati, ad esempio montaggi di Network File System o supporti di file system Amazon S3.

Formattazione di percorsi personalizzati

Un percorso personalizzato non può contenere più di 256 caratteri. Di seguito è riportato un esempio di come potrebbe apparire un percorso personalizzato:

Percorso di esempio

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

Note

Il limite di pacchetti per istanza è 5.000. Il tempo massimo di raccolta dell'inventario dei pacchi è di 15 minuti. Amazon Inspector consiglia di scegliere percorsi personalizzati per evitare questi limiti.

Impostazione di un percorso personalizzato nella console Amazon Inspector e con l'API Amazon Inspector

Le seguenti procedure descrivono come impostare un percorso personalizzato per l'ispezione approfondita di Amazon Inspector nella console Amazon Inspector e con l'API Amazon Inspector. Dopo aver impostato un percorso personalizzato, Amazon Inspector lo include nella successiva ispezione approfondita.

Console

1. Accedi Console di gestione AWS come amministratore delegato e apri la console Amazon Inspector all'indirizzo <https://console.aws.amazon.com/inspector/v2/home>

2. Usa il Regione AWS selettore per scegliere la regione in cui desideri attivare la scansione standard Lambda.
3. Dal pannello di navigazione, scegli Impostazioni generali, quindi scegli Impostazioni di scansione EC2.
4. In Percorsi personalizzati per il tuo account, scegli Modifica.
5. Nelle caselle di testo dei percorsi, inserisci i percorsi personalizzati.
6. Scegli Save (Salva).

API

Esegui il comando [UpdateEc2DeepInspectionConfiguration](#). Per `packagePaths` specificare una serie di percorsi da scansionare.

Pianificazioni personalizzate per l'ispezione approfondita di Amazon Inspector

Per impostazione predefinita, Amazon Inspector raccoglie un inventario delle applicazioni dalle istanze Amazon EC2 ogni 6 ore. Tuttavia, puoi eseguire i seguenti comandi per controllare la frequenza con cui Amazon Inspector esegue questa operazione.

Comando di esempio 1: elenca le associazioni per visualizzare l'ID dell'associazione e l'intervallo corrente

Il comando seguente mostra l'ID dell'associazione `InvokeInspectorLinuxSsmPlugin-do-not-delete`.

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

Comando di esempio 2: Aggiorna l'associazione per includere un nuovo intervallo

Il comando seguente utilizza l'ID dell'associazione per l'associazione `InvokeInspectorLinuxSsmPlugin-do-not-delete`. È possibile impostare la frequenza `schedule-expression` da 6 ore a un nuovo intervallo, ad esempio 12 ore.

```
aws ssm update-association \  

```

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

Note

A seconda del caso d'uso, se imposti la frequenza `schedule-expression` da 6 ore a un intervallo di 30 minuti, puoi [superare il limite giornaliero di inventario ssm](#). Ciò causa un ritardo nei risultati e potresti riscontrare istanze Amazon EC2 con stati di errore parziali.

Linguaggi di programmazione compatibili

Per le istanze Linux, l'ispezione approfondita di Amazon Inspector può produrre risultati per i pacchetti di linguaggi di programmazione delle applicazioni e i pacchetti del sistema operativo.

Per le istanze Mac e Windows, l'ispezione approfondita di Amazon Inspector può produrre risultati solo per i pacchetti del sistema operativo.

Per ulteriori informazioni sui linguaggi di programmazione supportati, consulta [Linguaggi di programmazione supportati: Amazon EC2 deep inspection](#).

Scansione Windows Istanze EC2 con Amazon Inspector

Note

Questa pagina si applica ai clienti che non hanno optato per Enhanced EC2 Scanning.

Amazon Inspector rileva automaticamente tutte le Windows istanze supportate e le include nella scansione continua senza azioni aggiuntive. Per informazioni sulle istanze supportate, consulta [Sistemi operativi e linguaggi di programmazione supportati da Amazon Inspector](#). Amazon Inspector esegue le Windows scansioni a intervalli regolari. Windowsle istanze vengono scansionate al momento del rilevamento e successivamente ogni 6 ore. Tuttavia, è possibile [regolare l'intervallo di scansione predefinito](#) dopo la prima scansione.

Quando la scansione di Amazon EC2 è attivata, Amazon Inspector crea le seguenti associazioni SSM per le Windows tue risorse `InspectorDistributor-do-not-delete`:

e. `InspectorInventoryCollection-do-not-delete InvokeInspectorSsmPlugin-do-not-delete` [Per installare il plug-in Amazon Inspector SSM sulle tue Windows istanze, l'associazione SSM utilizza il documento `InspectorDistributor-do-not-delete SSM` e il pacchetto `AWS-ConfigureAWSPackage SSM Distributor. AmazonInspector2-InspectorSsmPlugin`](#) Per ulteriori informazioni, consulta [il plugin Amazon Inspector SSM](#) per. Windows Per raccogliere dati sulle istanze e generare risultati di Amazon Inspector, l'associazione `InvokeInspectorSsmPlugin-do-not-delete SSM` esegue il plug-in Amazon Inspector SSM a intervalli di 6 ore. Tuttavia, puoi [personalizzare questa impostazione utilizzando](#) un'espressione cron o rate.

Note

Amazon Inspector inserisce i file di definizione Open Vulnerability and Assessment Language (OVAL) aggiornati nel bucket S3. `inspector2-oval-prod-your-AWS-Region` Il bucket Amazon S3 contiene le definizioni OVAL utilizzate nelle scansioni. Queste definizioni OVAL non devono essere modificate. Altrimenti, Amazon Inspector non cercherà nuovi CVE al momento del rilascio.

Requisiti di scansione di Amazon Inspector per Windows Istanze

Per eseguire la scansione di un'Windowsistanza, Amazon Inspector richiede che l'istanza soddisfi i seguenti criteri:

- L'istanza è un'istanza gestita da SSM. Per istruzioni sulla configurazione dell'istanza per la scansione, consulta [Configurazione dell'agente SSM](#).
- Il sistema operativo dell'istanza è uno dei sistemi Windows operativi supportati. Per un elenco completo dei sistemi operativi supportati, vedere [Valori di stato delle istanze Amazon EC2](#).
- Nell'istanza è installato il plug-in Amazon Inspector SSM. Amazon Inspector installa automaticamente il plug-in Amazon Inspector SSM per le istanze gestite al momento del rilevamento. Per informazioni dettagliate sul plug-in, consulta l'argomento successivo.

Note

Se il tuo host è in esecuzione su un Amazon VPC senza accesso a Internet in uscita, Windows la scansione richiede che l'host sia in grado di accedere agli endpoint Amazon S3 regionali. Per informazioni su come configurare un endpoint Amazon VPC Amazon

S3, consulta [Creare un endpoint gateway](#) nella Amazon Virtual Private Cloud User Guide. Se la tua policy sugli endpoint di Amazon VPC limita l'accesso ai bucket S3 esterni, devi specificamente consentire l'accesso al bucket gestito da Amazon Inspector nella Regione AWS tuo che memorizza le definizioni OVAL utilizzate per valutare l'istanza. Questo bucket ha il seguente formato: `inspector2-oval-prod-REGION`

Impostazione di pianificazioni personalizzate per Windows scansioni delle istanze

Puoi personalizzare l'intervallo tra le scansioni delle tue istanze Windows Amazon EC2 impostando un'espressione cron o un'espressione rate per l'InvokeInspectorSsmPlugin-do-not-delete associazione tramite SSM. Per ulteriori informazioni, consultate [Reference: Cron and rate expressions for Systems Manager](#) nella Guida per l'AWS Systems Manager utente o utilizzate le seguenti istruzioni.

Seleziona uno dei seguenti esempi di codice per modificare la cadenza di scansione per Windows le istanze da 6 ore predefinita a 12 ore utilizzando un'espressione rate o un'espressione cron.

Gli esempi seguenti richiedono l'utilizzo di AssociationId per l'associazione denominata.

InvokeInspectorSsmPlugin-do-not-delete È possibile recuperare il file AssociationId eseguendo il seguente AWS CLI comando:

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

Note

AssociationId è regionale, quindi devi prima recuperare un ID univoco per ciascuno. Regione AWS è quindi possibile eseguire il comando per modificare la cadenza di scansione in ciascuna regione in cui si desidera impostare una pianificazione di scansione personalizzata per Windows le istanze.

Example rate expression

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

```
--schedule-expression "rate(12 hours)"
```

Example cron expression

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

Scansione delle immagini dei container Amazon Elastic Container Registry con Amazon Inspector

[Amazon Inspector analizza le immagini dei container archiviate in Amazon Elastic Container Registry alla ricerca di vulnerabilità del software per generare risultati sulle vulnerabilità dei pacchetti.](#) Quando attivi la scansione Amazon ECR, imposti Amazon Inspector come servizio di scansione preferito per il tuo registro privato.

Note

Amazon ECR utilizza una politica di registro per concedere le autorizzazioni a un AWS principale. Questa entità dispone delle autorizzazioni necessarie per richiamare le API di Amazon Inspector per la scansione. Quando imposti l'ambito della politica del registro, non devi aggiungere l'`ecr:*`azione o inserirla. `PutRegistryScanningConfiguration deny` Ciò comporta errori a livello di registro durante l'attivazione e la disabilitazione della scansione per Amazon ECR.

Con la scansione di base, puoi configurare i tuoi repository per eseguire scansioni istantanee o eseguire scansioni manuali. Con la scansione avanzata, è possibile eseguire la scansione delle vulnerabilità del sistema operativo e dei pacchetti del linguaggio di programmazione a livello di registro. [Per un confronto fianco a fianco delle differenze tra scansione di base e avanzata, consulta le domande frequenti di Amazon Inspector.](#)

Note

La scansione di base viene fornita e fatturata tramite Amazon ECR. Per ulteriori informazioni, consulta i [prezzi di Amazon Elastic Container Registry](#). La scansione avanzata viene fornita

e fatturata tramite Amazon Inspector. Per ulteriori informazioni, consulta [Prezzi di Amazon Inspector](#).

Per informazioni su come attivare la scansione Amazon ECR, consulta [Attivazione di un tipo di scansione](#). Per informazioni su come visualizzare i risultati, consulta [Visualizzazione dei risultati di Amazon Inspector](#). Per informazioni su come visualizzare i risultati all'interno di Amazon ECR a livello di immagine, consulta [Image scanning](#) nella Amazon Elastic Container Registry User Guide. Puoi gestire i risultati utilizzando Servizi AWS non disponibili per la scansione di base, come [AWS Security Hub CSPM Amazon EventBridge](#).

Puoi visualizzare la configurazione di scansione per ogni repository in Amazon Inspector tramite pagine di copertura e API. Tuttavia, le impostazioni di configurazione per la scansione di base rispetto alla scansione continua possono essere modificate solo in Amazon ECR. Amazon Inspector offre visibilità su queste impostazioni ma non offre funzionalità di modifica diretta. Per ulteriori informazioni, consulta [Scansione delle immagini per individuare le vulnerabilità del software in Amazon ECR](#) nella Amazon ECR User Guide.

Questa sezione fornisce informazioni sulla scansione di Amazon ECR e descrive come configurare la scansione avanzata per i repository Amazon ECR.

Comportamenti di scansione per la scansione Amazon ECR

Quando attivi per la prima volta la scansione Amazon ECR, Amazon Inspector rileva le immagini inviate negli ultimi 14 giorni. Amazon Inspector esegue quindi la scansione delle immagini e imposta gli stati di scansione su. ACTIVE Amazon Inspector scansionerà solo le immagini attive in ECR (il `imageStatus` campo è). ACTIVE Le immagini con stato Archiviato in ECR (il `imageStatus` campo è ARCHIVED) non vengono scansionate da Amazon Inspector.

Se la scansione continua è abilitata, Amazon Inspector monitora le immagini a condizione che siano state inviate entro 14 giorni (per impostazione predefinita), la data dell'ultimo utilizzo sia entro 14 giorni (per impostazione predefinita) o che le immagini vengano scansionate entro la durata di nuova scansione configurata. Per gli account Amazon Inspector creati prima del 16 maggio 2025, la configurazione predefinita prevede una nuova scansione per monitorare le immagini se sono state inviate o recuperate negli ultimi 90 giorni. Per ulteriori informazioni, consulta [Configurazione della durata della nuova scansione di Amazon ECR](#).

Per la scansione continua, Amazon Inspector avvia nuove scansioni di vulnerabilità delle immagini dei container nelle seguenti situazioni:

- Ogni volta che viene inserita una nuova immagine del contenitore.
- Ogni volta che Amazon Inspector aggiunge un nuovo elemento CVE (Common Vulnerabilities and Exposures) al suo database e tale CVE è rilevante per l'immagine del contenitore (solo scansione continua).
- Ogni volta che l'immagine di un contenitore passa da archiviata a attiva in ECR.

Se configuri il repository per la scansione on push, le immagini vengono scansionate solo quando vengono inviate.

Puoi verificare l'ultima volta in cui è stata verificata la presenza di vulnerabilità in un'immagine del contenitore dalla scheda Immagini del contenitore nella pagina di gestione dell'account o utilizzando l'API. [ListCoverage](#) Amazon Inspector aggiorna il campo Last scanned at di un'immagine Amazon ECR in risposta ai seguenti eventi:

- Quando Amazon Inspector completa una scansione iniziale dell'immagine di un contenitore.
- Quando Amazon Inspector esegue nuovamente la scansione di un'immagine del contenitore, è stato aggiunto al database Amazon Inspector un nuovo elemento CVE (Common Vulnerabilities and Exposures) che influisce sull'immagine del contenitore.

Immagini archiviate del contenitore ECR

Amazon Inspector non esegue la scansione delle immagini dei container archiviate in ECR (is). `imageStatus ARCHIVED` Quando un'immagine attiva in ECR viene trasferita in archivio, Amazon Inspector chiude automaticamente i risultati e quindi li elimina dopo 3 giorni. Se un'immagine del contenitore archiviata viene trasformata in attiva in ECR, Amazon Inspector avvia una nuova scansione.

Mappatura delle immagini dei container ai container in esecuzione

Amazon Inspector offre una gestione completa della sicurezza dei container mappando le immagini dei container ai container in esecuzione su Amazon Elastic Container Service (Amazon ECS) e Amazon Elastic Kubernetes Service (Amazon EKS). Queste mappature forniscono informazioni sulle vulnerabilità delle immagini sui container in esecuzione.

Note

La policy gestita `AWSReadOnlyAccess` da sola non fornisce autorizzazioni sufficienti per visualizzare la mappatura tra le immagini di Amazon ECR e i container in esecuzione. Sono necessarie sia le politiche `AWSInspector2ReadOnlyAccess` gestite che quelle `AWSReadOnlyAccess` gestite per visualizzare le informazioni sulla mappatura delle immagini dei container.

È possibile dare priorità agli sforzi di riparazione in base ai rischi operativi e mantenere la copertura di sicurezza nell'intero ecosistema di container. Puoi visualizzare quante immagini di container sono attualmente in uso e quali immagini di container sono state utilizzate l'ultima volta su un cluster Amazon ECS o Amazon EKS nelle ultime 24 ore. Puoi anche visualizzare quante attività Amazon ECS e i pod Amazon EKS vengono distribuiti. Queste informazioni sono disponibili nella console Amazon Inspector nella schermata dei dettagli per i risultati delle immagini del contenitore e con i `ecrImageLastInUseAt` filtri per il `ecrImageInUseCount` tipo di [FilterCriteria](#) dati. Per le nuove immagini o i nuovi account dei container, possono essere necessarie fino a 36 ore prima che i dati siano disponibili. Successivamente, questi dati vengono aggiornati una volta ogni 24 ore. Per ulteriori informazioni, consulta [Visualizzazione dei risultati di Amazon Inspector e Visualizzazione dei dettagli dei risultati di Amazon Inspector](#).

Note

Questi dati vengono inviati automaticamente ai risultati di Amazon ECR quando attivi la scansione Amazon ECR e configuri il tuo repository per la scansione continua. La scansione continua deve essere configurata a livello di repository Amazon ECR. Per ulteriori informazioni, consulta [Scansione avanzata](#) nella Guida per l'utente di Amazon Elastic Container Registry.

Puoi anche [scansionare nuovamente le immagini dei container](#) dai cluster in base alla data dell'ultimo utilizzo.

Questa funzionalità è supportata anche su Fargate con Amazon ECS e Amazon EKS.

Sistemi operativi e tipi di supporti supportati

Per informazioni sui sistemi operativi supportati, vedere [Sistemi operativi supportati: scansione Amazon ECR con Amazon Inspector](#).

Le scansioni di Amazon Inspector dei repository Amazon ECR coprono i seguenti tipi di supporti supportati:

Manifesto dell'immagine

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

Configurazione dell'immagine

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

Livelli di immagine

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

Note

Amazon Inspector non supporta il tipo di supporto per la "application/vnd.docker.distribution.manifest.list.v2+json" scansione dei repository Amazon ECR.

Configurazione della durata della nuova scansione di Amazon ECR

L'impostazione della durata della nuova scansione di Amazon ECR determina per quanto tempo Amazon Inspector monitora continuamente le immagini dei container nei repository. Puoi configurare la durata della nuova scansione per la data dell'immagine, la data dell'ultima estrazione e la last-in-use data di invio. Come procedura ottimale, configura la durata della nuova scansione per adattarla al meglio al tuo ambiente.


Se crei immagini spesso, scegli una durata di scansione più breve. Per le immagini utilizzate per lunghi periodi di tempo, scegliete una durata di scansione più lunga. La durata di scansione predefinita per i nuovi account, inclusi i nuovi account aggiunti a un'organizzazione, è di 14 giorni.

Amazon Inspector continuerà a monitorare e ripetere la scansione di un'immagine finché è stata utilizzata l'ultima volta in un cluster o inviata entro 14 giorni (per impostazione predefinita). Se un'immagine non è stata inviata o utilizzata l'ultima volta su un contenitore in esecuzione entro le date di push e ultimo utilizzo configurate, Amazon Inspector interrompe il monitoraggio. Se necessario, è possibile modificare l'impostazione per monitorare le immagini in base alla data dell'ultima estrazione anziché alla data di ultimo utilizzo. Quando Amazon Inspector interrompe il monitoraggio di un'immagine, imposta il codice di stato della scansione dell'immagine su inattivo e il codice motivo su scaduto. Amazon Inspector pianifica quindi la chiusura di tutti i risultati associati alle immagini.

Se aumenti la durata della data push, Amazon Inspector applica la modifica a tutte le immagini scansionate attivamente nei repository configurati per la scansione continua. Tuttavia, le immagini inattive rimangono inattive, anche se vengono inserite entro la nuova durata.

Quando configuri la durata della nuova scansione da un account amministratore delegato, Amazon Inspector applica l'impostazione a tutti gli account membri dell'organizzazione. Se l'account amministratore delegato non abilita la scansione Amazon ECR, non può visualizzare i cluster per un'immagine API.

Per le immagini con più architetture, il tracciamento della last-in-use data non è supportato. Quando si utilizzano immagini con più architetture, si consiglia di configurare la scansione in base agli eventi pull o push dell'immagine anziché alla last-in-use data per garantire un corretto comportamento di nuova scansione.

 Note

Tutte le impostazioni della durata della nuova scansione configurate prima del 16 maggio 2025 rimarranno invariate. È possibile continuare a utilizzare tutte le impostazioni predefinite precedentemente configurate.

Durata della nuova scansione dell'immagine

La durata della nuova scansione dell'immagine determina per quanto tempo Amazon Inspector monitorerà le immagini. La durata della nuova scansione dell'immagine include due modalità: Data di ultimo utilizzo (impostazione predefinita) o Data ultima estrazione. Scegli Data di ultimo utilizzo (impostazione predefinita) se desideri utilizzare la data di ultimo utilizzo dell'attività del cluster Amazon ECS/Amazon EKS. Scegli Ultima data di estrazione se desideri utilizzare la data dell'ultima estrazione delle tue immagini Amazon ECR per scansionare nuovamente le immagini. Le seguenti opzioni sono disponibili come durate di nuova scansione:

- 14 giorni (impostazione predefinita)
- 30 giorni
- 60 giorni
- 90 giorni
- 180 giorni

Durata della data di invio dell'immagine

La durata della data di invio dell'immagine determina per quanto tempo Amazon Inspector monitorerà continuamente le immagini dopo essere state inviate ai repository. Le seguenti opzioni sono disponibili come durate di nuova scansione:

- 14 giorni (impostazione predefinita)
- 30 giorni
- 60 giorni
- 90 giorni
- 180 giorni
- Durata

Per configurare la durata della nuova scansione di Amazon ECR

1. [Accedi utilizzando le tue credenziali, quindi apri la console `https://console.aws.amazon.com/inspector/AmazonInspector/v2/home`.](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Seleziona Regione AWS dove desideri configurare la durata della nuova scansione di Amazon ECR.
3. Dal pannello di navigazione, scegli Impostazioni generali, quindi scegli Impostazioni di scansione ECR.
4. In Durata della nuova scansione ECR, scegli la modalità di nuova scansione dell'immagine, quindi scegli la durata corrispondente.
5. In Data push dell'immagine, scegli la data di invio dell'immagine.
6. Scegli Save (Salva).

Comprensione degli stati dell'immagine del contenitore ECR

Inspector esegue la scansione solo delle immagini contenute nelle ACTIVE immagini dei contenitori ECR. Le immagini dei contenitori ECR con uno ARCHIVED stato non vengono scansionate. Per ulteriori informazioni sui comportamenti di scansione, consulta [Comportamenti di scansione per la scansione Amazon ECR](#)

Quando lo stato dell'immagine di un'immagine contenitore ECR in ECR passa a, ACTIVE Inspector utilizza il `lastActivatedAt` campo per monitorare la durata della nuova scansione.

Scansione AWS Lambda funzioni con Amazon Inspector

Il supporto di Amazon Inspector per AWS Lambda funzioni e livelli fornisce valutazioni automatiche e continue delle vulnerabilità di sicurezza. Amazon Inspector offre due tipi di scansione della funzione Lambda:

[Scansione standard Amazon Inspector Lambda](#)

Questo tipo di scansione è il tipo di scansione Lambda predefinito. [Analizza le dipendenze delle applicazioni nelle funzioni e nei livelli Lambda per individuare le vulnerabilità dei pacchetti.](#)

[Scansione del codice Amazon Inspector Lambda](#)

[Questo tipo di scansione analizza il codice dell'applicazione personalizzato nelle funzioni e nei livelli Lambda alla ricerca di vulnerabilità del codice.](#) È possibile attivare la scansione standard Lambda o la scansione standard Lambda con la scansione del codice Lambda.

Se desideri attivare la scansione del codice Lambda, devi prima attivare la scansione standard Lambda. Per ulteriori informazioni, consulta [Attivazione di un](#) tipo di scansione.

Quando attivi la scansione della funzione Lambda, Amazon Inspector crea i seguenti canali collegati ai servizi nel tuo account: e. `cloudtrail:CreateServiceLinkedChannel` `cloudtrail>DeleteServiceLinkedChannel` Amazon Inspector gestisce questi canali e li utilizza per monitorare CloudTrail gli eventi per le scansioni. I canali ti consentono di visualizzare CloudTrail gli eventi nel tuo account come se avessi una traccia. CloudTrail Ti consigliamo di creare un percorso personalizzato CloudTrail per gestire gli eventi nel tuo account. Per informazioni su come visualizzare questi canali, consulta [Visualizzazione dei canali collegati ai servizi nella Guida per l'AWS CloudTrail utente.](#)

Note

Amazon Inspector non supporta la scansione delle [funzioni Lambda crittografate con](#) chiavi gestite dal cliente. Questo vale per la scansione standard Lambda e la scansione del codice Lambda.

Comportamenti di scansione per la scansione della funzione Lambda

Al momento dell'attivazione, Amazon Inspector analizza tutte le funzioni Lambda richiamate o aggiornate negli ultimi 90 giorni nel tuo account. Amazon Inspector avvia scansioni di vulnerabilità delle funzioni Lambda nelle seguenti situazioni:

- Non appena Amazon Inspector rileva una funzione Lambda esistente.
- Quando si distribuisce una nuova funzione Lambda nel servizio Lambda.
- Quando si implementa un aggiornamento al codice dell'applicazione o alle dipendenze di una funzione Lambda esistente o dei relativi livelli.
- Ogni volta che Amazon Inspector aggiunge un nuovo elemento di vulnerabilità ed esposizioni comuni (common vulnerabilities and exposures, CVE) al suo database e tale CVE è pertinente alla funzione.

Amazon Inspector monitora ogni funzione Lambda per tutta la sua durata fino a quando non viene eliminata o esclusa dalla scansione.

Puoi verificare quando una funzione Lambda è stata verificata l'ultima volta per verificare la presenza di vulnerabilità dalla scheda Funzioni Lambda nella pagina Gestione dell'account o utilizzando l'API.

[ListCoverage](#) Amazon Inspector aggiorna il campo Last scanned at per una funzione Lambda in risposta ai seguenti eventi:

- Quando Amazon Inspector completa una scansione iniziale di una funzione Lambda.
- Quando viene aggiornata una funzione Lambda.
- Quando Amazon Inspector esegue nuovamente la scansione di una funzione Lambda perché un nuovo elemento CVE che influisce su tale funzione è stato aggiunto al database Amazon Inspector.

Runtime e funzioni idonee supportati

Amazon Inspector supporta diversi runtime per la scansione standard Lambda e la scansione del codice Lambda. Per un elenco dei runtime supportati per ogni tipo di scansione, consulta e. [Runtime supportati: scansione standard di Amazon Inspector Lambda](#) [Runtime supportati: scansione del codice Amazon Inspector Lambda](#)

Oltre a disporre di un runtime supportato, una funzione Lambda deve soddisfare i seguenti criteri per essere idonea alle scansioni di Amazon Inspector:

- La funzione è stata richiamata o aggiornata negli ultimi 90 giorni.
- La funzione è contrassegnata LATEST.
- La funzione non è esclusa dalle scansioni per tag.

Note

Le funzioni Lambda che non sono state richiamate o modificate negli ultimi 90 giorni vengono automaticamente escluse dalle scansioni. Amazon Inspector riprenderà la scansione di una funzione esclusa automaticamente se viene richiamata nuovamente o se vengono apportate modifiche al codice della funzione Lambda.

Scansione standard Amazon Inspector Lambda

La scansione standard di Amazon Inspector Lambda identifica le vulnerabilità del software nelle dipendenze dei pacchetti applicativi che aggiungi al codice e ai livelli della funzione Lambda. Ad esempio, se la funzione Lambda utilizza una versione del `python-jwt` pacchetto con una vulnerabilità nota, la scansione standard Lambda genererà un risultato per quella funzione.

Se Amazon Inspector rileva una vulnerabilità nelle dipendenze dei pacchetti applicativi della funzione Lambda, Amazon Inspector fornisce una ricerca dettagliata del tipo di vulnerabilità del pacchetto.

Per istruzioni sull'attivazione di un tipo di scansione, consulta [Attivazione di un tipo di scansione](#)

Note

La scansione standard Lambda non analizza la dipendenza AWS SDK installata per impostazione predefinita nell'ambiente di runtime Lambda. Amazon Inspector analizza solo le dipendenze caricate con il codice della funzione o ereditate da un livello.

Note

La disattivazione della scansione standard di Amazon Inspector Lambda disattiverà anche la scansione del codice Amazon Inspector Lambda.

Esclusione delle funzioni dalla scansione standard Lambda

Puoi aggiungere tag alle funzioni Lambda, in modo da escluderle dalle scansioni standard di Amazon Inspector Lambda. L'esclusione di funzioni dalle scansioni può impedire avvisi non utilizzabili.

Quando si contrassegna una funzione per l'esclusione, il tag deve avere la seguente coppia chiave-valore.

- Chiave: `InspectorExclusion`
- Valore: `LambdaStandardScanning`

Questo argomento descrive come etichettare una funzione per l'esclusione dalle scansioni. Per ulteriori informazioni sull'aggiunta di tag in Lambda, consulta [Uso dei tag nelle funzioni Lambda](#).

Per escludere una funzione dalle scansioni

1. Accedi utilizzando le tue credenziali, quindi apri la console Lambda all'indirizzo. <https://console.aws.amazon.com/lambda/>
2. Dal pannello di navigazione, scegli Funzioni.
3. Scegli il nome della funzione che desideri escludere dalle scansioni standard di Amazon Inspector Lambda.
4. Scegli Configuration (Configurazione), quindi Tags (Tag).
5. Scegli Gestisci tag, quindi Aggiungi nuovo tag.
 - a. In Chiave, inserire `InspectorExclusion`.
 - b. In Valore, inserire `LambdaStandardScanning`.
6. Seleziona Salva.

Scansione del codice Amazon Inspector Lambda

Important

Questa funzionalità acquisisce frammenti di funzioni Lambda per evidenziare le vulnerabilità rilevate. Questi frammenti possono mostrare credenziali codificate e altri materiali sensibili.

Con questa funzionalità, Amazon Inspector analizza il codice dell'applicazione in una funzione Lambda alla ricerca di vulnerabilità del codice in base alle migliori pratiche di AWS sicurezza per rilevare fughe di dati, difetti di iniezione, crittografia mancante e crittografia debole. Amazon Inspector utilizza il ragionamento automatico e l'apprendimento automatico per valutare il codice applicativo della funzione Lambda. Utilizza inoltre rilevatori interni sviluppati in collaborazione con Amazon Q per identificare le violazioni e le vulnerabilità delle policy.

Amazon Inspector genera una [vulnerabilità del codice quando rileva una vulnerabilità](#) nel codice dell'applicazione della funzione Lambda. Questo tipo di ricerca include un frammento di codice che mostra il problema e dove è possibile trovarlo nel codice. Suggerisce inoltre come risolvere il problema. Il suggerimento include blocchi di plug-and-play codice che è possibile utilizzare per sostituire righe di codice vulnerabili. Queste correzioni al codice vengono fornite in aggiunta alle indicazioni generali sulla correzione del codice per questo tipo di risultato.

I suggerimenti per la correzione del codice si basano sul ragionamento automatico. Alcuni suggerimenti per la correzione del codice potrebbero non funzionare come previsto. Sei responsabile dei suggerimenti per la correzione del codice che adotti. Esamina sempre i suggerimenti per la correzione del codice prima di adottarli. Potrebbe essere necessario modificarli per assicurarti che il codice funzioni come previsto. Per ulteriori informazioni, consulta la [Politica sull'IA responsabile](#).

Se desideri attivare la scansione del codice Lambda, devi prima attivare la scansione standard Lambda. Per ulteriori informazioni, consulta [Attivazione di un](#) tipo di scansione. Per informazioni su quali dispositivi Regioni AWS supportano questa funzionalità, vedere [Region-specific disponibilità delle funzionalità](#).

Crittografia del codice nei risultati delle vulnerabilità del codice

Amazon Q archivia i frammenti di codice rilevati in relazione a una vulnerabilità del codice rilevata mediante la scansione del codice Lambda. Per impostazione predefinita, Amazon Q controlla [la chiave AWS di proprietà](#) utilizzata per crittografare il codice. Tuttavia, puoi utilizzare la tua chiave gestita dal cliente per la crittografia tramite l'API Amazon Inspector. Per ulteriori informazioni, consulta [Crittografia inattiva per il codice contenuto nei tuoi risultati](#).

Esclusione delle funzioni dalla scansione del codice Lambda

Puoi aggiungere tag alle funzioni Lambda, in modo da escluderle dalle scansioni del codice di Amazon Inspector Lambda. L'esclusione di funzioni dalle scansioni può impedire avvisi non utilizzabili. Quando si contrassegna una funzione per l'esclusione, il tag deve avere la seguente coppia chiave-valore.

- Chiave - InspectorCodeExclusion
- Valore: LambdaCodeScanning

Questo argomento descrive come etichettare una funzione per l'esclusione dalle scansioni del codice. Per ulteriori informazioni sull'aggiunta di tag in Lambda, consulta [Uso dei tag nelle funzioni Lambda](#).

Per escludere una funzione dalle scansioni del codice

1. Accedi utilizzando le tue credenziali, quindi apri la console Lambda all'indirizzo. <https://console.aws.amazon.com/lambda/>
2. Dal pannello di navigazione, scegli Funzioni.
3. Scegli il nome della funzione che desideri escludere dalle scansioni del codice Amazon Inspector Lambda.

4. Scegli Configuration (Configurazione), quindi Tags (Tag).
5. Scegli Gestisci tag, quindi Aggiungi nuovo tag.
 - a. In Chiave, inserire `InspectorCodeExclusion`.
 - b. In Valore, inserire `LambdaCodeScanning`.
6. Seleziona Salva.

Disattivazione di un tipo di scansione in Amazon Inspector

Quando disattivi un tipo di scansione, perdi l'accesso a tutti i risultati prodotti dal tipo di scansione. Se [riattivi il tipo di scansione](#), Amazon Inspector analizza tutte le risorse idonee per generare nuove scoperte. Se desideri tenere un registro dei risultati, puoi esportarli in un bucket Amazon Simple Storage Service (Amazon S3) come rapporto sui risultati. Per ulteriori informazioni, consulta [Esportazione dei report dei risultati di Amazon Inspector](#). Quando disattivi un tipo di scansione, potresti riscontrare le seguenti modifiche nell' AWS account in cui hai disattivato il tipo di scansione:

[Scansione Amazon EC2](#)

Quando disattivi la scansione di Amazon Inspector Amazon EC2 per un account, vengono eliminate le seguenti associazioni SSM:

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

Inoltre, il plug-in Amazon Inspector SSM viene rimosso da tutti gli host. Windows Per ulteriori informazioni, consulta [Scansione dell'istanza EC2 Windows](#).

[Scansione Amazon ECR](#)

Quando disattivi la scansione Amazon ECR per un account, l'account del tipo di scansione Amazon ECR passa da Scansione avanzata con Amazon Inspector a scansione di base con Amazon ECR.

[Scansione standard Lambda](#)

Quando si disattiva la scansione standard Lambda per un account, si disattiva la scansione del codice Lambda se il tipo di scansione è stato attivato. Inoltre, elimini il canale CloudTrail collegato al servizio creato da Amazon Inspector quando attivi la scansione standard Lambda.

[Codice di sicurezza di Amazon Inspector](#)

Quando disattivi Code Security per il tuo account, elimini tutte le integrazioni, i progetti e le configurazioni di scansione ad esso associate. Se il tuo account è l'amministratore delegato di un'organizzazione, disattivi solo Code Security per il tuo account e gli account dei membri diventano account autonomi.

Disattivazione delle scansioni

La disattivazione di tutti i tipi di scansione per un account disattiva Amazon Inspector per quell'account. Regione AWS Per ulteriori informazioni, consulta [Disattivazione di Amazon Inspector](#).

Per completare questa procedura per un ambiente con più account, segui questi passaggi dopo aver effettuato l'accesso come amministratore delegato di Amazon Inspector.

Console

Per disattivare le scansioni

1. Accedi utilizzando le tue credenziali, quindi apri la console Amazon Inspector all'indirizzo. <https://console.aws.amazon.com/inspector/v2/home>
2. Utilizzando il Regione AWS selettore nell'angolo superiore destro della pagina, seleziona la regione in cui desideri disattivare le scansioni.
3. Nel riquadro di navigazione, scegli Gestione account.
4. Scegli la scheda Account per mostrare lo stato di scansione di un account.
5. Seleziona la casella di controllo di ogni account per il quale desideri disattivare le scansioni.
6. Scegli Azioni e, tra le opzioni Disattiva, seleziona il tipo di scansione che desideri disattivare.
7. (Consigliato) Ripeti questi passaggi Regione AWS per ognuno dei quali desideri disattivare quel tipo di scansione.

API

Esegui l'operazione [Disable](#) API. Nella richiesta, fornisci gli ID dell'account per cui stai disattivando le scansioni e per `resourceTypes` fornire uno o più di `EC2`, `ECRLAMBDA`, o `LAMBDA_CODE` per disattivare le scansioni.

Center for Internet Security (CIS) esegue scansioni per i sistemi operativi delle istanze Amazon EC2

Le scansioni CIS di Amazon Inspector (scansioni CIS) eseguono benchmark dei sistemi operativi delle istanze Amazon EC2 per assicurarsi che siano configurati in base alle raccomandazioni delle migliori pratiche stabilite dal Center for Internet Security. [CIS Security Benchmarks](#) fornisce linee guida di configurazione standard del settore e best practice per configurare un sistema in modo sicuro. Puoi eseguire o pianificare scansioni CIS dopo aver abilitato la scansione Amazon Inspector EC2 per un account. Per informazioni su come attivare la scansione di Amazon EC2, consulta [Attivazione di un tipo di scansione](#).

Note

Gli standard CIS sono destinati ai sistemi operativi x86_64. Alcuni controlli potrebbero non essere valutati o restituire istruzioni di riparazione non valide su risorse basate su ARM.

Amazon Inspector esegue scansioni CIS su istanze Amazon EC2 di destinazione in base ai tag delle istanze e alla pianificazione di scansione definita. Amazon Inspector esegue una serie di controlli delle istanze su ciascuna istanza di destinazione. Ogni controllo valuta se la configurazione del sistema soddisfa le raccomandazioni specifiche di CIS Benchmark. Ogni controllo ha un ID e un titolo di controllo CIS, che corrispondono a una raccomandazione CIS Benchmark per quella piattaforma. Al termine di una scansione CIS, è possibile visualizzare i risultati per vedere quali controlli di istanza sono stati superati, ignorati o non riusciti per quel sistema.

Note

Per eseguire o pianificare le scansioni CIS, è necessario disporre di una connessione Internet sicura. Tuttavia, se desideri eseguire scansioni CIS su istanze private, devi utilizzare un endpoint VPC.

Argomenti

- [Requisiti delle istanze Amazon EC2 per le scansioni CIS di Amazon Inspector](#)
- [Esecuzione di scansioni CIS](#)

- [Considerazioni sulla gestione delle scansioni CIS di Amazon Inspector con AWS Organizations](#)
- [Bucket Amazon S3 di proprietà di Amazon Inspector utilizzati per le scansioni CIS di Amazon Inspector](#)
- [Creazione di una configurazione di scansione CIS](#)
- [Visualizzazione dei risultati della scansione CIS](#)
- [Modifica di una configurazione di scansione CIS](#)
- [Scaricamento dei risultati di una scansione CIS](#)

Requisiti delle istanze Amazon EC2 per le scansioni CIS di Amazon Inspector

Per eseguire una scansione CIS sulla tua istanza Amazon EC2, l'istanza Amazon EC2 deve soddisfare i seguenti criteri:

- Il sistema operativo dell'istanza è uno dei sistemi operativi supportati per le scansioni CIS. Per ulteriori informazioni, consulta [Sistemi operativi e linguaggi di programmazione supportati da Amazon Inspector](#).
- L'istanza è un'istanza di Amazon EC2 Systems Manager. Per ulteriori informazioni, consulta [Working with the SSM Agent nella Guida](#) per l'AWS Systems Manager utente.
- Il plug-in Amazon Inspector SSM è installato sull'istanza. Amazon Inspector installa automaticamente questo plug-in su istanze gestite.
- L'istanza ha un profilo di istanza che concede le autorizzazioni a SSM per gestire l'istanza e Amazon Inspector per eseguire scansioni CIS per quell'istanza. Per concedere queste autorizzazioni, collega le ManagedCisPolicy policy [Amazon SSManaged InstanceCore](#) and [AmazonInspector2](#) a un ruolo IAM. Quindi collega il ruolo IAM alla tua istanza come profilo di istanza. Per istruzioni sulla creazione e il collegamento di un profilo di istanza, consulta [Work with IAM roles](#) nella Amazon EC2 User Guide.

Note

Non è necessario abilitare l'ispezione approfondita di Amazon Inspector prima di eseguire una scansione CIS sulla tua istanza Amazon EC2. Se disabiliti l'ispezione approfondita di Amazon Inspector, Amazon Inspector installa automaticamente l'agente SSM, ma l'agente SSM non verrà più richiamato per eseguire l'ispezione approfondita. Tuttavia, di

conseguenza, l'`InspectorLinuxDistributor-do-not-delete` associazione è presente nel tuo account.

Requisiti degli endpoint di Amazon Virtual Private Cloud per l'esecuzione di scansioni CIS su istanze private di Amazon EC2

Puoi eseguire scansioni CIS su istanze Amazon EC2 su una rete Amazon. Tuttavia, se desideri eseguire scansioni CIS su istanze private di Amazon EC2, devi creare endpoint [Amazon VPC](#). I seguenti endpoint sono necessari per creare endpoint Amazon VPC per Systems Manager:

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

Per ulteriori informazioni, consulta [Creazione di endpoint Amazon VPC per Systems Manager](#) nella Guida per l'AWS Systems Manager utente.

Note

Attualmente, alcuni Regioni AWS non supportano l'`com.amazonaws.region.inspector2` endpoint.

Esecuzione di scansioni CIS

È possibile eseguire una scansione CIS una volta su richiesta o come scansione periodica pianificata. Per eseguire una scansione, è innanzitutto necessario creare una configurazione di scansione.

Quando si crea una configurazione di scansione, si specificano le coppie chiave-valore di tag da utilizzare per indirizzare le istanze. Se sei l'amministratore delegato di Amazon Inspector di un'organizzazione, puoi specificare più account nella configurazione di scansione e Amazon Inspector cercherà le istanze con i tag specificati in ciascuno di questi account. Scegli il livello CIS Benchmark per la scansione. Per ogni benchmark, CIS supporta un profilo di livello 1 e di livello 2 progettato per fornire linee di base per i diversi livelli di sicurezza richiesti da ambienti diversi.

- **Livello 1:** consiglia le impostazioni di sicurezza di base essenziali che possono essere configurate su qualsiasi sistema. L'implementazione di queste impostazioni dovrebbe causare interruzioni del servizio minime o nulle. L'obiettivo di queste raccomandazioni è ridurre il numero di punti di ingresso nei sistemi, riducendo i rischi complessivi per la sicurezza informatica.
- **Livello 2:** consiglia impostazioni di sicurezza più avanzate per ambienti ad alta sicurezza. L'implementazione di queste impostazioni richiede pianificazione e coordinamento per ridurre al minimo il rischio di impatto aziendale. L'obiettivo di queste raccomandazioni è aiutarti a raggiungere la conformità normativa.

Il livello 2 estende il livello 1. Quando scegli il livello 2, Amazon Inspector verifica tutte le configurazioni consigliate per il livello 1 e il livello 2.

Dopo aver definito i parametri per la scansione, puoi scegliere se eseguirla come scansione singola, che viene eseguita dopo aver completato la configurazione, o come scansione ricorrente. Le scansioni ricorrenti possono essere eseguite giornalmente, settimanalmente o mensilmente, in un momento a scelta.

Tip

Ti consigliamo di scegliere un giorno e un'ora che abbiano meno probabilità di influire sul sistema mentre la scansione è in esecuzione.

Considerazioni sulla gestione delle scansioni CIS di Amazon Inspector con AWS Organizations

Quando esegui scansioni CIS in un'organizzazione, gli amministratori delegati e gli account dei membri di Amazon Inspector interagiscono con le configurazioni e i risultati delle scansioni CIS in modo diverso.

In che modo gli amministratori delegati di Amazon Inspector possono interagire con le configurazioni e i risultati delle scansioni CIS

Quando l'amministratore delegato crea una configurazione di scansione, per tutti gli account o per uno specifico account membro, l'organizzazione è proprietaria della configurazione. Le configurazioni di scansione di proprietà di un'organizzazione hanno un ARN che specifica l'ID dell'organizzazione come proprietario:

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

L'amministratore delegato può gestire le configurazioni di scansione di proprietà di un'organizzazione, anche se sono state create da un altro account.

L'amministratore delegato può visualizzare i risultati della scansione per qualsiasi account della propria organizzazione.

Se l'amministratore delegato crea una configurazione di scansione e lo specifica SELF come account di destinazione, l'amministratore delegato è proprietario della configurazione di scansione, anche se lascia l'organizzazione. Tuttavia, l'amministratore delegato non può modificare la destinazione di una configurazione di scansione utilizzando come destinazione. SELF

Note

L'amministratore delegato non può aggiungere tag alle configurazioni di scansione CIS di proprietà dell'organizzazione.

In che modo gli account membri di Amazon Inspector possono interagire con le configurazioni e i risultati delle scansioni CIS

Quando un account membro crea una configurazione di scansione CIS, è proprietario della configurazione. Tuttavia, l'amministratore delegato può visualizzare la configurazione. Se un account membro lascia l'organizzazione, l'amministratore delegato non sarà in grado di visualizzare la configurazione.

Note

L'amministratore delegato non può modificare una configurazione di scansione creata dall'account membro.

Gli account membro, gli amministratori delegati che hanno SELF come destinazione e gli account autonomi possiedono tutti le configurazioni di scansione che creano. Queste configurazioni di scansione hanno un ARN che mostra l'ID dell'account come proprietario:

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

Un account membro può visualizzare i risultati delle scansioni nel proprio account, inclusi i risultati delle scansioni CIS pianificate dall'amministratore delegato.

Bucket Amazon S3 di proprietà di Amazon Inspector utilizzati per le scansioni CIS di Amazon Inspector

Open Vulnerability and Assessment Language (OVAL) è uno strumento di sicurezza delle informazioni che standardizza le modalità di valutazione e segnalazione dello stato delle macchine dei sistemi informatici. La tabella seguente elenca tutti i bucket Amazon S3 di proprietà di Amazon Inspector con definizioni OVAL utilizzati per le scansioni CIS. Amazon Inspector archivia i file di definizione OVAL necessari per le scansioni CIS. I bucket Amazon S3 di proprietà di Amazon Inspector dovrebbero essere inseriti nell'elenco consentito, se necessario. VPCs

Note

I dettagli per ciascuno dei seguenti bucket Amazon S3 di proprietà di Amazon Inspector non sono soggetti a modifiche. Tuttavia, la tabella potrebbe essere aggiornata in base alle nuove funzionalità supportate. Regioni AWS Non puoi utilizzare i bucket Amazon S3 di proprietà di Amazon Inspector per altre operazioni Amazon S3 o nei tuoi bucket Amazon S3.

Bucket CIS	Regione AWS
<code>cis-datasets-prod-arn-5908f6f</code>	Europa (Stoccolma)
<code>cis-datasets-prod-bah-8f88801</code>	Medio Oriente (Bahrein)
<code>cis-datasets-prod-bjs-0f40506</code>	Cina (Pechino)
<code>cis-datasets-prod-bom-435a167</code>	Asia Pacifico (Mumbai)
<code>cis-datasets-prod-cdg-f3a9c58</code>	Europa (Parigi)
<code>cis-datasets-prod-cgk-09eb12f</code>	Asia Pacifico (Giacarta)
<code>cis-datasets-prod-cmh-63030b9</code>	Stati Uniti orientali (Ohio)
<code>cis-datasets-prod-cpt-02c5c6f</code>	Africa (Città del Capo)

Bucket CIS	Regione AWS
<code>cis-datasets-prod-dub-984936f</code>	Europa (Irlanda)
<code>cis-datasets-prod-fra-6eb96eb</code>	Europa (Francoforte)
<code>cis-datasets-prod-gru-de69f99</code>	Sud America (San Paolo)
<code>cis-datasets-prod-hkg-8e30800</code>	Asia Pacifico (Hong Kong)
<code>cis-datasets-prod-iad-8438411</code>	Stati Uniti orientali (Virginia settentrionale)
<code>cis-datasets-prod-icn-f4eff1c</code>	Asia Pacifico (Seoul)
<code>cis-datasets-prod-kix-5743b21</code>	Asia Pacifico (Osaka)
<code>cis-datasets-prod-lhr-8b1fbd0</code>	Europa (Londra)
<code>cis-datasets-prod-mxp-7b1bbce</code>	Europa (Milano)
<code>cis-datasets-prod-nrt-464f684</code>	Asia Pacifico (Tokyo)
<code>cis-datasets-prod-osu-5bead6f</code>	AWS GovCloud (Stati Uniti orientali)
<code>cis-datasets-prod-pdt-adadf9c</code>	AWS GovCloud (Stati Uniti occidentali)
<code>cis-datasets-prod-pdx-acfb052</code>	Stati Uniti occidentali (Oregon)
<code>cis-datasets-prod-sfo-1515ba8</code>	Stati Uniti occidentali (California settentrionale)
<code>cis-datasets-prod-sin-309725b</code>	Asia Pacifico (Singapore)
<code>cis-datasets-prod-syd-f349107</code>	Asia Pacifico (Sydney)
<code>cis-datasets-prod-yul-5e0c95e</code>	Canada (Centrale)
<code>cis-datasets-prod-zhy-5a8each</code>	Cina (Ningxia)
<code>cis-datasets-prod-zrh-67e0e3d</code>	Europa (Zurigo)

Creazione di una configurazione di scansione CIS

Questo argomento descrive come creare una configurazione di scansione CIS.

Per eseguire una scansione CIS

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Utilizza il Regione AWS menu a discesa per selezionare il Regione AWS punto in cui desideri eseguire una scansione CIS.
3. Dal riquadro di navigazione, scegli Scansioni su richiesta, quindi scegli Scansioni CIS.
4. Scegli Crea nuova scansione.
5. Per Nome di configurazione della scansione, inserisci un nome di configurazione della scansione.
6. Per i tag delle risorse di Target, inserisci una chiave e un valore corrispondente per le istanze che desideri scansionare. Puoi specificare fino a cinque valori diversi per ogni chiave e un totale di 25 tag da includere nella scansione.
7. Per il livello CIS Benchmark, è possibile selezionare il Livello 1 per le configurazioni di sicurezza di base o il Livello 2 per le configurazioni di sicurezza avanzate.
8. Per gli account Target, specifica quali account includere nella scansione CIS. Per ulteriori informazioni, consulta [Considerazioni sulla gestione delle scansioni CIS di Amazon Inspector con AWS Organizations](#).

Se il tuo account è l'account amministratore delegato, puoi selezionare Tutti gli account o Specificare gli account. L'opzione Tutti gli account si rivolge a tutti gli account dell'organizzazione. L'opzione Specificare gli account si rivolge solo ai singoli account dell'organizzazione. Se si sceglie questa opzione, è possibile specificare più di un account separando i numeri di conto con una virgola. Puoi anche inserire SELF al posto di un ID account per creare una configurazione di scansione per il tuo account

Se il tuo account è un account autonomo o un account membro di un'organizzazione, puoi selezionare Self per creare una configurazione di scansione per il tuo account.

9. Per Pianificazione, scegli Scansione unica, che viene eseguita non appena hai finito di creare la configurazione di scansione, o Scansioni ricorrenti, che viene eseguita all'ora specificata.
10. Conferma le tue scelte, quindi scegli Crea.

Visualizzazione dei risultati della scansione CIS

Amazon Inspector crea un processo di scansione per ogni configurazione di scansione che viene eseguita e raccoglie i risultati di una scansione con un ID di scansione univoco. I risultati della scansione CIS sono disponibili per 90 giorni. È possibile visualizzare i risultati della scansione CIS tramite i relativi controlli o risorse scansionate:

- Risultati della scansione aggregati per controlli: raggruppa i risultati di una scansione per ogni singolo controllo eseguito durante la scansione. Per ogni controllo, viene visualizzato un rapporto sul numero di risorse non riuscite, ignorate o superate.
- Risultati della scansione aggregati per risorse analizzate: raggruppa i risultati di una scansione per ogni risorsa scansionata a cui è destinata la scansione durante la scansione. Per ogni risorsa, viene visualizzato un rapporto sul numero di controlli non riusciti, ignorati o superati da una risorsa.

Questo argomento descrive come visualizzare i risultati di una scansione CIS.

Per visualizzare i risultati della scansione

1. [Accedi utilizzando le tue credenziali, quindi apri la console \[https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home\]\(https://console.aws.amazon.com/inspector/AmazonInspector/v2/home\).](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Utilizza il Regione AWS menu a discesa per selezionare Regione AWS dove hai creato la configurazione di scansione CIS.
3. Dal pannello di navigazione, scegli Scansioni su richiesta, quindi scegli Scansioni CIS.
4. Scegli la scheda Risultati della scansione.
5. Nella colonna Pianificato per, scegli l'ID della pianificazione di scansione che desideri visualizzare. Oppure seleziona la riga con l'ID della pianificazione di scansione che desideri visualizzare, quindi scegli Visualizza dettagli.
6. Scegli Controlli per visualizzare ogni controllo eseguito o Risorse scansionate per visualizzare ogni risorsa scansionata presa di mira durante la scansione.

È inoltre possibile visualizzare i dettagli delle scansioni CIS pianificate.

Per visualizzare i dettagli delle scansioni CIS pianificate

1. [Accedi utilizzando le tue credenziali, quindi apri la console \[https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home\]\(https://console.aws.amazon.com/inspector/AmazonInspector/v2/home\).](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)

2. Utilizza il Regione AWS menu a discesa per selezionare Regione AWS dove hai creato la configurazione di scansione CIS.
3. Dal pannello di navigazione, scegli Scansioni su richiesta, quindi scegli Scansioni CIS.
4. Scegli la scheda Pianificato.
5. Nella colonna Nome della configurazione di scansione, scegli il nome della configurazione di scansione che desideri visualizzare. Oppure seleziona la riga con la configurazione di scansione che desideri visualizzare, quindi scegli Visualizza dettagli.

Modifica di una configurazione di scansione CIS

Questo argomento descrive come modificare una configurazione di scansione CIS.

Per modificare una configurazione di scansione CIS

1. [Accedi utilizzando le tue credenziali, quindi apri la console `https://console.aws.amazon.com/inspector/AmazonInspector/v2/home`.](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Utilizza il Regione AWS menu a discesa per selezionare Regione AWS dove hai creato la configurazione di scansione CIS.
3. Dal pannello di navigazione, scegli Scansioni su richiesta, quindi scegli Scansioni CIS.
4. Scegli la scheda Pianificato.
5. Seleziona la riga con la configurazione di scansione che desideri modificare, quindi scegli Modifica.

Scaricamento dei risultati di una scansione CIS

Puoi scaricare un PDF o CSV di una scansione CIS utilizzando la console o l'API di Amazon Inspector.

Note

Puoi scaricare solo un file CSV dei risultati delle scansioni CIS per le scansioni CIS raccolte dopo il 05/03/2024.

Questo argomento descrive come scaricare una scansione CIS utilizzando la console Amazon Inspector.

Per scaricare i risultati della scansione CIS dalla console

1. [Accedi utilizzando le tue credenziali, quindi apri la console \[https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home\]\(https://console.aws.amazon.com/inspector/AmazonInspector/v2/home\).](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Utilizza il Regione AWS menu a discesa per selezionare Regione AWS dove hai creato la configurazione di scansione CIS.
3. Dal pannello di navigazione, scegli Scansioni su richiesta, quindi scegli Scansioni CIS.
4. Scegli la scheda Risultati della scansione.
5. Nella colonna Pianificato da, scegli l'ID della pianificazione di scansione che desideri visualizzare. Oppure seleziona la riga con l'ID della pianificazione di scansione che desideri visualizzare, quindi scegli Visualizza dettagli.
6. Scegli Scarica, quindi scegli PDF o CSV. Se il tuo account è l'account amministratore delegato, puoi scegliere Seleziona account per scaricare i risultati per un account membro specifico.

Codice di sicurezza di Amazon Inspector

Amazon Inspector è un servizio di gestione delle vulnerabilità che rileva automaticamente i carichi di lavoro e li analizza continuamente per individuare vulnerabilità del software ed esposizione involontaria alla rete. Con Code Security, Amazon Inspector analizza il codice sorgente dell'applicazione proprietaria, le dipendenze delle applicazioni di terze parti e Infrastructure as Code alla ricerca di vulnerabilità. Puoi attivare Code Security nella console Amazon Inspector o con l'API Amazon Inspector. Una volta attivato Code Security, puoi creare e applicare una configurazione di scansione al tuo repository di codice per determinare con quale frequenza e quando verrà scansionato. È possibile visualizzare, modificare ed eliminare la configurazione di scansione in qualsiasi momento. Per informazioni sui paesi Regioni AWS in cui è disponibile Code Security, consulta [Regioni ed endpoint](#). Per informazioni sui prezzi, consulta i prezzi di [Amazon Inspector](#).

Prerequisiti per la sicurezza del codice

Prima di iniziare a utilizzare Code Security, è necessario attivare Code Security e decidere come crittografare i dati. Possono trattarsi di informazioni come credenziali di integrazione, codice o qualsiasi altra informazione relativa alle integrazioni, agli archivi di codice e ai progetti. [Per impostazione predefinita, i dati sono crittografati con una AWS chiave proprietaria](#). Ciò significa che la chiave viene creata, posseduta e gestita dal servizio. Se desideri possedere e gestire la chiave utilizzata per crittografare i tuoi dati, puoi creare una chiave [KMS gestita dal cliente](#).

Attivazione della sicurezza del codice

Code Security viene attivato nello stesso modo in cui si attivano tutti i tipi di scansione automatica. Per ulteriori informazioni, vedere [Attivazione di un tipo di scansione](#).

Creazione di una chiave di accesso gestita dal cliente AWS KMS

Per impostazione predefinita, i dati sono crittografati con una [chiave AWS proprietaria](#). Ciò significa che la chiave viene creata, posseduta e gestita dal servizio. Se desideri possedere e gestire la chiave utilizzata per crittografare i tuoi dati, puoi creare una chiave [KMS gestita dal cliente](#). Amazon Inspector non interagisce con i tuoi dati. Amazon Inspector acquisisce solo i metadati dai repository del tuo provider di codice sorgente. Per informazioni su come creare una chiave KMS gestita dal cliente, consulta [Creare una](#) chiave KMS nella Guida per l'utente AWS Key Management Service

Esempio di policy

Quando [crei la tua chiave gestita dai clienti](#), utilizza la seguente politica di esempio.

Note

Le [autorizzazioni FAS](#) nella seguente politica sono specifiche di Amazon Inspector, in quanto consentono ad Amazon Inspector di eseguire solo quelle chiamate API.

JSON

```
{
  "Version": "2012-10-17",
  "Id": "key-policy",
  "Statement": [
    {
      "Sid": "Allow Q to use Encrypt Decrypt GenerateDataKey and
GenerateDataKeyWithoutPlaintext",
      "Effect": "Allow",
      "Principal": {
        "Service": "q.amazonaws.com"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:GenerateDataKey",
        "kms:GenerateDataKeyWithoutPlaintext"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "StringLike": {
          "kms:EncryptionContext:aws:qdeveloper:codesecurity-scope": "111122223333"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:inspector2:us-east-1:111122223333:codesecurity-
integration/*"
        }
      }
    }
  ],
  {
```

```

    "Sid": "Allow Q to use DescribeKey",
    "Effect": "Allow",
    "Principal": {
      "Service": "q.amazonaws.com"
    },
    "Action": "kms:DescribeKey",
    "Resource": "*"
  },
  {
    "Sid": "Allow Inspector to use Encrypt Decrypt GenerateDataKey and
GenerateDataKeyWithoutPlaintext using FAS",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:role/inspectorCodeSecurity"
    },
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:GenerateDataKey",
      "kms:GenerateDataKeyWithoutPlaintext"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "inspector2.us-east-1.amazonaws.com"
      },
      "StringLike": {
        "kms:EncryptionContext:aws:qdeveloper:codesecurity-scope": "111122223333"
      }
    }
  },
  {
    "Sid": "Allow Inspector to use DescribeKey using FAS",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:role/inspectorCodeSecurity"
    },
    "Action": [
      "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "kms:ViaService": "inspector2.us-east-1.amazonaws.com"
      }
    }
  }
}

```

```
}  
  }  
    }  
  ]  
}
```

Dopo aver creato la tua chiave KMS, puoi utilizzare il seguente Amazon APIs Inspector.

- `UpdateEncryptionKey` — Da utilizzare con `CODE_REPOSITORY` for `resourceType` e `CODE` come tipo di scansione per configurare l'uso della chiave KMS gestita dal cliente.
- `GetEncryptionKey` — Da utilizzare con `CODE_REPOSITORY` for `resourceType` e `CODE` come tipo di scansione per configurare il recupero della configurazione della chiave KMS.
- `ResetEncryptionKey` — Da utilizzare con `CODE_REPOSITORY` per `resourceType` `CODE` ripristinare la configurazione della chiave KMS e per utilizzare una AWS chiave KMS di proprietà.

Creazione di un'integrazione tra Amazon Inspector, il tuo repository di codice

Questa sezione include argomenti che descrivono come creare un'integrazione tra Amazon Inspector e il tuo repository di codice. Quando crei un'integrazione, tutti gli archivi di codice vengono elencati come progetti nella console Amazon Inspector nella pagina Code Security. Altri argomenti di questa sezione descrivono come accedere alle integrazioni e ai progetti.

Code Security importa solo fino a 100.000 progetti e viene monitorato solo il ramo predefinito per ogni repository. Un progetto può essere associato a un massimo di tre configurazioni di scansione predefinite.

Code Security supporta solo un massimo di 100 integrazioni per account. Le integrazioni di Code Security non hanno alcun concetto di relazione tra account amministratore account/member delegato.

Per evitare restrizioni, consigliamo di non utilizzare lo stesso host per un'integrazione più di una volta.

Le integrazioni con GitHub SaaS e GitHub Enterprise Server richiedono GitHub Enterprise Cloud l'accesso pubblico a Internet.

⚠ Important

Le integrazioni di terze parti potrebbero essere disattivate temporaneamente o permanentemente senza preavviso per qualsiasi motivo, ad esempio per risolvere problemi di sicurezza.


Creazione di un'integrazione tra Amazon Inspector e GitHub

Questo argomento descrive come creare un'integrazione tra Amazon Inspector e GitHub


📘 Note

Se è la prima volta che crei un'integrazione, ti verrà richiesto di creare una configurazione di scansione predefinita nella Fase 2. Quando [crei una configurazione di scansione](#), scegli la frequenza di scansione, l'analisi di scansione e gli archivi da scansionare. La creazione di una configurazione di scansione predefinita equivale a creare una configurazione di scansione generale. Tuttavia, la configurazione di scansione predefinita viene associata automaticamente a tutti i progetti nuovi ed esistenti importati in Amazon Inspector. Se desideri creare una configurazione di scansione predefinita, scegli Continua con questa configurazione. È possibile creare una configurazione di scansione predefinita solo una volta. Se si crea una configurazione di scansione predefinita, non verrà richiesto di creare nuovamente una configurazione di scansione predefinita. È possibile creare una configurazione di scansione predefinita solo una volta per account e una volta per organizzazione. Se non desideri configurare una configurazione di scansione predefinita, scegli Ignora configurazione. Tuttavia, verrà richiesto di creare una configurazione di scansione predefinita alla successiva creazione di un'integrazione. Dopo aver creato una configurazione di scansione predefinita o aver saltato la creazione di una configurazione di scansione predefinita, verrai indirizzato alla Fase 3 del flusso di lavoro di integrazione in cui inserisci i dettagli dell'integrazione.

Le integrazioni con GitHub SaaS e GitHub Enterprise Server richiedono GitHub Enterprise Cloud l'accesso pubblico a Internet.

 Note

Amazon Inspector scansiona e monitora solo la filiale predefinita. Se crei un nuovo ramo predefinito, Amazon Inspector analizza e aggiorna il nuovo ramo predefinito.

 Important

Prima di completare la creazione dell'integrazione, ti viene richiesto di autorizzare la connessione tra Amazon Inspector GitHub e. È necessario completare questo passaggio per completare la procedura. Se chiudi il pop-up, non potrai procedere.

Per creare un'integrazione tra Amazon Inspector e GitHub

1. Accedi utilizzando le tue credenziali. [Apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Code Security. Scegli Connect a e scegli GitHub.
3. In Dettagli di integrazione, inserisci il nome dell'integrazione e scegli Connetti a GitHub.
4. Scegli Autorizza nel pop-up per creare una connessione tra Amazon Inspector e. GitHub
5. Nel banner di successo, scegli Vai alla pagina di creazione della GitHub connessione.
6. Immettete l'ID di installazione dell'GitHub applicazione. Se hai installato l'GitHub applicazione, puoi trovare l'ID GitHub di installazione nella pagina GitHub App o alla fine dell'URL dell'GitHub applicazione. Se non hai installato l'GitHub applicazione, scegli Installa una nuova app. Questo ti indirizza verso GitHub dove selezionare l'GitHub organizzazione e specificare l'ambito del repository.
7. Scegli Connect a GitHub.

Dopo aver creato l'integrazione, è possibile che Amazon Inspector non sia in grado di aggiornare il token di accesso. Ciò può verificarsi se l'host di integrazione non è disponibile o se Amazon Inspector riscontra altri problemi di comunicazione. Per risolvere il problema, puoi riautenticare la connessione dalla scheda Integrazioni nella pagina Code Security. Nella colonna Stato, l'integrazione viene visualizzata come Inattiva e Amazon Inspector offre la possibilità di effettuare nuovamente l'autenticazione. Scegli Riautentica. Verrai reindirizzato al flusso di lavoro di integrazione dove puoi completare la configurazione della connessione.

Se elimini le impostazioni di sistema per l'integrazione, puoi perdere la connessione a tempo indeterminato. In tal caso, è necessario [eliminare l'integrazione](#) e creare una nuova integrazione. Quando si elimina un'integrazione, si perdono tutti i progetti e le configurazioni di scansione associate all'integrazione.

Creazione di un'integrazione tra Amazon Inspector e GitLab Self Managed

Questo argomento descrive come creare un'integrazione tra Amazon Inspector e il tuo repository di codice in. GitLab Self Managed

Informazioni obbligatorie

Per creare una connessione è necessario quanto segue:

- Nome dell'integrazione: si tratta del nome aggiunto al corpo dell'integrazione.
- URL dell'endpoint: è l'URL utilizzato per accedere all'GitLab Self Managedistanza.
- Token di accesso personale: il token di accesso personale viene [creato GitLab Self Managed](#) da un account amministratore e deve includere i seguenti ambiti: `api`, `read_api`, e `read_repository`

Note

Amazon Inspector scansiona e monitora solo la filiale predefinita. Se crei un nuovo ramo predefinito, Amazon Inspector analizza e aggiorna il nuovo ramo predefinito.


Creazione di un'integrazione tra Amazon Inspector e GitLab Self Managed

La procedura seguente descrive come creare una connessione tra Amazon Inspector e il tuo repository di codice in. GitLab Self Managed

Note

Se è la prima volta che crei un'integrazione, ti verrà richiesto di creare una configurazione di scansione predefinita nella Fase 2. Quando si [crea una configurazione di scansione](#), si sceglie la frequenza di scansione, l'analisi di scansione e gli archivi da scansionare. La creazione di una configurazione di scansione predefinita equivale alla creazione di una configurazione di scansione generale. Tuttavia, la configurazione di scansione predefinita

viene associata automaticamente a tutti i progetti nuovi ed esistenti importati in Amazon Inspector. Se desideri creare una configurazione di scansione predefinita, scegli Continua con questa configurazione. È possibile creare una configurazione di scansione predefinita solo una volta. Se si crea una configurazione di scansione predefinita, non verrà richiesto di creare nuovamente una configurazione di scansione predefinita. È possibile creare una configurazione di scansione predefinita solo una volta per account e una volta per organizzazione. Se non desideri configurare una configurazione di scansione predefinita, scegli Ignora configurazione. Tuttavia, ti verrà richiesto di creare una configurazione di scansione predefinita la prossima volta che crei un'integrazione. Dopo aver creato una configurazione di scansione predefinita o aver saltato la creazione di una configurazione di scansione predefinita, verrai indirizzato alla Fase 3 del flusso di lavoro di integrazione in cui inserisci i dettagli dell'integrazione.

 Important

Prima di completare la creazione dell'integrazione, ti viene richiesto di autorizzare la connessione tra Amazon Inspector e Self Managed GitLab. È necessario completare questo passaggio per completare la procedura. Se chiudi il pop-up, non potrai procedere.

Per creare una connessione con GitLab Self Managed

1. Accedi utilizzando le tue credenziali. [Apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Code Security. Scegli Connect to e scegli GitLab Self Managed.
3. In Dettagli di integrazione, inserisci quanto segue:
 - a. Per Nome dell'integrazione, inserisci il nome aggiunto al corpo dell'integrazione.
 - b. Per Endpoint URL, inserisci l'URL utilizzato per accedere all'istanza GitLab autogestita.
 - c. Per Token di accesso personale, inserisci il tuo token di accesso personale con gli ambiti richiesti.
4. Scegli Connetti a. GitLab
5. Scegli Autorizza nella finestra pop-up per completare la creazione di una connessione tra Amazon Inspector e. GitLab

Dopo aver creato l'integrazione, è possibile che si verifichi uno scenario in cui Amazon Inspector non è in grado di aggiornare il token di accesso. Ciò può verificarsi se l'host di integrazione non è disponibile o se Amazon Inspector riscontra altri problemi di comunicazione. Per risolvere il problema, puoi riautenticare la connessione dalla scheda Integrazioni nella pagina Code Security. Nella colonna Stato, l'integrazione viene visualizzata come Inattiva e Amazon Inspector offre la possibilità di effettuare nuovamente l'autenticazione. Scegli Re-autenticate. Verrai reindirizzato al flusso di lavoro di integrazione dove puoi completare la configurazione della connessione.

Se elimini le impostazioni di sistema per l'integrazione, puoi perdere la connessione a tempo indeterminato. In tal caso, è necessario [eliminare l'integrazione](#) e creare una nuova integrazione. Quando si elimina un'integrazione, si perdono tutti i progetti e le configurazioni di scansione associate all'integrazione.

Visualizzazione delle integrazioni con gli archivi di codice

Questo argomento descrive come visualizzare le integrazioni nella console Amazon Inspector.

Per visualizzare le integrazioni nella console Amazon Inspector

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Integrations (Integrazioni). Da questa scheda, puoi esaminare tutte le integrazioni configurate e consultare le informazioni di base su tutte le integrazioni. Queste informazioni includono il nome dell'integrazione, lo stato dell'integrazione e il nome del provider del codice sorgente.

Effettua nuovamente l'autenticazione con il provider

Dopo aver creato l'integrazione, è possibile che si verifichi uno scenario in cui Amazon Inspector non è in grado di aggiornare il token di accesso. Ciò può verificarsi se l'host di integrazione non è disponibile o se Amazon Inspector riscontra altri problemi di comunicazione. Per risolvere il problema, puoi riautenticare la connessione dalla scheda Integrazioni nella pagina Code Security. Nella colonna Stato, l'integrazione viene visualizzata come Inattiva e Amazon Inspector offre la possibilità di effettuare nuovamente l'autenticazione. Scegli Re-autenticate. Verrai reindirizzato al flusso di lavoro di integrazione dove puoi completare la configurazione della connessione.

Se elimini le impostazioni di sistema per l'integrazione, puoi perdere la connessione a tempo indeterminato. In tal caso, è necessario [eliminare l'integrazione](#) e creare una nuova integrazione. Quando si elimina un'integrazione, si perdono tutti i progetti e le configurazioni di scansione associate all'integrazione.

Visualizzazione degli archivi di codice

L'argomento descrive come visualizzare gli archivi di codice nella console Amazon Inspector.

Per visualizzare gli archivi di codice nella console Amazon Inspector

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Code repositories. Da questa scheda, puoi esaminare tutti i tuoi repository di codice, elencati come progetti, e rivedere le informazioni di base su di essi. Queste informazioni includono il nome e lo stato di scansione di ogni progetto. Puoi anche rivedere le configurazioni associate ai tuoi progetti e la data dell'ultima scansione dei tuoi progetti. Puoi persino filtrare i tuoi progetti nella barra di ricerca.

Visualizzazione dei dettagli di un progetto

Questo argomento descrive come visualizzare i dettagli di un progetto nella console Amazon Inspector. Se il tuo account è l'amministratore delegato di un'organizzazione, puoi visualizzare i dettagli dei progetti che appartengono agli account dei membri.

Per visualizzare i progetti di codice nella console Amazon Inspector

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Code repositories. Da questa scheda, puoi esaminare tutti i tuoi repository di codice, elencati come progetti, e rivedere le informazioni di base su di essi. Queste informazioni includono il nome e lo stato di scansione di ogni progetto. Puoi anche rivedere le configurazioni associate ai tuoi progetti e la data dell'ultima scansione dei tuoi progetti. Puoi persino filtrare i tuoi progetti nella barra di ricerca.

4. Scegliere un progetto. Oppure seleziona un progetto e scegli Visualizza dettagli. Dalla schermata dei dettagli del progetto, puoi visualizzare le informazioni di base sul progetto. Queste informazioni includono il nome e l'ID del progetto, nonché l'ARN di integrazione. Include informazioni su quando il progetto è stato scansionato e sul tipo di fornitura. È anche possibile esaminare i risultati associati al progetto, [esportare i risultati](#) e [creare regole di soppressione per i risultati](#).

Eliminazione di un'integrazione

La procedura seguente descrive come eliminare un'integrazione nella console Amazon Inspector. Quando elimini un'integrazione, perdi tutti i progetti e le configurazioni di scansione associate all'integrazione.

Per eliminare un'integrazione nella console Amazon Inspector.

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home).
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Integrations (Integrazioni). Da questa scheda, puoi esaminare tutte le integrazioni configurate e consultare le informazioni di base su tutte le integrazioni. Queste informazioni includono il nome dell'integrazione, lo stato dell'integrazione e il tipo di provider di integrazione.
4. Seleziona un'integrazione e scegli Elimina.

Creazione di una configurazione di scansione

Prima di creare una configurazione di scansione, devi [creare un'integrazione con Amazon Inspector](#). La prima volta che crei un'integrazione, ti viene richiesto di creare una configurazione di scansione predefinita. Questo argomento descrive come creare una configurazione di scansione generale. La differenza tra una configurazione di scansione predefinita e una configurazione di scansione generale è che una configurazione di scansione predefinita viene associata automaticamente ai nuovi progetti. È possibile saltare la creazione di una configurazione di scansione predefinita.

Code Security supporta solo un massimo di 500 configurazioni di scansione generali. Code security supporta solo 1 configurazione di scansione predefinita per account e per organizzazione. Una configurazione di scansione può essere associata solo a un massimo di 100.000 progetti.

Un progetto può essere associato a un massimo di 4 configurazioni di scansione in totale. Ciò include una configurazione di scansione predefinita se è stata creata una configurazione di scansione predefinita. Le configurazioni di scansione per un'organizzazione non possono essere contrassegnate.

Se l'amministratore delegato di un'organizzazione crea una configurazione di scansione, la configurazione di scansione viene creata a livello di organizzazione e applicata a tutti gli account membri dell'organizzazione. Lo stesso accade se l'amministratore delegato crea una configurazione di scansione predefinita.

Quando si crea una configurazione di scansione, si sceglie la frequenza di scansione, l'analisi di scansione e gli archivi da scansionare. La frequenza di scansione può essere modificata in base a modifiche periodiche o personalizzate. La scansione periodica e basata sulle modifiche offre la possibilità di abilitare la scansione periodica. Se si abilita la scansione periodica, si imposta la frequenza di scansione sul giorno della settimana o del mese in cui viene eseguita la scansione. La scansione personalizzata offre la possibilità di abilitare la scansione quando il codice viene modificato e la scansione periodica. Se si abilita la scansione quando il codice viene modificato, si specifica il trigger di scansione da includere nelle richieste di unione e estrazione.

Le scansioni possono essere ignorate se un ID di commit non viene modificato entro un determinato periodo di tempo. Per la scansione periodica, le scansioni vengono ignorate se un ID di commit non è cambiato tra le scansioni in 1 settimana. Per le scansioni su richiesta, le scansioni vengono ignorate se un ID di commit non viene modificato tra le scansioni entro 24 ore.

Note

Se una configurazione di scansione prevede solo trigger per richieste di unione e richieste pull, vengono presentati solo i 25 risultati più critici o più importanti e solo nella piattaforma di gestione del codice sorgente. Nessuno sarà visibile in Amazon Inspector.

Per creare una configurazione di scansione generale

1. Accedi utilizzando le tue credenziali. [Apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/v2/home)
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Configurazioni, quindi scegli Crea configurazione di scansione.
4. In Dettagli di scansione, procedi come segue:

- Per Nome di configurazione, immettere un nome per la configurazione di scansione.
5. In Frequenza di scansione, specificate la frequenza di scansione del codice selezionando la scansione periodica e basata sulle modifiche o i tipi e i trigger di scansione personalizzati.
- a. (Opzione 1) Se scegli la scansione periodica e basata sulle modifiche, scegli Abilita scansione periodica o Disattiva scansione periodica.
- . Se scegli Abilita scansione periodica, imposta la frequenza di scansione scegliendo la settimana e il giorno in cui desideri che il codice venga scansionato.
- b. (Opzione 2) Se scegliete la scansione personalizzata, decidete se abilitare la scansione quando il codice viene modificato e la scansione periodica.
- i. Scegli Abilita la scansione quando il codice viene modificato o Disabilita la scansione quando il codice viene modificato. Se scegli Abilita la scansione quando il codice viene modificato, specifica quando vengono attivate le scansioni dal menu a discesa.
- ii. Scegli Abilita scansione periodica o Disabilita scansione periodica. Se scegli Abilita scansione periodica, imposta la frequenza di scansione scegliendo la settimana e il giorno in cui desideri che il codice venga scansionato. Puoi anche eseguire la scansione su trigger basati su eventi. Questi eventi includono quando una nuova pull request viene inizialmente aperta sul ramo predefinito e quando un commit viene unito o inviato al ramo predefinito. Le scansioni non vengono attivate in caso di aggiornamenti o revisioni successivi di una pull request esistente. Per attivare una nuova scansione, chiudi e riapri la pull request.
6. In Analisi di scansione, decidi se configurare un'analisi di scansione completa o un'analisi di scansione personalizzata:
- a. (Opzione 1) Se scegliete Analisi di scansione completa, applicate tutte le seguenti analisi di scansione:
- Test statici di sicurezza delle applicazioni: analizza il codice sorgente alla ricerca di vulnerabilità.
 - Scansione IaC: analizza gli script e il codice che configurano e forniscono l'infrastruttura.
 - Analisi statica della composizione del software: esamina i pacchetti open source nelle applicazioni.
- b. (Opzione 2) Se si sceglie Analisi di scansione personalizzata, è necessario scegliere almeno un tipo dei tipi di analisi di scansione menzionati in precedenza dal menu a discesa:

7. (Facoltativo) Per i tag, create una coppia chiave-valore da applicare al progetto. Puoi creare fino a 50 tag.
8. Scegli Next (Successivo).
9. In Selezione archivio, scegli Tutti gli archivi o Archivi specifici.
 - a. (Opzione 1) Se scegli Tutti gli archivi, la scansione è abilitata per tutti gli archivi esistenti.
 - b. (Opzione 2) Se si sceglie Archivi specifici, la scansione è abilitata solo per i repository specificati.
10. Scegli Next (Successivo).
11. Rivedi le tue scelte, quindi scegli Crea configurazione di scansione.

Note

Le configurazioni di scansione generali vengono applicate solo a tutti gli archivi di codice esistenti. Non verranno applicate ai nuovi repository di codice.

Visualizzazione delle configurazioni di scansione

La procedura seguente descrive come visualizzare le configurazioni di scansione nella console Amazon Inspector.

Note

Quando visualizzi la configurazione di scansione a livello di organizzazione, alcuni dettagli nella schermata Code Security differiranno in base alla tua. Account AWS

Per visualizzare i dettagli di una configurazione di scansione

1. [Accedi utilizzando le tue credenziali, quindi apri la console `https://console.aws.amazon.com/inspector/AmazonInspector/v2/home`.](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Configurazioni per visualizzare un elenco delle configurazioni di scansione. Se sei l'amministratore delegato, l'elenco include le configurazioni di scansione della tua

organizzazione. Puoi vedere il nome di ogni configurazione di scansione e chi ha creato ogni configurazione di scansione (Account AWS ID o ID dell'organizzazione). È inoltre possibile visualizzare i tipi di scansione e il tipo di analisi della scansione applicati alla configurazione. È anche possibile filtrare la configurazione di scansione in base a diversi campi nella barra di ricerca.

Visualizzazione dei dettagli per una configurazione di scansione

La procedura seguente descrive come visualizzare i dettagli di una configurazione di scansione nella console Amazon Inspector.

Per visualizzare i dettagli di una configurazione di scansione

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Configurazioni.
4. Scegli la configurazione di cui desideri visualizzare i dettagli. La schermata dei dettagli della configurazione della scansione fornisce una panoramica della configurazione della scansione. Da questa schermata, è possibile visualizzare la configurazione di scansione ARN, quali tipi di frequenza di scansione sono abilitati e quali tipi di analisi di scansione sono abilitati. È inoltre possibile [eliminare](#) la configurazione di scansione da questa schermata. Se stai visualizzando una configurazione di scansione che appartiene alla tua organizzazione, puoi [modificarla](#) anche da questa schermata.

Modifica di una configurazione di scansione

È possibile modificare una configurazione di scansione in qualsiasi momento. Quando si modifica una configurazione di scansione, è possibile modificare la frequenza di scansione, l'analisi di scansione, i tag e gli archivi da scansionare. Ad esempio, si modifica una configurazione di scansione per sospendere la scansione di un particolare repository. La procedura seguente descrive come modificare una configurazione di scansione.

Per modificare una configurazione di scansione

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)

2. Dal pannello di navigazione, scegli Code Security.
3. Scegli Configurazioni.
4. Seleziona la configurazione che desideri modificare, quindi scegli Modifica. Puoi anche scegliere la configurazione che desideri modificare e quindi scegliere Modifica.

Eliminazione di una configurazione di scansione

È possibile eliminare una configurazione di scansione in qualsiasi momento. Questo argomento descrive come eliminare una configurazione di scansione.

Per eliminare una configurazione di scansione

1. [Accedi utilizzando le tue credenziali, quindi apri la console \[https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home\]\(https://console.aws.amazon.com/inspector/AmazonInspector/v2/home\).](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Dal pannello di navigazione, scegli Code security.
3. Scegli Configurazioni.
4. Seleziona la configurazione che desideri eliminare, quindi scegli Elimina. Oppure scegli la configurazione che desideri eliminare, quindi scegli Elimina.

Esecuzione di una scansione su richiesta

Puoi eseguire un'operazione su richiesta per i tuoi progetti. Quando si esegue una scansione su richiesta, al progetto selezionato viene applicata un'unione di tutte le configurazioni di scansione configurate. Se il tuo account è l'account amministratore delegato di un'organizzazione, puoi eseguire una scansione su richiesta per i progetti che appartengono agli account dei membri. La procedura seguente descrive come eseguire una scansione su richiesta nella console Amazon Inspector.

Per eseguire una scansione su richiesta

1. [Accedi utilizzando le tue credenziali, quindi apri la console \[https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home\]\(https://console.aws.amazon.com/inspector/AmazonInspector/v2/home\).](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Dal pannello di navigazione, scegli Code security.
3. Scegli Code repositories.
4. Seleziona il progetto che desideri scansionare, quindi scegli Scansione su richiesta.

Lingue supportate per la sicurezza del codice Amazon Inspector

Questo argomento include le lingue supportate per Amazon Inspector Code Security.

Lingue supportate per SAST

- C#(sono consigliate tutte le versioni tranne la .Net 6.0 e successive)
- C(C11 o versioni precedenti)
- C++(C++17 o precedenti)
- Go(solo Go 1.18)
- Java(Java25 o versioni precedenti)
- JavaScript(EMCMA Script 2021 o precedenti)
- JSX(React 17 o versioni precedenti)
- Kotlin(Kotlin2.0 o precedente)
- PHP(PHP8.2 o precedente)
- Python(Python3.13 o precedente nella serie Python 3)
- Ruby(solo Ruby 2.7 e 3.2)
- Rust
- Scala(Scala3.2.2 o versioni precedenti)
- Shell
- TSX
- TypeScript (tutte le versioni)

Linguaggi supportati per l'analisi della composizione del software

- Go(solo Go 1.18)
- Java(Java25 o versioni precedenti)
- JavaScript(EMCMA Script 2021 o precedenti)
- PHP(PHP8.2 o versioni precedenti)
- Python(Python3.13 o precedente nella serie Python 3)
- .Net
- Ruby(solo Ruby 2.7 e 3.2)
- Rust

Linguaggi per l'infrastruttura come codice

- AWS CDK (PythonTypeScript)
- CloudFormation (2010-09-09)
- Terraform(1.6.2 o versioni precedenti)

Disattivazione di Code Security

Per ulteriori informazioni sulla disattivazione di Code Security, consulta [Disattivazione](#) di un tipo di scansione.

Comprendere i risultati di Amazon Inspector

Amazon Inspector genera un risultato quando rileva una vulnerabilità con una correzione o una correzione in sospeso nelle istanze Amazon EC2, nelle immagini dei contenitori Amazon ECR e nelle funzioni Lambda. Genera inoltre risultati per le vulnerabilità del codice rilevate nel codice sorgente delle applicazioni di prime parti, nelle dipendenze delle applicazioni di terze parti e nell'Infrastructure as Code. Un risultato è un rapporto dettagliato su una vulnerabilità che interessa una delle tue risorse. AWS

I risultati prendono il nome dalle vulnerabilità e forniscono valutazioni di gravità, informazioni sulle AWS risorse interessate e non, e dettagli che descrivono come correggere AWS le vulnerabilità rilevate. Amazon Inspector archivia tutti i risultati attivi fino a quando non li correggi.

Quando una risorsa viene eliminata, terminata o non è più idonea per la scansione, Amazon Inspector chiude automaticamente i risultati associati alla risorsa e quindi li elimina dopo 3 giorni. Se i risultati vengono chiusi per qualsiasi altro motivo, vengono eliminati dopo 30 giorni.

Note

Amazon Inspector riaprirà un problema risolto entro sette giorni dalla sua chiusura se il problema che ha causato la vulnerabilità si ripresenta.

Se disabiliti Amazon Inspector, i risultati vengono rimossi dopo 24 ore. Se una risorsa viene interrotta, qualsiasi risultato relativo alla risorsa viene rimosso dopo 3 giorni. Lo stesso vale per qualsiasi risultato associato a una risorsa per cui la scansione non è più idonea. Se AWS sospende il tuo account, i risultati vengono rimossi dopo 90 giorni. I risultati relativi alle istanze interrotte rimangono attivi.

I risultati affermano

Amazon Inspector classifica i risultati nei seguenti stati.

Attivo

Amazon Inspector classifica come Attivo un risultato che non è stato corretto.

Soppresso

Amazon Inspector classifica un risultato soggetto a una o più regole di [soppressione](#) come Soppresso.

Closed

Una volta risolto un problema, Amazon Inspector lo classifica come Chiuso.

Argomenti

- [Tipi di ricerca di Amazon Inspector](#)
- [Visualizzazione dei risultati di Amazon Inspector](#)
- [Visualizzazione dei dettagli relativi ai risultati di Amazon Inspector](#)
- [Visualizzazione del punteggio di Amazon Inspector e comprensione dei dettagli di vulnerability intelligence](#)
- [Comprensione dei livelli di gravità dei risultati di Amazon Inspector](#)

Tipi di ricerca di Amazon Inspector

Questa sezione descrive i diversi tipi di ricerca in Amazon Inspector.

Argomenti

- [Vulnerabilità del pacchetto](#)
- [Vulnerabilità del codice](#)
- [Raggiungibilità della rete](#)

Vulnerabilità del pacchetto

I risultati delle vulnerabilità dei pacchetti identificano i pacchetti software presenti AWS nell'ambiente che sono esposti a vulnerabilità ed esposizioni comuni (CWE). Gli aggressori possono sfruttare queste vulnerabilità prive di patch per compromettere la riservatezza, l'integrità o la disponibilità dei dati o per accedere ad altri sistemi. Il sistema CVE è un metodo di riferimento per vulnerabilità ed esposizioni alla sicurezza delle informazioni note pubblicamente. [Per ulteriori informazioni, vedere <https://www.cve.org/>](#).

Amazon Inspector può generare rilevamenti di vulnerabilità dei pacchetti per istanze EC2, immagini di contenitori ECR e funzioni Lambda. I risultati delle vulnerabilità dei pacchetti includono dettagli

esclusivi di questo tipo di rilevamento. Questi dettagli sono il [punteggio di Inspector e l'intelligence sulla vulnerabilità](#).

Per le istanze Windows EC2, i risultati delle vulnerabilità dei pacchetti possono essere identificati dalla Microsoft Knowledge Base (KB) IDs anziché individualmente. CVEs Se un aggiornamento KB riguarda uno o più risultati CVEs, Amazon Inspector riporta un singolo risultato KB, ad esempio KB5023697, anziché un risultato separato per ogni CVE. Una ricerca KB specifica il punteggio CVSS, il punteggio EPSS e la disponibilità di exploit più alti tra tutti i componenti. CVEs

Vulnerabilità del codice

I risultati delle vulnerabilità del codice aiutano a identificare le righe di codice che possono essere sfruttate. Le vulnerabilità del codice includono la crittografia mancante, le fughe di dati, i difetti di iniezione e la crittografia debole. [Amazon Inspector genera risultati di vulnerabilità del codice tramite la scansione della funzione Lambda e la funzionalità Code Security](#).

Amazon Inspector valuta il codice applicativo della funzione Lambda utilizzando il ragionamento automatico e l'apprendimento automatico per analizzare il codice dell'applicazione per la conformità generale alla sicurezza. Identifica le violazioni delle policy e le vulnerabilità sulla base di rilevatori interni sviluppati in collaborazione con Amazon Q. Per un elenco di possibili rilevamenti, consulta [Amazon Q Detector Library](#).

La scansione del codice acquisisce frammenti di codice per evidenziare le vulnerabilità rilevate. Ad esempio, un frammento di codice potrebbe mostrare credenziali codificate o altri materiali sensibili in testo non crittografato. Amazon Q archivia frammenti di codice associati a vulnerabilità del codice. [Per impostazione predefinita, il codice è crittografato con una AWS chiave proprietaria](#). Tuttavia, puoi creare una chiave gestita dal cliente per crittografare il codice se desideri un maggiore controllo su queste informazioni. Per ulteriori informazioni, consulta [Crittografia inattiva per il codice contenuto nei tuoi risultati](#).

Note

L'amministratore delegato di un'organizzazione non può visualizzare frammenti di codice che appartengono agli account dei membri.

Raggiungibilità della rete

I risultati sulla raggiungibilità della rete indicano che nel tuo ambiente esistono percorsi di rete aperti verso le istanze Amazon EC2. Questi risultati appaiono quando le porte TCP e UDP sono raggiungibili dai bordi del VPC, come un gateway Internet (incluse le istanze di Application Load Balancer o Classic Load Balancer), una connessione peering VPC o una VPN tramite un gateway virtuale. Questi risultati evidenziano configurazioni di rete che potrebbero essere eccessivamente permissive, come gruppi di sicurezza mal gestiti, elenchi di controllo degli accessi o gateway Internet, o che potrebbero consentire accessi potenzialmente dannosi.

Amazon Inspector genera risultati sulla raggiungibilità della rete solo per le istanze Amazon EC2. Amazon Inspector esegue scansioni per rilevare la raggiungibilità della rete ogni 12 ore una volta abilitato Amazon Inspector.

Amazon Inspector valuta le seguenti configurazioni durante la scansione dei percorsi di rete:

- [Istanze Amazon EC2](#)
- [Application Load Balancer](#)
- [Direct Connect](#)
- [Elastic Load Balancer](#)
- [Interfacce di rete elastiche](#)
- [Internet Gateway](#)
- [Elenchi di controllo dell'accesso alla rete](#)
- [Tabelle di routing](#)
- [Gruppi di sicurezza](#)
- [Sottoreti](#)
- [Cloud privati virtuali](#)
- [Gateway privati virtuali](#)
- [Endpoint VPC](#)
- [Endpoint gateway VPC](#)
- [Connessioni in peering di VPC](#)
- [Connessioni VPN](#)

Visualizzazione dei risultati di Amazon Inspector

Puoi visualizzare i risultati nella console Amazon Inspector e con l'API Amazon [ListFindings](#) Inspector. Nella console Amazon Inspector, puoi visualizzare tutti i risultati nelle schermate Dashboard e Findings. Per impostazione predefinita, queste schermate mostrano solo i risultati attivi e critici. Tuttavia, puoi filtrare i risultati o scegliere di visualizzarli per categoria. Puoi anche visualizzare alcuni risultati in [Security Hub CSPM e Amazon ECR](#) se attivi queste integrazioni. Le procedure in questa sezione descrivono come visualizzare i risultati nella console Amazon Inspector e con l'API Amazon ListFindings Inspector.

Console

Per visualizzare i risultati di Amazon Inspector

1. Accedi utilizzando le tue credenziali. [Apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. (Facoltativo) Dal pannello di navigazione, scegli Dashboard. La dashboard mostra una panoramica della copertura per l'ambiente e solo i risultati attivi e critici.
3. (Facoltativo) Dal pannello di navigazione, scegli Findings. Questa schermata elenca tutti i risultati attivi. È possibile utilizzare i criteri di filtro [per visualizzare risultati specifici](#). Per escludere i risultati dall'elenco, [create una regola di soppressione](#). Per visualizzare i dettagli di un risultato, scegliete il nome del risultato.
4. (Facoltativo) Dal riquadro di navigazione, scegli una delle seguenti opzioni per visualizzare i risultati per categoria:
 - Per vulnerabilità: mostra le vulnerabilità con i risultati più critici.
 - Per account: mostra gli account con i risultati più critici. Questa categoria è disponibile solo per gli amministratori delegati.
 - Per esempio: mostra le istanze di Amazon EC2 con i risultati più critici. Questa categoria non include informazioni sulla disponibilità della rete.
 - Per immagine del contenitore: mostra le immagini dei container Amazon ECR con i risultati più critici. Questa categoria fornisce anche informazioni di base sulle immagini dei container. Include anche dettagli, come il numero di attività Amazon ECS e i pod Amazon EKS distribuiti. Da questa schermata, puoi scoprire quanti tasks/pods erano in esecuzione nelle ultime 24 ore e si sono fermati.
 - Per repository di container: mostra i repository di container con i risultati più critici.

- Per funzione Lambda: mostra le funzioni Lambda con i risultati più critici.

API

Per visualizzare i risultati di Amazon Inspector

- Esegui l'operazione [ListFindings](#) API. Nella richiesta, specifica [FilterCriteria](#) per restituire risultati specifici.

Visualizzazione dei dettagli relativi ai risultati di Amazon Inspector

La procedura in questa sezione descrive come visualizzare i dettagli dei risultati di Amazon Inspector.

Per visualizzare i dettagli di una scoperta

1. [Accedi utilizzando le tue credenziali, quindi apri la console Amazon Inspector su v2/home https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Seleziona la regione in cui visualizzare i risultati.
3. Nel riquadro di navigazione, scegli Risultati per visualizzare l'elenco dei risultati
4. (Facoltativo) Utilizzate la barra dei filtri per selezionare un risultato specifico. Per ulteriori informazioni, consulta [Filtraggio dei risultati di Amazon Inspector](#).
5. Scegliete un risultato per visualizzarne il pannello dei dettagli.

Il pannello dei dettagli del risultato contiene le caratteristiche identificative di base del risultato. Ciò include il titolo della scoperta, una descrizione di base della vulnerabilità identificata, suggerimenti per la correzione e un punteggio di gravità. Per informazioni sul punteggio, vedere. [Comprensione dei livelli di gravità dei risultati di Amazon Inspector](#)

I dettagli disponibili per un risultato variano a seconda del tipo di risultato e della risorsa interessata.

Tutti i risultati contengono il numero Account AWS identificativo per cui è stato identificato il risultato, la gravità, il tipo di risultato, la data in cui è stato creato il risultato e una sezione relativa alla risorsa interessata con i dettagli sulla risorsa.

Il tipo di risultato determina le informazioni sulla correzione e sulle vulnerabilità disponibili per il risultato. A seconda del tipo di risultato, sono disponibili diversi dettagli di ricerca.

Vulnerabilità del pacchetto

I risultati delle vulnerabilità dei pacchetti sono disponibili per le istanze EC2, le immagini dei contenitori ECR e le funzioni Lambda. Per ulteriori informazioni, consulta [Vulnerabilità del pacchetto](#).

I risultati delle vulnerabilità dei pacchetti includono [Visualizzazione del punteggio di Amazon Inspector e comprensione dei dettagli di vulnerability intelligence](#) anche.

Questo tipo di risultato contiene i seguenti dettagli:

- **Correzione disponibile:** indica se la vulnerabilità è stata corretta in una versione più recente dei pacchetti interessati. Ha uno dei seguenti valori:
 - YES, il che significa che tutti i pacchetti interessati hanno una versione fissa.
 - NO, il che significa che nessun pacchetto interessato ha una versione fissa.
 - PARTIAL, il che significa che uno o più (ma non tutti) dei pacchetti interessati hanno una versione fissa.
- **Exploit disponibile:** indica che la vulnerabilità ha un exploit noto.
 - YES, il che significa che la vulnerabilità rilevata nell'ambiente in uso presenta un exploit noto. Amazon Inspector non ha visibilità sull'uso degli exploit in un ambiente.
 - NO, il che significa che questa vulnerabilità non ha un exploit noto.
- **Pacchetti interessati:** elenca ogni pacchetto identificato come vulnerabile nella ricerca e i dettagli di ogni pacchetto:
- **Filepath:** l'ID del volume EBS e il numero di partizione associati a un risultato. Questo campo è presente nei risultati delle istanze EC2 scansionate utilizzando [Scansione senza agenti](#)
- **Versione installata/Versione fissa:** il numero di versione del pacchetto attualmente installato per il quale è stata rilevata una vulnerabilità. Confronta il numero di versione installata con il valore dopo la barra (/). Il secondo valore è il numero di versione del pacchetto che corregge la vulnerabilità rilevata, come fornito dal documento Common Vulnerabilities and Exposures (CVEs) o dall'avviso associato al risultato. Se la vulnerabilità è stata corretta in più versioni, questo campo elenca la versione più recente che include la correzione. Se non è disponibile una correzione, questo valore è `None available`.

Note

Se è stato rilevato un risultato prima che Amazon Inspector iniziasse a includere questo campo nei risultati, il valore per questo campo è vuoto. Tuttavia, potrebbe essere disponibile una correzione.

- **Package manager** — Il gestore di pacchetti utilizzato per configurare questo pacchetto.
- **Riparazione**: i comandi di riparazione variano a seconda della configurazione del sistema. Consulta i riferimenti alla ricerca di riferimenti o la documentazione del sistema operativo per indicazioni più specifiche. Quando disponibile, i clienti devono aggiornare i pacchetti interessati alla versione fissa fornita (o più recente).
- **Dettagli sulla vulnerabilità**: fornisce un collegamento alla fonte preferita di Amazon Inspector per il CVE identificato nella scoperta, come National Vulnerability Database (NVD), REDHAT o un altro fornitore del sistema operativo. Inoltre, troverai i punteggi di gravità della scoperta. Per ulteriori informazioni sui punteggi di gravità, ad esempio, vedere [Comprensione dei livelli di gravità dei risultati di Amazon Inspector](#). Sono inclusi i seguenti punteggi, inclusi i vettori di punteggio per ciascuno di essi:
 - [Punteggio Exploit Prediction Scoring System \(EPSS\)](#)
 - Punteggio Inspector
 - CVSS 3.1 di Amazon CVE
 - CVSS 3.1 di NVD
 - CVSS 2.0 di NVD (se applicabile, per versioni precedenti) CVEs
- **Vulnerabilità correlate**: specifica altre vulnerabilità relative alla scoperta. In genere si tratta di altre versioni CVEs che influiscono sulla stessa versione del pacchetto o altre che CVEs appartengono allo stesso gruppo del CVE trovante, come determinato dal fornitore.
- **Risorse interessate**: include informazioni sul registro, sull'archivio, sul tipo di risorsa, sull'ID dell'immagine e sul sistema operativo dell'immagine. Include anche informazioni, ad esempio quando è stata inviata l'ultima volta un'immagine, quante attività Amazon ECS e pod Amazon EKS sono stati distribuiti e quando l'immagine è stata utilizzata l'ultima volta nelle ultime 24 ore. Se hai attività Amazon ECS e pod Amazon EKS distribuiti, puoi visualizzare i dettagli scegliendo il valore per il campo. In questo modo si accede a una schermata in cui è possibile visualizzare informazioni, ad esempio l'ARN del cluster, l'ultimo utilizzo della risorsa nelle ultime 24 ore, il conteggio delle risorse in esecuzione e quelle interrotte e il nome e il tipo di carico di lavoro.

Vulnerabilità del codice

I risultati delle vulnerabilità del codice sono disponibili solo per le funzioni Lambda. Per ulteriori informazioni, consulta [Vulnerabilità del codice](#). Questo tipo di risultato contiene i seguenti dettagli:

- **Correzione disponibile:** per le vulnerabilità del codice questo valore è sempre YES.
- **Nome del rilevatore:** il nome del rilevatore Amazon Q utilizzato per rilevare la vulnerabilità del codice. [Per un elenco di possibili rilevamenti, consulta la libreria Q Detector](#).
- **Tag del rilevatore:** i tag Amazon Q associati al rilevatore, Amazon Q utilizza i tag per classificare i rilevamenti.
- **CWE pertinenti:** IDs delle Common Weakness Enumeration (CWE) associate alla vulnerabilità del codice.
- **Percorso del file:** la posizione del file della vulnerabilità del codice.
- **Posizione della vulnerabilità:** per le vulnerabilità del codice di scansione Lambda, questo campo mostra le righe di codice esatte in cui Amazon Inspector ha rilevato la vulnerabilità.
- **Correzione suggerita:** suggerisce come modificare il codice per correggere il risultato.

Raggiungibilità della rete

I risultati sulla raggiungibilità della rete sono disponibili solo per le istanze EC2. Per ulteriori informazioni, consulta [Raggiungibilità della rete](#). Questo tipo di risultato contiene i seguenti dettagli:

- **Intervallo di porte aperto:** l'intervallo di porte attraverso il quale è possibile accedere all'istanza EC2.
- **Percorsi di rete aperti:** mostra il percorso di accesso aperto all'istanza EC2. Seleziona un elemento sul percorso per ulteriori informazioni.
- **Correzione:** consiglia un metodo per chiudere il percorso di rete aperto.

Visualizzazione del punteggio di Amazon Inspector e comprensione dei dettagli di vulnerability intelligence

Amazon Inspector crea un punteggio per i risultati delle istanze Amazon Elastic Compute Cloud (Amazon EC2). Puoi visualizzare il punteggio di Amazon Inspector e i dettagli sulle vulnerabilità nella console Amazon Inspector. Il punteggio di Amazon Inspector fornisce dettagli che puoi confrontare con le metriche del [Common Vulnerability Scoring System](#). [Questi dettagli sono disponibili solo per](#)

[le rilevazioni di vulnerabilità dei pacchetti](#). Questa sezione descrive come interpretare il punteggio di Amazon Inspector e comprendere i dettagli della vulnerability intelligence.

Punteggio Amazon Inspector

Amazon Inspector crea un punteggio per ogni risultato di Amazon EC2. Amazon Inspector determina il punteggio correlando le informazioni sul punteggio di base CVSS con le informazioni del tuo ambiente di calcolo, come i dati sulla raggiungibilità della rete e i dati di sfruttabilità. Amazon Inspector supporta i fornitori Amazon, Debian e RHEL. Ogni fornitore fornisce un punteggio di base CVSS v3.1. Per gli altri fornitori, Amazon Inspector utilizza un punteggio di base CVSS fornito dal [National Vulnerability Database \(NVD\)](#).

A causa dei requisiti FedRAMP, Amazon Inspector utilizza il punteggio di base CVSS v3.1 come punteggio predefinito. Tuttavia, un punteggio di base [CVSS 4.0](#) verrà incluso nei metadati di vulnerabilità, se disponibile. Il punteggio di base CVSS 4.0 fornisce metriche aggiuntive per migliorare la valutazione delle vulnerabilità. È possibile trovare la fonte e la versione di un punteggio di base CVSS nei dettagli sulla vulnerabilità relativi a un risultato e nei risultati esportati.

Note

Il punteggio Amazon Inspector non è disponibile per le istanze Linux che eseguono Ubuntu. Ubuntu utilizza un sistema di classificazione della gravità personalizzato che differisce dai punteggi CVSS.

Dettagli del punteggio Amazon Inspector

Quando apri la pagina dei dettagli di un risultato, puoi selezionare la scheda Inspector score and vulnerability intelligence. Questo pannello mostra la differenza tra il punteggio di base e il punteggio di Inspector. Questa sezione spiega come Amazon Inspector ha assegnato la classificazione di gravità in base a una combinazione del punteggio Amazon Inspector e del punteggio del fornitore per il pacchetto software. Se i punteggi sono diversi, questo pannello mostra una spiegazione del perché.

Nella sezione delle metriche del punteggio CVSS puoi vedere una tabella con i confronti tra le metriche del punteggio di base CVSS e il punteggio Inspector. [Le metriche confrontate sono le metriche di base definite nel documento delle specifiche CVSS gestito da first.org](#) Di seguito è riportato un riepilogo delle metriche di base:

Vettore di attacco

Il contesto in base al quale una vulnerabilità può essere sfruttata. Per i risultati di Amazon Inspector, questi possono essere Network, Adjacent Network o Local.

Complessità dell'attacco

Questo descrive il livello di difficoltà che un utente malintenzionato dovrà affrontare quando sfrutta la vulnerabilità. Un punteggio basso significa che l'aggressore dovrà soddisfare poche o nessuna condizione aggiuntiva per sfruttare la vulnerabilità. Un punteggio elevato significa che un aggressore dovrà investire una notevole quantità di sforzi per portare a termine con successo un attacco con questa vulnerabilità.

Privilegio richiesto

Questo descrive il livello di privilegio di cui un utente malintenzionato avrà bisogno per sfruttare una vulnerabilità.

Interazione con l'utente

Questa metrica indica se un attacco riuscito che utilizza questa vulnerabilità richiede un utente umano diverso dall'aggressore.

Scope (Ambito)

Indica se una vulnerabilità in un componente vulnerabile influisce sulle risorse dei componenti che esulano dall'ambito di sicurezza del componente vulnerabile. Se questo valore è immutato, la risorsa interessata e la risorsa interessata sono le stesse. Se questo valore viene modificato, il componente vulnerabile può essere sfruttato per influire sulle risorse gestite da diverse autorità di sicurezza.

La riservatezza

Questo misura il livello di impatto sulla riservatezza dei dati all'interno di una risorsa quando la vulnerabilità viene sfruttata. Si va da Nessuno, dove non si perde la riservatezza, a Alto, dove tutte le informazioni all'interno di una risorsa vengono divulgate o possono essere divulgate informazioni riservate come password o chiavi di crittografia.

Integrità

Questo misura il livello di impatto sull'integrità dei dati all'interno della risorsa interessata se la vulnerabilità viene sfruttata. L'integrità è a rischio quando l'aggressore modifica i file all'interno delle risorse interessate. Il punteggio va da Nessuno, dove l'exploit non consente a un utente

malintenzionato di modificare alcuna informazione, a Alto, dove, se sfruttata, la vulnerabilità consentirebbe a un utente malintenzionato di modificare alcuni o tutti i file, oppure i file che potrebbero essere modificati avrebbero gravi conseguenze.

Disponibilità

Questo misura il livello di impatto sulla disponibilità della risorsa interessata quando la vulnerabilità viene sfruttata. Il punteggio va da Nessuno, quando la vulnerabilità non influisce affatto sulla disponibilità, a Alto, dove, se sfruttato, l'aggressore può negare completamente la disponibilità della risorsa o rendere indisponibile un servizio.

Intelligenza sulla vulnerabilità

Questa sezione riassume le informazioni disponibili sul CVE di Amazon e le fonti di intelligence sulla sicurezza standard del settore come la Cybersecurity and Infrastructure Security Agency (CISA).

Note

Intel di CISA o Amazon non sarà disponibile per tutti CVEs.

Puoi visualizzare i dettagli delle informazioni sulle vulnerabilità nella console o utilizzando l'[BatchGetFindingDetails](#) API. Nella console sono disponibili i seguenti dettagli:

ATT&CK

Questa sezione mostra le tattiche, le tecniche e le procedure MITRE (TTPs) associate al CVE. TTPs Vengono mostrate le associate, se ce ne sono più di due applicabili TTPs è possibile selezionare il collegamento per visualizzare un elenco completo. La selezione di una tattica o di una tecnica apre informazioni al riguardo sul sito web MITRE.

CISA

Questa sezione copre le date rilevanti associate alla vulnerabilità. La data in cui la Cybersecurity and Infrastructure Security Agency (CISA) ha aggiunto la vulnerabilità al Known Exploited Vulnerabilities Catalog, sulla base delle prove di uno sfruttamento attivo, e la data di scadenza entro cui CISA prevede che i sistemi vengano corretti. Queste informazioni provengono dal CISA.

Malware noto

Questa sezione elenca i kit e gli strumenti di exploit noti che sfruttano questa vulnerabilità.

Ultima volta segnalata

Questa sezione mostra la data dell'ultimo exploit pubblico noto per questa vulnerabilità.

Comprensione dei livelli di gravità dei risultati di Amazon Inspector

Quando Amazon Inspector genera un risultato, assegna un grado di gravità al risultato. Le valutazioni di gravità ti aiutano a valutare e dare priorità ai risultati. Il grado di gravità di un risultato corrisponde a un punteggio e a un livello numerici: informativo, basso, medio, alto e critico. Amazon Inspector determina il grado di gravità di un risultato in base al tipo di [risultato](#). Questa sezione descrive come Amazon Inspector determina un livello di gravità per ogni tipo di risultato.

Gravità della vulnerabilità dei pacchetti software

Amazon Inspector utilizza il NVD/CVSS punteggio come base per il punteggio di gravità per le vulnerabilità dei pacchetti software. Il NVD/CVSS punteggio è il punteggio di gravità delle vulnerabilità pubblicato da NVD e definito dal CVSS. Il NVD/CVSS punteggio è una composizione di metriche di sicurezza, come la complessità degli attacchi, la maturità del codice degli exploit e i privilegi richiesti. Amazon Inspector produce un punteggio numerico da 1 a 10 che riflette la gravità della vulnerabilità. Amazon Inspector lo classifica come punteggio di base perché riflette la gravità di una vulnerabilità in base alle sue caratteristiche intrinseche, che sono costanti nel tempo. Questo punteggio presuppone anche l'impatto ragionevole nel peggiore dei casi su diversi ambienti distribuiti. [Lo standard CVSS v3 associa i punteggi CVSS](#) ai seguenti livelli di gravità.

Punteggio	Valutazione
0	Messaggio informativo
0,1—3,9	Bassa
4,0—6,9	Media
7,0—8,9	Elevata
9,0—10,0	Critica

Le vulnerabilità rilevate nei pacchetti possono anche avere una gravità pari a Untriaged. Ciò significa che il fornitore non ha ancora impostato un punteggio di vulnerabilità per la vulnerabilità rilevata.

In questo caso, consigliamo di utilizzare il riferimento relativo alla scoperta URLs per ricercare la vulnerabilità e rispondere di conseguenza.

I risultati delle vulnerabilità dei pacchetti includono i seguenti punteggi e i vettori di punteggio associati come parte dei dettagli dei risultati:

- Punteggio EPSS
- Punteggio Inspector
- CVSS 3.1 di Amazon CVE
- CVSS 3.1 di NVD
- CVSS 2.0 da NVD (dove applicabile)

Gravità della vulnerabilità del codice

Per rilevare le vulnerabilità del codice, Amazon Inspector utilizza i livelli di gravità definiti dai rilevatori di Amazon Q che hanno generato il risultato. A ciascun rilevatore viene assegnata una gravità utilizzando il sistema di punteggio CVSS v3.?

Severità della raggiungibilità della rete

Amazon Inspector determina la gravità di una vulnerabilità di raggiungibilità della rete in base al servizio, alle porte e ai protocolli esposti e al tipo di percorso aperto. La tabella seguente definisce questi livelli di gravità. Il valore nella colonna Open path rating rappresenta i percorsi aperti provenienti da gateway virtuali, reti peer e VPCs AWS Direct Connect reti. Tutti gli altri servizi, porte e protocolli esposti hanno una classificazione di gravità informativa.

Servizio	Porte TCP	Porte UDP	Classificazione del percorso Internet	Classificazione del percorso aperto
DHCP	67, 68, 546, 547	67, 68, 546, 547	Media	Messaggio informativo
Elasticsearch	9300, 9200	N/A	Media	Messaggio informativo
FTP	21	21	Elevata	Media

Global catalog LDAP	3268	N/A	Media	Messaggio informativo
Global catalog LDAP over TLS	3269	N/A	Media	Messaggio informativo
HTTP	80	80	Bassa	Messaggio informativo
HTTPS	443	443	Bassa	Messaggio informativo
Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752	Media	Messaggio informativo
LDAP	389	389	Media	Messaggio informativo
LDAP over TLS	636	N/A	Media	Messaggio informativo
MongoDB	27017, 27018, 27019, 28017	N/A	Media	Messaggio informativo
MySQL	3306	N/A	Media	Messaggio informativo
NetBIOS	137, 139	137, 138	Media	Messaggio informativo
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110	Media	Messaggio informativo
Oracle	1521, 1630	N/A	Media	Messaggio informativo
PostgreSQL	5432	N/A	Media	Messaggio informativo
Servizi di stampa	515	N/A	Elevata	Media

RDP	3389	3389	Media	Bassa
RPC	111, 135, 530	111, 135, 530	Media	Messaggio informativo
SMB	445	445	Media	Messaggio informativo
SSH	22	22	Media	Bassa
SQL Server	1433	1434	Media	Messaggio informativo
Syslog	601	514	Media	Messaggio informativo
Telnet	23	23	Elevata	Media
WINS	1512, 42	1512, 42	Media	Messaggio informativo

Gestione dei risultati in Amazon Inspector

Con Amazon Inspector, puoi gestire i risultati in diversi modi. Puoi filtrare i risultati in base al loro stato. Puoi cercare i risultati in base a criteri di filtro. È possibile creare regole di soppressione per escludere i risultati dall'elenco dei risultati. Puoi anche esportare i risultati in AWS Security Hub CSPM Amazon e Amazon EventBridge Simple Storage Service (Amazon S3).

Argomenti

- [Filtraggio dei risultati di Amazon Inspector](#)
- [Eliminazione dei risultati di Amazon Inspector](#)
- [Esportazione dei report dei risultati di Amazon Inspector](#)
- [Creazione di risposte personalizzate ai risultati di Amazon Inspector con Amazon EventBridge](#)

Filtraggio dei risultati di Amazon Inspector

Puoi filtrare i risultati di Amazon Inspector utilizzando criteri di filtro. Se un risultato non corrisponde ai criteri di filtro, Amazon Inspector lo esclude dalla visualizzazione. Questa sezione descrive come filtrare i risultati di Amazon Inspector utilizzando criteri di filtro.

Creazione di filtri nella console Amazon Inspector

In ogni visualizzazione dei risultati, puoi utilizzare la funzionalità di filtro per individuare i risultati con caratteristiche specifiche. I filtri vengono rimossi quando si passa a una visualizzazione a schede diversa.

Un filtro è costituito da un criterio di filtro, che consiste in un attributo di filtro abbinato a un valore di filtro. I risultati che non corrispondono ai criteri di filtro impostati vengono esclusi dall'elenco dei risultati. Ad esempio, per visualizzare tutti i risultati associati al tuo account amministratore, puoi scegliere l'attributo ID dell' AWS account e associarlo al valore dell'ID dell' AWS account a dodici cifre.

Alcuni criteri di filtro si applicano a tutti i risultati, mentre altri sono disponibili solo per tipi di risorse specifici o per tipi di ricerca.

Per applicare un filtro alla visualizzazione dei risultati

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Nel riquadro di navigazione, seleziona Esiti. La visualizzazione predefinita mostra tutti i risultati con uno stato Attivo.
3. Per filtrare i risultati in base a criteri, seleziona la barra Aggiungi filtro per visualizzare un elenco di tutti i criteri di filtro applicabili per quella vista. Sono disponibili criteri di filtro diversi in diverse visualizzazioni.
4. Scegliete un criterio in base al quale filtrare dall'elenco.
5. Dal riquadro di immissione del criterio, immettete i valori del filtro desiderati per definire tale criterio.
6. Scegliete Applica per applicare quel criterio di filtro ai risultati correnti. Puoi continuare ad aggiungere altri criteri di filtro selezionando nuovamente la barra di immissione del filtro.
7. (Facoltativo) Per visualizzare i risultati soppressi o chiusi, scegliete Attivo nella barra dei filtri, quindi scegliete Soppressi o Chiusi. Scegli Mostra tutto per visualizzare i risultati attivi, soppressi e chiusi nella stessa visualizzazione.

Eliminazione dei risultati di Amazon Inspector

È possibile creare regole di soppressione per nascondere i risultati che corrispondono ai criteri. Ad esempio, è possibile creare una regola di soppressione per nascondere i risultati in base ai relativi livelli di gravità. Se Amazon Inspector genera un risultato che corrisponde alla tua regola di soppressione, Amazon Inspector elimina il risultato e lo nasconde alla vista. Amazon Inspector archivia i risultati soppressi fino a quando non vengono corretti. Una volta corretto un risultato soppresso, Amazon Inspector chiude il risultato. Puoi visualizzare i risultati soppressi nella console.

È possibile creare regole di soppressione per dare priorità ai risultati più importanti. Le regole di soppressione non hanno alcun impatto sui risultati, poiché si limitano a nasconderli alla vista. Non è possibile creare una regola di soppressione che chiuda o corregga i risultati. Puoi anche [eliminare i risultati indesiderati AWS Security Hub CSPM con una EventBridge regola Amazon](#). Le procedure in questa sezione descrivono come creare, visualizzare, modificare ed eliminare una regola di soppressione.

Note

Solo l'amministratore delegato di un'organizzazione può creare e gestire le regole di soppressione.

Creazione di una regola di soppressione

È possibile creare regole di soppressione per filtrare l'elenco dei risultati visualizzati per impostazione predefinita. È possibile creare una regola di soppressione a livello di codice utilizzando l'[CreateFilter](#) API e specificando SUPPRESS come valore per `action`

Note

Solo gli account autonomi e gli amministratori delegati di Amazon Inspector possono creare e gestire regole di soppressione. I membri di un'organizzazione non vedranno l'opzione per le regole di soppressione nel pannello di navigazione.

Per creare una regola di soppressione (console)

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Nel pannello di navigazione, scegli Regole di soppressione. Quindi scegli Create rule (Crea regola).
3. Per ogni criterio, effettuate le seguenti operazioni:
 - Seleziona la barra dei filtri per visualizzare un elenco di criteri di filtro che puoi aggiungere alla regola di soppressione.
 - Seleziona i criteri di filtro per la tua regola di soppressione.
4. Dopo aver aggiunto i criteri, inserite un nome per la regola e una descrizione facoltativa.
5. Scegli Salva regola. Amazon Inspector applica la nuova regola di soppressione e nasconde tutti i risultati che corrispondono ai criteri. Tieni presente che le regole di soppressione possono richiedere fino a 24 ore per avere effetto.

Visualizzazione dei risultati soppressi

Per impostazione predefinita, Amazon Inspector non visualizza i risultati soppressi nella console Amazon Inspector. Tuttavia, puoi visualizzare i risultati soppressi da una regola particolare.

Per visualizzare i risultati soppressi

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Nel riquadro di navigazione, seleziona Regole di soppressione.
3. Nell'elenco delle regole di soppressione, seleziona il titolo della regola.

Modifica di una regola di soppressione

È possibile apportare modifiche alle regole di soppressione in qualsiasi momento.

Per modificare le regole di soppressione

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Regole di soppressione.
3. Scegliete il nome della regola di soppressione che desiderate modificare, quindi scegliete Modifica.
4. Apportate le modifiche desiderate, quindi scegliete Salva.

Eliminazione di una regola di soppressione

È possibile eliminare le regole di soppressione. Se elimini una regola di soppressione, Amazon Inspector smette di sopprimere le occorrenze nuove ed esistenti di risultati che soddisfano i criteri della regola e che non sono soppressi da altre regole.

Dopo aver eliminato una regola di soppressione, le occorrenze di risultati nuove ed esistenti che soddisfacevano i criteri della regola hanno lo stato Attivo. Ciò significa che vengono visualizzati per impostazione predefinita sulla console Amazon Inspector. Inoltre, Amazon Inspector pubblica questi risultati su AWS Security Hub CSPM e Amazon come eventi. EventBridge

Per eliminare una regola di soppressione

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Nel riquadro di navigazione, seleziona Regole di soppressione.
3. Seleziona la casella di controllo accanto al titolo della regola di soppressione che desideri eliminare.
4. Scegliete Elimina, quindi confermate la scelta di eliminare definitivamente la regola.

Esportazione dei report dei risultati di Amazon Inspector

Un rapporto sui risultati è un file CSV o JSON che fornisce un'istantanea dettagliata dei risultati. Puoi esportare un report dei risultati su Amazon e Amazon EventBridge Simple Storage Service (Amazon S3). AWS Security Hub CSPM Quando configuri un rapporto sui risultati, specifichi quali risultati includere in esso. Per impostazione predefinita, il rapporto sui risultati include i dati per tutti i risultati attivi. Se sei l'amministratore delegato di un'organizzazione, il rapporto sui risultati include i dati per tutti gli account dei membri dell'organizzazione. Per personalizzare un rapporto sui risultati, crea e applica [un filtro](#).

Quando esporti un report sui risultati, Amazon Inspector crittografa i dati dei risultati con un AWS KMS key metodo da te specificato. Dopo che Amazon Inspector ha crittografato i dati dei risultati, archivia il report dei risultati in un bucket Amazon S3 da te specificato. La AWS KMS chiave deve essere utilizzata nello Regione AWS stesso bucket Amazon S3. La tua policy AWS KMS chiave deve consentire ad Amazon Inspector di utilizzarlo e la tua policy sui bucket Amazon S3 deve consentire ad Amazon Inspector di aggiungervi oggetti. Dopo aver esportato il report dei risultati, puoi scaricarlo dal tuo bucket Amazon S3 o trasferirlo in una nuova posizione. Puoi anche usare il tuo bucket Amazon S3 come repository per altri report sui risultati esportati.

Questa sezione descrive come esportare un report dei risultati nella console Amazon Inspector. Le seguenti attività richiedono la verifica delle autorizzazioni, la configurazione di un bucket Amazon S3, la configurazione, la configurazione e l'esportazione di AWS KMS key un report sui risultati.

Note

Se esporti un report sui risultati con l'[CreateFindingsReport](#) API Amazon Inspector, puoi visualizzare solo i risultati attivi. Se desideri visualizzare i risultati eliminati o chiusi, devi specificarli SUPPRESSED o inserirli CLOSED nei criteri di [filtro](#).

Processi

- [Passaggio 1: verifica le tue autorizzazioni](#)
- [Passaggio 2: configura un bucket S3](#)
- [Fase 3: Configurare un AWS KMS key](#)
- [Fase 4: Configurare ed esportare un rapporto sui risultati](#)
- [Risolvi gli errori di esportazione](#)

Passaggio 1: verifica le tue autorizzazioni

Note

Dopo aver esportato un rapporto sui risultati per la prima volta, i passaggi da 1 a 3 sono facoltativi. La procedura da seguire dipende dal fatto che si desideri utilizzare lo stesso bucket Amazon S3 e AWS KMS key per altri report sui risultati esportati. Se desideri esportare un report dei risultati in modo programmatico dopo aver completato i passaggi 1-3, utilizza il [CreateFindingsReport](#) funzionamento dell'API Amazon Inspector.

Prima di esportare un report sui risultati da Amazon Inspector, verifica di disporre delle autorizzazioni necessarie sia per esportare i report sui risultati sia per configurare le risorse per la crittografia e l'archiviazione dei report. Per verificare le tue autorizzazioni, utilizza AWS Identity and Access Management (IAM) per rivedere le policy IAM associate alla tua identità IAM. Quindi confronta le informazioni contenute in tali policy con il seguente elenco di azioni che devi essere autorizzato a eseguire per esportare un rapporto sui risultati.

Amazon Inspector

Per Amazon Inspector, verifica di essere autorizzato a eseguire le seguenti azioni:

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

Queste azioni ti consentono di recuperare i dati dei risultati per il tuo account e di esportare tali dati nei report sui risultati.

Se prevedi di esportare report di grandi dimensioni a livello di codice, potresti anche verificare di essere autorizzato a eseguire le seguenti azioni: `inspector2:GetFindingsReportStatus`

controllare lo stato dei report e `inspector2:CancelFindingsReport` annullare le esportazioni in corso.

AWS KMS

Verifica AWS KMS, infatti, di avere il permesso di eseguire le seguenti azioni:

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

Queste azioni ti consentono di recuperare e aggiornare la policy chiave AWS KMS key che desideri venga utilizzata da Amazon Inspector per crittografare il report.

Per utilizzare la console Amazon Inspector per esportare un report, verifica anche di essere autorizzato a eseguire le seguenti AWS KMS azioni:

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

Queste azioni ti consentono di recuperare e visualizzare informazioni AWS KMS keys relative al tuo account. Puoi quindi scegliere una di queste chiavi per crittografare il rapporto.

Se intendi creare una nuova chiave KMS per la crittografia del rapporto, devi anche essere autorizzato a eseguire l'`kms:CreateKey` azione.

Amazon S3

Per Amazon S3, verifica di essere autorizzato a eseguire le seguenti azioni:

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

Queste azioni ti consentono di creare e configurare il bucket S3 in cui desideri che Amazon Inspector memorizzi il report. Consentono inoltre di aggiungere ed eliminare oggetti dal bucket.

Se prevedi di utilizzare la console Amazon Inspector per esportare il report, verifica anche di avere il permesso di eseguire le azioni `s3:ListAllMyBuckets` e `s3:GetBucketLocation`.

Queste azioni ti consentono di recuperare e visualizzare informazioni sui bucket S3 del tuo account. Puoi quindi scegliere uno di questi bucket per archiviare il rapporto.

Se non sei autorizzato a eseguire una o più delle azioni richieste, chiedi assistenza AWS all'amministratore prima di procedere al passaggio successivo.

Passaggio 2: configura un bucket S3

Dopo aver verificato le autorizzazioni, sei pronto per configurare il bucket S3 in cui desideri archiviare il rapporto sui risultati. Può essere un bucket esistente per il tuo account o un bucket esistente di proprietà di un altro Account AWS a cui puoi accedere. Se desideri archiviare il rapporto in un nuovo bucket, crea il bucket prima di procedere.

Il bucket S3 deve trovarsi nella Regione AWS stessa cartella dei dati dei risultati che desideri esportare. Ad esempio, se utilizzi Amazon Inspector nella regione Stati Uniti orientali (Virginia settentrionale) e desideri esportare i dati dei risultati per quella regione, il bucket deve trovarsi anche nella regione Stati Uniti orientali (Virginia settentrionale).

Inoltre, la politica del bucket deve consentire ad Amazon Inspector di aggiungere oggetti al bucket. Questo argomento spiega come aggiornare la policy del bucket e fornisce un esempio della dichiarazione da aggiungere alla policy. Per informazioni dettagliate sull'aggiunta e l'aggiornamento delle policy dei bucket, consulta [Using bucket policies](#) nella Amazon Simple Storage Service User Guide.

Se desideri archiviare il report in un bucket S3 di proprietà di un altro account, contatta il proprietario del bucket per aggiornare la policy del bucket. Ottieni anche l'URI per il bucket. Dovrai inserire questo URI quando esporti il rapporto.

Per aggiornare la policy sui bucket

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com Amazon S3 all'indirizzo /s3.](https://console.aws.amazon.com/AmazonS3/)
2. Nel pannello di navigazione, scegli Bucket.
3. Scegli il bucket S3 in cui desideri archiviare il report dei risultati.
4. Scegli la scheda Autorizzazioni.
5. Seleziona Modifica nella sezione Policy bucket.
6. Copia la seguente dichiarazione di esempio negli appunti:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "allow-inspector",
      "Effect": "Allow",
      "Principal": {
        "Service": "inspector2.amazonaws.com"
      },
      "Action": [
        "s3:PutObject",
        "s3:PutObjectAcl",
        "s3:AbortMultipartUpload"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:inspector2:us-
east-1:111122223333:report/*"
        }
      }
    }
  ]
}
```

7. Nell'editor di policy Bucket sulla console Amazon S3, incolla l'istruzione precedente nella policy per aggiungerla alla policy.

Quando aggiungi l'istruzione, assicurati che la sintassi sia valida. Le policy Bucket utilizzano il formato JSON. Ciò significa che è necessario aggiungere una virgola prima o dopo l'istruzione, a seconda di dove si aggiunge l'istruzione alla politica. Se aggiungete l'istruzione come ultima istruzione, aggiungete una virgola dopo la parentesi di chiusura dell'istruzione precedente. Se la aggiungete come prima istruzione o tra due istruzioni esistenti, aggiungete una virgola dopo la parentesi che chiude l'istruzione.

8. Aggiorna l'istruzione con i valori corretti per il tuo ambiente, dove:

- `amzn-s3-demo-bucket` è il nome del bucket.
- `111122223333` è l'ID dell'account per il tuo Account AWS.
- `Region` è la Regione AWS applicazione in cui utilizzi Amazon Inspector e desideri consentire ad Amazon Inspector di aggiungere report al bucket. Ad esempio, `us-east-1` per la regione Stati Uniti orientali (Virginia settentrionale).

Note

Se utilizzi Amazon Inspector in modalità abilitata manualmente Regione AWS, aggiungi anche il codice regionale appropriato al valore del campo. `Service` Questo campo specifica il principale del servizio Amazon Inspector.

Ad esempio, se utilizzi Amazon Inspector nella regione del Medio Oriente (Bahrein), che ha il codice regionale `me-south-1`, `inspector2.amazonaws.com` sostituisilo con nell'istruzione. `inspector2.me-south-1.amazonaws.com`

Tieni presente che l'istruzione di esempio definisce condizioni che utilizzano due chiavi di condizione globali IAM:

- [aws: SourceAccount](#) — Questa condizione consente ad Amazon Inspector di aggiungere report al bucket solo per il tuo account. Impedisce ad Amazon Inspector di aggiungere report per altri account. Più specificamente, la condizione specifica quale account può utilizzare il bucket per le risorse e le azioni specificate dalla condizione. `aws:SourceArn`

Per archiviare i report relativi ad account aggiuntivi nel bucket, aggiungi l'ID account per ogni account aggiuntivo a questa condizione. Esempio:

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

- [aws: SourceArn](#) — Questa condizione limita l'accesso al bucket in base alla fonte degli oggetti che vengono aggiunti al bucket. Impedisce ad altri Servizi AWS di aggiungere oggetti al bucket. Inoltre, impedisce ad Amazon Inspector di aggiungere oggetti al bucket mentre esegue altre azioni per il tuo account. Più specificamente, la condizione consente ad Amazon Inspector di aggiungere oggetti al bucket solo se si tratta di report sui risultati e solo se tali report vengono creati dall'account e nella regione specificata nella condizione.

Per consentire ad Amazon Inspector di eseguire le azioni specificate per account aggiuntivi, aggiungi Amazon Resource Names (ARNs) per ogni account aggiuntivo a questa condizione.

Esempio:

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

Gli account specificati dalle `aws:SourceArn` condizioni `aws:SourceAccount` e devono corrispondere.

Entrambe le condizioni aiutano a evitare che Amazon Inspector venga usato come [sostituto confuso](#) durante le transazioni con Amazon S3. Sebbene non sia consigliabile, puoi rimuovere queste condizioni dalla bucket policy.

9. Al termine dell'aggiornamento della policy del bucket, scegli Salva modifiche.

Fase 3: Configurare un AWS KMS key

Dopo aver verificato le autorizzazioni e configurato il bucket S3, stabilisci quale AWS KMS key vuoi che Amazon Inspector utilizzi per crittografare il report dei risultati. La chiave deve essere una chiave KMS di crittografia simmetrica gestita dal cliente. Inoltre, la chiave deve trovarsi nello stesso Regione AWS bucket S3 configurato per archiviare il report.

La chiave può essere una chiave KMS esistente del tuo account o una chiave KMS esistente di proprietà di un altro account. Se desideri utilizzare una nuova chiave KMS, crea la chiave prima di procedere. Se desideri utilizzare una chiave esistente di proprietà di un altro account, ottieni l'Amazon Resource Name (ARN) della chiave. Dovrai inserire questo ARN quando esporti il report da Amazon Inspector. Per informazioni sulla creazione e la revisione delle impostazioni per le chiavi KMS, consulta [Managing keys](#) nella Developer Guide.AWS Key Management Service

Dopo aver determinato quale chiave KMS desideri utilizzare, autorizza Amazon Inspector a utilizzare la chiave. Altrimenti, Amazon Inspector non sarà in grado di crittografare ed esportare il report. Per autorizzare Amazon Inspector a utilizzare la chiave, aggiorna la policy relativa alla chiave. Per informazioni dettagliate sulle politiche chiave e sulla gestione dell'accesso alle chiavi KMS, consulta [le politiche chiave AWS KMS nella Guida](#) per gli AWS Key Management Service sviluppatori.

Note

La procedura seguente serve per aggiornare una chiave esistente per consentire ad Amazon Inspector di utilizzarla. Se non disponi di una chiave esistente, consulta [Creating keys](#) nella AWS Key Management Service Developer Guide.

Per aggiornare la politica chiave

1. Accedi utilizzando le tue credenziali, quindi apri la AWS KMS console all'indirizzo <https://console.aws.amazon.com/kms>.
2. Nel riquadro di navigazione, scegli Chiavi gestite dal cliente.
3. Scegli la chiave KMS che desideri utilizzare per crittografare il rapporto. La chiave deve essere una chiave di crittografia simmetrica (SYMMETRIC_DEFAULT).
4. Nella scheda Policy della chiave, seleziona Modifica. Se non viene visualizzata una politica chiave con il pulsante Modifica, è necessario prima selezionare Passa alla visualizzazione dei criteri.
5. Copia la seguente dichiarazione di esempio negli appunti:

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

6. Nell'editor Key policy sulla AWS KMS console, incolla l'istruzione precedente nella policy chiave per aggiungerla alla policy.

Quando aggiungi l'istruzione, assicurati che la sintassi sia valida. Le politiche chiave utilizzano il formato JSON. Ciò significa che è necessario aggiungere una virgola prima o dopo l'istruzione, a seconda di dove si aggiunge l'istruzione alla politica. Se aggiungete l'istruzione come ultima istruzione, aggiungete una virgola dopo la parentesi di chiusura dell'istruzione precedente. Se la aggiungete come prima istruzione o tra due istruzioni esistenti, aggiungete una virgola dopo la parentesi che chiude l'istruzione.

7. Aggiorna l'istruzione con i valori corretti per il tuo ambiente, dove:
 - **111122223333** è l'ID dell'account per il tuo Account AWS.
 - **Region** è l'opzione Regione AWS in cui desideri consentire ad Amazon Inspector di crittografare i report con la chiave. Ad esempio, `us-east-1` per la regione Stati Uniti orientali (Virginia settentrionale).

Note

Se utilizzi Amazon Inspector in modalità abilitata manualmente Regione AWS, aggiungi anche il codice regionale appropriato al valore del campo. *Service* Ad esempio, se utilizzi Amazon Inspector nella regione del Medio Oriente (Bahrein), sostituiscilo con `inspector2.amazonaws.com inspector2.me-south-1.amazonaws.com`

Come l'istruzione di esempio per la bucket policy nel passaggio precedente, i `Condition` campi di questo esempio utilizzano due chiavi di condizione globali IAM:

- [aws:SourceAccount](#) — Questa condizione consente ad Amazon Inspector di eseguire le azioni specificate solo per il tuo account. Più specificamente, determina quale account può eseguire le azioni specificate per le risorse e le azioni specificate dalla `aws:SourceArn` condizione.

Per consentire ad Amazon Inspector di eseguire le azioni specificate per account aggiuntivi, aggiungi l'ID account per ogni account aggiuntivo a questa condizione. Esempio:

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

- [aws: SourceArn](#) — Questa condizione Servizi AWS impedisce ad altri di eseguire le azioni specificate. Inoltre, impedisce ad Amazon Inspector di utilizzare la chiave mentre esegue altre azioni per il tuo account. In altre parole, consente ad Amazon Inspector di crittografare gli oggetti S3 con la chiave solo se si tratta di report sui risultati e solo se tali report vengono creati dall'account e nella regione specificata nella condizione.

Per consentire ad Amazon Inspector di eseguire le azioni specificate per account aggiuntivi, aggiungi questa condizione ARNs per ogni account aggiuntivo. Esempio:

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

Gli account specificati dalle `aws:SourceArn` condizioni `aws:SourceAccount` e devono corrispondere.

Queste condizioni aiutano a evitare che Amazon Inspector venga usato come [assistente confuso](#) durante le transazioni con. AWS KMS Sebbene non sia consigliabile, puoi rimuovere queste condizioni dall'informativa.

8. Al termine dell'aggiornamento della politica chiave, scegli Salva modifiche.

Fase 4: Configurare ed esportare un rapporto sui risultati

Note

È possibile esportare solo un rapporto sui risultati alla volta. Se è attualmente in corso un'esportazione, è necessario attendere il completamento dell'esportazione prima di esportare un altro rapporto sui risultati.

Dopo aver verificato le autorizzazioni e configurato le risorse per crittografare e archiviare il rapporto sui risultati, sei pronto per configurare ed esportare il rapporto.

Per configurare ed esportare un rapporto sui risultati

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Nel riquadro di navigazione, in Risultati, scegli Tutti i risultati.
3. (Facoltativo) Utilizzando la barra dei filtri sopra la tabella Risultati, [aggiungete criteri di filtro](#) che specificano quali risultati includere nel rapporto. Man mano che aggiungi criteri, Amazon Inspector aggiorna la tabella per includere solo i risultati che soddisfano i criteri. La tabella fornisce un'anteprima dei dati che conterrà il rapporto.

Note

Ti consigliamo di aggiungere criteri di filtro. In caso contrario, il rapporto includerà i dati relativi a tutti i risultati attualmente trovati Regione AWS con lo stato Attivo. Se sei l'amministratore di Amazon Inspector di un'organizzazione, questo include i dati relativi ai risultati di tutti gli account membri della tua organizzazione.

Se un report include dati relativi a tutti o a molti risultati, la generazione e l'esportazione del report possono richiedere molto tempo e puoi esportare solo un report alla volta.

4. Scegli Esporta risultati.
5. Nella sezione Impostazioni di esportazione, per Tipo di file di esportazione, specifica un formato di file per il rapporto:

- Per creare un file JavaScript Object Notation (.json) che contenga i dati, scegliete JSON.

Se scegli l'opzione JSON, il rapporto includerà tutti i campi per ogni risultato. Per un elenco di possibili campi JSON, consulta il tipo di dati [Finding](#) nel riferimento all'API Amazon Inspector.

- Per creare un file con valori separati da virgole (.csv) che contenga i dati, scegli CSV.

Se scegli l'opzione CSV, il rapporto includerà solo un sottoinsieme dei campi per ogni risultato, circa 45 campi che riportano gli attributi chiave di un risultato. I campi includono: Tipo di ricerca, Titolo, Severità, Stato, Descrizione, Primo visualizzato, Ultima visualizzazione, Correzione disponibile, ID AWS account, ID risorsa, Tag risorsa e Correzione. Questi si aggiungono ai campi che contengono i dettagli del punteggio e i riferimenti URLs per ogni risultato. Di seguito è riportato un esempio delle intestazioni CSV di un rapporto sui risultati:

AVVERTENZE (URL) per i rapporti di sicurezza. Per informazioni sui rapporti di sicurezza, vedere Guida per l'utente di Amazon Inspector. Per informazioni sui rapporti di sicurezza, vedere Guida per l'utente di Amazon Inspector. Per informazioni sui rapporti di sicurezza, vedere Guida per l'utente di Amazon Inspector. Per informazioni sui rapporti di sicurezza, vedere Guida per l'utente di Amazon Inspector.

6. In Export location, per S3 URI, specifica il bucket S3 in cui desideri archiviare il report:

- Per archiviare il report in un bucket di proprietà del tuo account, scegli Browse S3. Amazon Inspector visualizza una tabella dei bucket S3 per il tuo account. Seleziona la riga per il bucket che desideri, quindi scegli Scegli.

Tip

Per specificare anche un prefisso di percorso Amazon S3 per il report, aggiungi una barra (/) e il prefisso al valore nella casella URI S3. Amazon Inspector include quindi il prefisso quando aggiunge il report al bucket e Amazon S3 genera il percorso specificato dal prefisso.

Ad esempio, se desideri utilizzare il tuo Account AWS ID come prefisso e l'ID dell'account è 111122223333, **/111122223333** aggiungilo al valore nella casella URI S3.

Un prefisso è simile al percorso di una directory all'interno di un bucket S3. Ti consente di raggruppare oggetti simili in un bucket, proprio come potresti archiviare file simili in una cartella su un file system. Per ulteriori informazioni, consulta [Organizzazione degli oggetti nella console Amazon S3 utilizzando le cartelle](#) nella Guida per l'utente di Amazon Simple Storage Service.

- Per archiviare il report in un bucket di proprietà di un altro account, inserisci l'URI del bucket, ad esempio **s3://DOC-EXAMPLE_BUCKET**, dove DOC-EXAMPLE_BUCKET è il nome del bucket. Il proprietario del bucket può trovare queste informazioni per te nelle proprietà del bucket.
7. Per la chiave KMS, specifica quella AWS KMS key che desideri utilizzare per crittografare il rapporto:

- Per utilizzare una chiave del tuo account, scegli la chiave dall'elenco. L'elenco mostra le chiavi KMS con crittografia simmetrica gestite dal cliente per il tuo account.
- Per utilizzare una chiave di proprietà di un altro account, inserisci l'Amazon Resource Name (ARN) della chiave. Il proprietario della chiave può trovare queste informazioni per te nelle proprietà della chiave. Per ulteriori informazioni, consulta [Finding the key ID and key ARN](#) nella AWS Key Management Service Developer Guide.

8. Scegli Export (Esporta).

Amazon Inspector genera il report dei risultati, lo crittografa con la chiave KMS specificata e lo aggiunge al bucket S3 specificato. A seconda del numero di risultati che hai scelto di includere nel report, questo processo può richiedere diversi minuti o ore. Una volta completata l'esportazione, Amazon Inspector visualizza un messaggio che indica che il report dei risultati è stato esportato correttamente. Facoltativamente, scegli Visualizza report nel messaggio per accedere al report in Amazon S3.

Tieni presente che puoi esportare solo un report alla volta. Se è attualmente in corso un'esportazione, attendi il completamento dell'esportazione prima di provare a esportare un altro rapporto.

Risolvi gli errori di esportazione

Se si verifica un errore durante il tentativo di esportare un report sui risultati, Amazon Inspector visualizza un messaggio che descrive l'errore. Puoi utilizzare le informazioni contenute in questo argomento come guida per identificare le possibili cause e soluzioni dell'errore.

Ad esempio, verifica che il bucket S3 sia nella versione corrente Regione AWS e che la politica del bucket consenta ad Amazon Inspector di aggiungere oggetti al bucket. Verifica inoltre che AWS KMS key sia abilitato nella regione corrente e assicurati che la policy chiave consenta ad Amazon Inspector di utilizzare la chiave.

Dopo aver risolto l'errore, prova a esportare nuovamente il report.

Non è possibile avere più segnalazioni di errore

Se stai tentando di creare un report ma Amazon Inspector lo sta già generando, riceverai un errore che indica Motivo: impossibile avere più report in corso. Questo errore si verifica perché Amazon Inspector può generare un solo report per account alla volta.

Per risolvere l'errore, puoi attendere che l'altro report finisca o annullarlo prima di richiedere un nuovo report.

È possibile controllare lo stato di un rapporto utilizzando l'[GetFindingsReportStatus](#) operazione, questa operazione restituisce l'ID del rapporto di qualsiasi rapporto attualmente in fase di generazione.

Se necessario, è possibile utilizzare l'ID del rapporto fornito dall'[GetFindingsReportStatus](#) operazione per annullare un'esportazione attualmente in corso utilizzando l'[CancelFindingsReport](#) operazione.

Creazione di risposte personalizzate ai risultati di Amazon Inspector con Amazon EventBridge

Amazon Inspector crea un evento in [Amazon EventBridge](#) per i risultati di nuova generazione e i risultati aggregati. Amazon Inspector crea anche un evento per eventuali modifiche allo stato di un risultato. Ciò significa che Amazon Inspector crea un nuovo evento per un risultato quando intraprendi azioni come il riavvio di una risorsa o la modifica dei tag associati a una risorsa. Quando Amazon Inspector crea un nuovo evento per un risultato aggiornato, il risultato `id` rimane invariato.

Note

Se il tuo account è un account amministratore delegato di Amazon Inspector, EventBridge pubblica gli eventi sul tuo account e sull'account membro da cui hanno avuto origine gli eventi.

Quando utilizzi EventBridge gli eventi con Amazon Inspector, puoi automatizzare le attività per aiutarti a rispondere ai problemi di sicurezza rivelati dai risultati. Per ricevere notifiche sui risultati di Amazon Inspector in base EventBridge agli eventi, devi creare [una EventBridge regola e specificare un](#) obiettivo per Amazon Inspector. La EventBridge regola consente di EventBridge inviare notifiche per i risultati di Amazon Inspector e la destinazione specifica dove inviare le notifiche.

Amazon Inspector invia eventi al bus degli eventi predefinito nel luogo in Regione AWS cui utilizzi attualmente Amazon Inspector. Ciò significa che devi configurare le regole degli eventi per ognuna delle Regione AWS quali hai attivato Amazon Inspector e configurato Amazon Inspector per ricevere eventi. EventBridge Amazon Inspector emette eventi con la massima diligenza possibile.

Questa sezione fornisce un esempio di schema di eventi e descrive come creare una regola.

EventBridge

Schema degli eventi

Di seguito è riportato un esempio del formato di evento Amazon Inspector per un evento di ricerca EC2. Per uno schema di esempio di altri tipi di ricerca e tipi di eventi, vedi [EventBridge schema](#)

```
{
  "version": "0",
  "id": "66a7a279-5f92-971c-6d3e-c92da0950992",
  "detail-type": "Inspector2 Finding",
  "source": "aws.inspector2",
  "account": "111122223333",
  "time": "2023-01-19T22:46:15Z",
  "region": "us-east-1",
  "resources": ["i-0c2a343f1948d5205"],
  "detail": {
    "awsAccountId": "111122223333",
    "description": "\n It was discovered that the sound subsystem in the Linux kernel contained a\n race condition in some situations. A local attacker could use this to cause\n a denial of service (system crash).",
    "exploitAvailable": "YES",
    "exploitabilityDetails": {
      "lastKnownExploitAt": "Oct 24, 2022, 11:08:59 PM"
    },
    "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/FINDING_ID",
    "firstObservedAt": "Jan 19, 2023, 10:46:15 PM",
    "fixAvailable": "YES",
    "lastObservedAt": "Jan 19, 2023, 10:46:15 PM",
    "packageVulnerabilityDetails": {
      "cvss": [{
        "baseScore": 4.7,
        "scoringVector": "CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:H",
        "source": "NVD",
        "version": "3.1"
      }],
      "referenceUrls": ["https://lore.kernel.org/all/CAFc06XN7JDM4xSXGhtusQfS2mSBcx50VJKwQpCq=WeLt57aaZA@mail.gmail.com/", "https://ubuntu.com/security/notices/USN-5792-1", "https://ubuntu.com/security/notices/USN-5791-2", "https://ubuntu.com/security/notices/USN-5791-1", "https://ubuntu.com/security/notices/USN-5793-2", "https://git.kernel.org/pub/scm/linux/kernel/git/
```

```

torvalds/linux.git/commit/?id=8423f0b6d513b259fdab9c9bf4aaa6188d054c2d", "https://
ubuntu.com/security/notices/USN-5793-1", "https://ubuntu.com/security/notices/
USN-5792-2", "https://ubuntu.com/security/notices/USN-5791-3", "https://ubuntu.com/
security/notices/USN-5793-4", "https://ubuntu.com/security/notices/USN-5793-3",
"https://git.kernel.org/linus/8423f0b6d513b259fdab9c9bf4aaa6188d054c2d(6.0-rc5)",
"https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-3303"],
  "relatedVulnerabilities": [],
  "source": "UBUNTU_CVE",
  "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2022/
CVE-2022-3303.html",
  "vendorCreatedAt": "Sep 27, 2022, 11:15:00 PM",
  "vendorSeverity": "medium",
  "vulnerabilityId": "CVE-2022-3303",
  "vulnerablePackages": [{
    "arch": "X86_64",
    "epoch": 0,
    "fixedInVersion": "0:5.15.0.1027.31~20.04.16",
    "name": "linux-image-aws",
    "packageManager": "OS",
    "remediation": "apt update && apt install --only-upgrade linux-image-
aws",
    "version": "5.15.0.1026.30~20.04.16"
  }]
},
"remediation": {
  "recommendation": {
    "text": "None Provided"
  }
},
"resources": [{
  "details": {
    "awsEc2Instance": {
      "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
      "imageId": "ami-0b7ff1a8d69f1bb35",
      "ipV4Addresses": ["172.31.85.212", "44.203.45.27"],
      "ipV6Addresses": [],
      "launchedAt": "Jan 19, 2023, 7:53:14 PM",
      "platform": "UBUNTU_20_04",
      "subnetId": "subnet-8213f2a3",
      "type": "t2.micro",
      "vpcId": "vpc-ab6650d1"
    }
  }
},

```

```
        "id": "i-0c2a343f1948d5205",
        "partition": "aws",
        "region": "us-east-1",
        "type": "AWS_EC2_INSTANCE"
    }],
    "severity": "MEDIUM",
    "status": "ACTIVE",
    "title": "CVE-2022-3303 - linux-image-aws",
    "type": "PACKAGE_VULNERABILITY",
    "updatedAt": "Jan 19, 2023, 10:46:15 PM"
}
}
```

Creazione di una EventBridge regola per notificarti i risultati di Amazon Inspector

Per aumentare la visibilità dei risultati di Amazon Inspector, puoi impostare avvisi EventBridge di ricerca automatizzati che vengono inviati a un hub di messaggistica. Questo argomento mostra come inviare avvisi CRITICAL e rilevazioni sulla HIGH gravità a e-mail, Slack o Amazon Chime. Imparerai come impostare un argomento di Amazon Simple Notification Service e quindi collegare tale argomento a una regola di EventBridge evento.

Passaggio 1. Configurare un argomento e un endpoint di Amazon SNS

Per configurare avvisi automatici, devi prima impostare un argomento in Amazon Simple Notification Service e aggiungere un endpoint. Per ulteriori informazioni, consulta la guida [SNS](#).


Questa procedura stabilisce dove inviare i dati relativi ai risultati di Amazon Inspector. L'argomento SNS può essere aggiunto a una regola di EventBridge evento durante o dopo la creazione della regola dell'evento.

Email setup

Creazione di un argomento SNS

1. [Accedi alla console Amazon SNS all'indirizzo `https://console.aws.amazon.com/sns/v3/home`.](https://console.aws.amazon.com/sns/v3/home)
2. Dal pannello di navigazione, seleziona Argomenti, quindi seleziona Crea argomento.
3. Nella sezione Crea argomento, seleziona Standard. Quindi, inserisci il nome di un argomento, ad esempio **Inspector_to_Email**. Altri dettagli sono facoltativi.

4. Seleziona Create Topic (Crea argomento). Verrà aperto un nuovo pannello con i dettagli del nuovo argomento.
5. Nella sezione Abbonamenti, seleziona Crea abbonamento.
6.
 - a. Dal menu Protocollo selezionare E-mail.
 - b. Nel campo Endpoint, inserisci l'indirizzo email a cui desideri ricevere le notifiche.

 Note

Ti verrà richiesto di confermare l'iscrizione tramite il tuo client di posta elettronica dopo aver creato l'abbonamento.

- c. Scegli Create Subscription (Crea sottoscrizione).
7. Cerca un messaggio di iscrizione nella tua casella di posta e scegli Conferma abbonamento.


Slack setup

Creazione di un argomento SNS

1. [Accedi alla console Amazon SNS all'indirizzo https://console.aws.amazon.com/sns/v3/home](https://console.aws.amazon.com/sns/v3/home).
2. Dal pannello di navigazione, seleziona Argomenti, quindi seleziona Crea argomento.
3. Nella sezione Crea argomento, seleziona Standard. Quindi, inserisci il nome di un argomento, ad esempio **Inspector_to_Slack**. Altri dettagli sono facoltativi. Scegli Crea argomento per completare la creazione dell'endpoint.

Configurazione di un Amazon Q Developer nel client di applicazioni di chat

1. Accedi alla console delle applicazioni di chat di Amazon Q Developer all'indirizzo <https://console.aws.amazon.com/chatbot/>.
2. Dal riquadro Client configurati, seleziona Configura nuovo client.
3. Scegli Slack, quindi scegli Configura per confermare.

 Note

Quando scegli Slack, devi confermare le autorizzazioni per Amazon Q Developer nelle applicazioni di chat per accedere al tuo canale selezionando consenti.

4. Seleziona Configura un nuovo canale per aprire il riquadro dei dettagli di configurazione.
 - a. Inserisci un nome per il canale.
 - b. Per il canale Slack, scegli il canale che desideri utilizzare.
 - c. In Slack, copia l'ID del canale privato facendo clic con il pulsante destro del mouse sul nome del canale e selezionando Copia collegamento.
 - d. Nella finestra delle Console di gestione AWS applicazioni di chat di Amazon Q Developer, incolla l'ID del canale che hai copiato da Slack nel campo ID canale privato.
 - e. In Autorizzazioni, scegli di creare un ruolo IAM utilizzando un modello se non disponi già di un ruolo.
 - f. Per i modelli di policy, scegli Autorizzazioni di notifica. Questo è il modello di policy IAM per Amazon Q Developer nelle applicazioni di chat. Questa politica fornisce le autorizzazioni di lettura ed elenco necessarie per CloudWatch allarmi, eventi e registri e per gli argomenti di Amazon SNS.
 - g. Per le politiche Channel Guardrail, scegli 2. AmazonInspector ReadOnlyAccess
 - h. Scegli la regione in cui hai precedentemente creato l'argomento SNS, quindi seleziona l'argomento Amazon SNS che hai creato per inviare notifiche al canale Slack.
5. Selezionare Configura.

Amazon Chime setup

Creazione di un argomento SNS

1. [Accedi alla console Amazon SNS all'indirizzo https://console.aws.amazon.com/sns/v3/home](https://console.aws.amazon.com/sns/v3/home).
2. Seleziona Argomenti dal riquadro di navigazione, quindi seleziona Crea argomento.
3. Nella sezione Crea argomento, seleziona Standard. Quindi, inserisci il nome di un argomento, ad esempio **Inspector_to_Chime**. Altri dettagli sono facoltativi. Scegli Crea argomento per completare.

Configurazione di un Amazon Q Developer nel client di applicazioni di chat


1. Accedi alla console delle applicazioni di chat di Amazon Q Developer all'indirizzo <https://console.aws.amazon.com/chatbot/>.
2. Dal pannello Client configurati, seleziona Configura nuovo client.
3. Scegli Chime, quindi scegli Configura per confermare.

4. Dal riquadro Dettagli di configurazione, inserisci un nome per il canale.
5. In Amazon Chime, apri la chat room desiderata.
 - a. Seleziona l'icona a forma di ingranaggio nell'angolo in alto a destra e scegli Manage webhooks and bots (Gestisci webhook e bot).
 - b. Seleziona Copia URL per copiare l'URL del webhook negli appunti.
6. Nella finestra delle Console di gestione AWS applicazioni di chat di Amazon Q Developer, incolla l'URL che hai copiato nel campo URL Webhook.
7. In Autorizzazioni, scegli di creare un ruolo IAM utilizzando un modello se non disponi già di un ruolo.
8. Per i modelli di policy, scegli Autorizzazioni di notifica. Questo è il modello di policy IAM per Amazon Q Developer nelle applicazioni di chat. Fornisce le autorizzazioni di lettura ed elenco necessarie per CloudWatch allarmi, eventi e registri e per gli argomenti di Amazon SNS.
9. Scegli la regione in cui hai precedentemente creato l'argomento SNS, quindi seleziona l'argomento Amazon SNS che hai creato per inviare notifiche alla sala Amazon Chime.
10. Selezionare Configura.

Passaggio 2. Crea una EventBridge regola per i risultati di Amazon Inspector

1. Accedi utilizzando le tue credenziali.
2. Apri la EventBridge console Amazon all'indirizzo <https://console.aws.amazon.com/events/>.
3. Seleziona Regole dal riquadro di navigazione, quindi seleziona Crea regola.
4. Inserisci un nome e una descrizione facoltativa per la regola.
5. Seleziona Regola con uno schema di eventi, quindi Avanti.
6. Nel riquadro Event Pattern, scegli Modelli personalizzati (editor JSON).
7. Incolla il seguente JSON nell'editor.

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

 Note

Questo pattern invia notifiche per qualsiasi rilevazione attiva CRITICAL o di HIGH gravità rilevata da Amazon Inspector.

Seleziona Avanti quando hai finito di inserire lo schema dell'evento.

8. Nella pagina Seleziona obiettivi, scegli Servizio AWS. Quindi, per Seleziona il tipo di destinazione, scegli l'argomento SNS.
9. Per Argomento, seleziona il nome dell'argomento SNS che hai creato nel passaggio 1. Quindi scegli Successivo.
10. Aggiungi tag opzionali se necessario e scegli Avanti.
11. Rivedi la regola, quindi scegli Crea regola.

EventBridge per ambienti con più account Amazon Inspector

Se sei un amministratore delegato di Amazon Inspector, EventBridge le regole vengono visualizzate sul tuo account in base ai risultati applicabili dei tuoi account membro. Se configuri le notifiche relative ai risultati tramite EventBridge il tuo account amministratore, come descritto nella sezione precedente, riceverai notifiche relative a più account. In altre parole, riceverai una notifica dei risultati e degli eventi generati dai tuoi account membro oltre a quelli generati dal tuo account.

Puoi utilizzare i `accountId` dettagli JSON del risultato per identificare l'account membro da cui ha avuto origine il risultato di Amazon Inspector.

Utilizzo del pannello di controllo in Amazon Inspector

La dashboard fornisce un'istantanea delle statistiche aggregate per le risorse scansionate da Amazon Inspector. Utilizza la dashboard per conoscere la copertura del tuo ambiente e i risultati critici.

Note

Se il tuo account è l'account amministratore delegato di un'organizzazione, la dashboard mostra le informazioni relative al tuo account e a tutti gli altri account dell'organizzazione.

Questo argomento descrive come visualizzare la dashboard e comprendere i componenti che la compongono.

Argomenti

- [Visualizzazione del pannello di controllo](#)
- [Comprensione dei componenti del dashboard e interpretazione dei dati](#)

Visualizzazione del pannello di controllo

La dashboard mostra una panoramica della copertura per l'ambiente in uso e dei risultati critici. La dashboard aggiorna automaticamente i dati ogni cinque minuti. Puoi aggiornare i dati manualmente scegliendo l'icona di aggiornamento nell'angolo in alto a destra dello schermo. È possibile visualizzare i dati di supporto per un elemento selezionando l'elemento.

Note

Se il tuo account è l'account amministratore delegato di un'organizzazione, puoi visualizzare le statistiche aggregate relative a un account membro inserendo l'ID dell'account membro nel campo Account.

Per visualizzare la dashboard:

1. [Accedi utilizzando le tue credenziali, quindi apri la console `https://console.aws.amazon.com/inspector/` Amazon Inspector su `v2/home`.](https://console.aws.amazon.com/inspector/)

2. Dal pannello di navigazione, scegli Dashboard.

Comprensione dei componenti del dashboard e interpretazione dei dati

Ogni sezione del pannello di controllo fornisce informazioni dettagliate sulle metriche e sui risultati chiave, in modo da poter comprendere il livello di vulnerabilità delle AWS risorse esistenti. Regione AWS

Copertura ambientale

La sezione Copertura ambientale fornisce statistiche sulle risorse analizzate da Amazon Inspector. In questa sezione, puoi visualizzare il numero e la percentuale di EC2 istanze Amazon, immagini e AWS Lambda funzioni Amazon ECR scansionate da Amazon Inspector. Se gestisci più account in AWS Organizations qualità di amministratore delegato di Amazon Inspector, vedrai anche il numero totale di account dell'organizzazione, il numero con Amazon Inspector attivato e la percentuale di copertura risultante per l'organizzazione. Puoi anche utilizzare questa sezione per determinare quali risorse non sono coperte da Amazon Inspector. Queste risorse possono contenere vulnerabilità che potrebbero essere sfruttate per mettere a rischio la tua organizzazione. Per ulteriori dettagli, consulta [Valutazione della copertura di Amazon Inspector del tuo ambiente AWS](#).

La scelta di un gruppo di copertura porta alla pagina di gestione dell'account relativa al raggruppamento selezionato. La pagina di gestione degli account mostra i dettagli su quali account, EC2 istanze Amazon e repository Amazon ECR sono coperti da Amazon Inspector.

Sono disponibili i seguenti gruppi di copertura:

- Account
- Istanze
- Archivi di contenitori
- Immagini di container
- Lambda

Risultati critici

La sezione Risultati critici fornisce un conteggio delle vulnerabilità critiche nell'ambiente e un conteggio totale di tutti i risultati presenti nell'ambiente. In questa sezione, i conteggi sono mostrati

per risorsa e tipo di valutazione. Per ulteriori informazioni sui risultati critici e su come Amazon Inspector determina la criticità, consulta. [Comprendere i risultati di Amazon Inspector](#)

La scelta di un gruppo di risultati critici ti porta alla pagina Tutti i risultati e applica automaticamente i filtri per mostrare tutti i risultati critici che corrispondono al raggruppamento selezionato.

Sono disponibili i seguenti gruppi di risultati critici:

- Risultati della scansione del codice di Amazon Inspector
- Risultati delle EC2 istanze Amazon
- Risultati delle immagini dei container Amazon ECR
- Risultati della funzione Lambda

Correzioni basate sul rischio

La sezione Correzioni basate sul rischio mostra i cinque principali pacchetti software con vulnerabilità critiche che interessano la maggior parte delle risorse dell'ambiente. La correzione di questi pacchetti può ridurre in modo significativo il numero di rischi critici per l'ambiente. Scegliete il nome del pacchetto software per visualizzare i dettagli delle vulnerabilità associate e le risorse interessate.

Account con i risultati più critici

La sezione Account con i risultati più critici mostra i primi cinque AWS account dell'ambiente con i risultati più critici e il numero totale di risultati per quell'account. Questa sezione è visualizzabile dall'account amministratore delegato solo se Amazon Inspector è configurato per la scansione di più account con. AWS Organizations Questa visualizzazione aiuta gli amministratori delegati a capire quali account possono essere maggiormente a rischio all'interno dell'organizzazione.

Scegli Account ID per visualizzare ulteriori informazioni sull'account membro interessato.

Repository Amazon ECR con i risultati più critici

La sezione Repositories Elastic Container Registry (ECR) con i risultati più critici mostra i cinque principali repository Amazon ECR del tuo ambiente con i risultati più critici relativi alle immagini dei container. La vista mostra il nome del repository, l'identificatore AWS dell'account, la data di creazione del repository, il numero di vulnerabilità critiche e il numero totale di vulnerabilità. Questa visualizzazione consente di identificare i repository più a rischio.

Scegli il nome del repository per visualizzare ulteriori informazioni sul repository interessato.

Immagini dei container con i risultati più critici

La sezione Immagini dei container con i risultati più critici mostra le prime cinque immagini dei container presenti nell'ambiente con i risultati più critici. La visualizzazione mostra i dati dei tag di immagine, il nome del repository, l'immagine digest, l'identificatore AWS dell'account, il numero di vulnerabilità critiche e il numero totale di vulnerabilità. Questa visualizzazione aiuta i proprietari delle applicazioni a identificare quali immagini del contenitore potrebbero dover essere ricostruite e riavviate.

Scegliete Immagine del contenitore per visualizzare ulteriori informazioni sull'immagine del contenitore interessata.

Istanze con i risultati più critici

La sezione Istanze con i risultati più critici mostra le prime cinque EC2 istanze Amazon con i risultati più critici. La vista mostra l'identificatore dell'istanza, l'identificatore AWS dell'account, l'identificatore Amazon Machine Image (AMI), il numero di vulnerabilità critiche e il numero totale di vulnerabilità. Questa visualizzazione aiuta i proprietari dell'infrastruttura a identificare quali istanze potrebbero richiedere l'applicazione di patch.

Scegli Instance ID per visualizzare ulteriori informazioni sull' EC2 istanza Amazon interessata.

Amazon Machine Images (AMI) con i risultati più critici

La sezione Amazon Machine Images (AMIs) con i risultati più critici mostra i primi cinque AMIs risultati del tuo ambiente con i risultati più critici. La visualizzazione mostra l'identificatore AMI, l'identificatore dell' AWS account, il numero di EC2 istanze interessate in esecuzione nell'ambiente, la data di creazione dell'AMI, la piattaforma del sistema operativo dell'AMI, il numero di vulnerabilità critiche e il numero totale di vulnerabilità. Questa visualizzazione aiuta i proprietari dell'infrastruttura a identificare quali potrebbero richiedere la ricostruzione. AMIs

Scegli Istanze interessate per visualizzare ulteriori informazioni sulle istanze avviate dall'AMI interessata.

AWS Lambda funzioni con i risultati più critici

La sezione AWS Lambda Funzioni con i risultati più critici mostra le cinque funzioni Lambda principali dell'ambiente con i risultati più critici. La vista mostra il nome della funzione Lambda, l'identificatore dell' AWS account, l'ambiente di runtime, il numero di vulnerabilità critiche, il numero di vulnerabilità elevate e il numero totale di vulnerabilità. Questa visualizzazione aiuta i proprietari dell'infrastruttura a identificare quali funzioni Lambda potrebbero richiedere una correzione.

Scegli il nome della funzione per visualizzare ulteriori informazioni sulla funzione interessata AWS Lambda .

Scansione del codice di Amazon Inspector con i risultati più critici

La sezione Progetti con le vulnerabilità del codice più critiche mostra i primi cinque progetti con risultati critici. Puoi scegliere un progetto per visualizzare i dettagli sui risultati. Quando scegli un progetto, vieni indirizzato al repository in cui si trovano i risultati. La scheda dei risultati mostra i nomi dei risultati e i relativi livelli di gravità. Mostra il tipo di analisi utilizzata per generare i risultati. Mostra anche quanti anni hanno i risultati e il loro stato.

Ricerca nel database delle vulnerabilità di Amazon Inspector

Puoi cercare vulnerabilità ed esposizioni comuni (CVE) nel database delle vulnerabilità di Amazon Inspector. Amazon Inspector utilizza le informazioni del database delle vulnerabilità per produrre dettagli relativi a un ID CVE. Puoi visualizzare questi dettagli nella schermata dei dettagli CVE. Amazon Inspector monitora e fornisce informazioni sulle [vulnerabilità](#) del software nel database delle vulnerabilità. Amazon Inspector supporta solo CVEs le piattaforme elencate nella sezione Piattaforme di rilevamento della schermata dei dettagli CVE. Questa sezione descrive come effettuare ricerche nel database delle vulnerabilità di Amazon Inspector utilizzando un ID CVE.

Note

Al momento, la ricerca CVE non supporta. Microsoft Windows

Ricerca nel database delle vulnerabilità

Questa sezione descrive come cercare nel database delle vulnerabilità nella console e con l'API Amazon Inspector.

Note

È necessario attivare Amazon Inspector nella versione corrente Regione AWS prima di poter effettuare ricerche nel database delle vulnerabilità.

Console

1. [Accedi utilizzando le tue credenziali, quindi apri la console Amazon Inspector su v2/home https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Vulnerability database search.
3. Nella barra di ricerca, inserisci un ID CVE e scegli Cerca.

API

Esegui l'[SearchVulnerabilities](#) API Amazon Inspector e fornisci un singolo ID CVE `filterCriteria` nel seguente formato: CVE-<year>-<ID>

Comprendere i dettagli del CVE

Questa sezione descrive come interpretare la pagina dei dettagli CVE.

Dettagli CVE

La sezione dei dettagli CVE include le seguenti informazioni:

- Descrizione e ID CVE
- Severità CVE
- Punteggi del Common Vulnerability Scoring System (CVSS) e dell'Exploit Prediction Scoring System (EPSS)
- Piattaforme di rilevamento

Note

Se questo campo è vuoto, Amazon Inspector non supporta il rilevamento del tuo ID CVE.

- Common Weakness Enumeration (CWE)
- Date di creazione e aggiornamento del fornitore

Intelligence sulle vulnerabilità

La sezione sull'intelligence sulle vulnerabilità fornisce dati di intelligence sulle minacce, come gli obiettivi degli exploit e la data dell'ultimo exploit pubblico nota.

Fornisce inoltre i dati della Cybersecurity and Infrastructure Security Agency (CISA), che includono l'azione di correzione, la data in cui il CVE è stato aggiunto al catalogo Known Exploited Vulnerability e la data e l'ora in cui CISA si aspetta che le agenzie federali risolvano il CVE.

Riferimenti

La sezione dei riferimenti fornisce collegamenti a risorse per ulteriori informazioni sul CVE.

Esportazione SBOMs con Amazon Inspector

Una distinta base del software (SBOM) è un inventario annidato di tutti i componenti software open source e di terze parti presenti nella codebase. Amazon Inspector fornisce SBOMs risorse individuali nel tuo ambiente. Puoi utilizzare la console Amazon Inspector o l'API Amazon Inspector per SBOMs generare per le tue risorse. Puoi esportare tutte SBOMs le risorse supportate e monitorate da Amazon Inspector. Exported SBOMs fornisce informazioni sulla fornitura di software. È possibile verificare lo stato delle risorse [valutando la copertura dell'ambiente](#). AWS Questa sezione descrive come configurare ed esportare SBOMs.

Alcuni componenti software e gestori di pacchetti utilizzano intervalli di versioni o riferimenti dinamici anziché versioni fisse per le dipendenze. Questa pratica crea hash irrisolti, in cui Amazon Inspector identifica un file hash o jar ma non può mapparli a un nome e una versione specifici per il rilevamento delle vulnerabilità. Amazon Inspector ora include questi hash irrisolti nelle esportazioni di Software Bill of Materials (SBOM). Sebbene questi pacchetti non possano essere scansionati alla ricerca di vulnerabilità, i loro valori hash sono disponibili nell'elenco dei componenti esportati.

Tip

A partire dal 28 febbraio 2026, Amazon Inspector ora supporta l'esportazione di SBOM per istanze Amazon EC2 Windows senza agenti. L'esportazione di SBOM per istanze Windows Amazon EC2 basate su agenti non è attualmente supportata.

Formati Amazon Inspector

Amazon Inspector supporta l'esportazione SBOMs in formati compatibili con CycloneDX 1.4 e SPDX 2.3. Amazon Inspector esporta SBOMs come JSON file nel bucket Amazon S3 scelto dall'utente.

Note

Le esportazioni in formato SPDX da Amazon Inspector sono compatibili con i sistemi che utilizzano SPDX 2.3, tuttavia non contengono il campo Creative Commons Zero (). CC0 Questo perché l'inclusione di questo campo consentirebbe agli utenti di ridistribuire o modificare il materiale.

Esempio di formato SBOM CyclonedX 1.4 di Amazon Inspector

```

    {
  "bomFormat": "CycloneDX",
  "specVersion": "1.4",
  "version": 1,
  "metadata": {
    "timestamp": "2023-06-02T01:17:46Z",
    "component": null,
    "properties": [
      {
        "name": "imageId",
        "value":
"sha256:c8ee97f7052776ef223080741f61fcdf6a3a9107810ea9649f904aa4269fdac6"
      },
      {
        "name": "architecture",
        "value": "arm64"
      },
      {
        "name": "accountId",
        "value": "111122223333"
      },
      {
        "name": "resourceType",
        "value": "AWS_ECR_CONTAINER_IMAGE"
      }
    ]
  },
  "components": [
    {
      "type": "library",
      "name": "pip",
      "purl": "pkg:pypi/pip@22.0.4?path=usr/local/lib/python3.8/site-packages/
pip-22.0.4.dist-info/METADATA",
      "bom-ref": "98dc550d1e9a0b24161daaa0d535c699"
    },
    {
      "type": "application",
      "name": "libss2",
      "purl": "pkg:dpkg/libss2@1.44.5-1+deb10u3?
arch=ARM64&epoch=0&upstream=libss2-1.44.5-1+deb10u3.src.dpkg",

```

```

    "bom-ref": "2f4d199d4ef9e2ae639b4f8d04a813a2"
  },
  {
    "type": "application",
    "name": "liblz4-1",
    "purl": "pkg:dpkg/liblz4-1@1.8.3-1+deb10u1?
arch=ARM64&epoch=0&upstream=liblz4-1-1.8.3-1+deb10u1.src.dpkg",
    "bom-ref": "9a6be8907ead891b070e60f5a7b7aa9a"
  },
  {
    "type": "application",
    "name": "mawk",
    "purl": "pkg:dpkg/mawk@1.3.3-17+b3?
arch=ARM64&epoch=0&upstream=mawk-1.3.3-17+b3.src.dpkg",
    "bom-ref": "c2015852a729f97fde924e62a16f78a5"
  },
  {
    "type": "application",
    "name": "libgmp10",
    "purl": "pkg:dpkg/libgmp10@6.1.2+dfsg-4+deb10u1?
arch=ARM64&epoch=2&upstream=libgmp10-6.1.2+dfsg-4+deb10u1.src.dpkg",
    "bom-ref": "52907290f5beef00dff8da77901b1085"
  },
  {
    "type": "application",
    "name": "ncurses-bin",
    "purl": "pkg:dpkg/ncurses-bin@6.1+20181013-2+deb10u3?
arch=ARM64&epoch=0&upstream=ncurses-bin-6.1+20181013-2+deb10u3.src.dpkg",
    "bom-ref": "cd20cfb9ebeeada3809764376f43bce"
  }
],
"vulnerabilities": [
  {
    "id": "CVE-2022-40897",
    "affects": [
      {
        "ref": "a74a4862cc654a2520ec56da0c81cdb3"
      },
      {
        "ref": "0119eb286405d780dc437e7dbf2f9d9d"
      }
    ]
  }
]
]

```

```
}

```

Esempio di formato SBOM SPDX 2.3 di Amazon Inspector

```
{
  "name": "409870544328/EC2/i-022fba820db137c64/ami-074ea14c08effb2d8",
  "spdxVersion": "SPDX-2.3",
  "creationInfo": {
    "created": "2023-06-02T21:19:22Z",
    "creators": [
      "Organization: 409870544328",
      "Tool: Amazon Inspector SBOM Generator"
    ]
  },
  "documentNamespace": "EC2://i-022fba820db137c64/AMAZON_LINUX_2/null/x86_64",
  "comment": "",
  "packages": [{
    "name": "elfutils-libelf",
    "versionInfo": "0.176-2.amzn2",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
      "referenceCategory": "PACKAGE-MANAGER",
      "referenceType": "purl",
      "referenceLocator": "pkg:rpm/elfutils-libelf@0.176-2.amzn2?
arch=X86_64&epoch=0&upstream=elfutils-libelf-0.176-2.amzn2.src.rpm"
    }],
    "SPDXID": "SPDXRef-Package-rpm-elfutils-libelf-ddf56a513c0e76ab2ae3246d9a91c463"
  },
  {
    "name": "libcurl",
    "versionInfo": "7.79.1-1.amzn2.0.1",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
      "referenceCategory": "PACKAGE-MANAGER",
      "referenceType": "purl",
      "referenceLocator": "pkg:rpm/libcurl@7.79.1-1.amzn2.0.1?
arch=X86_64&epoch=0&upstream=libcurl-7.79.1-1.amzn2.0.1.src.rpm"
    }
  ]
}
```

```

    },
    {
      "referenceCategory": "SECURITY",
      "referenceType": "vulnerability",
      "referenceLocator": "CVE-2022-32205"
    }
  ],
  "SPDXID": "SPDXRef-Package-rpm-libcurl-710fb33829bc5106559bcd380cddb7d5"
},
{
  "name": "hunspell-en-US",
  "versionInfo": "0.20121024-6.amzn2.0.1",
  "downloadLocation": "NOASSERTION",
  "sourceInfo": "/var/lib/rpm/Packages",
  "filesAnalyzed": false,
  "externalRefs": [{
    "referenceCategory": "PACKAGE-MANAGER",
    "referenceType": "purl",
    "referenceLocator": "pkg:rpm/hunspell-en-US@0.20121024-6.amzn2.0.1?
arch=NOARCH&epoch=0&upstream=hunspell-en-US-0.20121024-6.amzn2.0.1.src.rpm"
  }],
  "SPDXID": "SPDXRef-Package-rpm-hunspell-en-US-de19ae0883973d6cea5e7e079d544fe5"
},
{
  "name": "grub2-tools-minimal",
  "versionInfo": "2.06-2.amzn2.0.6",
  "downloadLocation": "NOASSERTION",
  "sourceInfo": "/var/lib/rpm/Packages",
  "filesAnalyzed": false,
  "externalRefs": [{
    "referenceCategory": "PACKAGE-MANAGER",
    "referenceType": "purl",
    "referenceLocator": "pkg:rpm/grub2-tools-minimal@2.06-2.amzn2.0.6?
arch=X86_64&epoch=1&upstream=grub2-tools-minimal-2.06-2.amzn2.0.6.src.rpm"
  }],
  {
    "referenceCategory": "SECURITY",
    "referenceType": "vulnerability",
    "referenceLocator": "CVE-2021-3981"
  }
  ],
  "SPDXID": "SPDXRef-Package-rpm-grub2-tools-minimal-c56b7ea76e5a28ab8f232ef6d7564636"
},
{

```

```

    "name": "unixODBC-devel",
    "versionInfo": "2.3.1-14.amzn2",
    "downloadLocation": "NOASSERTION",
    "sourceInfo": "/var/lib/rpm/Packages",
    "filesAnalyzed": false,
    "externalRefs": [{
      "referenceCategory": "PACKAGE-MANAGER",
      "referenceType": "purl",
      "referenceLocator": "pkg:rpm/unixODBC-devel@2.3.1-14.amzn2?
arch=X86_64&epoch=0&upstream=unixODBC-devel-2.3.1-14.amzn2.src.rpm"
    }],
    "SPDXID": "SPDXRef-Package-rpm-unixODBC-devel-1bb35add92978df021a13fc9f81237d2"
  }
],
"relationships": [{
  "spdxElementId": "SPDXRef-DOCUMENT",
  "relatedSpdxElement": "SPDXRef-Package-rpm-elfutils-libelf-
ddf56a513c0e76ab2ae3246d9a91c463",
  "relationshipType": "DESCRIBES"
},
{
  "spdxElementId": "SPDXRef-DOCUMENT",
  "relatedSpdxElement": "SPDXRef-Package-rpm-yajl-8476ce2db98b28cfab2b4484f84f1903",
  "relationshipType": "DESCRIBES"
},
{
  "spdxElementId": "SPDXRef-DOCUMENT",
  "relatedSpdxElement": "SPDXRef-Package-rpm-unixODBC-
devel-1bb35add92978df021a13fc9f81237d2",
  "relationshipType": "DESCRIBES"
}
],
"SPDXID": "SPDXRef-DOCUMENT"
}

```

Filtri per SBOMs

Quando esporti, SBOMs puoi includere filtri per creare report per sottoinsiemi specifici di risorse. Se non fornisci un filtro, vengono esportate tutte SBOMs le risorse attive e supportate. E se sei un amministratore delegato, questo include anche risorse per tutti i membri. I filtri disponibili sono:

- AccountID: questo filtro può essere utilizzato SBOMs per esportare qualsiasi risorsa associata a un ID account specifico.
- Tag di istanza EC2: questo filtro può essere utilizzato per esportare SBOM per istanze EC2 con tag specifici.
- Nome funzione: questo filtro può essere utilizzato SBOMs per esportare funzioni Lambda specifiche.
- Tag immagine: questo filtro può essere utilizzato SBOMs per esportare immagini di contenitori con tag specifici.
- Tag funzione Lambda: questo filtro può essere utilizzato per esportare funzioni SBOMs Lambda con tag specifici.
- Tipo di risorsa: questo filtro può essere utilizzato per filtrare il tipo di risorsa: EC2/ECR/Lambda.
- ID risorsa: questo filtro può essere utilizzato per esportare un SBOM per una risorsa specifica.
- Nome del repository: questo filtro può essere utilizzato SBOMs per generare immagini di contenitori in repository specifici.

Configura ed esporta SBOMs

Per esportare SBOMs, devi prima configurare un bucket Amazon S3 e una AWS KMS chiave che Amazon Inspector possa utilizzare. Puoi utilizzare i filtri SBOMs per esportare sottoinsiemi specifici delle tue risorse. SBOMs Per esportare per più account in un' AWS organizzazione, segui questi passaggi dopo aver effettuato l'accesso come amministratore delegato di Amazon Inspector.

Prerequisiti

- Risorse supportate che vengono monitorate attivamente da Amazon Inspector.
- Un bucket Amazon S3 configurato con una policy che consente ad Amazon Inspector di aggiungere oggetti. [Per informazioni sulla configurazione della policy, consulta Configurare le autorizzazioni di esportazione.](#)
- Una AWS KMS chiave configurata con una politica che consente di utilizzare Amazon Inspector per crittografare i report. Per informazioni sulla configurazione della politica, consulta [Configurare una AWS KMS chiave](#) per l'esportazione.

Note

Se in precedenza hai configurato un bucket Amazon S3 e una AWS KMS chiave per l'[esportazione dei risultati](#), puoi utilizzare lo stesso bucket e la stessa chiave per l'esportazione SBOM.

Scegli il tuo metodo di accesso preferito per esportare un SBOM.

Console

1. [Accedi utilizzando le tue credenziali, quindi apri la console `https://console.aws.amazon.com/inspector/AmazonInspector/v2/home`.](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Utilizzando il Regione AWS selettore nell'angolo superiore destro della pagina, seleziona la regione con le risorse per cui desideri esportare SBOM.
3. Nel pannello di navigazione, scegli Esporta. SBOMs
4. (Facoltativo) Nella SBOMs pagina Esporta, utilizzate il menu Aggiungi filtro per selezionare un sottoinsieme di risorse per cui creare report. Se non viene fornito alcun filtro, Amazon Inspector esporterà i report per tutte le risorse attive. Se sei un amministratore delegato, questo includerà tutte le risorse attive della tua organizzazione.
5. In Impostazioni di esportazione selezionate il formato desiderato per la SBOM.
6. Inserisci un URI Amazon S3 o scegli Browse Amazon S3 per selezionare una posizione Amazon S3 in cui archiviare la SBOM.
7. Inserisci una AWS KMS chiave configurata per Amazon Inspector da utilizzare per crittografare i report.

API

- SBOMs Per esportare le tue risorse in modo programmatico, utilizza il [CreateSbomExport](#) funzionamento dell'API Amazon Inspector.

Nella tua richiesta, utilizza il `reportFormat` parametro per specificare il formato di output SBOM, scegli o. `CYCLONEDX_1_4` `SPDX_2_3` Il `s3Destination` parametro è obbligatorio ed è necessario specificare un bucket S3 configurato con una policy che consenta ad Amazon Inspector di scrivere su di esso. Facoltativamente, utilizza `resourceFilterCriteria` i parametri per limitare l'ambito del report a risorse specifiche.

AWS CLI

- SBOMs Per esportare le tue risorse usando il AWS Command Line Interface comando seguente:

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

Nella richiesta, *FORMAT* sostituiscilo con il formato che preferisci, CYCLONEDX_1_4 oppureSPDX_2_3. Quindi sostituisci *user input placeholders* for the s3 destination con il nome del bucket S3 in cui esportare, il prefisso da usare per l'output in S3 e l'ARN per la chiave KMS che stai utilizzando per crittografare i report.

Schema di EventBridge eventi Amazon per gli eventi Amazon Inspector

[Amazon EventBridge](#) fornisce un flusso di dati in tempo reale da applicazioni e altro Servizi AWS alle destinazioni, come AWS Lambda funzioni, argomenti di Amazon Simple Notification Service e flussi di dati in Amazon Kinesis Data Streams. [Per supportare l'integrazione con altre applicazioni, servizi e sistemi, Amazon Inspector pubblica automaticamente i risultati su come eventi. EventBridge](#) Puoi utilizzare Amazon Inspector per pubblicare eventi per risultati, copertura e scansioni. Questa sezione fornisce schemi di esempio per gli eventi. EventBridge

Argomenti

- [Schema EventBridge di base Amazon per Amazon Inspector](#)
- [Esempio di schema di eventi di ricerca di Amazon Inspector](#)
- [Esempio di schema di eventi completo per la scansione iniziale di Amazon Inspector](#)
- [Esempio di schema degli eventi di copertura di Amazon Inspector](#)
- [Esempio di schema di attivazione automatica di Amazon Inspector](#)

Schema EventBridge di base Amazon per Amazon Inspector

Di seguito è riportato un esempio dello schema di base di un EventBridge evento per Amazon Inspector. I dettagli dell'evento variano in base al tipo di evento.

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

```
}
```

Esempio di schema di eventi di ricerca di Amazon Inspector

Di seguito sono inclusi esempi dello schema di un EventBridge evento per i risultati di Amazon Inspector. Gli eventi di ricerca vengono creati quando Amazon Inspector identifica una vulnerabilità del software o un problema di rete in una delle tue risorse. Per una guida alla creazione di notifiche in risposta a questo tipo di evento, consulta [Creazione di risposte personalizzate ai risultati di Amazon Inspector con Amazon EventBridge](#)

I seguenti campi identificano un evento di ricerca:

- `detail-type` è impostato su `Inspector2 Finding`.
- `detail` descrive il risultato.
- `detail.resources.tags` è dove vengono archiviati i dati chiave-valore.

Puoi filtrare le schede per visualizzare la ricerca di schemi di eventi per diverse risorse e tipi di ricerca.

Amazon EC2 package vulnerability finding

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

```

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

```

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

```

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

```

Amazon EC2 network reachability finding

```

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

```

```

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

```

```

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

```

Amazon ECR package vulnerability finding

```

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

```

```
    "relatedVulnerabilities": [
      "USN-6986-1"
    ],
    "source": "UBUNTU_CVE",
    "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2024/
CVE-2024-6119.html",
    "vendorCreatedAt": "Tue Sep 03 00:00:00.000 UTC 2024",
    "vendorSeverity": "medium",
    "vulnerabilityId": "CVE-2024-6119",
    "vulnerablePackages": [
      {
        "arch": "ARM64",
        "epoch": 0,
        "fixedInVersion": "0:3.0.13-0ubuntu3.4",
        "name": "libssl3t64",
        "packageManager": "OS",
        "release": "0ubuntu3.2",
        "remediation": "apt-get update && apt-get upgrade",
        "sourceLayerHash":
"sha256:1567e7ea90b67fc95ccdeec39bdc3045098dee7e0c604975b957a9f8c0e9616",
        "version": "3.0.13"
      },
      {
        "arch": "ARM64",
        "epoch": 0,
        "fixedInVersion": "0:3.0.13-0ubuntu3.4",
        "name": "openssl",
        "packageManager": "OS",
        "release": "0ubuntu3.2",
        "remediation": "apt-get update && apt-get upgrade",
        "sourceLayerHash":
"sha256:1567e7ea90b67fc95ccdeec39bdc3045098dee7e0c604975b957a9f8c0e9616",
        "version": "3.0.13"
      }
    ]
  },
  "remediation": {
    "recommendation": {
      "text": "None Provided"
    }
  },
  "resources": [
    {
      "details": {
```

```

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

```

Lambda package vulnerability finding

```

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

```

```

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

```

```

        "version": "3.1"
      }
    ],
    "referenceUrls": [
      "https://www.debian.org/security/2023/dsa-5442",
      "https://lists.debian.org/debian-lts-announce/2023/08/msg00024.html"
    ],
    "relatedVulnerabilities": [],
    "source": "NVD",
    "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2023-30861",
    "vendorCreatedAt": "Tue May 02 18:15:52.000 UTC 2023",
    "vendorSeverity": "HIGH",
    "vendorUpdatedAt": "Sun Aug 20 21:15:09.000 UTC 2023",
    "vulnerabilityId": "CVE-2023-30861",
    "vulnerablePackages": [
      {
        "epoch": 0,
        "filePath": "requirements.txt",
        "fixedInVersion": "2.3.2",
        "name": "flask",
        "packageManager": "PIP",
        "version": "2.0.0"
      }
    ]
  },
  "remediation": {
    "recommendation": {
      "text": "None Provided"
    }
  },
  "resources": [
    {
      "details": {
        "awsLambdaFunction": {
          "architectures": [
            "X86_64"
          ],
          "codeSha256": "07jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
          "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
          "functionName": "VulnerableFunction",
          "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
          "packageType": "ZIP",

```

```

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

```

Lambda code vulnerability finding

```

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

```

```

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

```

```

        "codeSha256": "07jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
        "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
        "functionName": "VulnerableFunction",
        "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
        "packageType": "ZIP",
        "runtime": "PYTHON_3_11",
        "version": "$LATEST"
    }
},
    "id": "arn:aws:lambda:eu-
central-1:123456789012:function:VulnerableFunction:$LATEST",
    "partition": "aws",
    "region": "eu-central-1",
    "type": "AWS_LAMBDA_FUNCTION"
}
],
"severity": "CRITICAL",
"status": "ACTIVE",
"title": "CWE-798 - Hardcoded credentials",
"type": "CODE_VULNERABILITY",
"updatedAt": "Wed Sep 04 16:51:01.869 UTC 2024"
}
}

```

Note

Il valore di dettaglio restituisce i dettagli JSON di un singolo risultato come oggetto. Non restituisce l'intera sintassi di risposta ai risultati, che supporta più risultati all'interno di un array.

Esempio di schema di eventi completo per la scansione iniziale di Amazon Inspector

Di seguito è riportato un esempio dello schema di eventi per un EventBridge evento Amazon Inspector per il completamento di una scansione iniziale. Questo evento viene creato quando Amazon Inspector completa una scansione iniziale di una delle tue risorse.

I seguenti campi identificano un evento di completamento della scansione iniziale:

- Il `detail-type` campo è impostato su `Inspector2 Scan`.
- L'`detail` oggetto contiene un `finding-severity-counts` oggetto che descrive in dettaglio il numero di risultati nelle categorie di gravità applicabili, ad esempio `CRITICALHIGH`, `eMEDIUM`.

Seleziona una delle opzioni per visualizzare diversi schemi di eventi di scansione iniziale in base al tipo di risorsa.

Amazon EC2 instance initial scan

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

Amazon ECR image initial scan

```
{
  "version": "0",
```

```

    "id": "fdaa751a-984c-a709-44f9-9a9da9cd3606",
    "detail-type": "Inspector2 Scan",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-20T23:15:18Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:ecr:us-east-1:111122223333:repository/inspector2"
    ],
    "detail": {
      "scan-status": "INITIAL_SCAN_COMPLETE",
      "repository-name": "arn:aws:ecr:us-east-1:111122223333:repository/
inspector2",
      "finding-severity-counts": {
        "CRITICAL": 0,
        "HIGH": 0,
        "MEDIUM": 0,
        "TOTAL": 0
      },
      "image-digest":
"sha256:965fbcae990b0467ed5657caceaec165018ef44a4d2d46c7cdea80a9dff0d1ea",
      "image-tags": [
        "ubuntu22"
      ],
      "version": "1.0"
    }
  }
}

```

Lambda function initial scan

```

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

```

```

],
"detail": {
  "scan-status": "INITIAL_SCAN_COMPLETE",
  "finding-severity-counts": {
    "CRITICAL": 0,
    "HIGH": 0,
    "MEDIUM": 0,
    "TOTAL": 0
  },
  "version": "1.0"
}
}

```

Esempio di schema degli eventi di copertura di Amazon Inspector

Di seguito è riportato un esempio dello schema di eventi per un EventBridge evento Amazon Inspector per la copertura. Questo evento viene creato quando la copertura di scansione di Amazon Inspector per una risorsa viene modificata. I seguenti campi identificano un evento di copertura:

- Il `detail-type` campo è impostato su `Inspector2 Coverage`.
- L'`detail` oggetto contiene un `scanStatus` oggetto che indica il nuovo stato di scansione della risorsa.

```

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

```

```
        "statusCodeValue": "INACTIVE"
    },
    "scanType": "PACKAGE",
    "eventTimestamp": "2023-01-20T22:51:35.665501Z",
    "version": "1.0"
}
}
```

Esempio di schema di attivazione automatica di Amazon Inspector

L'evento di attivazione automatica viene inviato all'amministratore delegato quando Amazon Inspector non è in grado di supportare il numero di membri di un'organizzazione. I seguenti campi identificano un evento di attivazione automatica:

- Il `detail-type` campo è impostato su `Inspector2 AutoEnable`
- L'`detailoggetto` descrive il motivo per cui l'evento di attivazione automatica non è riuscito.

```
{
  "version": "0",
  "id": "85fc3613-e913-7fc4-a80c-a3753e4aa9ae",
  "detail-type": "Inspector2 AutoEnable",
  "source": "aws.inspector2",
  "account": "123456789012",
  "time": "2024-08-21T02:36:48Z",
  "region": "us-east-1",
  "detail": {
    "version": "1.0.0",
    "AutoEnableStatus": "Failed",
    "Reason": "The number of member accounts enabled with AWS Inspector has reached
the maximum limit of 10,000"
  }
}
```

Scanner VM Amazon Inspector

Panoramica di

Amazon Inspector utilizza Amazon Inspector VM Scanner per eseguire valutazioni delle vulnerabilità sulle istanze Amazon EC2. Amazon Inspector VM Scanner sfrutta le funzionalità di raccolta dell'inventario di Amazon [Inspector SBOM Generator per generare una Software Bill of Materials \(SBOM\)](#) e invia la SBOM per la valutazione tramite il canale di telemetria Inspector. [Quando abiliti Enhanced EC2 Scanning sul tuo account, Inspector VM Scanner sostituisce il plugin Inspector SSM.](#)

Amazon Inspector offre due approcci per l'implementazione di VM Scanner:

- **Installazione automatica (consigliata):** quando abiliti Enhanced EC2 Scanning nella console Amazon Inspector, Amazon Inspector utilizza Amazon EC2 Systems Manager (SSM) per installare e gestire automaticamente lo scanner VM sulle istanze Amazon EC2. Questo è l'approccio più semplice e non richiede alcun intervento manuale. Per ulteriori informazioni, consulta [Attivazione dello scanner VM Inspector](#).
- **Installazione manuale:** è possibile installare manualmente VM Scanner utilizzando gestori di pacchetti standard (RPM, DEB, APK, MSI, PKG). Questo approccio non richiede SSM. Per ulteriori informazioni, consulta [Installazione e configurazione manuali](#).

Vantaggi

Inspector VM Scanner offre i seguenti vantaggi rispetto al precedente plug-in Inspector SSM:

- **Utilizzo ridotto delle risorse:** richiede meno elaborazione durante il funzionamento
- **Raccolta più granulare dei pacchetti:** fornisce un inventario dettagliato a livello di pacchetto per una valutazione ad alta fedeltà
- **Meccanismo di scansione migliorato:** utilizza Inspector SBOM Generator per un comportamento coerente con altre risorse supportate da Inspector
- **Separazione dei compiti:** i team addetti alla sicurezza possono abilitare la scansione a livello di account, mentre gli amministratori delle istanze mantengono il controllo sull'installazione e la configurazione manuali sulle singole istanze

Attivazione dello scanner VM Inspector

Quando abiliti Enhanced EC2 Scanning nella console Amazon Inspector, Amazon Inspector utilizza Amazon EC2 Systems Manager (SSM) per installare automaticamente lo scanner VM sulle istanze Amazon EC2 gestite. Una volta installato, lo scanner viene eseguito periodicamente (per impostazione predefinita ogni 3 ore) e invia i risultati al servizio di telemetria Amazon Inspector.

Requisiti

Per utilizzare il metodo di installazione automatica, le istanze Amazon EC2 devono soddisfare i seguenti requisiti:

- L'agente SSM deve essere installato e in esecuzione sull'istanza. Per ulteriori informazioni, consulta [Utilizzo dell'SSM Agent](#) nella Guida per l'utente di AWS Systems Manager Systems Manager.
- L'istanza deve avere un profilo di istanza IAM che consenta a SSM di gestire l'istanza. Per ulteriori informazioni, vedere [Configurare le autorizzazioni delle istanze per Systems Manager](#) nella Guida per l'AWS Systems Manager utente.
- L'istanza deve disporre di connettività di rete agli endpoint del servizio SSM.

Note

Se sulle istanze non è installato SSM Agent o non possono soddisfare questi requisiti, puoi utilizzare invece il metodo di installazione manuale. Per ulteriori informazioni, consulta [Installazione e configurazione manuali](#).

Abilitazione della scansione EC2 avanzata

Per abilitare la scansione EC2 avanzata e installare automaticamente lo scanner VM:

1. Apri la console Amazon Inspector in [Getting Started with](#) the Console di gestione AWS
2. Nel riquadro di navigazione, scegli Gestione account.
3. Nella sezione Scansione EC2, scegli Modifica.
4. Abilita la scansione EC2 avanzata.

Dopo aver abilitato Enhanced EC2 Scanning, Amazon Inspector crea un'associazione SSM che installa lo scanner VM su tutte le istanze idonee del tuo account. Lo scanner inizia a eseguire automaticamente le valutazioni delle vulnerabilità.

Requisiti degli endpoint Amazon VPC per la scansione EC2 avanzata su istanze Amazon EC2 private

Puoi eseguire Enhanced EC2 Scanning su istanze Amazon EC2 su una rete Amazon. Tuttavia, se desideri eseguire Enhanced EC2 Scanning su istanze Amazon EC2 private, devi creare endpoint Amazon VPC. Sono necessari i seguenti endpoint:

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

region Dov'è il codice regionale applicabile Regione AWS.

Per ulteriori informazioni, consulta [Migliorare la sicurezza delle istanze Amazon EC2 utilizzando gli endpoint Amazon VPC per Systems Manager nella Guida per l'utente.AWS Systems Manager](#)

Domande frequenti

In che modo Inspector VM Scanner è diverso dal plugin Inspector SSM? Perché dovrei migrare a Inspector VM Scanner?

Inspector VM Scanner utilizza Amazon Inspector SBOM Generator per la scansione e l'ispezione approfondita dei pacchetti di sistema, fornendo un meccanismo di scansione coerente indipendentemente dal modo in cui i pacchetti sono stati installati. Questo meccanismo di scansione è anche più performante, con il maggiore miglioramento delle prestazioni registrato. Windows Questo meccanismo è interamente di proprietà e gestito da Inspector, che consente a Inspector di rispondere rapidamente a qualsiasi problema che si presenti, nonché di espandere i rilevamenti in nuovi ecosistemi.

In che modo Inspector VM Scanner è diverso da Inspector SBOM Generator? Perché non utilizzare direttamente SBOM Generator?

Inspector VM Scanner utilizza SBOM Generator sotto il cofano per la raccolta dell'inventario, ma contiene anche funzionalità extra progettate specificamente per Inspector EC2 Scanning. Inspector VM Scanner può comunicare tramite il canale di telemetria Inspector, mentre Inspector SBOM Generator non può. Inoltre, Inspector VM Scanner è dotato di un'orchestrazione da richiamare regolarmente per la scansione pianificata.

Come verranno gestite le scansioni del Center for Internet Security (CIS)? Lo scanner VM Inspector verrà utilizzato per il CIS?

Le scansioni CIS verranno conservate nel plug-in Inspector SSM. Al momento, non è prevista la migrazione del supporto CIS a Inspector VM Scanner.

Inspector VM Scanner richiede SSM?

No. SSM viene utilizzato per l'installazione automatica quando si abilita la scansione EC2 avanzata nella console, ma non è una dipendenza rigida. È possibile installare manualmente VM Scanner utilizzando gestori di pacchetti standard senza SSM. Per ulteriori informazioni, consulta [Installazione e configurazione manuali](#).

Third-party attribuzione del software

Inspector VM Scanner include componenti software di terze parti concessi in licenza con licenze open source. In conformità con i termini di tali licenze, il seguente documento di attribuzione elenca i componenti di terze parti, le relative licenze e il testo della licenza applicabile.

```
https://inspector-vm-scanner.s3.amazonaws.com/latest/THIRD-PARTY-LICENSES.txt
```

Installazione e configurazione manuali

Questa sezione descrive come installare e configurare manualmente Inspector VM Scanner sulle istanze Amazon EC2. L'installazione manuale è considerata una scansione basata su agenti e non richiede Amazon EC2 Systems Manager (SSM).

Note

Se abiliti Enhanced EC2 Scanning nella console Amazon Inspector, Amazon Inspector installa e gestisce automaticamente lo scanner VM tramite SSM. L'installazione manuale è

necessaria solo se non è possibile utilizzare SSM o si preferisce gestire lo scanner in modo indipendente.

Installazione e disinstallazione manuale di Inspector VM Scanner

Inspector VM Scanner è disponibile per l'installazione autonoma tramite i seguenti link:

Pacchetti Inspector VM Scanner

Tipo di pacchetto	Architettura	URL
.apk	ARM	https://inspector-vm-scanner.s3.amazonaws.com/latest/APK-ARM/inspector-vm-scanner-latest-r0.apk
	X86_64	https://inspector-vm-scanner.s3.amazonaws.com/latest/APK-X86_64/inspector-vm-scanner-latest-r0.apk
.deb	ARM	https://inspector-vm-scanner.s3.amazonaws.com/latest/DEB-ARM/inspector-vm-scanner_latest_arm64.deb
	X86_64	https://inspector-vm-scanner.s3.amazonaws.com/latest/DEB-X86_64/inspector-vm-scanner_latest_amd64.deb
.msi	X86_64	https://inspector-vm-scanner.s3.amazonaws.com/latest/MSI-X86_64/inspector-vm-scanner-x86_64-latest.msi
.pkg	ARM	https://inspector-vm-scanner.s3.amazonaws.com/latest/PKG-ARM/inspector-vm-scanner.latest.arm64.pkg
.rpm	ARM	https://inspector-vm-scanner.s3.amazonaws.com/latest/RPM-ARM/inspector-vm-scanner-latest.arm64.rpm
	X86_64	https://inspector-vm-scanner.s3.amazonaws.com/latest/RPM-X86_64/inspector-vm-scanner-latest.x86_64.rpm

Per visualizzare la procedura per l'installazione manuale di Inspector VM Scanner sul gestore di pacchetti specificato, selezionare un collegamento dal seguente elenco:

- [APK](#)

- [DEB](#)
- [MSI](#)
- [PKG](#)
- [RPM](#)

APK

Installazione

BRACCIO

```
curl --output-dir /etc/apk/keys -0 https://inspector-vm-scanner.s3.amazonaws.com/latest/APK-ARM/inspector-vm-scanner.pem.pub
curl -0 https://inspector-vm-scanner.s3.amazonaws.com/latest/APK-ARM/inspector-vm-scanner-latest-r0.apk
apk add inspector-vm-scanner-latest-r0.apk
```

X86_64

```
curl --output-dir /etc/apk/keys -0 https://inspector-vm-scanner.s3.amazonaws.com/latest/APK-X86_64/inspector-vm-scanner.pem.pub
curl -0 https://inspector-vm-scanner.s3.amazonaws.com/latest/APK-X86_64/inspector-vm-scanner-latest-r0.apk
apk add inspector-vm-scanner-latest-r0.apk
```

disinstallazione

```
apk del inspector-vm-scanner
```

DEB

Installazione

BRACCIO

```
curl -0 https://inspector-vm-scanner.s3.amazonaws.com/latest/DEB-ARM/inspector-vm-scanner.gpg.pub
curl -0 https://inspector-vm-scanner.s3.amazonaws.com/latest/DEB-ARM/inspector-vm-scanner_latest_arm64.deb
```

```
gpg --import inspector-vm-scanner.gpg.pub
gpg --verify inspector-vm-scanner_latest_arm64.deb
sudo dpkg -i inspector-vm-scanner_latest_arm64.deb
```

X86_64

```
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/DEB-X86_64/inspector-vm-scanner.gpg.pub
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/DEB-X86_64/inspector-vm-scanner_latest_amd64.deb
gpg --import inspector-vm-scanner.gpg.pub
gpg --verify inspector-vm-scanner_latest_amd64.deb
sudo dpkg -i inspector-vm-scanner_latest_amd64.deb
```

disinstallazione

```
sudo dpkg -r inspector-vm-scanner
```

MSI

Installazione

X86_64

```
Invoke-WebRequest https://inspector-vm-scanner.s3.amazonaws.com/latest/MSI-X86_64/inspector-vm-scanner-x86_64-latest.msi -OutFile inspector-vm-scanner-x86_64-latest.msi
msiexec /i inspector-vm-scanner-x86_64-latest.msi /qn
```

disinstallazione

Per disinstallare Inspector VM Scanner attivoWindows, utilizzare il pannello di controllo Windows Programmi e funzionalità o il seguente comando: PowerShell

```
Get-WmiObject -Class Win32_Product | Where-Object {$_.Name -eq "Inspector VM Scanner"}
| ForEach-Object {$_.Uninstall()}
```

PKG

Installazione

BRACCIO

```
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/PKG-ARM/inspector-vm-scanner.latest.arm64.pkg
pkgutil --check-signature inspector-vm-scanner.latest.arm64.pkg
sudo installer -pkg inspector-vm-scanner.latest.arm64.pkg -target /
```

disinstallazione

```
sudo rm /opt/aws/inspector/bin/inspector-vm-scanner
sudo rm -rf /var/lib/amazon/inspector
```

RPM

Installazione

BRACCIO

```
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/RPM-ARM/inspector-vm-scanner.gpg.pub
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/RPM-ARM/inspector-vm-scanner-latest.arm64.rpm
rpm --import inspector-vm-scanner.gpg.pub
rpm --checksig inspector-vm-scanner-latest.arm64.rpm
sudo yum install inspector-vm-scanner-latest.arm64.rpm
```

X86_64

```
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/RPM-X86_64/inspector-vm-scanner.gpg.pub
curl -O https://inspector-vm-scanner.s3.amazonaws.com/latest/RPM-X86_64/inspector-vm-scanner-latest.x86_64.rpm
rpm --import inspector-vm-scanner.gpg.pub
rpm --checksig inspector-vm-scanner-latest.x86_64.rpm
sudo yum install inspector-vm-scanner-latest.x86_64.rpm
```

disinstallazione

```
sudo yum remove inspector-vm-scanner
```

Percorsi di installazione

Su tutti i sistemi Unix-based operativi (incluso macOS), Inspector VM Scanner si installa su. `/opt/aws/inspector/bin/inspector-vm-scanner` L'eccezione sono i sistemi Alpine-based operativi (incluso Chainguard), che utilizzano il percorso alternativo. `/usr/opt/aws/inspector/bin/inspector-vm-scanner`

AttivoWindows, Inspector VM Scanner si installa su. `C:\Program Files\Amazon\Inspector\inspector-vm-scanner.exe`

Questi percorsi di installazione (a parte Alpine) sono identici al plugin Inspector SSM, che archivia tutti i file binari di Inspector in un'unica posizione.

Disinstallazione di Inspector VM Scanner

Se disabiliti Enhanced EC2 Scanning, Inspector disinstalla automaticamente Inspector VM Scanner su tutte le istanze. Inspector-managed Per rimuovere un'installazione manuale, consultate le istruzioni di disinstallazione del vostro gestore di pacchetti nelle sezioni precedenti.

Esecuzione dello scanner VM Inspector

Inspector VM Scanner prevede che venga passato un tipo di scansione come primo parametro. Al momento, l'unico valore supportato è. `sbom`

Comando di utilizzo predefinito:

```
./inspector-vm-scanner sbom --send-results telemetry
```

Opzioni di stampa per la scansione SBOM:

```
./inspector-vm-scanner sbom --help
```

Visualizzazione dell'output

Il flusso di lavoro predefinito di Inspector non salva un SBOM localmente. Tuttavia, se si verificano errori durante l'invio della risorsa SBOM tramite telemetria, questa verrà scritta nelle seguenti posizioni:

- `/var/lib/amazon/inspector/state/sbom.json` su Unix

- C:\ProgramData\Amazon\Inspector\State\s bom.json - Windows

Gli utenti possono sovrascrivere questo percorso durante la chiamata di VM Scanner. Per ulteriori dettagli, consulta [Configurazione avanzata](#).

Visualizzazione dei log di

I log di Inspector VM Scanner iniziano sempre con e finiscono con `inspector-vm-scanner.log`

Sui sistemi Unix, Inspector VM Scanner scrive i log su. `/var/log/amazon/inspector`
AttivoWindows, Inspector VM Scanner scrive i log su. `C:\ProgramData\Amazon\Inspector`
`\Logs`

Entrambi questi percorsi sono identici a quelli utilizzati dal plug-in Inspector SSM. Ciò significa che potresti vedere i vecchi log del plugin Inspector SSM insieme ai log di Inspector VM Scanner.

Inspector VM Scanner può anche generare log metrici opzionali, che tengono traccia dell'utilizzo della CPU e della memoria durante l'esecuzione del programma. Questi log vengono scritti nelle directory di log menzionate in precedenza, in una sottodirectory denominata `(on).metrics`
Metrics
Windows Per ulteriori dettagli, consulta [Configurazione avanzata](#).

Visualizzazione dei log con systemd

Sui sistemi Linux che utilizzano systemd, è possibile utilizzare `journalctl` per visualizzare i log di Inspector VM Scanner.

Visualizza i log in tempo reale

```
journalctl -u inspector-vm-scanner -f
```

Visualizza i log recenti

```
journalctl -u inspector-vm-scanner --since "1 hour ago"
```

Visualizza i log di oggi

```
journalctl -u inspector-vm-scanner --since today
```

Visualizza lo stato del servizio e le esecuzioni recenti

```
systemctl status inspector-vm-scanner
```

Configurazione avanzata

Questa sezione descrive le opzioni di configurazione avanzate per Inspector VM Scanner.

Configurazione delle uscite locali

Inspector VM Scanner offre le seguenti opzioni per configurare la modalità di scrittura degli output locali:

- `--send-results` deve essere impostato su `o. telemetry disabled`. Se l'esito viene `superatodisabled`, Inspector VM Scanner procede senza inviare la SBOM.

Tip

Utilizzare `--state-dir` con `with` per salvare lo SBOM `--send-results disabled` localmente.

- `--log-dir` configura dove vengono scritti i log. Per impostazione predefinita, i log vengono scritti su `stdout`.
- `--log-level` configura la granularità dei log. Per impostazione predefinita, si tratta di `INFO`.
- `--log-retention` configura per quanti giorni conservare i log. Se un file di registro più vecchio di quello in cui si `--log-retention` trova `--log-dir`, viene eliminato. Per impostazione predefinita, questo periodo è di 7 giorni.
- `--debug` configura la registrazione a livello di debug e impone un file di registro dedicato per l'esecuzione corrente (anziché cercare di mantenere un file di registro per ogni giorno).
- `--state-dir` configura dove vengono scritti gli SBOM. Per impostazione predefinita, gli SBOM non vengono salvati.
- `--metric-dir` configura dove vengono scritti i log delle metriche. Per impostazione predefinita, i log delle metriche non vengono salvati.
- `--cpu-profile` abilita il profiler CPU Go runtime e configura dove viene scritto il risultato.

- `--mem-profile` abilita il profiler di memoria di runtime Go e configura dove viene scritto il risultato.
- `--config-path` indica a Inspector VM Scanner di derivare argomenti da un file di configurazione locale. Se lo stesso argomento viene passato sia nella CLI che nel file di configurazione, al valore CLI viene data la priorità.
 - I file di configurazione di Inspector VM Scanner sono specificati in TOML, con tutti i nomi degli argomenti identici alla CLI.

L'esempio seguente mostra un file di configurazione:

```
# Configuration file for Inspector VM Scanner
log-level = "INFO"
send-results = "telemetry"
cpu-profile = "cpuprofile"
mem-profile = "memprofile"
log-dir = "log"
state-dir = "state"
debug = false
log-retention = 7
scan-timeout = 300

[sbom]
max-scan-depth = 5
target-directory = ["~"]
```

Configurazione dell'utilizzo delle risorse

Inspector VM Scanner offre le seguenti opzioni per configurare l'utilizzo delle risorse:

- `--scan-timeout` forza il timeout dello scanner dopo un determinato numero di secondi. Per impostazione predefinita, lo scanner non scade.
- `--nice-priority` imposta la nice priorità del processo (disponibile per i sistemi Unix). Per impostazione predefinita, questo è 3.
- `--cpu-limit` imposta un limite massimo all'utilizzo della CPU (disponibile per i sistemi Linux che utilizzano `cgroups`). Per impostazione predefinita, questo valore è del 65%.
- `--process-priority` configura la priorità del processo (disponibile per i Windows sistemi). Per impostazione predefinita, questa è la BELOW NORMAL priorità.

Note

I valori predefiniti per `--cpu-limit` e `--process-priority` sono identici a Inspector SSM Plugin.

Configurazione delle destinazioni di scansione

Inspector VM Scanner sfrutta Inspector SBOM Generator per la raccolta dell'inventario. Di conseguenza, molte delle opzioni di copertura della scansione di Inspector VM Scanner vengono prese direttamente da SBOM Generator.

Per impostazione predefinita, Inspector VM Scanner utilizza il gruppo di scanner di SBOM Generator e gli `localhost scanner`. `certificate windows-kb`

Inspector VM Scanner offre le seguenti opzioni per configurare le destinazioni di scansione:

- `--max-scan-depth` configura il numero massimo di directory attraversate dalla scansione.
- `--target-directories` configura directory aggiuntive per la scansione al di fuori dei valori predefiniti.
- `--override-scanners` configura i filescanner esatti, sovrascrivendo le impostazioni predefinite di Inspector VM Scanner.
- `--additional-scanners` configura i filescanner da utilizzare in aggiunta alle impostazioni predefinite di Inspector VM Scanner.

È possibile utilizzare il seguente comando per elencare tutti gli scanner disponibili:

```
./inspector-vm-scanner sbom --list-scanners
```

Gestione dell'esecuzione periodica

Quando si installa Inspector VM Scanner tramite un gestore di pacchetti, l'installazione crea un'attività pianificata che esegue automaticamente le scansioni. È possibile visualizzare, modificare o disabilitare questa pianificazione.

Linux (systemd)

Visualizza lo stato del servizio e le esecuzioni recenti

```
systemctl status inspector-vm-scanner
```

Visualizza i log in tempo reale

```
journalctl -u inspector-vm-scanner -f
```

Visualizza i log recenti

```
journalctl -u inspector-vm-scanner --since "1 hour ago"
```

Controlla l'intervallo del timer corrente

```
systemctl cat inspector-vm-scanner.timer
```

Aggiorna l'intervallo del timer

Per modificare la frequenza di scansione, modifica il file dell'unità timer:

```
# Edit the timer unit file
systemctl edit inspector-vm-scanner.timer

# Add override configuration:
[Timer]
OnCalendar=
OnCalendar=daily

# Reload and restart
systemctl daemon-reload
systemctl restart inspector-vm-scanner.timer
```

Abilita o disabilita l'esecuzione automatica

```
systemctl enable inspector-vm-scanner.timer # Enable automatic runs
systemctl disable inspector-vm-scanner.timer # Disable automatic runs
```

Windows (Task Scheduler)

Visualizza lo stato dell'attività e l'ultima esecuzione

```
Get-ScheduledTask -TaskName "Inspector VM Scanner" | Get-ScheduledTaskInfo
```

Visualizza i registri delle attività recenti

```
Get-ScheduledTaskInfo -TaskName "Inspector VM Scanner"
```

Visualizza la cronologia dettagliata delle attività

```
schtasks /query /tn "Inspector VM Scanner" /v /fo list
```

Visualizza la pianificazione attuale delle attività

```
Get-ScheduledTask -TaskName "Inspector VM Scanner" | Select-Object -ExpandProperty Triggers
```

Aggiorna la pianificazione delle attività

Per modificare la frequenza di scansione:

```
# Modify trigger to run daily at 2 AM
$trigger = New-ScheduledTaskTrigger -Daily -At 2:00AM
Set-ScheduledTask -TaskName "Inspector VM Scanner" -Trigger $trigger
```

Attivare o disattivare l'attività

```
Enable-ScheduledTask -TaskName "Inspector VM Scanner" # Enable automatic runs
Disable-ScheduledTask -TaskName "Inspector VM Scanner" # Disable automatic runs
```

macOS (lanciato)

Visualizza l'attività launchd

```
sudo launchctl print system/com.amazon.inspector.vm-scanner
```

Esegui una singola attività

```
sudo launchctl start com.amazon.inspector.vm-scanner
```

Plugin Amazon Inspector SSM per Linux and Windows

Questo argomento descrive il plug-in Amazon Inspector SSM per Linux le istanze e le relative istanze. Windows

Il plug-in Amazon Inspector SSM per Linux

Se non hai abilitato Enhanced EC2 Scanning con Amazon Inspector VM Scanner, Amazon Inspector utilizza il plug-in Amazon Inspector SSM per eseguire scansioni di ispezione approfondita su istanze Linux. Il plug-in Amazon Inspector SSM viene installato automaticamente sulle istanze Linux presenti nella directory `/opt/aws/inspector/bin`. Il nome dell'eseguibile è `inspectorssmplugin`.

Amazon Inspector utilizza Systems Manager Distributor per distribuire il plug-in sulla tua istanza. Per eseguire scansioni di ispezione approfondita, Systems Manager Distributor e Amazon Inspector devono supportare il sistema operativo dell'istanza Amazon EC2. Per informazioni sui sistemi operativi supportati da Systems Manager Distributor, vedere [Piattaforme e architetture di pacchetti supportate](#) nella Guida per l'AWS Systems Manager utente.

Amazon Inspector crea directory di file per gestire i dati raccolti per l'ispezione approfondita dal plug-in Amazon Inspector SSM. Queste directory di file includono `/opt/aws/inspector/var/input` e `/opt/aws/inspector/var/output`.

Il `packages.txt` file `/opt/aws/inspector/var/output` memorizza i percorsi completi dei pacchetti rilevati da Deep Inspection. Se Amazon Inspector rileva lo stesso pacchetto più volte sull'istanza, il `packages.txt` file elenca ogni posizione in cui è stato trovato il pacchetto.

Amazon Inspector archivia i log del plug-in nella directory `/var/log/amazon/inspector`.

Disinstallazione del plug-in Amazon Inspector SSM

Se il `inspectorssmplugin` file viene eliminato inavvertitamente, l'associazione SSM `InspectorLinuxDistributor-do-not-delete` proverà a reinstallare il file all'intervallo di scansione successivo. `inspectorssmplugin`

Se disattivi la scansione di Amazon EC2, il plug-in verrà disinstallato automaticamente da tutti gli host Linux.

Il plug-in Amazon Inspector SSM per Windows

Se non hai abilitato la scansione EC2 avanzata con Amazon Inspector VM Scanner, è necessario il plug-in Amazon Inspector SSM per consentire ad Amazon Inspector di scansionare le istanze Windows. Il plug-in Amazon Inspector SSM viene installato automaticamente sulle Windows istanze in `C:\Program Files\Amazon\Inspector` e il file binario eseguibile viene denominato `InspectorSsmPlugin.exe`.

I seguenti percorsi di file vengono creati per archiviare i dati raccolti dal plug-in Amazon Inspector SSM:

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

Note


Per impostazione predefinita, il plug-in Amazon Inspector SSM viene eseguito con una priorità inferiore a quella normale.

Note

`ssm:PutInventoryRichieste` e autorizzazioni per la scansione Windows delle istanze.
`ssm:GetParameter` Se sull'istanza è configurato un profilo di istanza IAM, Amazon Inspector utilizza quel profilo e ignora il ruolo Default Host Management Configuration (DHMC). Il profilo dell'istanza deve includere queste autorizzazioni. Se non è impostato alcun profilo di istanza, Amazon Inspector utilizza il ruolo [Default Host Management Configuration](#) configurato, che deve includere queste autorizzazioni.

Disinstallazione del plug-in Amazon Inspector SSM

Se il `InspectorSsmPlugin.exe` file viene eliminato inavvertitamente, l'`InspectorDistributor-do-not-delete` associazione lo reinstallerà al successivo intervallo di scansione `InspectorSsmPlugin.exe`. Windows Se desideri disinstallare il plug-in Amazon Inspector SSM, puoi utilizzare l'azione di disinstallazione nel documento. `AmazonInspector2-ConfigureInspectorSsmPlugin` Tuttavia, il plug-in Amazon Inspector SSM verrà disinstallato automaticamente da tutti gli Windows host se disattivi la scansione di Amazon EC2.

 Note

Se disinstalli l'agente SSM prima di disattivare Amazon Inspector, il plug-in Amazon Inspector SSM rimarrà sull'Windows host, ma non invierà dati al plug-in Amazon Inspector SSM. Per ulteriori informazioni, consulta [Disattivazione di Amazon Inspector](#).

Generatore SBOM Amazon Inspector

Una Software Bill of Materials (SBOM) è [un elenco formalmente strutturato di componenti, librerie e moduli](#) necessari per creare un software. Amazon Inspector SBOM Generator (Sbomgen) è uno strumento che produce un SBOM per archivi, immagini di container, directory, sistemi locali, compilati e binari. Go Rust Sbomgen analizza i file che contengono informazioni sui pacchetti installati. Quando Sbomgen trova un file pertinente, estrae i nomi dei pacchetti, le versioni e altri metadati. Sbomgen quindi trasforma i metadati del pacchetto in un SBOM. CycloneDX Puoi utilizzarlo Sbomgen per generare lo CycloneDX SBOM come file o in STDOUT e inviarlo ad Amazon SBOMs Inspector per il rilevamento delle vulnerabilità. Puoi anche utilizzarlo Sbomgen come parte dell' [CI/CD integrazione, che scansiona automaticamente le](#) immagini dei container come parte della tua pipeline di distribuzione.

Amazon Inspector utilizza diverse funzionalità Sbomgen di scansione gestita. Quando Amazon Inspector esegue la scansione delle [immagini dei container Amazon ECR](#), la scansione della [funzione AWS Lambda e la scansione senza agente delle istanze di](#) Amazon Elastic Compute Cloud tramite l'analisi del volume di Amazon EBS, richiama il Sbomgen dietro le quinte per generare l'inventario software utilizzato per la valutazione delle vulnerabilità. SbomgenLo strumento autonomo descritto in questa pagina fornisce la stessa tecnologia di base, che consente di integrare la generazione di SBOM direttamente nei flussi di lavoro.

Tipi di pacchetti supportati

Sbomgen raccoglie l'inventario per i seguenti tipi di pacchi:

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

- Rust

Controlli di configurazione dell'immagine del contenitore supportati

Sbomgen può scansionare file Dockerfile autonomi e creare una cronologia da immagini esistenti per individuare problemi di sicurezza. Per ulteriori informazioni, consulta [Amazon Inspector Dockerfile checks](#).

Installazione di Sbomgen

Sbomgen è disponibile solo per i sistemi operativi Linux.

È necessario averlo Docker installato se si desidera analizzare Sbomgen le immagini memorizzate nella cache locale. Docker non è necessario analizzare le immagini esportate come `.tar` file o le immagini ospitate in registri di container remoti.

Amazon Inspector consiglia l'esecuzione Sbomgen da un sistema con almeno le seguenti specifiche hardware:

- CPU a 4 core
- 8 GB RAM

Per installare Sbomgen

1. Scarica il file Sbomgen zip più recente dall'URL corretto per la tua architettura:

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

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

In alternativa, puoi scaricare [le versioni precedenti del file zip di Amazon Inspector SBOM Generator](#).

2. Decomprimi il download utilizzando il seguente comando:

```
unzip inspector-sbomgen.zip
```

3. Verificate la presenza dei seguenti file nella directory estratta:

- `inspector-sbomgen`— Questo è lo strumento che eseguirete per generare SBOMs.
 - `README.txt`— Questa è la documentazione per l'utilizzo `Sbomgen`.
 - `LICENSE.txt`— Questo file contiene la licenza del software per `Sbomgen`.
 - `licenses`— Questa cartella contiene informazioni sulla licenza per i pacchetti di terze parti utilizzati da `Sbomgen`.
 - `checksums.txt`— Questo file fornisce gli hash dello `Sbomgen` strumento.
 - `sbom.json`— Questa è una CycloneDX SBOM per lo strumento. `Sbomgen`
 - `WhatsNew.txt`— Questo file contiene un registro delle modifiche riepilogativo, in modo da poter visualizzare rapidamente le principali modifiche e miglioramenti tra le `Sbomgen` versioni.
4. (Facoltativo) Verifica l'autenticità e l'integrità dello strumento utilizzando il seguente comando:

```
sha256sum < inspector-sbomgen
```

- Confrontate i risultati con il contenuto del `checksums.txt` file.

5. Concedi le autorizzazioni eseguibili allo strumento utilizzando il seguente comando:

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

6. Verificate che `Sbomgen` sia installato correttamente utilizzando il seguente comando:

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

L'output dovrebbe essere simile al seguente:

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

Uso di `Sbomgen`

Questa sezione descrive diversi modi di utilizzo `Sbomgen`. Puoi saperne di più su come utilizzare `Sbomgen` tramite esempi incorporati. Per visualizzare questi esempi, esegui il `list-examples` comando:

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

Genera un SBOM per un'immagine del contenitore e restituisci il risultato

È possibile Sbmngen utilizzarlo per generare immagini SBOMs per il contenitore e inviare il risultato in un file. Questa funzionalità può essere abilitata utilizzando il `container` sottocomando.

Esempio di comando

Nel frammento seguente, puoi sostituirlo *image:tag* con l'ID dell'immagine e *output_path.json* con il percorso dell'output che desideri salvare.

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

Note

Il tempo e le prestazioni di scansione dipendono dalle dimensioni dell'immagine e dal numero ridotto di livelli. Immagini più piccole non solo migliorano Sbmngen le prestazioni, ma riducono anche la potenziale superficie di attacco. Le immagini più piccole migliorano anche i tempi di creazione, download e caricamento delle immagini.

Quando viene utilizzata Sbmngen con [ScanSbom](#), l'API Amazon Inspector Scan non elabora prodotti SBOMs contenenti più di 5.000 pacchetti. In questo scenario, l'API Amazon Inspector Scan restituisce una risposta HTTP 400.

Se un'immagine include file o directory multimediali in blocco, valuta la possibilità di escluderli dall'Sbmngenutilizzo dell'argomento. `--skip-files`

Esempio: casi di errore comuni

La scansione delle immagini dei contenitori può fallire a causa dei seguenti errori:

- **InvalidImageFormat**— Si verifica durante la scansione di immagini di contenitori non valide con intestazioni TAR, file manifest o file di configurazione danneggiati.
- **ImageValidationFailure**— Si verifica quando la convalida del checksum o della lunghezza del contenuto non riesce per i componenti dell'immagine del contenitore, ad esempio intestazioni Content-Length non corrispondenti, digest del manifesto errati o verifica del checksum non riuscita. SHA256

- `ErrUnsupportedMediaType`— Si verifica quando i componenti dell'immagine includono tipi di supporti non supportati. Per informazioni sui tipi di file multimediali supportati, [consultate Sistemi operativi e tipi di supporti](#) supportati.

Amazon Inspector non supporta questo tipo di `application/vnd.docker.distribution.manifest.list.v2+json` supporto. Tuttavia, Amazon Inspector supporta gli elenchi di manifest. Durante la scansione di immagini che utilizzano elenchi manifest, puoi specificare in modo esplicito quale piattaforma utilizzare con l' `--platform` argomento. Se l' `--platform` argomento non è specificato, Amazon Inspector SBOM Generator seleziona automaticamente il manifesto in base alla piattaforma su cui è in esecuzione.

Genera un SBOM da directory e archivi

È possibile utilizzarlo `Sbomgen` per generare a SBOMs partire da cartelle e archivi. Questa funzionalità può essere abilitata utilizzando i `directory` `archive` sottocomandi o. Amazon Inspector consiglia di utilizzare questa funzionalità quando desideri generare un SBOM da una cartella di progetto, ad esempio un repository git scaricato.

Comando di esempio 1

Il frammento seguente mostra un sottocomando che genera un SBOM da un file di directory.

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

Esempio di comando 2

Il frammento seguente mostra un sottocomando che genera un SBOM da un file di archivio. Gli unici formati di archivio supportati sono, e. `.zip` `.tar` `.tar.gz`

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

Genera un SBOM da Go o Rust compilati file binari

È possibile utilizzarlo `Sbomgen` per generare SBOMs da file compilati Go e binari. Rust È possibile abilitare questa funzionalità tramite il sottocomando: `binary`

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

Genera un SBOM dai volumi montati

Puoi utilizzare Amazon Inspector SBOM Generator per generare SBOMs da volumi montati. Questa funzionalità può essere abilitata utilizzando il sottocomando `volume`. Ti consigliamo di utilizzare questa funzionalità quando desideri analizzare volumi di storage, come i volumi Amazon EBS che sono stati montati sul tuo sistema. A differenza del sottocomando `directory`, la scansione dei volumi montati rileva i pacchetti del sistema operativo e le informazioni sul sistema operativo.

Puoi scansionare un volume Amazon EBS collegandolo a un'istanza Amazon EC2 in cui è installato Amazon Inspector SBOM Generator e montandolo su quell'istanza. Per i volumi Amazon EBS attualmente utilizzati da altre istanze Amazon EC2, puoi creare uno snapshot Amazon EBS del volume e quindi creare un nuovo volume Amazon EBS da quello snapshot per scopi di scansione. Per ulteriori informazioni su Amazon EBS, consulta [What is Amazon EBS?](#) nella Guida per l'utente di Amazon Elastic Block Store.

Esempio di comando

Il seguente frammento mostra un sottocomando che genera un SBOM da un volume montato. L'opzione `--path` deve specificare la directory principale in cui è montato il volume.

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

Esempio di comando

Il frammento seguente mostra un sottocomando che genera un SBOM da un volume montato escludendo percorsi di file specifici con l'argomento `--exclude-suffix`. L'argomento `--exclude-suffix` è particolarmente utile quando un volume contiene file di massa (come file di registro o file multimediali). I file e le directory i cui percorsi terminano con i suffissi specificati verranno esclusi dalla scansione, il che può ridurre i tempi di scansione e l'utilizzo della memoria.

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

Tutti i percorsi dei file nel volume di destinazione vengono normalizzati ai percorsi originali. Ad esempio, durante la scansione di un volume montato su `/mnt/volume` che contiene un file in `/mnt/`

`volume/var/lib/rpm/rpmdb.sqlite`, il percorso verrà normalizzato a quello contenuto `/var/lib/rpm/rpmdb.sqlite` nella SBOM generata.

Invia un SBOM ad Amazon Inspector per l'identificazione delle vulnerabilità

Oltre a generare un SBOM, puoi inviare un SBOM per la scansione con un solo comando dall'API Amazon Inspector Scan. Amazon Inspector valuta i contenuti dello SBOM alla ricerca di vulnerabilità prima di restituire i risultati. Sбомgen A seconda dell'input, i risultati possono essere visualizzati o scritti su un file.

Note

È necessario disporre di un account attivo Account AWS con autorizzazioni di lettura `InspectorScan-ScanSbom` per utilizzare questa funzionalità.

Per abilitare questa funzionalità, si passa `--scan-sbom` al comando `sbomgen`. È inoltre possibile passare `--scan-sbom` a uno dei seguenti sottocomandi: `archive`, `binarycontainer`, `directory`, `localhost`

Note

L'API Amazon Inspector Scan non elabora SBOMs con più di 5.000 pacchetti. In questo scenario, l'API Amazon Inspector Scan restituisce una risposta HTTP 400.

Puoi autenticarti su Amazon Inspector tramite AWS un profilo o un ruolo IAM con i AWS CLI seguenti argomenti:

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

Puoi anche autenticarti su Amazon Inspector fornendo le seguenti variabili di ambiente a `sbomgen`

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

Per specificare il formato della risposta, usa l'`--scan-sbom-output-format cyclonedx` argomento o `--scan-sbom-output-format inspector` l'argomento.

Esempio di comando 1

Questo comando crea un SBOM per l'ultima Alpine Linux versione, analizza lo SBOM e scrive i risultati della vulnerabilità in un file JSON.

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

Esempio di comando 2

Questo comando esegue l'autenticazione su Amazon Inspector AWS utilizzando credenziali come variabili di ambiente.

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

Esempio di comando 3

Questo comando esegue l'autenticazione su Amazon Inspector utilizzando l'ARN per un ruolo IAM.

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

```
--aws-region your_region \  
--outfile /tmp/inspector_scan.json  
--aws-iam-role-arn arn:aws:iam::123456789012:role/your_role
```

Utilizza scanner aggiuntivi per migliorare le capacità di rilevamento

Amazon Inspector SBOM Generator applica scanner predefiniti in base al comando utilizzato.

Gruppi di scanner predefiniti

Ogni sottocomando di Amazon Inspector SBOM Generator applica automaticamente i seguenti gruppi di scanner predefiniti.

- Per il `directory` sottocomando: `binary`, `dockerfile` scanner groups programming-language-packages
- Per il `localhost` sottocomando: `os`,, `programming-language-packages` extra-ecosystems scanner groups
- Per il `container` sottocomando: `os`, `extra-ecosystems`, `programming-language-packages` `dockerfile`, `binary` scanner groups

Scanner speciali

Per includere scanner oltre ai gruppi di scanner predefiniti, utilizzate l'`--additional-scanners` opzione seguita dal nome dello scanner da aggiungere. Di seguito è riportato un comando di esempio che mostra come eseguire questa operazione.

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

Di seguito è riportato un comando di esempio che mostra come aggiungere più scanner con un elenco separato da virgole.

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

Ottimizza le scansioni dei contenitori regolando la dimensione massima del file da scansionare

Quando si analizza ed elabora l'immagine di un contenitore, per impostazione predefinita esegue la Sbomgen scansione di file di dimensioni pari o inferiori a 200 MB. I file di dimensioni superiori a 200 MB contengono raramente metadati dei pacchetti. È possibile che si verifichino errori quando si effettua l'inventario di un Rust file binario Go di dimensioni superiori a 200 MB. Per modificare il limite di dimensione, usa l'argomento. `--max-file-size` Ciò consente di aumentare il limite per includere file di grandi dimensioni e di diminuirlo per ridurre l'utilizzo delle risorse escludendo i file di grandi dimensioni.

Esempio

L'esempio seguente mostra come utilizzare l'`--max-file-size` argomento per aumentare le dimensioni del file.

```
# Increase the file size limit to scan files up to 300 MB
./inspector-sbomgen container --image alpine:latest \
--outfile /tmp/sbom.json \
--max-file-size 300000000
```

La regolazione di questa impostazione consente di controllare l'utilizzo del disco, il consumo di memoria e la durata complessiva della scansione.

Disattiva l'indicatore di avanzamento

Sbomgen visualizza un indicatore di avanzamento della rotazione che può causare un numero eccessivo di caratteri barra negli CI/CD ambienti.

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

È possibile disabilitare l'indicatore di avanzamento utilizzando l'`--disable-progress-bar` argomento:

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

Autenticazione in registri privati con Sbomgen

Fornendo le credenziali di autenticazione del registro privato, è possibile eseguire la generazione a SBOMs partire da contenitori ospitati in registri privati. È possibile fornire queste credenziali tramite i seguenti metodi:

Autenticazione tramite credenziali memorizzate nella cache (scelta consigliata)

Con questo metodo, ci si autentica nel registro dei contenitori. Ad esempio, se si utilizza Docker, è possibile autenticarsi nel registro dei contenitori utilizzando il comando di registrazione: Docker. `docker login`

1. Effettua l'autenticazione nel registro dei contenitori. Ad esempio, se si utilizza Docker, è possibile autenticarsi nel registro utilizzando il Docker `login` comando:
2. Dopo l'autenticazione nel registro dei contenitori, utilizzala Sbomgen su un'immagine del contenitore presente nel registro. Per utilizzare l'esempio seguente, sostituiscilo `image:tag` con il nome dell'immagine da scansionare:

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

Effettua l'autenticazione utilizzando il metodo interattivo

Per questo metodo, fornite il vostro nome utente come parametro e vi Sbomgen richiederà l'immissione sicura della password quando necessario.

Per utilizzare l'esempio seguente, `image:tag` sostituiscilo con il nome dell'immagine che desideri scansionare e `your_username` con un nome utente che abbia accesso all'immagine:

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

Effettua l'autenticazione utilizzando il metodo non interattivo

Per questo metodo, memorizza la password o il token di registro in un `.txt` file.

Note

L'utente corrente dovrebbe essere in grado di leggere solo questo file. Il file deve contenere anche la password o il token su una sola riga.

Per utilizzare l'esempio seguente, *your_username* sostituiscilo con il tuo nome utente, *password.txt* con il `.txt` file che include la password o il token su un'unica riga e *image:tag* con il nome dell'immagine da scansionare:

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

Esempi di risultati da Sbomgen

Di seguito è riportato un esempio di SBOM per un'immagine di contenitore inventariata utilizzando Sbomgen

Immagine del contenitore (SBOM)

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

```

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

```

```

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

```

}

Versioni precedenti di Amazon Inspector SBOM Generator

Questo argomento include collegamenti alle versioni più recenti e precedenti di Amazon Inspector SBOM Generator. Per informazioni sull'installazione di Sbmngen, consulta [Installazione di Sbmngen](#).

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD64	1.14.1	2b9512ea9acdc1eb1a a155f7f664f92b4dc9
Linux ARM64	1.14.1	6e14d3849fe1f1d3aa 51f481b627 f23dafdf0127dfd61a f2678aaa67643bf1f0 909689487febe68143 0302a0c8a3
Linux AMD 64	1.14.0	f884433321762afec8 89acb78b935a9a6f5f
Linux ARM64	1.14.0	b92fd4bbfa1a54ac03 7e9f24c351 55c62f54dd96e13c08 5acf946cd2a841453a f00918050331a86b2b 83d1f80eaa
Linux AMD 64	1.13.2	86011ffc7ee1a85d4c b411ad807566d7fde3
Linux ARM64	1.13.2	7fe9756a9a7e5a2d22 205645f2d2 d4b6328867f0cac169 8ec4e0862cbdf45af0 68888ce3c1ed30d780 b110c909af

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.13.1	c100ff7a65f2795ada 9193aea0f2944ec677
Linux ARM64	1.13.1	31313a6eb8b841a0ff bf0ebba2c9 3ceaac63796f6ca97e 9b5ae7c29bf8c896b2 0b2616bacde5ea1265 5c30467164
Linux AMD 64	1.13.0	f6d81b76f46a7461a2 191a72880
Linux ARM64	1.13,0	a7ee90a13ebd4c1b1a 2ab6cbc7b469613bd9 3 6d69c2394df58ccf09 40f3495044b90c3f5f eca2650d7af14f2000 a537c4e96e
Linux AMD 64	1.12.2	c1d77171941f324519 bccf78339e71a8961b
Linux ARM64	1,12,2	7bc2a26351772c09dc 6fb9dde83f d4b0eee17c269fcaef 7361f714976d31cb49 4c68eadf9a9cd8e628 ee14810a7e

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.12.1	2d698b47d
Linux ARM64	1.12.1	2a6ba0b431c1161bc1 fba7bef4d9041546e6 24086d4b0dfff122a3f c76fe1bd84ffe92709 21df1d708fc0e49c53 aa7717d29a90d5cec3 cee772f0c2
Linux AMD 64	1.12.0	1112b3952
Linux ARM64	1,12,0	81ea5d67e 20062429bf1603ba64 56746d6941f2edfa0e e0812b34b6 25e618000ffc4aceaf 002c2fa3e97d0564d3 76acc34e9e442cdd76 9f6c805248
Linux AMD64	1.11.2	bef68671bc532e4fb5
Linux ARM64	1.11.2	29500b62d7af836012 3cd967308d41ad0ce8 f43f7762fb 4f11d7037efa443f44 2c4edf7ba28774c4fa 706fb7622e4fba645b b3ad3958c9

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.11.1	809eb7cb80d24fbf6f fdd124438d53a90763
Linux ARM64	1.11.1	2c222e924913ebd610 44ca949490 057f9e4c9970aeda4b da0685e7e02436fd52 23fbe81cec65138551 c63ed77ba0
Linux AMD 64	1.11.0	5172a5556cf46f9fbc 5cf1d35bd382919fb6
Linux ARM64	1.11.0	b41aca1ec938db3a75 530060b0cf c9e2da7b076dc89dc3 9a962a7dd9c7d1fd29 230a4eec7eb95f951d 6a179093d0
Linux AMD 64	1.10.1	9e33622a7874adfe71 9ab7db75a1e44f4b5f
Linux ARM64	1.10.1	ae3573374068b501c8 9f0accfefe 78d5a7f800fc26ba86 adab5b634431a91c00 7075e06d6ce46e5068 7d5156184e

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.10.0	0b7a553d7d2d17c40a
Linux ARM64	1,10,0	62f1a11013bc46fa2c 3814f407c11130e15a f3fe313769 5ce9e315a4f8f90ff5 eed7ab058efc8dbff6 593d66d3fc455f1c37 e882ec6466
Linux AMD 64	1.9.1	d0ef4c14fec6c42e70 ae55b3e44
Linux ARM64	1.9.1	d17d02713 2947596e8ef861c0ef c3c0e5a871 2d8145011c13f5611f c30f4510785d53e98b 911717f6dbe69616af 4d4b0df61f
Linux AMD 64	1.9.0	78b377b27 30eb15476
Linux ARM64	1.9.0	173e40885 454ae191e953663af3 e0928dddfb8608f465 5 985bdc06d25eccb87c 4a81995c8a2d3c78e1 c02beea309a620b2de 4954767591

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD64	1.8.3	54eed5a772f68320f3 906bec5920e3a19da9
Linux ARM64	1.8.3	04abdace10f985b878 59015eef89 febd74a397fb0cdd33 56072503f08465ab87 2d1620d59 a2ab7d83bdb076c929 d
Linux AMD 64	1.8.2	2e4e3c754e23004634 9dd975feb48fa953ea
Linux ARM64	1.8.2	5a2de190cbbc17c1c8 5043936b5a 449a49e22 2a2bdffe0353435d7b 04b0556b35a391c7b9 714ce46d1a5382bc3e 2
Linux AMD 64	1.8.1	9ff7958e298d2b228b 0c7617f0a9a8732545
Linux ARM64	1.8.1	87fc26aee9826c3727 3650b389e9 6737584fd2c7d24b56 777d02846 d1737f47d0121344ba ea217a3e5368fd98fcc

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD64	1.8.0	ef32e7fb4ee0af1e47 d6b528b47293fc7127
Linux ARM64	18.0	c7a7539f7354e84452 626a4c204d 0b82ddc691a517bb8f c6ccd67b80ca566b11 7a1bb410c05764c9b7 e3ba76c510
Linux AMD64	1.7.3	3fba95d44aaea55ad0 6d3c7635a671662c48
Linux ARM64	1.7.3	3474578376d3f11e84 474f8de25f 1f4b52e3d80de87b92 b563a78bac4a2d898e 7af82db5b6791d899d 516e97cfbb
Linux AMD 64	1.7.2	c44ba9bf1cf3eb3ea2d 6d0b15d25
Linux ARM64	1.7.2	816800a50 45a438474f2f77c390 bac41ae4cb d37c5b1605bf82260d a0b0f36311c83b1646 a4327c3fd8169ba4b3 a978470c9c

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.7.1	b0beb602a
Linux ARM64	1.7.1	6ae439d4e
		307bd99682bc8a419f
		d7d5e78a278bfc718e
		b18e00b05e
		95ff2d9df2fcd1982d
		d705df1e763f57a0b4
		99b6fe06801e9a8086
		9e2e464831
Linux AMD 64	1.7.0	a6316c2ecd5fde7091
Linux ARM64	1.7.0	d1099335f45f0e2400
		b3977c92ee4d72bd1e
		b359320e61
		9751ba5e5c6c6c6c0a
		ef7d29b1c4adbd4088
		da3a07bb77eaa7de3f
		04aa33ad8562
Linux AMD 64	1.6.3	b6a309e87
Linux ARM64	1.6.3	9aaa78d7d
		8e224eb5214df5fd41
		5244d370885e6c8876
		db5a4181d2
		59ed0b7eb
		7d1eadadb691f058d3
		2634a03a856ba03ac2
		ddb8cd3599ceb55cb9
		a

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.6.2	8d8ba0653
Linux ARM64	1.6.2	5be614a4d44b1bd74c 66d1fd4874ff9ab788 ad5e23aa5229db9c68 7 2bd7b4a88b9c6b041a 6ff82f7f9bc116b76c f410bf6eb896fc8d68 e717b55f2a
Linux AMD 64	1.6.1	3e3d62dc794b31d9d2
Linux ARM64	1.6.1	de1904592cf42f25e9 f42c30eb90cc53385a 60b42f1a63 ad89f670908fb0b48b ca0242f3ac58e7179f 6fabfcc9a2b3fd0e5c 3d79e27539
Linux AMD 64	1.6.0	ffe671c2c1d1c2142a
Linux ARM64	1.6.0	4af056d1c179eaffbc 3925f5afaa6f3d655b d495ce5e1c a733c0b00c7225369c 68ad47c57846b4546e 2c9f47580ab98394ba efc765c134

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.5.5	ebcfbe565631de5bc6 1b1d55d70
Linux ARM64	1,5,5	a2d15b965f628678a2 b60cffd01cd0c3443f1 a8e018ceee3a76dd42 71f966015c216438b1 1ee807fcd970753e78 6baa335b56
Linux AMD 64	1.5.4	aa8c1ffacc563b8797 5497f53eddec0b2939
Linux ARM64	1.5.4	7a898fac19f4902b8a cb7eeb347b c6ba98d441aa88d3d3 150449c098cd13ce3b aeccee45ad4c9a1326 f8bb8f87fc
Linux AMD64	1.5.3	d493c23121101c9c3d f888e717bf81d7f7b8
Linux ARM64	1.5.3	1809754f3492e1ae52 f02b089b68 8dfa5c97b3bd45da48 7706e95d1894290f53 b113247bbb89b9fac1 6dab8184b6

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.5.2	ff6233d7da9f7e9635
Linux ARM64	1.5.2	89a0eb8f07bee2ca37 5360365cb6b6e35458 5cf1371910
Linux AMD64	1.5.1	fd31efb6031754b2bc 8414d7fe9dd14a0677 67704145af0559b350 0cc437c7ee
Linux ARM64	1.5.1	391fcc52117fed79ca e6e92a9e2 25732166a6df2582aa 7f6b5230149761f673 2
Linux AMD 64	1.5.0	f9bc90d18724f93db0 f5ca3b79136adb7b49 fa33fa179a5e87b4d5 12f256b56b
Linux ARM64	1.5.0	d7b6cb84053358e462 d76488d019140ecd05 ad405217a 60a96b727fb062880f e
		067dcf5c302160a527 0f89aed3f941bb0571 dcb8a59f75dddb1b77 47c2a82ec7

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.4.0	c8ca73761afd742e1d
Linux ARM64	1.4.0	eb98b04eb5714c9c2a 574b652a7 63b18e235 60e66aea24 188d97577 82278653e65605aaf1 86feda104345ba2f9d e438873e568f1ff6204
Linux AMD 64	1.3.2	57dd5d135
Linux ARM64	1.3.2	600e84690706cfe958 60e78149988d37cf81 429ce97b9256d179fb 4 91526ecdafc6cc3718 fabe75b2693ace5eff b9c0af3327b484b7f5 a154929997
Linux AMD 64	1.3.1	097ec83907c459a36d
Linux ARM64	1.3.1	e11c92d016ffd64f1 c33fd4bcbf2af465e0 979b0d9237 aa93a3d402abc4a986 a9ad9d3de8fcca81ee 25a55596ac6dc4502e d1d6819502

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.3.0	21439f92c314daf136 832ca6676a65d28876
Linux ARM64	1.3.0	8aa69fc6dcd2014a30 38b2701eeb 4a41779b0c3b32242e edef288de6c1bf40fd a0d4246b32fd0cd8d4 e51e58f94b
Linux AMD 64	1.2.1	e022e95e59f1790949 bca8dbbb6478a5d3fb
Linux ARM64	1.2.1	677ccd45aa4ba30ebd 91ae86ad65 824acc5bb5b0210954 fe9ab089d9461453a4 975d34292cc0c67683 7c3a7279b4
Linux AMD 64	1.2.0	9625b1a8ae1937ca21 79c2535a0ffceca934
Linux ARM64	1.2.0	138e0b66feac9ba3e3 4ffaa22ec5 7f387e560b41571fb5 2efd9e620bf2b9e3a0 67ca781e88aaa977b2 b8acdebf35

Platform (Piattaforma)	Versione	SHA-256 checksum
Linux AMD 64	1.1.1	6809b7e46675c66e3a f354c53433dc46c4d1
Linux ARM64	1.1.1	ddaf258e05ba15e38e 784ea0285e 6361e59fb2448c66c4 698ea33979ecaaefc2 af4420034aabbbe741 242f60dbdd
Linux AMD 64	1.1.0	f84c8815413d451490 b38509950235f88713
Linux ARM64	1.1.0	c0c61c7259a4831934 995664bd8f aaffefb5e44195dc55 d5fd3289e511720f64 c130644cbd58103cf7 f36e96f058
Linux AMD 64	1.0.0	cc126e24962f1a6497 cf17679b3e3b73be68
Linux ARM64	1.0.0	963c47e3968a56e73c aacf045b5c 5d5bf97a4acfeaaa73 ad6c918738188e0c82 2e475ef37a334e49d7 7ba907b08a

Raccolta completa di sistemi operativi Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator esegue la scansione di diversi sistemi operativi per garantire un'analisi affidabile e dettagliata dei componenti del sistema. La generazione di un SBOM ti aiuta a

comprendere la composizione del tuo sistema operativo, in modo da poter identificare le vulnerabilità nei pacchetti gestiti dal sistema. Questo argomento descrive le caratteristiche principali delle diverse raccolte di pacchetti del sistema operativo supportate da Amazon Inspector SBOM Generator.

Per informazioni sui sistemi operativi supportati da Amazon Inspector, consulta [Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector](#).

Artefatti del sistema operativo supportati

Amazon Inspector SBOM Generator supporta i seguenti elementi del sistema operativo:

Platform (Piattaforma)	Binario	Origine	Flusso
Alma Linux	N/D	Sì	Sì
Alpine Linux	Sì	Sì	N/D
Amazon Linux	N/D	Sì	N/D
CentOS	N/D	Sì	N/D
Chainguard	Sì	Sì	N/D
Debian	Sì	Sì	N/D
Distroless	Sì	Sì	N/D
Fedora	N/D	Sì	N/D
MinimOS	Sì	Sì	N/D
OpenSUSE	N/D	Sì	N/D
Oracle Linux	N/D	Sì	N/D
Photon OS	N/D	Sì	N/D
RHEL	N/D	Sì	Sì
Rocky Linux	N/D	Sì	Sì
SLES	N/D	Sì	N/D

Platform (Piattaforma)	Binario	Origine	Flusso
Ubuntu	Sì	Sì	N/D
Windows	N/D	N/D	N/D

Raccolta di pacchetti del sistema operativo basata su APK

Questa sezione include le piattaforme supportate e le funzionalità chiave per la raccolta di pacchetti APK basati sul sistema operativo. Per ulteriori informazioni, vedere [Alpine Package Keeper](#) sul [Alpine Linux sito Web](#).

Piattaforme supportate

Le seguenti sono le piattaforme supportate.

- Alpine Linux

Note

Per i sistemi APK basati, Amazon Inspector SBOM Generator raccoglie i metadati dei pacchetti dal file. [/lib/apk/db/](#)

Funzionalità principali

- Raccolta di nomi di pacchetto: estrae il nome di ogni pacchetto installato
- Raccolta di versioni: estrae la versione di ogni pacchetto installato
- Identificazione del pacchetto sorgente: identifica il pacchetto sorgente per ogni pacchetto installato

Esempio

Il seguente frammento è un esempio di file di database. APK

```
C:Q1J1boSJKrN4qkDcokr4zenpcWEXQ=
```

```
P:zlib
V:1.2.13-r1
A:x86_64
S:54253
I:110592
T:A compression/decompression Library
U:https://zlib.net/
L:Zlib
o:zlib
```

Raccolta di pacchetti del sistema operativo basata su DPKG

Questa sezione include le piattaforme supportate e le funzionalità chiave per la raccolta di pacchetti DPKG basati sul sistema operativo. Per ulteriori informazioni, vedere [Debian Package](#) sul Debian sito web.

Piattaforme supportate

Sono supportate le seguenti piattaforme.

- Debian
- Ubuntu

Note

Per i sistemi DPKG basati, Amazon Inspector SBOM Generator raccoglie i metadati dei pacchetti dal file. [/var/lib/dpkg/status](#)

Funzionalità principali

Di seguito sono riportate le caratteristiche principali dei pacchetti OS basati su sistemi operativi. DPKG

- Raccolta di nomi di pacchetto: estrae il nome di ogni pacchetto installato
- Raccolta di versioni: estrae la versione di ogni pacchetto installato
- [Identificazione del pacchetto sorgente](#): identifica il pacchetto sorgente per ogni pacchetto installato

Esempio

Il seguente frammento è un esempio di file. `/var/lib/dpkg/`

```
Package: zlib1g
Status: install ok installed
Priority: optional
Section: libs
Installed-Size: 168
Maintainer: Mark Brown <broonie@debian.org>
Architecture: amd64
Multi-Arch: same
Source: zlib
Version: 1:1.2.13.dfsg-1
Provides: libz1
Depends: libc6 (>= 2.14)
Breaks: libxml2 (<< 2.7.6.dfsg-2), texlive-binaries (<< 2009-12)
Conflicts: zlib1 (<= 1:1.0.4-7)
Description: compression library - runtime
  zlib is a library implementing the deflate compression method found
  in gzip and PKZIP. This package includes the shared library.
Homepage: http://zlib.net/
```

Raccolta di pacchetti del sistema operativo basata su RPM

Questa sezione include le piattaforme supportate e le funzionalità chiave per la raccolta di pacchetti RPM basati sul sistema operativo. Per ulteriori informazioni, vedere [RPM Package Manager](#) sul RPM sito Web.

Piattaforme supportate

Sono supportate le seguenti piattaforme.

- Alma Linux
- Amazon Linux
- CentOS
- Fedora
- OpenSUSE

- Oracle Linux
- PhotonOS
- RedHat Enterprise Linux
- Rocky Linux
- SUSE Linux Enterprise Server

Note

Per i sistemi RPM basati, Amazon Inspector SBOM Generator raccoglie i metadati dei pacchetti dal file. [/var/lib/rpm](#)

Funzionalità principali

Di seguito sono riportate le funzionalità principali per RPM le raccolte di pacchetti basati su sistemi operativi.

- Raccolta di nomi di pacchetto: estrae il nome di ogni pacchetto installato
- Raccolta di versioni: estrae la versione di ogni pacchetto installato
- [Identificazione del pacchetto sorgente](#): identifica il pacchetto sorgente per ogni pacchetto installato
- [Supporto Stream](#): estrae i metadati dello stream di ogni pacchetto installato

Esempio

Di seguito è riportato un esempio di frammento di file di RPM database.

```
/usr/lib/sysimage/rpm/rpmdb.sqlite  
/usr/lib/sysimage/rpm/Packages  
/usr/lib/sysimage/rpm/Packages.db  
/var/lib/rpm/rpmdb.sqlite  
/var/lib/rpm/Packages  
/var/lib/rpm/Packages.db
```

Raccolta di versioni del sistema operativo Windows

A differenza dei sistemi operativi basati su Linux, Windows non utilizza un sistema di gestione dei pacchetti per il sistema operativo stesso. Amazon Inspector SBOM Generator raccoglie solo le informazioni sulla versione del sistema operativo Windows. Per la scansione delle applicazioni Windows, utilizza invece lo scanner. windows-apps Lo windows-apps scanner raccoglie informazioni sulle applicazioni installate sui sistemi Windows. Per ulteriori informazioni, vedere [Applicazioni Microsoft raccolta di ecosistemi](#).

Funzionalità principali

- Raccolta di versioni del sistema operativo: estrae la versione del sistema operativo Windows dal registro di Windows. La versione del sistema operativo estratta viene utilizzata per il rilevamento delle vulnerabilità per il sistema operativo Windows.

Chiavi del registro e relativi valori

Le seguenti chiavi e valori del registro di Windows vengono utilizzati per raccogliere informazioni sul nome e sulla versione del sistema operativo.

- Chiave di registro

```
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion
```

- Valore del registro
 - ProductName— Nome ed edizione del sistema operativo "Windows Server 2025 Datacenter" (ad es.
 - CurrentMajorVersionNumber— la versione principale del sistema operativo
 - CurrentMinorVersionNumber— La versione secondaria del sistema operativo
 - CurrentBuild— Il numero di build del sistema operativo
 - UBR— Il numero di revisione del sistema operativo

Raccolta di pacchetti di immagini Chainguard

Questa sezione include le piattaforme supportate e le funzionalità chiave per la raccolta di pacchetti di Chainguard immagini. Per ulteriori informazioni, consulta [Immagini](#) sul Chainguard sito Web.

Piattaforme supportate

Sono supportate le seguenti piattaforme

- Wolfi Linux

Note

Per Chainguard le immagini, Amazon Inspector SBOM Generator raccoglie i metadati del pacchetto dal file. `/lib/apk/db/installed`

Funzionalità principali

Di seguito sono riportate le caratteristiche principali.

- Raccolta di nomi di pacchetto: estrae il nome di ogni pacchetto installato
- Raccolta di versioni: estrae la versione di ogni pacchetto installato
- Identificazione del pacchetto sorgente: identifica il pacchetto sorgente per ogni pacchetto installato

Esempio

Il seguente frammento è un esempio di file di Chainguard immagine.

```
P:wolfi-keys  
V:1-r8  
A:x86_64  
L:MIT  
T:Wolfi signing keyring  
o:wolfi-keys
```

Raccolta di pacchetti di immagini Distroless

Distroless contenitori sono immagini di contenitori che escludono i gestori di pacchetti, le shell e altre utilità nelle distribuzioni. Linux Distroless contenitori includono solo le dipendenze essenziali necessarie per eseguire l'applicazione e migliorare le prestazioni e la sicurezza.

Note

Per [Distrolessse immagini](#), Amazon Inspector SBOM Generator raccoglie i metadati del pacchetto dal file `/var/lib/dpkg/status.d`. Sono supportate solo le distribuzioni basate su Debian di Ubuntu esse. Queste possono essere identificate dal NAME campo nel `/etc/os-release` file system, che mostra "Debian" o "»Ubuntu.

Funzionalità principali

- Raccolta di nomi di pacchetto: estrae il nome di ogni pacchetto installato
- Raccolta di versioni: estrae la versione di ogni pacchetto installato

Esempio

Di seguito è riportato un esempio di file di Distroless immagine.

```
Package: tzdata
Version: 2021a-1+deb11u10
Architecture: all
Maintainer: GNU Libc Maintainers <debian-glibc@lists.debian.org>
Installed-Size: 3413
Depends: debconf (>= 0.5) | debconf-2.0
Provides: tzdata-bullseye
Section: localization
Priority: required
Multi-Arch: foreign
Homepage: https://www.iana.org/time-zones
Description: time zone and daylight-saving time data
 This package contains data required for the implementation of
 standard local time for many representative locations around the
 globe. It is updated periodically to reflect changes made by
 political bodies to time zone boundaries, UTC offsets, and
 daylight-saving rules.
```

Raccolta di pacchetti MiniMOS

Questa sezione include le piattaforme supportate e le funzionalità chiave per la raccolta di pacchetti di Minimus immagini. Per ulteriori informazioni, consultate il sito Web di [Minimus](#).

Piattaforme supportate

Sono supportate le seguenti piattaforme.

- Minimos

Note

Per Minimus le immagini, Amazon Inspector SBOM Generator raccoglie i metadati del pacchetto dal file. `/lib/apk/db/installed`

Funzionalità principali

Di seguito sono riportate le caratteristiche principali.

- Raccolta di nomi di pacchetto: estrae il nome di ogni pacchetto installato
- Raccolta di versioni: estrae il nome di ogni pacchetto installato
- Identificazione del pacchetto sorgente: identifica il pacchetto sorgente per ogni pacchetto installato

Quanto segue è un frammento di un Minimus file di immagine.

```
P:ca-certificates-bundle
V:20241121-r1
A:aarch64
L:MPL-2.0 AND MIT
T:
o:ca-certificates
```

Raccolta di dipendenze dal linguaggio di programmazione

Amazon Inspector SBOM Generator supporta diversi linguaggi e framework di programmazione, che costituiscono una raccolta solida e dettagliata di dipendenze. La generazione di un SBOM ti aiuta a comprendere la composizione del tuo software, in modo da poter identificare le vulnerabilità

e mantenere la conformità agli standard di sicurezza. Amazon Inspector SBOM Generator supporta i seguenti linguaggi di programmazione e formati di file.

Vai alla scansione delle dipendenze

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze e transitive	Bandiera privata	Ricorsivamente
Go	Go	go.mod	N/A	N/A	N/A	N/A	Si
		go.sum	N/A	N/A	N/A	N/A	Si
		Go Binaries	Si	N/A	N/A	N/A	Si
		GOMODCACH E	N/A	N/A	N/A	N/A	No

andare. mod/go.somma

Usa go.sum i file go.mod and per definire e bloccare le dipendenze nei Go progetti. Amazon Inspector SBOM Generator gestisce questi file in modo diverso in base alla versione della toolchain. Go

Funzionalità principali

- Raccoglie le dipendenze da go.mod (se la versione della Go toolchain è 1.17 o successiva)
- Raccoglie le dipendenze da go.sum (se la versione della Go toolchain è 1.17 o precedente)
- Analizza go.mod per identificare tutte le dipendenze e le versioni delle dipendenze dichiarate

Esempio go.mod file

Di seguito è riportato un esempio di file. go.mod

```
module example.com/project

go 1.17

require (
github.com/gin-gonic/gin v1.7.2
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123
)
```

Esempio `go.sum` file

Di seguito è riportato un esempio di file `go.sum`

```
github.com/gin-gonic/gin v1.7.2 h1:VZ7DdRl0sghbA6lVGSkX+UX02+J0aH7RbsNugG+FA8Q=
github.com/gin-gonic/gin v1.7.2/go.mod h1:ILZ1Ngh2f1pL1ASUj7gGk8lGFENC8cRTaN2ZhsBNbXU=
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123 h1:b6rCu+qHze
+BUsmC3CZzH8aNu8LzPZTVsNT0640ypSc=
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123/go.mod h1:K5Dkpb0Q4ewZW/
EzWlQphgJcUMBCzoWrLfd0VzpTGVQ=
```

Note

Ciascuno di questi file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Go Binaries

Amazon Inspector SBOM Generator estrae le dipendenze dai Go file binari compilati per garantire il codice in uso.

Note

Amazon Inspector SBOM Generator supporta l'acquisizione e la valutazione delle versioni della toolchain da file binari creati utilizzando il compilatore ufficiale. Go Go [Per ulteriori informazioni, consulta Download e installazione sul sito Web](#). Go Se si utilizza la Go toolchain

di un altro fornitore, ad esempio, la valutazione potrebbe non essere accurata a causa di potenziali differenze nella distribuzione e nella disponibilità dei metadati. Red Hat

Funzionalità principali

- Estrae le informazioni sulle dipendenze direttamente dai file binari Go
- Raccoglie le dipendenze incorporate nel file binario
- Rileva ed estrae la versione della Go toolchain utilizzata per compilare il file binario.

GOMODCACHE

Amazon Inspector SBOM Generator analizza la cache del Go modulo per raccogliere informazioni sulle dipendenze installate. Questa cache memorizza i moduli scaricati per garantire che le stesse versioni vengano utilizzate in build diverse.

Funzionalità principali

- Esegue la scansione della GOMODCACHE directory per identificare i moduli memorizzati nella cache
- Estrae i metadati dettagliati, inclusi i nomi dei moduli, le versioni e gli URL di origine

Struttura di esempio

Di seguito è riportato un esempio di GOMODCACHE struttura.

```
~/go/pkg/mod/  
### github.com/gin-gonic/gin@v1.7.2  
### golang.org/x/crypto@v0.0.0-20210616213533-5cf6c0f8e123
```

Note

Questa struttura produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione delle dipendenze in Java

Linguaggi o di programmazione	Programmi di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
Java	Maven	JavaApplicazioni compilate (.jar/.war/.ear) pom.xml	N/A N/A	N/A N/A	Sì Sì	N/A N/A	Sì Sì

Note

La nostra funzione di valutazione delle vulnerabilità supporta solo il repository Maven Central. Third-party I repository, ad esempio, non sono attualmente supportati JBoss Enterprise Maven Repository.

Amazon Inspector SBOM Generator esegue la scansione delle Java dipendenze analizzando applicazioni e file compilati. Java pom.xml Durante la scansione di applicazioni compilate, lo scanner genera hash SHA—1 per la verifica dell'integrità, estrae i file incorporati e analizza i file annidati. pom.properties pom.xml

Raccolta di hash SHA—1 (per file compilati in formato.jar, .war, .ear)

Amazon Inspector SBOM Generator cerca di raccogliere hash SHA—1 per tutti e .war file in un progetto per garantire l'.ear integrità .jar e la tracciabilità degli artefatti compilati. Java

Funzionalità principali

- Genera hash SHA—1 per tutti gli artefatti compilati Java

Esempio di artefatto

Di seguito è riportato un esempio di artefatto SHA—1.

```
{
  "bom-ref": "comp-52",
  "type": "library",
  "name": "jul-to-slf4j",
  "version": "2.0.6",
  "hashes": [
    {
      "alg": "SHA-1",
      "content": ""
    }
  ],
  "purl": "pkg:maven/jul-to-slf4j@2.0.6",
  "properties": [
    {
      "name": "amazon:inspector:sbom_generator:source_path",
      "value": "test-0.0.1-SNAPSHOT.jar/BOOT-INF/lib/jul-to-slf4j-2.0.6.jar"
    }
  ]
}
```

Note

Questo artefatto produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

pom.properties

Il `pom.properties` file viene utilizzato nei Maven progetti per archiviare i metadati del progetto, inclusi i nomi e le versioni dei pacchetti. Amazon Inspector SBOM Generator analizza questo file per raccogliere informazioni sul progetto.

Funzionalità principali

- Analizza ed estrae gli elementi, i gruppi di pacchetti e le versioni dei pacchetti

Esempio **pom.properties** file

Di seguito è riportato un esempio di un file `pom.properties`.

```
#Generated by Maven
#Tue Mar 16 15:44:02 UTC 2021

version=1.6.0
groupId=net.datafaker
artifactId=datafaker
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Escluso annidato **pom.xml** analisi

Se si desidera escludere l'`pom.xml` analisi durante la scansione di Java applicazioni compilate, utilizzare l'`--skip-nested-pomxml` argomento.

`pom.xml`

Il `pom.xml` file è il file di configurazione principale per i Maven progetti. Contiene informazioni sui progetti e sulle dipendenze dei progetti. Amazon Inspector SBOM Generator analizza `pom.xml` i file per raccogliere le dipendenze, scansiona i file autonomi nei repository e i file all'interno di file compilati. `.jar`

Funzionalità principali

- Analizza ed estrae gli elementi dei pacchetti, i gruppi di pacchetti e le versioni dei pacchetti dai file. `pom.xml`

Supportata Maven ambiti e tag

Le dipendenze vengono raccolte con i seguenti ambiti: Maven

- `compile`
- `fornito`
- `runtime`
- `test`
- `sistema`
- `importare`

Le dipendenze vengono raccolte con il seguente tag:Maven. `<optional>true</optional>`

Esempio `pom.xml` file con un ambito

Di seguito è riportato un esempio di `pom.xml` file con un ambito.

```
<dependency>
<groupId>jakarta.servlet</groupId>
<artifactId>jakarta.servlet-api</artifactId>
</version>6.0.0</version>
<scope>provided</scope>
</dependency>
<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>8.0.28</version>
<scope>runtime</scope>
</dependency>
```

Esempio `pom.xml` file senza ambito

Di seguito è riportato un esempio di pom.xml file senza ambito.

```
<dependency>
<groupId>com.fasterxml.jackson.core</groupId>
<artifactId>jackson-databind</artifactId>
<version>2.17.1</version>
</dependency>

<dependency>
<groupId>org.jenkins-ci.plugins</groupId>
<artifactId>plain-credentials</artifactId>
<version>183.va_de8f1dd5a_2b_</version>
</dependency>

<dependency>
<groupId>org.jenkins-ci.plugins</groupId>
<artifactId>jackson2-api</artifactId>
<version>2.15.2-350.v0c2f3f8fc595</version>
</dependency>
```

Note

Ciascuno di questi file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

JavaScript scansione delle dipendenze

linguaggio di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
Javascript	Node Modules	node_modules/	N/A	N/A	Si	Si	Si

linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
	NPM	*/package.json	N/A	Sì	N/A	N/A	No
	PNPM		N/A	Sì	N/A	N/A	No
	YARN	package-lock.json (v1, v2, and v3) / npm-shrinkwrap.json pnpm-lock.yaml yarn.lock	N/A	Sì	N/A	N/A	No

package.json

Il `package.json` file è un componente fondamentale dei progetti. Node.js Contiene metadati sui pacchetti installati. Amazon Inspector SBOM Generator analizza questo file per identificare i nomi e le versioni dei pacchetti.

Funzionalità principali

- Analizza la struttura dei file JSON per estrarre i nomi e le versioni dei pacchetti
- Identifica i pacchetti privati con valori privati

Esempio `package.json` file

Di seguito è riportato un esempio di un file `package.json`.

```
{
  "name": "arrify",
  "private": true,
  "version": "2.0.1",
  "description": "Convert a value to an array",
  "license": "MIT",
  "repository": "sindresorhus/arrify"
}
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

`package-lock.json`

Il `package-lock.json` file viene generato automaticamente da npm per bloccare le versioni esatte delle dipendenze installate per un progetto. Garantisce la coerenza negli ambienti memorizzando le versioni esatte di tutte le dipendenze e le relative sottodipendenze. Questo file può distinguere tra dipendenze regolari e dipendenze di sviluppo.

Funzionalità principali

- Analizza la struttura dei file JSON per estrarre i nomi e le versioni dei pacchetti
- Supporta il rilevamento delle dipendenze degli sviluppatori

Esempio `package-lock.json` file

Di seguito è riportato un esempio di un file `package-lock.json`.

```
"verror": {
  "version": "1.10.0",
  "resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
  "integrity": "sha1-0hBcoXBTr1XW4nDB+CiGguGNpAA=",
  "requires": {
    "assert-plus": "^1.0.0",
    "core-util-is": "1.0.2",
    "extsprintf": "^1.2.0"
  }
},
"wrappy": {
  "version": "1.0.2",
  "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",
  "integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",
  "dev": true
},
"yallist": {
  "version": "3.0.2",
  "resolved": "https://registry.npmjs.org/yallist/-/yallist-3.0.2.tgz",
  "integrity": "sha1-hFK0u36Dx8GI2AqcGoN8dz1ti7k="
}
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

npm-shrinkwrap.json

npm genera automaticamente `npm-shrinkwrap.json` file `package-lock.json` e file per bloccare le versioni esatte delle dipendenze installate per un progetto. Ciò garantisce la coerenza negli ambienti memorizzando le versioni esatte di tutte le dipendenze e sottodipendenze. I file distinguono tra dipendenze regolari e dipendenze di sviluppo.

Funzionalità principali

- Analizza `package-lock` le versioni 1, 2 e 3 della struttura del JSON file per estrarre il nome e la versione del pacchetto
- È supportato il rilevamento delle dipendenze degli sviluppatori (`package-lock.json` acquisisce le dipendenze di produzione e sviluppo, consentendo agli strumenti di identificare quali pacchetti vengono utilizzati negli ambienti di sviluppo)
- Al `npm-shrinkwrap.json` file viene data la priorità rispetto al file `package-lock.json`

Esempio

Di seguito è riportato un esempio di un file `package-lock.json`.

```
"verror": {
  "version": "1.10.0",
  "resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
  "integrity": "sha1-0hBcoXBTTr1XW4nDB+CiGguGNpAA=",
  "requires": {
    "assert-plus": "^1.0.0",
    "core-util-is": "1.0.2",
    "extsprintf": "^1.2.0"
  }
},
"wrappy": {
  "version": "1.0.2",
  "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",
  "integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",
  "dev": true
},
"yallist": {
  "version": "3.0.2",
  "resolved": "https://registry.npmjs.org/yallist/-/yallist-3.0.2.tgz",
  "integrity": "sha1-hFK0u36Dx8GI2AQcGoN8dz1ti7k="
}
}
```

pnpm-yaml.lock

Il `pnpm-lock.yaml` file viene generato da `pnpm` per mantenere un registro delle versioni di dipendenza installate. Inoltre, tiene traccia delle dipendenze di sviluppo separatamente.

Funzionalità principali

- Analizza la struttura dei file YAML per estrarre i nomi e le versioni dei pacchetti
- Supporta il rilevamento delle dipendenze degli sviluppatori

Esempio

Di seguito è riportato un esempio di un file `pnpm-lock.yaml`.

```
lockfileVersion: 5.3
importers:
  my-project:
    dependencies:
      lodash: 4.17.21
    devDependencies:
      jest: 26.6.3
    specifiers:
      lodash: ^4.17.21
      jest: ^26.6.3
  packages:
    /lodash/4.17.21:
      resolution:
        integrity: sha512-xyz
    engines:
      node: '>=6'
    dev: false
    /jest/26.6.3:
      resolution:
        integrity: sha512-xyz
    dev: true
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

yarn.lock

Amazon Inspector SBOM Generator cerca di raccogliere hash SHA—1 e `.war` file in un progetto per garantire l'integrità `.jar` e la tracciabilità degli artefatti compilati. Java

Funzionalità principali

- Genera hash SHA—1 per tutti gli artefatti compilati Java

Esempio di artefatto SHA—1

Di seguito è riportato un esempio di artefatto SHA—1.

```
"@ampproject/remapping@npm:^2.2.0":
  version: 2.2.0
  resolution: "@ampproject/remapping@npm:2.2.0"
  dependencies:
    "@jridgewell/gen-mapping": ^0.1.0
    "@jridgewell/trace-mapping": ^0.3.9
  checksum:
    d74d170d06468913921d72430259424b7e4c826b5a7d39ff839a29d547efb97dc577caa8ba3fb5cf023624e9af9d09
  languageName: node
  linkType: hard

"@babel/code-frame@npm:^7.0.0, @babel/code-frame@npm:^7.12.13, @babel/code-
frame@npm:^7.18.6, @babel/code-frame@npm:^7.21.4":
  version: 7.21.4
  resolution: "@babel/code-frame@npm:7.21.4"
  dependencies:
    "@babel/highlight": ^7.18.6
  checksum:
    e5390e6ec1ac58dcef01d4f18eaf1fd2f1325528661ff6d4a5de8979588b9f5a8e852a54a91b923846f7a5c681b217
  languageName: node
  linkType: hard
```

Note

Questo artefatto produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione

di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione delle dipendenze.NET

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
.NET	.NET Core	*.deps.json	N/A	N/A	N/A	N/A	Si
	Nuget	Packages.config	N/A	N/A	N/A	N/A	Si
	Nuget	packages.lock.json	N/A	N/A	Si	N/A	Si
	.NET	packages.lock.json	N/A	N/A	N/A	N/A	Si
			.csproj				

Packages.config

Il Packages.config file è un file XML utilizzato da una versione precedente di Nuget per gestire le dipendenze del progetto. Elenca tutti i pacchetti a cui fa riferimento il progetto, incluse versioni specifiche.

Funzionalità principali

- Analizza la struttura XML per estrarre gli ID e le versioni dei pacchetti

Esempio

Di seguito è riportato un esempio di un file Packages.config.

```
<?xml version="1.0" encoding="utf-8"? >
<packages>
<package id="FluentAssertions" version="5.4.1" targetFramework="net461" />
<package id="Newtonsoft.Json" version="11.0.2" targetFramework="net461" />
<package id="SpecFlow" version="2.4.0" targetFramework="net461" />
<package id="SpecRun.Runner" version="1.8.0" targetFramework="net461" />
<package id="SpecRun.SpecFlow" version="1.8.0" targetFramework="net461" />
<package id="SpecRun.SpecFlow.2-4-0" version="1.8.0" targetFramework="net461" />
<package id="System.ValueTuple" version="4.5.0" targetFramework="net461" />
</packages>
```

Note

Questo file produce un output che contiene l'URL del pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

*.deps.json

Il *.deps.json file viene generato dai .NET Core progetti e contiene informazioni dettagliate su tutte le dipendenze, inclusi percorsi, versioni e dipendenze di runtime. Questo file assicura che il runtime disponga delle informazioni necessarie per caricare le versioni corrette delle dipendenze.

Funzionalità principali

- Analizza la struttura JSON per dettagli completi sulle dipendenze
- Estrae i nomi e le versioni dei pacchetti in un elenco. `libraries`

Esempio `.deps.json` file

Di seguito è riportato un esempio di un file `.deps.json`.

```
{
  "runtimeTarget": {
```

```
    "name": ".NETCoreApp,Version=v7.0",
    "signature": ""
  },
  "libraries": {
    "sample-Nuget/1.0.0": {
      "type": "project",
      "serviceable": false,
      "sha512": ""
    },
    "Microsoft.EntityFrameworkCore/7.0.5": {
      "type": "package",
      "serviceable": true,
      "sha512": "sha512-
RXbRLHHPW2Z3pq8qcL5nQ6LPeo0yp8hasM5bd0Te8PiQi3RjWQR4tcbyY5XMqQ+oT09wA8/RLhZRn/
hnx1TDnQ==",
      "path": "microsoft.entityframeworkcore/7.0.5",
      "hashPath": "microsoft.entityframeworkcore.7.0.5.nupkg.sha512"
    },
  }
}
```

Note

Questo file produce un output che contiene l'URL del pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

packages.lock.json

Il `packages.lock.json` file viene utilizzato dalle versioni più recenti di Nuget per bloccare le versioni esatte delle dipendenze di un .NET progetto per garantire che le stesse versioni vengano utilizzate in modo coerente in ambienti diversi.

Funzionalità principali

- Analizza la struttura JSON per elencare le dipendenze bloccate
- Supporta dipendenze dirette e transitive
- Estrae il nome del pacchetto e le versioni risolte

Esempio `packages.lock.json` file

Di seguito è riportato un esempio di un file `packages.lock.json`.

```
{
  "version": 1,
  "dependencies": {
    "net7.0": {
      "Microsoft.EntityFrameworkCore": {
        "type": "Direct",
        "requested": "[7.0.5, )",
        "resolved": "7.0.5",
        "contentHash": "RXbRLHHWP2Z3pq8qcL5nQ6LPeo0yp8hasM5bd0Te8PiQi3RjWQR4tcbdY5XMqQ
+oT09wA8/RLhZRn/hnxlTDnQ==",
        "dependencies": {
          "Microsoft.EntityFrameworkCore.Abstractions": "7.0.5",
          "Microsoft.EntityFrameworkCore.Analyzers": "7.0.5",
          "Microsoft.Extensions.Caching.Memory": "7.0.0",
          "Microsoft.Extensions.DependencyInjection": "7.0.0",
          "Microsoft.Extensions.Logging": "7.0.0"
        }
      },
      "Newtonsoft.Json": {
        "type": "Direct",
        "requested": "[13.0.3, )",
        "resolved": "13.0.3",
        "contentHash": "HrC5BXdl00IP9zeV+0Z848QWPAoCr9P3bDEZguI+gkLcBKA0xix/tLEAAHC
+UvDNPv4a2d18l0ReHM0agPa+zQ==",
      },
      "Microsoft.Extensions.Primitives": {
        "type": "Transitive",
        "resolved": "7.0.0",
        "contentHash": "um1KU5kxcRp3CNUi8o/GrZtD4AI0XDk
+RLsyTjZ9QPok3ttLUelLKpilVPuaFT3TFj0hSibUAs0odb0aCDj3Q=="
      }
    }
  }
}
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

.csproj

Il `.csproj` file è scritto in XML e il file di progetto è per i progetti. .NET Include riferimenti a Nuget pacchetti, proprietà del progetto e configurazioni di build.

Funzionalità principali

- Analizza la struttura XML per estrarre i riferimenti ai pacchetti

Esempio .csproj file

Di seguito è riportato un esempio di un file `.csproj`.

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>net7.0</TargetFramework>
    <RootNamespace>sample_Nuget</RootNamespace>
    <ImplicitUsings>enable</ImplicitUsings>
    <Nullable>enable</Nullable>
    <RestorePackagesWithLockFile>true</RestorePackagesWithLockFile>
  </PropertyGroup>
  <ItemGroup>
  </ItemGroup>
  <ItemGroup>
    <PackageReference Include="Newtonsoft.Json" Version="13.0.3" />
    <PackageReference Include="Microsoft.EntityFrameworkCore" Version="7.0.5" />
  </ItemGroup>
</Project>
```

Esempio .csproj file

Di seguito è riportato un esempio di un file `.csproj`.

```
<PackageReference Include="ExamplePackage" Version="6.*" />
<PackageReference Include="ExamplePackage" Version="(4.1.3,)" />
<PackageReference Include="ExamplePackage" Version="(,5.0)" />
<PackageReference Include="ExamplePackage" Version="[1,3)" />
<PackageReference Include="ExamplePackage" Version="[1.3.2,1.5)" />
```

Note

Ciascuno di questi file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione delle dipendenze PHP

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze e transitive	Bandiera privata	Ricorsivamente
PHP	Composer	composer.lock	N/A	N/A	Si	N/A	Si
		/vendor/composer/installed.json	N/A	N/A	Si	N/A	Si

composer.lock

Il `composer.lock` file viene generato automaticamente quando si eseguono i comandi `composer install` o `composer update`. Questo file garantisce che le stesse versioni delle dipendenze siano installate in ogni ambiente. Ciò fornisce un processo di compilazione coerente e affidabile.

Funzionalità principali

- Analizza il formato JSON per i dati strutturati
- Estrae i nomi e le versioni delle dipendenze

Esempio `composer.lock` file

Di seguito è riportato un esempio di un file `composer.lock`.

```
{
"packages": [
  {
    "name": "nesbot/carbon",
    "version": "2.53.1",
    // TRUNCATED
  },
  {
    "name": "symfony/deprecation-contracts",
    "version": "v3.2.1",
    // TRUNCATED
  },
  {
    "name": "symfony/polyfill-mbstring",
    "version": "v1.27.0",
    // TRUNCATED
  }
]
// TRUNCATED
}
```

Note

Questo produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

vendor/composer//installed.json

Il `/vendor/composer/installed.json` file si trova nella `vendor/composer` directory e fornisce un elenco completo di tutti i pacchetti e le versioni dei pacchetti installati.

Funzionalità principali

- Analizza il formato JSON per i dati strutturati
- Estrae i nomi e la versione delle dipendenze

Esempio `//installed.json` `vendor/composer` file

Di seguito è riportato un esempio di un file `/vendor/composer/installed.json`.

```
{
  "packages": [
    {
      "name": "nesbot/carbon",
      "version": "2.53.1",
      // TRUNCATED
    },
    {
      "name": "symfony/deprecation-contracts",
      "version": "v3.2.1",
      // TRUNCATED
    },
    {
      "name": "symfony/polyfill-mbstring",
      "version": "v1.27.0",
      // TRUNCATED
    }
  ]
}
```

```
]
// TRUNCATED
}
```

Note

Questo file produce un output che contiene l'URL del pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione delle dipendenze in Python

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
Python	pip	requirements.txt	N/A	N/A	N/A	N/A	Sì
	Poetry	Poetry.lock	N/A	N/A	N/A	N/A	Sì
	Pipenv	Pipfile.lock	N/A	N/A	N/A	N/A	Sì
	uv	Pipfile.lock	N/A	Sì	N/A	N/A	Sì
	Egg/Wheel	uv.lock	N/A	N/A	N/A	N/A	Sì
			.egg-info/PKG-INFO .dist-info/	N/A	N/A	N/A	N/A

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze e transitive	Bandiera privata	Ricorsivamente
		METADATA					

requirements.txt

Il `requirements.txt` file è un formato ampiamente utilizzato nei Python progetti per specificare le dipendenze del progetto. Ogni riga di questo file include un pacchetto con i relativi vincoli di versione. Amazon Inspector SBOM Generator analizza questo file per identificare e catalogare accuratamente le dipendenze.

Funzionalità principali

- Supporta gli specificatori di versione (`==` e `=`)
- Supporta commenti e linee di dipendenza complesse

Note

Gli specificatori di versione `<=` e `=>` non sono supportati.

Esempio `requirements.txt` file

Di seguito è riportato un esempio di un file `requirements.txt`.

```
flask==1.1.2
requests==2.24.0
numpy==1.18.5
foo~=1.2.0
# Comment about a dependency
scipy. # invalid
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Pipfile.lock

Pipenv è uno strumento che offre il meglio di tutti i mondi del packaging (impacchettato, bloccato e sbloccato). `Pipfile.lock` Blocca le versioni esatte delle dipendenze per facilitare le build deterministiche. Amazon Inspector SBOM Generator legge questo file per elencare le dipendenze e le relative versioni risolte.

Funzionalità principali

- Analizza il formato JSON per la risoluzione delle dipendenze
- Supporta le dipendenze predefinite e di sviluppo

Esempio `Pipfile.lock` file

Di seguito è riportato un esempio di un file `Pipfile.lock`.

```
{
  "default": {
    "requests": {
      "version": "==2.24.0",
      "hashes": [
        "sha256:cc718bb187e53b8d"
      ]
    }
  },
  "develop": {
    "blinker": {
      "hashes": [
        "sha256:1779309f71bf239144b9399d06ae925637cf6634cf6bd131104184531bf67c01",
        "sha256:8f77b09d3bf7c795e969e9486f39c2c5e9c39d4ee07424be2bc594ece9642d83"
      ],
      "markers": "python_version >= '3.8'",

```

```
    "version": "==1.8.2"  
  }  
}  
}
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Poetry.lock

Poetry è uno strumento di gestione e pacchettizzazione delle dipendenze per Python. Il `Poetry.lock` file blocca le versioni esatte delle dipendenze per facilitare ambienti coerenti. Amazon Inspector SBOM Generator estrae informazioni dettagliate sulle dipendenze da questo file.

Funzionalità principali

- Analizza il formato TOML per i dati strutturati
- Estrae i nomi e le versioni delle dipendenze

Esempio **Poetry.lock** file

Di seguito è riportato un esempio di un file `Poetry.lock`.

```
[[package]]  
name = "flask"  
version = "1.1.2"  
description = "A simple framework for building complex web applications."  
category = "main"  
optional = false  
python-versions = ">=3.5"  
[[package]]  
name = "requests"  
version = "2.24.0"  
description = "Python HTTP for Humans."
```

```
category = "main"
optional = false
python-versions = ">=3.5"
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

uv.lock

uv è un gestore di Python pacchetti veloce scritto in Rust. Il `uv.lock` file blocca le versioni esatte delle dipendenze per facilitare ambienti coerenti. Amazon Inspector SBOM Generator estrae informazioni dettagliate sulle dipendenze da questo file.

Funzionalità principali

- Analizza il formato TOML per i dati strutturati `uv.lock`
- Estrae i nomi e le versioni delle dipendenze
- Supporta le dipendenze di sviluppo
- Raccoglie solo i pacchetti in cui l'origine è un registro

Esempio `uv.lock` file

Di seguito è riportato un esempio di un file `uv.lock`.

```
version = 1
requires-python = ">=3.12"

[[package]]
name = "flask"
version = "3.1.0"
source = { registry = "https://pypi.org/simple" }
dependencies = [
  { name = "blinker" },
```

```
{ name = "click" },
{ name = "itsdangerous" },
{ name = "jinja2" },
{ name = "markupsafe" },
{ name = "werkzeug" },
]

[[package]]
name = "pytest"
version = "8.3.4"
source = { registry = "https://pypi.org/simple" }
dependencies = [
  { name = "iniconfig" },
  { name = "packaging" },
  { name = "pluggy" },
]

[package.dev-dependencies]
dev = [
  { name = "pytest" },
]
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Egg/Wheel

Per i pacchetti Python installati a livello globale, Amazon Inspector SBOM Generator supporta l'analisi dei file di metadati presenti nelle directory `and. .egg-info/PKG-INFO .dist-info/METADATA`. Questi file forniscono metadati dettagliati sui pacchetti installati.

Funzionalità principali

- Estrae il nome e la versione del pacchetto
- Supporta sia i formati a uovo che a forma di ruota

Percorsi di scansione predefiniti estesi (Sbomgen 1.13.0 e versioni successive)

In Amazon Inspector SBOM Generator versione 1.13.0 e successive, i percorsi di scansione localhost predefiniti dello scanner sono stati ampliati per includere le installazioni a livello utente () e le pip installazioni prodotte dal comando. `pip install --user uv tool install` Ciò consente il rilevamento di pacchetti Python installati all'esterno del sistema `site-packages`, inclusi framework e strumenti come CrewAI, Aider, GPT Engineer e Open Interpreter, i cui metodi di installazione consigliati producono installazioni in queste posizioni (ad esempio,). `uv tool install crewai`

Percorsi di scansione predefiniti aggiunti:

- Linux—`/home/*/local/lib/python*/site-packages/`, `/home/*/local/share/uv/tools/*/lib/python*/site-packages/` (con `/root/` equivalenti per i contenitori eseguiti come root).
- Windows – `%APPDATA%\Python\Python*\Lib\site-packages\`, `%LOCALAPPDATA%\uv\tools*\Lib\site-packages\`.

Per i prefissi di Python installazione non predefiniti su qualsiasi piattaforma, utilizzate l' `--path` argomento per puntare lo scanner verso la posizione di installazione.

Esempio **PKG-INFO/METADATA** file

Di seguito è riportato un esempio di un file **PKG-INFO/METADATA**.

```
Metadata-Version: 1.2
Name: Flask
Version: 1.1.2
Summary: A simple framework for building complex web applications.
Home-page: https://palletsprojects.com/p/flask/
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una

distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione delle dipendenze con Ruby

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per il toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
Ruby	Bundler	Gemfile.lock	N/A	N/A	Si	N/A	Si
		.gemspec	N/A	N/A	N/A	N/A	Si
		global installed Gems	N/A	N/A	N/A	N/A	Si

Gemfile.lock

Il `Gemfile.lock` file blocca le versioni esatte di tutte le dipendenze per garantire che le stesse versioni vengano utilizzate in ogni ambiente.

Funzionalità principali

- Analizza il `Gemfile.lock` file in base alle dipendenze e alle versioni delle dipendenze
- Estrae i nomi dettagliati dei pacchetti e le versioni dei pacchetti

Esempio **Gemfile.lock** file

Di seguito è riportato un esempio di un file `Gemfile.lock`.

```
GEM
```

```
remote: https://rubygems.org/  
specs:  
ast (2.4.2)  
awesome_print (1.9.2)  
diff-lcs (1.5.0)  
json (2.6.3)  
parallel (1.22.1)  
parser (3.2.2.0)  
nokogiri (1.16.6-aarch64-linux)
```

Note

Questo file produce un output che contiene l'URL del pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

.gemspec

Il `.gemspec` file è un RubyGem file contenente metadati su una gemma. Amazon Inspector SBOM Generator analizza questo file per raccogliere informazioni dettagliate su una gemma.

Funzionalità principali

- Analizza ed estrae il nome e la versione della gemma

Note

La specifica di riferimento non è supportata.

Esempio `.gemspec` file

Di seguito è riportato un esempio di un file `.gemspec`.

```
Gem::Specification.new do |s|  
s.name = "generategem"
```

```
s.version      = "2.0.0"
s.date         = "2020-06-12"
s.summary      = "generategem"
s.description  = "A Gemspec Builder"
s.email        = "edersondeveloper@gmail.com"
s.files        = ["lib/generategem.rb"]
s.homepage     = "https://github.com/edersonferreira/generategem"
s.license      = "MIT"
s.executables = ["generategem"]
s.add_dependency('colorize', '~> 0.8.1')
end
```

```
# Not supported
```

```
Gem::Specification.new do |s|
  s.name      = &class1
  s.version   = &foo.bar.version
```

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Gemme installate a livello globale

Amazon Inspector SBOM Generator supporta la scansione di gem installate a livello globale, che si trovano in directory standard, come `/usr/local/lib/ruby/gems/<ruby_version>/gems/` Amazon ECR e Lambda. EC2/Amazon `ruby/gems/<ruby_version>/gems/` Questo assicura che tutte le dipendenze installate a livello globale siano identificate e catalogate.

Funzionalità principali

- Identifica e analizza tutte le gem installate a livello globale in directory standard
- Estrae i metadati e le informazioni sulla versione per ogni gem installata a livello globale

Esempio di struttura di directory

Di seguito è riportato un esempio di struttura di directory.

```
.
### /usr/local/lib/ruby/3.5.0/gems/
### actrivesupport-6.1.4
### concurrent-ruby-1.1.9
### i18n-1.8.10
```

Note

Questa struttura produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione delle dipendenze da Rust

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
Rust	Cargo.toml	Cargo.toml	N/A	N/A	N/A	N/A	Sì
			N/A	N/A	Sì	N/A	Sì
	Cargo.lock	Sì	N/A	N/A	N/A	Sì	
	Rust binary (built with						

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze e transitive	Bandiera privata	Ricorsivamente
		cargo-auditabile)					

Cargo.toml

Il `Cargo.toml` file è il file manifesto dei progetti. Rust

Funzionalità principali

- Analizza ed estrae il `Cargo.toml` file per identificare il nome e la versione del pacchetto del progetto.

Esempio **Cargo.toml** file

Di seguito è riportato un esempio di un file `Cargo.toml`.

```
[package]
name = "wait-timeout"
version = "0.2.0"
description = "A crate to wait on a child process with a timeout specified across Unix and\nWindows platforms.\n"
homepage = "https://github.com/alexcrichon/wait-timeout"
documentation = "https://docs.rs/wait-timeout"
readme = "README.md"
categories = ["os"]
license = "MIT/Apache-2.0"
repository = "https://github.com/alexcrichon/wait-timeout"
[target."cfg(unix)".dependencies.libc]
version = "0.2"
[badges.appveyor]
repository = "alexcrichon/wait-timeout"
```

Note

Questo file produce un output che contiene l'URL del pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Cargo.lock

Il `Cargo.lock` file blocca le versioni dipendenti per garantire che vengano utilizzate le stesse versioni ogni volta che viene creato un progetto.

Funzionalità principali

- Analizza il `Cargo.lock` file per identificare tutte le dipendenze e le versioni delle dipendenze.

Esempio **Cargo.lock** file

Di seguito è riportato un esempio di un file `Cargo.lock`.

```
# This file is automatically @generated by Cargo.
# It is not intended for manual editing.
[[package]]
name = "adler32"
version = "1.0.3"
source = "registry+https://github.com/rust-lang/crates.io-index"

[[package]]
name = "aho-corasick"
version = "0.7.4"
source = "registry+https://github.com/rust-lang/crates.io-index"
```

Note

Questo file produce un output che contiene l'URL del pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una

distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

File binari Rust con cargo-auditable

Amazon Inspector SBOM Generator raccoglie le dipendenze dai Rust file binari creati con la libreria. `cargo-auditable` Ciò fornisce informazioni aggiuntive sulle dipendenze abilitando l'estrazione delle dipendenze dai binari compilati.

Funzionalità principali

- Estrae le informazioni sulle dipendenze direttamente dai Rust file binari creati con la libreria `cargo-auditable`
- Recupera i metadati e le informazioni sulla versione per le dipendenze incluse nei file binari

Note

Questo file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Scansione rapida delle dipendenze

Linguaggi o di programmazione	Programma di gestione dei pacchetti	Artefatti supportati	Supporto per la toolchain	Dipendenze di sviluppo	Dipendenze transitive	Bandiera privata	Ricorsivamente
Swift	Swift Package Manager	Package.resolved (v1, v2, and v3)	N/A	N/A	N/A	N/A	Sì

Package.resolved

Il `Package.resolved` file viene generato da Swift Package Manager per bloccare le versioni esatte delle dipendenze per un Swift progetto. Amazon Inspector SBOM Generator analizza questo file per raccogliere i nomi delle dipendenze, le versioni e le informazioni del repository di origine. Sono supportate le versioni 1, 2 e 3 dello schema del formato. `Package.resolved`

Note

Amazon Inspector SBOM Generator non genera il file. `Package.resolved` Se il progetto non contiene già un `Package.resolved` file, è necessario generarlo prima della scansione eseguendolo `swift package resolve` nella directory del progetto.

Artefatti supportati

- `Package.resolved`(versione dello schema 1)
- `Package.resolved`(versione dello schema 2)
- `Package.resolved`(versione dello schema 3)

Funzionalità principali

- Estrae i nomi delle dipendenze e le versioni bloccate dai pacchetti bloccati
- Deriva lo spazio dei nomi del pacchetto dall'URL del repository di origine (ad esempio,) `github.com/Alamofire`
- Ignora i pin riservati ai soli rami che non hanno una versione risolta
- Filtra il controllo locale del codice sorgente e le dipendenze del file system
- Esclude `Package.resolved` i file all'interno `.build/` delle directory per impedire la raccolta delle dipendenze dalla cache di build di Swift Package Manager

Note

Il `Package.resolved` file è indipendente dalla piattaforma. Amazon Inspector SBOM Generator rileva questo file in modo ricorsivo su, e. Linux macOS Windows

Esempio **Package.resolved** file (v2)

Di seguito è riportato un esempio di `Package.resolved` file della versione 2.

```
{
  "pins" : [
    {
      "identity" : "alamofire",
      "kind" : "remoteSourceControl",
      "location" : "https://github.com/Alamofire/Alamofire.git",
      "state" : {
        "revision" : "a1b2c3...",
        "version" : "5.8.1"
      }
    },
    {
      "identity" : "swift-argument-parser",
      "kind" : "remoteSourceControl",
      "location" : "https://github.com/apple/swift-argument-parser.git",
      "state" : {
        "revision" : "d4e5f6...",
        "version" : "1.3.0"
      }
    }
  ],
  "version" : 2
}
```

Esempio **Package.resolved** file (v1)

Di seguito è riportato un esempio di `Package.resolved` file della versione 1.

```
{
  "object": {
    "pins": [
      {
        "package": "Alamofire",
        "repositoryURL": "https://github.com/Alamofire/Alamofire.git",
        "state": {
          "branch": null,

```

```
        "revision": "a1b2c3...",
        "version": "5.8.1"
    }
}
],
},
"version": 1
}
```

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per una dipendenza di Swift Package Manager.

```
pkg:swift/github.com/Alamofire/Alamofire@5.8.1
```

Note

Ciascuno di questi file produce un output che contiene l'URL di un pacchetto. Questo URL può essere utilizzato per specificare informazioni sui pacchetti software durante la generazione di una distinta base del software e può essere incluso nell'[ScanSbomAPI](#). Per ulteriori informazioni, vedere [package-url sul sito Web](#). GitHub

Artefatti non supportati

Questa sezione descrive gli artefatti non supportati.

Java

[Il generatore Amazon Inspector SBOM Generator supporta solo il rilevamento delle vulnerabilità per le dipendenze provenienti dal repository principale. Maven](#) Gli Maven archivi privati o personalizzati, come e, non sono supportati. Red Hat Maven Jenkins Per un rilevamento accurato delle vulnerabilità, assicurati che Java le dipendenze vengano estratte dal repository principale. Maven Le dipendenze da altri repository non verranno coperte nelle scansioni di vulnerabilità.

JavaScript

pacchetti esbuild

Per i esbuild pacchetti ridotti, Amazon Inspector SBOM Generator non supporta la scansione delle dipendenze per i progetti in uso. esbuild Le mappe di origine generate da esbuild non includono metadati sufficienti (nomi e versioni delle dipendenze) necessari per una generazione accurata. Sbmngen Per risultati affidabili, scansiona i file di progetto originali, come `node_modules/directory/package-lock.json`, prima del processo di raggruppamento.

`package.json`

Amazon Inspector SBOM Generator non supporta la scansione del file `package.json` a livello di root per informazioni sulle dipendenze. Questo file specifica solo i nomi dei pacchetti e gli intervalli di versioni, ma non include le versioni dei pacchetti completamente risolte. Per risultati di scansione accurati, utilizzate `package.json` o altri file di blocco, come `yarn.lock` e `pnpm.lock`, che includono versioni risolte.

Dotnet

Quando si utilizzano versioni mobili o intervalli di versioni `PackageReference`, diventa più difficile determinare l'esatta versione del pacchetto utilizzata in un progetto senza eseguire la risoluzione del pacchetto. Le versioni e gli intervalli di versioni fluttuanti consentono agli sviluppatori di specificare un intervallo di versioni del pacchetto accettabili anziché una versione fissa.

Binari Go

Amazon Inspector SBOM Generator non esegue la scansione di Go file binari creati con flag di build configurati per escludere l'ID di build. Questi flag di build impediscono ad Amazon Inspector SBOM Generator di mappare accuratamente il file binario alla sua fonte originale. I Go file binari non chiari non sono supportati a causa dell'impossibilità di estrarre le informazioni sui pacchetti. Per una scansione accurata delle dipendenze, assicurati che i Go file binari siano compilati con le impostazioni predefinite, incluso l'ID di build.

Binari Rust

[Amazon Inspector SBOM Generator analizza i Rust file binari solo se questi sono stati creati utilizzando la libreria `cargo-auditable`.](#) Rusti file binari che non utilizzano questa libreria non dispongono dei metadati necessari per un'estrazione accurata delle dipendenze. Amazon Inspector SBOM Generator estrae la versione compilata della Rust toolchain a partire dalla Rust 1.7.3, ma solo per i file binari in un ambiente. Linux Per una scansione completa, crea file binari utilizzando `cargo-auditable`. Rust Linux

Note

Il rilevamento delle vulnerabilità per la Rust toolchain stessa non è supportato, anche se la versione della toolchain viene estratta.

Raccolta completa di ecosistemi Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator è uno strumento per creare una distinta base del software (SBOM) ed eseguire la scansione delle vulnerabilità per i pacchetti supportati dai sistemi operativi e dai linguaggi di programmazione. Supporta la scansione di vari ecosistemi oltre ai sistemi operativi principali, garantendo un'analisi solida e dettagliata dei componenti dell'infrastruttura. Generando una SBOM, è possibile comprendere la composizione degli stack tecnologici moderni, identificare le vulnerabilità nei componenti dell'ecosistema e ottenere visibilità sul software di terze parti.

Ecosistemi supportati

La raccolta di ecosistemi estende la generazione di SBOM oltre ai pacchetti installati tramite gestori di pacchetti del sistema operativo. Ciò avviene attraverso la raccolta di applicazioni distribuite con metodi alternativi, come l'installazione manuale. Il generatore SBOM di Amazon Inspector supporta la scansione per i seguenti ecosistemi:

Ecosistemi	Applicazioni
7-Zip	7-Ziparchiviatore (versione 21.07 e successive)
Amazon Q Developer	Amazon Q Developer CLI Amazon Q VS Code extension Amazon Q JetBrains plugin
Anthropic	Claude Code CLI
Anysphere	Cursor
Apache	Apache Cassandra

Ecosistemi	Applicazioni
	Apache httpd Apache Struts Apache tomcat
Atlassian	Jira Core Confluence Jira Software Jira Service Management
Codeium	Windsurf
Conda	Ambienti Miniconda Ambienti Anaconda Ambienti Miniforge Ambienti Mambaforge
Curl	Curl Libcurl
Docker binaries	docker dockerd containerd runc
Elasticsearch	Elasticsearch

Ecosistemi	Applicazioni
Generic AI/ML Models	.gguf – GGUF .safetensors – Safetensors .onnx – ONNX .pt – PyTorch .pth – PyTorch .h5 – Keras / HDF5
GitHub	GitHub Copilot CLI GitHub Copilot VS Code extension GitHub Copilot JetBrains plugin
Google	Chrome
HuggingFace	HuggingFace CLI Models Cache
Java	JDK JRE Amazon Corretto
Jenkins	Jenkins(versione 2.400.* e successive)
Kiro	Kiro CLI
MariaDB e MySQL	MariaDB Server(10.6+, 11.x, 12.x) Oracle MySQL Server Server(8,0, 8,4, 9,4+)

Ecosistemi	Applicazioni
Microsoft applications	PowerShell NuGet CLI Visual Studio Code Microsoft Edge SharePoint Server Microsoft Defender Exchange Server Visual Studio .NET Core Runtime .NET Framework ASP.NET Core Runtime Microsoft Teams Outlook for Windows Microsoft Office Microsoft 365
Microsoft SQL Server	Microsoft SQL Server
MongoDB	MongoDB Server(7,0+, 8,0+)
Nginx	Nginx
Node	Node
Node.JS	node

Ecosistemi	Applicazioni
Ollama	Ollama Ollama Model Collector
OpenSSH	OpenSSH(versioni 9 e 10)
OpenSSL	OpenSSL
Oracle	Oracle Database Server
PHP	PHP(versione 8.1 e successive)
Redis	Redis(versione 7.2 e successive)
WordPress	core plugin theme
Zed Industries	Zed

7-Zip collezione ecosistemica

Applicazioni supportate

- 7 Zip archiver (versione 21.07 o successiva)

Funzionalità principali

- Esamina i 7-Zip file binari per estrarre le informazioni sulla versione incorporata.

Note

In particolare, cerca il valore della versione del prodotto dal file binario.

Piattaforme supportate: Windows

- C:/Program Files/7-Zip/7z.exe
- C:/Program Files/7-Zip/7za.exe
- C:/Program Files/7-Zip/7zz.exe
- C:/Program Files/7-Zip/7zr.exe
- C:/Program Files (x86)/7-Zip/7z.exe
- C:/Program Files (x86)/7-Zip/7za.exe
- C:/Program Files (x86)/7-Zip/7zz.exe
- C:/Program Files (x86)/7-Zip/7zr.exe

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per 7-Zip.

```
pkg:generic/7zip/7zip@25.01
```

Amazon Q Developer raccolta di ecosistemi

Applicazioni supportate

- Amazon Q Developer CLI
- Amazon Q VS Code extension
- Amazon Q JetBrains plugin

Funzionalità principali

- Esamina il file binario q CLI per la stringa di identità AmazonQ-For-CLI incorporata ed estrae la versione AWS dai metadati dell'agente utente di Tooling.
- Per le estensioni IDE, analizza package.json (VS Code) e META-INF/plugin.xml () per estrarre la versione. JetBrains

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux and macOS

- `~/.local/bin/q`
- `/usr/local/bin/q`
- `~/.vscode/extensions/amazonwebservices.amazon-q-vscode-*/`
- `<IDE>/plugins/amazon-q/META-INF/plugin.xml`

Windows

- `%APPDATA%\npm\q.cmd`
- `%USERPROFILE%\vscode\extensions\amazonwebservices.amazon-q-vscode-*\`
- `%APPDATA%\JetBrains\<IDE>\plugins\amazon-q\`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per Amazon Q Developer.

```
Sample PURL: pkg:generic/amazon/amazon-q@1.19.7?distro=linux
```

Apache raccolta di ecosistemi

Questa sezione fornisce dettagli su Apache Cassandra, Apache httpd, Apache Struts, e Apache tomcat applicazioni.

Apache Cassandra

Applicazioni supportate

- Apache Cassandra

Funzionalità principali

- Rileva i `apache-cassandra-<specific.version>.jar` file e li decomprime per estrarre la stringa della versione dalla `Implementation-Version` voce all'interno del file. `META-INF/MANIFEST.MF`

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- /usr/share/cassandra/lib/
- /opt/cassandra/lib/

macOS

- /opt/homebrew/Cellar/cassandra/
- /usr/local/Cellar/cassandra/

Windows

- /Program Files/cassandra/lib/
- /Program Files/apache-cassandra/lib/
- /Program Files (x86)/cassandra/lib/
- /Program Files (x86)/apache-cassandra/lib/

Esempio **apache-cassandra - versione specifica**. < > jar/META-INF/MANIFEST.MF file

Di seguito è riportato un esempio di contenuto all'interno di un `apache-cassandra-<specific.version>.jar/META-INF/MANIFEST.MF` file.

```
//truncated

Manifest-Version: 1.0
Implementation-Title: Cassandra
Implementation-Version: 4.1.3
Implementation-Vendor: Apache

//truncated
```

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un'Apache Cassandra applicazione.

```
Sample PURL: pkg:generic/apache/cassandra@4.1.3
```

Apache httpd

Applicazioni supportate

- Apache httpd

Note

La valutazione delle vulnerabilità si applica solo a Apache httpd versione 2.0 e successive.

Funzionalità principali

- Analizza il `/include/ap_release.h` file per estrarre le macro di installazione, che contengono stringhe di identificazione principali, stringhe di identificatori minori e stringhe di identificazione delle patch.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Unix

- `/usr/local/apache2/include/`

Windows

- `/Apache24/include/`
- `/Program Files/Apache24/include/`
- `/Program Files (x86)/Apache24/include/`

Esempio `ap_release.h` file

Di seguito è riportato un esempio di contenuto all'interno di un file. `ap_release.h`

```
//truncated

#define AP_SERVER_BASEVENDOR "Apache Software Foundation"
#define AP_SERVER_BASEPROJECT "Apache HTTP Server"
#define AP_SERVER_BASEPRODUCT "Apache"

#define AP_SERVER_MAJORVERSION_NUMBER 2
#define AP_SERVER_MINORVERSION_NUMBER 4
#define AP_SERVER_PATCHLEVEL_NUMBER 1
#define AP_SERVER_DEVBUILD_BOOLEAN 0

//truncated
```

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un'Apache `httpd` applicazione.

```
Sample PURL: pkg:generic/apache/httpd@2.4.1
```

Apache Struts

Applicazioni supportate

- Apache Struts

Funzionalità principali

- Rileva i `struts2-core-<specific.version>.jar` file (in genere distribuiti all'interno della `WEB-INF/lib/` directory di un'applicazione Web su un server di Java applicazioni) e li decomprime per estrarre la stringa della versione dalla voce all'interno del `Implementation-Version` file. `META-INF/MANIFEST.MF`

Piattaforme supportate

Amazon Inspector SBOM Generator analizza le installazioni in percorsi di server Java applicativi comuni su più piattaforme:

Linux

- /opt/tomcat/
- /usr/share/tomcat/
- /var/lib/tomcat/
- /usr/local/tomcat/
- /opt/wildfly/
- /opt/jboss/
- /opt/jetty/

Windows

- /Program Files/Apache Software Foundation/
- /Program Files (x86)/Apache Software Foundation/

Esempio **struts2-core - versione specifica.** < > **jar/META-INF/MANIFEST.MF** file

Di seguito è riportato un esempio di contenuto all'interno di un `struts2-core-<specific.version>.jar/META-INF/MANIFEST.MF` file.

```
//truncated

Manifest-Version: 1.0
Implementation-Title: Struts 2 Core
Implementation-Version: 6.4.0
Implementation-Vendor: Apache Software Foundation

//truncated
```

Esempio PURL

Di seguito è riportato un esempio di URL di pacchetto per un'Apache Struts applicazione.

```
Sample PURL: pkg:generic/apache/struts@6.4.0
```

Tomcat Apache

Applicazioni supportate

- Apache tomcat

Note

La valutazione delle vulnerabilità si applica solo alla Apache tomcat versione 9.0 e successive.

Funzionalità principali

- Decodifica il `catalina.jar` file per estrarre le macro di installazione all'interno del `META-INF/MANIFEST.MF` file, che contiene la stringa della versione.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `/opt/tomcat/lib/`
- `/usr/share/tomcat/lib`
- `/var/lib/tomcat/lib/`

macOS

- `/Library/Tomcat/lib/`
- `/usr/local/tomcat/lib`

Windows

- /Program Files/Apache Software Foundation
- /Program Files (x86)/Apache Software Foundation/

Esempio `catalina.jar/META-INF/MANIFEST.MF` file

Di seguito è riportato un esempio di contenuto all'interno di un `catalina.jar/META-INF/MANIFEST.MF` file.

```
//truncated

Implementation-Title: Apache Tomcat
Implementation-Vendor: Apache Software Foundation
Implementation-Version: 10.1.31

//truncated
```

Esempio PURL

Di seguito è riportato un esempio di URL di pacchetto per un'Apache tomcat applicazione.

```
Sample PURL: pkg:generic/apache/tomcat@10.1.31
```

Atlassian raccolta di ecosistemi

Questa sezione fornisce dettagli sui prodotti e le applicazioni Atlassian server.

Prodotti Atlassian Server

Applicazioni supportate

- Jira Core
- Confluence

Funzionalità principali

- Jira Core— Analizza le proprietà POM di Maven per estrarre le informazioni sulla versione.
atlassian-jira-webapp
- Confluence— Analizza le proprietà POM di Maven per estrarre le informazioni sulla versione.
confluence-webapp

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni:

Linux

- /opt/atlassian/jira/atlassian-jira/META-INF/maven/com.atlassian.jira/atlassian-jira-webapp/pom.properties
- /opt/atlassian/confluence/confluence/META-INF/maven/com.atlassian.confluence/confluence-webapp/pom.properties

Esempio: PURL

Di seguito sono riportati alcuni esempi di URL di pacchetti per prodotti Atlassian server.

```
// Jira Core
pkg:generic/atlassian/jira-core@10.0.1?distro=linux

// Confluence
pkg:generic/atlassian/confluence@9.2.7?distro=linux
```

Applicazioni Atlassian

Applicazioni supportate

- Jira Software
- Jira Service Management

Funzionalità principali

- Jira Software— Rileva tramite `jira-software-application` JAR ed estrae la versione dalle proprietà POM di Maven.
- Jira Service Management— Rileva tramite `jira-servicedesk-application` JAR ed estrae la versione dalle proprietà POM di Maven.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni:

Linux

- `/opt/atlassian/jira/atlassian-jira/WEB-INF/application-installation/jira-software-application/jira-software-application-*.jar`
- `/opt/atlassian/jira/atlassian-jira/WEB-INF/application-installation/jira-servicedesk-application/jira-servicedesk-application-*.jar`

Esempio: PURL

Di seguito sono riportati alcuni esempi di URL di pacchetti per Atlassian le applicazioni.

```
// Jira Software
pkg:generic/atlassian/jira-software@10.3.9?distro=linux

// Jira Service Management
pkg:generic/atlassian/jira-service-management@10.3.9?distro=linux
```

Codice Claude collezione ecosistemica

Applicazioni supportate

- Claude Code CLI(Antropico)

Funzionalità principali

- Claude Code è distribuito come pacchetto npm (`@anthropic-ai/claude-code`). Lo scanner rileva il file binario della `claude CLI` e lo `package.json` risolve in relazione ad esso utilizzando

lo standard npm per il layout. `<prefix>/bin/<binary> <prefix>/lib/node_modules/@anthropic-ai/claude-code/package.json`

- La versione viene letta dal campo di `Version: package.json`

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux and macOS

- `/usr/bin/claude`
- `/usr/local/bin/claude`
- `~/.local/bin/claude`
- `~/.npm-global/bin/claude`

Windows

- `%APPDATA%\npm\claude.cmd`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per Claude Code.

```
Sample PURL: pkg:generic/anthropic/claude-code@1.0.5?distro=linux
```

Conda raccolta di ecosistemi

Applicazioni supportate

- Ambienti Miniconda
- Ambienti Anaconda
- Ambienti Miniforge
- Ambienti Mambaforge

Funzionalità principali

- Analizza i record dei conda-meta/*.json pacchetti scritti da conda e mamba in ogni ambiente conda.
- [Estrae il nome del pacchetto, la versione, la stringa di compilazione, la subdir \(piattaforma di destinazione\) e il tipo di archivio \(condaortar.bz2\) secondo CEP 26.](#)
- Genera gli URL dei conda Package.
- Filtra i file macOS resource-fork (._*) e salta i record con un nome o una versione vuoti.

Piattaforme supportate

Per le scansioni localhost, Amazon Inspector SBOM Generator analizza le directory conda-meta nei seguenti percorsi di installazione, predefiniti e per utente. Per le scansioni di contenitori, directory e archivi, Amazon Inspector SBOM Generator conda-meta/*.json rileva i record in qualsiasi punto dell'albero.

Linux

- /opt/conda
- /opt/miniconda3, /opt/anaconda3, /opt/miniforge3
- /root/miniconda3, /root/anaconda3, /root/miniforge3
- /home/<user>/miniconda3, /home/<user>/anaconda3, /home/<user>/miniforge3
- ~/.conda/envs/<env>

macOS

- /opt/miniconda3, /opt/anaconda3
- /opt/homebrew/anaconda3, /usr/local/anaconda3
- /opt/homebrew/Caskroom/miniconda/base, /opt/homebrew/Caskroom/miniforge/base
- /Users/<user>/miniconda3, /Users/<user>/anaconda3, /Users/<user>/miniforge3
- ~/.conda/envs/<env>

Windows

- C:\ProgramData\miniconda3, C:\ProgramData\anaconda3, C:\ProgramData\miniforge3
- C:\Users\\miniconda3, C:\Users\\anaconda3, C:\Users\\miniforge3
- C:\Users\\.conda\envs\

Esempio **conda-meta** percorso di registrazione

Di seguito è riportato un esempio di percorso per un record di conda pacchetto.

```
/opt/miniforge3/conda-meta/numpy-1.24.0-py311h64a7726_0.json
```

Esempio: PURL

Di seguito sono riportati alcuni esempi di URL di pacchetto per un pacchetto e un pacchetto specifici della piattaformaconda. noarch conda

```
pkg:conda/numpy@1.24.0?build=py311h64a7726_0&subdir=linux-64&type=conda  
pkg:conda/flask@3.0.0?build=pyhd8ed1ab_0&subdir=noarch&type=conda
```

Curl raccolta ecosistemica

Questa sezione fornisce dettagli sulle Libcurl applicazioni Curl e.

Curl

Applicazioni supportate

- Curl

Piattaforme supportate

- Unix: Linux e macOS
 - /usr/local/bin/curl

Caratteristiche principali: Curl

- Esamina i curl file binari per estrarre le informazioni sulla versione incorporata.

Note

In particolare, cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (per i binari ELF su Linux), `.rdata` nella sezione (per i binari PE su Windows) o nella sezione `__cstring` (per i binari MaCHo su macOS).

Stringa della versione Curl

Di seguito è riportato un esempio di stringa di versione incorporata in un Curl file binario:

```
curl/8.14.1
```

8.14.1La versione viene estratta dalla stringa per identificare la Curl versione.

Esempio PURL (Curl)

Di seguito è riportato un esempio di URL del pacchetto per un file di Curl versione.

```
Sample PURL: pkg:generic/curl/curl@8.14.1
```

Libcurl

Applicazioni supportate

- Libcurl

Piattaforme supportate

- Unix: Linux e macOS
 - `///curlver.h` `usr/local bin/curl`

Caratteristiche principali: Libcurl

- Esamina `curlver.h` per estrarre le informazioni sulla versione incorporata per. Libcurl

Note

In particolare, estrae la versione dalle variabili definite `LIBCURL_VERSION_MAJOR`, `LIBCURL_VERSION_MINOR` e `LIBCURL_VERSION_PATCH`.

Stringa della versione di Libcurl

Di seguito è riportato un esempio delle variabili di versione in un `curlver.h` file:

```
#define LIBCURL_VERSION_MAJOR 8
#define LIBCURL_VERSION_MINOR 14
#define LIBCURL_VERSION_PATCH 1
```

8.14.1 La versione viene estratta da queste righe per identificare la Libcurl versione.

Esempio PURL (Libcurl)

Di seguito è riportato un esempio di URL del pacchetto per un Libcurl file di versione.

```
Sample PURL: pkg:generic/curl/libcurl@8.14.1
```

Cursore raccolta ecosistemica

Applicazioni supportate

- Cursor(diAnysphere)

Funzionalità principali

- Esamina le installazioni diCursor, un editor di codice AI Electron basato su. Anysphere Estrae la versione dal manifest `resources/app/package.json` Electron dell'applicazione.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `/usr/share/cursor/`
- `/usr/bin/cursor`
- `~/.local/bin/cursor`

Windows

- `%LOCALAPPDATA%\Programs\cursor\`
- `<SystemDrive>\Program Files\cursor\`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perCursor.

```
Sample PURL: pkg:generic/cursor/cursor@0.42.3?distro=linux
```

Binari Docker raccolta di ecosistemi

Applicazioni supportate

- docker (docker version 20.10.18+)
- dockerd (docker version 20.10.18+)
- containerd (docker version 20.10.18+)
- runc (docker version 29.0.0+)

Note

La valutazione delle vulnerabilità si applica solo alle Docker versioni 25.0.0+. Le versioni precedenti dei Docker file binari possono essere raccolte, ma non sono più supportate da Docker.

Funzionalità principali

- Esamina i file binari Docker statici per estrarre informazioni sulla versione.

Piattaforme supportate

Il generatore SBOM cerca i file binari nei percorsi di installazione consigliati nella documentazione di installazione binaria statica di Docker:

Linux

- `/usr/bin/`

macOS

- `/usr/local/bin/`

Windows

- `C:/Program Files/Docker/`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per Docker.

```
pkg:generic/docker/docker@29.4.0
pkg:generic/docker/dockerd@29.4.0
pkg:generic/docker/containerd@2.2.2
pkg:generic/docker/runc@1.3.5
```

Elasticsearch raccolta di ecosistemi

Applicazioni supportate

- Elasticsearch

Note

La valutazione delle vulnerabilità si applica solo alla Elasticsearch versione 7.17.0.

Funzionalità principali

- **Version**—Decomprime il `elasticsearch-<specific.version>.jar` file per estrarre le macro di installazione all'interno dei `META-INF/MANIFEST.MF` file, che contengono la stringa della versione. Elasticsearch

Piattaforme supportate

- **Linux**—`/etc/elasticsearch/lib`, e `/opt/elasticsearch/lib/ /usr/share/elasticsearch/lib/`
- **macOS**—`/usr/local/var/lib/elasticsearch/lib/`
- **Windows**—`/elasticsearch//Program Files (x86)/Elastic/elasticsearch/lib/`, e `/Program Files/Elastic/elasticsearch/lib/`

Esempio `elasticsearch - versione specifica. < > jar/META-INF/MANIFEST.MF` file

Di seguito è riportato un esempio di `elasticsearch-<specific.version>.jar/META-INF/MANIFEST.MF` file.

```
//truncated

Manifest-Version: 1.0
Module-Origin: git@github.com:elastic/elasticsearch.git
X-Compile-Elasticsearch-Version: 8.19.0-SNAPSHOT
X-Compile-Lucene-Version: 9.12.1
X-Compile-Elasticsearch-Snapshot: true

//truncated
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per un `elasticsearch-<specific.version>.jar/META-INF/MANIFEST.MF` file.

```
pkg:generic/elastic/elasticsearch@8.19.0-SNAPSHOT
```

AI/ML Modelli generici collezione di ecosistemi

Applicazioni supportate

- file di AI/ML modello archiviati localmente su framework e strumenti comuni

Funzionalità principali

- Rileva i file di AI/ML modello in base all'estensione del file: `.gguf`, `.safetensors`, `.onnx.pt`, `.pth` e `.h5`
- Scansiona le directory dei modelli più comuni utilizzate dai AI/ML framework e dagli strumenti di inferenza locali più diffusi.
- I file rilevati vengono generati come componenti con il tipo di componente `machine-learning-model`

Note

Non viene generato alcun PURL per i componenti rilevati da questo raccogliatore.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione dei file modello in percorsi comuni tra piattaforme:

Linux and macOS

- `~/.keras/models`
- `~/.cache/lm-studio/models`
- `~/.local/share/nomic.ai/GPT4All`
- `~/.jan/models`
- `~/.cache/llama.cpp`
- `~/.tabby/models`
- `~/.local/share/localai/models`
- `~/text-generation-webui/models`
- `~/ComfyUI/models`

- `~/stable-diffusion-webui/models`

GitHub Copilota raccolta ecosistemica

Applicazioni supportate

- GitHub Copilot CLI
- GitHub Copilot VS Code extension
- GitHub Copilot JetBrains plugin

Funzionalità principali

- Rileva Copilot su tre superfici di installazione: CLI distribuita da npm (`package.json` risolve dal percorso binario tramite il layout npm), estensione VS Code (`package.json` analizza `github.copilot-<version>` nelle directory; esclude l'estensione `github.copilot-chat-*` separata) e plugin (analizza l'elemento). JetBrains `META-INF/plugin.xml <version>`

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux and macOS

- `/usr/bin/github-copilot-cli`
- `/usr/local/bin/github-copilot-cli`
- `~/.local/bin/github-copilot-cli`
- `~/.vscode/extensions/github.copilot-*/`
- `<IDE>/plugins/github-copilot/META-INF/plugin.xml`

Windows

- `%APPDATA%\npm\github-copilot-cli.cmd`
- `%USERPROFILE%\vscode\extensions\github.copilot-*`
- `%APPDATA%\JetBrains\<IDE>\plugins\github-copilot\`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perGitHub Copilot.

```
Sample PURL: pkg:generic/github/copilot@0.1.36?distro=linux
```

Google raccolta di ecosistemi

Applicazioni supportate

- Google Chrome
- Puppeteer(supporta la libreria puppeteer; puppeteer-core non è incluso)

Note

Puppeteersupporta la libreria puppeteer. Puppeteeril core non è incluso.

Artefatti supportati

Amazon Inspector raccoglie Google Chrome informazioni da quanto segue:

- Il `chrome/VERSION` file (sorgente della build)
- Il `chrome.exe` file (Windows Chromeinstallazione)
- Il `puppeteer` file (installazione)

Per ciascuno degli elementi supportati, Sbomgen analizza e raccoglie il `chrome` file o il `file.puppeteer`. Per le `puppeteer` installazioni, la Chromium versione corrispondente viene raccolta in base alla versione. `puppeteer` Per ulteriori informazioni, consulta [Browser supportati sul sito Web](#) di Puppeteer.

Quando la variabile di `PUPPETEER_SKIP_CHROMIUM_DOWNLOAD` ambiente è impostata su `true`, la valutazione viene ignorata e il `skip_chromium_download=true` qualificatore viene aggiunto all'URL del pacchetto. Puppeteer

Esempio **chrome/VERSION** file di versione

Di seguito è riportato un esempio del file di `chrome/VERSION` versione.

```
MAJOR=130  
MINOR=0  
BUILD=6723  
PATCH=58
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per un file di chrome/VERSION versione.

```
Sample PURL: pkg:generic/google/chrome@131.0.6778.87
```

Esempio **burattinaio** file di versione

Di seguito è riportato un esempio del file di puppeteer versione.

```
{  
  "name": "puppeteer",  
  "version": "23.9.0",  
  "description": "A high-level API to control headless Chrome over the DevTools  
  Protocol",  
  "keywords": [  
    "puppeteer",  
    "chrome",  
    "headless",  
    "automation"  
  ]  
}
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per un file di puppeteer versione.

```
Sample PURL: pkg:generic/google/puppeteer@23.9.0
```

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto con il qualificatore skip per un file di versione.
puppeteer

```
pkg:generic/google/puppeteer@22.15.0?distro=linux&skip_chromium_download=true
```

HuggingFace raccolta ecosistemica

Applicazioni supportate

- HuggingFacehfCLI

Funzionalità principali

- Estrae i modelli memorizzati nella cache AI/ML locale installati da HuggingFace
- Genera gli URL dei HuggingFace pacchetti
- I modelli scaricati utilizzando `hf download --local-dir` sono attualmente supportati

Percorso di esempio

Di seguito è riportato un esempio di percorso del HuggingFace modello memorizzato nella cache.

```
/home/ec2-user/.cache/huggingface/hub/models--MiniMaxAI--MiniMax-M2.5/snapshots/<hash>
```

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un HuggingFace modello. Il tipo di componente è `machine-learning-model`.

```
pkg:huggingface/MiniMaxAI/MiniMax-M2.5@<hash>
```

Java raccolta di ecosistemi

Applicazioni supportate

- Oracle JDK
- Oracle JRE
- Amazon Corretto

Funzionalità principali

- Estrae la stringa dell'Javainstallazione.
- Identifica il percorso della directory che contiene il Java runtime.

- Identifica il fornitore come Oracle JDK, Oracle JRE e Amazon Corretto

Amazon Inspector SBOM Generator esegue la scansione delle Java installazioni attraverso i seguenti percorsi e piattaforme di installazione:

- macOS: `/Library/Java/JavaVirtualMachines`
- Linux 32-bit: `/usr/lib/jvm`
- Linux 64-bit: `/usr/lib64/jvm`
- Linux (generic): `/usr/java` and `/opt/java`

Esempio Java informazioni sulla versione

Di seguito è riportato un esempio di release. Oracle Java

```
// Amazon Corretto
IMPLEMENTOR="Amazon.com Inc."
IMPLEMENTOR_VERSION="Corretto-17.0.11.9.1"
JAVA_RUNTIME_VERSION="17.0.11+9-LTS"
JAVA_VERSION="17.0.11"
JAVA_VERSION_DATE="2024-04-16"
LIBC="default"
MODULES="java.base java.compiler java.datatransfer java.xml java.prefs java.desktop
java.instrument java.logging java.management java.security.sasl java.naming
java.rmi java.management.rmi java.net.http java.scripting java.security.jgss
java.transaction.xa java.sql java.sql.rowset java.xml.crypto java.se java.smartcardio
jdk.accessibility jdk.internal.jvmstat jdk.attach jdk.charsets jdk.compiler
jdk.crypto.ec jdk.crypto.cryptoki jdk.dynalink jdk.internal.ed jdk.editpad
jdk.hotspot.agent jdk.httpserver jdk.incubator.foreign jdk.incubator.vector
jdk.internal.le jdk.internal.opt jdk.internal.vm.ci jdk.internal.vm.compiler
jdk.internal.vm.compiler.management jdk.jartool jdk.javadoc jdk.jcmd jdk.management
jdk.management.agent jdk.jconsole jdk.jdeps jdk.jdwp.agent jdk.jdi jdk.jfr jdk.jlink
jdk.jpackage jdk.jshell jdk.jsobject jdk.jstatd jdk.localedata jdk.management.jfr
jdk.naming.dns jdk.naming.rmi jdk.net jdk.nio.mapmode jdk.random jdk.sctp
jdk.security.auth jdk.security.jgss jdk.unsupported jdk.unsupported.desktop
jdk.xml.dom jdk.zipfs"
OS_ARCH="x86_64"
OS_NAME="Darwin"
SOURCE=".:git:7917f11551e8+"

// JDK
```

```
IMPLEMENTOR="Oracle Corporation"
JAVA_VERSION="19"
JAVA_VERSION_DATE="2022-09-20"
LIBC="default"
MODULES="java.base java.compiler java.datatransfer java.xml java.prefs java.desktop
java.instrument java.logging java.management java.security.sasl java.naming
java.rmi java.management.rmi java.net.http java.scripting java.security.jgss
java.transaction.xa java.sql java.sql.rowset java.xml.crypto java.se java.smartcardio
jdk.accessibility jdk.internal.jvmstat jdk.attach jdk.charsets jdk.zipfs jdk.compiler
jdk.crypto.ec jdk.crypto.cryptoki jdk.dynalink jdk.internal.ed jdk.editpad
jdk.hotspot.agent jdk.httpserver jdk.incubator.concurrent jdk.incubator.vector
jdk.internal.le jdk.internal.opt jdk.internal.vm.ci jdk.internal.vm.compiler
jdk.internal.vm.compiler.management jdk.jartool jdk.javadoc jdk.jcmd jdk.management
jdk.management.agent jdk.jconsole jdk.jdeps jdk.jdwp.agent jdk.jdi jdk.jfr jdk.jlink
jdk.jpackage jdk.jshell jdk.jshell.jobject jdk.jstatd jdk.localedata jdk.management.jfr
jdk.naming.dns jdk.naming.rmi jdk.net jdk.nio.mapmode jdk.random jdk.sctp
jdk.security.auth jdk.security.jgss jdk.unsupported jdk.unsupported.desktop
jdk.xml.dom"
OS_ARCH="x86_64"
OS_NAME="Darwin"
SOURCE=".:git:53b4a11304b0 open:git:967a28c3d85f"
```

Esempio: PURL


Di seguito è riportato un esempio di URL del pacchetto per una Oracle Java versione.

```
Sample PURL:
# Amazon Corretto
pkg:generic/amazon/amazon-corretto@21.0.3
# Oracle JDK
pkg:generic/oracle/jdk@11.0.16
# Oracle JRE
pkg:generic/oracle/jre@20
```

Jenkins raccolta di ecosistemi

Applicazioni supportate

- Jenkins core

 Note

La valutazione delle vulnerabilità si applica alla Jenkins versione 2.400.* e successive.

Funzionalità principali

- Estrae le informazioni sulla versione dal `jenkins.war` file leggendo il `META-INF/MANIFEST.MF` file, che contiene la stringa della versione. Jenkins

Amazon Inspector SBOM Generator cerca le installazioni Jenkins in percorsi di installazione comuni su più piattaforme:

Linux

- `/usr/share/jenkins/jenkins.war`
- `./war usr/share java/jenkins`

macOS

- `/opt/homebrew/opt/jenkins-lts/libexec/jenkins.war`

Windows

- `/Program Files/Jenkins/Jenkins.war`
- `/Program Files (x86)/Jenkins/Jenkins.war`

File di esempio

Di seguito sono riportati alcuni esempi di `jenkins.war/META-INF/MANIFEST.MF` file per diverse versioni.

```
Manifest-Version: 1.0
Created-By: Maven WAR Plugin 3.4.0
Build-Jdk-Spec: 21
Implementation-Title: Jenkins war
Main-Class: executable.Main
Implementation-Version: 2.516.2
```

```
Jenkins-Version: 2.516.2
```

```
Manifest-Version: 1.0
Jenkins-Version: 2.414.1
Implementation-Title: Jenkins
Implementation-Version: 2.414.1
Built-By: kohsuke
Created-By: Apache Maven 3.8.6
```

PURL di esempio

Di seguito sono riportati gli URL dei pacchetti per la versione 2.516.2 della versione Jenkins LTS e la versione 2.414 della versione del server di automazione. Jenkins

```
LTS: pkg:generic/jenkins/jenkins-core-lts@2.516.2.1
Regular: pkg:generic/jenkins/jenkins-core@2.414
```

CLI Kiro collezione di ecosistemi

Applicazioni supportate

- Kiro CLI(Assistente di codifica Amazon AI)

Funzionalità principali

- Esamina il `kiro-cli` file binario (`okiro-cli.appimage`) per estrarre le informazioni sulla versione incorporata.
- Cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (per i binari ELF attivatiLinux) o `__cstring` nella sezione (per i Mach-O binari su macOS). La versione viene estratta dal pattern di stringhe dello user agent di AWS Tooling. `Version<X.Y.Z>/`

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `~/.local/bin/kiro-cli`

- `/usr/bin/kiro-cli`
- `~/Applications/kiro-cli.appimage`

macOS

- `/Applications/Kiro CLI.app/Contents/MacOS/kiro-cli`
- `~/local/bin/kiro-cli`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perKiro CLI.

```
Sample PURL: pkg:generic/amazon/kiro@1.29.5?distro=linux
```

MariaDB and MySQL raccolta di ecosistemi

MariaDB

Applicazioni supportate

- MariaDB Server(10,6+, 11x, 12x)

Funzionalità principali

- Estrae le informazioni sulla versione dai file binari del server di database e dai file di intestazione utilizzando modelli specifici del database.
- Identifica il percorso della directory contenente l'installazione del server di database.
- Distingue automaticamente tra MySQL installazioni MariaDB e installazioni utilizzando il rilevamento del tipo di file basato sui dati.

SBOM Generator cerca MariaDB l'installazione in percorsi di installazione comuni tra piattaforme:

Linux

- `/usr/bin/mariadb`
- `/usr/sbin/mariadb`
- `/usr/local/bin/mariadb`

macOS

- C:/Program Files (x86)/MariaDB/include/mysql/mariadb_version.h (MariaDB)
- C:/Program Files/MariaDB/include/mysql/mariadb_version.h (MariaDB)

Windows

- C:/Program Files (x86)/MariaDB/include/mysql/mariadb_version.h (MariaDB)
- C:/Program Files/MariaDB/include/mysql/mariadb_version.h (MariaDB)

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un MariaDB server.

```
# MariaDB Server  
  
pkg:generic/mysql/mariadb-server@10.11.8
```

MySQL raccolta di ecosistemi

Applicazioni supportate

- Oracle MySQL Server Server(8.0, 8.4, 9.4+)

Funzionalità principali

- Estrae le informazioni sulla versione dai file binari del server di database e dai file di intestazione utilizzando modelli specifici del database.
- Identifica il percorso della directory contenente l'installazione del server di database.
- Distingue automaticamente tra MariaDB installazioni MySQL e installazioni utilizzando il rilevamento del tipo di file basato sui dati.

SBOM Generator cerca MySQL l'installazione in percorsi di installazione comuni tra piattaforme:

Linux

- /usr/local/bin/mysqld

- `/usr/bin/mysqld`
- `/usr/sbin/mysqld`

macOS

- `/usr/local/mysql/include/mysql_version.h` (MySQL)

Windows

- `C:/Program Files/MySQL/MySQL Server/include/mysql_version.h` (MySQL)
- `C:/Program Files (x86)/MySQL/MySQL Server/include/mysql_version.h` (MySQL)

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un MySQL server.

```
# Oracle MySQL Server  
  
pkg:generic/mysql/mysql-server@8.0.43
```

Applicazioni Microsoft raccolta di ecosistemi

Le seguenti applicazioni Microsoft sono inventariate dal generatore SBOM di Amazon Inspector. A causa delle limitazioni dell'API Microsoft CVRF, i rilevamenti nell' InspectorScan API sono supportati solo per le versioni di queste applicazioni rilasciate nel 2021 (o successive). I risultati verranno mappati su KB o CVE di Microsoft (ove applicabile).

Applicazioni Microsoft supportate (2021+)

- PowerShell
- NuGet CLI
- Visual Studio Code
- Microsoft Edge
- SharePoint Server
- Microsoft Defender

- Exchange Server
- Visual Studio
- .NET Core Runtime
- .NET Framework
- ASP.NET Core Runtime
- Microsoft Teams
- Outlook for Windows
- Microsoft Office
- Microsoft 365

Funzionalità principali

- PowerShell— Esamina il `pwsh.exe` file per estrarre le informazioni sulla versione incorporata.
- NuGet CLI— Esamina il `nuget.exe` file per estrarre le informazioni sulla versione incorporata.
- Visual Studio Code— Esamina il `Code.exe` file per estrarre le informazioni sulla versione incorporata.
- Microsoft Edge— Esamina il `msedge.exe` file per estrarre le informazioni sulla versione incorporata.
- SharePoint Server— Esamina il `Microsoft.SharePoint.dll` file per estrarre le informazioni sulla versione incorporata.
- Microsoft Defender— Esamina il `MsMpEng.exe` file per estrarre le informazioni sulla versione incorporata.
- Exchange Server— Esamina il `Exsetup.exe` file per estrarre le informazioni sulla versione incorporata.
- Visual Studio— Analizza il `state.json` file per recuperare la stringa della versione dal campo `catalogInfo.productDisplayVersion`
- .NET Core Runtime— Cerca `Microsoft.NETCore.App.deps.json` il file nei percorsi di installazione ed estrae la stringa della versione dal seguente schema di percorso del file.

```
Microsoft.NETCore.App/<VERSION>/Microsoft.NETCore.App.deps.json
```

- .NET Framework— Analizza il registro di Windows e legge i metadati dei file per rilevare le versioni installate.NET Framework. Lo scanner controlla la chiave e il valore di registro e i file seguenti.

- Registry Key(<VERSION_SUB_KEY> rappresenta la versione di .NET Framework v2.0.50727, ad esempio v3.5, o v4\Full)

```
HKLM\SOFTWARE\Microsoft\.NET Framework Setup\NDP\<VERSION_SUB_KEY>
```

```
HKLM\SOFTWARE\Wow6432Node\Microsoft\.NET Framework Setup\NDP\<VERSION_SUB_KEY>
```

- Registry Value
 - Install— Indica se è installata la versione .NET Framework.
 - Version— Versione .NET Framework installata (versione 4.0 o precedente)
 - Release— Un REG_DWORD valore mappato alla versione .NET Framework installata (versione 4.5 o successiva)
- DLL Files

Lo scanner estrae la versione del file da `mscorlib.dll` e `System.dll`. Se questi file esistono, vengono aggiunti alla SBOM come componenti di file annidati. Per la versione 4.5 o successiva di .NET Framework, la versione del file più grande tra i file viene riportata come versione.

- ASP.NET Core Runtime— Cerca `Microsoft.AspNetCore.App.deps.json` il file nei percorsi di installazione ed estrae la stringa della versione dal seguente schema di percorso del file.

```
Microsoft.AspNetCore.App/<VERSION>/Microsoft.AspNetCore.App.deps.json
```

- Outlook for Windows— Analizza il registro di Windows ed estrae la versione dalla seguente chiave di registro.

```
HKLM\SOFTWARE\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppModel\PackageRepository\Packages\Microsoft.OutlookForWindows_<VERSION>_<ARCH>__8wekyb3d8bbwe
```

- Microsoft Teams— Analizza il registro di Windows ed estrae la versione dalla seguente chiave di registro.

```
HKLM\SOFTWARE\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppModel\PackageRepository\Packages\MSTeams_<VERSION>_<ARCH>__8wekyb3d8bbwe
```

- Microsoft Office 365 / Microsoft 365— Analizza il registro di Windows ed estrae la versione dalla chiave e dal valore di registro seguenti.
 - Chiave di registro

```
HKLM\SOFTWARE\Microsoft\Office\ClickToRun\Configuration
```

- Valore del registro
 - VersionToReport— Versione Microsoft Office
 - ProductReleaselds— Elenco degli ID dei prodotti. Viene utilizzato per identificare i prodotti Office installati. Per ulteriori informazioni sugli ID dei prodotti, consulta [product IDs](#)il Microsoft sito Web.
- Microsoft Office Suite— Raccoglie tutte le applicazioni Office installate esaminando i seguenti file eseguibili:
 - EXCEL . EXE – Microsoft Excel
 - WINWORD . EXE – Microsoft Word
 - POWERPNT . EXE – Microsoft PowerPoint
 - OUTLOOK . EXE – Microsoft Outlook

Il numero di versione nel registro di Windows viene utilizzato come numero di versione autorevole per ogni applicazione di Office installata.

Esempio **state.json** file

Di seguito è riportato un esempio di `state.json` file da utilizzare per raccogliere la versione installata. Visual Studio

```
{
  "icon": {
    "mimeType": "image/svg+xml",
    "fileName": "product.svg"
  },
  "updateDate": "2025-11-06T05:05:35.6517471Z",
  "installDate": "2025-11-06T05:05:35.6527436Z",
  "enginePath": "C:\\Program Files (x86)\\Microsoft Visual Studio\\Installer\\resources\\app\\ServiceHub\\Services\\Microsoft.VisualStudio.Setup.Service",
  "installationName": "VisualStudio/17.14.19+36623.8",
  "catalogInfo": {
    "id": "VisualStudio/17.14.19+36623.8",
    "buildBranch": "d17.14",
    "buildVersion": "17.14.36623.8",
    "localBuild": "build-lab",
    "manifestName": "VisualStudio",
```

```
"manifestType": "installer",  
"productDisplayVersion": "17.14.19",  
// truncated
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per ciascuno Microsoft Applications.

```
// PowerShell  
Sample PURL: pkg:generic/microsoft/powershell@7.5.3  
  
// NuGet CLI  
Sample PURL: pkg:generic/microsoft/nuget@6.14.0  
  
// Visual Studio Code  
Sample PURL: pkg:generic/microsoft/visualstudiocode@1.104.2  
  
// Microsoft Edge  
Sample PURL: pkg:generic/microsoft/edge@140.0.3485.94  
  
// SharePoint Server  
Sample PURL: pkg:generic/microsoft/sharepoint@23.38.219.1  
  
// Microsoft Defender  
Sample PURL: pkg:generic/microsoft/defender@4.18.23110.3  
  
// Exchange Server  
Sample PURL: pkg:generic/microsoft/exchangeserver@15.2.2562.17  
  
// Visual Studio  
Sample PURL: pkg:generic/microsoft/visualstudio@17.14.19  
  
// .NET Core Runtime  
Sample PURL: pkg:generic/microsoft/dotnet@8.0.18  
  
// .NET Framework  
Sample PURL: pkg:generic/microsoft/dotnet-framework-v4.8.1@4.8.9320.0  
  
// ASP.NET Core Runtime  
Sample PURL: pkg:generic/microsoft/aspdotnet@8.0.18  
  
// Microsoft Teams  
Sample PURL: pkg:generic/microsoft/teams@25241.203.3947.4411
```

```
// Outlook for Windows
Sample PURL: pkg:generic/microsoft/outlookforwindows@1.2025.916.400

// Microsoft 365 / Office 365
Sample PURL: pkg:generic/microsoft/office@16.0.19127.20264?
product_ids=0365HomePremRetail

// Microsoft Word
Sample PURL: pkg:generic/microsoft/word@16.0.19127.20264

// Microsoft Excel
Sample PURL: pkg:generic/microsoft/excel@16.0.19127.20264

// Microsoft PowerPoint
Sample PURL: pkg:generic/microsoft/powerpoint@16.0.19127.20264

// Microsoft Outlook
Sample PURL: pkg:generic/microsoft/outlook@16.0.19127.20264
```

Microsoft SQL Server raccolta di ecosistemi

Applicazioni supportate

- Microsoft SQL Server

Funzionalità principali

- Legge dal Windows registro per scoprire le Microsoft SQL Server istanze installate ed estrarre informazioni sulla versione.
- Rileva le istanze attraverso un processo in due fasi: legge il InstalledInstances valore, risolve ogni percorso di istanza dalla sottochiave, quindi legge le informazioni di configurazione dalla Instance Names\SQL sottochiave di ogni istanza. Setup
- Raccoglie il nome dell'istanza, la versione base, il livello di patch, l'edizione, il service pack (se presente) e il percorso della chiave di registro.
- La versione del componente e PURL utilizzano il livello di patch (numero di build completo).

Piattaforme supportate: Windows

Il generatore SBOM di Amazon Inspector legge la seguente chiave di Windows registro per individuare le istanze installate:

```
HKLM\SOFTWARE\Microsoft\Microsoft SQL Server
```

Lo scanner legge il `InstalledInstances` valore per enumerare le istanze, risolve ogni percorso di istanza dalla sottochiave, quindi legge le informazioni di configurazione dalla sottochiave di ogni istanza. `Instance Names\SQL Setup`

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un'Microsoft SQL Server istanza.

```
pkg:generic/microsoft/sqlserver@16.0.1000.6
```

MongoDB raccolta di ecosistemi

Applicazioni supportate

- MongoDB Server(7.0+, 8.0+)

Funzionalità principali

- Esamina i mongod file binari per estrarre le informazioni sulla versione incorporata.

Note

La dimensione del mongod file binario può superare i 200 MB. Per eseguire la scansione MongoDB, il limite di dimensione dei file di Amazon Inspector SBOM Generator deve essere configurato in modo da consentire file di dimensioni superiori a 200 MB.

Amazon Inspector SBOM Generator cerca le MongoDB installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `/usr/bin/mongod`
- `/usr/local/bin/mongod`

macOS

- `/usr/local/bin/mongod`
- `/opt/homebrew/bin/mongod`

Windows

- `C:\Program Files\MongoDB\Server\bin\mongod.exe`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per MongoDB Server.

```
pkg:generic/mongodb/mongodb-server@8.2.4?platform=linux
```

Nginx raccolta di ecosistemi

Applicazioni supportate

- Nginx

Piattaforme supportate

Le seguenti sono le piattaforme supportate.

Linux

- `/usr/sbin/nginx`
- `/usr/local/nginx`
- `/usr/local/etc/nginx`
- `/usr/local/nginx/nginx`
- `/usr/local/nginx/sbin/nginx`
- `/etc/nginx/nginx`

Windows

- `C:\nginx\nginx.exe`

- C:\nginx-x.y.z\nginx.exe (x.y.z è una versione arbitraria)

macOS

- /usr/local/etc/nginx/nginx

Funzionalità principali

Questa raccolta esamina i file binari per estrarre le informazioni sulla versione incorporata. Cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (per i binari ELF attiviLinux), nella `.rdata` sezione (per i binari PE su) o `__cstring` nella sezione (per i binariWindows). MachO

Stringa di versione di esempio

Di seguito è riportato un esempio di stringa di versione incorporata in un Nginx file binario.

```
nginx version: nginx/1.27.5
```

La versione 1.27.5 viene estratta per identificare la Nginx versione.

Esempio: PURL

Quello che segue è un esempio di URL del pacchetto per Nginx.

```
Sample PURL: pkg:generic/nginx/nginx@1.27.5
```

Node.JS raccolta runtime

Applicazioni supportate

- node runtime binario per Node.JS

Piattaforme supportate

Le seguenti sono le piattaforme supportate. (* è una versione arbitraria)

Linux

- /usr/local/bin/node
- //node usr/bin

- `//nodo nodejs/bin`
- `~/ . nvm/versions/nodo/ */ bin/node`
- `~/ . local/share/fnm/node-versions/ */ /node installation/bin`
- `~/ . asdf/installs/nodejs/ */ bin/node`
- `~/ . local/share/mise/installs/nodo/ */ bin/node`
- `~/ .volta/tools/image/node/ */ bin/node`

Windows

- `C:\Program Files\nodejs\node.exe`
- `C:\Program Files (x86)\nodejs\node.exe`
- `~\RoamingAppData\ fnm\node-versions\ *\ installation\node.exe`

macOS

- `/opt/homebrew/Cellar/node/ */ bin/node`

Funzionalità principali

Questa raccolta esamina i file binari per estrarre le informazioni sulle versioni incorporate. Cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (per i binari ELF attiviLinux), nella `.rdata` sezione (per i binari PE su) o `__cstring` nella sezione (per i binariWindows). MachO

Stringa di versione di esempio

Di seguito è riportato un esempio di stringa di versione incorporata in un file binario Node.JS di runtime.

```
node.js/v24.11.1
```

24.11.1La versione viene estratta per identificare la versione Node.JS di runtime.

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perNode.JS.

```
Sample PURL: pkg:generic/nodejs/node@24.11.1
```

Ollama collezione ecosistemica

Applicazioni supportate

- Ollama(runtime LLM locale)

Funzionalità principali

- Esamina il ollama file binario (o attivoWindows) per estrarre le informazioni ollama.exe sulla versione incorporata.
- Ollama è un Go file binario con la versione iniettata in fase di compilazione tramite `-ldflags -X`, che appare nella sezione dei dati del file binario come stringa con prefisso tilde (ad esempio, `~0.21.0`). Nei macOS binari Homebrew `-build`, la versione può in alternativa essere presente nelle informazioni di compilazione del modulo come `Go ollama\tv<version>`

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `/usr/local/bin/ollama`
- `/usr/bin/ollama`
- `~/.local/bin/ollama`

macOS

- `/Applications/Ollama.app/Contents/MacOS/ollama`
- `/usr/local/bin/ollama`
- `/opt/homebrew/bin/ollama`

Windows

- `%LOCALAPPDATA%\Programs\Ollama\ollama.exe`
- `%ProgramFiles%\Ollama\ollama.exe`

Stringa di versione di esempio

Di seguito è riportato un esempio di stringa di versione incorporata in un Ollama file binario.

```
~0.21.0
```

La versione `0.21.0` viene estratta per identificare la Ollama versione.

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per Ollama.

```
Sample PURL: pkg:generic/ollama/ollama@0.21.0?distro=linux
```

Collezionista di modelli Ollama collezione ecosistemica

Applicazioni supportate

- Ollama CLI

Funzionalità principali

- Estrae i AI/ML modelli memorizzati nella cache locale installati da Ollama
- Genera gli URL dei Ollama pacchetti

Percorso di esempio

Di seguito è riportato un esempio di percorso del Ollama modello memorizzato nella cache.

```
/usr/share/ollama/.ollama/models/manifests/registry.ollama.ai/library/gemma4/latest
```

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un Ollama modello. Il tipo di componente è `machine-learning-model`.

```
pkg:ollama/gemma4@<hash>
```

Raccolta di ecosistemi OpenSSH

Applicazioni supportate

- OpenSSH(Versione 9)
- OpenSSH(Versione 10)

Piattaforme supportate Linux/macOS

- `/usr/sbin/sshd`
- `/usr/local/sbin/sshd`

Piattaforme supportate Windows

- `C:/Windows/System32/OpenSSH/sshd.exe`
- `C:/Program Files/OpenSSH/sshd.exe`
- `C:/Program Files (x86)/OpenSSH/sshd.exe`
- `C:/OpenSSH/sshd.exe`

Funzionalità principali

- Esamina i `sshd` file binari per estrarre le informazioni sulla versione incorporata.
- Cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (per i binari ELF attivati)Linux, `__cstring` nella sezione (per i Mach-O binari attivi) o `.rdata` nella sezione (per i binari PE attivatiMacOs). Windows

Stringa di versione di esempio

Di seguito è riportato un esempio di stringa di versione incorporata in un OpenSSH file binario.

```
OpenSSH_9.9p2
```

La versione 9.9p2 viene estratta per identificare la OpenSSH versione.

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perOpenSSH.

```
Sample PURL: pkg:generic/openssh/openssh@9.9p2
```

Collezione di ecosistemi OpenSSL

Applicazioni supportate

Il supporto per le librerie e i pacchetti di sviluppo OpenSSL è limitato al software creato con OpenSSL ufficiale per le versioni 3.0.0 e successive. Il software deve inoltre seguire il controllo delle versioni semantiche. Le varianti e le versioni OpenSSL personalizzate o biforcute precedenti alla 3.0.0 non sono supportate.

Amazon Inspector SBOM Generator estrae le informazioni chiave sui pacchetti per ogni istanza OpenSSL installata.

Funzionalità principali

- Estrae la stringa della versione SEMVER di base dal file di intestazione OpenSSL
- Identifica il percorso della directory contenente l'installazione di OpenSSL

Amazon Inspector SBOM Generator cerca le installazioni OpenSSL eseguendo la scansione del file in percorsi di installazione comuni su più `opensslv.h` piattaforme.

Esempio di percorso di installazione per Linux/Unix

Di seguito è riportato un esempio di percorso di installazione per Linux/Unix.

```
/usr/local/include/openssl/opensslv.h  
/usr/local/ssl/include/openssl/opensslv.h  
/usr/local/openssl/include/openssl/opensslv.h  
/usr/local/opt/openssl/include/openssl/opensslv.h  
/usr/include/openssl/opensslv.h
```

Amazon Inspector SBOM Generator estrae le informazioni sulla versione analizzando il `opensslv.h` file e cercando le definizioni delle versioni.

```
# define OPENSSL_VERSION_MAJOR 3  
# define OPENSSL_VERSION_MINOR 4
```

```
# define OPENSSL_VERSION_PATCH 0
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per la versione OpenSSL.

```
Sample PURL: pkg:generic/openssl/openssl@3.4.0
```

Collezione Oracle Database Server

Applicazioni supportate

- Oracle Database

Piattaforme supportate Linux

- /opt/oracle
- /u01/app/oracle

Note

La valutazione delle vulnerabilità si applica solo alla versione 19 e successive di Oracle Database Server.

Funzionalità principali

- Esamina i Oracle file binari per estrarre le informazioni sulla versione incorporata.
- Cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (attiva i binari ELF). Linux
- Le informazioni sulla versione seguono un formato specifico che include la stringa della versione RDBMS.

Stringa di versione di esempio

Di seguito è riportato un esempio di stringa di versione incorporata in un Oracle Database file binario:

```
RDBMS_23.7.0.25.01DBRU_LINUX.X64_240304
```

La versione 23.7.0.25.01 viene estratta per identificare la Oracle Database versione.

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per Oracle Database.

```
Sample PURL: pkg:generic/oracle/database@23.7.0.25.01
```

PHP raccolta di ecosistemi

Applicazioni supportate

- PHP(versione 8.1 e successive)

Funzionalità principali

- Estrae le informazioni sulla versione dagli eseguibili PHP binari utilizzando stringhe di versione incorporate.
- Identifica il percorso della directory contenente il file binario. PHP
- Rileva automaticamente sia i PHP file binari standard che le installazioni con versione, ad esempio, e. php8.1 php8.2 php8.3

Amazon Inspector SBOM Generator cerca le PHP installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `usr/bin/php8.1` through `usr/bin/php8.9`
- `usr/sbin/php8.1` through `usr/sbin/php8.9`
- `usr/local/bin/php`, `usr/bin/php`, `usr/sbin/php`
- `usr/local/bin/php8.1` through `usr/local/bin/php8.9`(file binari con versione)

macOS

- `opt/homebrew/bin/php`
- `usr/bin/php`

- `/usr/local/bin/php`

Windows

- `C:/php/php.exe`
- `C:/php8.1/php.exe` through `C:/php8.9/php.exe`(directory con versione)

Esempio PHP estrazione della versione

Amazon Inspector SBOM Generator estrae le informazioni sulla versione dai PHP file binari cercando stringhe di versione incorporate utilizzando lo schema seguente.

```
X-Powered-By: PHP/8.4.12
```

8.4.12 viene estratto da questo schema per identificare la versione. PHP

Esempio: PURL

Di seguito è riportato un esempio di URL di pacchetto per un PHP pattern.

```
pkg:generic/php/php@8.4.12
```

Redis raccolta di ecosistemi

Applicazioni supportate

- Redis(versione 7.2 e successive)

Funzionalità principali

- Estrae le informazioni sulla versione dagli eseguibili Redis `redis-server` binari utilizzando stringhe di versione incorporate.
- Cerca le stringhe di versione nella `.rodata` sezione eseguibile binaria (per i binari ELF attiviLinux) o `__cstring` nella sezione (per i binari su). Mach-O macOS

Amazon Inspector SBOM Generator cerca le Redis installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `/usr/bin/redis-server`
- `/usr/local/bin/redis-server`

macOS

- `/opt/homebrew/bin/redis-server`
- `/usr/local/bin/redis-server`

Stringa di versione di esempio

Di seguito è riportato un esempio di stringa di versione incorporata in un Redis file binario.

```
redis-7.2.6
```

La versione 7.2.6 viene estratta per identificare la Redis versione.

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perRedis.

```
pkg:generic/redis/redis@7.2.6
```

Windsurf collezione ecosistemica

Applicazioni supportate

- Windsurf(diCodeium)

Funzionalità principali

- Esamina le installazioni diWindsurf, un editor di codice AI Electron basato su. Codeium Estrae la versione dal manifest `resources/app/package.json` Electron dell'applicazione.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- `/usr/share/windsurf/`
- `/usr/bin/windsurf`

Windows

- `%LOCALAPPDATA%\Programs\windsurf\`
- `<SystemDrive>\Program Files\Windsurf\`

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perWindsurf.

```
Sample PURL: pkg:generic/codeium/windsurf@1.10.5?distro=linux
```

WordPress raccolta di ecosistemi

Componenti supportati

- WordPress core
- WordPressplugin
- WordPresstemi

Funzionalità principali

- WordPresscore — analizza il `/wp-includes/version.php` file per estrarre il valore della versione dalla variabile `$wp_version`.
- WordPressplugins: analizza il `/wp-content/plugins/<WordPress Plugin>/readme.txt` file o il `/wp-content/plugins/<WordPress Plugin>/readme.md` file per estrarre il tag come stringa della versione. Stable
- WordPressthemes: analizza il `/wp-content/themes/<WordPress Theme>/style.css` file per estrarre la versione dai metadati della versione.

Esempio **version.php** file

Di seguito è riportato un esempio di `version.php` file WordPress principale.

```
// truncated

/**
 * The WordPress version string.
 *
 * Holds the current version number for WordPress core. Used to bust caches
 * and to enable development mode for scripts when running from the /src directory.
 *
 * @global string $wp_version
 */
$wp_version = '6.5.5';

// truncated
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per WordPress core.

```
Sample PURL: pkg:generic/wordpress/core/wordpress@6.5.5
```

Esempio **readme.txt** file

Di seguito è riportato un esempio di `readme.txt` file di WordPress plugin.

```
=== Plugin Name ===
Contributors: (this should be a list of wordpress.org userid's)
Donate link: https://example.com/
Tags: tag1, tag2
Requires at least: 4.7
Tested up to: 5.4
Stable tag: 4.3
Requires PHP: 7.0
License: GPLv2 or later
License URI: https://www.gnu.org/licenses/gpl-2.0.html

// truncated
```

Esempio: PURL

Quello che segue è un esempio di URL di pacchetto per un WordPress plugin.

```
Sample PURL: pkg:generic/wordpress/plugin/exclusive-addons-for-elementor@1.0.0
```

Esempio `style.css` file

Di seguito è riportato un esempio di `style.css` file WordPress tematico.

```
/*
Author: the WordPress team
Author URI: https://wordpress.org
Description: Twenty Twenty-Four is designed to be flexible, versatile and applicable
to any website. Its collection of templates and patterns tailor to different needs,
such as presenting a business, blogging and writing or showcasing work. A multitude
of possibilities open up with just a few adjustments to color and typography. Twenty
Twenty-Four comes with style variations and full page designs to help speed up the
site building process, is fully compatible with the site editor, and takes advantage
of new design tools introduced in WordPress 6.4.
Requires at least: 6.4
Tested up to: 6.5
Requires PHP: 7.0
Version: 1.2
License: GNU General Public License v2 or later
License URI: http://www.gnu.org/licenses/gpl-2.0.html
Text Domain: twentytwentyfour
Tags: one-column, custom-colors, custom-menu, custom-logo, editor-style, featured-
images, full-site-editing, block-patterns, rtl-language-support, sticky-post,
threaded-comments, translation-ready, wide-blocks, block-styles, style-variations,
accessibility-ready, blog, portfolio, news
*/
```

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto per un WordPress tema.

```
Sample PURL: pkg:generic/wordpress/theme/avada@1.0.0
```

Zed collezione ecosistemica

Applicazioni supportate

- Zed(diZed Industries)

Funzionalità principali

- Esamina il Zed file binario autonomo distribuito da. Zed Industries
- Zedè un Rust file binario senza manifesto. La versione viene estratta da una stringa costante incorporata nel file binario.

Piattaforme supportate

Amazon Inspector SBOM Generator esegue la scansione delle installazioni in percorsi di installazione comuni su più piattaforme:

Linux

- /usr/bin/zed
- /usr/local/bin/zed
- ~/.local/bin/zed

Windows

- %LOCALAPPDATA%\Zed\
- <SystemDrive>\Program Files\Zed\

Esempio: PURL

Di seguito è riportato un esempio di URL del pacchetto perZed.

```
Sample PURL: pkg:generic/zed-industries/zed@1.2.6?distro=linux
```

Scansioni dei certificati di Amazon Inspector SBOM Generator SSL/TLS

Questa sezione descrive come utilizzare Amazon Inspector SBOM Generator per inventariare i certificati. SSL/TLS L'Sbomgeninventario SSL/TLS dei certificati cercando i certificati in posizioni predefinite e nelle directory fornite dall'utente. La funzionalità ha lo scopo di consentire agli utenti di effettuare l'inventario SSL/TLS dei certificati e di identificare i certificati scaduti. I certificati CA verranno visualizzati anche nell'inventario di output.

Utilizzo delle scansioni dei Sbmngen certificati

È possibile abilitare la raccolta dell'inventario dei SSL/TLS certificati utilizzando l'`--scanners certificates` argomento. Le scansioni dei certificati possono essere combinate con qualsiasi altro scanner. Per impostazione predefinita, le scansioni dei certificati non sono abilitate.

SbmngenCerca i certificati in diverse posizioni a seconda dell'elemento da scansionare. In tutti i casi, i Sbmngen tentativi di estrarre i certificati in file con le seguenti estensioni.

```
.pem  
.crt  
.der  
.p7b  
.p7m  
.p7s  
.p12  
.pfx
```

Il tipo di artefatto localhost

Se lo scanner di certificati è abilitato e il tipo di artefatto è localhost, cerca Sbmngen ricorsivamente i certificati in, e `/etc/*/ssl/opt/*/ssl/certs`, `/usr/local/*/ssl` where non è vuoto. `/var/lib/*/certs` * Le directory fornite dall'utente verranno ricercate in modo ricorsivo, indipendentemente dal nome delle directory. In genere, CA/system i certificati non vengono inseriti in questi percorsi. Questi certificati si trovano spesso in cartelle denominate `pkica-certs`, `oCA`. Possono inoltre apparire nei percorsi di scansione predefiniti di localhost.

Elementi della directory e del contenitore

Durante la scansione degli elementi della directory o del contenitore, Sbmngen cerca i certificati posizionati in qualsiasi punto dell'elemento.

Esempi di comandi di scansione dei certificati

Di seguito sono riportati esempi di comandi di scansione dei certificati. Uno genera un SBOM che contiene solo certificati in una directory locale. Un altro genera un SBOM che contiene certificati e Alpine RHEL pacchetti in una directory locale. Debian Un altro genera un SBOM che contiene i certificati che si trovano in posizioni di certificati comuni.

```
# generate SBOM only containing certificates in a local directory
./inspector-sbomgen directory --path ./project/ --scanners certificates

# generate SBOM only containing certificates and Alpine, Debian, and RHEL OS packages
in a local directory
./inspector-sbomgen directory --path ./project/ --scanners certificates,dpkg,alpine-
apk,rhel-rpm

# generate SBOM only containing certificates, taken from common localhost certificate
locations
./inspector-sbomgen localhost --scanners certificates
```

Componente di file di esempio

Di seguito sono riportati due esempi di componenti per la ricerca di certificati. Quando un certificato scade, è possibile visualizzare una proprietà aggiuntiva che identifica la data di scadenza.

```
{
  "bom-ref": "comp-2",
  "type": "file",
  "name": "certificate:expired.pem",
  "properties": [
    {
      "name": "amazon:inspector:sbom_generator:certificate_finding:IN-
CERTIFICATE-001",
      "value": "expired:2015-06-06T11:59:59Z"
    },
    {
      "name": "amazon:inspector:sbom_generator:source_path",
      "value": "/etc/ssl/expired.pem"
    }
  ]
},
{
  "bom-ref": "comp-3",
  "type": "file",
```

```

    "name": "certificate:unexpired.pem",
    "properties": [
      {
        "name": "amazon:inspector:sbom_generator:source_path",
        "value": "/etc/ssl/unexpired.pem"
      }
    ]
  }
}

```

Esempio di componente di risposta alla vulnerabilità

L'esecuzione di Amazon Inspector SBOM Generator con il `--scan-sbom` flag invia la SBOM risultante ad Amazon Inspector per la scansione delle vulnerabilità. Di seguito è riportato un esempio di ricerca di certificati per un componente di risposta alle vulnerabilità.

```

{
  "advisories": [
    {
      "url": "https://aws.amazon.com/inspector/"
    },
    {
      "url": "https://docs.aws.amazon.com/wellarchitected/latest/security-pillar/sec_protect_data_transit_encrypt.html"
    }
  ],
  "affects": [
    {
      "ref": "comp-2"
    }
  ],
  "analysis": {
    "state": "in_triage"
  },
  "bom-ref": "vuln-1",
  "created": "2025-04-17T18:48:20Z",
  "cwes": [
    324,
    298
  ],
  "description": "Expired Certificate: The associated certificate(s) are no longer valid. Replace certificate in order to reduce risk.",
  "id": "IN-CERTIFICATE-001",
  "properties": [

```

```
{
  "name": "amazon:inspector:sbom_scanner:priority",
  "value": "standard"
},
{
  "name": "amazon:inspector:sbom_scanner:priority_intelligence",
  "value": "unverified"
}
],
"published": "2025-04-17T18:48:20Z",
"ratings": [
  {
    "method": "other",
    "severity": "medium",
    "source": {
      "name": "AMAZON_INSPECTOR",
      "url": "https://aws.amazon.com/inspector/"
    }
  }
],
"source": {
  "name": "AMAZON_INSPECTOR",
  "url": "https://aws.amazon.com/inspector/"
},
"updated": "2025-04-17T18:48:20Z"
}
```

Raccolta di licenze Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator aiuta a tenere traccia delle informazioni sulle licenze in una distinta base del software (SBOM). Raccoglie informazioni sulle licenze dai pacchetti supportati nei sistemi operativi e nei linguaggi di programmazione. Con espressioni di licenza standardizzate nella SBOM generata, potete comprendere i vostri obblighi di licenza.

Raccogli informazioni sulla licenza

Esempio di comando

L'esempio seguente mostra come raccogliere informazioni sulla licenza da una directory.

```
./inspector-sbomgen directory --path /path/to/your/directory/ --collect-licenses
```

Esempio di componente SBOM

L'esempio seguente mostra una voce di componente nella SBOM generata.

```
"components": [
  {
    "bom-ref": "comp-2",
    "type": "application",
    "name": "sample-js-pkg",
    "version": "1.2.3",
    "licenses": [
      {
        "expression": "Apache-2.0 AND (MIT OR GPL-2.0-only)"
      }
    ],
    "purl": "pkg:npm/sample-js-pkg@1.2.3",
  }
]
```

Pacchetti supportati

I seguenti linguaggi di programmazione e pacchetti del sistema operativo sono supportati per la raccolta delle licenze.

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
Alma Linux	RPM	<ul style="list-style-type: none"> /usr/lib/sysimage/rpm/rpmdb.sqlite /usr/lib/sysimage/rpm/Packages /.db usr/lib/sysimage/rpm/Packages /.sqlite var/lib/rpm/rpmdb 	Sistema operativo

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
		<ul style="list-style-type: none"> • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages 	
Amazon Linux	RPM	<ul style="list-style-type: none"> • /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages 	Sistema operativo
CentOS	RPM	<ul style="list-style-type: none"> • /.sqlite usr/lib/sysimage/rpm/rpmdb • /usr/lib/sysimage/rpm/Packages • /.db usr/lib/sysimage/rpm/Packages • /.sqlite var/lib/rpm/rpmdb • /var/lib/rpm/Packages • /.db var/lib/rpm/Packages 	Sistema operativo

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
Fedora	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo
OpenSUSE	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
Oracle Linux	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo
Photon OS	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
RHEL	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo
Rocky Linux	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
SLES	RPM	<ul style="list-style-type: none"> • <code>/.sqlite usr/lib/sysimage/rpm/rpmdb</code> • <code>/usr/lib/sysimage/rpm/Packages</code> • <code>/.db usr/lib/sysimage/rpm/Packages</code> • <code>/.sqlite var/lib/rpm/rpmdb</code> • <code>/var/lib/rpm/Packages</code> • <code>/.db var/lib/rpm/Packages</code> 	Sistema operativo
Alpine Linux	APK	<code>/lib/apk/db/installed</code>	Sistema operativo
Chainguard	APK	<code>/lib/apk/db/installed</code>	Sistema operativo
Debian	DPKG	<code>/usr/share/doc/*/copyright</code>	Sistema operativo
Ubuntu	DPKG	<code>/usr/share/doc/*/copyright</code>	Sistema operativo
Node.js	Javascript	<code>node_modules/*/package.json</code>	Linguaggio di programmazione
PHP	Pacchetto Composer	<ul style="list-style-type: none"> • <code>composer.lock</code> • <code>/vendor/composer/installed.json</code> 	Linguaggio di programmazione

Target	Programma di gestione dei pacchetti	Fonte di informazioni sulla licenza	Tipo
Go	Go	LICENSE	Linguaggio di programmazione
Python	Python/Egg/Wheel	<ul style="list-style-type: none"> • <code>.dist-info/METADATA</code> • <code>.egg-info</code> • <code>.egg-info/PKG-INFO</code> 	Linguaggio di programmazione
Ruby	RubyGem	* <code>.gemspec</code>	Linguaggio di programmazione
Rust	crate	<code>Cargo.toml</code>	Linguaggio di programmazione

Standardizzazione delle espressioni di licenza

Il formato delle espressioni di licenza SPDX fornisce una rappresentazione accurata dei termini di licenza presenti nel software open source. Amazon Inspector SBOM Generator standardizza tutte le informazioni sulla licenza in espressioni di licenza SPDX attraverso le regole descritte in questa sezione. Le regole garantiscono coerenza e compatibilità tra le informazioni sulle licenze.

Mappatura degli identificatori in formato abbreviato SPDX

Tutti i nomi di licenza sono mappati su identificatori in formato breve SPDX. Ad esempio, MIT License è abbreviata in MIT.

Combinazione multipla di licenze

È possibile combinare più di una licenza con l'AND operatore. Di seguito è riportato un comando di esempio che mostra come formattare il comando.

```
MIT AND Apache-2.0
```

Prefisso di licenza personalizzato

Le licenze personalizzate hanno il prefisso `LicenseRef`, ad esempio. `LicenseRef-CompanyPrivate`

Prefisso di eccezione personalizzato

Le eccezioni personalizzate hanno il prefisso `AdditionRef-`, ad esempio. `AdditionRef-CustomException`

Cos'è l'URL di un pacchetto?

[L'URL di un pacchetto o PURL](#) è un formato standardizzato utilizzato per identificare pacchetti software, componenti e librerie in diversi sistemi di gestione dei pacchetti. Il formato semplifica il monitoraggio, l'analisi e la gestione delle dipendenze nei progetti software, in particolare durante la generazione di una distinta base del software (SBOMs).

Struttura PURL

La struttura PURL è simile a un URL ed è composta da più componenti:

- `pkg`— Il prefisso letterale
- `type`— Il tipo di pacchetto
- `namespace`— Il raggruppamento
- `name`— Il nome del pacchetto
- `version`— La versione del pacchetto
- `qualifiers`— Coppie chiave-valore aggiuntive
- `subpath`— Il percorso del file nel pacchetto

Esempio: PURL

Di seguito è riportato un esempio di come potrebbe apparire un PURL.

```
pkg:<type>/<namespace>/<name>@<version>?<qualifiers>#<subpath>
```

Il PURL generico

Un PURL generico viene utilizzato per rappresentare pacchetti e componenti software che non rientrano negli ecosistemi di pacchetti consolidati, ad esempio npm, o pypi maven. Identifica i componenti software e acquisisce metadati che potrebbero non essere in linea con sistemi di gestione dei pacchetti specifici. Un PURL generico è utile per una varietà di progetti software, dai file binari compilati alle piattaforme, come e. Apache WordPress. Ne consente l'applicazione in un'ampia gamma di casi d'uso, inclusi file binari compilati, piattaforme web e distribuzioni software personalizzate.

Casi d'uso principali

- Supporta i binari compilati ed è utile per e Go Rust
- Supporta piattaforme web, come Apache and WordPress, in cui un pacchetto potrebbe non essere associato ai gestori di pacchetti tradizionali.
- Supporta software legacy personalizzato consentendo alle organizzazioni di fare riferimento a software o sistemi sviluppati internamente privi di pacchetti formali.

Formato di esempio

Di seguito è riportato un esempio del formato PURL generico.

```
pkg:generic/<namespace>/<name>@<version>?<qualifiers>
```

Esempi aggiuntivi del formato PURL generico

Di seguito sono riportati altri esempi del formato PURL generico.

Binario compilato Go

Quanto segue rappresenta il `inspector-sbomgen` binario compilato con aGo.

```
pkg:generic/inspector-sbomgen?go_toolchain=1.22.5
```

RustBinario compilato

Quanto segue rappresenta il `myrustapp` file binario compilato con Rust.

```
pkg:generic/myrustapp?rust_toolchain=1.71.0
```

Progetto Apache

Quanto segue si riferisce a un progetto http nel Apache namespace.

```
pkg:generic/apache/httpd@1.0.0
```

Software WordPress

Quanto segue si riferisce a un software di baseWordPress.

```
pkg:generic/wordpress/core/wordpress@6.0.0
```

WordPresstema

Quanto segue si riferisce a un WordPress tema personalizzato.

```
pkg:generic/wordpress/theme/mytheme@1.0.0
```

WordPress Plugin

Quanto segue si riferisce a un WordPress plugin personalizzato.

```
pkg:generic/wordpress/plugin/myplugin@1.0.0
```

Gestione di riferimenti di versione non risolti o non standard in Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator individua e analizza gli artefatti supportati all'interno di un sistema identificando le dipendenze direttamente dai file di origine. Non è un gestore di pacchetti e non risolve intervalli di versioni, deduce versioni basate su riferimenti dinamici o gestisce le ricerche nel registro. Raccoglie le dipendenze solo così come sono definite negli artefatti sorgenti del progetto. In molti casi, le dipendenze nei manifesti dei pacchetti, ad esempio, o, vengono specificate utilizzando `package.json` versioni non `pom.xml` risolte o `requirements.txt` basate su intervalli. Questo argomento include esempi di come potrebbero apparire queste dipendenze.

Raccomandazioni

Amazon Inspector SBOM Generator estrae le dipendenze dagli artefatti di origine, ma non risolve o interpreta intervalli di versioni o riferimenti dinamici. Per una scansione più accurata delle vulnerabilità

SBOMs, consigliamo di utilizzare identificatori di versione semantici risolti nelle dipendenze del progetto.

Java

InfattiJava, i Maven progetti possono utilizzare intervalli di versioni per definire le dipendenze nel file `pom.xml`

```
<dependency>
  <groupId>org.inspector</groupId>
  <artifactId>inspector-api</artifactId>
  <version>(,1.0]</version>
</dependency>
```

L'intervallo specifica che qualsiasi versione fino alla 1.0 inclusa è accettabile. Tuttavia, se una versione non è una versione risolta, Amazon Inspector SBOM Generator non la raccoglierà perché non può essere mappata a una versione specifica.

JavaScript

Il package .json file può JavaScript infatti includere intervalli di versioni simili ai seguenti:

```
"dependencies": {
  "ky": "^1.2.0",
  "registry-auth-token": "^5.0.2",
  "registry-url": "^6.0.1",
  "semver": "^7.6.0"
}
```

L'^operatore specifica che qualsiasi versione superiore o uguale alla versione specificata è accettabile. Tuttavia, se la versione specificata non è una versione risolta, Amazon Inspector SBOM Generator non la raccoglierà perché così facendo si possono generare falsi positivi durante il rilevamento delle vulnerabilità.

Python

InfattiPython, il `requirements.txt` file può includere voci con un'espressione booleana.

```
requests>=1.0.0
```

L'`>=` operatore specifica che qualsiasi versione maggiore o uguale a `1.0.0` è accettabile.

Poiché questa particolare espressione non specifica una versione esatta, Amazon Inspector SBOM Generator non è in grado di raccogliere in modo affidabile una versione per l'analisi delle vulnerabilità.

Amazon Inspector SBOM Generator non supporta identificatori di versione non standard o ambigui, come `beta`, `latest` o `snapshot`.

```
pkg:maven/org.example.com/testmaven@1.0.2%20Beta-RC-1_Release
```

Note

L'uso di un suffisso non standard, ad esempio, non è conforme al controllo delle versioni semantiche standard e non può essere valutato per individuare le vulnerabilità all'interno del motore di rilevamento Amazon Inspector. `Beta-RC-1_Release`

Utilizzo dei CycloneDX namespace con Amazon Inspector

Amazon Inspector ti fornisce CycloneDX namespace e nomi di proprietà con cui puoi utilizzare.

SBOMs Questa sezione descrive tutte le key/value proprietà personalizzate che possono essere aggiunte ai componenti in. CycloneDX SBOMs Per ulteriori informazioni, consulta la [tassonomia delle proprietà CyclonedX](#) sul sito Web. GitHub

amazon:inspector:sbom_scannertassonomia dei namespace

L'API Amazon Inspector Scan utilizza lo spazio dei `amazon:inspector:sbom_scanner` nomi e ha le seguenti proprietà:

Proprietà	Descrizione
<code>amazon:inspector:sbom_scanner:cisa_key_date_added</code>	Indica quando la vulnerabilità è stata aggiunta al catalogo CISA Known Exploited Vulnerabilities.

Proprietà	Descrizione
<code>amazon:inspector:sbom_scanner:cisa_key_date_due</code>	Indica quando è necessaria la correzione della vulnerabilità in base al catalogo CISA Known Exploited Vulnerabilities.
<code>amazon:inspector:sbom_scanner:critical_vulnerabilities</code>	Conteggio del numero totale di vulnerabilità di gravità critica rilevate nella SBOM.
<code>amazon:inspector:sbom_scanner:exploit_available</code>	Indica se è disponibile un exploit per la vulnerabilità specificata.
<code>amazon:inspector:sbom_scanner:exploit_last_seen_in_public</code>	Indica quando un exploit è stato visto l'ultima volta in pubblico per una determinata vulnerabilità.
<code>amazon:inspector:sbom_scanner:fixed_version: <i>component_bom_ref</i></code>	Fornisce la versione fissa del componente indicato per la vulnerabilità specificata.
<code>amazon:inspector:sbom_scanner:high_vulnerabilities</code>	Conteggio del numero totale di vulnerabilità ad alta gravità rilevate nella SBOM.
<code>amazon:inspector:sbom_scanner:info</code>	Fornisce il contesto di scansione per un determinato componente, ad esempio: «Componente scansionato: nessuna vulnerabilità trovata».
<code>amazon:inspector:sbom_scanner:is_malicious</code>	Indica se OpenSSF identifica i componenti interessati come dannosi.
<code>amazon:inspector:sbom_scanner:low_vulnerabilities</code>	Conteggio del numero totale di vulnerabilità di bassa gravità rilevate nello SBOM.
<code>amazon:inspector:sbom_scanner:medium_vulnerabilities</code>	Conteggio del numero totale di vulnerabilità di media gravità rilevate nello SBOM.
<code>amazon:inspector:sbom_scanner:path</code>	Il percorso del file che fornisce le informazioni sull'oggetto del pacchetto.

Proprietà	Descrizione
<code>amazon:inspector:sbom_scanner:priority</code>	La priorità consigliata per correggere una determinata vulnerabilità. I valori in ordine decrescente sono «IMMEDIATE», «URGENT», «MODERATE» e «STANDARD».
<code>amazon:inspector:sbom_scanner:priority_intelligence</code>	La qualità dell'intelligence utilizzata per determinare la priorità di una determinata vulnerabilità. I valori includono «VERIFIED» o «UNVERIFIED».
<code>amazon:inspector:sbom_scanner:warning</code>	Fornisce un contesto sul motivo per cui un determinato componente non è stato analizzato, ad esempio: «Componente ignorato: nessun purl fornito».

amazon:inspector:sbom_generator tassonomia dello spazio dei nomi

Il generatore SBOM di Amazon Inspector utilizza lo spazio dei nomi `amazon:inspector:sbom_generator` e presenta le seguenti proprietà:

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:cpu_architecture</code>	L'architettura della CPU del sistema da inventariare (x86_64).
<code>amazon:inspector:sbom_generator:ec2:instance_id</code>	L'ID dell'istanza Amazon EC2.
<code>amazon:inspector:sbom_generator:ec2:instance_type</code>	Il tipo di istanza Amazon EC2
<code>amazon:inspector:sbom_generator:live_patching_enabled</code>	Un valore booleano che indica se il live patching è abilitato su Amazon EC2 Amazon Linux

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:live_patched_cves</code>	Un elenco di CVE patchati tramite live patching su Amazon EC2 Amazon. Linux
<code>amazon:inspector:sbom_generator:dockerfile_finding: <i>inspector_finding_id</i></code>	Indica che una ricerca di Amazon Inspector in un componente è correlata ai Dockerfile controlli.
<code>amazon:inspector:sbom_generator:image_id</code>	L'hash appartenente al file di configurazione dell'immagine del contenitore (noto anche come ID immagine).
<code>amazon:inspector:sbom_generator:image_arch</code>	L'architettura dell'immagine del contenitore.
<code>amazon:inspector:sbom_generator:image_author</code>	L'autore dell'immagine del contenitore.
<code>amazon:inspector:sbom_generator:image:cmd: <i>count</i></code>	Una directory assoluta all'interno dell'immagine del contenitore definita nel CMD predefinito configurato al momento della creazione dell'immagine.
<code>amazon:inspector:sbom_generator:image:entrypoint: <i>count</i></code>	Una directory assoluta all'interno dell'immagine del contenitore definita in ENTRYPOINT predefinito configurata al momento della creazione dell'immagine.
<code>amazon:inspector:sbom_generator:image:workdir</code>	La directory WORKDIR dell'immagine del contenitore configurata al momento della creazione dell'immagine.
<code>amazon:inspector:sbom_generator:image_docker_version</code>	La versione docker utilizzata per creare l'immagine del contenitore.
<code>amazon:inspector:sbom_generator:is_duplicate_package</code>	Indica che il pacchetto oggetto è stato trovato da più di uno scanner di file.

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:duplicate_purl</code>	Indica il pacchetto duplicato PURL trovato da un altro scanner.
<code>amazon:inspector:sbom_generator:kernel_name</code>	Il nome del kernel del sistema da inventariare.
<code>amazon:inspector:sbom_generator:kernel_version</code>	La versione del kernel del sistema da inventariare.
<code>amazon:inspector:sbom_generator:kernel_component</code>	Un valore booleano che indica se un pacchetto oggetto è un componente del kernel
<code>amazon:inspector:sbom_generator:running_kernel</code>	Un valore booleano che indica se un pacchetto oggetto è il kernel in esecuzione
<code>amazon:inspector:sbom_generator:layer_diff_id</code>	L'hash del livello di immagine del contenitore non compresso.
<code>amazon:inspector:sbom_generator:replaced_by</code>	Il valore che sostituisce il modulo corrente. Go
<code>amazon:inspector:sbom_generator:os_hostname</code>	Il nome host del sistema da inventariare.
<code>amazon:inspector:sbom_generator:source_file_scanner</code>	Lo scanner che ha trovato il file che contiene le informazioni sul pacchetto, ad esempio: <code>/var/lib/dpkg/status</code>
<code>amazon:inspector:sbom_generator:source_package_collector</code>	Il raccogliitore che ha estratto il nome e la versione del pacchetto da un file specifico.
<code>amazon:inspector:sbom_generator:source_path</code>	Il percorso del file da cui sono state estratte le informazioni del pacchetto oggetto.
<code>amazon:inspector:sbom_generator:file_size_bytes</code>	Indica la dimensione del file di un determinato elemento.

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:unresolved_version</code>	Indica una stringa di versione che non è stata risolta dal gestore di pacchetti.
<code>amazon:inspector:sbom_generator:experimental:transitive_dependency</code>	Indica le dipendenze indirette da un gestore di pacchetti.
<code>amazon:inspector:sbom_generator:subscription:enabled</code>	Un valore booleano che indica se un abbonamento è abilitato, ad esempio RHEL EUS/E4S o Pro. Ubuntu
<code>amazon:inspector:sbom_generator:subscription:name</code>	Il nome dell'abbonamento attivo (ad esempio, EUS, E4S, Pro).
<code>amazon:inspector:sbom_generator:subscription:locked_version</code>	La versione bloccata dall'abbonamento RHEL attivo (solo RHEL EUS/E4S).
<code>amazon:inspector:sbom_generator:metadata:host:hostname</code>	Il nome host del sistema scansionato.
<code>amazon:inspector:sbom_generator:metadata:host:kernel_name</code>	Il nome del kernel del sistema operativo (ad esempio Linux, Darwin, Windows_NT).
<code>amazon:inspector:sbom_generator:metadata:host:kernel_version</code>	La stringa della versione del kernel del sistema operativo.
<code>amazon:inspector:sbom_generator:metadata:host:cpu_architecture</code>	L'architettura della CPU del sistema (ad esempio, x86_64, arm64).
<code>amazon:inspector:sbom_generator:metadata:host:bootdisk_id</code>	Identificatore univoco del disco di avvio.
<code>amazon:inspector:sbom_generator:metadata:host:boot_id</code>	Identificatore univoco per la sessione di avvio corrente.

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:metadata:host:boot_time</code>	Tempo di avvio del sistema in formato ISO 8601.
<code>amazon:inspector:sbom_generator:metadata:host:system_id</code>	Identificatore di sistema persistente (machine-id su Linux, MachineGuid su Windows).
<code>amazon:inspector:sbom_generator:metadata:host:system_serial</code>	Numero di serie dell'hardware riportato nel firmware del sistema.
<code>amazon:inspector:sbom_generator:metadata:host:network_interfaces: <i>name</i>:hardware</code>	Indirizzo MAC dell'interfaccia di rete.
<code>amazon:inspector:sbom_generator:metadata:host:network_interfaces: <i>name</i>:ipv4</code>	IPv4 indirizzo/i assegnato/i all'interfaccia.
<code>amazon:inspector:sbom_generator:metadata:host:network_interfaces: <i>name</i>:ipv6</code>	IPv6 indirizzo/i assegnato/i all'interfaccia.
<code>amazon:inspector:sbom_generator:metadata:host:sbomgen_tag: <i>key</i></code>	Tag personalizzati definiti dall'utente passati tramite l'argomento <code>--tag</code> CLI.
<code>amazon:inspector:sbom_generator:metadata:imds:provider</code>	Il provider cloud rilevato tramite IMDS (aws, azure, gcp).
<code>amazon:inspector:sbom_generator:metadata:imds:instance_id</code>	L'ID dell'istanza Amazon EC2 o il nome della macchina virtuale di Azure.
<code>amazon:inspector:sbom_generator:metadata:imds:instance_type</code>	Il tipo di istanza (ad esempio, t3.micro, Standard_D2S_v3).

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:metadata:imds:instance_location</code>	L' region/location istanza.
<code>amazon:inspector:sbom_generator:metadata:imds:instance_partition</code>	La partizione cloud (aws, aws-cn AWS, aws-us-gov per o AzurePublicCloud per Azure).
<code>amazon:inspector:sbom_generator:metadata:imds:account_id</code>	L'ID AWS account dell'istanza Amazon EC2, ottenuto dal documento di identità dell'istanza (AWS solo).
<code>amazon:inspector:sbom_generator:metadata:imds:resource_type</code>	Il tipo di risorsa cloud da scansionare (ad esempio, <code>aws_ec2_instance</code>).
<code>amazon:inspector:sbom_generator:metadata:imds:instance_managed_id</code>	ID dell'istanza gestita di Amazon EC2 Systems Manager AWS (solo).
<code>amazon:inspector:sbom_generator:metadata:imds:tenant_id</code>	Il tenant Microsoft Entra ID a cui appartiene la sottoscrizione dell'istanza (solo Azure, popolato quando viene assegnata un'identità gestita alla macchina virtuale).
<code>amazon:inspector:sbom_generator:metadata:imds:resource_group</code>	Il gruppo di risorse di Azure a cui appartiene l'istanza (solo Azure).
<code>amazon:inspector:sbom_generator:metadata:imds:subscription_id</code>	L'ID della sottoscrizione di Azure associato all'istanza (solo Azure).
<code>amazon:inspector:sbom_generator:metadata:imds:vm_id</code>	Identificatore univoco di macchina virtuale di Azure (solo Azure).

Proprietà	Descrizione
<code>amazon:inspector:sbom_generator:metadata:imds:project_id</code>	L'ID del progetto Google Cloud a cui appartiene l'istanza (solo GCP).
<code>amazon:inspector:sbom_generator:metadata:imds:numeric_project_id</code>	L'ID numerico del progetto Google Cloud a cui appartiene l'istanza (solo GCP).
<code>amazon:inspector:sbom_generator:metadata:imds:instance_name</code>	Il nome dell'istanza come riportato dal servizio di metadati GCP (solo GCP).
<code>amazon:inspector:sbom_generator:metadata:host:open_port: <i>port:protocol</i></code>	Indica una porta aperta di una risorsa di runtime (ad esempio EC2)
<code>amazon:inspector:sbom_generator:hardened_image:vendor</code>	Il fornitore dell'immagine di un contenitore rinforzato

Guida per sviluppatori di plugin

Questa guida spiega come estendere Amazon Inspector SBOM Generator (inspector-sbomgen) con plugin Lua personalizzati. I plugin consentono di aggiungere il supporto per nuovi ecosistemi di pacchetti senza modificare il codice sorgente di sbomgen o doverlo ricompilare.

Per il catalogo completo delle funzioni, vedi [the section called “Riferimento all'API del plugin”](#) Per indicazioni sui test di scrittura, vedere [ilthe section called “Guida al test dei plugin”](#).

Panoramica di

I plugin Sbmgen sono scritti in Lua e seguono una pipeline in due fasi:

- **Discovery:** scansiona l'elenco dei file dell'artefatto e segnala quali file sono rilevanti per il tuo ecosistema.
- **Raccolta:** analizza ogni file scoperto e inserisci i risultati del pacchetto nello SBOM.

Modello di eventi del plugin

I plugin Discovery necessitano di un modo per comunicare ai plugin di raccolta che i file contenenti i metadati dei pacchetti sono stati scoperti nell'artefatto sotto inventario. Per facilitare questa condivisione dei dati, i plugin di discovery definiscono un nome di evento e restituiscono un elenco di percorsi di file. I plugin di raccolta si iscrivono a quell'evento e ricevono ogni percorso di file corrispondente. Questo separa il rilevamento dei file dall'analisi: puoi fare in modo che un plugin di discovery alimenti più raccoglitori (modello fan-out). Tuttavia, ogni plug-in di scoperta deve avere un nome di evento univoco e ogni plug-in di raccolta deve avere un nome di raccoglitore univoco. Per informazioni dettagliate, vedi [the section called "Regole di collisione dei plugin"](#).

Gli sviluppatori potrebbero riconoscerlo come il modello di progettazione dell'osservatore.

Questo design consente a un singolo plug-in di rilevamento di attivare più analisi indipendenti in modo efficiente. Ad esempio, un plugin di scoperta può localizzare ogni elemento `requirements.txt` in un artefatto, quindi alimentare:

- Un raccoglitore di pacchetti che analizza ogni riga in risultati SBOM (). `name==version`
- Un raccoglitore di segreti che contrassegna le righe contenenti chiavi API o token aggiunti accidentalmente come versioni.
- Un raccoglitore di policy che riporta gli identificatori di versione non bloccati o con caratteri jolly.

Ogni raccoglitore viene eseguito in modo indipendente sullo stesso elenco di file senza dover ripercorrere il file system dell'elemento. Ciò consente inoltre agli autori di plug-in di aggiungere nuovi plug-in di raccolta che sottoscrivono eventi esistenti senza dover modificare il plug-in di rilevamento corrispondente.

Avvio rapido: creazione di nuovi plugin

Il modo più veloce per creare un nuovo plugin è utilizzare il comando scaffolding integrato:

```
inspector-sbomgen plugin new
```

Il comando richiede il nome del plugin e la directory del progetto. Premendo Invio si accetta l'impostazione predefinita mostrata tra parentesi:

```
Plugin name (identifies the software ecosystem your plugin will inventory, e.g. debian-dpkg, rhel-rpm, python-pip, cmake) [my-custom-ecosystem]: cmake
```

```
Project directory [my-sbomgen-plugins]:
```

Puoi anche passare direttamente gli argomenti:

```
inspector-sbomgen plugin new --name cmake --path /tmp/custom-plugins
```

A partire da un esempio funzionante

Se vuoi sperimentare con i plugin prima di scrivere la tua logica, crea un nuovo plugin usando il `--with-example` flag:

```
inspector-sbomgen plugin new --with-example
```

Questo genera un progetto di plugin completamente funzionante con un parser di lockfile di esempio, dati di test e test di superamento. Il plugin di esempio scopre `example.lock` i file, analizza le `name==version` voci e inserisce i pacchetti nella SBOM. Puoi eseguire immediatamente i test per vedere il sistema di plugin in azione, quindi modificare il codice per adattarlo al tuo ecosistema attuale.

Per uno scaffolding avanzato che includa tutte le funzioni di override opzionali (`get_scanner_name`, `get_event_name`, `get_scanner_groups`, multi-event discovery, ecc.), usa il `--with-overrides` flag (ne parleremo più avanti):

```
inspector-sbomgen plugin new --with-overrides
```

Completamento del plugin

Dopo lo scaffolding, i file dei plugin generati contengono dei TODO marcatori che indicano dove aggiungere la logica specifica dell'ecosistema. Usa questi marker per trasformare lo scaffold in un plugin funzionante:

- Sostituisci tutti i TODO marker con i tuoi valori effettivi
- Aggiorna il modello del file in `discover()` modo che corrisponda ai file di destinazione
- Implementa la logica di analisi `collect()` e richiama `sbomgen.push_package()` ogni pacchetto

Metti alla prova il tuo plugin

Esistono due modi per testare i plugin:

1. Built-in test harness — Utilizzato `inspector-sbomgen plugin test` per convalidare la logica del plugin durante lo sviluppo. Questo esegue i file di `init_test.lua` test senza bisogno di un vero artefatto da scansionare:

```
inspector-sbomgen plugin test --path ./my-plugins
```

Vedi i dettagli sulla scrittura [the section called “Guida al test dei plugin”](#) di file di test e sull'utilizzo dell'API. `testing.*`

2. End-to-end scan: richiama il plugin usando i comandi `sbomgen` standard per verificare che funzioni su artefatti reali. Per questo approccio, è necessario fornire sia un artefatto contenente i file a cui è destinato il plugin (ad esempio, una directory con `requirements.txt` o equivalente) sia il percorso della directory del plugin:

```
inspector-sbomgen directory \  
  --path /path/to/test/dir \  
  --plugin-dir ./my-plugins \  
  --disable-native-scanners \  
  -o sbom.json
```

Il `--disable-native-scanners` flag garantisce il funzionamento solo dei plugin Lua, facilitando il test senza l'output degli scanner integrati (nativi).

Configurazione IDE

Sbomgen fornisce il completamento del codice, il controllo del tipo e la documentazione in linea per l'intera `sbomgen.*` API in VS Code.

VS Code con Lua Language Server

- [Installa l'estensione Lua: `sumneko.lua`](#)
- Apri qualsiasi file nel tuo progetto di plugin `.lua`

Sono state completate tutte le operazioni! Il `plugin new` comando genera `.vscode/settings.json` e `library/sbomgen.lua` viene rilevato automaticamente dal Lua Language Server. Otterrai immediatamente:

- Completamento del codice per tutte le sbomgen . * funzioni
- Suggerimenti sui parametri con tipi
- Documentazione Hover
- Controllo dei tipi

Struttura delle cartelle dei plugin

I plugin di scoperta e raccolta di S bomgen devono rispettare la seguente struttura di directory:

```
{plugin-dir}/
### discovery/
#   ### {platform}/
#       ### {category}/
#           ### {ecosystem}/
#               ### init.lua           # REQUIRED endpoint
### collection/
#   ### {platform}/
#       ### {category}/
#           ### {ecosystem}/
#               ### init.lua           # REQUIRED endpoint
```

Questi nomi di directory hanno un significato semantico: sbomgen li usa per ricavare i metadati predefiniti per il plugin, tra cui il nome dello scanner, il nome dell'evento, i gruppi di scanner e il filtraggio della piattaforma. Ciò riduce la quantità di testo standard che gli sviluppatori dovrebbero altrimenti scrivere. La scelta dei valori corretti assicura che il plugin si integri correttamente con il modello di selezione ed esecuzione dello scanner di sbomgen.

Le sezioni seguenti esplorano la struttura delle directory in modo più dettagliato, fornendo indicazioni sul significato e sulle convenzioni semantiche.

Platform (Piattaforma)

La directory della piattaforma controlla i sistemi operativi su cui viene eseguito il plug-in.

Valore	Quando usare
cross-platform	Il plugin funziona su qualsiasi sistema operativo (la maggior parte dei plugin)

Valore	Quando usare
linux	Linux-specific logica di rilevamento
windows	Windows-specific logica di rilevamento
macos	logica OS-specific di rilevamento mac

Categoria

La directory delle categorie determina i gruppi di scanner predefiniti assegnati al plug-in, che controlla se viene eseguito per impostazione predefinita o richiede un consenso esplicito. Scoprite come [the section called "Selezione dello scanner"](#) i gruppi influiscono sull'esecuzione.

Valore	Gruppi predefiniti	Quando usare
proglang	programming-language-packages , pkg-scanner	Pacchetti di linguaggi di programmazione (pip, npm, maven, ecc.)
os	os, pkg-scanner	Gestori di pacchetti del sistema operativo (dpkg, rpm, apk, ecc.)
extra-ecosystems	extra-ecosystems , pkg-scanner	Applicazioni e runtime (nginx, curl, wordpress, ecc.)

Se si utilizza un nome di categoria che non corrisponde a nessuno dei nomi precedenti, il nome della categoria stesso viene utilizzato come gruppo.

Ecosistema

Un nome per l'ecosistema di pacchetti specifico (ad esempio python-pip, python-poetry, debian-dpkg, curl). I nomi con trattino sono una convenzione comune ma non un requisito rigoroso.

Il nome dello scanner e il nome del raccogliatore derivano direttamente dal nome della directory dell'ecosistema.

Associazione dei nomi degli eventi

I plugin di scoperta e raccolta nello stesso percorso di directory vengono associati automaticamente. Ad esempio, un plug-in di rilevamento in si accoppia `discovery/cross-platform/proglang/python-pip/` automaticamente con. `collection/cross-platform/proglang/python-pip/`

Puoi sovrascriverlo definendo `get_event_name()` e `subscribe_to_event()` nei tuoi plugin.

Plugin Discovery

Un plugin di scoperta richiede solo la `discover()` funzione. Tutte le altre funzioni sono opzionali: le impostazioni predefinite derivano dal percorso della directory.

La maggior parte dei plugin di discovery funziona localizzando file i cui nomi o percorsi identificano un ecosistema specifico, ad esempio per `requirements.txt` Python pip, `package.json` per npm o per Rust cargo. `Cargo.lock` Le `sbomgen.find_files_by_*` funzioni eseguono questa corrispondenza all'esterno della macchina virtuale Lua, il che le rende significativamente più veloci rispetto all'iterazione dell'elenco completo dei file in Lua:

```
-- REQUIRED: Scans the artifact and returns a table of file paths.
function discover()
    return sbomgen.find_files_by_name({"requirements.txt"})
end
```

`discover()` deve restituire una tabella Lua (array) di stringhe. Se non viene trovato alcun file, restituisci una tabella vuota. `{}`

Schemi di scoperta comuni

Obiettivo	Funzione consigliata
Corrisponde a uno o più nomi di file esatti	<code>sbomgen.find_files_by_name({names})</code>
Abbina i nomi dei file senza distinzione tra maiuscole e minuscole	<code>sbomgen.find_files_by_name_icense({names})</code>
Corrispondenza per suffisso di percorso (ad es.) <code>/pom.properties</code>	<code>sbomgen.find_files_by_suffix({suffixes})</code>

Obiettivo	Funzione consigliata
Corrispondenza per espressione regolare a percorso completo	<code>sbomgen.find_files_by_path_regex({patterns})</code>
Glob-style corrispondenza del nome di base (ad esempio,) <code>*.lock</code>	<code>sbomgen.glob_find_files(pattern)</code>

Quando la tua logica richiede un post-filtraggio, ad esempio, mantenendo i file corrispondenti a un suffisso ma escludendo le directory build-output, combina una chiamata con un ciclo Lua: `find_files_by_*`

```
function discover()
  local found = {}
  for _, f in ipairs(sbomgen.find_files_by_suffix({".conda-meta.json"})) do
    if not f:match("[/\\]%.cache[/\\]") then
      table.insert(found, f)
    end
  end
  return found
end
```

Evita di effettuare il discovery `sbomgen.get_file_list()` a meno che non sia adatto a nessun altro matcher: copia ogni percorso nella macchina virtuale Lua e può richiedere diversi secondi su artefatti di grandi dimensioni. Vedi per i dettagli. [the section called “Riferimento all'API del plugin”](#)

Multi-event scoperta

Per impostazione predefinita, tutti i file restituiti da `discover()` vengono pubblicati in un singolo evento (`daget_event_name()`). Se lo scanner deve indirizzare file diversi a raccoglitori diversi, restituisci invece una tabella con chiavi:

```
function discover()
  return {
    EventNameFoundCurl = sbomgen.find_files_by_name({"curl", "curl.exe"}),
    EventNameFoundLibcurl = sbomgen.find_files_by_name({"curlver.h"}),
  }
end
```

Quando `discover()` restituisce una tabella con chiavi di stringa, ogni chiave viene trattata come un nome di evento separato e il relativo valore (una tabella di percorsi di file) viene pubblicato in quell'evento. I plugin di raccolta si iscrivono a eventi specifici `subscribe_to_event()` come di consueto.

È retrocompatibile: la restituzione di una tabella sequenziale funziona `{"file1", "file2"}` ancora come modalità a evento singolo. Il rilevamento è automatico: le tabelle con qualsiasi chiave di stringa sono multievento, le tabelle con solo chiavi intere (o vuote) sono a evento singolo.

Quando si utilizza un evento multiplo, non `get_event_name()` viene utilizzato per la pubblicazione (i nomi degli eventi provengono dalle chiavi della tabella restituite). Tuttavia, viene ancora chiamato durante il caricamento del plugin per il rilevamento delle collisioni, quindi dovrebbe restituire un valore univoco o essere omissso per utilizzare l'impostazione predefinita.

Funzioni di rilevamento opzionali

Tutti questi hanno valori predefiniti validi derivati dal percorso della directory. Definiscili solo se devi sovrascrivere:

Funzione	Impostazione predefinita	Sostituisci quando...
<code>get_scanner_name()</code>	<code>{ecosystem}</code> (ad esempio, <code>python-pip</code>)	Desideri un nome personalizzato per lo scanner
<code>get_scanner_description()</code>	"Lua discovery plugin: <code>{ecosystem}</code> "	Vuoi una descrizione personalizzata
<code>get_scanner_groups()</code>	Derivato dalla directory delle categorie	Sono necessari gruppi non standard
<code>get_event_name()</code>	Derivato dal percorso della directory	È necessario un routing degli eventi personalizzato
<code>get_localhost_scan_paths()</code>	Nessuno	Il tuo plugin necessita di percorsi specifici scansionati durante le scansioni localhost

Percorsi di scansione Localhost

Quando sbomgen esegue una localhost scansione, percorre le directory specificate dall'utente e tutti i percorsi predefiniti dichiarati dagli scanner. Per impostazione predefinita, i plugin di scoperta Lua non forniscono alcun percorso, quindi i file al di fuori delle directory specificate dall'utente non verranno visualizzati nell'elenco dei file.

Definire `get_localhost_scan_paths()` di restituire le directory o i percorsi dei file che il localhost walker deve includere:

```
function get_localhost_scan_paths()
  return {
    "/usr/bin",
    "/usr/local/bin",
  }
end
```

I percorsi restituiti vengono aggiunti all'elenco di scansione del walker solo durante le localhost scansioni e non hanno alcun effetto su, o sulle scansioni. `container directory archive`

Platform-specific percorsi di scansione

Quando i file che ti interessano risiedono in posizioni diverse su Windows, macOS e Linux, esegui il branch `sbomgen.get_platform()` e restituisci i percorsi appropriati per l'host:

```
function get_localhost_scan_paths()
  local platform = sbomgen.get_platform()

  if platform == sbomgen.platform.WINDOWS then
    local drive = sbomgen.get_system_drive()
    return {
      drive .. "/Program Files/MyApp/myapp.exe",
      drive .. "/Program Files (x86)/MyApp/myapp.exe",
    }
  end

  if platform == sbomgen.platform.DARWIN then
    return {"/Applications/MyApp.app/Contents/MacOS/myapp"}
  end

  -- Linux
```

```
return {
    "/usr/bin/myapp",
    "/usr/local/bin/myapp",
}
end
```

In Windows, utilizzatela `sbomgen.get_system_drive()` per risolvere la lettera dell'unità di sistema (ad esempio "C:") anziché codificarla. Per i percorsi derivati da variabili di ambiente come `LOCALAPPDATA` or `PROGRAMFILES`, iterate `sbomgen.get_env_vars()` e cercate il valore per chiave. Vedi [the section called “Riferimento all'API del plugin”](#) per i dettagli.

Plugin di raccolta

Un plugin di raccolta richiede solo la `collect()` funzione. Tutte le altre funzioni sono opzionali.

`collect(file_path)` viene chiamato una volta per file scoperto dal plug-in di rilevamento associato. Lo schema tipico è:

- Leggi il contenuto del file usando `sbomgen.read_file()` (per file di piccole dimensioni caricati in memoria) o `sbomgen.open_file()` (per file di grandi dimensioni letti riga per riga).
- Analizza il contenuto: la corrispondenza delle stringhe per i manifesti semplici, per JSON, `sbomgen.json_decode()` per XML o per i binari `sbomgen.xml_decode()` compilati. `sbomgen.search_binary()`
- Pubblica ogni pacchetto scoperto chiamando `sbomgen.push_package()` con i metadati del pacchetto.

```
-- REQUIRED: Called once per discovered file.
-- Parse the file and call sbomgen.push_package() for each package found.
function collect(file_path)
    local content, err = sbomgen.read_file(file_path)
    if err or not content then return end

    for line in content:gmatch("[^\r\n]+") do
        local name, version = line:match("^([%w%-_%_.]+)==(.)$")
        if name and version then
            sbomgen.push_package({
                name = name,
                version = version,
                purl_type = "pypi",
            })
        end
    end
end
```

```

        component_type = sbomgen.component_types.LIBRARY,
    })
end
end
end

```

`collect()` non restituisce un valore. Ogni `push_package()` chiamata richiede `namepurl_type`, `ecomponent_type`. Consulta la sezione [the section called “Riferimento all'API del plugin”](#) per tutti i campi supportati.

Allegare metadati ai componenti

Sbomgen supporta due modi per allegare i metadati a un componente del pacchetto: i qualificatori PURL e le proprietà CyclonedX. Servono a scopi diversi e la scelta tra di essi ha implicazioni sul modo in cui Amazon Inspector identifica le vulnerabilità nella SBOM risultante.

Meccanismo	Dove appare	Utilizzare per
<code>qualifiers</code>	All'interno dell'URL del pacchetto (ad esempio, <code>pkg:deb/debian/curl@7.88.1?arch=amd64</code>)	Dati che fanno parte dell'identità del pacchetto
<code>properties</code>	Nell'array della SBOM <code>components[].properties</code>	Metadati descrittivi che non modificano il modo in cui il pacchetto viene identificato

Raccomandazione: preferisci le proprietà CyclonedX (con il tuo spazio dei nomi) per i metadati personalizzati. Le proprietà non alterano l'identità di un componente, quindi non possono influire sull'identificazione delle vulnerabilità di Amazon Inspector. Riserva i qualificatori PURL nei casi in cui il tipo PURL del tuo ecosistema li richieda.

Qualificatori PURL

Alcuni qualificatori PURL hanno un significato semantico per Amazon Inspector e influenzano l'identificazione delle vulnerabilità. Ad esempio, sui `deb` componenti Inspector utilizza qualificatori come `arch` e `distro` per selezionare il feed di vulnerabilità corretto; sui `generic` componenti

per i file binari compilati, qualificatori come `go_toolchain` o `rust_toolchain` identificano la toolchain utilizzata. L'impostazione di un qualificatore che Inspector non riconosce o l'omissione di uno previsto può far sì che le vulnerabilità non vengano riconosciute o attribuite erroneamente.

[Vedi Cos'è l'URL di un pacchetto?](#) nella guida per l'utente di Amazon Inspector per le convenzioni di qualificazione riconosciute da Inspector per tipo di PURL.

Imposta i qualificatori tramite la tabella su: `qualifiers sbomgen.push_package()`

```
sbomgen.push_package({
  name = "curl",
  version = "7.88.1",
  purl_type = "deb",
  namespace = "debian",
  component_type = sbomgen.component_types.LIBRARY,
  qualifiers = {
    arch = "amd64",
    distro = "debian-12",
  },
})
```

Imposta i qualificatori solo quando sono in linea con le aspettative di Inspector per il tipo PURL. Se devi registrare metadati che non fanno parte dell'identità del pacchetto, usa invece le proprietà CyclonedX.

proprietà CyclonedX

Le proprietà CyclonedX sono annotazioni chiave-valore che appaiono nell'array di SBOM. `components[].properties` Descrivono un componente senza influire sul modo in cui viene identificato, quindi sono la scelta sicura per i metadati definiti dal plug-in.

I **`amazon:inspector:*`** namespace sono riservati ad Amazon Inspector. Nello specifico:

- `amazon:inspector:sbom_generator:*`— riservato a `sbomgen` e ai suoi scanner integrati.
- `amazon:inspector:sbom_scanner:*`— riservato all'API Amazon Inspector Scan.

Gli autori dei plugin non devono emettere chiavi all'interno di questi namespace riservati. La scrittura in essi può interferire con il comportamento di Inspector e può essere sovrascritta. Per l'elenco completo delle chiavi riservate, consulta [Utilizzo degli spazi dei nomi CyclonedX con Amazon Inspector](#).

Usa il tuo spazio dei nomi (in genere l'identificatore dell'organizzazione o del plug-in) per definire le proprietà:

```
sbomgen.push_package({
    name = "requests",
    version = "2.28.1",
    purl_type = "pypi",
    component_type = sbomgen.component_types.LIBRARY,
    properties = {
        ["acme:python:manifest_path"] = file_path,
        ["acme:python:pinned"] = "true",
        ["acme:python:source"] = "requirements.txt",
    },
})
```

Regole di denominazione chiave

Le chiavi di proprietà vengono elaborate da sbomgen come segue:

- Una chiave che contiene i due punti viene usata letteralmente in SBOM. Includi sempre almeno un punto nelle chiavi in modo da controllare lo spazio dei nomi.
- Una chiave che non contiene i due punti viene automaticamente preceduta da `amazon:inspector:sbom_generator:` — inserendola all'interno dello spazio dei nomi riservato Inspector. Evita questa forma per le proprietà personalizzate.

```
properties = {
    ["acme:my_plugin:detected_via"] = "lockfile", -- used as-is (recommended)
    detected_via                    = "lockfile", -- becomes
    "amazon:inspector:sbom_generator:detected_via" (avoid)
}
```

Le `sbomgen.properties.*` costanti esistono in modo che gli scanner ufficiali emettano chiavi coerenti all'interno dello spazio dei nomi riservato. Non sono punti di estensione per plugin personalizzati: usa invece il tuo spazio dei nomi.

Proprietà e qualificatori sui componenti secondari

I componenti annidati `children` sono indipendenti. Ogni figlio ha le proprie `properties` `qualifiers` tabelle; i metadati impostati sul genitore non si propagano ai figli. Imposta i valori in modo esplicito per ogni bambino che ne ha bisogno.

Funzioni di raccolta opzionali

Funzione	Impostazione predefinita	Sostituisci quando...
<code>get_collector_name()</code>	{ecosystem} (ad esempio,python-pip)	Vuoi un nome da collezionista personalizzato
<code>get_collector_description()</code>	Stringa vuota	Vuoi una descrizione
<code>subscribe_to_event()</code>	Derivato dal percorso della directory	È necessario un routing degli eventi personalizzato

Esecuzione dei plugin

Affinché i plugin producano i metadati dei pacchetti, a sbomgen deve essere assegnato un artefatto da scansionare che contenga i file a cui sono destinati i plugin (ad esempio, una directory con `requirements.txt` `package.json`, o file manifest del pacchetto equivalenti).

Utilizzo di base

```
inspector-sbomgen <artifact type> <arguments> --plugin-dir /path/to/plugins
```

Esempio:

```
inspector-sbomgen directory --path /target -o /tmp/sbom.json --plugin-dir /path/to/plugins
```

Con gli scanner nativi disabilitati (modalità) Lua-only

```
inspector-sbomgen directory --path /target --plugin-dir /path/to/plugins --disable-native-scanners -o sbom.json
```

Con registrazione dettagliata

```
inspector-sbomgen directory --path /target --plugin-dir /path/to/plugins --verbose -o sbom.json
```

Elenco degli scanner disponibili

Usa `list-scanners` per vedere tutti gli scanner disponibili per sbomgen. Ciò include gli scanner nativi integrati, tutti i plugin Lua ufficiali forniti in bundle con sbomgen e tutti i plugin Lua personalizzati che hai fornito tramite: `--plugin-dir`

```
inspector-sbomgen list-scanners --plugin-dir /path/to/plugins
```

```
#####
#   SCANNER NAME       # SOURCE #           GROUPS           #           DESCRIPTION
#
#####
# curl                 # custom # extra-ecosystems       # Discovers curl version
#
#                       #         # pkg-scanner            # header files
(curlver.h) #
#####
# python-requirements # custom # pkg-scanner            # Discovers
requirements*.txt #
#                       #         # programming-language-packages # files for Python pip
#
#                       #         #                             # packages
#
#####
```

La colonna SOURCE mostra la provenienza di ogni scanner:

Origine	Significato
native	Built-in scanner fornito in dotazione con sbomgen
official	Plugin Lua in bundle con sbomgen
custom	User-provided Plugin Lua caricato tramite <code>--plugin-dir</code>

Running `list-scanners` without include `--plugin-dir` ancora entrambi native official gli scanner, sempre disponibili. La `--plugin-dir` bandiera aggiunge custom gli scanner all'elenco.

❗ I componenti dei plugin personalizzati sono contrassegnati con l'ambito CyclonedX: «opzionale»

I componenti emessi dai plugin Lua forniti dall'utente caricati tramite `--plugin-dir` sono etichettati con il campo CyclonedX nella SBOM risultante. `scope: "optional"` Questo li distingue dai componenti prodotti dagli scanner nativi integrati e dai plugin Lua ufficiali forniti in bundle con `sbomgen`, che lasciano `unset` (i consumatori preimpostano un cannocchiale mancante secondo le specifiche CyclonedX). `scope "required"` Il `scope` valore si applica a tutti i componenti emessi dal plugin, compresi i componenti secondari annidati, e non è necessario impostarlo manualmente. `sbomgen.push_package()`

Per elencare solo gli scanner Lua senza scanner nativi:

```
inspector-sbomgen list-scanners --plugin-dir /path/to/plugins --disable-native-scanners
```

Selezione dello scanner

I plugin Lua Discovery rientrano nello stesso modello di selezione degli scanner nativi integrati. Per impostazione predefinita, `sbomgen` esegue tutti gli scanner i cui gruppi corrispondono ai gruppi di scanner predefiniti per il tipo di artefatto. Puoi sovrascriverlo con tre flag:

Esegui solo scanner specifici

Utilizzare `--scanners` per eseguire solo gli scanner indicati. Sono esclusi tutti gli altri scanner:

```
inspector-sbomgen directory --path /target \  
  --plugin-dir /path/to/plugins \  
  --scanners python-requirements \  
  -o sbom.json
```

Questo fa funzionare solo lo `python-requirements` scanner. È possibile passare più nomi di scanner separati da virgole o passare il nome di un gruppo di scanner (ad esempio `programming-language-packages`) per abilitare tutti gli scanner che appartengono a quel gruppo.

Escludi scanner specifici

Usa `--skip-scanners` per escludere gli scanner con nome mentre esegui tutto il resto:

```
inspector-sbomgen directory --path /target \  
  --plugin-dir /path/to/plugins \  
  --skip-scanners python-poetry \  
  -o sbom.json
```

Questo esegue tutti gli scanner predefiniti tranne `python-poetry`. Ad esempio `--scanners`, questo flag accetta anche nomi di gruppo, quindi il passaggio `--skip-scanners programming-language-packages` disabilita tutti gli scanner di quel gruppo.

Note

`--scanner` e `--skip-scanners` escludono a vicenda. Il passaggio di entrambi produce un errore.

Aggiungere scanner da gruppi non predefiniti

Il set di scanner predefinito dipende dal tipo di artefatto da scansionare (vedi la matrice riportata di seguito). [the section called “In che modo i gruppi influiscono sulla selezione”](#) Uno scanner i cui gruppi non fanno parte del set predefinito per il tipo di artefatto non funzionerà a meno che non venga attivato. `--additional-scanners` Utilizzatelo per aggiungere gli scanner al set predefinito senza sostituirlo:

```
inspector-sbomgen directory --path /target \  
  --plugin-dir /path/to/plugins \  
  --additional-scanners my-extra-scanner \  
  -o sbom.json
```

Questo esegue tutti gli scanner predefiniti per il tipo di artefatto, e in più. `my-extra-scanner` Il flag accetta un elenco separato da virgole di nomi di scanner o nomi di gruppi e si aggiunge al set predefinito anziché sostituirlo. Si usa `list-scanners` per controllare a quali gruppi appartiene uno scanner.

In che modo i gruppi influiscono sulla selezione

La `get_scanner_groups()` funzione del plugin di scoperta determina a quali gruppi appartiene lo scanner. Il funzionamento predefinito di uno scanner dipende sia dai suoi gruppi che dal tipo di artefatto sottoposto a scansione. La matrice seguente mostra quali gruppi sono inclusi nel set di scanner predefinito per ogni tipo di artefatto:

Group (Gruppo)	directory / archive	container	localhost	volume	binary
os	—	✓	✓	✓	—
programming-language-packages	✓	✓	✓	✓	—
binary	✓	✓	—	—	✓
extra-ecosystems	—	✓	✓	✓	—
dockerfiles	✓	✓	—	—	—
custom	✓	✓	✓	✓	✓
certificates	—	—	—	—	—
machine-learning	—	—	—	—	—
pkg-scanner	—	—	—	—	—

A ✓ indica che ogni scanner di quel gruppo viene eseguito per impostazione predefinita per quel tipo di artefatto. A — significa che il gruppo non è nel set predefinito, quindi i suoi scanner funzionano solo se selezionati esplicitamente tramite `o. --scanners --additional-scanners`

Dettagli importanti:

- **custom** è sempre nel set predefinito: i plugin personalizzati caricati tramite `ricevono --plugin-dir` automaticamente il custom gruppo, quindi vengono eseguiti per impostazione predefinita indipendentemente dal tipo di artefatto.

- **extra-ecosystems** è l'impostazione predefinita per container `localhost`, e `volume` `scansiona`, ma non per `directory`, `archive` o `scansiona.binary`. Per questi tipi è necessario passare `--additional-scanners` (per nome o per `extra-ecosystems` gruppo) per includerli.
- **pkg-scanner** è informativo: contrassegna lo scanner come raccoglitore di pacchetti in cui visualizzarlo `list-scanners`, ma di per sé non causa il funzionamento dello scanner. Associa a un gruppo di esecuzione (ad esempio, `programming-language-packages`) in `get_scanner_groups()`

Ad esempio, un plug-in che restituisce `{sbomgen.groups.EXTRA_ECOSYSTEMS, sbomgen.groups.PACKAGE_COLLECTOR}` verrà eseguito per impostazione predefinita sulle scansioni di contenitori, `localhost` e volumi, ma richiederà `--additional-scanners (o --scanners)` sulle scansioni di `directory`, archivi e binarie.

Regole di collisione dei plugin

Sbomgen applica metadati unici su tutti i plugin caricati per prevenire sovrascritture silenziose e garantire l'integrità SBOM. Quando viene rilevata una collisione, il plugin successivo viene ignorato e viene registrato un avviso.

Cosa viene controllato

Metadati	Scope (Ambito)	In caso di collisione
Nome dell'evento Discovery () <code>get_event_name</code>	Tutti i plugin di scoperta	Il secondo plugin è stato ignorato
Nome dello scanner () <code>get_scanner_name</code>	Tutti i plugin di scoperta	Il secondo plugin è stato ignorato
Nome del collezionista () <code>get_collector_name</code>	Tutti i plugin di raccolta	Il secondo plugin è stato ignorato

Cosa è permesso

Più plugin di raccolta possono iscriversi allo stesso evento tramite `subscribe_to_event()`. Questo è lo schema di fan-out previsto: un plugin di discovery può alimentare più raccoglitori, ognuno dei quali fa cose diverse (ad esempio, uno estrae pacchetti, un altro rileva segreti).

Evitare le collisioni

Se due plugin utilizzano lo stesso nome di scanner, nome di evento o nome di raccoglitore, il secondo plugin caricato viene saltato. Per risolvere le collisioni, rinomina i metadati in conflitto definendo la funzione di override appropriata nel tuo plugin (, or). `get_scanner_name()` `get_event_name()` `get_collector_name()`

Esempio di avviso di collisione

```
[custom:python-pip] SKIPPED: discovery event name "EventNameFoundPythonRequirements"
is already registered by [official:python-pip]. Each discovery plugin must have a
unique event name. Rename get_event_name() in your plugin to use a unique name.
```

L'avviso indica quale plugin è stato ignorato, cosa è entrato in collisione, quale plugin possiede già quel nome e quale funzione modificare.

Debug

Registrazione di log della console

I plugin possono inviare messaggi all'output della console di sbomgen usando le seguenti funzioni:

Funzione	Livello	Visibile per impostazione predefinita?
<code>sbomgen.log_debug(message)</code>	DEBUG	No, richiede <code>--verbose</code>
<code>sbomgen.log_info(message)</code>	INFO	Sì
<code>sbomgen.log_warn(message)</code>	WARN	Sì
<code>sbomgen.log_error(message)</code>	ERRORE	Sì

Tutti gli output di log di un plugin sono automaticamente preceduti dall'origine e dal percorso del plugin (ad esempio, `[custom:python-pip]`), in modo che i messaggi provenienti da diversi plugin

siano facili da distinguere. `log_info`, `log_warn`, e stampa `log_error` sempre; stampa `log_debug` solo quando viene invocato `sbomgen`. `--verbose`

```
function discover()
  sbomgen.log_info("starting discovery")
  local files = sbomgen.find_files_by_name({"requirements.txt"})
  sbomgen.log_debug(string.format("matched %d files", #files))
  if #files == 0 then
    sbomgen.log_warn("no requirements.txt files found")
  end
  return files
end
```

Punti di interruzione

Si usa `sbomgen.breakpoint()` per mettere in pausa l'esecuzione del plugin e bloccarlo finché non si preme Invio. Funziona come un semplice debugger: combinalo con le istruzioni di registro per ispezionare lo stato in punti specifici.

```
function discover()
  local files = sbomgen.find_files_by_name({"requirements.txt"})

  sbomgen.log_info(string.format("about to inspect %d files", #files))
  sbomgen.breakpoint("before file inspection - press Enter to continue")

  local found = {}
  for _, f in ipairs(files) do
    if not f:match("[\\/]tests[\\/]") then
      table.insert(found, f)
    end
  end

  sbomgen.log_info(string.format("kept %d files after filtering", #found))
  sbomgen.breakpoint("after filtering - press Enter to continue")

  return found
end
```

Il messaggio di breakpoint viene stampato su `stderr`. L'esecuzione si interrompe finché non si preme Invio, dandovi il tempo di esaminare l'output del registro.

Problemi comuni

Sintomo	Causa	Correggere
Plugin non caricato	<code>init.lua</code> mancante	Assicurati che il punto di ingresso esista alla profondità corretta della directory
«funzione richiesta mancante»	Errore di battitura nel nome della funzione	Verifica che <code>get_scanner_name</code> , <code>get_scanner_description</code> , <code>get_scanner_groups</code> , <code>discover</code> , <code>get_event_name</code> , <code>get_local_host_scan_paths</code> , <code>get_collector_name</code> , <code>collect</code> , <code>subscribe_to_event</code> siano definiti
Il plugin di raccolta non è mai stato chiamato	Mancata corrispondenza del nome dell'evento	Verifica <code>get_event_name()</code> e <code>subscribe_to_event()</code> restituisci la stessa stringa
Nessun pacchetto in SBOM	<code>push_package</code> campi non chiamati o obbligatori mancanti	Assicurati <code>name</code> e <code>url_type</code> <code>component_type</code> sono impostati in ogni <code>push_package</code> chiamata (bambini compresi). Usa le <code>sbomgen.component_types.*</code> costanti.
Errore di runtime nel plugin	Errore Lua durante l'esecuzione	Controlla l'output di <code>sbomgen</code> per i messaggi di avviso con i dettagli dell'errore

Sintomo	Causa	Correggere
«SKIPPED: il nome dell'evento di scoperta... è già registrato»	Un altro plugin utilizza lo stesso nome di evento	Rinomina <code>get_event_name()</code> con un valore univoco
«SKIPPED: il nome dello scanner... è già registrato»	Un altro plugin utilizza lo stesso nome dello scanner	Rinomina <code>get_scanner_name()</code> con un valore univoco
«SKIPPED: il nome del collezionista... è già registrato»	Un altro plugin utilizza lo stesso nome del raccoglitore	Rinomina con un <code>get_collector_name()</code> valore univoco

Documentazione di riferimento delle API

Il catalogo completo delle funzioni è conservato in un documento complementare:

→ [the section called “Riferimento all'API del plugin”](#)

Il riferimento all'API copre ogni sbomgen.* funzione (file I/O, utilità binarie, output di pacchetti, regex, analisi strutturata, registro di Windows, registrazione, debug), l'`testing.*` API disponibile nei file di test, tutte le costanti integrate (,,,) e i dati globali del ciclo di `properties` vita dei `groups` `plugin-component_types.platform`

Gestione errori

Le funzioni API che possono fallire restituiscono due valori: `value`, `err` In caso di successo, `err` è `nil`. In caso di errore, il primo valore è `nil` ed `err` è una stringa di errore.

```
local content, err = sbomgen.read_file(path)
if err then
    sbomgen.log_error("failed to read " .. path .. ": " .. err)
    return
end
-- content is safe to use here
```

Se un plugin genera un errore Lua non gestito, sbomgen registra un avviso e continua con il file o il plugin successivo. Gli altri plugin non sono interessati.

Restrizioni Sandbox

I plugin vengono eseguiti in una macchina virtuale Lua in modalità sandbox con accesso limitato alla libreria standard:

Libreria	Disponibilità	Note
base	✓	dofileloadfileloadstring, vengono rimossi
string	✓	Manipolazione completa delle stringhe
table	✓	Manipolazione completa della tabella
math	✓	Libreria matematica completa
package	✓	require() limitato alla directory dei plugin
io	✗	Usa invece sbomgen.* I/O le funzioni
os	✗	Bloccato per motivi di sicurezza
debug	✗	Bloccato per impedire l'introspezione delle VM
coroutine	✗	Non caricato

L'accesso diretto al file system tramite `io.open` o `non os.execute` è disponibile. Tutte le operazioni sui file devono passare attraverso l'`sbomgenAPI`, che garantisce un comportamento coerente tra i tipi di artefatti e impedisce ai plugin di accedere ai file esterni all'artefatto.

`require()` può caricare i moduli solo dall'interno dell'albero di directory del plugin. Parent-directory attraversamento, ad esempio `bloccatorequire("../shared")`.

Condivisione del codice tra plugin

Puoi usare `require()` per caricare i moduli di supporto dall'interno della `directory` del tuo plugin:

```
my-ecosystem/  
### init.lua  
### helpers.lua
```

```
-- helpers.lua  
local M = {}  
function M.parse_version(s)  
    return string.match(s, "(%d+%.%d+%.%d+)")  
end  
return M
```

```
-- init.lua  
local helpers = require("helpers")  
  
function subscribe_to_event() return "MyEvent" end  
  
function collect(file_path)  
    local content, err = sbomgen.read_file(file_path)  
    if err then return end  
    local version = helpers.parse_version(content)  
    -- ...  
end
```

Sono supportate anche le sottodirectory con `init.lua`:

```
my-ecosystem/  
### init.lua  
### parsers/  
    ### init.lua
```

```
local parsers = require("parsers")
```

`require()` è limitato alla cartella del tuo plugin. Non è possibile caricare moduli da altri plugin o percorsi di sistema. Third-party Le librerie Lua (ad esempio, from LuaRocks) non sono supportate: possono essere caricati solo i moduli di supporto locali all'interno della `directory` dei plugin.

Guida al test dei plugin

Gli autori dei plugin scrivono e testano i plugin Lua interamente in Lua, senza bisogno della toolchain Go. I test sono collocati insieme al plugin sotto `init_test.lua` ed eseguiti tramite il comando.

```
inspector-sbomgen plugin test
```

Per la creazione generale dei plugin, consulta [the section called “Guida per sviluppatori di plugin”](#) Per il catalogo completo delle funzioni (inclusa l'`testing.*API`), consulta [the section called “Riferimento all'API del plugin”](#).

```
-- init_test.lua (next to init.lua)
function test_discovers_curl_version()
    local result = testing.scan_directory("_testdata/include/curl")
    testing.assert_equals(1, #result.findings)
    testing.assert_equals("libcurl", result.findings[1].name)
    testing.assert_equals("8.14.1", result.findings[1].version)
end
```

```
# Run every test under a plugin directory
inspector-sbomgen plugin test --path ./my-plugins

# Verbose output – show each test name and result
inspector-sbomgen plugin test --path ./my-plugins -v
```

Quick Start

1. Crea un file di test

Posiziona `init_test.lua` accanto a quello del tuo plugin `init.lua`:

```
my-plugin/
### discovery/cross-platform/extra-ecosystems/curl/
#   ### init.lua
#   ### init_test.lua
#   ### _testdata/
#       ### include/curl/curlver.h
```

2. Scrivi funzioni di test

Qualsiasi funzione globale che inizia con `test_` viene rilevata ed eseguita:

```
function test_finds_libcurl()
    local result = testing.scan_directory("_testdata/include/curl")
    testing.assert_equals(1, #result.findings)
    testing.assert_equals("libcurl", result.findings[1].name)
end

function test_no_findings_for_empty_dir()
    local result = testing.scan_directory("_testdata/empty")
    testing.assert_equals(0, #result.findings)
end
```

3. Esegui test

```
inspector-sbomgen plugin test --path ./my-plugin
```

Layout delle directory

Struttura del plugin

I file di test e i dati di test si trovano insieme al plugin:

```
my-plugins/
### discovery/
#   ### cross-platform/
#       ### extra-ecosystems/
#           ### curl/
#               ### init.lua           # plugin source
#               ### init_test.lua     # test file
#               ### _testdata/        # test fixtures
#               ### include/curl/curlver.h
#               ### binaries/unix/curl
### collection/
#   ### cross-platform/
#       ### extra-ecosystems/
#           ### curl-installation/
#               ### init.lua
#               ### init_test.lua
```

Denominazione dei file di test

- Predefinito: `init_test.lua` accanto al plugin `init.lua`

- Più file di test per plugin: *_test.lua viene rilevata qualsiasi corrispondenza tra file
- Esempi: init_test.lua, parsing_test.lua, discovery_test.lua

Dati di test: **_testdata/**

I dati di test si `_testdata/` trovano accanto al plugin. La caratteristica principale è una convenzione che mantiene le fixture visivamente separate dalla sorgente del plugin; il `plugin test` comando non viene utilizzato `_testdata/` durante la ricerca di *_test.lua file, quindi le fixture non vengono mai scambiate per file di test.

I file di test fanno riferimento a dispositivi con percorsi relativi:

```
local result = testing.scan_directory("_testdata/include/curl")
```

I percorsi vengono risolti in relazione alla directory contenente il file di test.

L'API **testing.***

Funzioni di scansione

Ogni funzione di scansione crea un artefatto, esegue la pipeline discovery → collection del plugin e restituisce i risultati. L'autore del test non crea mai manualmente artefatti, bus di eventi o registri.

```
-- Scan a directory of test fixtures (most common)
local result = testing.scan_directory("_testdata/curl")

-- Alias for scan_directory (archives use the same backend)
local result = testing.scan_archive("_testdata/app.tar.gz")

-- Scan as a localhost artifact
local result = testing.scan_localhost("_testdata/curl")

-- Scan a compiled binary
local result = testing.scan_binary("_testdata/binaries/curl")

-- Scan a mounted volume
local result = testing.scan_volume("_testdata/volume-root")

-- Scan a container image tarball
```

```
local result = testing.scan_container("_testdata/images/alpine.tar")
```

Ogni funzione di scansione:

- Crea un nuovo artefatto per ogni chiamata (nessuna perdita di stato tra le chiamate)
- Carica solo la coppia di raccolta discovery+ del plugin corrente
- Restituisce una tabella dei risultati

Tabella dei risultati

```
result.findings          -- array of finding tables
result.findings[1].name  -- package name (string)
result.findings[1].version -- package version (string)
result.findings[1].purl  -- package URL (string)
result.findings[1].component_type -- component type (string)
result.findings[1].properties -- table<string, string>
result.findings[1].children -- array of nested finding tables (same shape)
```

Asserzioni

```
-- Equality
testing.assert_equals(expected, actual, message?)
testing.assert_not_equals(expected, actual, message?)

-- Truthiness
testing.assert_true(value, message?)
testing.assert_false(value, message?)

-- Nil checks
testing.assert_nil(value, message?)
testing.assert_not_nil(value, message?)

-- String
testing.assert_contains(haystack, needle, message?)
testing.assert_matches(string, pattern, message?)

-- Tables
testing.assert_length(table, expected_length, message?)

-- Control flow
```

```
testing.fail(message)    -- immediately fail the current test
testing.skip(message)   -- skip the current test (not a failure)
```

sbomgen.*API standard

L'`sbomgen.*API` completa (file I/O, regex, informazioni di sistema, registrazione, ecc.) è disponibile nei file di test, come nei plugin di produzione. Tuttavia, `sbomgen.*` le funzioni che richiedono un artefatto (ad esempio `sbomgen.read_file()`) funzionano solo all'interno di un `testing.scan_*` callback: non sono disponibili al livello superiore di una funzione di test.

Esecuzione di test.

```
# Run all tests under a plugin directory
inspector-sbomgen plugin test --path ./my-plugins

# Filter by regex pattern
inspector-sbomgen plugin test --path ./my-plugins --run curl

# Verbose output (show each test name and result)
inspector-sbomgen plugin test --path ./my-plugins -v

# Stop on the first failing test
inspector-sbomgen plugin test --path ./my-plugins --fail-fast
```

Il `--path` flag accetta una directory principale del plug-in (contenente `discovery/` and/or `collection/`) o una singola directory dell'ecosistema (rilevata automaticamente). Il comando esce da un valore diverso da zero se un test fallisce.

Formato di output

Con `-v`, ogni test stampa una `=== RUN` riga e una riga di risultato (`--- PASS---` `FAIL`, o `--- SKIP`). Senza `-v`, vengono stampati solo i test non riusciti. Alla fine viene stampata una riga riassuntiva:

```
=== RUN   curl/discovery/init_test/test_discovers_libcurl_header
--- PASS: curl/discovery/init_test/test_discovers_libcurl_header (0.04s)
=== RUN   curl/discovery/init_test/test_discovers_curl_binary_unix
--- PASS: curl/discovery/init_test/test_discovers_curl_binary_unix (0.04s)
=== RUN   curl/discovery/init_test/test_no_findings_for_unrelated_files
--- PASS: curl/discovery/init_test/test_no_findings_for_unrelated_files (0.04s)
```

```
ok    3 tests passed
```

Testare gli helper dei plugin

Il file di test viene caricato nella stessa macchina virtuale Lua del plugin. `init.lua` Le funzioni globali definite nel plugin sono richiamabili dai test. Per testare le funzioni di supporto, esponile come globali o in una tabella di moduli:

```
-- init.lua
M = {}
function M.parse_version(raw)
    return string.match(raw, "(%d+%.%d+%.%d+)")
end

function collect(file_path)
    local ver = M.parse_version(...)
    -- ...
end
```

```
-- init_test.lua
function test_parse_version_extract_semver()
    testing.assert_equals("1.2.3", M.parse_version("curl/1.2.3"))
end

function test_parse_version_returns_nil_for_garbage()
    testing.assert_nil(M.parse_version("not-a-version"))
end
```

Le funzioni dichiarate `local` in `init.lua` sono visibili nel file di test. Questo è lo scoping Lua standard.

Comportamento e invarianti

VM condivisa, stato condiviso

Le funzioni di test all'interno di un singolo file condividono una VM Lua. Una variabile globale impostata in una `test_*` funzione è visibile alle funzioni successive. Se due funzioni definiscono lo stesso globale, la seconda sovrascrive la prima. Ogni test deve essere autonomo e non dipendere dallo stato degli altri test.

Ordine di esecuzione non deterministico

Le funzioni di test vengono scoperte iterando la tabella globale di Lua, che utilizza l'ordinamento basato su hash. Non è garantito che i test vengano eseguiti nell'ordine in cui sono stati definiti. Non scrivete test che dipendono dall'ordine di esecuzione.

Nuovi artefatti per ogni chiamata di scansione

Ogni chiamata a `testing.scan_directory()` (o qualsiasi funzione di scansione) crea un artefatto completamente nuovo. Non viene rilevato alcuno stato tra le chiamate di scansione all'interno di un test o tra i test.

Caricamento del plugin

I plugin vengono caricati una volta per esecuzione di test, non una volta per file di test. Il test runner carica tutti i plugin dal filesystem fornito, quindi abbina ogni file di test alla macchina virtuale del plugin corrispondente per fase, piattaforma, categoria ed ecosistema.

Comportamento di asserzione

Quando un'asserzione fallisce, l'errore viene registrato ma la funzione di test continua a essere eseguita e le asserzioni e le istruzioni successive continuano a essere eseguite. Se più di un'asserzione fallisce nello stesso test, l'errore più recente è il messaggio riportato nel riepilogo; gli errori precedenti vengono sovrascritti. Per interrompere un test al primo errore, tornate dalla funzione dopo l'asserzione fallita (o `testing.fail()` usatela all'interno di un condizionale).

Limitazioni

- **locale** funzioni non sono testabili. Solo le funzioni globali di `init.lua` sono visibili nel file di test. Esponete gli helper tramite una tabella di moduli se hanno bisogno di essere testati.
- Nessun **sbomgen.*** file I/O al di fuori delle chiamate di scansione. Funzioni come quelle `sbomgen.read_file()` richiedono un contesto di artefatti, che esiste solo all'interno `testing.scan_*` delle chiamate.
- Nessun aggancio al ciclo di vita. Non c'è `before_each`, `after_each`, `setup` o `teardown`. Ogni funzione di test gestisce il proprio stato.
- Nessun timeout per il test. Una funzione di test che si ripete per sempre bloccherà il runner.
- Nessuna segnalazione sulla copertura. Non c'è modo di misurare quali linee `init.lua` sono state esercitate.

- Nessun benchmark. Il framework di test non supporta i benchmark delle prestazioni.

Responsabilità degli sviluppatori

Quando si scrive un nuovo plugin Lua

- Crea `init_test.lua` accanto al tuo `init.lua`
- Crea `_testdata/` con dispositivi minimi che esercitano la logica del tuo plugin
- Scrivi `test_*` funzioni che coprono: rilevamento riuscito, estrazione della versione, casi limite e scenari senza corrispondenza
- Esegui i test localmente prima di inviarli

Linee guida sui dati dei test

- Riduci al minimo le apparecchiature: utilizza il file più piccolo che esercita il comportamento
- Evita di inserire file binari di grandi dimensioni in `_testdata/` cui sarebbe sufficiente una piccola finestra di testo
- Ogni plugin `_testdata/` dovrebbe essere autonomo, senza riferimenti a file al di fuori della directory dei plugin

Riferimento all'API del plugin

Riferimento API completo per i plugin Lua inspector-sbomgen. Per una guida sulla scrittura di plugin, vedi. [the section called “Guida per sviluppatori di plugin”](#) Per i test, vedi [the section called “Guida al test dei plugin”](#).

Panoramica di

È possibile accedere a tutte le funzioni fornite in fase di esecuzione tramite la sbomgen tabella globale (file I/O, regex, registrazione, costanti, ecc.). Inoltre, ogni plugin definisce un piccolo insieme di funzioni globali di primo livello (`discover`, `collect` `get_scanner_names` `subscribe_to_event`, e così via) che sbomgen chiama in punti definiti del ciclo di vita del plugin. Queste sono documentate in. [the section called “Plugin Lifecycle Globals”](#)

All'interno `*_test.lua` dei file, sbomgen presenta inoltre una funzionalità `testing` globale che consente agli autori dei test di guidare la pipeline `discovery`→`collection` e fare asserzioni. Per informazioni, consulta [the section called “API di test”](#).

Restrizioni della sandbox

I plugin vengono eseguiti in una macchina virtuale Lua in modalità `sandbox` con accesso limitato alla libreria standard. Sono disponibili i seguenti moduli di libreria standard Lua:

Modulo	Note
<code>base</code>	Funzioni principali (<code>printtype</code> , <code>tostring</code> , <code>tonumber</code> , <code>pairs</code> , <code>ipairspcall</code> , <code>error</code> , <code>select</code> , <code>unpack</code> , <code>rawget</code> , <code>rawse</code> ecc.). <code>dofile</code> <code>loadfile</code> , e <code>loadstring</code> vengono rimossi.
<code>string</code>	Manipolazione completa delle stringhe (<code>string.match</code> <code>string.find</code> <code>string.format</code> „ <code>string.gsub</code> „ ecc.)
<code>table</code>	Manipolazione completa della tabella (<code>table.insert</code> <code>table.remove</code> „ <code>table.sort</code> <code>table.concat</code> „ ecc.)
<code>math</code>	Libreria matematica completa (<code>math.floor</code> „ <code>math.max</code> <code>math.min</code> , ecc.)
<code>package</code>	<code>require()</code> è disponibile ma limitato ai moduli all'interno dell'albero di <code>directory</code> del plugin. Parent-directory traversal (<code>require("../shared")</code>) è bloccato. <code>package.cpath</code> e <code>package.path</code> vengono cancellati.

I seguenti moduli di libreria standard non sono esplicitamente consentiti per motivi di sicurezza e stabilità:

Modulo	Motivo
io	L'accesso diretto al file system è bloccato. Tutte le operazioni sui file devono passare attraverso <code>sbomgen.*</code> funzioni, che passano attraverso o l'interfaccia Artifact per un comportamento coerente tra i tipi di artefatti (directory, contenitore, volume, ecc.).
os	System-level le operazioni (<code>os.execut</code> e <code>os.remove</code> , <code>os.rename</code> <code>os.getenv</code> , ecc.) sono bloccate per impedire ai plugin di modificare il sistema host.
debug	La libreria di debug è bloccata per impedire l'ispezione o la modifica degli interni della VM Lua.
coroutine	Le coroutine non vengono caricate.

Questi moduli non sono nella lista consentita della VM e non sono accessibili dai plugin.

Note

Importante: tutti i file I/O devono essere sottoposti a `sbomgen.*` funzioni (ad esempio, `sbomgen.read_file`, `sbomgen.open_file`). `sbomgen.get_file_list` L'utilizzo `io.open` o l'accesso diretto al filesystem genererà un errore di runtime. L'`sbomgenAPI` assicura che i plugin interagiscano con il livello di astrazione degli artefatti, che fornisce un comportamento coerente durante la scansione di una directory, di un'immagine del contenitore, di un archivio o di un volume.

Plugin Lifecycle Globals

Un plugin è un file Lua denominato `init.lua` che definisce alcune funzioni globali di primo livello. Questi valori globali non sono sul `sbomgen` tavolo: sono funzioni che il plugin definisce per la chiamata a `sbomgen`. L'insieme di globals validi differisce tra i plugin di scoperta e i plugin di raccolta.

Per ogni funzione riportata di seguito, se il plugin la omette, viene utilizzata l'impostazione predefinita mostrata nella tabella.

Plugin Discovery

Funzione	Arity	Campo obbligatorio	Predefinito (se omissivo)	Descrizione
<code>discover()</code>	0	Si	—	Restituisce i file trovati da questo plugin. Restituisce una tabella sequenziale di stringhe di percorso (modalità a evento singolo) o una tabella composta da stringhe di nomi di eventi i cui valori sono tabelle di percorsi (modalità multievento).
<code>get_event_name()</code>	0	No	"lua:{platform}/{category}/{ecosystem}"	Restituisce il nome dell'evento con cui vengono pubblicati i file. Deve essere unico per tutti i plugin di discovery.

Funzione	Arity	Campo obbligatorio	Predefinito (se omissivo)	Descrizione
<code>get_scanner_name()</code>	0	No	nome della directory dell'ecosistema	Restituisce il nome visualizzato dello scanner. Deve essere unico per tutti i plugin di discovery.
<code>get_scanner_description()</code>	0	No	"Lua discovery plugin: {ecosystem}"	Restituisce una descrizione leggibile dall'uomo.
<code>get_scanner_groups()</code>	0	No	Derivato dalla directory delle categorie (consulta la guida per gli sviluppatori)	Restituisce una tabella di stringhe di gruppi di scanner. Usa <code>sbongen.groups.*</code> costanti.
<code>get_localhost_scan_paths()</code>	0	No	—	Restituisce una tabella di file/directory percorsi da includere durante la scansione di un artefatto localhost. Consultato solo per le scansioni. localhost

Plugin di raccolta

Funzione	Arity	Campo obbligatorio	Predefinito (se omissivo)	Descrizione
<code>collect(file_path)</code>	1	Sì	—	Chiamato una volta per file pubblicato nell'evento sottoscritto. Analizza il file ed emette i risultati tramite <code>sbomgen.push_package()</code> Non restituisce nulla.
<code>subscribe_to_event()</code>	0	No	"lua:{platform}/{category}/{ecosystem}"	Restituisce il nome dell'evento a cui questo raccogliatore è abbonato. Dovrebbe corrispondere al plugin di scoperta corrispondente. <code>get_event_name()</code>
<code>get_collector_name()</code>	0	No	nome della directory dell'ecosistema	Restituisce il nome visualizzato del raccogliatore. Deve essere unico per

Funzione	Arity	Campo obbligatorio	Predefinito (se omissivo)	Descrizione
				tutti i plugin di raccolta.
<code>get_collector_description()</code>	0	No	""(vuoto)	Restituisce una descrizione leggibile dall'uomo.

File I/O

Tutte le operazioni sui file devono passare attraverso l'`sbomgen.*API`. L'accesso diretto al file system tramite la `io` libreria di Lua non è disponibile (vedi). [the section called “Restrizioni della sandbox”](#) I/O Le funzioni dei `sbomgen` file vengono instradate attraverso l'interfaccia `Artifact`, assicurando che il plugin funzioni in modo identico sia che si esegua la scansione di una directory su disco, di un'immagine di un contenitore, di un archivio compresso o di un volume montato.

`sbomgen.get_file_list ()`

Restituisce tutti i percorsi dei file nell'artefatto come tabella di stringhe.

- Restituisce: `{string, ...}` — tabella di stringhe di percorsi assoluti dei file
- Prestazioni: questa funzione copia ogni percorso di file nell'artefatto nella macchina virtuale Lua come stringa Lua. Su artefatti di grandi dimensioni (ad esempio, una scansione localhost con oltre 300.000 file), solo questo richiede diversi secondi. L'iterazione della tabella restituita in Lua con un `string.match()` ulteriore sovraccarico: una scansione completa può richiedere più di 15 secondi. Maggiore è il numero di file nell'artefatto, più lento sarà il plugin.

Note

Preferisci queste alternative mirate quando possibile:

Funzione	Usa quando...
<code>sbomgen.find_files_by_name()</code>	Conosci i nomi esatti dei file da abbinare (ad esempio, "requirements.txt") "curl"
<code>sbomgen.find_files_by_name_icode()</code>	Come sopra, ma senza distinzione tra maiuscole e minuscole
<code>sbomgen.find_files_by_suffix()</code>	È necessario che i suffissi del percorso corrispondano (ad es.,) "/pom.properties" "curlver.h"
<code>sbomgen.find_files_by_path_regex()</code>	È necessaria la corrispondenza delle espressioni regolari a percorso completo
<code>sbomgen.glob_find_files()</code>	È necessaria una corrispondenza dei nomi di base in stile glob

Queste funzioni eseguono la corrispondenza all'esterno della macchina virtuale Lua e restituiscono solo i percorsi corrispondenti, completandosi in meno di 1 millisecondo anche su 300 artefatti. K-file Utilizza `get_file_list()` solo quando la tua logica di corrispondenza non può essere espressa con nessuno dei precedenti.

```
-- AVOID in discovery plugins when possible:
local files = sbomgen.get_file_list()
for _, f in ipairs(files) do
    if string.match(f, "pattern$") then ... end
end

-- PREFER:
local matches = sbomgen.find_files_by_name({"target-file.txt"})
```

sbomgen.read_file (percorso)

Legge l'intero contenuto di un file e lo restituisce come stringa.

- Restituisce: `string`, `err`
- In caso di fallimento: `nil`, `error_string`

```
local content, err = sbomgen.read_file("/app/package.json")
if err then
  sbomgen.log_error("read failed: " .. err)
  return
end
```

sbomgen.open_file (percorso)

Apri un file per le letture in streaming. Restituisce un `FileHandle` oggetto. Usalo per file di grandi dimensioni in cui caricare l'intero contenuto in memoria non è pratico.

- Restituisce: `FileHandle`, `err`

```
local fh, err = sbomgen.open_file(path)
if err then return end
local line = fh:read_line()
while line do
  -- process line
  line = fh:read_line()
end
fh:close()
```

sbomgen.glob_find_files (modello)

Restituisce i file che corrispondono a un pattern Go glob. `filepath.Match` Il modello viene confrontato con il nome del file di base.

- Restituisce: `{string, ...}`, `err`

```
local files, err = sbomgen.glob_find_files("*.txt")
```

Usa `sbomgen.get_file_list()` with `string.match` per la corrispondenza completa del modello di percorso.

sbomgen.find_files_by_name (nomi)

Restituisce file il cui nome di base (ultimo componente del percorso) corrisponde esattamente a uno dei nomi indicati. L'iterazione e il confronto avvengono in Go, il che lo rende significativamente più veloce rispetto all'`sbomgen.get_file_list()` iterazione in Lua.

- Parametri: `names` — tabella di stringhe (nomi di base da abbinare)
- Restituisce: `{string, ...}` — percorsi di file corrispondenti (nessuna tupla di errore)

```
local curl_bins = sbomgen.find_files_by_name({"curl", "curl.exe"})
local headers = sbomgen.find_files_by_name({"curlver.h"})
```

sbomgen.find_files_by_name_icode (nomi)

Restituisce i file il cui nome di base corrisponde a uno dei nomi indicati, ignorando le maiuscole e minuscole. Ad esempio, "version" corrisponde a, VERSION eVersion. version Ad esempio `find_files_by_name`, l'abbinamento avviene al di fuori della macchina virtuale Lua.

- Parametri: `names` — tabella di stringhe (nomi di base da abbinare, senza distinzione tra maiuscole e minuscole)
- Restituisce: `{string, ...}` — percorsi di file corrispondenti (nessuna tupla di errore)

```
local version_files = sbomgen.find_files_by_name_icode({"version"})
local war_files = sbomgen.find_files_by_name_icode({"jenkins.war"})
```

sbomgen.find_files_by_suffix (suffissi)

Restituisce i file il cui percorso completo (forward-slash-normalized) termina con uno dei suffissi specificati. Ad esempio, l'abbinamento avviene al di fuori della macchina virtuale Lua. `find_files_by_name`

- Parametri: `suffixes` — tabella di stringhe (suffissi di percorso da abbinare)
- Restituisce: `{string, ...}` — percorsi di file corrispondenti (nessuna tupla di errore)

```
local pom_files = sbomgen.find_files_by_suffix({"/pom.properties"})
```

```
local release_headers = sbomgen.find_files_by_suffix({"ap_release.h", "opensslv.h"})
```

sbomgen.find_files_by_path_regex (modelli)

Restituisce file il cui percorso normalizzato con barra forward-slash corrisponde a uno qualsiasi dei modelli regex Go (RE2) specificati. La corrispondenza avviene all'esterno della macchina virtuale Lua, il che la rende efficiente su elenchi di file di grandi dimensioni.

- Parametri: `patterns` — tabella delle stringhe regex Go
- Restituisce: `{string, ...}` — percorsi di file corrispondenti (nessuna tupla di errore)
- Genera: un errore Lua se qualche pattern non riesce a essere compilato

```
local configs = sbomgen.find_files_by_path_regex({"etc/.*\\.conf$", "opt/.*config\\  
\\.json$"})
```

Prestazioni: find_files_by_* vs get_file_list

Per i plugin di scoperta, preferisci o esegui un'iterazione eccessiva in Lua. `find_files_by_name` `find_files_by_suffix` `find_files_by_path_regex` `get_file_list()` In una scansione localhost con 300.000 file, l'iterazione dell'elenco dei file in Lua `string.match()` richiede ~ 15 secondi, mentre viene completata in meno di 1 millisecondo. `find_files_by_name` La differenza è che `get_file_list()` copia ogni percorso di file nella macchina virtuale Lua come una stringa, quindi Lua interpreta il ciclo e il pattern match per ognuno di essi. Le `find_files_by_*` funzioni eseguono la corrispondenza all'esterno della macchina virtuale Lua e restituiscono solo i percorsi corrispondenti, evitando sia il sovraccarico di copia che quello di interpretazione per percorso.

Utilizzare `get_file_list()` solo quando è necessaria una logica di corrispondenza personalizzata che non può essere espressa come nome di base, suffisso o corrispondenza regex.

sbomgen.read_dir (percorso)

Elenca le voci in una directory.

- Restituisce: `{{name, is_dir}, ...}`, `err`

```
local entries, err = sbomgen.read_dir("/app/node_modules")
```

```
if err then return end
for _, e in ipairs(entries) do
  if e.is_dir then
    sbomgen.log_debug("directory: " .. e.name)
  end
end
end
```

sbomgen.file_stat (percorso)

Restituisce i metadati relativi a un file.

- Restituisce: {is_regular, is_dir, size}, err

```
local info, err = sbomgen.file_stat(path)
if err then return end
if info.is_regular and info.size > 0 then
  -- process file
end
```

sbomgen.read_zip_entry (percorso, percorso_ingresso)

Legge una singola voce da un archivio ZIP, JAR o WAR.

- Restituisce: string, err

```
local manifest, err = sbomgen.read_zip_entry(
  "/app/lib/example.jar",
  "META-INF/MANIFEST.MF"
)
```

sbomgen.search_binary (percorso, espressione regolare)

Analizza un file come ELF, PE o Mach-O binario e cerca nella sezione predefinita una corrispondenza regex Go. constant/variable

- Restituisce: string|nil, err — la stringa corrispondente o nil se nessuna corrispondenza

```
local version, err = sbomgen.search_binary(path, "Version:\\s+([\\d.]+)")
```

```
if version then
  sbomgen.log_info("found version: " .. version)
end
```

sbomgen.search_binary_all (percorso, espressione regolare [, n])

Analizza un file come ELF, PE o Mach-O binario e restituisce tutte le corrispondenze univoche del primo gruppo di acquisizione dalla sezione predefinita. constant/variable Passa n per limitare i risultati.

- Restituisce: {string, ...}|nil, err — tabella delle stringhe corrispondenti o nil se nessuna corrispondenza

```
local versions, err = sbomgen.search_binary_all(path, "version[= ]+([\d.]+)", 5)
if versions then
  for _, v in ipairs(versions) do
    sbomgen.log_info("found: " .. v)
  end
end
```

sbomgen.search_binary_raw (percorso, espressione regolare)

Cerca nell'intero file binario la prima corrispondenza regex, non limitata a una sezione specifica. Viene utilizzato quando la ricerca basata sulle sezioni (search_binary) è insufficiente, ad esempio quando le stringhe di versione si trovano in sezioni non standard.

- Restituisce: string|nil, err — la stringa corrispondente o nil se nessuna corrispondenza

```
local version, err = sbomgen.search_binary_raw(path, "ProductVersion[\\x00\\s]+([\d.]+)")
```

FileHandle Metodi

FileHandle gli oggetti vengono restituiti da `sbomgen.open_file()`.

fh: read_line ()

Legge la riga successiva (senza il carattere di nuova riga). Ritorna a nil EOF.

- Restituisce: `string|nil, err`

fh: read (n)

Legge fino a byte. n Restituisce `nil` in EOF.

- Restituisce: `string|nil, err`

fh: close ()

Chiude l'handle del file. Al termine, chiudi sempre le maniglie.

Utilità binarie

sbomgen.hash (dati, algoritmo)

Restituisce il digest con codifica esadecimale di una stringa di byte in memoria secondo l'algoritmo specificato. Associalo `sbomgen.read_file(path)` a per eseguire l'hash di un file che hai già letto per l'analisi, preferibilmente a `sbomgen.hash_file(path)` quando hai bisogno sia dei byte che del digest, poiché non rilegge il file. `hash`

- Restituisce: `string, err`
- Algoritmi: vedi [the section called "Hash dei componenti"](#) l'elenco delle costanti di algoritmo accettate.

```
local data, err = sbomgen.read_file("/path/to/manifest.json")
if data then
    local sha256 = sbomgen.hash(data, sbomgen.hash_algorithms.SHA256)
    sbomgen.log_info("SHA-256: " .. sha256)
end
```

sbomgen.hash_file (percorso, algoritmo)

Restituisce il digest con codifica esadecimale del contenuto di un file secondo l'algoritmo specificato. Instrada la lettura attraverso il I/O livello degli artefatti, in modo che funzioni in modo uniforme tra directory, container, archivio, volume e artefatti localhost. Usalo quando il digest è l'unica cosa di cui hai bisogno dal file.

- Restituisce: `string, err`

```
local sha256, err = sbomgen.hash_file("/app/bin/server",
  sbomgen.hash_algorithms.SHA256)
if sha256 then
  sbomgen.log_info("SHA-256: " .. sha256)
end
```

sbomgen.sha256 (percorso)

⚠ Important

Obsoleta. Usare invece `sbomgen.hash_file(path, sbomgen.hash_algorithms.SHA256)`. Questo alias viene mantenuto per motivi di compatibilità con le versioni precedenti e continuerà a funzionare, ma verrà rimosso in una versione futura. I nuovi plugin dovrebbero chiamare in `hash_file` modo che la scelta dell'algorithmo sia esplicita.

Equivalente a `sbomgen.hash_file(path, "SHA-256")`.

- Restituisce: `string, err`

```
local hash, err = sbomgen.sha256("/app/bin/server")
if hash then
  sbomgen.log_info("SHA-256: " .. hash)
end
```

sbomgen.contains_bytes (percorso, modelli)

Verifica se un file contiene ciascuno dei modelli di byte specificati. Restituisce una tabella di booleani nello stesso ordine dei pattern di input.

- Restituisce: `{bool, ...}, err`

```
local results, err = sbomgen.contains_bytes(path, {
  "\\xff Go buildinf:", -- Go build identifier
```

```

    "/rustc/",          -- Rust build identifier
  })
  if results then
    local is_go = results[1]
    local is_rust = results[2]
  end
end

```

sbomgen.get_pe_version_info (percorso)

Analizza le risorse della versione di Windows PE da un file binario. Restituisce una tabella con i campi della versione o nil, err se il file non è un file binario PE o non ha una risorsa di versione.

- Restituisce: {product_version, file_version, string_table}, err

I file_version campi product_version and provengono dalla FixedFileInfo struttura PE, formattata come "major.minor.build.revision". Il string_table campo è una tabella nidificata con codice locale (ad esempio, "040904B0" per Unicode in inglese americano). Ogni locale è mappato su una tabella di name/value coppie tratte dal PE StringFileInfo (ProductVersion,, ProductNameFileDescription, ecc.). Un binario PE può esporre una o più impostazioni locali.

```

local info, err = sbomgen.get_pe_version_info(file_path)
if err then return end

-- Fixed version fields (always flat)
local product_ver = info.product_version -- e.g. "25.1.0.0"
local file_ver    = info.file_version    -- e.g. "25.1.0.0"

-- String table – iterate locales, or address a known locale by key
for locale, fields in pairs(info.string_table or {}) do
  sbomgen.log_info(string.format("%s ProductName=%s", locale, fields.ProductName or
  ""))
end

-- US English Unicode is the most common locale for PE files
local us = (info.string_table or {})[ "040904B0" ]
if us then
  local display_ver = us.ProductVersion -- e.g. "25.01"
  local name        = us.ProductName    -- e.g. "7-Zip"
end
end

```

sbomgen.parse_product_version (percorso)

Convenience wrapper che restituisce solo la stringa della versione del prodotto da un file binario PE. FixedFileInfo Equivalente a chiamare `get_pe_version_info(path)` e leggere. `product_version`

- Restituisce: `string`, `err`

```
local version, err = sbomgen.parse_product_version(file_path)
if version then
    sbomgen.log_info("product version: " .. version)
end
```

sbomgen.parse_file_version (percorso)

Convenience wrapper che restituisce solo la stringa della versione del file da un file binario PE. FixedFileInfo Equivalente a chiamare `get_pe_version_info(path)` e leggere. `file_version`

- Restituisce: `string`, `err`

```
local version, err = sbomgen.parse_file_version(file_path)
if version then
    sbomgen.log_info("file version: " .. version)
end
```

Output del pacchetto

sbomgen.push_package (pkg)

Inserisce la ricerca di un pacchetto nello SBOM. Disponibile solo nei plugin di raccolta.

La pkg tabella supporta i seguenti campi:

Campo	Tipo	Campo obbligatorio	Descrizione
name	stringa	Sì	Nome pacchetto

Campo	Tipo	Campo obbligatorio	Descrizione
<code>version</code>	stringa	No	Stringa di versione risolta
<code>namespace</code>	stringa	No	Namespace PURL (ad es.,) <code>"curl"</code> <code>"wordpress/plugin"</code>
<code>purl_type</code>	stringa	Sì	Tipo PURL (ad es.,, <code>"pypi"</code> , <code>"npm"</code>) <code>"cargo"</code> <code>"deb"</code> <code>"generic"</code>
<code>component_type</code>	stringa	Sì	Tipo di component e CyclonedX; usa <code>sbomgen.c component_types.*</code> costanti (ad es.) <code>sbomgen.c component_types.LIBRARY</code>
<code>qualifiers</code>	table	No	qualificatori PURL come coppie chiave-valore (appaiono nell'URL del pacchetto)
<code>properties</code>	table	No	Proprietà dei componenti CyclonedX come coppie chiave-valore (vedi) the section called "Proprietà CyclonedX"

Campo	Tipo	Campo obbligatorio	Descrizione
hashes	table	No	Hash dei component i codificati in base al nome dell'algorithm; vedi the section called "Hash dei component i"
children	table	No	Pacchetti secondari annidati, ciascuno con la stessa forma di pkg (i campi obbligatori vengono convalidati in modo ricorsivo)

```
sbomgen.push_package({
  name = "requests",
  version = "2.28.1",
  purl_type = "pypi",
  component_type = sbomgen.component_types.LIBRARY,
  qualifiers = { example_qualifier = "example_qualifier_value" },
  properties = {
    -- Use your own namespace; amazon:inspector:* is reserved for Amazon Inspector.
    ["acme:example:extra_field"] = "example_value",
  },
  hashes = {
    [sbomgen.hash_algorithms.SHA256] =
    "2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824",
  },
})
```

Hash dei componenti

Il hashes campo opzionale sull'integrità sbomgen.push_package() dei record digest per un componente. Le voci sono codificate in base al nome dell'algorithm e serializzate nell'components[].hashesarray CyclonedX, in base allo schema previsto da Amazon Inspector.

Algoritmi supportati

Usa le costanti sotto in `sbomgen.hash_algorithms` modo che il nome dell'algoritmo passato a/sia la stessa stringa accettata da `sbomgen.hash()`: `sbomgen.hash_file()`
`push_package({ hashes = ... })`

Costante	Valore	Lunghezza del digest esadecimale
<code>sbomgen.hash_algorithms.SHA1</code>	"SHA-1"	40
<code>sbomgen.hash_algorithms.SHA256</code>	"SHA-256"	64

Regole di convalida

`push_package()` convalida `hashes` prima di emettere un risultato. Un pacchetto i cui hash non superano la convalida viene eliminato e viene registrato un avviso. Convalida:

- I nomi degli algoritmi devono corrispondere a una voce in `sbomgen.hash_algorithms` (con distinzione tra maiuscole e minuscole, esattamente "SHA-1" o "SHA-256").
- I valori devono essere cifre esadecimali minuscole non vuote.
- I valori devono avere la lunghezza corretta per l'algoritmo (40 caratteri per, 64 caratteri per). SHA-1
SHA-256
- La convalida è ricorsiva: un hash non valido all'interno rifiuta l'intero pacchetto.
`children[].hashes`

Esempio: eseguire l'hashing di un manifesto e allegare il digest

```
function collect(file_path)
  local data, err = sbomgen.read_file(file_path)
  if err then return end

  local sha256 = sbomgen.hash(data, sbomgen.hash_algorithms.SHA256)

  sbomgen.push_package({
```

```
    name = "skill-manifest",
    version = "1.0.0",
    purl_type = "generic",
    component_type = sbomgen.component_types.DATA,
    hashes = {
      [sbomgen.hash_algorithms.SHA256] = sha256,
    },
  })
end
```

Se il digest è l'unica cosa di cui avete bisogno dal file, preferite `sbomgen.hash_file(path, algo)` leggerlo due volte, perché indirizza la lettura attraverso il livello degli artefatti I/O in un unico passaggio.

Proprietà CyclonedX

Le proprietà CyclonedX sono metadati chiave-valore collegati a un componente nello SBOM. Sono distinti dai qualificatori PURL:

- **qualifiers**— qualificatori PURL. Questi diventano parte della stringa dell'URL del pacchetto (ad es.). `pkg:deb/debian/curl@7.88.1?arch=amd64` Alcuni qualificatori PURL hanno un significato semantico per Amazon Inspector e influenzano l'identificazione delle vulnerabilità. [Vedi Cos'è l'URL di un pacchetto?](#) per le convenzioni per tipo di Inspector.
- **properties**— proprietà del componente CyclonedX. Queste appaiono nell'`components[].propertiesarray` di SBOM e non modificano il modo in cui il componente viene identificato.

Spazi dei nomi riservati

La `amazon:inspector:*` famiglia di namespace di proprietà CyclonedX è riservata ad Amazon Inspector:

- `amazon:inspector:sbom_generator:*`— utilizzato da `sbomgen` e dai suoi scanner integrati.
- `amazon:inspector:sbom_scanner:*`— utilizzato dall'API Amazon Inspector Scan.

Plugin-defined le proprietà non devono utilizzare questi namespace. La scrittura in uno spazio dei nomi riservato può creare un'ombra o entrare in conflitto con i valori su cui si basa Inspector e lo SBOM risultante potrebbe essere interpretato in modo errato durante l'identificazione delle

vulnerabilità. Vedi [Utilizzo degli spazi dei nomi CyclonedX con Amazon Inspector](#) per l'elenco completo delle chiavi riservate.

Regole di denominazione delle chiavi

Le chiavi di proprietà passate a `sbomgen.push_package()` vengono elaborate come segue:

Chiave di input	Chiave risultante in SBOM	Consigliato per plugin personalizzati?
Contiene : (ad es.) <code>acme:my_plugin:field</code>	Usato alla lettera	Sì, posiziona ogni proprietà definita dal plug-in nel tuo spazio dei nomi
No : (ad esempio,) <code>field</code>	Auto-prefixed a <code>amazon:inspector:sbom_generator:field</code>	No: viene scritto in un namespace riservato

Includi sempre almeno un punto nelle chiavi di proprietà che definisci. Usa uno spazio dei nomi unico per la tua organizzazione o il tuo plugin (ad esempio `acme:python-pip:*`):

```
properties = {
    -- Custom namespace – safe to use (recommended)
    ["acme:python-pip:manifest_path"] = file_path,
    ["acme:python-pip:pinned"]       = "true",

    -- Fully-qualified key outside amazon:inspector:* – also fine
    ["my:custom:namespace:key"] = "value",

    -- No colon: avoid – ends up as "amazon:inspector:sbom_generator:custom_field"
    -- custom_field = "value",
}
```

Proprietà impostate da sbomgen

Sbomgen può assegnare proprietà proprie a ogni componente che emette. Questi valori provengono dallo spazio dei nomi `amazon:inspector:sbom_generator:*` nomi riservato e non dovrebbero essere prodotti dai plugin. Comportamento di runtime osservato:

- `source_path` viene sempre aggiunto da sbomgen.

- `source_file_scannere` `source_package_collector` vengono aggiunti quando `--enable-debug-props` è abilitato.

La tassonomia completa delle chiavi riservate è gestita nella guida per l'utente di Amazon Inspector: [Utilizzo degli spazi dei nomi CyclonedX con Amazon Inspector](#).

Costanti di proprietà

Built-in le costanti delle chiavi di proprietà sono disponibili tramite `sbomgen.properties`

Ogni costante seguente si risolve in una chiave all'interno dello spazio dei nomi riservato.

`amazon:inspector:sbom_generator:*` Queste costanti esistono in modo che gli scanner integrati di `sbomgen` emettano chiavi di proprietà coerenti. Non sono punti di estensione per plug-in personalizzati: utilizzandoli in un plug-in personalizzato si scrive in uno spazio dei nomi riservato, su cui può fare affidamento i valori shadow di Inspector. Vedi sopra. [the section called “Spazi dei nomi riservati”](#)

Gli autori di plugin personalizzati dovrebbero definire le proprietà nel proprio spazio dei nomi (ad esempio `acme:my_plugin:*`) anziché riutilizzare queste costanti.

Costante	Valore risolto
<code>sbomgen.properties.NAMESPACE</code>	<code>amazon:inspector:sbom_generator:</code>
<code>sbomgen.properties.VENDOR</code>	<code>amazon:inspector:sbom_generator:vendor</code>
<code>sbomgen.properties.FILE_SIZE_BYTES</code>	<code>amazon:inspector:sbom_generator:file_size_bytes</code>
<code>sbomgen.properties.KERNEL_COMPONENT</code>	<code>amazon:inspector:sbom_generator:kernel_component</code>
<code>sbomgen.properties.RUNNING_KERNEL</code>	<code>amazon:inspector:sbom_generator:running_kernel</code>
<code>sbomgen.properties.UNRESOLVED_VERSION</code>	<code>amazon:inspector:sbom_generator:unresolved_version</code>

Costante	Valore risolto
<code>sbomgen.properties.TRANSITIVE_DEPENDENCY</code>	<code>amazon:inspector:sbom_generator:experimental:transitive_dependency</code>
<code>sbomgen.properties.GO_REPLACE_DIRECTIVE</code>	<code>amazon:inspector:sbom_generator:replaced_by</code>
<code>sbomgen.properties.DUPLICATE_PACKAGE</code>	<code>amazon:inspector:sbom_generator:is_duplicate_package</code>
<code>sbomgen.properties.DUPLICATE_PURL</code>	<code>amazon:inspector:sbom_generator:duplicate_purl</code>
<code>sbomgen.properties.DOCKERFILE_CHECK</code>	<code>amazon:inspector:sbom_generator:dockerfile_finding</code>
<code>sbomgen.properties.CERTIFICATE_FINDING</code>	<code>amazon:inspector:sbom_generator:certificate_finding</code>
<code>sbomgen.properties.CERTIFICATE_SUBJECT_NAME</code>	<code>amazon:inspector:sbom_generator:certificate:subject_name</code>
<code>sbomgen.properties.CERTIFICATE_ISSUER_NAME</code>	<code>amazon:inspector:sbom_generator:certificate:issuer_name</code>
<code>sbomgen.properties.CERTIFICATE_SIGNATURE_ALGORITHM</code>	<code>amazon:inspector:sbom_generator:certificate:signature_algorithm</code>
<code>sbomgen.properties.CERTIFICATE_NOT_VALID_BEFORE</code>	<code>amazon:inspector:sbom_generator:certificate:not_valid_before</code>
<code>sbomgen.properties.CERTIFICATE_NOT_VALID_AFTER</code>	<code>amazon:inspector:sbom_generator:certificate:not_valid_after</code>

Costante	Valore risolto
<code>sbomgen.properties.WINDOWS_REGISTRY_KEY</code>	<code>amazon:inspector:sbom_generator:registry_key</code>
<code>sbomgen.properties.SUBSCRIPTION_ENABLED</code>	<code>amazon:inspector:sbom_generator:subscription:enabled</code>
<code>sbomgen.properties.SUBSCRIPTION_NAME</code>	<code>amazon:inspector:sbom_generator:subscription:name</code>
<code>sbomgen.properties.SUBSCRIPTION_LOCKED_VERSION</code>	<code>amazon:inspector:sbom_generator:subscription:locked_version</code>
<code>sbomgen.properties.OPENSSSL_FULL_VERSION</code>	<code>amazon:inspector:sbom_generator:openssl:full_version</code>
<code>sbomgen.properties.HARDENED_IMAGE_VENDOR</code>	<code>amazon:inspector:sbom_generator:hardened_image:vendor</code>

Gruppi di scanner

I plugin Discovery devono dichiarare i propri gruppi di scanner tramite `get_scanner_groups()`. I gruppi classificano gli scanner e consentono agli utenti di abilitare o disabilitare selettivamente le categorie. Le costanti sono disponibili tramite: `sbomgen.groups`

Costante	Valore	Descrizione
<code>sbomgen.groups.OS</code>	<code>"os"</code>	gestori di pacchetti del sistema operativo (dpkg, rpm, ecc.)
<code>sbomgen.groups.PROGRAMMING_LANGUAGE</code>	<code>"programming-language-packages"</code>	Gestori di pacchetti linguistici (pip, npm, maven, ecc.)
<code>sbomgen.groups.BINARY</code>	<code>"binary"</code>	Analisi binaria compilata (Go, Rust)

Costante	Valore	Descrizione
<code>sbomgen.groups.PACKAGE_COLLECTOR</code>	"pkg-scanner"	Raccolta generale di pacchetti
<code>sbomgen.groups.EXTRA_ECOSYSTEMS</code>	"extra-ecosystems"	Ecosistemi aggiuntivi (curl, nginx, ecc.)
<code>sbomgen.groups.CERTIFICATE</code>	"certificate"	Scansione dei certificati
<code>sbomgen.groups.CUSTOM</code>	"custom"	Aggiunto automaticamente a tutti i plugin personalizzati caricati tramite <code>--plugin-dir</code>
<code>sbomgen.groups.MACHINE_LEARNING</code>	"machine-learning"	Rilevamento di modelli di apprendimento automatico

Esempio:

```
function get_scanner_groups()
  return {sbomgen.groups.PROGRAMMING_LANGUAGE, sbomgen.groups.PACKAGE_COLLECTOR}
end
```

Costanti del tipo di componente

Il `component_type` campo in `push_package()` deve essere uno dei tipi di componenti CyclonedX 1.5. Le costanti sono disponibili tramite: `sbomgen.component_types`

Costante	Valore
<code>sbomgen.component_types.APPLICATION</code>	"application"
<code>sbomgen.component_types.FRAMEWORK</code>	"framework"

Costante	Valore
<code>sbomgen.component_types.LIBRARY</code>	<code>"library"</code>
<code>sbomgen.component_types.CONTAINER</code>	<code>"container"</code>
<code>sbomgen.component_types.PLATFORM</code>	<code>"platform"</code>
<code>sbomgen.component_types.OPERATING_SYSTEM</code>	<code>"operating-system"</code>
<code>sbomgen.component_types.DEVICE</code>	<code>"device"</code>
<code>sbomgen.component_types.DEVICE_DRIVER</code>	<code>"device-driver"</code>
<code>sbomgen.component_types.FIRMWARE</code>	<code>"firmware"</code>
<code>sbomgen.component_types.FILE</code>	<code>"file"</code>
<code>sbomgen.component_types.MACHINE_LEARNING_MODEL</code>	<code>"machine-learning-model"</code>
<code>sbomgen.component_types.DATA</code>	<code>"data"</code>

Esempio:

```
sbomgen.push_package({
    name = "requests",
    version = "2.28.1",
    purl_type = "pypi",
    component_type = sbomgen.component_types.LIBRARY,
})
```

Costanti dell'algoritmo Hash

Costanti per il parametro dell'algoritmo di `sbomgen.hash()` `sbomgen.hash_file()`, e il `hashes` campo di `sbomgen.push_package()` I valori delle stringhe corrispondono ai nomi degli algoritmi hash CyclonedX, quindi la stessa costante attraversa l'intero percorso di hashing senza traduzione.

Costante	Valore
<code>sbomgen.hash_algorithms.SHA1</code>	"SHA-1"
<code>sbomgen.hash_algorithms.SHA256</code>	"SHA-256"

Esempio:

```
local digest = sbomgen.hash_file(path, sbomgen.hash_algorithms.SHA256)
sbomgen.push_package({
  name = "example",
  purl_type = "generic",
  component_type = sbomgen.component_types.LIBRARY,
  hashes = { [sbomgen.hash_algorithms.SHA256] = digest },
})
```

Costanti della piattaforma

Costanti con cui confrontare. `sbomgen.get_platform()` Disponibile tramite `sbomgen.platform`:

Costante	Valore
<code>sbomgen.platform.LINUX</code>	"linux"
<code>sbomgen.platform.WINDOWS</code>	"windows"
<code>sbomgen.platform.DARWIN</code>	"darwin"

Esempio:

```
if sbomgen.get_platform() == sbomgen.platform.WINDOWS then
  -- Windows-specific logic
end
```

Informazioni sull'Artifact

sbomgen.get_platform ()

Restituisce la stringa della piattaforma di runtime (ad es.,). "linux" "windows" "darwin"

sbomgen.get_artifact_type ()

Restituisce il tipo di artefatto da scansionare (ad esempio). "directory" "archive"

sbomgen.should_collect_licenses ()

Restituisce se l'utente ha abilitato la raccolta delle licenze tramite. true --collect-licenses

sbomgen.get_env_vars ()

Restituisce le variabili di ambiente dall'artefatto come tabella di voci. {key, value}

```
local env_vars = sbomgen.get_env_vars()
for _, env in ipairs(env_vars) do
    if env.key == "NODE_ENV" then
        sbomgen.log_info("Node environment: " .. env.value)
    end
end
```

sbomgen.get_system_drive ()

Restituisce la lettera dell'unità di sistema (ad esempio) dall'ambiente dell'artefatto. "C:" Legge la variabile di SystemDrive ambiente, impostata come predefinita se non è impostata. "C:" Questo è l'equivalente Lua di `strutils.GetSystemDriverLetter()`

```
local drive = sbomgen.get_system_drive()
local program_files = drive .. "/Program Files/"
```

sbomgen.resolve_glob_paths (modelli)

Espande i pattern glob del filesystem rispetto al filesystem host. Localhost-only: restituisce nil più un errore su altri tipi di artefatti.

```
function get_localhost_scan_paths()
```

```
return sbongen.resolve_glob_paths({
    "/home/*/.cache/huggingface/hub",
    "/Users/*/.cache/huggingface/hub",
    "C:/Users/*/.cache/huggingface/hub",
})
end
```

Comportamento:

- La sintassi del pattern segue quella di Go [filepath.Match](#):*,?,[abc],[a-z].
- I modelli di input e i percorsi di output sono normalizzati: i separatori ridondanti (a//b), i segmenti di punti (.) e i separatori finali (a/. /b) vengono compressi. a/b/
- L'output viene deduplicato; vince la prima occorrenza di un percorso. L'ordine del modello di input viene mantenuto in tutto il risultato.
- I pattern che non corrispondono a nulla non restituiscono alcun elemento. Empty-string gli schemi vengono ignorati silenziosamente. I pattern malformati (ad esempio parentesi non corrispondenti) emettono un avviso e vengono ignorati.

Cross-platform separatori di percorso:

- Usa le barre (/) per tutti i percorsi. Le barre in avanti funzionano su Linux, macOS e Windows; la logica del percorso dei file di Go le traduce nel separatore nativo di Windows.
- I separatori backslash funzionano solo su Windows. Su Linux e macOS, \ è un carattere letterale del nome di file, non un separatore di percorso, uno schema che non corrisponde a nulla sui sistemi POSIX. "C:\\Users*"
- Evita i percorsi letterali nelle stringhe Lua. Windows-style Una stringa Lua like "C:\Users" viene interpretata come «C:<form-feed>ersbecause» non \U è un escape Lua valido (e\f, \t ecc. lo sono)\n, quindi il pattern fallisce silenziosamente. Utilizzate le barre in avanti, le barre rovesciate con escape () o una stringa non elaborata tra parentesi lunghe ("C:\\Users"). [[C:\Users]]

Informazioni di sistema

Queste funzioni restituiscono metadati sul sistema operativo e sull'hardware dell'artefatto. I valori possono essere stringhe vuote se le informazioni non sono disponibili (ad esempio, durante la scansione di una directory senza metadati del sistema operativo).

Funzione	Valori restituiti
<code>sbomgen.get_os_name()</code>	Nome del sistema operativo (ad es., "Ubuntu") "Alpine Linux"
<code>sbomgen.get_os_version()</code>	Versione del sistema operativo (ad es. "22.04", "3.18")
<code>sbomgen.get_os_codename()</code>	Nome in codice del sistema operativo (ad es., "jammy") "bookworm"
<code>sbomgen.get_os_id()</code>	Identificatore del sistema operativo (ad es.,) "ubuntu" "alpine"
<code>sbomgen.get_kernel_name()</code>	Nome del kernel (ad esempio,) "Linux"
<code>sbomgen.get_kernel_version()</code>	Stringa della versione del kernel
<code>sbomgen.get_cpu_arch()</code>	Architettura della CPU (ad es. "x86_64", "aarch64")
<code>sbomgen.get_hostname()</code>	Nome host del sistema

Espressioni regolari

I pattern integrati di Lua mancano di funzionalità come alternation (`|`), quantifier ranges (`{n,}`) e lookahead. Per colmare questa lacuna, sbomgen espone direttamente il pacchetto di Go. `regexp`. Queste funzioni usano la sintassi Go regex (RE2), non i pattern Lua.

`sbomgen.regex_find (str, pattern)`

Restituisce la prima corrispondenza di un pattern regex Go, o se non corrisponde. `nil`

- Restituisce: `string|nil, err`

```
local version = sbomgen.regex_find(content, "\\d+\\.\\d+\\.\\d+")
```

sbomgen.regex_match (str, modello)

Restituisce i gruppi di acquisizione della prima partita. L'indice 1 è la partita completa, 2+ sono i gruppi di acquisizione.

- Restituisce: {string, ...}|nil, err

```
local groups = sbomgen.regex_match(content, "(MySQL|MariaDB) (\\d+)\\.\\.\\. (\\d+)\\.\\.\\. (\\d+)")
if groups then
    local db_type = groups[2]    -- "MySQL" or "MariaDB"
    local major   = groups[3]
end
```

sbomgen.regex_find_all (str, modello [, n])

Restituisce tutte le corrispondenze non sovrapposte. Passa n ai risultati limite (impostazione predefinita: tutti).

- Restituisce: {string, ...}|nil, err

```
local versions = sbomgen.regex_find_all(content, "\\d+\\.\\.\\.\\d+\\.\\.\\.\\d+")
```

sbomgen.regex_replace (str, pattern, replacement)

Sostituisce tutte le partite. La stringa sostitutiva può utilizzare \$1\$2, ecc. per acquisire riferimenti al gruppo.

- Restituisce: string, err

```
local cleaned = sbomgen.regex_replace(raw_version, "(1[6-9]\\d{8,}|
buildkitsandbox.*)$", "")
```

Quando usare i pattern regex vs Lua

Usa `string.match/()` integrato in Lua `string.find` per modelli semplici: sono più veloci e non richiedono l'evasione delle barre rovesciate. `sbomgen.regex_*` Usalo quando ne hai bisogno:

- Alternanza: (foo|bar)

- Intervalli di quantificazione: `\d{8,}`
- Classi di caratteri complesse non esprimibili nei modelli Lua

Analisi strutturata

Sbomgen espone aiutanti leggeri per la decodifica di formati di testo strutturati direttamente nelle tabelle Lua.

sbomgen.json_decode (str)

Analizza una stringa JSON in una tabella Lua.

- Restituisce: `table|nil, err`

```
local doc, err = sbomgen.json_decode('{ "name": "requests", "version": "2.28.1" }')
if err then return end
sbomgen.log_info(doc.name)
```

sbomgen.xml_decode (str)

Analizza una stringa XML in una tabella Lua.

- Restituisce: `table|nil, err`

I valori XML utilizzano la forma seguente:

- `_name`— nome dell'elemento
- `_attr`— tabella degli attributi, se presente
- `_text`— contenuto testuale tagliato, se presente
- indici numerici — elementi 1..n secondari

```
local doc, err = sbomgen.xml_decode('<package id="Newtonsoft.Json" version="13.0.3" />')
if err then return end
sbomgen.log_info(doc._attr.id)
```

Registro di sistema di Windows

Queste funzioni forniscono l'accesso in sola lettura al registro di Windows. Sugli artefatti non Windows, restituisce un errore. `registry_open_key` L'accessor del registro viene inizializzato pigramente al primo utilizzo e supporta sia l'accesso live alle API di Windows (scansioni localhost su Windows) che l'analisi degli hive REGF basata su file (scansioni). `container/volume`

`sbomgen.registry_open_key` (percorso)

Apri una chiave di registro. Restituisce una maniglia chiave che deve essere chiusa con `registry_close`.

- Restituisce: `key`, `err`

```
local key, err = sbomgen.registry_open_key("SOFTWARE\\Microsoft\\Windows\\
\\CurrentVersion\\Uninstall\\7-Zip")
if err then return end
-- use key...
sbomgen.registry_close(key)
```

`sbomgen.registry_get_string` (chiave, nome_valore)

Legge un valore di stringa da una chiave di registro aperta.

- Restituisce: `string`, `err`

```
local version, err = sbomgen.registry_get_string(key, "DisplayVersion")
```

`sbomgen.registry_get_integer` (chiave, nome_valore)

Legge un valore intero da una chiave di registro aperta.

- Restituisce: `number`, `err`

`sbomgen.registry_get_strings` (chiave, nome_valore)

Legge un valore multistringa (`REG_MULTI_SZ`) da una chiave di registro aperta. Restituisce una tabella di stringhe.

- Restituisce: {string, ...}, err

```
local paths, err = sbomgen.registry_get_strings(key, "DependsOnService")
if paths then
  for _, p in ipairs(paths) do
    sbomgen.log_info("depends on: " .. p)
  end
end
```

sbomgen.registry_get_subkeys (chiave)

Restituisce tutti i nomi delle sottochiavi in una chiave di registro aperta.

- Restituisce: {string, ...}, err

```
local subkeys, err = sbomgen.registry_get_subkeys(key)
for _, name in ipairs(subkeys) do
  local subkey, err = sbomgen.registry_open_key(parent_path .. "\\\" .. name)
  -- ...
end
```

sbomgen.registry_close (chiave)

Chiude l'handle di una chiave di registro. Anche le maniglie dei tasti vengono chiuse automaticamente dal Garbage Collector, ma si consiglia una chiusura esplicita.

Registrazione dei log

I messaggi di log vengono scritti nell'output della console di sbomgen. Ogni messaggio emesso da un plugin viene automaticamente preceduto dall'etichetta sorgente e dall'ecosistema del plugin, ad esempio:

```
[custom:python-pip] Parsing requirements.txt
```

log_info, log_warn, e stampa log_error sempre. log_debug stampa solo quando sbomgen viene invocato con. --verbose

Funzione	Livello	Visibile per impostazione predefinita?
<code>sbomgen.log_debug(message)</code>	DEBUG	No, richiede <code>--verbose</code>
<code>sbomgen.log_info(message)</code>	INFO	Sì
<code>sbomgen.log_warn(message)</code>	WARN	Sì
<code>sbomgen.log_error(message)</code>	ERRORE	Sì

Usa `string.format` per messaggi formattati:

```
sbomgen.log_info(string.format("found %d packages in %s", count, file_path))
```

Funzioni di debug

sbomgen.breakpoint (messaggio)

Stampa su `message stderr` e blocca l'esecuzione finché l'utente non preme Invio. Se `message` viene omissso, stampa un messaggio predefinito.

Utilizzatelo come semplice debugger, posizionando i punti di interruzione nei punti chiave del plugin ed eseguitelo per vedere l'output del log circostante. `--verbose`

```
sbomgen.log_info("state: " .. some_variable)
sbomgen.breakpoint("paused after state dump – press Enter to continue")
```

API di test

Le funzioni nella `testing` tabella globale sono disponibili solo all'interno dei file di test del plugin (`*_test.lua`), caricati da `inspector-sbomgen plugin test`. Non sono disponibili in fase di esecuzione nei plugin di `discovery` o `collection`. L'`sbomgen.*` API completa è disponibile anche all'interno dei file di test, ma `sbomgen.*` le funzioni che richiedono un artefatto (ad

esempiosbomgen.read_file()) producono risultati significativi solo se richiamate dall'interno di una scansione. Per una guida narrativa, consulta il [the section called "Guida al test dei plugin"](#)

Funzioni di scansione

Ogni funzione di scansione crea un artefatto del tipo specificato, esegue la pipeline discovery→collection del plugin corrente e restituisce i risultati risultanti. L'pathargomento viene risolto in relazione alla directory del file di test.

Funzione	Tipo di artefatto
testing.scan_directory(path)	Directory
testing.scan_archive(path)	Directory (alias di) scan_directory
testing.scan_localhost(path)	Host locale
testing.scan_binary(path)	Binario
testing.scan_volume(path)	Volume
testing.scan_container(path)	Contenitore

Tutti e sei restituiscono una tabella dei risultati con la forma seguente.

Forma del risultato

Ogni tabella di ricerca proietta solo i campi elencati di seguito. In particolare, namespace e non purl_type vengono proiettati separatamente, ma sono incorporati nella purl stringa completa.

```
local result = testing.scan_directory("_testdata/example")
-- result.findings                -- array of finding tables
-- result.findings[i].name        -- string
-- result.findings[i].version     -- string
-- result.findings[i].component_type -- string
-- result.findings[i].purl        -- string (the full Package URL, or "" if
  none)
-- result.findings[i].properties  -- table<string, string>
-- result.findings[i].children    -- array of finding tables (same shape,
  recursive)
```

Asserzioni

Funzione	Firma	Descrizione
<code>testing.assert_equals</code>	<code>(expected: any, actual: any, message?: string)</code>	Fallisce se <code>toString(actual) != toString(expected)</code> .
<code>testing.assert_not_equals</code>	<code>(expected: any, actual: any, message?: string)</code>	Fallisce se <code>toString(actual) == toString(expected)</code> .
<code>testing.assert_true</code>	<code>(value: any, message?: string)</code>	Fallisce se <code>value</code> è <code>false</code> o <code>nil</code> .
<code>testing.assert_false</code>	<code>(value: any, message?: string)</code>	Fallisce se non lo <code>value</code> è <code>false</code> e non lo è <code>nil</code> .
<code>testing.assert_nil</code>	<code>(value: any, message?: string)</code>	Fallisce se non lo <code>value</code> è <code>nil</code> .
<code>testing.assert_not_nil</code>	<code>(value: any, message?: string)</code>	Fallisce se <code>value</code> lo è <code>nil</code> .
<code>testing.assert_contains</code>	<code>(haystack: string, needle: string, message?: string)</code>	Fallisce se <code>haystack</code> non contiene <code>needle</code> (corrispondenza tra le sottostringhe).
<code>testing.assert_matches</code>	<code>(str: string, pattern: string, message?: string)</code>	Fallisce se <code>str</code> non corrisponde all'espressione regolare Go (RE2) specificata.
<code>testing.assert_length</code>	<code>(tbl: table, expected: integer, message?: string)</code>	Fallisce se <code>#tbl</code> non è uguale a <code>expected</code> .

Flusso di controllo

Funzione	Firma	Descrizione
<code>testing.fail</code>	<code>(message: string)</code>	Fallisce immediatamente il test corrente con il messaggio fornito.
<code>testing.skip</code>	<code>(message: string)</code>	Salta il test corrente. Il risultato viene segnalato come ignorato, non fallito.

Scoperta del test

Qualsiasi funzione Lua globale il cui nome inizia con `test_` una corrispondenza tra file `*_test.lua` viene considerata un test. Il file di test deve trovarsi accanto `init.lua` a una `{phase}/`
`{platform}/{category}/{ecosystem}/` profondità normale. I dati del dispositivo vengono inseriti `_testdata/` accanto al file di test, in cui il runner non scende `_testdata/` durante la ricerca dei file di test.

Gestione errori

Le funzioni API che possono fallire restituiscono due valori: `value`, `err` In caso di successo, `err` è `nil`. In caso di errore, il primo valore è `nil` ed `err` è una stringa di errore.

```
local content, err = sbomgen.read_file(path)
if err then
    sbomgen.log_error("failed to read " .. path .. ": " .. err)
    return
end
-- content is safe to use here
```

Se un plugin genera un errore Lua non gestito, `sbomgen` registra un avviso e continua con il file o il plugin successivo. Gli altri plugin non sono interessati.

Integrazione delle scansioni di Amazon Inspector nella tua pipeline CI/CD

L' CI/CD integrazione con Amazon Inspector utilizza Amazon Inspector SBOM Generator e Amazon Inspector Scan API per produrre report di vulnerabilità per le immagini dei container. Amazon Inspector SBOM Generator crea una distinta base software (SBOM) per archivi, immagini di container, directory, sistemi locali, compilati e binari. Go Rust L'API Amazon Inspector Scan analizza lo SBOM per creare un report con dettagli sulle vulnerabilità rilevate. Puoi integrare le scansioni delle immagini dei container di Amazon Inspector con la tua CI/CD pipeline per individuare le vulnerabilità del software e produrre report sulle vulnerabilità, che consentono di indagare e correggere i rischi prima della distribuzione. Per configurare CI/CD l'integrazione, puoi utilizzare i plug-in o creare un' CI/CD integrazione personalizzata utilizzando Amazon Inspector SBOM Generator e l'API Amazon Inspector Scan.

Argomenti

- [Integrazione con i plugin](#)
- [Integrazione personalizzata](#)
- [Configurare un AWS account per utilizzare l'integrazione Amazon Inspector CI/CD](#)
- [Controlli dei file Dockerfile di Amazon Inspector](#)
- [Creazione di un'integrazione di CI/CD pipeline personalizzata con Amazon Inspector Scan](#)
- [Utilizzo di Amazon Inspector con GitHub actions](#)
- [Utilizzo di Amazon Inspector Jenkins plugin](#)
- [Utilizzo CodeCatalyst azioni con Amazon Inspector](#)
- [Utilizzo delle azioni di scansione di Amazon Inspector con CodePipeline](#)
- [Utilizzo di Amazon Inspector TeamCity plugin](#)
- [Utilizzo di Amazon Inspector con GitLab componenti](#)

Integrazione con i plugin

Amazon Inspector fornisce plug-in per le soluzioni supportate. CI/CD Puoi installare questi plugin dai rispettivi marketplace e poi utilizzarli per aggiungere Amazon Inspector Scans come fase di costruzione della tua pipeline. La fase di creazione del plug-in esegue il generatore Amazon Inspector SBOM sull'immagine fornita, quindi esegue l'API Amazon Inspector Scan sull' SBOM generato.

Di seguito è riportata una panoramica di come funziona un' CI/CD integrazione con Amazon Inspector tramite i plugin:

1. Si configura un Account AWS per consentire l'accesso all'API Amazon Inspector Scan. Per istruzioni, consulta [Configurare un AWS account per utilizzare l'integrazione Amazon Inspector CI/CD](#).
2. Installa il plug-in Amazon Inspector dal marketplace.
3. Installa e configura il binario Amazon Inspector SBOM Generator. Per istruzioni, consulta [Generatore SBOM Amazon Inspector](#).
4. Aggiungi Amazon Inspector Scans come fase di costruzione nella tua CI/CD pipeline e configuri la scansione.
5. Quando esegui una build, il plug-in prende l'immagine del contenitore come input e quindi esegue Amazon Inspector SBOM Generator sull'immagine per generare un SBOM compatibile. CycloneDX
6. Da lì, il plug-in invia la SBOM generata a un endpoint dell'API Amazon Inspector Scan che valuta ogni componente SBOM alla ricerca di vulnerabilità.
7. La risposta dell'API Amazon Inspector Scan viene trasformata in un report di vulnerabilità nei formati CSV, SBOM JSON e HTML. Il rapporto contiene dettagli su eventuali vulnerabilità rilevate da Amazon Inspector.

Soluzioni supportate CI/CD

Amazon Inspector attualmente supporta le seguenti CI/CD soluzioni. Per istruzioni complete sulla configurazione dell' CI/CD integrazione tramite un plug-in, seleziona il plug-in adatto alla tua CI/CD soluzione:

- [GitHub azioni](#)
- [Plugin Jenkins](#)
- [TeamCity Plugin](#)

Integrazione personalizzata

Se Amazon Inspector non fornisce plug-in per la tua CI/CD soluzione, puoi creare un' CI/CD integrazione personalizzata utilizzando una combinazione di Amazon Inspector SBOM Generator e

Amazon Inspector Scan API. Puoi anche utilizzare un'integrazione personalizzata per ottimizzare le scansioni utilizzando le opzioni disponibili tramite Amazon Inspector SBOM Generator.

Di seguito è riportata una panoramica di come funziona un' CI/CD integrazione personalizzata con Amazon Inspector:

1. Si configura un Account AWS per consentire l'accesso all'API Amazon Inspector Scan. Per istruzioni, consulta [Configurare un AWS account per utilizzare l'integrazione Amazon Inspector CI/CD](#).
2. Installa e configura il binario Amazon Inspector SBOM Generator. Per istruzioni, consulta [Generatore SBOM Amazon Inspector](#).
3. Utilizzi Amazon Inspector SBOM Generator per generare un SBOM CycloneDX compatibile per l'immagine del contenitore.
4. Utilizzi l'API Amazon Inspector Scan sulla SBOM generata per produrre un report di vulnerabilità.

Per istruzioni sulla configurazione di un'integrazione personalizzata, consulta. [Creazione di un'integrazione di CI/CD pipeline personalizzata con Amazon Inspector Scan](#)

Configurare un AWS account per utilizzare l'integrazione Amazon Inspector CI/CD

Per utilizzare l' CI/CD integrazione con Amazon Inspector, devi registrarti a un Account AWS. Account AWS Deve avere un ruolo IAM che consenta alla tua CI/CD pipeline di accedere all'API Amazon Inspector Scan. Completa le attività nei seguenti argomenti per registrarti Account AWS, creare un utente amministratore e configurare un ruolo IAM per l'integrazione. CI/CD

Note

Se ti sei già registrato a un Account AWS, puoi passare a [Configura un ruolo IAM per CI/CD l'integrazione](#).

Argomenti

- [Iscriviti a un Account AWS](#)
- [Configura un ruolo IAM per CI/CD l'integrazione](#)

Iscriviti a un Account AWS

Per iniziare AWS, hai bisogno di un Account AWS. Per informazioni sulla creazione di un Account AWS, vedi Guida [introduttiva a un Account AWS](#) nella Guida Gestione dell'account AWS di riferimento.

Configura un ruolo IAM per CI/CD l'integrazione

Per integrare la scansione di Amazon Inspector nella tua CI/CD pipeline devi creare una policy IAM che consenta l'accesso all'API Amazon Inspector Scan che analizza la distinta base del software (SBOM). Quindi, puoi collegare tale policy a un ruolo IAM che il tuo account può assumere per eseguire l'API Amazon Inspector Scan.

1. Accedi Console di gestione AWS e apri la console IAM all'indirizzo <https://console.aws.amazon.com/iam/>.
2. Nel pannello di navigazione della console IAM, seleziona Policies, quindi scegli Create Policy.
3. In Policy Editor seleziona JSON e incolla la seguente dichiarazione:

JSON

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

4. Scegli Next (Successivo).
5. Assegna un nome alla politica, ad esempio `InspectorCICDscan-policy`, e aggiungi una descrizione opzionale, quindi scegli Crea politica. Questa politica verrà allegata al ruolo che creerai nei passaggi successivi.
6. Nel riquadro di navigazione della console IAM, seleziona Ruoli, quindi seleziona Crea nuovo ruolo.

7. Per il tipo di entità affidabile, scegli Custom trust policy e incolla la seguente policy:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      },
      "Action": "sts:AssumeRole",
      "Condition": {}
    }
  ]
}
```

8. Scegli Next (Successivo).
9. In Aggiungi autorizzazioni cerca e seleziona la politica che hai creato in precedenza, quindi scegli Avanti.
10. Assegna un nome al ruolo, ad esempio `InspectorCICDscan-role`, e aggiungi una descrizione opzionale, quindi scegli `Create Role`.

Controlli dei file Dockerfile di Amazon Inspector

Questa sezione descrive come utilizzare Amazon Inspector SBOM Generator per scansionare immagini Dockerfiles e Docker container alla ricerca di configurazioni errate che introducono vulnerabilità di sicurezza.

Argomenti

- [Utilizzo dei controlli Dockerfile Sbogmen](#)
- [Controlli Dockerfile supportati](#)

Utilizzo dei controlli Dockerfile Sbogmen

I controlli Dockerfile vengono eseguiti automaticamente quando viene scoperto un file denominato `Dockerfile` o `e` quando `*.Dockerfile` viene scansionata un'immagine Docker.

È possibile disabilitare i controlli Dockerfile utilizzando l'argomento. `--skip-scanners dockerfile` Puoi anche combinare i controlli Dockerfile con qualsiasi scanner disponibile, come sistemi operativi o pacchetti di terze parti.

Esempi di comandi Docker check

I seguenti comandi di esempio mostrano come generare immagini SBOMs per Dockerfiles e container Docker, nonché per sistemi operativi e pacchetti di terze parti.

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

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

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

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

Componente di file di esempio

Di seguito è riportato un esempio di ricerca di Dockerfile per un componente di file.

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

Esempio di componente di risposta alla vulnerabilità

Di seguito è riportato un esempio di ricerca di Dockerfile per un componente di risposta alle vulnerabilità.

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

Note

Se si richiama Sbmngen senza il `--scan-sbom` flag, è possibile visualizzare solo i risultati non elaborati di Dockerfile.

Controlli Dockerfile supportati

SbmngenI controlli Dockerfile sono supportati per quanto segue:

- Il pacchetto binario Sudo
- Utilità Debian APT
- Segreti codificati
- Contenitori per radici
- Flag di comando che indeboliscono il runtime
- Variabili di ambiente che indeboliscono il runtime

Ciascuno di questi controlli Dockerfile ha un indice di gravità corrispondente, riportato all'inizio dei seguenti argomenti.

Note

Le raccomandazioni descritte nei seguenti argomenti si basano sulle migliori pratiche del settore.

Il pacchetto binario Sudo

Note

Il grado di gravità di questo controllo è Info.

Si consiglia di non installare o utilizzare il pacchetto binario Sudo perché ha un comportamento TTY e di inoltro del segnale imprevedibile. [Per ulteriori informazioni, consulta User nel sito Web Docker Docs.](#) [Se il tuo caso d'uso richiede funzionalità simili al pacchetto binario Sudo, ti consigliamo di utilizzare Gosu.](#)

DebianUtilità APT

Note

Il grado di severità di questo controllo è Alto.

Di seguito sono riportate le migliori pratiche per l'utilizzo delle utilità Debian APT.

Combinare **apt-get** i comandi in un'unica **Run** istruzione per evitare problemi di memorizzazione nella cache

Ti consigliamo di combinare `apt-get` i comandi in un'unica istruzione RUN all'interno del contenitore Docker. L'utilizzo `apt-get update` da solo comporta problemi di memorizzazione nella cache e l'esito negativo delle `apt-get install` istruzioni successive. Per ulteriori informazioni, vedere [apt-get](#) nel sito Web di Docker Docs.

Note

Il comportamento di memorizzazione nella cache descritto può verificarsi anche all'interno del Docker contenitore se il software del contenitore Docker non è aggiornato.

Utilizzo dell'utilità da riga di comando APT in modo non interattivo

Si consiglia di utilizzare l'utilità da riga di comando APT in modo interattivo. L'utilità da riga di comando APT è progettata come strumento per l'utente finale e il suo comportamento cambia tra le versioni. Per maggiori informazioni, vedere [Utilizzo degli script e differenze rispetto agli altri strumenti APT nel sito](#) web di Debian.

Segreti codificati

Note

Il grado di severità di questo controllo è Critico.

Le informazioni riservate nel tuo Dockerfile sono considerate un segreto codificato. I seguenti segreti codificati possono essere identificati tramite i controlli dei file Docker: `Sbomgen`

- AWS IDs chiave di accesso — AKIAIOSFODNN7EXAMPLE
- AWS chiavi segrete — wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
- DockerHub token di accesso personali — dckr_pat_thisisa27charexample1234567
- GitHub token di accesso personali — ghp_examplev61wY7Pj1YnotrealUoY123456789
- GitLab token di accesso personali — glpat-12345example12345678

contenitori root

Note

L'indicatore di gravità per questo controllo è Info.

Consigliamo di eseguire contenitori Docker senza privilegi di root. Per i carichi di lavoro containerizzati che non possono essere eseguiti senza i privilegi di root, consigliamo di creare le applicazioni utilizzando un principio con il minor numero di privilegi. Per ulteriori informazioni, consulta [User](#) nel sito Web Docker Docs.

Variabili di ambiente che indeboliscono il runtime

Note

Il grado di severità per questo controllo è Alto.

Diverse utilità da riga di comando o runtime dei linguaggi di programmazione supportano l'aggiornamento delle impostazioni predefinite sicure, il che consente l'esecuzione con metodi non sicuri.

`NODE_TLS_REJECT_UNAUTHORIZED=0`

Quando i processi vengono eseguiti con `set to`, la convalida del certificato TLS è disabilitata. Node.js `NODE_TLS_REJECT_UNAUTHORIZED 0` Per ulteriori informazioni, consulta [NODE_TLS_REJECT_UNAUTHORIZED=0](#) nel sito Web Node.js.

`GIT_SSL_NO_VERIFY=*`

Quando i processi della riga di comando git vengono eseguiti con `GIT_SSL_NO_VERIFY` set, Git salta la verifica dei certificati TLS. Per ulteriori informazioni, consulta [Variabili di ambiente](#) nel sito Web Git.

`PIP_TRUSTED_HOST=*`

Quando i processi della riga di comando Python pip vengono eseguiti con `PIP_TRUSTED_HOST` set, Pip salta la verifica dei certificati TLS sul dominio specificato. Per ulteriori informazioni, consulta [--trusted-host](#) nel sito Web di Pip.

`NPM_CONFIG_STRICT_SSL=false`

Quando i processi della riga di comando Node.js npm vengono eseguiti con `NPM_CONFIG_STRICT_SSL` set to false, l'utilità Node Package Manager (npm) si conatterà al registro NPM senza convalidare i certificati TLS. Per ulteriori informazioni, vedere [strict-ssl](#) nel sito Web di npm Docs.

Flag di comando che indeboliscono il runtime

Note

Il grado di severità di questo controllo è Alto.

Analogamente alle variabili di ambiente che indeboliscono il runtime, diverse utilità della riga di comando o runtime dei linguaggi di programmazione supportano l'aggiornamento delle impostazioni predefinite sicure, il che consente l'esecuzione con metodi non sicuri.

npm --strict-ssl=false

Quando i processi della riga di comando npm di Node.js vengono eseguiti con il `--strict-ssl=false` flag, l'utilità Node Package Manager (npm) si connette al registro NPM senza convalidare i certificati TLS. Per ulteriori informazioni, vedere [strict-ssl](#) nel sito Web di npm Docs.

apk --allow-untrusted

Quando l'Alpine Package Keeper utilità viene eseguita con il `--allow-untrusted` flag, apk installerà i pacchetti senza firme o non attendibili. Per ulteriori informazioni, consulta [il seguente repository](#) nel sito Web di Apline.

apt-get --allow-unauthenticated

Quando l'utilità dei pacchetti Debian `apt-get` viene eseguita con il flag `--allow-unauthenticated`, `apt-get` non controlla la validità del pacchetto. Per maggiori informazioni, vedere [apt-Get \(8\)](#) nel sito web di Debian.

pip --trusted-host

Quando l'utilità Python `pip` viene eseguita con il flag `--trusted-host`, il nome host specificato ignorerà la convalida del certificato TLS. Per ulteriori informazioni, consulta [--trusted-host](#) nel sito Web di Pip.

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

Quando il gestore di pacchetti basato su RPM `rpm` viene eseguito con i flag `--nofiledigest`, `--nodigest`, `--nosignature` e `--noverify`, il gestore di pacchetti RPM non convalida le intestazioni, le firme o i file dei pacchetti durante l'installazione di un pacchetto. Per ulteriori informazioni, consultate la seguente pagina di manuale RPM nel sito Web di [RPM](#).

yum-config-manager --setopt=sslverify false

Quando il gestore di pacchetti basato su RPM `yum-config-manager` viene eseguito con il flag `--setopt=sslverify` impostato su `false`, il gestore di pacchetti YUM non convalida i certificati TLS. Per ulteriori informazioni, consulta la seguente pagina di [manuale YUM nel sito Web](#) di Man7.

yum --nogpgcheck

Quando il gestore di pacchetti basato su RPM `yum` viene eseguito con il flag `--nogpgcheck`, il gestore di pacchetti YUM salta il controllo delle firme GPG sui pacchetti. Per maggiori informazioni, vedete [yum \(8\)](#) nel sito web di Man7.

curl --insecure, curl -k

Quando `curl` viene eseguito con il flag `--insecure` o `-k`, la convalida del certificato TLS è disabilitata. Per impostazione predefinita, ogni connessione sicura `curl` effettuata viene verificata prima che il trasferimento abbia luogo. Questa opzione consente di `curl` saltare la fase di verifica e di procedere senza controllare. Per ulteriori informazioni, consulta la seguente [pagina di manuale di Curl nel sito web](#) di Curl.

wget --no-check-certificate

Quando `wget` viene eseguito con il flag `--no-check-certificate`, la convalida del certificato TLS è disabilitata. Per ulteriori informazioni, consultate la seguente [pagina di manuale di Wget nel sito web](#) GNU.

Controlli di rimozione per i database dei pacchetti del sistema operativo all'interno dei contenitori

Note

Il grado di gravità di questo controllo è Info.

La rimozione di un database dei pacchetti del sistema operativo riduce la capacità di scansionare l'inventario completo del software dell'immagine di un contenitore. Questi database devono essere lasciati intatti durante le fasi di creazione del contenitore.

I controlli di rimozione per un database di pacchetti del sistema operativo sono supportati per i seguenti gestori di pacchetti:

Alpine Package Keeper (APK)

Le immagini dei container che utilizzano il gestore di pacchetti APK per il software installato devono garantire che i file di sistema APK non vengano rimossi durante la compilazione. Per ulteriori informazioni, consulta la documentazione dei file di sistema [APK manpages](#) sul Arch Linux sito Web.

Gestore di pacchetti Debian (DPKG)

I contenitori che utilizzano il gestore di pacchetti DPKG, come le immagini basate su Debian, Ubuntu o Distrosless, devono assicurarsi che il database DPKG non venga rimosso durante la creazione di un contenitore. Per ulteriori informazioni, consultate la documentazione dei file di sistema delle pagine di sistema di [DPKG sul sito web](#). Ubuntu

Gestore di pacchetti RPM (RPM)

I contenitori che utilizzano RPM Package Manager (yum/dnf), come Amazon Linux o Red Hat Enterprise Linux, devono assicurarsi che il database RPM non venga rimosso durante la creazione di un contenitore. Per ulteriori informazioni, consulta la documentazione relativa ai file di sistema di [RPM manpages sul sito Web RPM](#).

Creazione di un'integrazione di CI/CD pipeline personalizzata con Amazon Inspector Scan

Ti consigliamo di utilizzare i plug-in [Amazon Inspector se CI/CD i plug-in](#) Amazon Inspector sono disponibili per la tua CI/CD soluzione. CI/CD Se i CI/CD plugin Amazon Inspector non sono disponibili

per la tua CI/CD soluzione, puoi utilizzare una combinazione di Amazon Inspector SBOM Generator e Amazon Inspector Scan API per creare un'integrazione personalizzata. CI/CD I passaggi seguenti descrivono come creare un'integrazione di CI/CD pipeline personalizzata con Amazon Inspector Scan.

Tip

Puoi utilizzare [Amazon Inspector SBOM Generator \(Sbomgen\)](#) per saltare i passaggi 3 e 4 se desideri [generare e scansionare il tuo SBOM](#) con un solo comando.

Passaggio 1. Configurazione Account AWS

Configura un Account AWS dispositivo che fornisca l'accesso all'API Amazon Inspector Scan. Per ulteriori informazioni, consulta [Configurare un AWS account per utilizzare l'integrazione Amazon Inspector CI/CD](#).

Passaggio 2. Installazione Sborgen binary

Installa e configura il file binario. Sborgen Per ulteriori informazioni, consulta l'argomento relativo all'[installazione di Sborgen](#).

Fase 3. Utilizzo Sbomgen

Utilizzate il Sbomgen per creare un file SBOM per l'immagine di un contenitore che desiderate scansionare.

È possibile utilizzare l'esempio seguente. Sostituire *image:id* con il nome dell'immagine da scansionare. Sostituisci *sbom_path.json* con la posizione in cui desideri salvare l'output SBOM.

Esempio

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

Passaggio 4. Chiamata dell'API Amazon Inspector Scan

Chiama l'`inspector-scanAPI` per scansionare la SBOM generata e fornire un report di vulnerabilità.

È possibile utilizzare l'esempio seguente. Sostituire *sbom_path.json* con la posizione di un file SBOM compatibile con CyclonedX valido. *ENDPOINT* Sostituiscilo con l'endpoint API relativo al luogo in cui sei attualmente autenticato Regione AWS . Sostituisci *REGION* con la regione corrispondente.

Esempio

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

Per un elenco completo degli Regioni AWS endpoint, consulta [Regioni ed endpoint](#).

(Facoltativo) Fase 5. Genera e scansiona SBOM con un solo comando

Note

Completa questo passaggio solo se hai saltato i passaggi 3 e 4.

Genera e scansiona il tuo SBOM con un unico comando usando il flag. `--scan-bom`

È possibile utilizzare l'esempio seguente. Sostituisci *image:id* con il nome dell'immagine che desideri scansionare. Sostituire *profile* con il profilo corrispondente. Sostituisci *REGION* con la regione corrispondente. Sostituire */tmp/scan.json* con la posizione del file scan.json nella directory tmp.

Esempio

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

[Per un elenco completo degli endpoint, consulta Regioni AWS Regioni ed endpoint.](#)

Formati di output delle API

L'API Amazon Inspector Scan può generare un report di vulnerabilità in formato CycloneDX 1.5 o Amazon Inspector che trova JSON. L'impostazione predefinita può essere modificata utilizzando il flag. `--output-format`

Esempio di CyclonedX Output in formato 1.5 - Linux

```
{  
  "status": "SBOM parsed successfully, 1 vulnerabilities found",
```

```
"sbom": {
  "bomFormat": "CycloneDX",
  "specVersion": "1.5",
  "serialNumber": "urn:uuid:0077b45b-ff1e-4dbb-8950-ded11d8242b1",
  "metadata": {
    "properties": [
      {
        "name": "amazon:inspector:sbom_scanner:critical_vulnerabilities",
        "value": "1"
      },
      {
        "name": "amazon:inspector:sbom_scanner:high_vulnerabilities",
        "value": "0"
      },
      {
        "name": "amazon:inspector:sbom_scanner:medium_vulnerabilities",
        "value": "0"
      },
      {
        "name": "amazon:inspector:sbom_scanner:low_vulnerabilities",
        "value": "0"
      }
    ],
    "tools": [
      {
        "name": "CycloneDX SBOM API",
        "vendor": "Amazon Inspector",
        "version": "empty:083c9b00:083c9b00:083c9b00"
      }
    ],
    "timestamp": "2023-06-28T14:15:53.760Z"
  },
  "components": [
    {
      "bom-ref": "comp-1",
      "type": "library",
      "name": "log4j-core",
      "purl": "pkg:maven/org.apache.logging.log4j/log4j-core@2.12.1",
      "properties": [
        {
          "name": "amazon:inspector:sbom_scanner:path",
          "value": "/home/dev/foo.jar"
        }
      ]
    }
  ]
}
```

```
    }
  ],
  "vulnerabilities": [
    {
      "bom-ref": "vuln-1",
      "id": "CVE-2021-44228",
      "source": {
        "name": "NVD",
        "url": "https://nvd.nist.gov/vuln/detail/CVE-2021-44228"
      },
      "references": [
        {
          "id": "GHSA-jfh8-c2jp-5v3q",
          "source": {
            "name": "GITHUB",
            "url": "https://github.com/advisories/GHSA-jfh8-c2jp-5v3q"
          }
        }
      ],
      "ratings": [
        {
          "source": {
            "name": "NVD",
            "url": "https://www.first.org/cvss/v3-1/"
          },
          "score": 10.0,
          "severity": "critical",
          "method": "CVSSv31",
          "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
        },
        {
          "source": {
            "name": "NVD",
            "url": "https://www.first.org/cvss/v2/"
          },
          "score": 9.3,
          "severity": "critical",
          "method": "CVSSv2",
          "vector": "AC:M/Au:N/C:C/I:C/A:C"
        },
        {
          "source": {
            "name": "EPSS",
            "url": "https://www.first.org/epss/"
          }
        }
      ]
    }
  ]
}
```

```
    },
    "score": 0.97565,
    "severity": "none",
    "method": "other",
    "vector": "model:v2023.03.01,date:2023-06-27T00:00:00+0000"
  },
  {
    "source": {
      "name": "GITHUB",
      "url": "https://github.com/advisories/GHSA-jfh8-c2jp-5v3q"
    },
    "score": 10.0,
    "severity": "critical",
    "method": "CVSSv31",
    "vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H"
  }
],
"cwes": [
  400,
  20,
  502
],
"description": "Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints. An attacker who can control log messages or log message parameters can execute arbitrary code loaded from LDAP servers when message lookup substitution is enabled. From log4j 2.15.0, this behavior has been disabled by default. From version 2.16.0 (along with 2.12.2, 2.12.3, and 2.3.1), this functionality has been completely removed. Note that this vulnerability is specific to log4j-core and does not affect log4net, log4cxx, or other Apache Logging Services projects.",
"advisories": [
  {
    "url": "https://www.intel.com/content/www/us/en/security-center/advisory/intel-sa-00646.html"
  },
  {
    "url": "https://support.apple.com/kb/HT213189"
  },
  {
    "url": "https://msrc-blog.microsoft.com/2021/12/11/microsofts-response-to-cve-2021-44228-apache-log4j2/"
  }
]
```

```
    "url": "https://logging.apache.org/log4j/2.x/security.html"
  },
  {
    "url": "https://www.debian.org/security/2021/dsa-5020"
  },
  {
    "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf"
  },
  {
    "url": "https://www.oracle.com/security-alerts/alert-cve-2021-44228.html"
  },
  {
    "url": "https://www.oracle.com/security-alerts/cpujan2022.html"
  },
  {
    "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf"
  },
  {
    "url": "https://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/message/M5CSVUNV4HWZZXG0KNSK6L7RPM7B0KIB/"
  },
  {
    "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf"
  },
  {
    "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf"
  },
  {
    "url": "https://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/message/VU57UJDCFIASI035GC55JMKSXRJMCDFM/"
  },
  {
    "url": "https://www.oracle.com/security-alerts/cpuapr2022.html"
  },
  {
    "url": "https://twitter.com/kurtseifried/status/1469345530182455296"
  },
  {
    "url": "https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd"
  },
  {
    "url": "https://lists.debian.org/debian-lts-announce/2021/12/msg00007.html"
  },
  },
```

```

    {
      "url": "https://www.kb.cert.org/vuls/id/930724"
    }
  ],
  "created": "2021-12-10T10:15:00Z",
  "updated": "2023-04-03T20:15:00Z",
  "affects": [
    {
      "ref": "comp-1"
    }
  ],
  "properties": [
    {
      "name": "amazon:inspector:sbom_scanner:exploit_available",
      "value": "true"
    },
    {
      "name": "amazon:inspector:sbom_scanner:exploit_last_seen_in_public",
      "value": "2023-03-06T00:00:00Z"
    },
    {
      "name": "amazon:inspector:sbom_scanner:cisa_kev_date_added",
      "value": "2021-12-10T00:00:00Z"
    },
    {
      "name": "amazon:inspector:sbom_scanner:cisa_kev_date_due",
      "value": "2021-12-24T00:00:00Z"
    },
    {
      "name": "amazon:inspector:sbom_scanner:fixed_version:comp-1",
      "value": "2.15.0"
    }
  ]
}
]
}
}

```

Esempio di CyclonedX Output in formato 1.5 - Windows

```

{
  "sbom": {
    "specVersion": "1.5",

```

```
"metadata": {
  "tools": {
    "services": [
      {
        "name": "Amazon Inspector Scan SBOM API",
        "version": "d79c681c+d73b8663+5e50a5ab"
      }
    ]
  },
  "properties": [
    {
      "name": "amazon:inspector:sbom_scanner:critical_vulnerabilities",
      "value": "0"
    },
    {
      "name": "amazon:inspector:sbom_scanner:high_vulnerabilities",
      "value": "0"
    },
    {
      "name": "amazon:inspector:sbom_scanner:medium_vulnerabilities",
      "value": "1"
    },
    {
      "name": "amazon:inspector:sbom_scanner:low_vulnerabilities",
      "value": "0"
    },
    {
      "name": "amazon:inspector:sbom_scanner:other_vulnerabilities",
      "value": "0"
    }
  ],
  "timestamp": "2026-03-17T00:00:52.344Z"
},
"components": [
  {
    "bom-ref": "comp-1",
    "name": "defender",
    "purl": "pkg:generic/microsoft/defender@4.18.25110.5",
    "type": "application",
    "version": "4.18.25110.5",
    "properties": [
      {
        "name": "amazon:inspector:sbom_scanner:source_file_scanner",
        "value": "windows-apps"
      }
    ]
  }
]
```

```

    },
    {
      "name": "amazon:inspector:sbom_scanner:source_package_collector",
      "value": "windows-app-defender"
    },
    {
      "name": "amazon:inspector:sbom_scanner:path",
      "value": "vol-0d994b0984fdaa2af:\\ProgramData\\Microsoft\\Windows Defender\\
\\platform\\4.18.25110.5-0"
    }
  ]
},
"serialNumber": "urn:uuid:6bed582d-191e-4cb7-9875-950dd0b99700",
"bomFormat": "CycloneDX",
"vulnerabilities": [
  {
    "advisories": [
      {
        "url": "https://support.microsoft.com/help/5011487"
      },
      {
        "url": "https://catalog.update.microsoft.com/v7/site/Search.aspx?q=KB5011487"
      }
    ],
    "bom-ref": "vuln-1",
    "references": [
      {
        "id": "CVE-2022-23278",
        "source": {
          "name": "MICROSOFT",
          "url": "https://msrc.microsoft.com/update-guide/vulnerability/CVE-2022-23278"
        }
      }
    ],
    "ratings": [
      {
        "severity": "none",
        "score": 0.02691,
        "method": "other",
        "vector": "model:v2025.03.14,date:2026-03-15T12:55:00Z",
        "source": {

```

```

        "name": "EPSS",
        "url": "https://api.first.org/data/v1/epss?cve=CVE-2022-23278"
    }
},
{
    "severity": "medium",
    "score": 5.9,
    "method": "CVSSv31",
    "vector": "CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:H/A:N",
    "source": {
        "name": "MICROSOFT",
        "url": "https://catalog.update.microsoft.com/v7/site/Search.aspx?q=KB5011487"
    }
}
],
"created": "2022-03-08T08:00:00Z",
"description": "Security Update for Defender (2022-03). Install KB5011487 to remediate. A reboot is required for this update to take effect.",
"affects": [
    {
        "ref": "comp-1"
    }
],
"id": "KB5011487",
"source": {
    "name": "MICROSOFT",
    "url": "https://catalog.update.microsoft.com/v7/site/Search.aspx?q=KB5011487"
},
"published": "2022-03-08T08:00:00Z",
"analysis": {
    "state": "in_triage"
},
"properties": [
    {
        "name": "amazon:inspector:sbom_scanner:priority",
        "value": "standard"
    },
    {
        "name": "amazon:inspector:sbom_scanner:priority_intelligence",
        "value": "unverified"
    },
    {
        "name": "amazon:inspector:sbom_scanner:fixed_version:comp-1",

```

```

        "value": "10.0.19042.1586"
      }
    ]
  }
}
}

```

Esempio di output in formato Inspector - Linux

```

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

```

```
"references": [
  "https://www.intel.com/content/www/us/en/security-center/advisory/intel-sa-00646.html",
  "https://support.apple.com/kb/HT213189",
  "https://msrc-blog.microsoft.com/2021/12/11/microsofts-response-to-cve-2021-44228-apache-log4j2/",
  "https://logging.apache.org/log4j/2.x/security.html",
  "https://www.debian.org/security/2021/dsa-5020",
  "https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf",
  "https://www.oracle.com/security-alerts/alert-cve-2021-44228.html",
  "https://www.oracle.com/security-alerts/cpujan2022.html",
  "https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf",
  "https://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/message/M5CSVUNV4HWZZXG0KNSK6L7RPM7B0KIB/",
  "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf",
  "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf",
  "https://lists.fedoraproject.org/archives/list/package-announce@lists.fedoraproject.org/message/VU57UJDCFIASI035GC55JMKSRXJMCDFM/",
  "https://www.oracle.com/security-alerts/cpuapr2022.html",
  "https://twitter.com/kurtseifried/status/1469345530182455296",
  "https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd",
  "https://lists.debian.org/debian-lts-announce/2021/12/msg00007.html",
  "https://www.kb.cert.org/vuls/id/930724"
],
"created": "2021-12-10T10:15:00Z",
"updated": "2023-04-03T20:15:00Z",
"properties": {
  "cisa_kev_date_added": "2021-12-10T00:00:00Z",
  "cisa_kev_date_due": "2021-12-24T00:00:00Z",
  "cwes": [
    400,
    20,
    502
  ],
},
"cvss": [
  {
    "source": "NVD",
    "severity": "critical",
    "cvss3_base_score": 10.0,
    "cvss3_base_vector": "AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H",
    "cvss2_base_score": 9.3,
    "cvss2_base_vector": "AC:M/Au:N/C:C/I:C/A:C"
  },
]
```

```

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

```

Esempio di output in formato Inspector - Windows

```

{
  "sbom": {
    "vulnerabilities": [
      {
        "severity": "medium",
        "priority_intelligence": "unverified",
        "related": [
          "CVE-2022-23278"
        ],
        "references": [
          "https://support.microsoft.com/help/5011487",
          "https://catalog.update.microsoft.com/v7/site/Search.aspx?q=KB5011487"
        ],
        "created": "2022-03-08T08:00:00Z",
        "description": "Security Update for Defender (2022-03). Install KB5011487 to remediate. A reboot is required for this update to take effect.",
      }
    ]
  }
}

```

```

    "affects": [
      {
        "path": "vol-0d994b0984fdaa2af:\\ProgramData\\Microsoft\\Windows Defender\\
\\platform\\4.18.25110.5-0",
        "fixed_version": "10.0.19042.1586",
        "installed_version": "pkg:generic/microsoft/defender@4.18.25110.5"
      }
    ],
    "id": "KB5011487",
    "source": "https://catalog.update.microsoft.com/v7/site/Search.aspx?
q=KB5011487",
    "published": "2022-03-08T08:00:00Z",
    "priority": "standard",
    "properties": {
      "epss": 0.0269099995,
      "cvss": [
        {
          "severity": "medium",
          "cvss_3_base_score": 5.90000000954,
          "cvss_3_base_vector": "CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:H/A:N",
          "source": "MICROSOFT",
          "url": "https://catalog.update.microsoft.com/v7/site/Search.aspx?
q=KB5011487"
        }
      ]
    }
  ],
  "vulnerability_count": {
    "high": 0,
    "other": 0,
    "critical": 0,
    "low": 0,
    "medium": 1
  }
}
}

```

Utilizzo di Amazon Inspector con GitHub actions

Puoi usare Amazon Inspector con [GitHub actions](#) per aggiungere scansioni di vulnerabilità di Amazon Inspector ai tuoi flussi di lavoro. GitHub Questo sfrutta [Amazon Inspector SBOM Generator e](#)

l'API Amazon [Inspector Scan](#) per produrre report dettagliati alla fine della build, in modo da poter analizzare e correggere i rischi prima della distribuzione. Le scansioni delle vulnerabilità di Amazon Inspector possono essere configurate per superare o fallire i flussi di lavoro in base al numero e alla gravità delle vulnerabilità rilevate. [È possibile visualizzare la versione più recente dell'azione Amazon Inspector sul GitHub sito Web](#). Per informazioni su come integrare Amazon Inspector Scan nella tua CI/CD pipeline, consulta [Integrazione delle scansioni di Amazon Inspector](#) nella tua pipeline. CI/CD Per un elenco dei sistemi operativi e dei linguaggi di programmazione supportati da Amazon Inspector, consulta [Sistemi operativi e linguaggi di programmazione supportati](#).

Utilizzo di Amazon Inspector Jenkins plugin

Il Jenkins plug-in sfrutta il binario [Amazon Inspector SBOM Generator](#) e l'API Amazon Inspector Scan per produrre report dettagliati alla fine della build, in modo da poter analizzare e correggere i rischi prima della distribuzione. Con il Jenkins plug-in Amazon Inspector, puoi aggiungere scansioni di vulnerabilità di Amazon Inspector alla tua pipeline. Le scansioni delle vulnerabilità di Amazon Inspector possono essere configurate per superare o fallire le esecuzioni della pipeline in base al numero e alla gravità delle vulnerabilità rilevate. Puoi visualizzare la versione più recente del plug-in nel marketplace all'indirizzo Jenkins <https://plugins.jenkins.io/amazon-inspector-image-scanner/>. I passaggi seguenti descrivono come configurare il plug-in Amazon Inspector Jenkins.

Important

Prima di completare i seguenti passaggi, è necessario aggiornare Jenkins alla versione 2.387.3 o superiore per consentire l'esecuzione del plug-in.

Passaggio 1. Configura un Account AWS

Configura un Account AWS con un ruolo IAM che consenta l'accesso all'API Amazon Inspector Scan. Per istruzioni, consulta [Configurare un AWS account per utilizzare l'integrazione Amazon Inspector CI/CD](#).

Passaggio 2. Installa il plugin Amazon Inspector Jenkins

La procedura seguente descrive come installare il plug-in Amazon Inspector Jenkins dalla dashboard Jenkins

1. Dalla dashboard di Jenkins, scegli Manage Jenkins, quindi scegli Manage Plugins.

2. Scegli Disponibile.
3. Dalla scheda Available, cerca Amazon Inspector Scans, quindi installa il plug-in.

(Facoltativo) Passaggio 3. Aggiungi le credenziali docker a Jenkins

Note

Aggiungi le credenziali docker solo se l'immagine docker si trova in un repository privato. In caso contrario, puoi ignorare questo passaggio.

La procedura seguente descrive come aggiungere credenziali docker Jenkins dalla dashboard. Jenkins

1. Dalla dashboard di Jenkins, scegli Gestisci Jenkins, Credenziali e quindi Sistema.
2. Scegli Credenziali globali, quindi Aggiungi credenziali.
3. Per Tipo, seleziona Nome utente con password.
4. Per Scope, seleziona Global (Jenkins, nodes, items, all child items, ecc.).
5. Inserisci i tuoi dati, quindi scegli OK.

(Facoltativo) Fase 4. Add (Aggiungi) AWS credenziali

Note

Aggiungi AWS le credenziali solo se desideri autenticarti in base a un utente IAM. In caso contrario, puoi ignorare questo passaggio.

La procedura seguente descrive come aggiungere AWS credenziali dalla dashboard. Jenkins

1. Dalla dashboard Jenkins, scegli Gestisci Jenkins, Credenziali e quindi Sistema.
2. Scegli Credenziali globali, quindi Aggiungi credenziali.
3. Per Tipo, seleziona AWS Credentials.
4. Inserisci i tuoi dati, tra cui l'ID della chiave di accesso e la chiave di accesso segreta, quindi scegli OK.

Fase 5. Aggiungi il supporto CSS in un Jenkins script

La procedura seguente descrive come aggiungere il supporto CSS in uno Jenkins script.

1. Riavvia Jenkins.
2. Dalla dashboard, scegli Manage Jenkins, Nodes, Built-In Node e quindi Script Console.
3. Nella casella di testo, aggiungi la riga `System.setProperty("hudson.model.DirectoryBrowserSupport.CSP", "")`, quindi scegli Esegui.

Fase 6. Aggiungi Amazon Inspector Scan alla tua build

Puoi aggiungere Amazon Inspector Scan alla tua build aggiungendo una fase di compilazione nel tuo progetto o utilizzando la pipeline Jenkins dichiarativa.

Amazon Inspector Scansiona la tua build aggiungendo una fase di compilazione al tuo progetto

1. Nella pagina di configurazione, scorri verso il basso fino a Build Steps e scegli Aggiungi fase di compilazione. Quindi seleziona Amazon Inspector Scan.
2. Scegli tra due metodi di installazione di inspector-sbomgen: automatico o manuale. L'opzione automatica consente al plugin di scaricare la versione più recente. Inoltre, ti assicura di avere sempre le funzionalità più recenti, gli aggiornamenti di sicurezza e le correzioni di bug.
 - a. (Opzione 1) Scegli Automatico per scaricare l'ultima versione di inspector-sbomgen. Questa opzione rileva automaticamente il sistema operativo e l'architettura della CPU attualmente in uso.
 - b. (Opzione 2) Scegli Manuale se desideri configurare il binario Amazon Inspector SBOM Generator per la scansione. Se scegli questo metodo, assicurati di fornire il percorso completo di una versione di inspector-sbomgen scaricata in precedenza.

[Per ulteriori informazioni, consulta Installazione di Amazon Inspector SBOM Generator \(Sbomgen\) in Amazon Inspector SBOM Generator](#).

3. Completa quanto segue per completare la configurazione della fase di compilazione di Amazon Inspector Scan:

- a. Inserisci il tuo ID immagine. L'immagine può essere locale, remota o archiviata. I nomi delle immagini devono seguire la convenzione di Docker denominazione. Se state analizzando un'immagine esportata, fornite il percorso del file tar previsto. Vedi il seguente esempio di percorsi Image Id:
 - i. Per contenitori locali o remoti: `NAME[:TAG|@DIGEST]`
 - ii. Per un file tar: `/path/to/image.tar`
 - b. Seleziona un tramite Regione AWS il quale inviare la richiesta di scansione.
 - c. (Facoltativo) Per Report Artifact Name, inserite un nome personalizzato per gli artefatti generati durante il processo di creazione. Questo aiuta a identificarli e gestirli in modo univoco.
 - d. (Facoltativo) Per Skip files, specificate una o più directory da escludere dalla scansione. Considerate questa opzione per le cartelle che non devono essere scansionate a causa delle loro dimensioni.
 - e. (Facoltativo) Per le credenziali Docker, seleziona il tuo nome utente. Docker Esegui questa operazione solo se l'immagine del contenitore si trova in un repository privato.
 - f. (Facoltativo) È possibile fornire i seguenti metodi di AWS autenticazione supportati:
 - i. (Facoltativo) Per il ruolo IAM, fornisci un ruolo ARN (`arn:aws:iam: :role/`).
AccountNumber RoleName
 - ii. (Facoltativo) Per le credenziali AWS, specifica AWS le credenziali per l'autenticazione in base a un utente IAM.
 - iii. (Facoltativo) Per il nome del AWS profilo, fornisci il nome di un profilo da autenticare utilizzando un nome di profilo.
 - g. (Facoltativo) Seleziona Abilita le soglie di vulnerabilità. Con questa opzione, puoi determinare se la compilazione fallisce se una vulnerabilità analizzata supera un valore. Se tutti i valori sono uguali 0, la compilazione ha esito positivo, indipendentemente dal numero di vulnerabilità analizzate. Per il punteggio EPSS, il valore può essere compreso tra 0 e 1. Se una vulnerabilità analizzata supera un valore, la compilazione fallisce e tutti i CVE con un punteggio EPSS superiore al valore vengono visualizzati nella console.
4. Scegli Save (Salva).

Aggiungi Amazon Inspector Scan alla tua build utilizzando Jenkins pipeline dichiarativa

Puoi aggiungere Amazon Inspector Scan alla tua build utilizzando la pipeline dichiarativa Jenkins automaticamente o manualmente.

Per scaricare automaticamente la pipeline dichiarativa SBOMGen

- Per aggiungere Amazon Inspector Scan a una build, usa la seguente sintassi di esempio. Sostituisci *IMAGE_PATH* con il percorso dell'immagine (ad esempio *alpine:latest*), *IAM_ROLE* con l'ARN del ruolo IAM che hai configurato nella fase 1 e *ID* con il tuo ID di Docker credenziale se utilizzi un repository privato. Facoltativamente, puoi abilitare le soglie di vulnerabilità e specificare i valori per ogni gravità.

```
pipeline {
  agent any
  stages {
    stage('amazon-inspector-image-scanner') {
      steps {
        script {
          step([
            $class:
'com.amazon.inspector.jenkins.amazoninspectorbuildstep.AmazonInspectorBuilder',
            archivePath: 'IMAGE_PATH', // Path to your container image or tar file
            awsRegion: 'REGION', // AWS region for scan requests
            iamRole: 'IAM_ROLE', // IAM role ARN for authentication
            credentialId: 'Id', // Docker credentials (empty if public repo)
            awsCredentialId: 'AWS ID', // AWS credential ID for authentication
            awsProfileName: 'Profile Name', // AWS profile name to use
            sbomgenSkipFiles: '*.log,node_modules,/tmp/*', // Files/directories to
exclude from scanning

            // Vulnerability threshold settings (updated parameter names)
            isSeverityThresholdEnabled: false, // Enable/disable build failure on
vulnerability count
            countCritical: 0, // Max critical vulnerabilities before build fails
            countHigh: 0, // Max high vulnerabilities before build fails
            countMedium: 5, // Max medium vulnerabilities before build fails
            countLow: 10, // Max low vulnerabilities before build fails

            // EPSS (Exploit Prediction Scoring System) settings
```

```

threshold
    isEpssThresholdEnabled: false, // Enable/disable EPSS-based failure
    epssThreshold: 0.7, // EPSS score threshold (0.0 to 1.0)

// NEW FEATURE: CVE Suppression - ignore specific false positives
thresholds
    isSuppressedCveEnabled: false, // Enable CVE suppression feature
    suppressedCveList: '', // Comma-separated list of CVEs to ignore in

issues
    // NEW FEATURE: Auto-Fail CVEs - always fail on critical security
    isAutoFailCveEnabled: false, // Enable auto-fail CVE feature
    autoFailCveList: '' // Comma-separated list of CVEs that always fail

build
    ])
    }
    }
}

```

Per scaricare manualmente la pipeline dichiarativa SBOMGen

- Per aggiungere Amazon Inspector Scan a una build, usa la seguente sintassi di esempio. Sostituisci **SBOMGEN_PATH** con il percorso del generatore SBOM di Amazon Inspector installato nella fase 3, **IMAGE_PATH** con il percorso dell'immagine (ad esempio `alpine:latest`), con **IAM_ROLE** l'ARN del ruolo IAM configurato nella fase 1 e **ID** con il tuo ID Docker credenziale se utilizzi un repository privato. Facoltativamente, puoi abilitare le soglie di vulnerabilità e specificare i valori per ogni gravità.

Note

SbomgenInseriscilo nella directory Jenkins e fornisci il percorso della directory Jenkins nel plugin (ad esempio). `/opt/folder/arm64/inspector-sbomgen`

```

pipeline {
    agent any
    stages {
        stage('amazon-inspector-image-scanner') {
            steps {

```

```

    script {
    step([
    $class:
'com.amazon.inspector.jenkins.amazoninspectorbuildstep.AmazonInspectorBuilder',
    archivePath: 'IMAGE_PATH', // Path to your container image or tar file
    awsRegion: 'REGION', // AWS region for scan requests
    iamRole: 'IAM ROLE', // IAM role ARN for authentication
    credentialId: 'Id', // Docker credentials (empty if public repo)
    awsCredentialId: 'AWS ID', // AWS credential ID for authentication
    awsProfileName: 'Profile Name', // AWS profile name to use
    sbomgenSkipFiles: '*.log,node_modules,/tmp/*', // Files/directories to
exclude from scanning

    // Vulnerability threshold settings (updated parameter names)
    isSeverityThresholdEnabled: false, // Enable/disable build failure on
vulnerability count
    countCritical: 0, // Max critical vulnerabilities before build fails
    countHigh: 0, // Max high vulnerabilities before build fails
    countMedium: 5, // Max medium vulnerabilities before build fails
    countLow: 10, // Max low vulnerabilities before build fails

    // EPSS (Exploit Prediction Scoring System) settings
    isEpsThresholdEnabled: false, // Enable/disable EPSS-based failure
threshold
    epsThreshold: 0.7, // EPSS score threshold (0.0 to 1.0)

    // NEW FEATURE: CVE Suppression - ignore specific false positives
    isSuppressedCveEnabled: false, // Enable CVE suppression feature
    suppressedCveList: '', // Comma-separated list of CVEs to ignore in
thresholds

    // NEW FEATURE: Auto-Fail CVEs - always fail on critical security
issues
    isAutoFailCveEnabled: false, // Enable auto-fail CVE feature
    autoFailCveList: '' // Comma-separated list of CVEs that always fail
build

    ])
    }
}
}
}
}

```

Il plugin include funzionalità per la gestione delle vulnerabilità di sicurezza.

Elenco CVE soppresso

Le scansioni possono occasionalmente rilevare vulnerabilità che non sono minacce reali. Per evitare che questi falsi positivi interrompano la compilazione, puoi aggiungerli a un elenco soppresso.

```
isSuppressedCveEnabled: true,  
suppressedCveList: 'CVE-2023-1234,CVE-2023-5678'
```

Questo ignora i CVE specifici quando si verifica se la build deve fallire. Dovresti aggiungere falsi positivi all'elenco soppresso solo se li hai risolti. Dopo aver aggiunto queste vulnerabilità all'elenco delle vulnerabilità eliminate, le CVE continuano a essere visualizzate nel rapporto di sicurezza, ma non causeranno errori di compilazione.

Auto-Fail Elenco CVE

Per le vulnerabilità di sicurezza critiche, puoi creare un elenco che causi sempre il fallimento della compilazione.

```
isAutoFailCveEnabled: true,  
autoFailCveList: 'CVE-2024-9999'
```

Ciò causa sempre il fallimento delle tue build, indipendentemente dalle impostazioni che hai abilitato. Dovresti creare questo elenco solo per problemi di sicurezza ad alta priorità che non dovrebbero mai essere implementati. L'elenco sostituisce tutte le altre impostazioni di soglia per la massima sicurezza.

Fase 7. Visualizza il report sulla vulnerabilità di Amazon Inspector

1. Completa una nuova build del tuo progetto.
2. Al termine della compilazione, seleziona un formato di output dai risultati. Se selezioni HTML, hai la possibilità di scaricare una versione JSON SBOM o CSV del rapporto. Di seguito viene illustrato un esempio di report HTML:

Inspector Vulnerability Report

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

[Download SBOM](#)
[Download CSV](#)

SBOM parsed successfully, 7 vulnerabilities found.

Information

Image name	Image SHA
file:///Users/naveshal/Downloads/alpine.tar	sha256:5977be310a9d079b4febfe923ccd67daf776253c0baddf2488259b3b7c5ef70

Vulnerability by severity

Critical	High	Medium	Low
1	4	2	0

All vulnerabilities (7)

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

Note

Puoi usare script precedenti, poiché il plugin supporta vecchi nomi di parametri. Tuttavia, nella console verranno visualizzati degli avvisi che suggeriscono di aggiornare questi parametri con quelli più recenti. Ad esempio, se si utilizza `isThresholdEnabled`, verrà visualizzato un avviso che suggerisce di aggiornare il parametro a `isSeverityThresholdEnabled`.

Risoluzione dei problemi

Di seguito sono riportati gli errori più comuni che puoi riscontrare quando utilizzi il plug-in Amazon Inspector Scan per Jenkins.

Caricamento delle credenziali non riuscito o errore di eccezione

Errore:

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

Risoluzione

Ottieni `aws_access_key_id` e `aws_secret_access_key` per il tuo account. AWS Configura `aws_access_key_id` e `aws_secret_access_key` accedi~/ .aws/credentials.

Impossibile caricare l'immagine da fonti tarball, locali o remote

Errore:

```
2024/10/16 02:25:17 [ImageDownloadFailed]: failed to load image from tarball, local, or remote sources.
```

Note

Questo errore può verificarsi se il plug-in Jenkins non è in grado di leggere l'immagine del contenitore, l'immagine del contenitore non viene trovata nel Docker motore e l'immagine del contenitore non viene trovata nel registro del contenitore remoto.

Risoluzione:

Verifica quanto segue;

- L'utente del plugin Jenkins dispone dei permessi di lettura per l'immagine che desideri scansionare.
- L'immagine che desideri scansionare è presente nel Docker motore.
- L'URL dell'immagine remota è corretto.
- Sei autenticato nel registro remoto (se applicabile).

Inspector-sbomgen errore di percorso

Errore:

```
Exception:com.amazon.inspector.jenkins.amazoninspectorbuildstep.exception.SbomgenException: There was an issue running inspector-sbomgen, is /opt/inspector/inspector-sbomgen the correct path?
```

Risoluzione:

Completare la procedura seguente per risolvere il problema.

1. Inserisci l'architettura del sistema operativo corretta Inspector-sbomgen nella Jenkins directory. Per ulteriori informazioni, consulta [Amazon Inspector SBOM Generator](#).
2. Concedi le autorizzazioni eseguibili al file binario utilizzando il seguente comando: `chmod +x inspector-sbomgen`

3. Fornisci il percorso corretto del Jenkins computer nel plug-in, ad esempio/opt/folder/arm64/inspector-sbomgen.
4. Salva la configurazione ed esegui il Jenkins lavoro.

Utilizzo CodeCatalyst azioni con Amazon Inspector

[Puoi usare Amazon Inspector con Amazon CodeCatalyst per aggiungere scansioni di vulnerabilità di Amazon Inspector ai tuoi flussi di lavoro.](#) CodeCatalyst Questo sfrutta [Amazon Inspector SBOM Generator](#) e l'API Amazon [Inspector Scan](#) per produrre report dettagliati alla fine della build, in modo da poter analizzare e correggere i rischi prima della distribuzione. Le scansioni delle vulnerabilità di Amazon Inspector possono essere configurate per superare o fallire i flussi di lavoro in base al numero e alla gravità delle vulnerabilità rilevate. Per informazioni su come integrare Amazon Inspector Scan nella tua CI/CD pipeline, consulta [Integrazione delle scansioni di Amazon Inspector](#) nella tua pipeline. CI/CD Per un elenco dei sistemi operativi e dei linguaggi di programmazione supportati da Amazon Inspector, consulta [Sistemi operativi e linguaggi di programmazione supportati](#).

Utilizzo delle azioni di scansione di Amazon Inspector con CodePipeline

Puoi usare Amazon Inspector AWS CodePipeline aggiungendo scansioni di vulnerabilità ai tuoi flussi di lavoro. Questa integrazione sfrutta Amazon Inspector SBOM Generator e Amazon Inspector Scan API per produrre report dettagliati alla fine della build. L'integrazione ti aiuta a indagare e correggere i rischi prima della distribuzione. L'InspectorScanazione è un'azione di elaborazione gestita CodePipeline che automatizza il rilevamento e la correzione delle vulnerabilità di sicurezza nel codice open source. Puoi utilizzare questa azione con il codice sorgente dell'applicazione nel tuo repository di terze parti, ad esempio Bitbucket Cloud, GitHub o con immagini per applicazioni container. Per ulteriori informazioni, consulta il [riferimento all'azione InspectorScan invoke](#) nella Guida per l'utente.AWS CodePipeline

Utilizzo di Amazon Inspector TeamCity plugin

Note

Il TeamCity plug-in Amazon Inspector non è più disponibile ed è archiviato su. GitHub

Il TeamCity plug-in Amazon Inspector sfrutta il binario Amazon Inspector SBOM Generator e l'API Amazon Inspector Scan per produrre report dettagliati alla fine della build, in modo da poter esaminare e correggere i rischi prima della distribuzione. Con il TeamCity plug-in Amazon Inspector, puoi aggiungere scansioni di vulnerabilità di Amazon Inspector alla tua pipeline. TeamCity Le scansioni delle vulnerabilità di Amazon Inspector possono essere configurate per superare o fallire le esecuzioni della pipeline in base al numero e alla gravità delle vulnerabilità rilevate. È possibile visualizzare la versione archiviata del plug-in GitHub Amazon TeamCity Inspector all'indirizzo. <https://github.com/aws/amazon-inspector-container-image-scanner-teamcity-plugin>

Utilizzo di Amazon Inspector con GitLab componenti

Note

Il GitLab componente Amazon Inspector non è supportato ufficialmente da Amazon. È gestito e fornito come open source da. GitLab

Puoi usare Amazon Inspector con [GitLab CI/CD i componenti per aggiungere scansioni](#) di vulnerabilità di Amazon Inspector ai tuoi progetti. GitLab Questo sfrutta [Amazon Inspector SBOM Generator](#) e l'API Amazon [Inspector Scan](#) per produrre report dettagliati alla fine della build, in modo da poter analizzare e correggere i rischi prima della distribuzione. Le scansioni delle vulnerabilità di Amazon Inspector possono essere configurate per superare o fallire i flussi di lavoro in base al numero e alla gravità delle vulnerabilità rilevate. [Puoi visualizzare la versione più recente del componente Amazon Inspector sul GitLab sito Web](#). Per informazioni su come integrare Amazon Inspector Scan nella tua CI/CD pipeline, consulta Integrazione delle scansioni di [Amazon Inspector](#) nella tua pipeline. CI/CD Per un elenco dei sistemi operativi e dei linguaggi di programmazione supportati da Amazon Inspector, consulta [Sistemi operativi e linguaggi di programmazione supportati](#).

Valutazione della copertura di Amazon Inspector del tuo ambiente AWS

Puoi valutare la copertura di Amazon Inspector del tuo AWS ambiente dalla schermata di gestione dell'account nella console Amazon Inspector, che mostra dettagli e statistiche sullo stato delle scansioni di Amazon Inspector per i tuoi account e le tue risorse.

Note

Se sei l'amministratore delegato di un'organizzazione, puoi visualizzare dettagli e statistiche per tutti gli account dell'organizzazione.

La procedura seguente descrive come valutare la copertura del tuo ambiente Amazon Inspector.

Per valutare la copertura di Amazon Inspector del tuo ambiente AWS

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Dal pannello di navigazione, scegli Gestione dell'account.
3. Per rivedere la copertura, scegli una delle seguenti schede:
 - Scegli Account per esaminare la copertura a livello di account.
 - Scegli Istanze per esaminare la copertura delle istanze Amazon Elastic Compute Cloud (Amazon EC2).
 - Scegli i repository Container per esaminare la copertura dei repository Amazon Elastic Container Registry (Amazon ECR).
 - Scegli le immagini dei container per esaminare la copertura delle immagini dei container Amazon ECR.
 - Scegli le funzioni Lambda per esaminare la copertura delle funzioni Lambda.

I seguenti argomenti descrivono le informazioni fornite da ciascuna di queste schede.

Argomenti

- [Valutazione della copertura a livello di account](#)

- [Valutazione della copertura delle istanze Amazon EC2](#)
- [Valutazione della copertura dei repository Amazon ECR](#)
- [Valutazione della copertura delle immagini dei container Amazon ECR](#)
- [Valutazione della copertura delle funzioni AWS Lambda](#)

Valutazione della copertura a livello di account

Se il tuo account non fa parte di un'organizzazione o non è l'account amministratore delegato di Amazon Inspector di un'organizzazione, la scheda Account fornisce informazioni sul tuo account e sullo stato della scansione delle risorse per il tuo account. In questa scheda, puoi attivare o disattivare la scansione di tutti o solo tipi specifici di risorse per il tuo account. Per ulteriori informazioni, consulta [Tipi di scansione automatizzati in Amazon Inspector](#).

Se il tuo account è l'account amministratore delegato di Amazon Inspector per un'organizzazione, la scheda Account fornisce le impostazioni di attivazione automatica per gli account della tua organizzazione ed elenca tutti gli account dell'organizzazione. Per ogni account, l'elenco indica se Amazon Inspector è attivato per l'account e, in caso affermativo, i tipi di scansione delle risorse attivati per l'account. In qualità di amministratore delegato, puoi utilizzare questa scheda per modificare le impostazioni di attivazione automatica per la tua organizzazione. È inoltre possibile attivare o disattivare tipi specifici di scansione delle risorse per gli account dei singoli membri. Per ulteriori informazioni, consulta [Attivazione delle scansioni Amazon Inspector per gli account dei membri](#).

Valutazione della copertura delle istanze Amazon EC2

La scheda Istanze mostra le istanze Amazon EC2 nel tuo ambiente. AWS Gli elenchi sono organizzati in gruppi nelle seguenti schede:

- **Tutto:** mostra tutte le istanze presenti nell'ambiente. La colonna Stato indica lo stato di scansione corrente di un'istanza.
- **Scansione:** mostra tutte le istanze che Amazon Inspector monitora e analizza attivamente nel tuo ambiente.
- **Nessuna scansione:** mostra tutte le istanze che Amazon Inspector non monitora e non analizza nel tuo ambiente. La colonna Reason indica perché Amazon Inspector non monitora e analizza un'istanza.

Un'istanza EC2 può apparire nella scheda Not scanning per diversi motivi. Amazon Inspector utilizza AWS Systems Manager (SSM) e l'agente SSM per monitorare e scansionare automaticamente le istanze EC2 alla ricerca di vulnerabilità. Se un'istanza non ha l'agente SSM in esecuzione, non ha un ruolo AWS Identity and Access Management (IAM) che supporti Systems Manager o non esegue un sistema operativo o un'architettura supportati, Amazon Inspector non può monitorare e scansionare l'istanza. Per ulteriori informazioni, consulta [Scansione delle istanze Amazon EC2](#).

In ogni scheda, la colonna Account specifica chi possiede un' Account AWS istanza.

Tag dell'istanza EC2: questa colonna mostra i tag associati all'istanza e può essere utilizzata per determinare se l'istanza è stata esclusa dalle scansioni per tag.

Sistema operativo: questa colonna mostra il tipo di sistema operativo, che può essere WINDOWS, MAC LINUX, o UNKNOWN

Utilizzo monitorato: questa colonna mostra se Amazon Inspector utilizza il metodo di scansione [basato su agenti](#) o [senza agente](#) su questa istanza.

Ultima scansione: questa colonna mostra l'ultima volta che Amazon Inspector ha verificato la presenza di vulnerabilità nella risorsa. La frequenza con cui Amazon Inspector esegue le scansioni dipende dal metodo di scansione utilizzato per scansionare l'istanza.

Per visualizzare ulteriori dettagli su un'istanza EC2, scegli il link nella colonna delle istanze EC2. Amazon Inspector visualizza quindi i dettagli sull'istanza e i risultati correnti relativi all'istanza. Per esaminare i dettagli di un risultato, scegli il link nella colonna Titolo. Per informazioni su questi dettagli, consulta [Visualizzazione dei dettagli relativi ai risultati di Amazon Inspector](#).

Valori dello stato di scansione per le istanze Amazon EC2

Per un'istanza Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute EC2, i valori Status possibili sono:

- Monitoraggio attivo: Amazon Inspector monitora e analizza continuamente l'istanza.
- Limite di storage dell'istanza senza agente superato: Amazon Inspector utilizza questo stato quando la dimensione combinata di tutti i volumi collegati a un'istanza è superiore a 1200 GB o un'istanza ha più di 8 volumi collegati.

- Limite di tempo di raccolta delle istanze senza agente superato: Amazon Inspector si verifica un timeout durante il tentativo di eseguire una scansione senza agente su un'istanza.
- Istanza EC2 interrotta: Amazon Inspector ha sospeso la scansione dell'istanza perché l'istanza si trova in uno stato interrotto. Tutti i risultati esistenti persisteranno fino alla chiusura dell'istanza. Se l'istanza viene riavviata, Amazon Inspector riprenderà automaticamente la scansione dell'istanza.
- Errore interno: si è verificato un errore interno quando Amazon Inspector ha tentato di scansionare l'istanza. Amazon Inspector risolverà automaticamente l'errore e riprenderà la scansione il prima possibile.
- Nessun inventario: Amazon Inspector non è riuscito a trovare l'inventario delle applicazioni software da scansionare per l'istanza. Le associazioni Amazon Inspector per l'istanza potrebbero essere state eliminate o potrebbero non essere state eseguite.

Per risolvere questo problema, usa AWS Systems Manager per assicurarti che l'`InspectorInventoryCollection-do-not-delete` associazione esista e che il suo stato di associazione abbia esito positivo. Inoltre, utilizzate AWS Systems Manager Fleet Manager per verificare l'inventario delle applicazioni software per l'istanza.

- Disattivazione in sospenso: Amazon Inspector ha interrotto la scansione dell'istanza. L'istanza viene disabilitata, in attesa del completamento delle attività di pulizia.
- Scansione iniziale in sospenso: Amazon Inspector ha messo in coda l'istanza per una scansione iniziale.
- Risorsa terminata: l'istanza è stata terminata. Amazon Inspector sta attualmente ripulendo i risultati e i dati di copertura esistenti per l'istanza.
- Inventario obsoleto: Amazon Inspector non è stato in grado di raccogliere un inventario aggiornato delle applicazioni software acquisito negli ultimi 7 giorni per l'istanza.

Per risolvere questo problema, assicurati che AWS Systems Manager le associazioni Amazon Inspector richieste esistano e siano in esecuzione per l'istanza. Inoltre, utilizza AWS Systems Manager Fleet Manager per verificare l'inventario delle applicazioni software per l'istanza.

- Istanza EC2 non gestita: Amazon Inspector non monitora o analizza l'istanza. L'istanza non è gestita da AWS Systems Manager

Per risolvere questo problema, puoi utilizzare il [AWS Support-TroubleshootManagedInstance runbook](#) servizio fornito da AWS Systems Manager Automation. Dopo la configurazione AWS Systems Manager per la gestione dell'istanza, Amazon Inspector inizierà automaticamente a monitorare e scansionare continuamente l'istanza.

- Sistema operativo non supportato: Amazon Inspector non monitora o scansiona l'istanza. L'istanza utilizza un sistema operativo o un'architettura che Amazon Inspector non supporta. Per un elenco dei sistemi operativi supportati da Amazon Inspector, consulta [Valori di stato delle istanze Amazon EC2](#)
- Monitoraggio attivo con errori parziali: questo stato indica che la scansione EC2 è attiva, ma sono presenti errori associati. [Ispezione approfondita di Amazon Inspector per istanze Amazon Linux-based EC2](#) I possibili errori nelle ispezioni approfondite sono:
 - Limite di raccolta dei pacchetti con ispezione approfondita superato: l'istanza ha superato il limite di 5000 pacchetti per l'ispezione approfondita di Amazon Inspector. Per riprendere l'ispezione approfondita per questa istanza, puoi provare a modificare i percorsi personalizzati associati all'account.
 - Superato il limite di inventario SSM giornaliero di Deep Inspector: l'agente SSM non è riuscito a inviare l'inventario ad Amazon Inspector perché la quota SSM per i dati di inventario raccolti per istanza al giorno è già stata raggiunta per questa istanza. Per ulteriori informazioni, consulta [Endpoint e quote di Amazon EC2 Systems Manager](#).
 - Superato il limite di ritiro per l'ispezione approfondita: Amazon Inspector non è riuscito a estrarre l'inventario del pacco perché il tempo di ritiro del pacco ha superato la soglia massima di 15 minuti.
 - L'ispezione approfondita non ha un inventario: il [plug-in Amazon Inspector SSM](#) non è ancora stato in grado di raccogliere un inventario dei pacchetti per questo caso. Di solito è il risultato di una scansione in sospeso, tuttavia, se questo stato persiste dopo 6 ore, usa Amazon EC2 Systems Manager per assicurarti che le associazioni Amazon Inspector richieste esistano e siano in esecuzione per l'istanza. Questo errore può verificarsi anche se all'istanza mancano le autorizzazioni e le autorizzazioni richieste. `ssm:PutInventory` `ssm:GetParameter` Se l'istanza ha un profilo di istanza IAM, verifica che tali autorizzazioni siano incluse nel profilo. Se non è configurato alcun profilo di istanza, verifica che queste autorizzazioni siano incluse nel ruolo [Default Host Management Configuration](#).

Per i dettagli sulla configurazione delle impostazioni di scansione per un'istanza EC2, consulta [Scansione delle istanze Amazon EC2](#)

Valutazione della copertura dei repository Amazon ECR

La scheda Repositories mostra i repository Amazon ECR presenti nel tuo ambiente. AWS Gli elenchi sono organizzati in gruppi nelle seguenti schede:

- **Tutti:** mostra tutti i repository presenti nell'ambiente. La colonna Stato indica lo stato di scansione corrente di un repository.
- **Attivato:** mostra tutti i repository che Amazon Inspector è configurato per monitorare e scansionare nel tuo ambiente. La colonna Status indica lo stato di scansione corrente di un repository.
- **Non attivato:** mostra tutti i repository che Amazon Inspector non monitora e non analizza nel tuo ambiente. La colonna Reason indica perché Amazon Inspector non monitora e scansiona un repository.

In ogni scheda, la colonna Account specifica il Account AWS proprietario di un repository.

Per esaminare ulteriori dettagli su un repository, scegli il nome del repository. Amazon Inspector visualizza quindi un elenco di immagini dei container nel repository e i dettagli per ogni immagine. I dettagli includono il tag dell'immagine, l'immagine digest e lo stato della scansione. Includono anche statistiche chiave sui risultati, come il numero di risultati critici per l'immagine. Per approfondire ed esaminare i dati di supporto per la ricerca di statistiche, scegli il tag dell'immagine.

Note

Le immagini Amazon ECR senza scansione continua non sono incluse nei widget di copertura.

Valori dello stato di scansione per i repository Amazon ECR

Per un repository Amazon Elastic Container Registry (Amazon ECR), i valori Status possibili sono:

- **Attivato (continuo):** per un repository, Amazon Inspector monitora continuamente le immagini in questo repository. L'impostazione di scansione avanzata per il repository è impostata sulla scansione continua. Amazon Inspector esegue inizialmente la scansione di nuove immagini quando vengono inviate e scansiona nuovamente le immagini se viene pubblicato un nuovo CVE relativo a quell'immagine. Amazon Inspector continuerà a monitorare le immagini in questo repository per la durata della [nuova scansione di Amazon ECR configurata](#).
- **Attivato (in modalità push):** Amazon Inspector analizza automaticamente le immagini dei singoli container nel repository quando viene inviata una nuova immagine. La scansione avanzata è attivata per il repository e impostata per la scansione in modalità push.
- **Accesso negato:** Amazon Inspector non è autorizzato ad accedere al repository o alle immagini dei container in esso contenute.

Per risolvere questo problema, assicurati che le policy AWS Identity and Access Management (IAM) per il repository consentano ad Amazon Inspector di accedere al repository.

- Disattivato (manuale): Amazon Inspector non monitora o scansiona le immagini dei container nel repository. L'impostazione di scansione Amazon ECR per il repository è impostata sulla scansione manuale di base.

Per iniziare a scansionare le immagini nel repository con Amazon Inspector, modifica l'impostazione di scansione del repository su Scansione avanzata, quindi scegli se scansionare le immagini in modo continuo o solo quando viene inviata una nuova immagine.

- Attivato (in modalità push): Amazon Inspector analizza automaticamente le immagini dei singoli container nel repository quando viene inviata una nuova immagine. L'impostazione di scansione avanzata per il repository è impostata per la scansione in modalità push.
- Errore interno: si è verificato un errore interno quando Amazon Inspector ha tentato di scansionare il repository. Amazon Inspector risolverà automaticamente l'errore e riprenderà la scansione il prima possibile.

Per informazioni dettagliate sulla configurazione delle impostazioni di scansione per gli archivi.

[Scansione delle immagini dei contenitori Amazon ECR](#)

Valutazione della copertura delle immagini dei container Amazon ECR

La scheda Immagini mostra le immagini dei container Amazon ECR nel tuo AWS ambiente. Gli elenchi sono organizzati in gruppi nelle seguenti schede:

- Tutto: mostra tutte le immagini dei contenitori presenti nell'ambiente. La colonna Stato indica lo stato di scansione corrente di un'immagine.
- Scansione: mostra tutte le immagini dei container che Amazon Inspector è configurato per monitorare e scansionare nel tuo ambiente. La colonna Status indica lo stato di scansione corrente di un'immagine.
- Nessuna scansione: mostra tutte le immagini dei container che Amazon Inspector non monitora e non analizza nel tuo ambiente. La colonna Reason indica perché Amazon Inspector non monitora e scansiona un'immagine.

L'immagine di un contenitore può apparire nella scheda Non attivato per diversi motivi. L'immagine potrebbe essere archiviata in un repository per il quale le scansioni di Amazon Inspector non sono attivate oppure le regole di filtro di Amazon ECR impediscono la scansione di tale repository. Oppure l'immagine non è stata spostata o recuperata entro il numero di giorni configurato per la durata della nuova scansione ECR. Per ulteriori informazioni, consulta [Configurazione della durata della nuova scansione di Amazon ECR](#).

In ogni scheda, la colonna Repository name specifica il nome del repository che memorizza l'immagine del contenitore. La colonna Account specifica il proprietario del Account AWS repository. La colonna Ultima scansione mostra l'ultima volta che Amazon Inspector ha verificato la presenza di vulnerabilità in quella risorsa. Ciò può includere controlli in caso di aggiornamento della ricerca dei metadati, di aggiornamento dell'inventario delle applicazioni della risorsa o di esecuzione di una nuova scansione in risposta a un nuovo CVE. Per ulteriori informazioni, consulta [Comportamenti di scansione per la scansione Amazon ECR](#).

Per visualizzare ulteriori dettagli sull'immagine di un contenitore, scegliete il link nella colonna Immagine del contenitore ECR. Amazon Inspector visualizza quindi i dettagli sull'immagine e i risultati attuali relativi all'immagine. Per esaminare i dettagli di un risultato, scegli il link nella colonna Titolo. Per informazioni su questi dettagli, consulta [Visualizzazione dei dettagli relativi ai risultati di Amazon Inspector](#).

Valori dello stato di scansione per le immagini dei container Amazon ECR

Per un'immagine del contenitore Amazon Elastic Container Registry, i possibili valori Status sono:

- **Monitoraggio attivo (continuo):** Amazon Inspector monitora continuamente e l'immagine e le nuove scansioni vengono eseguite su di essa ogni volta che viene pubblicato un nuovo CVE pertinente. La durata della nuova scansione di Amazon ECR per l'immagine viene aggiornata ogni volta che l'immagine viene spinta o estratta. La scansione avanzata è abilitata per l'archivio che memorizza l'immagine e l'impostazione di scansione avanzata per il repository è impostata sulla scansione continua.
- **Attivato (in modalità push):** Amazon Inspector esegue automaticamente la scansione dell'immagine ogni volta che viene inviata una nuova immagine. La scansione avanzata è attivata per l'archivio che memorizza l'immagine e l'impostazione di scansione avanzata per il repository è impostata per la scansione in modalità push.

- **Errore interno:** si è verificato un errore interno quando Amazon Inspector ha tentato di scansionare l'immagine del contenitore. Amazon Inspector risolverà automaticamente l'errore e riprenderà la scansione il prima possibile.
- **Scansione iniziale in sospenso:** Amazon Inspector ha messo in coda l'immagine per una scansione iniziale.
- **Idoneità alla scansione scaduta (continua):** Amazon Inspector ha sospeso la scansione dell'immagine. L'immagine non è stata aggiornata entro la durata specificata per le scansioni automatiche delle immagini nel repository. È possibile spingere o tirare l'immagine per riprendere la scansione.
- **Idoneità alla scansione scaduta (in fase di invio):** Amazon Inspector ha sospeso la scansione dell'immagine. L'immagine non è stata aggiornata entro la durata specificata per le scansioni automatiche delle immagini nel repository. È possibile premere l'immagine per riprendere la scansione.
- **Frequenza di scansione manuale (manuale):** Amazon Inspector non esegue la scansione dell'immagine del contenitore Amazon ECR. L'impostazione di scansione Amazon ECR per il repository che memorizza l'immagine è impostata sulla scansione manuale di base. Per avviare la scansione automatica dell'immagine con Amazon Inspector, modifica l'impostazione del repository su Enhanced Scanning, quindi scegli se scansionare le immagini in modo continuo o solo quando viene inviata una nuova immagine.
- **Sistema operativo non supportato:** Amazon Inspector non monitora o scansiona l'immagine. L'immagine è basata su un sistema operativo non supportato da Amazon Inspector o utilizza un tipo di supporto non supportato da Amazon Inspector.

Per un elenco dei sistemi operativi supportati da Amazon Inspector, consulta [Sistemi operativi supportati: scansione Amazon ECR con Amazon Inspector](#). Per un elenco dei tipi di file multimediali supportati da Amazon Inspector, consulta [Tipi di file multimediali supportati](#).

Per informazioni dettagliate sulla configurazione delle impostazioni di scansione per archivi e immagini, consulta [Scansione delle immagini dei contenitori Amazon ECR](#).

Valutazione della copertura delle funzioni AWS Lambda

La scheda Lambda mostra le funzioni Lambda nel tuo ambiente. AWS Questa pagina contiene due tabelle, una che mostra i dettagli della copertura delle funzioni per la scansione standard Lambda e l'altra per la scansione del codice Lambda. È possibile raggruppare le funzioni in base alle seguenti schede:

- **Tutte:** mostra tutte le funzioni Lambda nel tuo ambiente. La colonna Status indica lo stato di scansione corrente per una funzione Lambda.
- **Scansione:** mostra le funzioni Lambda che Amazon Inspector è configurato per scansionare. La colonna Status indica lo stato di scansione corrente per ogni funzione Lambda.
- **Nessuna scansione:** mostra le funzioni Lambda che Amazon Inspector non è configurato per scansionare. La colonna Reason indica perché Amazon Inspector non monitora e analizza una funzione.

Una funzione Lambda può apparire nella scheda Not scanning per diversi motivi. La funzione Lambda potrebbe appartenere a un account che non è stato aggiunto ad Amazon Inspector o le regole di filtro impediscono la scansione di questa funzione. Per ulteriori informazioni, consulta [Funzione di scansione Lambda](#).

In ogni scheda, la colonna Nome funzione specifica il nome della funzione Lambda. La colonna Account specifica il proprietario della Account AWS funzione. Runtime specifica il runtime della funzione. La colonna Status indica lo stato di scansione corrente per ogni funzione Lambda. I tag delle risorse mostrano i tag che sono stati applicati alla funzione. La colonna Ultima scansione mostra l'ultima volta che Amazon Inspector ha verificato la presenza di vulnerabilità in quella risorsa. Ciò può includere controlli in caso di aggiornamento della ricerca dei metadati, di aggiornamento dell'inventario delle applicazioni della risorsa o di esecuzione di una nuova scansione in risposta a un nuovo CVE. Per ulteriori informazioni, consulta [Comportamenti di scansione per la scansione della funzione Lambda](#).

Scansione dei valori di stato delle funzioni AWS Lambda

Per una funzione Lambda, i possibili valori Status sono:

- **Monitoraggio attivo:** Amazon Inspector monitora e analizza continuamente le funzioni Lambda. La scansione continua include una scansione iniziale delle nuove funzioni quando vengono inserite nell'archivio e una nuova scansione automatica delle funzioni quando vengono aggiornate o quando vengono rilasciate nuove vulnerabilità ed esposizioni comuni (CWEs).
- **Esclusa per tag:** Amazon Inspector non analizza questa funzione perché è stata esclusa dalle scansioni tramite tag.
- **Idoneità alla scansione scaduta:** Amazon Inspector non monitora questa funzione perché sono trascorsi 90 giorni o più dall'ultima volta che è stata richiamata o aggiornata.

- **Errore interno:** si è verificato un errore interno quando Amazon Inspector ha tentato di scansionare la funzione. Amazon Inspector risolverà automaticamente l'errore e riprenderà la scansione il prima possibile.
- **Scansione iniziale in sospenso:** Amazon Inspector ha messo in coda la funzione per una scansione iniziale.
- **Non supportato:** la funzione Lambda ha un runtime non supportato.

Gestione di più account in Amazon Inspector con AWS Organizations

Puoi utilizzare Amazon Inspector per gestire più account in [un'organizzazione](#). Amazon Inspector supporta due approcci per la gestione di più account:

- **Amministratore delegato per AWS Organizations le politiche:** fornisce una governance centralizzata all'amministratore delegato con l'abilitazione automatica di Amazon Inspector tra gli account dell'organizzazione in tutte le regioni. Le politiche dell'organizzazione stabiliscono quali tipi di scansione sono abilitati e hanno la precedenza sulle abilitazioni degli account delegati di amministratori e membri non gestiti da policy.
- **Amministratore delegato per mancanza di AWS Organizations policy:** account designato per gestire Amazon Inspector per l'organizzazione senza utilizzare le politiche dell'organizzazione. L'amministratore delegato può abilitare Amazon Inspector per gli account dei membri e configurare le impostazioni di scansione.

Questi approcci possono essere usati insieme. Quando le politiche dell'organizzazione sono in atto, controllano l'abilitazione dei tipi di risorse (quali tipi di scansione sono abilitati), mentre gli amministratori delegati mantengono il controllo sulle impostazioni di configurazione della scansione, come le modalità di scansione e i percorsi di ispezione approfondita. I seguenti argomenti descrivono questi approcci di gestione, come designare un amministratore delegato e come gestire gli account dei membri.

Argomenti

- [Informazioni sull'account amministratore delegato e sull'account membro in Amazon Inspector](#)
- [Designazione di un account amministratore delegato per Amazon Inspector](#)

Informazioni sull'account amministratore delegato e sull'account membro in Amazon Inspector

Quando si utilizza Amazon Inspector in un ambiente con più account, l'account amministratore delegato ha accesso a metadati specifici. I metadati includono la scansione standard per Amazon EC2, Amazon ECR e Lambda e la scansione del codice Lambda. Include anche i risultati dei risultati

di ricerca sulla sicurezza per gli account dei membri. Questa sezione fornisce informazioni sulle azioni che l'account amministratore delegato può eseguire e sugli account membro.

Modello di governance delle politiche organizzative

Quando vengono utilizzate AWS Organizations policy per abilitare Amazon Inspector, viene applicato un modello di governance che determina quali azioni sono consentite:

Risorse gestite mediante policy

Le risorse abilitate o disabilitate esplicitamente dalle politiche dell'organizzazione non possono essere modificate dagli amministratori delegati o dagli account dei membri. Le richieste API per abilitare o disabilitare i tipi di scansione gestiti tramite policy avranno esito negativo e verrà visualizzato un chiaro errore che indica che la risorsa è gestita dai criteri dell'organizzazione.

Non-policy-managed resources

Le risorse non specificate nelle politiche dell'organizzazione possono essere gestite normalmente da amministratori delegati e account membri utilizzando la console o l'API Amazon Inspector.

Gestione della configurazione delle scansioni

Gli amministratori delegati possono sempre configurare le impostazioni di scansione come le modalità di scansione EC2, i [percorsi di ispezione approfondita](#) e la durata delle nuove scansioni ECR, indipendentemente dal fatto che i tipi di risorse siano gestiti in base a policy. Le politiche dell'organizzazione controllano solo se la scansione è abilitata, non il suo funzionamento.

Per ulteriori informazioni sulla creazione e la gestione delle politiche organizzative di Amazon Inspector, consulta la AWS Organizations documentazione relativa alle politiche di Amazon Inspector.

Azioni degli amministratori delegati

In genere, quando l'amministratore delegato applica impostazioni al proprio account, tali impostazioni vengono applicate a tutti gli altri account dell'organizzazione. L'amministratore delegato può inoltre visualizzare e recuperare informazioni relative al proprio account e a qualsiasi membro associato. Un account amministratore delegato di Amazon Inspector può eseguire le seguenti azioni:

- Solo l'account AWS Organizations di gestione può designare e rimuovere un amministratore delegato.
- Quando si designa un amministratore delegato, è necessario appartenere alla stessa organizzazione degli account dei membri che si desidera gestire.

- Visualizza e gestisci lo stato di Amazon Inspector per gli account associati, inclusa l'attivazione e la disattivazione di Amazon Inspector.
- Attiva o disattiva i tipi di scansione per tutti gli account membri dell'organizzazione.
- Visualizza i dati di ricerca aggregati in tutta l'organizzazione e i dettagli di ricerca per tutti gli account dei membri all'interno dell'organizzazione.
- Crea e gestisci regole di soppressione che si applicano ai risultati per tutti gli account dell'organizzazione.
- Attiva la scansione avanzata di Amazon ECR per tutti i membri dell'organizzazione.
- Visualizza la copertura delle risorse per l'intera organizzazione.
- Definisci la durata delle scansioni automatiche delle immagini dei contenitori ECR per tutti gli account membri dell'organizzazione. L'impostazione della durata della scansione dell'amministratore delegato ha la precedenza su qualsiasi impostazione precedentemente impostata dall'account membro. Tutti gli account dell'organizzazione condividono la durata di risanamento automatico di Amazon ECR degli amministratori delegati. Non è possibile impostare durate di risanamento diverse per singoli account.
- Specificate cinque percorsi personalizzati per l'ispezione approfondita di Amazon Inspector per Amazon EC2 che verranno utilizzati in tutti gli account dell'organizzazione. Questo si aggiunge ai cinque percorsi personalizzati che un amministratore delegato può impostare per il proprio account individuale. Per ulteriori informazioni sulla configurazione dei percorsi personalizzati di Deep Inspection, vedere. [Percorsi personalizzati per l'ispezione approfondita di Amazon Inspector](#)
- Attiva e disattiva l'ispezione approfondita di Amazon Inspector per gli account dei membri.
- [Esporta SBOMs](#) per tutti gli account dei membri dell'organizzazione.
- Imposta la modalità di scansione di Amazon EC2 per tutti gli account membri dell'organizzazione. Per ulteriori informazioni, consulta [Gestione della modalità di scansione](#).
- Crea e gestisci configurazioni di scansione CIS per tutti gli account dell'organizzazione, ad eccezione delle configurazioni di scansione create dagli account dei membri.

Note

Se un account membro lascia l'organizzazione, l'amministratore delegato non sarà più in grado di vedere le configurazioni di scansione pianificate da quell'account.

- Visualizza i risultati della scansione CIS per tutti gli account dell'organizzazione.

- Quando sono in uso i criteri dell'organizzazione, configura le impostazioni di scansione per le risorse gestite tramite policy, ma non può abilitare o disabilitare i tipi di scansione gestiti mediante policy stessi.

Azioni relative all'account dei membri

Un account membro può visualizzare e recuperare informazioni sul proprio account in Amazon Inspector, mentre le impostazioni dell'account sono gestite dall'amministratore delegato. Gli account dei membri all'interno di un'organizzazione possono eseguire le seguenti azioni in Amazon Inspector:

- Attiva Amazon Inspector per il proprio account.
- Visualizza la copertura delle risorse per il proprio account.
- Visualizza i dettagli dei risultati per il proprio account.
- Visualizza l'impostazione della durata della nuova scansione automatica dell'immagine del contenitore ECR per il proprio account.
- Specificate cinque percorsi personalizzati per l'ispezione approfondita di Amazon Inspector per EC2 che verranno utilizzati per il loro account individuale. Questi percorsi vengono analizzati in aggiunta a tutti i percorsi personalizzati che l'amministratore delegato ha specificato per l'organizzazione. Per ulteriori informazioni sulla configurazione dei percorsi di ispezione approfondita, vedere. [Percorsi personalizzati per l'ispezione approfondita di Amazon Inspector](#)
- Visualizza i percorsi personalizzati impostati dal tuo amministratore delegato per l'ispezione approfondita di Amazon Inspector.
- [Esporta](#) tutte SBOMs le risorse associate al loro account.
- Visualizza la modalità di scansione del loro account.
- Crea e gestisci le configurazioni di scansione CIS per il loro account.
- Visualizza i risultati di tutte le scansioni CIS relative alle risorse del relativo account, incluse quelle pianificate dall'amministratore delegato.
- Abilita i tipi di scansione che non sono gestiti dalle politiche dell'organizzazione. I tipi di scansione gestiti tramite policy non possono essere abilitati o disabilitati dagli account dei membri.

Note

Dopo l'attivazione, Amazon Inspector può essere disattivato solo da un account amministratore delegato.

Designazione di un account amministratore delegato per Amazon Inspector

L'amministratore delegato è un account che gestisce un servizio per un'organizzazione. Questo argomento descrive come designare un amministratore delegato per Amazon Inspector.

Considerazioni

Prima di designare un amministratore delegato, tieni presente quanto segue:

L'amministratore delegato può gestire un massimo di 10.000 membri.

Se superi i 10.000 account membri, ricevi una notifica tramite Amazon CloudWatch Personal Health Dashboard e un'e-mail all'account amministratore delegato.

Note

Quando Amazon Inspector è abilitato tramite AWS Organizations politiche per organizzazioni con più di 10.000 account (fino a 50.000), la politica si applica a tutti gli account. Tuttavia, solo 10.000 account saranno associati all'organizzazione Amazon Inspector, ovvero l'amministratore delegato può visualizzare i risultati e lo stato dell'account solo per questi 10.000 account nella console Amazon Inspector.

L'amministratore delegato è regionale.

Amazon Inspector è un servizio regionale. È necessario ripetere i passaggi della procedura in ogni Regione AWS luogo in cui si prevede di utilizzare Amazon Inspector.

Un'organizzazione può avere un solo amministratore delegato.

Se si designa un account come amministratore delegato in uno di essi Regione AWS, tale account deve essere l'amministratore delegato di tutti gli altri. Regioni AWS

La modifica di un amministratore delegato non disattiva Amazon Inspector per gli account dei membri.

Se rimuovi un amministratore delegato, gli account dei membri diventano account autonomi e le impostazioni di scansione non ne risentono.

La tua AWS organizzazione deve avere tutte le funzionalità attivate.

Questa è l'impostazione predefinita per AWS Organizations. Se non è attivata, vedi [Attivazione di tutte le funzionalità nell'organizzazione](#).

Le politiche dell'organizzazione hanno la precedenza sulle impostazioni degli amministratori delegati.

Se la tua organizzazione utilizza AWS Organizations policy per abilitare Amazon Inspector, le impostazioni delle policy determinano quali tipi di scansione sono abilitati. Ti consigliamo di designare l'amministratore delegato prima di creare politiche organizzative per garantire una governance coerente. Per ulteriori informazioni, consulta [Modello di governance delle politiche organizzative](#).

Autorizzazioni necessarie per designare un amministratore delegato

È necessario disporre dell'autorizzazione per attivare Amazon Inspector e designare un amministratore delegato di Amazon Inspector. Aggiungi la seguente dichiarazione alla fine della tua policy IAM per concedere queste autorizzazioni. Per ulteriori informazioni, consulta [Gestione delle politiche IAM](#).

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

Designazione di un amministratore delegato per l'organizzazione AWS

La procedura seguente descrive come designare un amministratore delegato per l'organizzazione. Prima di completare la procedura, assicurati di far parte della stessa organizzazione degli account dei membri che desideri vengano gestiti dall'amministratore delegato.

Note

È necessario utilizzare l'account AWS Organizations di gestione per completare questa procedura. Solo l'account AWS Organizations di gestione può designare un amministratore delegato. Potrebbero essere necessarie autorizzazioni per designare un amministratore delegato. Per ulteriori informazioni, consulta [Autorizzazioni necessarie per designare un amministratore delegato](#).

Quando attivi Amazon Inspector per la prima volta, Amazon Inspector crea il `AWSServiceRoleForAmazonInspector` ruolo collegato al servizio per l'account. Per informazioni su come Amazon Inspector utilizza i ruoli collegati ai servizi, consulta [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#)

Console

Per designare un amministratore delegato per Amazon Inspector

1. [Accedi all'account di AWS Organizations gestione, quindi apri la console Amazon Inspector su `https://console.aws.amazon.com/inspector/v2/home`.](https://console.aws.amazon.com/inspector/v2/home)
2. Usa il Regione AWS selettore per specificare Regione AWS dove vuoi designare l'amministratore delegato.
3. Dal riquadro di navigazione, scegli Impostazioni generali.
4. In Amministratore delegato, inserisci l'ID a 12 cifre della persona Account AWS che desideri designare come amministratore delegato.
5. Scegli Delegato, quindi scegli nuovamente Delegato.

Quando si designa un amministratore delegato, per impostazione predefinita vengono attivati [tutti i tipi di scansione](#) per l'account. Se desideri attivare Amazon Inspector per l'account di AWS Organizations gestione, completa la seguente procedura.

Per attivare Amazon Inspector per l'account di gestione AWS Organizations

1. [Accedi all'account amministratore delegato, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Dal pannello di navigazione, scegli Gestione dell'account.
3. In Account, seleziona l'account di AWS Organizations gestione, quindi scegli Attiva.
4. Seleziona i tipi di scansione che desideri attivare per l'account di AWS Organizations gestione, quindi scegli Invia.

API

Designate un amministratore delegato utilizzando l'API

- Esegui l'operazione [EnableDelegatedAdminAccount](#)API utilizzando le credenziali dell'account Account AWS di gestione Organizations. Puoi anche usare AWS Command Line Interface per farlo eseguendo il seguente comando CLI: `aws inspector2 enable-delegated-admin-account --delegated-admin-account-id 111111111111`

Note

Assicurati di specificare l'ID dell'account che desideri rendere amministratore delegato di Amazon Inspector.

Attivazione delle scansioni Amazon Inspector per gli account dei membri

Puoi attivare Amazon Inspector per gli account dei membri della tua organizzazione con diversi metodi. Il metodo scelto dipende dai requisiti di governance e dalla struttura organizzativa.

AWS Organizations politiche (consigliate per una governance centralizzata)

Utilizza AWS Organizations le policy per abilitare automaticamente Amazon Inspector in tutta l'organizzazione con un controllo centralizzato. Questo approccio garantisce una copertura di scansione coerente e si applica automaticamente ai nuovi account. Per istruzioni dettagliate, consulta la AWS Organizations documentazione per la creazione delle politiche di Amazon Inspector.

Attivazione con amministratore delegato

In qualità di amministratore delegato, puoi attivare manualmente Amazon Inspector per account membri specifici o per tutti gli account membro tramite la console o l'API Amazon Inspector. Questo approccio offre flessibilità quando le politiche organizzative non vengono utilizzate.

Auto-attivazione dell'account membro


Gli account dei membri possono attivare Amazon Inspector per il proprio account se non sono limitati dalle politiche dell'organizzazione. Una volta attivato, l'account viene associato all'amministratore delegato.

Attiva la scansione degli account dei membri

Le seguenti procedure descrivono come attivare la scansione degli account dei membri utilizzando i metodi amministratore delegato e account membro. Per informazioni sui tipi di scansione di Amazon Inspector, consulta [Tipi di scansione automatizzati in Amazon Inspector](#)


Per attivare automaticamente la scansione per tutti gli account dei membri

1. [Accedi utilizzando le credenziali dell'account amministratore delegato, quindi apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Usa il selettore di regione per scegliere Regione AWS dove vuoi attivare la scansione per tutti gli account dei membri.
3. Dal pannello di navigazione, scegli Gestione account. La scheda Account mostra tutti gli account dei membri associati all'account AWS Organizations di gestione.
4. In Organizzazione, seleziona la casella accanto a Numero di account. Quindi scegli Attiva per selezionare le opzioni di scansione che desideri applicare agli account dei membri. È possibile selezionare i seguenti tipi di scansione:
 - Scansione Amazon EC2
 - Scansione Amazon ECR
 - Scansione standard Lambda
 - Scansione del codice Lambda
- Dopo aver selezionato i tipi di scansione preferiti, scegli Salva.

 Note

Se disponi di più pagine di account, devi ripetere questo passaggio su ogni pagina. Puoi scegliere l'icona a forma di ingranaggio per modificare il numero di account visualizzati su ogni pagina.

5. Attiva l'impostazione Attiva automaticamente Inspector per gli account dei nuovi membri e seleziona le opzioni di scansione che desideri applicare ai nuovi account membro aggiunti alla tua organizzazione. È possibile selezionare i seguenti tipi di scansione:
 - Scansione Amazon EC2
 - Scansione Amazon ECR
 - Scansione standard Lambda
 - Scansione del codice Lambda
 - Dopo aver selezionato i tipi di scansione preferiti, scegli Attiva.

 Note

L'impostazione Attiva automaticamente Inspector per gli account dei nuovi membri attiva Amazon Inspector per tutti i futuri membri della tua organizzazione.


Se il numero di account membri è superiore a 5.000, questa impostazione viene disattivata automaticamente. Se il numero totale di account membri scende a meno di 5.000, l'impostazione viene riattivata automaticamente.

6. (Consigliato) Ripeti ciascuno di questi passaggi in ogni Regione AWS punto in cui desideri attivare la scansione degli account dei membri.

Per attivare la scansione di account membri specifici

1. [Accedi utilizzando le credenziali dell'account amministratore delegato, quindi apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/v2/home)
2. Usa il selettore di regione per scegliere Regione AWS dove vuoi attivare la scansione per tutti gli account dei membri.

3. Dal pannello di navigazione, scegli Gestione account. La scheda Account mostra tutti gli account dei membri associati all'account AWS Organizations di gestione.
4. In Organizzazione, seleziona la casella accanto al numero di account di ciascun membro per cui desideri attivare la scansione. Quindi scegli Attiva per selezionare le opzioni di scansione che desideri applicare agli account dei membri. È possibile selezionare i seguenti tipi di scansione:
 - Scansione Amazon EC2
 - Scansione Amazon ECR
 - Scansione standard Lambda
 - Scansione del codice Lambda
- Dopo aver selezionato i tipi di scansione preferiti, scegli Salva.

 Note

Se disponi di più pagine di account, devi ripetere questo passaggio su ogni pagina. Puoi scegliere l'icona a forma di ingranaggio per modificare il numero di account visualizzati su ogni pagina.

5. (Consigliato) Ripeti ciascuno di questi passaggi in ogni Regione AWS punto in cui desideri attivare la scansione per membri specifici.

Per attivare la scansione come account membro

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home.](https://console.aws.amazon.com/inspector/)
2. Usa il selettore di regione per scegliere Regione AWS dove vuoi attivare la scansione per tutti gli account dei membri.
3. Dal pannello di navigazione, scegli Gestione account. La scheda Account mostra tutti gli account dei membri associati all'account AWS Organizations di gestione.
4. In Organizzazione, seleziona la casella accanto al numero del tuo account. Quindi scegli Attiva per selezionare le opzioni di scansione che desideri applicare. È possibile selezionare i seguenti tipi di scansione:
 - Scansione Amazon EC2

- Scansione Amazon ECR
 - Scansione standard Lambda
 - Scansione del codice Lambda
- Dopo aver selezionato i tipi di scansione preferiti, scegli Salva.
5. (Consigliato) Ripeti questi passaggi in ogni regione in cui desideri attivare la scansione per il tuo account membro.

Note

Se il tuo account di AWS Organizations gestione dispone di un account amministratore delegato per Amazon Inspector, puoi attivarlo come account membro per visualizzare i dettagli della scansione.

Importante

Se le politiche dell'organizzazione gestiscono l'abilitazione di Amazon Inspector per i tuoi account, gli account amministratore e membro delegati non possono modificare i tipi di scansione gestiti dalle policy utilizzando Amazon Inspector. enablement/disablement APIs Le richieste API falliranno con un errore che indica che la risorsa è gestita dai criteri dell'organizzazione. È comunque possibile abilitare tipi di scansione aggiuntivi non gestiti dalla policy.

Dissociazione degli account dei membri in Amazon Inspector

In qualità di amministratore delegato, potresti dover dissociare un account membro dal tuo account. Quando si annulla l'associazione di un account membro, Amazon Inspector rimane attivo nell'account e l'account diventa un account autonomo. Inoltre, non sei più autorizzato a gestire Amazon Inspector per l'account. Tuttavia, puoi associare account membri precedentemente dissociati al tuo account in qualsiasi momento. Questa sezione descrive come dissociare gli account dei membri dalla funzione di amministratore delegato.

Note

Per dissociare gli account gestiti tramite policy, non dovrebbe essere associata alcuna policy organizzativa di Amazon Inspector a tale account per il tipo di scansione.

Console

Per dissociare gli account dei membri utilizzando la console

1. [Accedi utilizzando le credenziali dell'account amministratore delegato, quindi apri la console Amazon Inspector su v2/home https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Usa il selettore di regione per scegliere Regione AWS dove vuoi dissociare gli account dei membri.
3. Dal pannello di navigazione, scegli Gestione account.
4. In Organizzazione, seleziona la casella accanto a ciascun numero di account da dissociare.
5. Scegli il menu Azioni, quindi scegli Dissocia account.

API

Per dissociare gli account dei membri utilizzando l'API

Esegui l'operazione [DisassociateMember](#)API. Nella richiesta, fornisci l'account da IDs cui stai dissociando.

Rimozione dell'amministratore delegato in Amazon Inspector

Potrebbe essere necessario rimuovere l'account amministratore delegato di Amazon Inspector. Puoi farlo dall'account di AWS Organizations gestione. Quando rimuovi l'account amministratore delegato di Amazon Inspector, Amazon Inspector è ancora attivato nell'account e in tutti gli account dei membri. L'account amministratore delegato e tutti i relativi account membri diventano account autonomi e mantengono le impostazioni di scansione originali.

Note

Se AWS Organizations le policy gestiscono l'abilitazione di Amazon Inspector, la rimozione dell'amministratore delegato non influisce sull'applicazione delle policy. Gli account

rimarranno abilitati in base alle impostazioni delle politiche dell'organizzazione, tuttavia i risultati degli account dei membri non saranno più visibili in una console di amministrazione delegata centrale finché non verrà designato un nuovo amministratore delegato.

Questa sezione descrive come rimuovere l'account amministratore delegato.

Rimuovere l'amministratore delegato di Amazon Inspector

Le seguenti procedure descrivono come rimuovere l'amministratore delegato di Amazon Inspector e come associare gli account membro dall'account amministratore delegato.

Per informazioni su come assegnare un amministratore delegato di Amazon Inspector, [consulta Designazione di un account amministratore delegato per Amazon Inspector](#).

Note

Dopo aver assegnato un amministratore delegato di Amazon Inspector, l'amministratore delegato di Amazon Inspector deve associare gli account dei membri manualmente.

Per rimuovere l'amministratore delegato

1. Accedere all'account di gestione Console di gestione AWS utilizzando l'account AWS Organizations di gestione.
2. [Apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
3. Usa il selettore di regione per scegliere Regione AWS dove vuoi rimuovere l'amministratore delegato.
4. Dal riquadro di navigazione, scegli Impostazioni generali.
5. In Amministratore delegato, scegli Rimuovi, quindi conferma l'azione.

Per associare i membri a un nuovo amministratore delegato

1. [Accedi utilizzando le credenziali dell'account amministratore delegato, quindi apri la console Amazon Inspector su v2/home. https://console.aws.amazon.com/inspector/](https://console.aws.amazon.com/inspector/)
2. Usa il selettore della regione per scegliere Regione AWS dove vuoi associare i membri.
3. Dal riquadro di navigazione, scegli Gestione account.

4. In Organizzazione, seleziona la casella accanto a Numero di account.
5. Scegli Azioni, quindi scegli Aggiungi membro.

Etichettare le risorse di Amazon Inspector

Un tag è un'etichetta che aggiungi a una AWS risorsa. I tag consentono di classificare AWS le risorse in base a criteri specifici. I tag sono costituiti da una coppia chiave-valore. La chiave tag è un'etichetta generica. Il valore del tag è una descrizione della chiave del tag. Con Amazon Inspector, puoi impostare [regole di soppressione](#) dei tag e configurazioni di scansione [CIS](#). Puoi aggiungere fino a 50 tag a ciascuna delle tue risorse Amazon Inspector.

Nozioni fondamentali sull'etichettatura

Ciascun tag è costituito da una coppia chiave-valore. La chiave tag è un'etichetta generica. Il valore del tag è una descrizione della chiave del tag. Questo argomento descrive i fondamentali dell'etichettatura delle risorse di Amazon Inspector. Quando tagghi le risorse di Amazon Inspector, considera quanto segue:

- Puoi etichettare le [regole di soppressione e le configurazioni](#) di scansione [CIS](#).
- Puoi aggiungere fino a 50 tag a ciascuna delle tue risorse Amazon Inspector.
- Le chiavi dei tag devono essere uniche.
- Una chiave tag può avere un solo valore di tag.
- Le chiavi e i valori dei tag possono avere un massimo di 128 caratteri UTF-8. I caratteri possono essere lettere, numeri, spazi o i seguenti simboli: `_ . : / = + - @`
- Non puoi utilizzare il `aws` prefisso in nessuno dei tuoi tag o modificare i tag con questo prefisso. I tag con il `aws` prefisso possono essere utilizzati solo da AWS.
- I tag assegnati a una risorsa Amazon Inspector sono disponibili solo nel tuo AWS account e nel luogo in Regione AWS cui li hai creati.
- Quando elimini una risorsa, vengono eliminati anche tutti i tag ad essa associati.

Per ulteriori informazioni sui tag, consulta [le migliori pratiche e strategie](#) nella Guida per l'utente di Tagging AWS Resources and Tag Editor.

Note

I tag non sono destinati a memorizzare informazioni riservate o sensibili. Non utilizzare mai i tag per archiviare questo tipo di dati. I tag possono essere accessibili da altri AWS servizi.

Aggiunta di tag

Puoi aggiungere tag alle risorse di Amazon Inspector. Queste risorse includono regole di soppressione e configurazioni di scansione CIS. I tag consentono di classificare le AWS risorse in base a criteri specifici. Questo argomento descrive come aggiungere tag alle risorse di Amazon Inspector.

Aggiungere tag alle risorse di Amazon Inspector

Puoi impostare [regole di soppressione dei tag e configurazioni](#) di scansione [CIS](#). Le seguenti procedure descrivono come aggiungere tag nella console e con l'API Amazon Inspector.

Aggiungere tag nella console

Puoi aggiungere tag alle risorse di Amazon Inspector nella console.

Aggiungere tag alle regole di soppressione

È possibile aggiungere tag alle regole di soppressione durante la creazione. Per ulteriori informazioni, vedere [Creazione di una regola di soppressione](#).

È inoltre possibile modificare una regola di soppressione per includere i tag. Per ulteriori informazioni, vedere [Modifica di una regola di soppressione](#).

Aggiungere tag a una configurazione di scansione CIS

È possibile aggiungere tag a una configurazione di scansione CIS durante la creazione. Per ulteriori informazioni, vedere [Creazione di una configurazione di scansione CIS](#).

È inoltre possibile modificare una configurazione di scansione CIS per includere i tag. Per ulteriori informazioni, vedere [Modifica di una configurazione di scansione CIS](#).

Aggiungere tag con l'API Amazon Inspector

Puoi aggiungere tag alle risorse di Amazon Inspector con l'API Amazon Inspector.

Aggiungere tag alle risorse di Amazon Inspector

Utilizza l'[TagResource](#) API per aggiungere tag alle risorse di Amazon Inspector. È necessario includere l'ARN della risorsa e la coppia chiave-valore per il tag nel comando. Il comando di esempio

seguinte utilizza una risorsa ARN vuota per un filtro di soppressione. La chiave è `CostAllocation` e il valore è `dev`. Per informazioni sui tipi di risorse per Amazon Inspector, consulta [Azioni, risorse e chiavi di condizione per Amazon Inspector2](#) nel Service Authorization Reference.

```
aws inspector2 tag-resource \  
--resource-arn "arn:${Partition}:inspector2:${Region}:${Account}:owner/${OwnerId}/  
filter/${FilterId}" \  
--tags CostAllocation=dev \  
--region us-west-2
```

Aggiungere tag alle regole di soppressione durante la creazione

Utilizza l'[CreateFilter](#) API per aggiungere tag a una regola di soppressione durante la creazione.

```
aws inspector2 create-filter \  
--name "ExampleSuppressionRuleECR" \  
--action SUPPRESS \  
--filter-criteria 'resourceType=[{comparison="EQUALS", value="AWS_ECR_IMAGE"}]' \  
--tags Owner=ApplicationSecurity \  
--region us-west-2
```

Aggiungere tag a una configurazione di scansione CIS

Utilizzate l'[CreateCisScanConfiguration](#) API per aggiungere un tag a una configurazione di scansione CIS.

```
aws inspector2 create-cis-scan-configuration \  
--scan-name "CreateConfigWithTagsSample" \  
--security-level LEVEL_2 \  
--targets accountIds=SELF,targetResourceTags={InspectorCisScan=True} \  
--schedule 'daily={startTime={timeOfDay=11:10,timezone=UTC}}' \  
--tags Owner=SecurityEngineering \  
--region us-west-2
```

Rimozione dei tag

Puoi rimuovere i tag dalle risorse di Amazon Inspector. Queste risorse includono regole di soppressione e configurazioni di scansione CIS. I tag consentono di classificare le AWS risorse in base a criteri specifici. Questo argomento descrive come rimuovere i tag dalle risorse di Amazon Inspector.

Rimuovere i tag dalle risorse di Amazon Inspector

Puoi rimuovere i tag dalle [regole di soppressione](#) e dalle configurazioni di scansione [CIS](#). Le seguenti procedure descrivono come rimuovere i tag nella console e con l'API Amazon Inspector.

Rimuovere i tag nella console

Puoi rimuovere i tag dalle risorse di Amazon Inspector nella console.

Rimuovere i tag dalle regole di soppressione

È possibile rimuovere un tag da una regola di soppressione modificando la regola di soppressione in modo che non includa più il tag. Per ulteriori informazioni, vedere [Modifica di una regola di soppressione](#).

Rimozione di tag da una configurazione di scansione CIS

È possibile rimuovere un tag da una configurazione di scansione CIS modificando la configurazione di scansione CIS in modo che non includa più il tag. Per ulteriori informazioni, vedere [Modifica di una configurazione di scansione CIS](#).

Rimuovere i tag con l'API Amazon Inspector

Puoi rimuovere un tag da una risorsa Amazon Inspector con l'API Amazon Inspector.

Rimuovere i tag dalle risorse di Amazon Inspector

Utilizza l'[UntagResource](#) API per rimuovere i tag dalle risorse di Amazon Inspector.

Il seguente frammento mostra un esempio di come rimuovere un tag da una risorsa Amazon Inspector utilizzando `UntagResource`. È necessario includere l'ARN della risorsa e la chiave per il tag nel comando. L'esempio seguente utilizza una risorsa ARN vuota per un filtro di soppressione. La chiave è `CostAllocation`. Per informazioni sui tipi di risorse per Amazon Inspector, consulta [Azioni, risorse e chiavi di condizione per Amazon Inspector2](#) nel Service Authorization Reference.

```
aws inspector2 untag-resource \  
--resource-arn "arn:#{Partition}:inspector2:#{Region}:#{Account}:owner/#{OwnerId}/cis-  
configuration/#{CISScanConfigurationId}" \  
--tag-keys CostAllocation \  
--region us-west-2
```

Monitoraggio dell'utilizzo e dei costi in Amazon Inspector

Puoi utilizzare la console e l'API Amazon Inspector per proiettare i costi mensili di Amazon Inspector per il tuo ambiente. Se sei l'amministratore di Amazon Inspector per un ambiente con più account, puoi visualizzare il costo totale del tuo ambiente e le metriche di costo per tutti gli account membri. Questa sezione descrive come accedere alle statistiche di utilizzo e calcolare i costi di utilizzo.

Utilizzo della console di utilizzo

Puoi valutare l'utilizzo e il costo previsto per Amazon Inspector dalla console.

Per accedere alle statistiche di utilizzo

1. [Accedi utilizzando le tue credenziali, quindi apri la console \[https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home\]\(https://console.aws.amazon.com/inspector/AmazonInspector/v2/home\).](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home)
2. Utilizzando il Regione AWS selettore nell'angolo superiore destro della pagina, seleziona la regione in cui desideri monitorare i costi.
3. Nel riquadro di navigazione, scegli Utilizzo.

Nella scheda Per account vedrai il costo totale previsto in base al periodo di 30 giorni elencato nella sezione Utilizzo dell'account. Nella tabella sotto la colonna Costo previsto, seleziona un valore per visualizzare una suddivisione dell'utilizzo per tipo di scansione per quell'account. In questo riquadro dei dettagli puoi anche vedere per quali tipi di scansione è attiva una versione di prova gratuita per quell'account.

Se sei l'amministratore delegato di un'organizzazione, vedrai una riga nella tabella per ogni account all'interno dell'organizzazione. Se un account dell'organizzazione non è associato, la console mostra il costo previsto come -.

Nella scheda Per tipo di scansione è possibile visualizzare una suddivisione dell'utilizzo effettivo finora nell'attuale periodo di 30 giorni per tipo di scansione. Queste sono le informazioni utilizzate per calcolare i costi previsti nella scheda Per account.

Se sei l'amministratore delegato di un'organizzazione, puoi vedere l'utilizzo per ogni account dell'organizzazione.

In questa scheda, puoi espandere uno dei seguenti riquadri per le statistiche di utilizzo:

EC2 Scansione Amazon

La console di utilizzo di Amazon Inspector tiene traccia delle seguenti metriche per la scansione basata su agenti e la scansione senza agente:

- **Istanze (media):** Amazon Inspector utilizza le ore di copertura per calcolare il numero medio di risorse EC2 per la scansione delle istanze. La media è il totale delle ore di copertura divise per 720 ore (il numero di ore in un periodo di 30 giorni).
- **Ore di copertura:** per EC2 la scansione di Amazon si tratta del numero totale di ore negli ultimi 30 giorni in cui Amazon Inspector ha fornito copertura attiva per ogni EC2 istanza in un account. Ad esempio, EC2 le ore di copertura sono le ore che intercorrono tra il momento in cui Amazon Inspector ha scoperto l'istanza fino alla sua chiusura o arresto o all'esclusione dalle scansioni tramite tag. (quando riavvii un'istanza interrotta o rimuovi un tag di esclusione, Amazon Inspector riprende la copertura e le ore di copertura per quell'istanza continueranno ad accumularsi).

Scansioni delle istanze CIS: il numero totale di scansioni CIS eseguite per le istanze dell'account.

Scansione Amazon ECR

Scansioni iniziali: il totale delle prime scansioni delle immagini nell'account negli ultimi 30 giorni.

Scansioni ripetute: la somma totale delle scansioni ripetute delle immagini nell'account negli ultimi 30 giorni. Una nuova scansione è qualsiasi scansione eseguita su un'immagine ECR precedentemente scansionata da Amazon Inspector. Se hai configurato il tuo repository ECR per la scansione continua, le scansioni vengono eseguite automaticamente quando Amazon Inspector aggiunge un nuovo Common Vulnerabilities and Exposures (CVE) al database.

Scansione Lambda

La console di utilizzo di Amazon Inspector tiene traccia delle seguenti metriche per la scansione standard Lambda e la scansione del codice Lambda:

- **Numero di funzioni Lambda (media):** Amazon Inspector utilizza le ore di copertura per calcolare il numero medio di funzioni per la scansione della funzione Lambda. La media è il totale delle ore di copertura diviso per 720 ore (il numero di ore in un periodo di 30 giorni).
- **Ore di copertura:** per la scansione della funzione Lambda, si tratta del numero totale di ore negli ultimi 30 giorni in cui Amazon Amazon Inspector ha fornito la copertura attiva per ogni funzione Lambda in un account. Per quanto riguarda AWS Lambda le funzioni, le ore di copertura vengono calcolate dal momento in cui Amazon Inspector rileva una funzione fino a quando

questa viene eliminata o esclusa dalle scansioni. Se una funzione esclusa viene nuovamente inclusa, le ore di copertura per quella funzione continueranno a maturare.

Scopri come Amazon Inspector calcola i costi di utilizzo


I costi forniti da Amazon Inspector sono stime, non costi effettivi, pertanto possono differire da quelli indicati nella tua AWS Billing console.

Tieni presente quanto segue su come Amazon Inspector calcola i costi nella pagina Utilizzo:

- Il costo di utilizzo riflette solo la regione corrente. I prezzi per tipo di scansione variano in base alla AWS regione, per verificare i prezzi esatti per regione, consulta la pagina [Prezzi](#) di Amazon Inspector
- Tutte le proiezioni di utilizzo sono arrotondate al dollaro USA più vicino.
- Gli sconti non sono inclusi nei costi previsti.
- Il costo previsto rappresenta il costo totale per il periodo di utilizzo di 30 giorni per tipo di scansione. Se un account è stato utilizzato per meno di 30 giorni, Amazon Inspector prevede il costo dopo 30 giorni come se le risorse attualmente coperte restassero coperte per il resto del periodo di 30 giorni.
- Il costo per tipo di scansione viene calcolato in base a quanto segue:
 - EC2 scansione: il costo riflette il numero medio di EC2 istanze coperte da Amazon Inspector negli ultimi 30 giorni.
 - Scansione dei contenitori ECR: il costo riflette la somma del numero di scansioni iniziali delle immagini e delle scansioni ripetute delle immagini negli ultimi 30 giorni.
 - Scansione standard Lambda: il costo riflette il numero medio di funzioni Lambda coperte da Amazon Inspector negli ultimi 30 giorni.
 - Scansione del codice Lambda: il costo riflette il numero medio di funzioni Lambda coperte da Amazon Inspector negli ultimi 30 giorni.

Informazioni sulla versione di prova gratuita di Amazon Inspector

In Amazon Inspector, ogni [tipo di scansione](#) ha una traccia gratuita. Quando attivi un tipo di scansione, ti iscrivi automaticamente a una prova gratuita di 15 giorni per quel tipo di scansione. Una volta iniziata, la prova gratuita scade automaticamente dopo 15 giorni, anche se si disattiva il tipo di scansione.

 Note

La versione di prova gratuita non si applica alla scansione [CIS](#).

Sicurezza in Amazon Inspector

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di data center e architetture di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra te e te. AWS Il [modello di responsabilità condivisa](#) descrive questo aspetto come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi in Cloud AWS. AWS fornisce inoltre servizi che è possibile utilizzare in modo sicuro. Third-party i revisori testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito dei [AWS Programmi di AWS conformità dei Programmi di conformità](#) dei di . Per ulteriori informazioni sui programmi di conformità applicabili ad Amazon Inspector, consulta [AWS Services in Scope by Compliance Program](#) Program.
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. L'utente è anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della propria azienda e le leggi e normative vigenti.

Questa documentazione ti aiuta a capire come applicare il modello di responsabilità condivisa quando usi Amazon Inspector. I seguenti argomenti mostrano come configurare Amazon Inspector per soddisfare i tuoi obiettivi di sicurezza e conformità. Scopri anche come utilizzare altri AWS servizi che ti aiutano a monitorare e proteggere le tue risorse Amazon Inspector.

Argomenti

- [Protezione dei dati in Amazon Inspector](#)
- [Identity and Access Management per Amazon Inspector](#)
- [Monitoraggio di Amazon Inspector](#)
- [Convalida della conformità per Amazon Inspector](#)
- [Resilienza in Amazon Inspector](#)
- [Sicurezza dell'infrastruttura in Amazon Inspector](#)
- [Risposta agli incidenti in Amazon Inspector](#)
- [Ricerca sulla sicurezza di Amazon Inspector](#)
- [Accedi ad Amazon Inspector utilizzando un endpoint di interfaccia \(AWS PrivateLink\)](#)

Protezione dei dati in Amazon Inspector

Il [modello di](#) si applica alla protezione dei dati in Amazon Inspector. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che gestisce tutti i Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, consulta [Domande frequenti sulla privacy dei dati](#). Per ulteriori informazioni sulla protezione dei dati in Europa, consulta il [Centro generale sulla protezione dei dati \(GDPR\)](#).

Ai fini della protezione dei dati, ti consigliamo di proteggere Account AWS le credenziali e configurare singoli utenti con AWS IAM Identity Center o AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- SSL/TLS Da utilizzare per comunicare con AWS le risorse. È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con AWS CloudTrail. Per informazioni sull'utilizzo dei CloudTrail percorsi per acquisire AWS le attività, consulta [Lavorare con i CloudTrail percorsi](#) nella Guida per l'AWS CloudTrail utente.
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se hai bisogno di moduli crittografici convalidati FIPS 140-3 per accedere AWS tramite un'interfaccia a riga di comando o un'API, usa un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il [Federal Information Processing Standard \(FIPS\) 140-3](#).

Ti consigliamo di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando lavori con Amazon Inspector o altri utenti Servizi AWS utilizzando la console, l'API o AWS gli AWS CLI SDK. I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando si fornisce un URL a un server esterno, suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la richiesta al server.

Argomenti

- [Crittografia dei dati a riposo](#)
- [Crittografia dei dati in transito](#)

Crittografia dei dati a riposo

Per impostazione predefinita, Amazon Inspector archivia i dati inattivi utilizzando soluzioni di AWS crittografia. Amazon Inspector crittografa i dati, come i seguenti:

- Inventario delle risorse raccolto con AWS Systems Manager
- Inventario delle risorse analizzato dalle immagini di Amazon Elastic Container Registry
- Risultati di sicurezza generati utilizzando chiavi AWS di crittografia di proprietà di AWS Key Management Service

Non è possibile gestire, utilizzare o visualizzare le chiavi AWS di proprietà. Tuttavia, non è necessario agire o modificare i programmi per proteggere le chiavi che crittografano i dati. Per ulteriori informazioni, consulta [Chiavi di proprietà di AWS](#).

Se disabiliti Amazon Inspector, elimina definitivamente tutte le risorse che archivia o gestisce per te, come l'inventario raccolto e i risultati di sicurezza.

Crittografia inattiva per il codice contenuto nei tuoi risultati

Per la scansione del codice Amazon Inspector Lambda, Amazon Inspector collabora con Amazon Q per scansionare il codice alla ricerca di vulnerabilità. Quando viene rilevata una vulnerabilità, Amazon Q estrae un frammento di codice contenente la vulnerabilità e lo archivia fino a quando Amazon Inspector non richiede l'accesso. Per impostazione predefinita, Amazon Q utilizza una chiave AWS proprietaria per crittografare il codice estratto. Tuttavia, puoi configurare Amazon Inspector per utilizzare la tua chiave gestita dal cliente AWS KMS per la crittografia.

Il seguente flusso di lavoro spiega come Amazon Inspector utilizza la chiave configurata per crittografare il codice:

1. Fornisci una AWS KMS chiave ad Amazon Inspector utilizzando l'API Amazon [UpdateEncryptionKey](#) Inspector.
2. Amazon Inspector inoltra le informazioni sulla tua chiave AWS KMS ad Amazon Q e Amazon Q le memorizza per utilizzi futuri.

3. Amazon Q utilizza la chiave KMS configurata in Amazon Inspector tramite la policy chiave.
4. Amazon Q crea una chiave dati crittografata a partire dalla tua AWS KMS chiave e la archivia. Questa chiave dati viene utilizzata per crittografare i dati del codice archiviati da Amazon Q.
5. Quando Amazon Inspector richiede dati da scansioni di codice, Amazon Q utilizza la chiave KMS per decrittografare la chiave dati. Quando disabiliti la scansione del codice Lambda, Amazon Q elimina la chiave dati associata.

Autorizzazioni per la crittografia del codice con una chiave gestita dal cliente

Per la crittografia, è necessario creare una chiave KMS con [una politica](#) che includa un'istruzione che consenta ad Amazon Inspector e Amazon Q di eseguire le seguenti azioni.

- kms:Decrypt
- kms:DescribeKey
- kms:Encrypt
- kms:GenerateDataKey
- kms:GenerateDataKeyWithoutPlainText

Dichiarazione delle policy

Puoi utilizzare la seguente dichiarazione di politica durante la creazione della chiave KMS.

Note

*account-id*Sostituiscila con il tuo ID a 12 cifre Account AWS . *Region*Sostituiscilo con quello Regione AWS in cui hai abilitato Amazon Inspector e la scansione del codice Lambda. Sostituiscilo *role-ARN* con Amazon Resource Name per il tuo ruolo IAM.

```
{
  "Effect": "Allow",
  "Principal": {
    "Service": "q.amazonaws.com"
  },
  "Action": [
    "kms:Encrypt",
```

```

    "kms:Decrypt",
    "kms:GenerateDataKeyWithoutPlaintext",
    "kms:GenerateDataKey"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "kms:EncryptionContext:aws:qdeveloper:lambda-codescan-scope": "account-id"
    },
    "StringEquals": {
      "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
      "aws:SourceArn": "arn:aws:qdeveloper:Region:account-id:scans/*"
    }
  }
},
{
  "Effect": "Allow",
  "Principal": {
    "Service": "q.amazonaws.com"
  },
  "Action": "kms:DescribeKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "aws:SourceAccount": "account-id"
    },
    "ArnLike": {
      "aws:SourceArn": "arn:aws:qdeveloper:Region:account-id:scans/*"
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:GenerateDataKeyWithoutPlaintext",
    "kms:GenerateDataKey"
  ],
  "Principal": {
    "AWS": "role-ARN"
  }
},

```

```
"Resource": "*",
"Condition": {
  "StringEquals": {
    "kms:ViaService": "inspector2.Region.amazonaws.com"
  },
  "StringLike": {
    "kms:EncryptionContext:aws:qdeveloper:lambda-codescan-scope": "account-id"
  }
},
{
  "Effect": "Allow",
  "Action": [
    "kms:DescribeKey"
  ],
  "Principal": {
    "AWS": "role-ARN"
  },
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": "inspector2.Region.amazonaws.com"
    }
  }
}
```

La dichiarazione politica è formattata in JSON. Dopo aver incluso l'istruzione, rivedi la politica per assicurarti che la sintassi sia valida. Se l'istruzione è l'ultima dichiarazione della politica, inserisci una virgola dopo la parentesi di chiusura dell'istruzione precedente. Se l'istruzione è la prima dichiarazione o tra due istruzioni esistenti nella politica, inserisci una virgola dopo la parentesi che chiude la dichiarazione.

Note

Amazon Inspector non supporta più le [sovvenzioni](#) per crittografare frammenti di codice estratti dai pacchetti. Se utilizzi una politica basata sulle sovvenzioni, puoi comunque accedere ai risultati. Tuttavia, se aggiorni o reimposti la tua chiave KMS o disabiliti la scansione del codice Lambda, dovrai utilizzare la politica delle chiavi KMS descritta in questa sezione.

Se imposti, aggiorni o reimposti la chiave di crittografia per il tuo account, devi utilizzare una politica di amministratore di Amazon Inspector, ad esempio la politica AWS gestita. `AmazonInspector2FullAccess`

Configurazione della crittografia con una chiave gestita dal cliente

Per configurare la crittografia per il tuo account utilizzando una chiave gestita dal cliente, devi essere un amministratore di Amazon Inspector con le autorizzazioni descritte in [Autorizzazioni per la crittografia del codice con una chiave gestita dal cliente](#). Inoltre, avrai bisogno di una [AWS KMS chiave nella stessa AWS regione dei risultati o di una chiave multiregionale](#). Puoi utilizzare una chiave simmetrica esistente nel tuo account o creare una chiave simmetrica gestita dal cliente utilizzando la console di AWS gestione o le API. AWS KMS Per ulteriori informazioni, consulta [Creazione di chiavi di crittografia simmetriche](#) nella guida per l'utente. AWS KMS AWS KMS

Note

A partire dal 13 giugno 2025, il principale servizio nelle AWS KMS richieste registrate CloudTrail durante lo snippet encryption/decryption di codice passerà da «codeguru-reviewer» a «q».

Utilizzo dell'API Amazon Inspector per configurare la crittografia

Per impostare una chiave per la crittografia, il [UpdateEncryptionKey](#) funzionamento dell'API Amazon Inspector dopo aver effettuato l'accesso come amministratore di Amazon Inspector. Nella richiesta API, utilizza il `kmsKeyId` campo per specificare l'ARN della AWS KMS chiave che desideri utilizzare. Per `scanType` entrare `CODE` e per `resourceType` entrare `AWS_LAMBDA_FUNCTION`.

Puoi utilizzare l'[UpdateEncryptionKey](#) API per verificare la visualizzazione della AWS KMS chiave utilizzata da Amazon Inspector per la crittografia.

Note

Se tenti di utilizzare `GetEncryptionKey` quando non hai impostato una chiave gestita dal cliente, l'operazione restituisce un `ResourceNotFoundException` errore, il che significa che per la crittografia viene utilizzata una chiave di AWS proprietà.

Se elimini la chiave o ne modifichi la politica per negare l'accesso ad Amazon Inspector o Amazon Q, non sarai in grado di accedere ai risultati delle vulnerabilità del codice e la scansione del codice Lambda non riuscirà per il tuo account.

Puoi utilizzare `ResetEncryptionKey` per riprendere a utilizzare una chiave AWS proprietaria per crittografare il codice estratto come parte dei risultati di Amazon Inspector.

Crittografia dei dati in transito

AWS crittografa tutti i dati in transito tra sistemi AWS interni e altri AWS servizi. AWS Systems Manager raccoglie i dati di telemetria dalle istanze EC2 di proprietà del cliente a cui vengono inviati AWS tramite un canale protetto da Transport Layer Security (TLS) per la valutazione. I risultati delle scansioni delle funzioni Amazon ECR e AWS Lambda inviati a Security Hub CSPM vengono crittografati utilizzando un canale TLS-protected. Per ulteriori informazioni, vedere [Protezione dei dati in Systems Manager](#) per capire come SSM crittografa i dati in transito.

Identity and Access Management per Amazon Inspector

AWS Identity and Access Management (IAM) è uno strumento Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuare l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse Amazon Inspector. IAM è uno strumento Servizio AWS che puoi utilizzare senza costi aggiuntivi.

Argomenti

- [Destinatari](#)
- [Autenticazione con identità](#)
- [Gestione dell'accesso tramite policy](#)
- [Come funziona Amazon Inspector con IAM](#)
- [Identity-based esempi di policy per Amazon Inspector](#)
- [AWS politiche gestite per Amazon Inspector](#)
- [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#)
- [Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Inspector](#)

Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia in base al tuo ruolo:

- Utente del servizio: richiedi le autorizzazioni all'amministratore se non riesci ad accedere alle funzionalità (consulta [Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Inspector](#))
- Amministratore del servizio: determina l'accesso degli utenti e invia le richieste di autorizzazione (consulta [Come funziona Amazon Inspector con IAM](#))
- Amministratore IAM: scrivi policy per gestire l'accesso (consulta [Identity-based esempi di policy per Amazon Inspector](#))

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi autenticarti come utente IAM o assumendo un ruolo IAM. Utente root dell'account AWS

Puoi accedere come identità federata utilizzando credenziali provenienti da una fonte di identità come AWS IAM Identity Center (IAM Identity Center), autenticazione Single Sign-On o credenziali. Google/Facebook Per ulteriori informazioni sull'accesso, consulta [Come accedere all' Account AWS](#) nella Guida per l'utente di Accedi ad AWS .

Per l'accesso programmatico, AWS fornisce un SDK e una CLI per firmare crittograficamente le richieste. Per ulteriori informazioni, consulta [AWS Signature Version 4 per le richieste API](#) nella Guida per l'utente di IAM.

Account AWS utente root

Quando si crea un Account AWS, si inizia con un'identità di accesso denominata utente Account AWS root che ha accesso completo a tutte Servizi AWS le risorse. Consigliamo vivamente di non utilizzare l'utente root per le attività quotidiane. Per le attività che richiedono le credenziali dell'utente root, consulta [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente IAM.

Identità federata

Come procedura ottimale, richiedi agli utenti umani di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente della directory aziendale, del provider di identità Web o Directory Service che accede Servizi AWS utilizzando le credenziali di una fonte di identità. Le identità federate assumono ruoli che forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, si consiglia di utilizzare AWS IAM Identity Center. Per ulteriori informazioni, consulta [Che cos'è il Centro identità IAM?](#) nella Guida per l'utente di AWS IAM Identity Center .

Utenti e gruppi IAM

Un [utente IAM](#) è una identità che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ti consigliamo di utilizzare credenziali temporanee invece di utenti IAM con credenziali a lungo termine. Per ulteriori informazioni, consulta [Richiedere agli utenti umani di utilizzare la federazione con un provider di identità per accedere AWS utilizzando credenziali temporanee](#) nella Guida per l'utente IAM.

Un [gruppo IAM](#) specifica una raccolta di utenti IAM e semplifica la gestione delle autorizzazioni per gestire gruppi di utenti di grandi dimensioni. Per ulteriori informazioni, consulta [Casi d'uso per utenti IAM](#) nella Guida per l'utente di IAM.

Ruoli IAM

Un [ruolo IAM](#) è un'identità con autorizzazioni specifiche che fornisce credenziali temporanee. Puoi assumere un ruolo [passando da un ruolo utente a un ruolo IAM \(console\)](#) o chiamando un'operazione AWS CLI o AWS API. Per ulteriori informazioni, consulta [Metodi per assumere un ruolo](#) nella Guida per l'utente di IAM.

I ruoli IAM sono utili per l'accesso degli utenti federati, le autorizzazioni utente IAM temporanee, l'accesso multi-account, l'accesso multi-servizio e le applicazioni in esecuzione su Amazon EC2. Per maggiori informazioni, consultare [Accesso a risorse multi-account in IAM](#) nella Guida per l'utente IAM.

Gestione dell'accesso tramite policy

Puoi controllare l'accesso AWS creando policy e associandole a AWS identità o risorse. Una policy definisce le autorizzazioni quando è associata a un'identità o a una risorsa. AWS valuta queste politiche quando un preside effettua una richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per maggiori informazioni sui documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente IAM.

Utilizzando le policy, gli amministratori specificano chi ha accesso a cosa definendo quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Un amministratore IAM crea le policy IAM e le aggiunge ai ruoli, che gli utenti possono quindi assumere. Le policy IAM definiscono le autorizzazioni indipendentemente dal metodo utilizzato per eseguirle.

Identity-based politiche

Identity-based le politiche sono documenti di policy sulle autorizzazioni JSON che alleggi a un'identità (utente, gruppo o ruolo). Tali policy controllano le operazioni autorizzate per l'identità, nonché le risorse e le condizioni in cui possono essere eseguite. Per informazioni su come creare una policy basata su identità, consultare [Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente](#) nella Guida per l'utente IAM.

Identity-based le politiche possono essere politiche in linea (incorporate direttamente in una singola identità) o politiche gestite (politiche autonome collegate a più identità). Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scegliere tra policy gestite e policy in linea](#) nella Guida per l'utente di IAM.

Resource-based politiche

Resource-based le politiche sono documenti di policy JSON allegati a una risorsa. Gli esempi includono le policy di trust dei ruoli IAM e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. In una policy basata sulle risorse è obbligatorio [specificare un'entità principale](#).

Resource-based le politiche sono politiche in linea che si trovano in quel servizio. Non è possibile utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi che possono impostare le autorizzazioni massime concesse dai tipi di policy più comuni:

- Limiti delle autorizzazioni: imposta il numero massimo di autorizzazioni che una policy basata su identità ha la possibilità di concedere a un'entità IAM. Per ulteriori informazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente di IAM.

- Policy di controllo dei servizi (SCP): specifica il numero massimo di autorizzazioni per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. Per ulteriori informazioni, consultare [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations .
- Policy di controllo delle risorse (RCP): imposta le autorizzazioni massime disponibili per le risorse degli account. Per ulteriori informazioni, consulta [Policy di controllo delle risorse \(RCP\)](#) nella Guida per l'utente di AWS Organizations .
- Policy di sessione: policy avanzate passate come parametro quando si crea una sessione temporanea per un ruolo o un utente federato. Per maggiori informazioni, consultare [Policy di sessione](#) nella Guida per l'utente IAM.

Più tipi di policy

Quando a una richiesta si applicano più tipi di politiche, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire o meno una richiesta quando sono coinvolti più tipi di policy, consulta [Logica di valutazione delle policy](#) nella IAM User Guide.

Come funziona Amazon Inspector con IAM

Prima di utilizzare IAM per gestire l'accesso ad Amazon Inspector, scopri quali funzionalità IAM sono disponibili per l'uso con Amazon Inspector.

Funzionalità IAM che puoi utilizzare con Amazon Inspector

Funzionalità IAM	Supporto per Amazon Inspector
Identity-based politiche	Sì
Resource-based politiche	No
Operazioni di policy	Sì
Risorse relative alle policy	Sì
Chiavi di condizione della policy (specifica del servizio)	Sì
Liste di controllo degli accessi (ACL)	No
ABAC (tag nelle policy)	Parziale

Funzionalità IAM	Supporto per Amazon Inspector
Credenziali temporanee	Sì
Autorizzazioni del principale	Sì
Ruoli di servizio	No
Service-linked ruoli	Sì

Per avere una panoramica generale del funzionamento di Amazon Inspector e Servizi AWS altri con la maggior parte delle funzionalità IAM, [Servizi AWS consulta la sezione dedicata alla compatibilità con IAM](#) nella IAM User Guide.

Identity-based politiche per Amazon Inspector

Supporta le policy basate sull'identità: sì

Identity-based le policy sono documenti relativi alle policy in materia di autorizzazioni JSON che puoi allegare a un'identità, ad esempio un utente IAM, un gruppo di utenti o un ruolo. Tali policy definiscono le operazioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente](#) nella Guida per l'utente di IAM.

Con le policy basate sull'identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta [Guida di riferimento agli elementi delle policy JSON IAM](#) nella Guida per l'utente IAM.

Identity-based esempi di policy per Amazon Inspector

Per visualizzare esempi di politiche basate sull'identità di Amazon Inspector, consulta. [Identity-based esempi di policy per Amazon Inspector](#)

Resource-based politiche all'interno di Amazon Inspector

Supporta le policy basate su risorse: no

Resource-based le politiche sono documenti di policy JSON allegati a una risorsa. Esempi di policy basate sulle risorse sono le policy di attendibilità dei ruoli IAM e le policy di bucket Amazon S3. Nei

servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le operazioni che un principale può eseguire su tale risorsa e a quali condizioni. In una policy basata sulle risorse è obbligatorio [specificare un'entità principale](#). I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, è possibile specificare un intero account o entità IAM in un altro account come entità principale in una policy basata sulle risorse. Per ulteriori informazioni, consulta [Accesso a risorse multi-account in IAM](#) nella Guida per l'utente IAM.

Azioni politiche per Amazon Inspector

Supporta le operazioni di policy: si

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento `Action` di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso in una policy. Includere le operazioni in una policy per concedere le autorizzazioni di eseguire l'operazione associata.

Per visualizzare un elenco delle azioni di Amazon Inspector, consulta [Azioni definite da Amazon Inspector](#) nel Service Authorization Reference.

Le azioni politiche in Amazon Inspector utilizzano il seguente prefisso prima dell'azione:

```
inspector2
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

```
"Action": [  
  "inspector2:action1",  
  "inspector2:action2"  
]
```

Per visualizzare esempi di politiche basate sull'identità di Amazon Inspector, consulta. [Identity-based esempi di policy per Amazon Inspector](#)

Risorse relative alle policy per Amazon Inspector

Supporta le risorse relative alle policy: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Come best practice, specifica una risorsa utilizzando il suo [nome della risorsa Amazon \(ARN\)](#). Per le azioni che non supportano le autorizzazioni a livello di risorsa, si utilizza un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

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

Per visualizzare un elenco dei tipi di risorse di Amazon Inspector e dei relativi ARN, consulta [Risorse definite da Amazon Inspector](#) nel Service Authorization Reference. Per sapere con quali azioni è possibile specificare l'ARN di ogni risorsa, consulta [Azioni definite da Amazon Inspector](#).

Per visualizzare esempi di politiche basate sull'identità di Amazon Inspector, consulta [Identity-based esempi di policy per Amazon Inspector](#)

Chiavi relative alle condizioni delle politiche per Amazon Inspector

Supporta le chiavi di condizione delle policy specifiche del servizio: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento `Condition` specifica quando le istruzioni vengono eseguite in base a criteri definiti. È possibile compilare espressioni condizionali che utilizzano [operatori di condizione](#), ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta. Per visualizzare tutte le chiavi di condizione AWS globali, consulta le chiavi di [contesto delle condizioni AWS globali nella Guida](#) per l'utente IAM.

Per visualizzare un elenco dei codici di condizione di Amazon Inspector, consulta [Condition keys for Amazon Inspector](#) nel Service Authorization Reference. Per sapere con quali azioni e risorse puoi utilizzare una chiave di condizione, consulta [Azioni definite da Amazon Inspector](#).

Per visualizzare esempi di politiche basate sull'identità di Amazon Inspector, consulta [Identity-based esempi di policy per Amazon Inspector](#)

ACL in Amazon Inspector

Supporta le ACL: no

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

ABAC con Amazon Inspector

Supporta ABAC (tag nelle policy): parzialmente

Attribute-based il controllo degli accessi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base ad attributi chiamati tag. È possibile allegare tag a entità e AWS risorse IAM, quindi progettare politiche ABAC per consentire le operazioni quando il tag del principale corrisponde al tag sulla risorsa.

Per controllare l'accesso basato su tag, fornire informazioni sui tag nell'[elemento condizione](#) di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Sì. Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per maggiori informazioni su ABAC, consulta [Definizione delle autorizzazioni con autorizzazione ABAC](#) nella Guida per l'utente di IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consulta [Utilizzo del controllo degli accessi basato su attributi \(ABAC\)](#) nella Guida per l'utente di IAM.

Utilizzo di credenziali temporanee con Amazon Inspector

Supporta le credenziali temporanee: sì

Le credenziali temporanee forniscono un accesso a breve termine alle AWS risorse e vengono create automaticamente quando si utilizza la federazione o si cambia ruolo. AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per

ulteriori informazioni, consulta [Credenziali di sicurezza temporanee in IAM](#) e [Servizi AWS compatibili con IAM](#) nella Guida per l'utente IAM.

Cross-service autorizzazioni principali per Amazon Inspector

Supporta l'inoltro delle sessioni di accesso (FAS): sì

Le sessioni di accesso inoltrato (FAS) utilizzano le autorizzazioni del principale che chiama un Servizio AWS, combinate con la richiesta di effettuare richieste Servizio AWS ai servizi downstream. Per i dettagli delle policy relative alle richieste FAS, consulta [Forward access sessions](#).

Ruoli di servizio per Amazon Inspector

Supporta i ruoli di servizio: no

Un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Create a role to delegate permissions to an Servizio AWS](#) nella Guida per l'utente IAM.

Warning

La modifica delle autorizzazioni per un ruolo di servizio potrebbe interrompere la funzionalità di Amazon Inspector. Modifica i ruoli di servizio solo quando Amazon Inspector fornisce indicazioni in tal senso.

Service-linked ruoli per Amazon Inspector

Supporta i ruoli collegati ai servizi: Sì

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo di eseguire un'azione per conto dell'utente. Service-linked i ruoli vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati al servizio, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati al servizio, consulta [Servizi AWS supportati da IAM](#). Trova un servizio nella tabella che include un servizio Yes nella colonna del Service-linked ruolo. Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Identity-based esempi di policy per Amazon Inspector

Per impostazione predefinita, gli utenti e i ruoli non sono autorizzati a creare o modificare risorse Amazon Inspector. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta [Creazione di policy IAM \(console\)](#) nella Guida per l'utente di IAM.

Per dettagli sulle azioni e sui tipi di risorse definiti da Amazon Inspector, incluso il formato degli ARN per ciascun tipo di risorsa, consulta [Azioni, risorse e chiavi di condizione per Amazon Inspector](#) nel Service Authorization Reference.

Argomenti

- [Best practice per le policy](#)
- [Utilizzo della console Amazon Inspector](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)
- [Consenti l'accesso in sola lettura a tutte le risorse Amazon Inspector](#)
- [Consenti l'accesso completo a tutte le risorse di Amazon Inspector](#)

Best practice per le policy

Identity-based le politiche determinano se qualcuno può creare, accedere o eliminare risorse Amazon Inspector nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l'Account AWS. Quando si creano o modificano policy basate sull'identità, seguire queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche AWS gestite che concedono le autorizzazioni per molti casi d'uso comuni. Sono disponibili nel tuo Account AWS. Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai clienti AWS specifiche per i tuoi casi d'uso. Per maggiori informazioni, consulta [Policy gestite da AWS](#) o [Policy gestite da AWS per le funzioni dei processi](#) nella Guida per l'utente di IAM.
- Applicazione delle autorizzazioni con privilegio minimo - Quando si impostano le autorizzazioni con le policy IAM, concedere solo le autorizzazioni richieste per eseguire un'attività. È possibile farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche,

note anche come autorizzazioni con privilegio minimo. Per maggiori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta [Policy e autorizzazioni in IAM](#) nella Guida per l'utente di IAM.

- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso - Per limitare l'accesso ad azioni e risorse è possibile aggiungere una condizione alle policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio CloudFormation. Per maggiori informazioni, consultare la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.
- Utilizzo dello strumento di analisi degli accessi IAM per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali - Lo strumento di analisi degli accessi IAM convalida le policy nuove ed esistenti in modo che aderiscano al linguaggio (JSON) della policy IAM e alle best practice di IAM. Lo strumento di analisi degli accessi IAM offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per maggiori informazioni, consultare [Convalida delle policy per il Sistema di analisi degli accessi IAM](#) nella Guida per l'utente di IAM.
- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungere le condizioni MFA alle policy. Per maggiori informazioni, consultare [Protezione dell'accesso API con MFA](#) nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consulta [Best practice di sicurezza in IAM](#) nella Guida per l'utente di IAM.

Utilizzo della console Amazon Inspector

Per accedere alla console Amazon Inspector, devi disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle risorse Amazon Inspector presenti nel tuo Account AWS. Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario consentire autorizzazioni minime per la console agli utenti che effettuano chiamate solo verso AWS CLI o l'API. AWS AI contrario, è opportuno concedere l'accesso solo alle azioni che corrispondono all'operazione API che stanno cercando di eseguire.

Per garantire che utenti e ruoli possano continuare a utilizzare la console Amazon Inspector, collega anche Amazon *ConsoleAccess* Inspector *ReadOnly* AWS o la policy gestita alle entità. Per maggiori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente di IAM.

Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono collegate alla relativa identità utente. Questa policy include le autorizzazioni per completare questa azione sulla console o utilizzando programmaticamente l'API o. AWS CLI AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}
```

Consenti l'accesso in sola lettura a tutte le risorse Amazon Inspector

Questo esempio mostra una policy che consente l'accesso in sola lettura a tutte le risorse di Amazon Inspector.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "inspector2:Describe*",
        "inspector2:Get*",
        "inspector2:BatchGet*",
        "inspector2:List*"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "organizations:ListDelegatedAdministrators",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganizationalUnit",
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization"
      ],
      "Resource": "*"
    }
  ]
}
```

Consenti l'accesso completo a tutte le risorse di Amazon Inspector

Questo esempio mostra una policy che consente l'accesso completo a tutte le risorse di Amazon Inspector.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "inspector2:*",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:CreateServiceLinkedRole",
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "iam:AWSServiceName": "inspector2.amazonaws.com"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganizationalUnit",
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization"
      ],
      "Resource": "*"
    }
  ]
}
```

AWS politiche gestite per Amazon Inspector

Una politica AWS gestita è una politica autonoma creata e amministrata da AWS. Le politiche gestite sono progettate per fornire autorizzazioni per molti casi d'uso comuni, in modo da poter iniziare ad assegnare autorizzazioni a utenti, gruppi e ruoli.

Tieni presente che le policy AWS gestite potrebbero non concedere le autorizzazioni con il privilegio minimo per i tuoi casi d'uso specifici, poiché sono disponibili per tutti i clienti. AWS consiglia pertanto di ridurre ulteriormente le autorizzazioni definendo [policy gestite dal cliente](#) specifiche per i propri casi d'uso.

Non è possibile modificare le autorizzazioni definite nelle politiche gestite. Se AWS aggiorna le autorizzazioni definite in una politica AWS gestita, l'aggiornamento ha effetto su tutte le identità principali (utenti, gruppi e ruoli) a cui è associata la politica. AWS è più probabile che aggiorni una policy AWS gestita quando nel Servizio AWS viene lanciata una nuova o quando diventano disponibili nuove operazioni API per i servizi esistenti.

Per ulteriori informazioni, consultare [Policy gestite da AWS](#) nella Guida per l'utente di IAM.

AWS politica gestita: AmazonInspector2FullAccess_v2

È possibile allegare la policy AmazonInspector2FullAccess_v2 alle identità IAM.

Questa politica garantisce l'accesso completo ad Amazon Inspector e l'accesso ad altri servizi correlati.

Dettagli delle autorizzazioni

Questa policy include le seguenti autorizzazioni:

- `inspector2`— Consente l'accesso completo alle API di Amazon Inspector.
- `codeguru-security`— Consente agli amministratori di recuperare i risultati di sicurezza e le impostazioni di configurazione per un account.
- `iam`— Consente ad Amazon Inspector di creare ruoli collegati ai servizi e.

`AWSServiceRoleForAmazonInspector2`

`AWSServiceRoleForAmazonInspector2Agentless`

`AWSServiceRoleForAmazonInspector2` è necessario per Amazon Inspector per eseguire operazioni come il recupero di informazioni sulle istanze Amazon EC2, i repository Amazon ECR

e le immagini dei contenitori Amazon ECR. È inoltre necessario decrittografare gli snapshot di Amazon EBS crittografati con chiavi. AWS KMS Per ulteriori informazioni, consulta [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#).

- `organizations— AllowServicePrincipalBasedAccessToOrganizationApis` consente solo ai responsabili del servizio di creare ruoli collegati ai servizi per Account AWS, registrarne uno Account AWS come amministratore delegato per un'organizzazione ed elencare gli amministratori delegati di un'organizzazione. `AllowOrganizationalBasedAccessToOrganizationApis` consente al titolare della polizza di recuperare informazioni, in particolare ARN a livello di risorsa, su un'unità organizzativa. `AllowAccountsBasedAccessToOrganizationApis` consente al titolare della polizza di recuperare informazioni, in particolare ARN a livello di risorsa, su un Account AWS. `AWSAAllowAccessToOrganizationApis` consente al titolare della polizza di visualizzare informazioni Servizi AWS integrate con un'organizzazione e un'organizzazione. La politica consente di elencare le politiche organizzative di Inspector filtrando in base ai tipi di criteri di Inspector, di visualizzare le politiche delle risorse di delega stabilite dagli account di gestione e di visualizzare le politiche di Inspector efficaci applicate agli account.

Note

Amazon Inspector non esegue più scansioni CodeGuru Lambda. AWS interromperà il supporto il 20 novembre 2025. Per ulteriori informazioni, consulta [Fine del supporto per la CodeGuru sicurezza](#). Amazon Inspector ora utilizza Amazon Q per eseguire scansioni Lambda e non richiede le autorizzazioni descritte in questa sezione.

Per esaminare le autorizzazioni per questa politica, consulta [AmazonInspector2FullAccess_v2](#) nella Managed Policy Reference Guide.AWS

AWS politica gestita: `AWSInspector2OrganizationsAccess`

È possibile allegare la policy `AWSInspector2OrganizationsAccess` alle identità IAM.

Questa politica concede le autorizzazioni amministrative per abilitare e gestire Amazon Inspector per un'organizzazione in AWS Organizations. Le autorizzazioni per questa politica consentono all'account di gestione dell'organizzazione di designare l'account amministratore delegato per Amazon Inspector. Consentono inoltre all'account amministratore delegato di abilitare gli account dell'organizzazione come account membro.

Questa politica fornisce solo le autorizzazioni per. AWS Organizations L'account di gestione dell'organizzazione e l'account amministratore delegato richiedono inoltre le autorizzazioni per le azioni associate. Queste autorizzazioni possono essere concesse utilizzando la politica gestita `AmazonInspector2FullAccess_v2`.

Dettagli delle autorizzazioni

Questa policy include le seguenti autorizzazioni:

- `organizations:ListAccounts`— Consente ai responsabili di recuperare l'elenco degli account che fanno parte di un'organizzazione.
- `organizations:DescribeOrganization`— Consente ai dirigenti di recuperare informazioni sull'organizzazione.
- `organizations:ListRoots`— Consente ai dirigenti di elencare la radice di un'organizzazione.
- `organizations:ListDelegatedAdministrators`— Consente ai dirigenti di elencare l'amministratore delegato di un'organizzazione.
- `organizations:ListAWSServiceAccessForOrganization`— Consente ai dirigenti di elencare le informazioni utilizzate da un' Servizi AWS organizzazione.
- `organizations:ListOrganizationalUnitsForParent`— Consente ai responsabili di elencare le unità organizzative (OU) secondarie di un'unità organizzativa principale.
- `organizations:ListAccountsForParent`— Consente ai responsabili di elencare gli account secondari di un'unità organizzativa principale.
- `organizations:ListParents`— Elenca le unità organizzative o principali (OU) che fungono da elemento principale dell'unità organizzativa o dell'account figlio specificato.
- `organizations:DescribeAccount`: consente ai principali di recuperare informazioni su un account nell'organizzazione.
- `organizations:DescribeOrganizationalUnit`— Consente ai responsabili di recuperare informazioni su un'unità organizzativa dell'organizzazione.
- `organizations:ListPolicies`— Recupera l'elenco di tutte le politiche di un'organizzazione di un tipo specificato.
- `organizations:ListPoliciesForTarget`— Elenca le politiche direttamente collegate alla radice, all'unità organizzativa (OU) o all'account di destinazione specificati.
- `organizations:ListTargetsForPolicy`— Elenca tutte le radici, le unità organizzative (OU) e gli account a cui è associata la politica specificata.

- `organizations:DescribeResourcePolicy`— Recupera informazioni su una politica delle risorse.
- `organizations:EnableAWSServiceAccess`— Consente ai presidi di abilitare l'integrazione con Organizations.
- `organizations:RegisterDelegatedAdministrator`— Consente ai responsabili di designare l'account amministratore delegato.
- `organizations:DeregisterDelegatedAdministrator`— Consente ai responsabili di rimuovere l'account di amministratore delegato.
- `organizations:DescribePolicy`— Recupera informazioni su una politica.
- `organizations:DescribeEffectivePolicy`— Restituisce il contenuto della politica effettiva per il tipo di politica e l'account specificati.
- `organizations>CreatePolicy`— Crea una politica di un tipo specifico che è possibile allegare a una radice, a un'unità organizzativa (OU) o a un individuo Account AWS.
- `organizations:UpdatePolicy`— Aggiorna una politica esistente con un nuovo nome, descrizione o contenuto.
- `organizations>DeletePolicy`— Elimina la politica specificata dall'organizzazione.
- `organizations:AttachPolicy`— Associa una policy a una root, a un'unità organizzativa (OU) o a un account individuale.
- `organizations:DetachPolicy`— Scollega una politica da una radice, un'unità organizzativa (OU) o un account di destinazione.
- `organizations:EnablePolicyType`— Abilita un tipo di policy in una radice.
- `organizations:DisablePolicyType`— Disattiva un tipo di politica organizzativa in una radice.
- `organizations:TagResource`— Aggiunge uno o più tag a una risorsa specificata.
- `organizations:UntagResource`— Rimuove tutti i tag con le chiavi specificate da una risorsa specificata.
- `organizations:ListTagsForResource`— Elenca i tag allegati a una risorsa specificata.

Per esaminare le autorizzazioni relative a questa policy, consulta [AWSInspector2OrganizationsAccess](#) la AWS Managed Policy Reference Guide.

AWS politica gestita: AmazonInspector2FullAccess

È possibile allegare la policy AmazonInspector2FullAccess alle identità IAM.

Questa politica concede autorizzazioni amministrative che consentono l'accesso completo ad Amazon Inspector.

⚠ Important

[Per una maggiore sicurezza e autorizzazioni restrittive per i principali servizi di Inspector 2, si consiglia di utilizzare `_v2. AmazonInspector2FullAccess`](#)

Dettagli delle autorizzazioni

Questa policy include le seguenti autorizzazioni:

- `inspector2`— Consente l'accesso completo alle funzionalità di Amazon Inspector.
- `iam`— Consente ad Amazon Inspector di creare ruoli collegati ai servizi e.
`AWSServiceRoleForAmazonInspector2`
`AWSServiceRoleForAmazonInspector2Agentless`
`AWSServiceRoleForAmazonInspector2` è necessario per consentire ad Amazon Inspector di eseguire operazioni come il recupero di informazioni sulle istanze Amazon EC2, i repository Amazon ECR e le immagini dei contenitori. È inoltre necessario che Amazon Inspector analizzi la tua rete VPC e descriva gli account associati alla tua organizzazione.
`AWSServiceRoleForAmazonInspector2Agentless` è necessario per consentire ad Amazon Inspector di eseguire operazioni, come il recupero di informazioni sulle istanze Amazon EC2 e sugli snapshot di Amazon EBS. È inoltre necessario decrittografare gli snapshot di Amazon EBS crittografati con chiavi AWS KMS. Per ulteriori informazioni, consulta [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#).
- `organizations`— Consente agli amministratori di utilizzare Amazon Inspector per un'organizzazione in AWS Organizations. Quando [attivi l'accesso affidabile](#) per Amazon Inspector in AWS Organizations, i membri dell'account amministratore delegato possono gestire le impostazioni e visualizzare i risultati in tutta l'organizzazione.
- `codeguru-security`— Consente agli amministratori di utilizzare Amazon Inspector per recuperare frammenti di codice informativo e modificare le impostazioni di crittografia per il codice archiviato da Security. CodeGuru. Per ulteriori informazioni, consulta [Crittografia inattiva per il codice contenuto nei tuoi risultati](#).

Per esaminare le autorizzazioni relative a questa politica, consulta la Managed Policy Reference [AmazonInspector2FullAccess](#)Guide.AWS

AWS politica gestita: AmazonInspector2ReadOnlyAccess

È possibile allegare la policy AmazonInspector2ReadOnlyAccess alle identità IAM.

Questa politica concede autorizzazioni che consentono l'accesso in sola lettura ad Amazon Inspector.

Dettagli delle autorizzazioni

Questa policy include le seguenti autorizzazioni:

- `inspector2`— Consente l'accesso in sola lettura alle funzionalità di Amazon Inspector.
- `organizations`— Consente di visualizzare i dettagli sulla copertura di Amazon Inspector per un'organizzazione. AWS Organizations Inoltre, consente la visualizzazione delle politiche organizzative di Inspector tramite il filtraggio in base `ListPolicies` ai tipi di policy di Inspector, la visualizzazione delle politiche relative alle risorse di delega `DescribeResourcePolicy` e la visualizzazione delle politiche di Inspector efficaci applicate agli account tramite `DescribeEffectivePolicy` Ciò consente agli utenti di comprendere l'attivazione centralizzata degli ispettori stabilita attraverso le politiche organizzative senza la possibilità di modificarle.
- `codeguru-security`— Consente di recuperare frammenti di codice da Security. CodeGuru Consente inoltre di visualizzare le impostazioni di crittografia per il codice memorizzato in CodeGuru Security.

Per esaminare le autorizzazioni relative a questa politica, consulta [AmazonInspector2ReadOnlyAccess](#)la AWS Managed Policy Reference Guide.

AWS politica gestita: AmazonInspector2ManagedCisPolicy

Puoi collegare la policy AmazonInspector2ManagedCisPolicy anche alle tue entità IAM. Questa policy deve essere associata a un ruolo che concede le autorizzazioni alle istanze Amazon EC2 per eseguire scansioni CIS dell'istanza. Puoi utilizzare un ruolo IAM per gestire le credenziali temporanee per le applicazioni in esecuzione su un'istanza EC2 e che effettuano richieste API. AWS CLI AWS Ciò è preferibile all'archiviazione delle chiavi di accesso nell'istanza EC2. Per assegnare un AWS ruolo a un'istanza EC2 e renderlo disponibile per tutte le sue applicazioni, crei un profilo di istanza

collegato all'istanza. Un profilo dell'istanza contiene il ruolo e consente ai programmi in esecuzione sull'istanza EC2 di ottenere le credenziali temporanee. Per ulteriori informazioni, consulta [Utilizzare un ruolo IAM per concedere autorizzazioni alle applicazioni che eseguono istanze Amazon EC2](#) nella Guida per l'utente IAM.

Dettagli delle autorizzazioni

Questa policy include le seguenti autorizzazioni:

- `inspector2`— Consente l'accesso alle azioni utilizzate per eseguire scansioni CIS.

Per esaminare le autorizzazioni relative a questa politica, consulta [AmazonInspector2ManagedCisPolicy](#) la AWS Managed Policy Reference Guide.

AWS politica gestita: `AmazonInspector2ServiceRolePolicy`

Non è possibile allegare la policy `AmazonInspector2ServiceRolePolicy` alle entità IAM. Questa policy è associata a un ruolo collegato al servizio che consente ad Amazon Inspector di eseguire azioni per tuo conto. Per ulteriori informazioni, consulta [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#).

AWS politica gestita: `AmazonInspector2AgentlessServiceRolePolicy`

Non è possibile allegare la policy `AmazonInspector2AgentlessServiceRolePolicy` alle entità IAM. Questa policy è associata a un ruolo collegato al servizio che consente ad Amazon Inspector di eseguire azioni per tuo conto. Per ulteriori informazioni, consulta [Utilizzo di ruoli collegati ai servizi per Amazon Inspector](#).

AWS politica gestita: `AmazonInspector2ManagedTelemetryPolicy`

Puoi collegare la policy `AmazonInspector2ManagedTelemetryPolicy` anche alle tue entità IAM. Questa politica concede le autorizzazioni per le operazioni di telemetria di Amazon Inspector, consentendo al servizio di raccogliere e trasmettere i dati di inventario dei pacchi per la scansione delle vulnerabilità.

Dettagli delle autorizzazioni

Questa policy include le seguenti autorizzazioni:

- `inspector2-telemetry`— Consente l'accesso alle azioni per la trasmissione dei dati relativi all'inventario dei pacchetti.

Per visualizzare ulteriori dettagli sulla policy, inclusa la versione più recente del documento di policy JSON, consulta [AmazonInspector2ManagedTelemetryPolicy](#) nella Guida di riferimento alle policy gestite da AWS .

Amazon Inspector si aggiorna a AWS policy gestite

Visualizza i dettagli sugli aggiornamenti delle politiche AWS gestite per Amazon Inspector da quando questo servizio ha iniziato a tracciare queste modifiche. Per ricevere avvisi automatici sulle modifiche a questa pagina, iscriviti al feed RSS nella pagina della cronologia dei documenti di Amazon [Inspector](#).

Modifica	Descrizione	Data
AWSInspector2OrganizationsAccess : nuova policy	Amazon Inspector ha aggiunto una nuova policy gestita che concede le autorizzazioni necessarie per abilitare e gestire Amazon Inspector tramite policy. AWS Organizations	3 marzo 2026
AmazonInspector2ManagedTelemetryPolicy : nuova policy	Amazon Inspector ha aggiunto una nuova policy gestita che concede le autorizzazioni per le operazioni di telemetria di Amazon Inspector, consentendo al servizio di raccogliere e trasmettere i dati di inventario dei pacchi per la scansione delle vulnerabilità.	5 febbraio 2026
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto una nuova autorizzazione che consente ad Amazon Inspector di descrivere i	3 febbraio 2026

Modifica	Descrizione	Data
	<p>metadati del firewall per l'analisi della raggiungibilità della rete. Inoltre, Amazon Inspector ha aggiunto un ulteriore ambito delle risorse per consentire ad Amazon Inspector di creare, aggiornare e avviare associazioni SSM con documenti SSM. <code>AWS-ConfigureAWSPackage</code></p>	
<p>AmazonInspector2FullAccess_v2 e AmazonInspector2ReadOnlyAccess— Aggiornamenti alle politiche esistenti</p>	<p>Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ai titolari delle polizze di visualizzare le politiche organizzative e le configurazioni di delega di Inspector. Ciò supporta la gestione e la visibilità centralizzate dell'abilitazione di Inspector tramite policy AWS Organizations .</p>	<p>14 novembre 2025</p>
<p>AmazonInspector2ServiceRolePolicy: aggiornamenti a una policy esistente</p>	<p>Amazon Inspector ha aggiunto nuove autorizzazioni che consentono alla AWS Organizations policy di Amazon Inspector di imporre l'attivazione e la disabilitazione di Amazon Inspector.</p>	<p>10 novembre 2025</p>

Modifica	Descrizione	Data
AmazonInspector2FullAccess_v2 — Nuova politica	Amazon Inspector ha aggiunto una nuova policy gestita che fornisce l'accesso completo ad Amazon Inspector e l'accesso ad altri servizi correlati.	3 luglio 2025
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto una nuova autorizzazione che consente ad Amazon Inspector di descrivere indirizzi IP e gateway Internet.	29 aprile 2025
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono l'accesso in sola lettura alle azioni di Amazon ECS e Amazon EKS.	25 marzo 2025
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di restituire i tag di funzione. AWS Lambda	31 luglio 2024
AmazonInspector2FullAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto autorizzazioni che consentono ad Amazon Inspector di creare il ruolo collegato al servizio. <code>AWSServiceRoleForAmazonInspector2Agentless</code> Ciò consente agli utenti di eseguire scansioni basate su agenti e scansioni senza agenti quando abilitano Amazon Inspector.	24 aprile 2024

Modifica	Descrizione	Data
AmazonInspector2ManagedCisPolicy : nuova policy	Amazon Inspector ha aggiunto una nuova policy gestita che puoi utilizzare come parte di un profilo di istanza per consentire le scansioni CIS su un'istanza.	23 gennaio 2024
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di avviare scansioni CIS sulle istanze di destinazione.	23 gennaio 2024
AmazonInspector2AgentlessServiceRolePolicy : nuova policy	Amazon Inspector ha aggiunto una nuova politica relativa ai ruoli collegati ai servizi per consentire la scansione senza agenti dell'istanza EC2.	27 novembre 2023
AmazonInspector2ReadOnlyAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di sola lettura di recuperare i dettagli di vulnerability intelligence per rilevare le vulnerabilità dei pacchetti.	22 settembre 2023

Modifica	Descrizione	Data
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di scansionare le configurazioni di rete delle istanze Amazon EC2 che fanno parte dei gruppi target Elastic Load Balancing.	31 agosto 2023
AmazonInspector2ReadOnlyAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di sola lettura di esportare Software Bill of Materials (SBOM) per le proprie risorse.	29 giugno 2023
AmazonInspector2ReadOnlyAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di sola lettura di recuperare i dettagli delle impostazioni di crittografia per i risultati della scansione del codice Lambda per il proprio account.	13 giugno 2023
AmazonInspector2FullAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di configurare una chiave KMS gestita dal cliente per crittografare il codice nei risultati della scansione del codice Lambda.	13 giugno 2023

Modifica	Descrizione	Data
AmazonInspector2ReadOnlyAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di sola lettura di recuperare i dettagli dello stato e dei risultati della scansione del codice Lambda per il proprio account.	02 maggio 2023
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di creare canali AWS CloudTrail collegati ai servizi nel tuo account quando attivi la scansione Lambda. Ciò consente ad Amazon Inspector di monitorare e CloudTrail gli eventi nel tuo account.	30 aprile 2023
AmazonInspector2FullAccess : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di recuperare i dettagli delle vulnerabilità del codice rilevate dalla scansione del codice Lambda.	21 aprile 2023

Modifica	Descrizione	Data
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di inviare informazioni ad Amazon EC2 Systems Manager sui percorsi personalizzati definiti da un cliente per l'ispezione approfondita di Amazon EC2.	17 aprile 2023
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di creare canali AWS CloudTrail collegati ai servizi nel tuo account quando attivi la scansione Lambda. Ciò consente ad Amazon Inspector di monitorare e CloudTrail gli eventi nel tuo account.	30 aprile 2023

Modifica	Descrizione	Data
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di richiedere scansioni del codice di sviluppo nelle AWS Lambda funzioni e ricevere dati di scansione da Amazon Security. CodeGuru Inoltre, Amazon Inspector ha aggiunto le autorizzazioni per la revisione delle politiche IAM. Amazon Inspector utilizza queste informazioni per scansionare le funzioni Lambda alla ricerca di vulnerabilità del codice.	28 febbraio 2023
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha aggiunto una nuova istruzione e che consente ad Amazon Inspector di recuperare informazioni CloudWatch sull'ultima volta che AWS Lambda una funzione è stata richiamata. Amazon Inspector utilizza queste informazioni per concentrare le scansioni sulle funzioni Lambda del tuo ambiente che sono state attive negli ultimi 90 giorni.	20 febbraio 2023

Modifica	Descrizione	Data
<p>AmazonInspector2ServiceRolePolicy: aggiornamenti a una policy esistente</p>	<p>Amazon Inspector ha aggiunto una nuova dichiarazione che consente ad Amazon Inspector di recuperare informazioni AWS Lambda sulle funzioni, inclusa ogni versione di livello associata a ciascuna funzione. Amazon Inspector utilizza queste informazioni per scansionare le funzioni Lambda alla ricerca di vulnerabilità di sicurezza.</p>	<p>28 novembre 2022</p>
<p>AmazonInspector2ServiceRolePolicy: aggiornamenti a una policy esistente</p>	<p>Amazon Inspector ha aggiunto una nuova azione per consentire ad Amazon Inspector di descrivere le esecuzioni delle associazioni SSM. Inoltre, Amazon Inspector ha aggiunto un ulteriore ambito delle risorse per consentire ad Amazon Inspector di creare, aggiornare, eliminare e avviare associazioni SSM con documenti SSM di proprietà. AmazonInspector2</p>	<p>31 agosto 2022</p>
<p>AmazonInspector2ServiceRolePolicyAggiornamenti a una policy esistente</p>	<p>Amazon Inspector ha aggiornato l'ambito delle risorse della policy per consentire ad Amazon Inspector di raccogliere l'inventario del software in altre partizioni. AWS</p>	<p>12 agosto 2022</p>

Modifica	Descrizione	Data
AmazonInspector2ServiceRolePolicy : aggiornamenti a una policy esistente	Amazon Inspector ha ristrutturato l'ambito delle risorse delle azioni che consentono ad Amazon Inspector di creare, eliminare e aggiornare le associazioni SSM.	10 agosto 2022
AmazonInspector2ReadOnlyAccess : nuova policy	Amazon Inspector ha aggiunto una nuova policy per consentire e l'accesso in sola lettura alle funzionalità di Amazon Inspector.	21 gennaio 2022
AmazonInspector2FullAccess : nuova policy	Amazon Inspector ha aggiunto una nuova policy per consentire e l'accesso completo alle funzionalità di Amazon Inspector.	29 novembre 2021
AmazonInspector2ServiceRolePolicy : nuova policy	Amazon Inspector ha aggiunto una nuova politica per consentire ad Amazon Inspector di eseguire azioni in altri servizi per tuo conto.	29 novembre 2021
Amazon Inspector ha iniziato a tracciare le modifiche	Amazon Inspector ha iniziato a tracciare le modifiche per le sue politiche AWS gestite.	29 novembre 2021

Utilizzo di ruoli collegati ai servizi per Amazon Inspector

Amazon Inspector utilizza un ruolo collegato al [servizio AWS Identity and Access Management \(IAM\) denominato](#) `AWSServiceRoleForAmazonInspector2`. Questo ruolo collegato al servizio è un ruolo IAM collegato direttamente ad Amazon Inspector. È predefinito da Amazon Inspector e include

tutte le autorizzazioni richieste da Amazon Inspector per chiamare altri utenti per tuo conto. Servizi AWS

Un ruolo collegato al servizio semplifica la configurazione di Amazon Inspector perché non è necessario aggiungere manualmente le autorizzazioni necessarie. Amazon Inspector definisce le autorizzazioni del suo ruolo collegato al servizio e, se non diversamente definito, solo Amazon Inspector può assumere il ruolo. Le autorizzazioni definite includono la policy di attendibilità e la policy delle autorizzazioni. Una policy delle autorizzazioni specifica non può essere collegata a un'altra entità IAM.

È necessario configurare le autorizzazioni per consentire a un'entità IAM (come un gruppo o un ruolo) di creare, modificare o eliminare un ruolo collegato al servizio. Per ulteriori informazioni, consulta le [autorizzazioni dei Service-linked ruoli nella Guida](#) per l'utente IAM. Puoi eliminare un ruolo collegato al servizio solo dopo aver eliminato le relative risorse. In questo modo proteggi le tue risorse Amazon Inspector perché non puoi rimuovere inavvertitamente l'autorizzazione ad accedere alle risorse.

Per informazioni su altri servizi che supportano ruoli collegati ai servizi, consulta i [AWS servizi che funzionano con IAM](#) e cerca i servizi con Sì nella colonna dei ruoli. Service-linked Scegli un Sì con un link per consultare la documentazione relativa ai ruoli collegati ai servizi per quel servizio.

Service-linked autorizzazioni di ruolo per Amazon Inspector

Amazon Inspector utilizza la policy gestita denominata [AWSServiceRoleForAmazonInspector2](#). Questo ruolo collegato al servizio si fida che il `inspector2.amazonaws.com` servizio assuma il ruolo.

La politica di autorizzazione per il ruolo, che è denominato [AmazonInspector2ServiceRolePolicy](#), consente ad Amazon Inspector di eseguire attività come:

- Usa le azioni di Amazon Elastic Compute Cloud (Amazon EC2) per recuperare informazioni sulle tue istanze e sui percorsi di rete.
- Utilizza AWS Systems Manager le azioni per recuperare l'inventario dalle tue istanze Amazon EC2 e per recuperare informazioni sui pacchetti di terze parti da percorsi personalizzati.
- Utilizzate l' AWS Systems Manager SendCommandazione per richiamare le scansioni CIS per le istanze di destinazione.
- Utilizza le azioni di Amazon Elastic Container Registry per recuperare informazioni sulle immagini dei contenitori.
- Usa AWS Lambda le azioni per recuperare informazioni sulle tue funzioni Lambda.

- Usa AWS Organizations le azioni per descrivere gli account associati.
- Usa CloudWatch le azioni per recuperare informazioni sull'ultima volta che le tue funzioni Lambda sono state richiamate.
- Utilizza azioni IAM selezionate per recuperare informazioni sulle tue policy IAM che potrebbero creare vulnerabilità di sicurezza nel tuo codice Lambda.
- Usa le azioni di Amazon Q per eseguire scansioni del codice nelle tue funzioni Lambda. Amazon Inspector utilizza le seguenti azioni Amazon Q:
 - codeguru-security: CreateScan — Concede l'autorizzazione a creare Amazon Q; scan.
 - codeguru-security: GetScan — Concede l'autorizzazione a recuperare i metadati di scansione di Amazon Q.
 - codeguru-security: ListFindings — Concede l'autorizzazione a recuperare i risultati generati da Amazon Q.
 - codeguru-security: DeleteScansByCategory — Autorizza Amazon Q a eliminare le scansioni avviate da Amazon Inspector.
 - codeguru-security: BatchGetFindings — Concede l'autorizzazione a recuperare un batch di risultati specifici generati da Amazon Q.
- Utilizza determinate azioni Elastic Load Balancing per eseguire scansioni di rete delle istanze EC2 che fanno parte dei gruppi target di Elastic Load Balancing.
- Utilizza le azioni Amazon ECS e Amazon EKS per consentire l'accesso in sola lettura per visualizzare cluster e attività e descrivere le attività.
- Utilizza AWS Organizations le azioni per elencare gli amministratori delegati di Amazon Inspector in tutte le organizzazioni.
- Utilizza le azioni di Amazon Inspector per abilitare e disabilitare Amazon Inspector in tutte le organizzazioni.
- Utilizza le azioni di Amazon Inspector per designare account amministratori delegati e associare gli account dei membri tra le organizzazioni.

Note

Amazon Inspector non esegue più scansioni CodeGuru Lambda. AWS interromperà il supporto il 20 novembre 2025. Per ulteriori informazioni, consulta [Fine del supporto per la CodeGuru sicurezza](#). Amazon Inspector ora utilizza Amazon Q per eseguire scansioni Lambda e non richiede le autorizzazioni descritte in questa sezione.

Per esaminare le autorizzazioni relative a questa politica, consulta la Managed Policy Reference [AmazonInspector2ServiceRolePolicy](#)Guide.AWS

Creazione di un ruolo collegato ai servizi per Amazon Inspector

Non hai bisogno di creare manualmente un ruolo collegato ai servizi. Quando attivi Amazon Inspector nell'API Console di gestione AWS, nella o nell' AWS API AWS CLI, Amazon Inspector crea il ruolo collegato al servizio per te.

Modifica di un ruolo collegato al servizio per Amazon Inspector

Amazon Inspector non consente di modificare il ruolo collegato al `AWSServiceRoleForAmazonInspector2` servizio. Dopo aver creato un ruolo collegato al servizio, non è possibile modificare il nome del ruolo perché diverse entità potrebbero fare riferimento al ruolo. È possibile tuttavia modificarne la descrizione utilizzando IAM. Per ulteriori informazioni, consulta [Modifica di un ruolo collegato al servizio](#) nella Guida per l'utente di IAM.

Eliminazione di un ruolo collegato al servizio per Amazon Inspector

Se non hai più bisogno di utilizzare Amazon Inspector, ti consigliamo di eliminare il ruolo collegato al `AWSServiceRoleForAmazonInspector2` servizio. Prima di poter eliminare il ruolo, devi disattivare Amazon Inspector in Regione AWS ogni luogo in cui è attivato. Quando disattivi Amazon Inspector, il ruolo non viene eliminato per te. Pertanto, se attivi nuovamente Amazon Inspector, può utilizzare il ruolo esistente. In questo modo puoi evitare di avere un'entità inutilizzata che non viene monitorata o gestita attivamente. Tuttavia, è necessario effettuare la pulizia delle risorse associate al ruolo collegato al servizio prima di poterlo eliminare manualmente.

Se devi ricreare un ruolo collegato ai servizi che hai precedentemente eliminato, puoi utilizzare lo stesso processo per ricreare il ruolo nel tuo account. Quando attivi Amazon Inspector, Amazon Inspector ricrea per te il ruolo collegato al servizio.

Note

Se il servizio Amazon Inspector utilizza il ruolo quando tenti di eliminare le risorse, l'eliminazione potrebbe non riuscire. In tal caso, attendi qualche minuto e poi riprova a eseguire l'operazione.

Puoi utilizzare la console IAM AWS CLI, o l' AWS API per eliminare il ruolo `AWSServiceRoleForAmazonInspector2` collegato al servizio. Per ulteriori informazioni, consulta [Eliminazione del ruolo collegato ai servizi](#) nella Guida per l'utente IAM.

Service-linked autorizzazioni di ruolo per le scansioni senza agenti di Amazon Inspector

La scansione senza agenti di Amazon Inspector utilizza il ruolo collegato al servizio denominato `AWSServiceRoleForAmazonInspector2Agentless`. Questa reflex consente ad Amazon Inspector di creare uno snapshot del volume Amazon EBS nel tuo account e quindi accedere ai dati da tale snapshot. Questo ruolo collegato al servizio si fida che il servizio assuma il ruolo `agentless.inspector2.amazonaws.com`.

Important

Le istruzioni in questo ruolo collegato al servizio impediscono ad Amazon Inspector di eseguire scansioni senza agenti su qualsiasi istanza EC2 che hai escluso dalle scansioni utilizzando il tag `InspectorEc2Exclusion`. Inoltre, le istruzioni impediscono ad Amazon Inspector di accedere ai dati crittografati da un volume quando la chiave KMS utilizzata per crittografarli ha il tag `InspectorEc2Exclusion`. Per ulteriori informazioni, consulta [Esclusione delle istanze dalle scansioni di Amazon Inspector](#).

La politica di autorizzazione per il ruolo, che è denominato `AmazonInspector2AgentlessServiceRolePolicy`, consente ad Amazon Inspector di eseguire attività come:

- Usa le azioni di Amazon Elastic Compute Cloud (Amazon EC2) per recuperare informazioni sulle istanze, i volumi e gli snapshot EC2.
 - Usa le azioni di tagging di Amazon EC2 per etichettare gli snapshot per le scansioni con la chiave tag `InspectorScan`.
 - Utilizza le azioni snapshot di Amazon EC2 per creare istantanee, etichettarle con la chiave `InspectorScan` tag e quindi eliminare le istantanee dei volumi Amazon EBS a cui è stato assegnato il tag key `InspectorScan`.
- Utilizza le azioni di Amazon EBS per recuperare informazioni dagli snapshot contrassegnati con la `InspectorScan` chiave tag.

- Utilizza azioni di decrittografia selezionate per AWS KMS decrittografare istantanee crittografate con chiavi gestite dal cliente. AWS KMS Amazon Inspector non decrittografa le istantanee quando la chiave KMS utilizzata per crittografarle è contrassegnata con il tag. `InspectorEc2Exclusion`

Il ruolo è configurato con la seguente politica di autorizzazioni.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "InstanceIdentification",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstances",
        "ec2:DescribeVolumes",
        "ec2:DescribeSnapshots"
      ],
      "Resource": "*"
    },
    {
      "Sid": "GetSnapshotData",
      "Effect": "Allow",
      "Action": [
        "ebs:ListSnapshotBlocks",
        "ebs:GetSnapshotBlock"
      ],
      "Resource": "arn:aws:ec2:*:*:snapshot/*",
      "Condition": {
        "StringLike": {
          "aws:ResourceTag/InspectorScan": "*"
        }
      }
    },
    {
      "Sid": "CreateSnapshotsAnyInstanceOrVolume",
      "Effect": "Allow",
      "Action": "ec2:CreateSnapshots",
      "Resource": [
        "arn:aws:ec2:*:*:instance/*",
```

```
    "arn:aws:ec2:*:*:volume/*"
  ]
},
{
  "Sid": "DenyCreateSnapshotsOnExcludedInstances",
  "Effect": "Deny",
  "Action": "ec2:CreateSnapshots",
  "Resource": "arn:aws:ec2:*:*:instance/*",
  "Condition": {
    "StringEquals": {
      "ec2:ResourceTag/InspectorEc2Exclusion": "true"
    }
  }
},
{
  "Sid": "CreateSnapshotsOnAnySnapshotOnlyWithTag",
  "Effect": "Allow",
  "Action": "ec2:CreateSnapshots",
  "Resource": "arn:aws:ec2:*:*:snapshot/*",
  "Condition": {
    "Null": {
      "aws:TagKeys": "false"
    },
    "ForAllValues:StringEquals": {
      "aws:TagKeys": "InspectorScan"
    }
  }
},
{
  "Sid": "CreateOnlyInspectorScanTagOnlyUsingCreateSnapshots",
  "Effect": "Allow",
  "Action": "ec2:CreateTags",
  "Resource": "arn:aws:ec2:*:*:snapshot/*",
  "Condition": {
    "StringLike": {
      "ec2:CreateAction": "CreateSnapshots"
    },
    "Null": {
      "aws:TagKeys": "false"
    },
    "ForAllValues:StringEquals": {
      "aws:TagKeys": "InspectorScan"
    }
  }
}
```

```

    },
    {
      "Sid": "DeleteOnlySnapshotsTaggedForScanning",
      "Effect": "Allow",
      "Action": "ec2:DeleteSnapshot",
      "Resource": "arn:aws:ec2:*:*:snapshot/*",
      "Condition": {
        "StringLike": {
          "ec2:ResourceTag/InspectorScan": "*"
        }
      }
    },
    {
      "Sid": "DenyKmsDecryptForExcludedKeys",
      "Effect": "Deny",
      "Action": "kms:Decrypt",
      "Resource": "arn:aws:kms:*:*:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/InspectorEc2Exclusion": "true"
        }
      }
    },
    {
      "Sid": "DecryptSnapshotBlocksVolContext",
      "Effect": "Allow",
      "Action": "kms:Decrypt",
      "Resource": "arn:aws:kms:*:*:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceAccount": "${aws:PrincipalAccount}"
        },
        "StringLike": {
          "kms:ViaService": "ec2.*.amazonaws.com",
          "kms:EncryptionContext:aws:ebs:id": "vol-*"
        }
      }
    },
    {
      "Sid": "DecryptSnapshotBlocksSnapContext",
      "Effect": "Allow",
      "Action": "kms:Decrypt",
      "Resource": "arn:aws:kms:*:*:key/*",
      "Condition": {

```

```

    "StringEquals": {
      "aws:ResourceAccount": "${aws:PrincipalAccount}"
    },
    "StringLike": {
      "kms:ViaService": "ec2.*.amazonaws.com",
      "kms:EncryptionContext:aws:ebs:id": "snap-*"
    }
  },
  {
    "Sid": "DescribeKeysForEbsOperations",
    "Effect": "Allow",
    "Action": "kms:DescribeKey",
    "Resource": "arn:aws:kms:*:*:key/*",
    "Condition": {
      "StringEquals": {
        "aws:ResourceAccount": "${aws:PrincipalAccount}"
      },
      "StringLike": {
        "kms:ViaService": "ec2.*.amazonaws.com"
      }
    }
  },
  {
    "Sid": "ListKeyResourceTags",
    "Effect": "Allow",
    "Action": "kms:ListResourceTags",
    "Resource": "arn:aws:kms:*:*:key/*"
  }
]
}

```

Creazione di un ruolo collegato al servizio per la scansione senza agenti

Non hai bisogno di creare manualmente un ruolo collegato ai servizi. Quando attivi Amazon Inspector nell'API Console di gestione AWS, nella o nell' AWS API AWS CLI, Amazon Inspector crea il ruolo collegato al servizio per te.

Modifica di un ruolo collegato al servizio per una scansione senza agenti

Amazon Inspector non consente di modificare il ruolo collegato al `AWSServiceRoleForAmazonInspector2Agentless` servizio. Dopo aver creato un ruolo collegato al servizio, non è possibile modificare il nome del ruolo perché diverse entità potrebbero fare riferimento al ruolo. È possibile tuttavia modificarne la descrizione utilizzando IAM. Per ulteriori informazioni, consulta [Modifica di un ruolo collegato al servizio](#) nella Guida per l'utente di IAM.

Eliminazione di un ruolo collegato al servizio per la scansione senza agenti

Se non è più necessario utilizzare una funzionalità o un servizio che richiede un ruolo collegato al servizio, consigliamo di eliminare il ruolo. In questo modo non hai un'entità non utilizzata che non viene monitorata o gestita attivamente.

Important

Per eliminare il `AWSServiceRoleForAmazonInspector2Agentless` ruolo, è necessario impostare la modalità di scansione su basata su agenti in tutte le regioni in cui è disponibile la scansione senza agenti.

Per eliminare manualmente il ruolo collegato ai servizi mediante IAM

Utilizza la console IAM AWS CLI, o l' AWS API per eliminare il ruolo collegato al servizio.

`AWSServiceRoleForAmazonInspector2Agentless` Per ulteriori informazioni, consulta [Eliminazione del ruolo collegato ai servizi](#) nella Guida per l'utente IAM.

Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Inspector

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con Amazon Inspector e IAM.

Argomenti

- [Non sono autorizzato a eseguire un'azione in Amazon Inspector](#)
- [Non sono autorizzato a eseguire iam: PassRole](#)
- [Voglio consentire l'accesso a persone esterne al mio Account AWS per accedere alle mie risorse Amazon Inspector](#)

Non sono autorizzato a eseguire un'azione in Amazon Inspector

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa `my-example-widget` fittizia ma non dispone di autorizzazioni `inspector2:GetWidget` fittizie.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
inspector2:GetWidget on resource: my-example-widget
```

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa `my-example-widget` utilizzando l'azione `inspector2:GetWidget`.

Se hai bisogno di assistenza, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un messaggio di errore indicante che non sei autorizzato a eseguire l'`iam:PassRole` azione, le tue politiche devono essere aggiornate per consentirti di trasferire un ruolo ad Amazon Inspector.

Alcuni Servizi AWS consentono di trasferire un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

Il seguente errore di esempio si verifica quando un utente IAM denominato `marymajor` tenta di utilizzare la console per eseguire un'azione in Amazon Inspector. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per trasmettere il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire l'accesso a persone esterne al mio Account AWS per accedere alle mie risorse Amazon Inspector

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per servizi che supportano policy basate su risorse o liste di controllo degli accessi (ACL), utilizzare tali policy per concedere alle persone l'accesso alle proprie risorse.

Per maggiori informazioni, consulta gli argomenti seguenti:

- Per sapere se Amazon Inspector supporta queste funzionalità, consulta [Come funziona Amazon Inspector con IAM](#)
- Per sapere come fornire l'accesso alle tue risorse su tutto Account AWS ciò che possiedi, consulta [Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà](#) nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta [Fornire l'accesso a soggetti Account AWS di proprietà di terze parti](#) nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta [Fornire l'accesso a utenti autenticati esternamente \(federazione delle identità\)](#) nella Guida per l'utente IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consulta [Accesso a risorse multi-account in IAM](#) nella Guida per l'utente di IAM.

Monitoraggio di Amazon Inspector

Il monitoraggio è una parte importante per mantenere la disponibilità, l'affidabilità e le prestazioni di Amazon Inspector e di altre AWS soluzioni. AWS fornisce strumenti per monitorare Amazon Inspector, segnalare i problemi che si verificano e intraprendere azioni per porvi rimedio:

- [Amazon EventBridge](#) è un AWS servizio che utilizza gli eventi per connettere tra loro i componenti delle applicazioni, semplificando la creazione di applicazioni scalabili basate sugli eventi. EventBridge offre un flusso di dati in tempo reale dalle applicazioni, dalle applicazioni Software-as-a-Service (SaaS), dai AWS servizi e dai percorsi, in modo da poter monitorare gli eventi che si verificano nei servizi e creare architetture basate sugli eventi.
- [AWS CloudTrail](#) è un AWS servizio che acquisisce le chiamate API e gli eventi correlati effettuati da o per conto dell'utente. Account AWS CloudTrail invia i file di log a un bucket Amazon S3 da te

specificato, in modo da poter identificare quali utenti e account hanno chiamato AWS, l'indirizzo IP di origine da cui sono state effettuate le chiamate e quando sono avvenute le chiamate.

Registrazione delle chiamate API di Amazon Inspector tramite AWS CloudTrail

Amazon Inspector è integrato con AWS CloudTrail un servizio che fornisce una registrazione delle azioni intraprese da un utente o ruolo IAM o da un Servizio AWS utente in Amazon Inspector. CloudTrail acquisisce tutte le chiamate API per Amazon Inspector come eventi. Le chiamate acquisite includono chiamate dalla console Amazon Inspector e chiamate alle operazioni dell'API Amazon Inspector. Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon S3, inclusi gli eventi per Amazon Inspector. Se non configuri un trail, è comunque possibile visualizzare gli eventi più recenti nella console di CloudTrail in Event history (Cronologia eventi). Utilizzando le informazioni raccolte da CloudTrail, puoi determinare:

- La richiesta che è stata fatta ad Amazon Inspector.
- Indirizzo IP dal quale è stata effettuata la richiesta.
- Chi ha effettuato la richiesta.
- Quando è stata effettuata la richiesta.

Per ulteriori informazioni CloudTrail, consulta la [Guida AWS CloudTrail per l'utente](#).

Informazioni su Amazon Inspector in CloudTrail

CloudTrail è abilitato sul tuo account al Account AWS momento della creazione dell'account. Quando si verifica un'attività in Amazon Inspector, tale attività viene registrata in un CloudTrail evento insieme ad altri Servizio AWS eventi nella cronologia degli eventi. Puoi visualizzare, cercare e scaricare eventi recenti in Account AWS. Per ulteriori informazioni, consulta [Visualizzazione degli eventi con la cronologia degli CloudTrail eventi](#).

Per una registrazione continua degli eventi del tuo Account AWS, compresi gli eventi per Amazon Inspector, crea un percorso. Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Per impostazione predefinita, quando si crea un percorso nella console, questo sarà valido in tutte le Regioni AWS. Il percorso registra gli eventi di tutte le Regioni nella partizione AWS e distribuisce i file di log nel bucket Amazon S3 specificato. Inoltre, puoi configurarne altri Servizi AWS per analizzare

ulteriormente e agire in base ai dati sugli eventi raccolti nei CloudTrail log. Per ulteriori informazioni, consulta i seguenti argomenti:

- [Panoramica della creazione di un percorso](#)
- [CloudTrail servizi e integrazioni supportati](#)
- [Configurazione delle notifiche Amazon SNS per CloudTrail](#)
- [Ricezione di file di CloudTrail registro da più account](#)
- [Ricezione di file di CloudTrail registro da più regioni](#)

Tutte le azioni di Amazon Inspector vengono registrate da. CloudTrail Tutte le azioni che Amazon Inspector può eseguire sono documentate nell'[Amazon](#) Inspector API Reference. Ad esempio, le chiamate alle operazioni `CreateFindingsReport`, `ListCoverage` e `UpdateOrganizationConfiguration` generano voci nei file di log CloudTrail .

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con le credenziali utente root o utente IAM.
- Se la richiesta è stata effettuata con credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro Servizio AWS.

Per ulteriori informazioni, consulta [Elemento CloudTrail userIdentity](#).

Informazioni sulle voci dei file di log di Amazon Inspector

Un trail è una configurazione che consente la distribuzione di eventi come file di log in un bucket Amazon S3 specificato dall'utente. CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta da un'origine. Gli eventi includono informazioni sull'azione richiesta, la data e l'ora dell'azione, i parametri della richiesta e così via. CloudTrail i file di registro non sono una traccia ordinata dello stack delle chiamate API pubbliche, quindi non vengono visualizzati in un ordine specifico.

Amazon Inspector Scansiona le informazioni in CloudTrail

Amazon Inspector Scan è integrato con. CloudTrail Tutte le operazioni dell'API Amazon Inspector Scan vengono registrate come eventi di gestione. Per un elenco delle operazioni dell'API Amazon

Inspector Scan a cui Amazon Inspector accede, CloudTrail consulta Amazon Inspector [Scan nel riferimento alle API di Amazon Inspector](#).

L'esempio seguente mostra una voce di CloudTrail registro che dimostra l'azione: ScanSbom

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAI23456789EXAMPLE:akua_mansa",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/akua_mansa",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAI23456789EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2023-10-17T15:22:59Z",
        "mfaAuthenticated": "false"
      }
    }
  },
  "eventTime": "2023-10-17T16:02:34Z",
  "eventSource": "gamma-inspector-scan.amazonaws.com",
  "eventName": "ScanSbom",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.0",
  "userAgent": "aws-sdk-java/2.20.162 Mac_OS_X/13.5.2 OpenJDK_64-Bit_Server_VM/17.0.8+7-LTS Java/17.0.8 vendor/Amazon.com_Inc. io/sync http/URLConnection cfg/retry-mode/legacy",
  "requestParameters": {
    "sbom": {
      "specVersion": "1.5",
      "metadata": {
        "component": {
          "name": "debian",
          "type": "operating-system",
```

```
        "version": "9"
      }
    },
    "components": [
      {
        "name": "package0ne",
        "purl": "pkg:deb/debian/package0ne@1.0.0?arch=x86_64&distro=9",
        "type": "application"
      }
    ],
    "bomFormat": "CycloneDX"
  }
},
"responseElements": null,
"requestID": "f041a27f-f33e-4f70-b09b-5fbc5927282a",
"eventID": "abc8d1e4-d214-4f07-bc56-8a31be6e36fe",
"readOnly": true,
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"eventCategory": "Management"
}
```

Convalida della conformità per Amazon Inspector

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione [Scope by Compliance Program Servizi AWS](#) e scegli il programma di conformità che ti interessa. Per informazioni generali, consulta Programmi di [AWS conformità Programmi](#) di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta [Scaricamento dei report in AWS Artifact](#) .

La vostra responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. Per ulteriori informazioni sulla responsabilità di conformità durante l'utilizzo Servizi AWS, consulta [AWS la documentazione sulla sicurezza](#).

Resilienza in Amazon Inspector

L'infrastruttura AWS globale è costruita attorno Regioni AWS a zone di disponibilità. Regioni AWS forniscono zone di disponibilità multiple, fisicamente separate e isolate, collegate a reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità è possibile progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

Sicurezza dell'infrastruttura in Amazon Inspector

In quanto servizio gestito, Amazon Inspector è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi di AWS sicurezza e su come AWS protegge l'infrastruttura, consulta [AWS Cloud Security](#). Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Utilizza chiamate API AWS pubblicate per accedere ad Amazon Inspector attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di crittografia con Perfect Forward Secrecy (PFS) come DHE (Ephemeral) o ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Diffie-Hellman La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Risposta agli incidenti in Amazon Inspector

Per AWS, la sicurezza ha la massima priorità. Come indicato nel [modello di responsabilitàAWS condivisa](#) alla voce «Security of the Cloud», AWS è responsabile della protezione dell'infrastruttura che gestisce tutti i servizi nel AWS Cloud. AWS è inoltre responsabile di qualsiasi risposta agli incidenti associata al servizio Amazon Inspector.

In qualità di AWS cliente, condividi la responsabilità di mantenere la sicurezza nel AWS cloud. Ciò significa che sei tu a controllare la sicurezza che scegli di implementare, che include tutti AWS gli strumenti e le funzionalità a cui accedi. Significa anche che sei responsabile della risposta agli incidenti dalla tua parte del modello di responsabilità condivisa.

Stabilendo una linea di base di sicurezza che soddisfi tutti gli obiettivi delle applicazioni eseguite nel AWS cloud, puoi rilevare le deviazioni a cui puoi rispondere. Poiché la risposta agli incidenti

è un argomento complesso, consulta le seguenti risorse per comprendere meglio l'impatto della risposta agli incidenti e come le tue scelte potrebbero influenzare gli obiettivi aziendali: [AWS Security Incident Response Guide](#), [AWS Security Best Practices](#) e [AWS Cloud Adoption Framework: Security Perspective](#).

Ricerca sulla sicurezza di Amazon Inspector

Amazon Inspector Security Research è un programma di monitoraggio continuo che identifica i pacchetti dannosi pubblicati nei registri pubblici dei pacchetti. Gli attacchi alla catena di fornitura del software rivolti agli ecosistemi open source sono una minaccia crescente e i pacchetti dannosi vengono spesso rimossi dai registri originali solo dopo essere stati scaricati da sviluppatori ignari. Amazon Inspector Security Research aiuta a proteggere le tue applicazioni rilevando questi pacchetti, pubblicando avvisi e incorporando queste informazioni nei risultati di Amazon Inspector in modo che i carichi di lavoro che utilizzano un pacchetto noto come dannoso vengano visualizzati automaticamente.

Registri di pacchetti pubblici supportati:

- NPM: [il registro dei Node.js pacchetti \(npmjs.com\)](https://www.npmjs.com)
- PyPI — [l'indice dei pacchetti Python \(pypi.org\)](https://pypi.org)

Metodologia di ricerca

Il team di ricerca sulla sicurezza di Amazon Inspector combina pipeline di rilevamento automatizzato con la revisione di analisti esperti per identificare i pacchetti dannosi nei registri supportati. A ogni pacchetto dannoso confermato viene assegnato un pacchetto dannoso MAL-ID, documentato come avviso pubblico e integrato nei risultati di Amazon Inspector in modo che i clienti vengano avvisati quando i loro carichi di lavoro consumano i pacchetti interessati. Non è richiesta alcuna azione da parte del cliente per trarre vantaggio da questa ricerca: i rilevamenti vengono applicati automaticamente.

[Amazon Inspector è un partner collaboratore del Malicious Packages Repository della Open Source Security Foundation \(OpenSSF\)](#). Gli avvisi prodotti da Amazon Inspector Security Research vengono pubblicati su questo set di dati aperto, offrendo alla comunità open source più ampia l'accesso alla stessa intelligence sulle minacce utilizzata da Amazon Inspector per proteggere i clienti. AWS Gli avvisi MAL-ID assegnati a ciascun avviso sono compatibili con il formato di avviso OpenSSF.

Riepilogo del rilevamento

Le tabelle seguenti riepilogano i rilevamenti di pacchetti dannosi da parte di Amazon Inspector Security Research in tutti i registri supportati.

Note

Ultimo aggiornamento: 2026-05-13 21:00:00 UTC

Totali relativi all'intero ciclo di vita per registro

Registro	Pacchetti Lifetime identificati
NPM	188.538
PyPI	12
Totale	188.550

Attività recente di registry

Periodo	Finestra (UTC)	NPM	PyPI	Totale
Questa settimana	2026-05-11.. 2026-05-17	61	12	73
Settimana scorsa	2026-05-04.. 2026-05-10	84	0	84
Questo mese	2026-05	180	12	192
Mese scorso	2026-04	619	0	619

Segnalazioni recenti di pacchetti dannosi

La tabella seguente elenca i dieci avvisi di pacchetti dannosi più recenti pubblicati da Amazon Inspector Security Research, ordinati per data di pubblicazione.

Nome pacchetto	MAL-ID	Registro	Data di rilevamento
d4rktg	MAL-2026-3688	PyPI	2026-05-13
@dropout - ai/runtime	MAL-2026-3683	NPM	2026-05-13
amino.fix	MAL-2026-3686	PyPI	2026-05-13
@gusmano/reext	MAL-2026-3684	GIRI/MIN	2026-05-12
sempre aggiornato	MAL-2026-3685	PyPI	2026-05-12
@a91082900/test_package	MAL-2026-3680	GIRI/MIN	2026-05-12
kaggle-runner	MAL-2026-3693	PyPI	2026-05-12
88 q	MAL-2026-3676	GIRI/MIN	2026-05-12
66o	MAL-2026-3674	GIRI/MIN	2026-05-12
6 cc	MAL-2026-3675	GIRI/MIN	2026-05-12

Risorse correlate

I seguenti post del AWS Security Blog forniscono un contesto aggiuntivo su Amazon Inspector Security Research e sulle recenti campagne sulle minacce alla catena di approvvigionamento:

- [Amazon Inspector rileva oltre 150.000 pacchetti dannosi collegati alla campagna di token farming](#)
- [Difesa da attacchi alla catena di fornitura come chalk, debug e worm Shai-Hulud](#)
- [Cosa ha imparato AWS Security rispondendo alle recenti campagne di npm Supply Chain Threat](#)

Accedi ad Amazon Inspector utilizzando un endpoint di interfaccia (AWS PrivateLink)

Puoi utilizzarlo AWS PrivateLink per creare una connessione privata tra il tuo VPC e Amazon Inspector. Puoi accedere ad Amazon Inspector come se fosse nel tuo VPC, senza l'uso di un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. Direct Connect Le istanze nel tuo VPC non necessitano di indirizzi IP pubblici per accedere ad Amazon Inspector.

Stabilisci questa connessione privata creando un endpoint di interfaccia attivato da AWS PrivateLink. In ciascuna sottorete viene creata un'interfaccia di rete endpoint da abilitare per l'endpoint di interfaccia. Si tratta di interfacce di rete gestite dai richiedenti che fungono da punto di ingresso per il traffico destinato ad Amazon Inspector.

Per ulteriori informazioni, consulta [Access Servizi AWS](#) through nella Guida. AWS PrivateLinkAWS PrivateLink

Considerazioni per Amazon Inspector

Prima di configurare un endpoint di interfaccia per Amazon Inspector, [consulta](#) le considerazioni nella Guida.AWS PrivateLink

Amazon Inspector supporta l'esecuzione di chiamate a tutte le sue azioni API tramite l'endpoint dell'interfaccia.

Le policy degli endpoint VPC non sono supportate per Amazon Inspector. Per impostazione predefinita, l'accesso completo ad Amazon Inspector è consentito tramite l'endpoint dell'interfaccia. In alternativa, puoi associare un gruppo di sicurezza alle interfacce di rete degli endpoint per controllare il traffico verso Amazon Inspector attraverso l'endpoint dell'interfaccia.

Crea un endpoint di interfaccia per Amazon Inspector

Puoi creare un endpoint di interfaccia per Amazon Inspector utilizzando la console Amazon VPC o (). AWS Command Line Interface AWS CLI Per ulteriori informazioni, consulta la sezione [Creazione di un endpoint di interfaccia](#) nella Guida per l'utente di AWS PrivateLink .

Quando crei un endpoint di interfaccia per Amazon Inspector, utilizza uno dei seguenti nomi di servizio:

```
com.amazonaws.region.inspector2
```

```
com.amazonaws.region.inspector-scan
```

Sostituiscilo *region* con il Regione AWS codice applicabile. Regione AWS

Se abiliti il DNS privato per l'endpoint dell'interfaccia, puoi effettuare richieste API ad Amazon Inspector utilizzando il nome DNS regionale predefinito, ad esempio `service-name.us-east-1.amazonaws.com` , `service-name.us-east-1.api.aws.com` o per gli Stati Uniti orientali (Virginia settentrionale).

Integrazioni con Amazon Inspector

Amazon Inspector si integra con altri servizi. AWS Questi servizi possono importare dati da Amazon Inspector, quindi puoi visualizzare i risultati in diversi modi. Consulta le seguenti opzioni di integrazione per saperne di più.

Utilizzo di Amazon Inspector con AWS Organizations

[AWS Organizations](#) ti aiuta a gestire e governare centralmente il tuo AWS ambiente. Puoi utilizzare AWS Organizations le policy per abilitare e gestire Amazon Inspector su più account della tua organizzazione in modo automatico.

Le politiche organizzative di Amazon Inspector consentono di:

- Abilita centralmente i tipi di scansione di Amazon Inspector (EC2, ECR, Lambda, Code Repository) in tutta l'organizzazione
- Applica automaticamente l'abilitazione di Amazon Inspector ai nuovi account che si uniscono all'organizzazione
- Applica una copertura di scansione coerente tra le unità organizzative
- Impedisce agli account dei membri di disabilitare la scansione richiesta

Le politiche dell'organizzazione controllano l'abilitazione dei tipi di risorse, mentre gli amministratori delegati mantengono il controllo sulle impostazioni di configurazione della scansione. Per informazioni su come le politiche dell'organizzazione interagiscono con le autorizzazioni delegate degli account amministratore e membro, vedere [Gestione di più account in Amazon Inspector con AWS Organizations](#) Per istruzioni dettagliate sulla creazione delle politiche di Amazon Inspector, consulta la AWS Organizations documentazione relativa alle politiche di Amazon Inspector.

Integrazione di Amazon Inspector con Amazon ECR

[Amazon Elastic Container Registry \(Amazon ECR\)](#) è AWS un registro di immagini di container gestito che supporta registri privati. I registri privati di Amazon ECR ospitano immagini di container in un'architettura altamente disponibile e scalabile. Puoi usare Amazon Inspector per scansionare le immagini dei container che risiedono nel tuo repository Amazon ECR alla ricerca di pacchetti di sistemi operativi e pacchetti di linguaggi di programmazione vulnerabili. Per ulteriori informazioni, consulta [Integrazione di Amazon Inspector con Amazon Elastic Container Registry \(Amazon ECR\)](#).

Integrazione di Amazon Inspector con AWS Security Hub CSPM

[AWS Security Hub CSPM](#) fornisce una visione completa dello stato di sicurezza AWS e ti aiuta a controllare il tuo ambiente rispetto agli standard e alle migliori pratiche del settore della sicurezza. Security Hub CSPM raccoglie dati di sicurezza da AWS account, servizi e prodotti supportati. Puoi utilizzare Security Hub CSPM per importare i dati dei risultati di Amazon Inspector e creare una posizione centrale per i risultati in tutti i tuoi AWS servizi integrati e prodotti Partner Network. AWS Per ulteriori informazioni, consulta [Integrazione di Amazon Inspector con AWS Security Hub CSPM](#).

Integrazione di Amazon Inspector con Amazon Elastic Container Registry (Amazon ECR)

Amazon Elastic Container Registry è un registro di container completamente gestito che supporta immagini e AWS artefatti Docker e OCI. Se usi Amazon ECR, puoi attivare [Enhanced Scanning](#) per il registro dei container. Quando attivi la scansione avanzata, Amazon Inspector rileva e scansiona automaticamente le immagini dei container alla ricerca di pacchetti di sistemi operativi e linguaggi di programmazione vulnerabili. Questa integrazione consente di visualizzare i risultati di Amazon Inspector per le immagini dei container e di gestire la frequenza e l'ambito delle scansioni nella console Amazon ECR. Per ulteriori informazioni, consulta [Scansione delle immagini dei container Amazon ECR con Amazon Inspector](#).

Attivazione dell'integrazione

Puoi attivare l'integrazione attivando la scansione di Amazon Inspector tramite la console o l'API di Amazon Inspector oppure configurando il tuo repository per utilizzare la scansione avanzata con Amazon Inspector tramite la console o l'API Amazon ECR.

Per ulteriori informazioni sull'attivazione dell'integrazione tramite Amazon Inspector, consulta [Tipi di scansione automatizzati in Amazon Inspector](#)

Per informazioni sull'attivazione e la configurazione della scansione avanzata in Amazon ECR, consulta [Enhanced Scanning](#) nella guida per l'utente di Amazon ECR.

Utilizzo dell'integrazione con un ambiente multi-account

Se sei un membro di un ambiente con più account, puoi attivare la scansione avanzata tramite Amazon ECR. Tuttavia, una volta attivato, può essere disattivato solo dall'amministratore delegato di

Amazon Inspector. Se è disattivata, torna alla scansione di base. Per ulteriori informazioni, consulta [Disattivazione di Amazon Inspector](#).

Integrazione di Amazon Inspector con AWS Security Hub CSPM

Security Hub CSPM offre una visione completa dello stato di sicurezza in AWS. Questo ti aiuta a controllare il tuo ambiente rispetto agli standard e alle migliori pratiche del settore della sicurezza. Security Hub CSPM raccoglie dati di sicurezza da AWS account, servizi e prodotti supportati. È possibile utilizzare queste informazioni per analizzare le tendenze di sicurezza e identificare i problemi di sicurezza. Quando attivi l'integrazione di Amazon Inspector con Security Hub CSPM, Amazon Inspector può inviare i risultati a Security Hub CSPM e Security Hub CSPM può analizzare tali risultati come parte del tuo livello di sicurezza.

Security Hub CSPM tiene traccia dei problemi di sicurezza come risultati. Alcuni risultati possono essere il risultato di problemi di sicurezza rilevati in altri AWS servizi o prodotti di terze parti. Security Hub CSPM utilizza una serie di regole per rilevare problemi di sicurezza e generare risultati e fornisce strumenti per gestirli. Security Hub CSPM archivia i risultati di Amazon Inspector una volta che i risultati sono stati chiusi in Amazon Inspector. Puoi anche [visualizzare una cronologia dei risultati e dei dettagli delle scoperte, nonché monitorare lo stato di un'indagine su un risultato](#).

Security Hub CSPM elabora i risultati nel [AWS Security Finding Format \(ASFF\)](#). Questo formato include dettagli quali identificatori univoci, livelli di gravità, risorse interessate, indicazioni sulla correzione, stato del flusso di lavoro e informazioni contestuali.

Note

I risultati di sicurezza generati da [Amazon Inspector Code Security](#) non sono disponibili per questa integrazione. Tuttavia, puoi accedere a questi risultati particolari nella console Amazon Inspector e tramite l'API [Amazon Inspector](#).

Argomenti

- [Visualizzazione dei risultati di Amazon Inspector in AWS Security Hub CSPM](#)
- [Attivazione e configurazione dell'integrazione di Amazon Inspector con Security Hub CSPM](#)
- [Attivazione di Amazon Inspector da Security Hub \(CSPM\) utilizzando i criteri dell'organizzazione](#)
- [Disattivazione del flusso di risultati da un'integrazione](#)
- [Visualizzazione dei controlli di sicurezza per Amazon Inspector in Security Hub CSPM](#)

Visualizzazione dei risultati di Amazon Inspector in AWS Security Hub CSPM

Puoi visualizzare i risultati di Amazon Inspector Classic e Amazon Inspector in Security Hub CSPM.

Note

Per filtrare solo in base ai risultati di Amazon Inspector, aggiungili "aws/inspector/ProductVersion": "2" alla barra dei filtri. Questo filtro esclude i risultati di Amazon Inspector Classic dal dashboard CSPM di Security Hub.

Esempio di ricerca da Amazon Inspector

```
{
  "SchemaVersion": "2018-10-08",
  "Id": "arn:aws:inspector2:us-east-1:123456789012:finding/FINDING_ID",
  "ProductArn": "arn:aws:securityhub:us-east-1::product/aws/inspector",
  "ProductName": "Inspector",
  "CompanyName": "Amazon",
  "Region": "us-east-1",
  "GeneratorId": "AWSInspector",
  "AwsAccountId": "123456789012",
  "Types": [
    "Software and Configuration Checks/Vulnerabilities/CVE"
  ],
  "FirstObservedAt": "2023-01-31T20:25:38Z",
  "LastObservedAt": "2023-05-04T18:18:43Z",
  "CreatedAt": "2023-01-31T20:25:38Z",
  "UpdatedAt": "2023-05-04T18:18:43Z",
  "Severity": {
    "Label": "HIGH",
    "Normalized": 70
  },
  "Title": "CVE-2022-34918 - kernel",
  "Description": "An issue was discovered in the Linux kernel through 5.18.9. A type confusion bug in nft_set_elem_init (leading to a buffer overflow) could be used by a local attacker to escalate privileges, a different vulnerability than CVE-2022-32250. (The attacker can obtain root access, but must start with an unprivileged user namespace to obtain CAP_NET_ADMIN access.) This can be fixed in nft_setelem_parse_data in net/netfilter/nf_tables_api.c.",
  "Remediation": {
```

```

    "Recommendation": {
      "Text": "Remediation is available. Please refer to the Fixed version in the
vulnerability details section above. For detailed remediation guidance for each of the
affected packages, refer to the vulnerabilities section of the detailed finding JSON."
    }
  },
  "ProductFields": {
    "aws/inspector/FindingStatus": "ACTIVE",
    "aws/inspector/inspectorScore": "7.8",
    "aws/inspector/resources/1/resourceDetails/awsEc2InstanceDetails/platform":
"AMAZON_LINUX_2",
    "aws/inspector/ProductVersion": "2",
    "aws/inspector/instanceId": "i-0f1ed287081bdf0fb",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/inspector/
arn:aws:inspector2:us-east-1:123456789012:finding/FINDING_ID",
    "aws/securityhub/ProductName": "Inspector",
    "aws/securityhub/CompanyName": "Amazon"
  },
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "arn:aws:ec2:us-east-1:123456789012:i-0f1ed287081bdf0fb",
      "Partition": "aws",
      "Region": "us-east-1",
      "Tags": {
        "Patch Group": "SSM",
        "Name": "High-SEv-Test"
      }
    },
    "Details": {
      "AwsEc2Instance": {
        "Type": "t2.micro",
        "ImageId": "ami-0cff7528ff583bf9a",
        "IPv4Addresses": [
          "52.87.229.97",
          "172.31.57.162"
        ],
        "KeyName": "ACloudGuru",
        "IamInstanceProfileArn": "arn:aws:iam::123456789012:instance-profile/
AmazonSSMRoleForInstancesQuickSetup",
        "VpcId": "vpc-a0c2d7c7",
        "SubnetId": "subnet-9c934cb1",
        "LaunchedAt": "2022-07-26T21:49:46Z"
      }
    }
  ]
}

```

```
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE",
  "Vulnerabilities": [
    {
      "Id": "CVE-2022-34918",
      "VulnerablePackages": [
        {
          "Name": "kernel",
          "Version": "5.10.118",
          "Epoch": "0",
          "Release": "111.515.amzn2",
          "Architecture": "X86_64",
          "PackageManager": "OS",
          "FixedInVersion": "0:5.10.130-118.517.amzn2",
          "Remediation": "yum update kernel"
        }
      ],
      "Cvss": [
        {
          "Version": "2.0",
          "BaseScore": 7.2,
          "BaseVector": "AV:L/AC:L/Au:N/C:C/I:C/A:C",
          "Source": "NVD"
        },
        {
          "Version": "3.1",
          "BaseScore": 7.8,
          "BaseVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H",
          "Source": "NVD"
        },
        {
          "Version": "3.1",
          "BaseScore": 7.8,
          "BaseVector": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H",
          "Source": "NVD",
          "Adjustments": []
        }
      ]
    }
  ],
  "Vendor": {
```

```
    "Name": "NVD",
    "Url": "https://nvd.nist.gov/vuln/detail/CVE-2022-34918",
    "VendorSeverity": "HIGH",
    "VendorCreatedAt": "2022-07-04T21:15:00Z",
    "VendorUpdatedAt": "2022-10-26T17:05:00Z"
  },
  "ReferenceUrls": [
    "https://git.kernel.org/pub/scm/linux/kernel/git/netdev/net.git/commit/?id=7e6bc1f6cabcd30aba0b11219d8e01b952eacbb6",
    "https://lore.kernel.org/netfilter-devel/cd9428b6-7ffb-dd22-d949-d86f4869f452@randorisec.fr/T/",
    "https://www.debian.org/security/2022/dsa-5191"
  ],
  "FixAvailable": "YES"
}
],
"FindingProviderFields": {
  "Severity": {
    "Label": "HIGH"
  },
  "Types": [
    "Software and Configuration Checks/Vulnerabilities/CVE"
  ]
},
"ProcessedAt": "2023-05-05T20:28:38.822Z"
}
```

Attivazione e configurazione dell'integrazione di Amazon Inspector con Security Hub CSPM

Puoi attivare l'integrazione con Amazon Inspector AWS Security Hub CSPM [abilitando Security Hub CSPM](#). [Dopo aver abilitato Security Hub CSPM, l'integrazione AWS Security Hub CSPM con Amazon Inspector viene attivata automaticamente e Amazon Inspector inizia a inviare tutti i risultati a Security Hub CSPM utilizzando AWS il Security Finding Format \(ASFF\)](#).

Attivazione di Amazon Inspector da Security Hub (CSPM) utilizzando i criteri dell'organizzazione

Puoi gestire l'attivazione di Amazon Inspector in tutta l'organizzazione utilizzando le policy di AWS Organizations direttamente dalla console CSPM di Security Hub. Questo approccio centralizzato

consente di abilitare la scansione di più account da parte di Amazon Inspector contemporaneamente tramite la gestione delle policy a livello di organizzazione.

Per istruzioni dettagliate sulla gestione dell'attivazione di Amazon Inspector tramite Security Hub CSPM utilizzando le politiche dell'organizzazione, consulta [Gestire gli account di amministratore delegato per Security Hub CSPM](#) nella Guida per l'utente.AWS Security Hub CSPM

Disattivazione del flusso di risultati da un'integrazione

[Per impedire ad Amazon Inspector di inviare i risultati a Security Hub CSPM, puoi utilizzare la console o l'API Security Hub CSPM e.. AWS CLI](#)

Visualizzazione dei controlli di sicurezza per Amazon Inspector in Security Hub CSPM

Security Hub CSPM analizza i risultati dei prodotti supportati AWS e di terze parti ed esegue controlli di sicurezza automatici e continui rispetto alle regole per generare risultati propri. Le regole sono rappresentate dai controlli di sicurezza, che aiutano a determinare se i requisiti di uno standard sono soddisfatti.

Amazon Inspector utilizza controlli di sicurezza per verificare se le funzionalità di Amazon Inspector sono o devono essere abilitate. Le caratteristiche principali comprendono:

- Scansione Amazon EC2
- Scansione Amazon ECR
- Scansione standard Lambda
- Scansione del codice Lambda

Per ulteriori informazioni, consulta i [controlli di Amazon Inspector nella Guida](#) per l'AWS Security Hub CSPM utente.

Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector

Amazon Inspector è in grado di scansionare le applicazioni software installate nei seguenti sistemi:

- Istanze Amazon Elastic Compute Cloud (Amazon EC2)

Note

Per le istanze Amazon EC2, Amazon Inspector può eseguire la scansione alla ricerca di vulnerabilità dei pacchetti nei sistemi operativi che supportano la scansione basata su agenti. Amazon Inspector può anche scansionare le vulnerabilità dei pacchetti nei sistemi operativi e nei linguaggi di programmazione che supportano la scansione ibrida. Amazon Inspector non esegue la scansione delle vulnerabilità della toolchain. La versione del compilatore del linguaggio di programmazione utilizzata per creare l'applicazione introduce queste vulnerabilità.

- Immagini dei container archiviate nei repository Amazon Elastic Container Registry (Amazon ECR)

Note

Per le immagini dei container ECR, Amazon Inspector è in grado di rilevare le vulnerabilità del sistema operativo e dei pacchetti del linguaggio di programmazione. Amazon Inspector supporta anche immagini protette fornite da Chainguard, Minimus, Echo e Docker. Amazon Inspector non esegue la scansione delle vulnerabilità della toolchain in Rust — la versione del compilatore del linguaggio di programmazione utilizzata per creare l'applicazione introduce queste vulnerabilità.

- AWS Lambda funzioni

Note

Per le funzioni Lambda, Amazon Inspector è in grado di rilevare le vulnerabilità dei pacchetti del linguaggio di programmazione e delle vulnerabilità del codice. Amazon Inspector non esegue la scansione delle vulnerabilità della toolchain. La versione del

compilatore del linguaggio di programmazione utilizzata per creare l'applicazione introduce queste vulnerabilità.

Quando Amazon Inspector analizza le risorse, Amazon Inspector recupera più di 50 feed di dati per generare risultati su vulnerabilità ed esposizioni comuni (CVE). Esempi di queste fonti includono avvisi di sicurezza dei fornitori, feed di dati e feed di intelligence sulle minacce, nonché il National Vulnerability Database (NVD) e MITRE. Amazon Inspector aggiorna i dati sulle vulnerabilità dai feed di origine almeno una volta al giorno.

Affinché Amazon Inspector esegua la scansione di una risorsa, è necessario che la risorsa esegua un sistema operativo supportato o utilizzi un linguaggio di programmazione supportato. Gli argomenti di questa sezione elencano i sistemi operativi, i linguaggi di programmazione e i runtime supportati da Amazon Inspector per diverse risorse e tipi di scansione. Sono inoltre elencati i sistemi operativi fuori produzione.

Note

Amazon Inspector può fornire solo un supporto limitato per un sistema operativo dopo che un fornitore ha interrotto il supporto per il sistema operativo.

Argomenti

- [Sistemi operativi supportati](#)
- [Sistemi operativi fuori produzione](#)
- [Linguaggi di programmazione compatibili](#)
- [Runtime supportati](#)

Sistemi operativi supportati

Questa sezione elenca i sistemi operativi supportati da Amazon Inspector.

Sistemi operativi supportati: Amazon EC2 scanning

La tabella seguente elenca i sistemi operativi supportati da Amazon Inspector per la scansione delle istanze Amazon EC2. [Specifica l'avviso di sicurezza del fornitore per ogni sistema operativo e quali sistemi operativi supportano la scansione basata su agenti e la scansione senza agenti.](#)

Quando si utilizza il metodo di scansione basato su agenti, si configura l'agente SSM per eseguire scansioni continue su tutte le istanze idonee. Amazon Inspector consiglia di configurare una versione dell'agente SSM successiva alla 3.2.2086.0. Per ulteriori informazioni, consulta [Working with the SSM Agent](#) nella Amazon EC2 Systems Manager User Guide.

I rilevamenti del sistema operativo Linux sono supportati solo per l'archivio predefinito del gestore di pacchetti (rpm e dpkg) e non includono applicazioni di terze parti, repository di supporto esteso e repository opzionali (flussi di applicazioni), se non diversamente specificato di seguito. Amazon Inspector analizza il kernel in esecuzione alla ricerca di vulnerabilità. Per alcuni sistemi operativi, ad esempio Ubuntu, è necessario un riavvio affinché gli aggiornamenti vengano visualizzati nei risultati attivi.

Sistema operativo	Versione	Avvisi sulla sicurezza dei fornitori	Supporto per la scansione senza agente	Agent-based supporto per la scansione
AlmaLinux	8	Errata CVE	Sì	Sì
AlmaLinux	9	Errata CVE	Sì	Sì
AlmaLinux	10	Errata CVE	Sì	Sì
Amazon Linux (AL2)	AL2	Errata ALAS ed extra CVE	Sì	Sì
Amazon Linux 2023 (AL2023)	AL2023	Errata ALAS CVE	Sì	Sì
Portabottiglie	1.7.0 e versioni successive	Errata CVE	Sì	Sì
Chainguard	*	Chainguard		Sì
Server Debian (Bullseye)	11	GROTTA DSA	Sì	Sì
Server Debian (Bookworm)	12	GROTTA DSA	Sì	Sì

Sistema operativo	Versione	Avvisi sulla sicurezza dei fornitori	Supporto per la scansione senza agente	Agent-based supporto per la scansione
Server Debian (Trixie)	13	GROTTA DSA	Sì	Sì
Fedora	43	Errata CVE	Sì	Sì
Oracle Linux (Oracle)	8	Errata CVE	Sì	Sì
Oracle Linux (Oracle)	9	Errata CVE	Sì	Sì
Oracle Linux (Oracle)	10	Errata CVE	Sì	Sì
Red Hat Enterprise Linux (RHEL)	8	RHEL CVE (BaseOS e) EUS/E2S/E4S	Sì	Sì
Red Hat Enterprise Linux (RHEL)	9	RHEL CVE (BaseOS e) EUS/E2S/E4S	Sì	Sì
Red Hat Enterprise Linux (RHEL)	10	RHEL CVE (BaseOS e) EUS/E2S/E4S	Sì	Sì
Rocky Linux	8	Errata CVE	Sì	Sì
Rocky Linux	9	Errata CVE	Sì	Sì
Rocky Linux	10	Errata CVE	Sì	Sì
SUSE Linux Enterprise Server (SLES)	15.7	UNA GROTTA DI SUUSE	Sì	Sì

Sistema operativo	Versione	Avvisi sulla sicurezza dei fornitori	Supporto per la scansione senza agente	Agent-based supporto per la scansione
Ubuntu (Bionico)	18.04	USN, Ubuntu Pro (esm-infra e esm-apps)	Sì	Sì
Ubuntu (focale)	20.04	USN, Ubuntu Pro (esm-infra e esm-apps)	Sì	Sì
Ubuntu (Jammy)	22.04	USN, Ubuntu Pro (esm-infra e esm-apps)	Sì	Sì
Ubuntu (Noble Numbat)	24.04	USN, Ubuntu Pro (esm-infra & esm-apps)	Sì	Sì
Ubuntu (Questing Quokka)	25.10	USN, Ubuntu Pro (esm-infra & esm-apps)	Sì	Sì
Windows Server	2016	MSKB	Sì	Sì
Windows Server	2019	MSKB	Sì	Sì
Windows Server	2022	MSKB	Sì	Sì
Windows Server	2025	MSKB	Sì	Sì
macOS (Mojave)	10.14	APPLE-SA	No	Sì
macOS (Catalina)	10.15	APPLE-SA	No	Sì
macOS (Big Sur)	11	APPLE-SA	No	Sì

Sistema operativo	Versione	Avvisi sulla sicurezza dei fornitori	Supporto per la scansione senza agente	Agent-based supporto per la scansione
macOS (Monterey)	12	APPLE-SA	No	Sì
macOS (Ventura)	13	APPLE-SA	No	Sì
macOS (Sonoma)	14	APPLE-SA	No	Sì
macOS (Sequoia)	15	APPLE-SA	No	Sì
macOS (Tahoe)	26	APPLE-SA	No	Sì

Sistemi operativi supportati: scansione Amazon ECR con Amazon Inspector

La tabella seguente elenca i sistemi operativi supportati da Amazon Inspector per la scansione delle immagini dei container nei repository Amazon ECR. Inoltre, specifica l'avviso di sicurezza del fornitore per ciascun sistema operativo.

Sistema operativo	Versione	Avvisi di sicurezza del fornitore
AlmaLinux	8	Errata CVE
AlmaLinux	9	Errata CVE
AlmaLinux	10	Errata CVE
Alpine Linux (Alpine)	3.21	Errata CVE
Alpine Linux (Alpine)	3.22	Errata CVE
Alpine Linux (Alpine)	3.23	Errata CVE

Sistema operativo	Versione	Avvisi di sicurezza del fornitore
Amazon Linux (AL2)	AL2	Errata ALAS ed extra CVE
Amazon Linux 2023 (AL2023)	AL2023	ALAS Errata CVE
Azure Linux	3	Errata CVE
BusyBox	–	MITRE CVE
Chainguard	–	Errata CVE
Debian Server (Bullseye)	11	DSA CVE
Debian Server (Bookworm)	12	DSA CVE
Debian Server (Trixie)	13	DSA CVE
Echo	2	Errata CVE
Fedora	43	Errata CVE
Minimus	–	Errata CVE
Oracle Linux (Oracle)	8	Errata CVE
Oracle Linux (Oracle)	9	Errata CVE
Oracle Linux (Oracle)	10	Errata CVE
Photon OS	4	Errata CVE
Photon OS	5	Errata CVE
Red Hat Enterprise Linux (RHEL)	8	RHEL CVE (BaseOS & EUS/E2S/E4S)
Red Hat Enterprise Linux (RHEL)	9	RHEL CVE (BaseOS & EUS/E2S/E4S)

Sistema operativo	Versione	Avvisi di sicurezza del fornitore
Red Hat Enterprise Linux (RHEL)	10	RHEL CVE (BaseOS & EUS/E2S/E4S)
Rocky Linux	8	Errata CVE
Rocky Linux	9	Errata CVE
Rocky Linux	10	Errata CVE
SUSE Linux Enterprise Server (SLES)	15.7	SUSE CVE
Ubuntu (Bionic)	18.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Focal)	20.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Jammy)	22.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Noble Numbat)	24.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Questing Quokka)	25.10	USN, Ubuntu Pro (esm-infra & esm-apps)
Windows Server	2016	MSKB
Windows Server	2019	MSKB
Windows Server	2022	MSKB
Windows Server	2025	MSKB
Wolfi	–	Errata CVE

Sistemi operativi supportati: scansione CIS

La tabella seguente elenca i sistemi operativi supportati da Amazon Inspector per le scansioni CIS. Specifica inoltre la versione del benchmark CIS per ogni sistema operativo.

Note

Gli standard CIS sono destinati ai sistemi operativi x86_64. Alcuni controlli potrebbero non essere valutati o restituire istruzioni di riparazione non valide sulle risorse ARM-based

Sistema operativo	Versione	Versione benchmark CIS
Amazon Linux 2	AL2	3.0.0
Amazon Linux 2023	AL2023	1.0.0
Red Hat Enterprise Linux (RHEL)	8	3.0.0
Red Hat Enterprise Linux (RHEL)	9	2.0.0
Rocky Linux	8	2.0.0
Rocky Linux	9	1.0.0
SUSE Linux Enterprise Server	15	2.0.1
Ubuntu (Bionico)	18.04	2.2.0
Ubuntu (focale)	20.04	3.0.0
Ubuntu (Jammy)	22.04	2.0.0
Ubuntu (Noble Numbat)	24.04	1.0.0
Windows Server	2016	3.0.0
Windows Server	2019	4.0.0

Sistema operativo	Versione	Versione benchmark CIS
Windows Server	2022	4.0.0
Windows Server	2025	1.0.0

Sistemi operativi supportati: Amazon Inspector Scan API

La tabella seguente elenca i sistemi operativi supportati per l'API Amazon Inspector Scan. Per ulteriori informazioni, consulta [ScanSbom](#) Amazon Inspector V2 API Reference.

Sistema operativo	Versione
AlmaLinux 8	8
AlmaLinux	9
AlmaLinux	10
Alpine Linux	3.21
Alpine Linux	3,22
Alpine Linux	3,23
Amazon Linux	2
Amazon Linux	2023
Bottlerocket	–
BusyBox	1,36,0 +
Chainguard	–
Debian	11
Debian	12
Debian	13

Sistema operativo	Versione
Debian Sid	–
Echo	2
Fedora	43
macOS	11+
MinimOS	–
Oracle Linux	8
Oracle Linux	9
Oracle Linux	10
Photon OS	4
Photon OS	5
Red Hat Enterprise Linux	8
Red Hat Enterprise Linux	9
Red Hat Enterprise Linux	10
Rocky Linux	8
Rocky Linux	9
Rocky Linux	10
SUSE Server	15,7
SUSE Server	16,0
Ubuntu	18,04
Ubuntu	20,04

Sistema operativo	Versione
Ubuntu	22,04
Ubuntu	24,04
Ubuntu	25,10
Wolfi Linux	–
Windows Server	2016
Windows Server	2019
Windows Server	2022
Windows Server	2025

Sistemi operativi fuori produzione

La tabella seguente elenca i sistemi operativi che sono stati interrotti e quando sono stati interrotti.

Anche se Amazon Inspector non fornisce il supporto completo per i sistemi operativi fuori produzione, Amazon Inspector continua a scansionare le istanze Amazon EC2 e le immagini dei container Amazon ECR che le eseguono. Come best practice di sicurezza, Amazon Inspector genererà un risultato CRITICO per le risorse che utilizzano un sistema operativo fuori produzione e consiglia di passare a una versione supportata. I risultati generati da Amazon Inspector per i sistemi operativi fuori produzione devono essere utilizzati solo a scopo informativo.

In conformità alla politica del fornitore, i sistemi operativi fuori produzione non ricevono più gli aggiornamenti delle patch. È possibile che non vengano rilasciati nuovi avvisi di sicurezza per i sistemi operativi fuori produzione. I fornitori possono rimuovere gli avvisi e i rilevamenti di sicurezza esistenti dai propri feed per i sistemi operativi che raggiungono la fine del supporto standard. Di conseguenza, Amazon Inspector può smettere di generare risultati per CVE noti.

Sistema operativo	Versione	Discontinuo
Alpine Linux (Alpine)	3.2	1 maggio 2017

Sistema operativo	Versione	Discontinuo
Linux Alpine (Alpine)	3.3	1° novembre 2017
Linux Alpine (Alpine)	3.4	1° maggio 2018
Linux Alpine (Alpine)	3.5	1° novembre 2018
Linux Alpine (Alpine)	3.6	1 maggio 2019
Linux Alpine (Alpine)	3.7	1° novembre 2019
Linux Alpine (Alpine)	3.8	1 maggio 2020
Linux Alpine (Alpine)	3.9	1 novembre 2020
Linux Alpine (Alpine)	3.10	1 maggio 2021
Alpine Linux (Alpine)	3.11	1° novembre 2021
Alpine Linux (Alpine)	3.12	1 maggio 2022
Alpine Linux (Alpine)	3.13	1 novembre 2022
Linux Alpine (Alpine)	3.14	1 maggio 2023
Alpine Linux (Alpine)	3.15	1 novembre 2023
Alpine Linux (Alpine)	3.16	23 maggio 2024
Alpine Linux (Alpine)	3.17	22 novembre 2024
Alpine Linux (Alpine)	3.18	9 maggio 2025
Alpine Linux (Alpine)	3.19	1 novembre 2025
Alpine Linux (Alpine)	3.20	1 aprile 2026
Amazon Linux (AL1)	2012	31 dicembre 2021
CentOS Linux (CentOS)	7	30 giugno 2024

Sistema operativo	Versione	Discontinuo
CentOS Linux (CentOS)	8	31 dicembre 2021
Server Debian (Jessie)	8	30 giugno 2020
Server Debian (Stretch)	9	30 giugno 2022
Server Debian (Buster)	10	30 giugno 2024
Fedora	33	30 novembre 2021
Fedora	34	7 luglio 2022
Fedora	35	13 dicembre 2022
Fedora	36	16 maggio 2023
Fedora	37	15 dicembre 2023
Fedora	38	21 maggio 2024
Fedora	39	26 novembre 2024
Fedora	40	13 maggio 2025
Fedora	41	19 novembre 2025
Fedora	42	13 maggio 2026
openSUSE Leap	15.2	1° dicembre 2021
openSUSE Leap	15.3	1 dicembre 2022
openSUSE Leap	15.4	7 dicembre 2023
openSUSE Leap	15.5	31 dicembre 2024
openSUSE Leap	15.6	30 aprile 2026
Oracle Linux (Oracle)	6	1° marzo 2021

Sistema operativo	Versione	Discontinuo
Oracle Linux (Oracle)	7	31 dicembre 2024
Sistema operativo Photon	2	2 dicembre 2021
Sistema operativo Photon	3	1° marzo 2024
Red Hat Enterprise Linux (RHEL)	6	30 giugno 2020
Red Hat Enterprise Linux (RHEL)	7	30 giugno 2024
SUSE Linux Enterprise Server (SLES)	12	30 giugno 2016
SUSE Linux Enterprise Server (SLES)	12.1	31 maggio 2017
SUSE Linux Enterprise Server (SLES)	12.2	31 marzo 2018
SUSE Linux Enterprise Server (SLES)	12.3	30 giugno 2019
SUSE Linux Enterprise Server (SLES)	12.4	30 giugno 2020
SUSE Linux Enterprise Server (SLES)	12,5	31 ottobre 2024
SUSE Linux Enterprise Server (SLES)	15	31 dicembre 2019
SUSE Linux Enterprise Server (SLES)	15.1	31 gennaio 2021
SUSE Linux Enterprise Server (SLES)	15.2	31 dicembre 2021

Sistema operativo	Versione	Discontinuo
SUSE Linux Enterprise Server (SLES)	15.3	31 dicembre 2022
SUSE Linux Enterprise Server (SLES)	15,4	31 dicembre 2023
SUSE Linux Enterprise Server (SLES)	15,5	31 dicembre 2024
SUSE Linux Enterprise Server (SLES)	15,6	31 dicembre 2025
Ubuntu (affidabile)	12.04	28 aprile 2017
Ubuntu (affidabile)	14.04	1 aprile 2024
Ubuntu (Xenial)	16.04	1 aprile 2026
Ubuntu (Groovy)	20.10	22 luglio 2021
Ubuntu (Hirsute)	21.04	20 gennaio 2022
Ubuntu (Impish)	21.10	31 luglio 2022
Ubuntu (cinetico)	22.10	20 luglio 2023
Ubuntu (Lunar Lobster)	23.04	25 gennaio 2024
Ubuntu (Minotauro Mantico)	23.10	11 luglio 2024
Ubuntu (Oracular Oriole)	24.10	10 luglio 2025
Ubuntu (Plucky Puffin)	25.04	15 gennaio 2026
Windows Server	2012	10 ottobre 2023
Windows Server	2012 R2	10 ottobre 2023

Linguaggi di programmazione compatibili

Questa sezione elenca i linguaggi di programmazione supportati da Amazon Inspector.

Linguaggi di programmazione supportati: scansione senza agenti di Amazon EC2

Amazon Inspector attualmente supporta i seguenti linguaggi di programmazione per l'esecuzione di scansioni senza agente su istanze Amazon EC2 idonee. [Per ulteriori informazioni, consulta la sezione Scansione senza agente.](#)

Note

Amazon Inspector non esegue la scansione delle vulnerabilità della toolchain in and. Go Rust La versione del compilatore del linguaggio di programmazione utilizzata per creare l'applicazione introduce queste vulnerabilità.

- C#
- Go
- Java
- JavaScript
- PHP
- Python
- Ruby
- Rust


Linguaggi di programmazione supportati: Amazon EC2 deep inspection

Amazon Inspector attualmente supporta i seguenti linguaggi di programmazione per l'esecuzione di scansioni di ispezione approfondita su istanze Amazon EC2 Linux. Per ulteriori informazioni, consulta [l'ispezione approfondita di Amazon Inspector per le istanze Amazon Linux-based EC2.](#)

- Java(formati di archivio.ear, .jar, .par e.war)
- JavaScript

- Python

Amazon Inspector utilizza Systems Manager Distributor per distribuire il plug-in per l'ispezione approfondita dell'istanza Amazon EC2.


 Note

L'ispezione approfondita non è supportata per i sistemi operativi Bottlerocket.

Per eseguire scansioni di ispezione approfondita, Systems Manager Distributor e Amazon Inspector devono supportare il sistema operativo dell'istanza Amazon EC2. Per informazioni sui sistemi operativi supportati in Systems Manager Distributor, vedere [Piattaforme e architetture di pacchetti supportate](#) nella Guida per l'utente di Systems Manager.

Linguaggi di programmazione supportati: Amazon ECR scanning

Amazon Inspector attualmente supporta i seguenti linguaggi di programmazione per la scansione delle immagini dei container nei repository Amazon ECR:

 Note

Amazon Inspector non esegue la scansione delle vulnerabilità della toolchain in Rust. La versione del compilatore del linguaggio di programmazione utilizzata per creare l'applicazione introduce queste vulnerabilità. Per Python le applicazioni [Chainguard che utilizzano Libraries](#), Amazon Inspector riconosce le correzioni di sicurezza con backport e le esclude dai risultati.

- C#
- Go
- Gocateni di strumenti
- Java
- JavaJDK
- JavaScript
- PHP
- Python (incluse Chainguard le librerie)

- Ruby
- Rust

Runtime supportati

Questa sezione elenca i runtime supportati da Amazon Inspector.

Runtime supportati: scansione standard di Amazon Inspector Lambda

La scansione standard di Amazon Inspector Lambda attualmente supporta i seguenti runtime per i linguaggi di programmazione che può utilizzare per la scansione delle funzioni Lambda alla ricerca di vulnerabilità nei pacchetti software di terze parti:

Note

Amazon Inspector non esegue la scansione delle vulnerabilità della toolchain in Rust. La versione del compilatore del linguaggio di programmazione utilizzata per creare l'applicazione introduce queste vulnerabilità.

- Go
 - go1.x
- Java
 - java8
 - java8.al2
 - java11
 - java17
 - java21
 - java25
- .NET
 - .NET 6
 - .NET 8
 - .NET 10
- Node.js

- nodejs12.x
- nodejs14.x
- nodejs16.x
- nodejs18.x
- nodejs20.x
- nodejs22.x
- nodejs24.x
- Python
 - python3.7
 - python3.8
 - python3.9
 - python3.10
 - python3.11
 - python3.12
 - python3.13
- Ruby
 - ruby2.7
 - ruby3.2
 - ruby3.3
- Custom runtimes
 - AL2
 - AL2023

Runtime supportati: scansione del codice Amazon Inspector Lambda

La scansione del codice Amazon Inspector Lambda attualmente supporta i seguenti runtime per i linguaggi di programmazione che può utilizzare durante la scansione delle funzioni Lambda alla ricerca di vulnerabilità nel codice:

- Java
 - java8
 - java8.al2

- java11
- java17
- java21
- java25
- .NET
 - .NET 6
 - .NET 8
- Node.js
 - nodejs12.x
 - nodejs14.x
 - nodejs16.x
 - nodejs18.x
 - nodejs20.x
 - nodejs22.x
- Python
 - python3.7
 - python3.8
 - python3.9
 - python3.10
 - python3.11
 - python3.12
 - python3.13
- Ruby
 - ruby2.7
 - ruby3.2
 - ruby3.3

Disattivazione di Amazon Inspector

Puoi disattivare Amazon Inspector nella console Amazon Inspector o con l'API Amazon Inspector. Se disattivi tutti i tipi di scansione per un account, Amazon Inspector viene disattivato automaticamente per quell'account.

Se disattivi Amazon Inspector per un account, tutti i tipi di scansione vengono disattivati per quell'account. Inoltre, tutte le impostazioni di scansione di Amazon Inspector, inclusi filtri, regole di soppressione e risultati, vengono eliminate per l'account.

Quando disattivi Amazon Inspector Amazon Scan, EC2 Amazon Inspector elimina le seguenti associazioni SSM:

- `InspectorDistributor-do-not-delete`
- `InspectorInventoryCollection-do-not-delete`
- `InvokeInspectorSsmPlugin-do-not-delete`. Inoltre, il plug-in Amazon Inspector SSM installato tramite questa associazione viene rimosso da tutti i tuoi host. Windows Per ulteriori informazioni, consulta [Scansione dell'istanza EC2 Windows](#).

Note

Una volta disattivato Amazon Inspector, non dovrai più sostenere costi di servizio. Tuttavia, puoi riattivare Amazon Inspector in qualsiasi momento.

Per informazioni su come disattivare i tipi di scansione per diverse risorse, consulta [Disattivazione](#) di un tipo di scansione.

Prerequisiti

A seconda del tipo di account, considera quanto segue:

- Se il tuo account è un account Amazon Inspector autonomo, puoi disattivare Amazon Inspector in qualsiasi momento.
- Se il tuo account è un account membro in un ambiente con più account, non puoi disattivare Amazon Inspector. Devi contattare l'amministratore delegato della tua organizzazione per disattivare Amazon Inspector.

- Se sei l'amministratore delegato di un'organizzazione, devi [dissociare tutti i tuoi account membro prima di](#) disattivare Amazon Inspector.
- Se l'abilitazione di Amazon Inspector del tuo account è gestita da AWS Organizations policy, non puoi disattivare i tipi di scansione gestiti da policy tramite la console o l'API di Amazon Inspector. Per disattivare i tipi di scansione di Amazon Inspector, devi modificare la politica dell'organizzazione per disabilitarli esplicitamente tramite AWS Organizations la console o l'API. Puoi disattivare i tipi di scansione che non sono gestiti dalle politiche dell'organizzazione tramite la console o l'API di Amazon Inspector.

Note

Quando disattivi Amazon Inspector come amministratore delegato, disattivi la funzionalità di attivazione automatica per la tua organizzazione.

Disattivazione di Amazon Inspector gestita dalle politiche dell'organizzazione

Se Amazon Inspector è abilitato nei tuoi account tramite AWS Organizations policy, devi utilizzare la AWS Organizations console o l'API per disabilitare Inspector. Gli account dei membri e gli amministratori delegati non possono disabilitare i tipi di scansione gestiti dalle policy tramite la console o l'API di Amazon Inspector.

Per disattivare Amazon Inspector per gli account gestiti tramite policy:

Per disattivare l'abilitazione di Amazon Inspector gestita da policy

1. Accedi all'account di AWS Organizations gestione o all'account amministratore delle politiche.
2. Modifica la politica dell'organizzazione per disattivare in modo esplicito i tipi di scansione nelle regioni in cui desideri disabilitare Inspector. È necessario aggiornare il contenuto della policy per specificare le aree disabilite per i tipi di scansione che si desidera disattivare.
3. AWS Organizations applicherà automaticamente le modifiche alle policy e Amazon Inspector disabiliterà i tipi di scansione specificati negli account interessati.

Per istruzioni dettagliate su come modificare o rimuovere le politiche dell'organizzazione, consulta la AWS Organizations documentazione relativa alle politiche di Amazon Inspector.

Note

Quando scolleghi una politica aziendale dagli account, tali account mantengono le impostazioni correnti di Amazon Inspector (abilitate o disabilitate in base all'ultima politica applicata). Gli account non sono più gestiti dalla policy e possono quindi gestire le impostazioni di Amazon Inspector in modo indipendente o tramite l'amministratore delegato.

Disattiva Amazon Inspector

Note

Prima di disattivare Amazon Inspector, [valuta la possibilità di esportare i risultati](#).

Console

Per disattivare Amazon Inspector

1. [Accedi utilizzando le tue credenziali, quindi apri la console https://console.aws.amazon.com/inspector/ Amazon Inspector su v2/home](https://console.aws.amazon.com/inspector/AmazonInspector/v2/home).
2. Utilizzando il Regione AWS selettore nell'angolo superiore destro della pagina, scegli la regione in cui desideri disattivare Amazon Inspector.
3. Nel pannello di navigazione, scegli Impostazioni generali.
4. Scegliete Disattiva Inspector.
5. Quando viene richiesta la conferma, immettete deactivate nella casella di testo, quindi scegliete Deactivate Inspector.
6. (Consigliato) Ripeti questi passaggi in ogni regione per cui desideri disattivare Amazon Inspector.

API

Esegui l'operazione [Disable](#) API. Nella richiesta, fornisci l'account IDs che stai disattivando e EC2, ECR, LAMBDA se desideri disattivare tutte le scansioni, l'account verrà disattivato.
`resourceTypes`

Quote Amazon Inspector

Questa sezione elenca le quote di Amazon Inspector per. Regione AWS

Risorsa	Default	Commenti
Account membri	10.000	Il numero massimo di account membro associati a un account amministratore delegato di Amazon Inspector. Il limite si basa su Quotas for. AWS Organizations
Regole di eliminazione	500	Il numero massimo di regole di soppressione salvate per AWS account per regione. Non è possibile richiedere un aumento della quota.
Risultati EC2 della rete Amazon	10.000	Il numero massimo di risultati della EC2 rete Amazon per AWS account. Non è possibile richiedere un aumento della quota.
Configurazioni di scansione CIS	500	Il numero massimo di configurazioni di scansione CIS. Non è possibile richiedere un aumento della quota.

Per un elenco delle quote associate ad Amazon Inspector Classic, consulta le quote del [servizio Amazon Inspector Classic](#) nel. Riferimenti generali di AWS [Per un elenco delle quote associate a AWS Organizations](#), consulta [AWS Organizations le quote di servizio](#) nel. Riferimenti generali di AWS

Regioni ed endpoint

Questo argomento include tabelle che mostrano gli endpoint per Amazon Inspector e Amazon Inspector Scan. Include anche tabelle che mostrano quali Regioni AWS supportano le funzionalità di Amazon Inspector. Per visualizzare Regioni AWS dove è disponibile Amazon Inspector, consulta [l'endpoint e le quote di Amazon Inspector](#) nel. Riferimenti generali di Amazon Web Services

Endpoint di servizio per Amazon Inspector

La tabella seguente mostra gli endpoint del servizio per Amazon Inspector. La convenzione di denominazione per gli endpoint Amazon Inspector è. `inspector2.Region.amazonaws.com`

Nome della regione	Regione	Endpoint	Protocollo
US East (Ohio)	us-east-2	inspector2.us-east-2.amazonaws.com	HTTPS
		inspector2-fips.us-east-2.amazonaws.com	HTTPS
US East (N. Virginia)	us-east-1	inspector2.us-east-1.amazonaws.com	HTTPS
		inspector2-fips.us-east-1.amazonaws.com	HTTPS
Stati Uniti occidentali (California settentrionale)	us-west-1	inspector2.us-west-1.amazonaws.com	HTTPS
		inspector2-fips.us-west-1.amazonaws.com	HTTPS
US West (Oregon)	us-west-2	inspector2.us-west-2.amazonaws.com	HTTPS
		inspector2-fips.us-west-2.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo	
Africa (Cape Town)	af-south-1	inspector2.af-south-1.amazonaws.com	HTTPS	
Asia Pacifico (Hong Kong)	ap-east-1	inspector2.ap-east-1.amazonaws.com	HTTPS	
Asia Pacifico (Hyderabad)	ap-south-2	inspector2.ap-south-2.amazonaws.com	HTTPS	
Asia Pacifico (Giacarta)	ap-southeast-3	inspector2.ap-southeast-3.amazonaws.com	HTTPS	
Asia Pacifico (Malesia)	ap-southeast-5	inspector2.ap-southeast-5.amazonaws.com	HTTPS	
Asia Pacifico (Melbourne)	ap-southeast-4	inspector2.ap-southeast-4.amazonaws.com	HTTPS	
Asia Pacifico (Mumbai)	ap-south-1	inspector2.ap-south-1.amazonaws.com	HTTPS	

Nome della regione	Regione	Endpoint	Protocollo
Asia Pacifico (Osaka-Locale)	ap-northeast-3	inspector2.ap-northeast-3.amazonaws.com	HTTPS
Asia Pacifico (Seoul)	ap-northeast-2	inspector2.ap-northeast-2.amazonaws.com	HTTPS
Asia Pacifico (Singapore)	ap-southeast-1	inspector2.ap-southeast-1.amazonaws.com	HTTPS
Asia Pacifico (Sydney)	ap-southeast-2	inspector2.ap-southeast-2.amazonaws.com	HTTPS
Asia Pacifico (Taipei)	ap-east-2	inspector2.ap-east-2.amazonaws.com	HTTPS
Asia Pacifico (Thailandia)	ap-southeast-7	inspector2.ap-southeast-7.amazonaws.com	HTTPS
Asia Pacifico (Tokyo)	ap-northeast-1	inspector2.ap-northeast-1.amazonaws.com	HTTPS
Canada (Centrale)	ca-central-1	inspector2.ca-central-1.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Canada occidentale (Calgary)	ca-west-1	inspector2.ca-west-1.amazonaws.com	HTTPS
Europa (Francoforte)	eu-central-1	inspector2.eu-central-1.amazonaws.com	HTTPS
Europa (Irlanda)	eu-west-1	inspector2.eu-west-1.amazonaws.com	HTTPS
Europa (Londra)	eu-west-2	inspector2.eu-west-2.amazonaws.com	HTTPS
Europa (Milano)	eu-south-1	inspector2.eu-south-1.amazonaws.com	HTTPS
Europa (Parigi)	eu-west-3	inspector2.eu-west-3.amazonaws.com	HTTPS
Europa (Spagna)	eu-south-2	inspector2.eu-south-2.amazonaws.com	HTTPS
Europa (Stoccolma)	eu-north-1	inspector2.eu-north-1.amazonaws.com	HTTPS
Europa (Zurigo)	eu-central-2	inspector2.eu-central-2.amazonaws.com	HTTPS
Israele (Tel Aviv)	il-central-1	inspector2.il-central-1.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Messico (Centrale)	mx-central-1	inspector2.mx-central-1.amazonaws.com	HTTPS
Medio Oriente (Bahrein)	me-south-1	inspector2.me-south-1.amazonaws.com	HTTPS
Medio Oriente (Emirati Arabi Uniti)	me-central-1	inspector2.me-central-1.amazonaws.com	HTTPS
Sud America (São Paulo)	sa-east-1	inspector2.sa-east-1.amazonaws.com	HTTPS
AWS GovCloud (US-East)	us-gov-east-1	inspector2.us-gov-east-1.amazonaws.com	HTTPS
		inspector2-fips.us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud (US-West)	us-gov-west-1	inspector2.us-gov-west-1.amazonaws.com	HTTPS
		inspector2-fips.us-gov-west-1.amazonaws.com	HTTPS

Endpoint per l'API Amazon Inspector Scan

La tabella seguente mostra gli endpoint regionali che possono essere utilizzati per chiamare l'API [Amazon Inspector Scan](#). Quando utilizzi l'API, devi fornire l'endpoint e la regione corrispondente per la AWS regione in cui sei attualmente autenticato.

La convenzione di denominazione per gli endpoint Amazon Inspector Scan è `inspector-scan.region.amazonaws.com`. Ad esempio, se sei autenticato in `us-west-2`, utilizzerai l'endpoint per chiamare l'API `inspector-scan.us-west-2.amazonaws.com`.

Nome della regione	Regione	Endpoint	Protocollo
US East (Ohio)	us-east-2	<code>inspector-scan.us-east-2.amazonaws.com</code>	HTTPS
		<code>inspector-scan-fips.us-east-2.amazonaws.com</code>	HTTPS
US East (N. Virginia)	us-east-1	<code>inspector-scan.us-east-1.amazonaws.com</code>	HTTPS
		<code>inspector-scan-fips.us-east-1.amazonaws.com</code>	HTTPS
Stati Uniti occidentali (California settentrionale)	us-west-1	<code>inspector-scan.us-west-1.amazonaws.com</code>	HTTPS
		<code>inspector-scan-fips.us-west-1.amazonaws.com</code>	HTTPS
US West (Oregon)	us-west-2	<code>inspector-scan.us-west-2.amazonaws.com</code>	HTTPS
		<code>inspector-scan-fips.us-west-2.amazonaws.com</code>	HTTPS
Africa (Città del Capo)	af-south-1	<code>inspector-scan.af-south-1.amazonaws.com</code>	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Asia Pacifico (Hong Kong)	ap-east-1	inspector-scan.ap-east-1.amazonaws.com	HTTPS
Asia Pacifico (Hyderabad)	ap-south-2	inspector-scan.ap-south-2.amazonaws.com	HTTPS
Asia Pacifico (Giacarta)	ap-southeast-3	inspector-scan.ap-southeast-3.amazonaws.com	HTTPS
Asia Pacifico (Malesia)	ap-southeast-5	inspector-scan.ap-southeast-5.amazonaws.com	HTTPS
Asia Pacifico (Melbourne)	ap-southeast-4	inspector-scan.ap-southeast-4.amazonaws.com	HTTPS
Asia Pacifico (Mumbai)	ap-south-1	inspector-scan.ap-south-1.amazonaws.com	HTTPS
Asia Pacifico (Osaka-Locale)	ap-northeast-3	inspector-scan.ap-northeast-3.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Asia Pacifico (Seoul)	ap-northeast-2	inspector-scan.ap-northeast-2.amazonaws.com	HTTPS
Asia Pacifico (Singapore)	ap-southeast-1	inspector-scan.ap-southeast-1.amazonaws.com	HTTPS
Asia Pacifico (Sydney)	ap-southeast-2	inspector-scan.ap-southeast-2.amazonaws.com	HTTPS
Asia Pacifico (Taipei)	ap-east-2	inspector-scan.ap-east-2.amazonaws.com	HTTPS
Asia Pacifico (Thailandia)	ap-southeast-7	inspector-scan.ap-southeast-7.amazonaws.com	HTTPS
Asia Pacifico (Tokyo)	ap-northeast-1	inspector-scan.ap-northeast-1.amazonaws.com	HTTPS
Canada (Centrale)	ca-central-1	inspector-scan.ca-central-1.amazonaws.com	HTTPS
Canada occidentale (Calgary)	ca-west-1	inspector-scan.ca-west-1.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Europa (Francoforte)	eu-central-1	inspector-scan.eu-central-1.amazonaws.com	HTTPS
Europa (Irlanda)	eu-west-1	inspector-scan.eu-west-1.amazonaws.com	HTTPS
Europa (Londra)	eu-west-2	inspector-scan.eu-west-2.amazonaws.com	HTTPS
Europa (Milano)	eu-south-1	inspector-scan.eu-south-1.amazonaws.com	HTTPS
Europa (Parigi)	eu-west-3	inspector-scan.eu-west-3.amazonaws.com	HTTPS
Europa (Spagna)	eu-south-2	inspector-scan.eu-south-2.amazonaws.com	HTTPS
Europa (Stoccolma)	eu-north-1	inspector-scan.eu-north-1.amazonaws.com	HTTPS
Europa (Zurigo)	eu-central-2	inspector-scan.eu-central-2.amazonaws.com	HTTPS
Israele (Tel Aviv)	il-central-1	inspector-scan.il-central-1.amazonaws.com	HTTPS
Messico (Centrale)	mx-central-1	inspector-scan.mx-central-1.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Medio Oriente (Bahrein)	me-south-1	inspector-scan.me-south-1.amazonaws.com	HTTPS
Medio Oriente (Emirati Arabi Uniti)	me-central-1	inspector-scan.me-central-1.amazonaws.com	HTTPS
Sud America (São Paulo)	sa-east-1	inspector-scan.sa-east-1.amazonaws.com	HTTPS
AWS GovCloud (US-East)	us-gov-east-1	inspector-scan.us-gov-east-1.amazonaws.com	HTTPS
		inspector-scan-fips.us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud (US-West)	us-gov-west-1	inspector-scan.us-gov-west-1.amazonaws.com	HTTPS
		inspector-scan-fips.us-gov-west-1.amazonaws.com	HTTPS

Region-specific disponibilità delle funzionalità

Questa sezione descrive la disponibilità delle funzionalità di Amazon Inspector di Regione AWS

Scansione EC2 senza agente per le regioni Amazon EC2

La tabella seguente mostra Regioni AWS dove è attualmente disponibile la scansione senza agente per Amazon EC2.

Nome della regione	Codice regione
Stati Uniti orientali (Virginia settentrionale)	us-east-1
Stati Uniti orientali (Ohio)	us-east-2
Stati Uniti occidentali (California settentrionale)	us-west-1
US West (Oregon)	us-west-2
Africa (Città del Capo)	af-south-1
Asia Pacific (Hong Kong)	ap-east-1
Asia Pacifico (Tokyo)	ap-northeast-1
Asia Pacifico (Seoul)	ap-northeast-2
Asia Pacifico (Osaka-Locale)	ap-northeast-3
Asia Pacifico (Mumbai)	ap-south-1
Asia Pacifico (Hyderabad)	ap-south-2
Asia Pacifico (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pacifico (Giacarta)	ap-southeast-3
Asia Pacifico (Melbourne)	ap-southeast-4
Asia Pacifico (Malesia)	ap-southeast-5
Asia Pacifico (Thailandia)	ap-southeast-7
Canada (Centrale)	ca-central-1
Canada occidentale (Calgary)	ca-west-1
Europa (Stoccolma)	eu-north-1

Nome della regione	Codice regione
Europa (Francoforte)	eu-central-1
Europa (Zurigo)	eu-central-2
Europa (Irlanda)	eu-west-1
Europe (London)	eu-west-2
Europe (Paris)	eu-west-3
Europe (Milan)	eu-south-1
Europa (Spagna)	eu-south-2
Israele (Tel Aviv)	il-central-1
Medio Oriente (Emirati Arabi Uniti)	me-central-1
Medio Oriente (Bahrein)	me-south-1
Messico (Centrale)	mx-central-1
Sud America (San Paolo)	sa-east-1
AWS GovCloud (US-East)	us-gov-east-1
AWS GovCloud (US-West)	us-gov-west-1

Regioni di scansione del codice Lambda

La tabella seguente mostra Regioni AWS dove è attualmente disponibile la [scansione del codice Lambda](#).

Nome della regione	Codice regione
Stati Uniti orientali (Virginia settentrionale)	us-east-1
Stati Uniti occidentali (Oregon)	us-west-2

Nome della regione	Codice regione
Stati Uniti orientali (Ohio)	us-east-2
Asia Pacifico (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1
Europa (Francoforte)	eu-central-1
Europe (Ireland)	eu-west-1
Europe (London)	eu-west-2
Europa (Stoccolma)	eu-north-1
Asia Pacifico (Singapore)	ap-southeast-1

Important

Se tenti di abilitare la scansione del codice Lambda con l'API Amazon [Inspector](#) Enable Regione AWS in un luogo in cui la scansione del codice Lambda non è disponibile, ricevi il seguente errore di accesso negato:

```
An error occurred (AccessDeniedException) when calling the Enable operation:
Lambda code scanning is not supported in unsupported-Regione AWS
```

Regioni di sicurezza del codice Amazon Inspector

La tabella seguente mostra Regioni AWS dove Amazon Inspector Code Security è attualmente disponibile.

Nome della regione	Codice regione
Stati Uniti orientali (Virginia settentrionale)	us-east-1
Stati Uniti occidentali (Oregon)	us-west-2

Nome della regione	Codice regione
Stati Uniti orientali (Ohio)	us-east-2
Asia Pacifico (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1
Europa (Francoforte)	eu-central-1
Europe (Ireland)	eu-west-1
Europe (London)	eu-west-2
Europa (Stoccolma)	eu-north-1
Asia Pacifico (Singapore)	ap-southeast-1

AWS GovCloud (US) Regioni

Per le informazioni più recenti, consulta [Amazon Inspector nella Guida](#) per l'AWS GovCloud (US) utente.

Cronologia dei documenti

La tabella seguente descrive le modifiche importanti in ogni versione della Amazon Inspector User Guide, a partire da novembre 2021. Per ricevere notifiche sugli aggiornamenti della documentazione, puoi abbonarti a un feed RSS.

Aggiornamenti dei prodotti Amazon Inspector

Modifica	Descrizione	Data
Aggiornamenti per il plug-in Amazon Inspector SSM, Amazon Inspector SBOM Generator e Amazon Inspector VM Scanner	Amazon Inspector è a conoscenza di uno scenario in cui il plug-in Amazon Inspector SSM, Amazon Inspector SBOM Generator e Amazon Inspector VM Scanner possono generare risultati di vulnerabilità per, e. CVE-2026-27145 CVE-2026-42504 CVE-2026-42507 È stato confermato che il plug-in Amazon Inspector SSM, Amazon Inspector SBOM Generator e Amazon Inspector VM Scanner non sono interessati da queste vulnerabilità. Questi problemi verranno risolti nella prossima versione del plug-in Amazon Inspector SSM, Amazon Inspector SBOM Generator e Amazon Inspector VM Scanner.	3 giugno 2026
Nuova funzionalità	Per eseguire valutazioni di vulnerabilità avanzate	29 maggio 2026

delle istanze Amazon Elastic Compute Cloud, ora puoi utilizzare Amazon Inspector VM Scanner anziché il plug-in Amazon Inspector SSM. Lo scanner VM di Amazon Inspector offre diversi vantaggi, come una raccolta più granulare dei pacchetti e utilizza meno risorse di elaborazione. Per ulteriori informazioni, consulta [Amazon Inspector VM Scanner](#).

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità per, e. CVE-2026-39821 CVE-2026-39827 CVE-2026-39824 È stato confermato che Amazon Inspector non è interessato da queste vulnerabilità. Questi problemi verranno risolti nella prossima versione di Amazon Inspector SBOM Generator.

28 maggio 2026

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità. CVE-2026-29181 È stato confermato che il generatore SBOM di Amazon Inspector non è interessato da questa vulnerabilità. Questo problema può essere risolto aggiornando Amazon Inspector SBOM Generator alla versione 1.12.2 o successiva.

27 aprile 2026

[Aggiornamenti per il plug-in Amazon Inspector SSM](#)

Amazon Inspector è a conoscenza di uno scenario in cui il plug-in Amazon Inspector SSM potrebbe generare risultati di vulnerabilità CVE-2026-32283 per CVE-2026-33810,,,, CVE-2026-32289 e. CVE-2026-32281 CVE-2026-32288 CVE-2026-32282 È stato confermato che il plug-in Amazon Inspector SSM non è interessato da queste vulnerabilità. Questi problemi verranno risolti nella prossima versione del plug-in Amazon Inspector SSM.

13 aprile 2026

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità. CVE-2026-32280 Questo problema viene risolto eseguendo l'aggiornamento alla versione 1.12.1 di Amazon Inspector SBOM Generator.

9 aprile 2026

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità per, e. CVE-2026-33997 CVE-2026-34040 CVE-2026-33747 CVE-2026-33748 È stato confermato che il generatore SBOM di Amazon Inspector non è interessato da queste vulnerabilità. Questi problemi verranno risolti nella versione 1.12.0 di Amazon Inspector SBOM Generator.

26 marzo 2026

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità per, e. CVE-2026-25679 CVE-2026-27142 CVE-2026-27139 È stato confermato che il generatore SBOM di Amazon Inspector non è interessato da queste vulnerabilità. Questa vulnerabilità può essere risolta aggiornando la versione di Amazon Inspector SBOM Generator alla versione 1.11.2 o successiva.

11 marzo 2026

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità. CVE-2025-15558 È stato confermato che il generatore SBOM di Amazon Inspector non è interessato da CVE-2025-15558 Questa vulnerabilità può essere risolta aggiornando la versione di Amazon Inspector SBOM Generator alla versione 1.11.1 o successiva.

5 marzo 2026

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità. CVE-2025-68121 È stato confermato che il generatore SBOM di Amazon Inspector non è interessato da CVE-2025-68121 Questa vulnerabilità può essere risolta aggiornando la versione di Amazon Inspector SBOM Generator alla 1.11.0 o successiva.

2 marzo 2026

[Nuova politica gestita](#)

Amazon Inspector ha rilasciato una nuova policy gestita AmazonInspector2ManagedTelemetryPolicy che concede le autorizzazioni per le operazioni di telemetria di Amazon Inspector, consentendo al servizio di raccogliere e trasmettere i dati di inventari o dei pacchi per la scansione delle vulnerabilità. Per informazioni, consulta [gli aggiornamenti di Amazon Inspector alle policy AWS gestite](#).

5 febbraio 2026

Politica aggiornata

Amazon Inspector aggiunge nuove autorizzazioni al ruolo collegato al servizio denominato [AmazonInspector2ServiceRolePolicy](#)

3 febbraio 2026

Amazon Inspector ha aggiunto una nuova autorizzazione che consente ad Amazon Inspector di descrivere i metadati del firewall per l'analisi della raggiungibilità della rete. Inoltre, Amazon Inspector ha aggiunto un ulteriore ambito delle risorse per consentire ad Amazon Inspector di creare, aggiornare e avviare associazioni SSM con documenti SSM. `AWS-ConfigureAWSPackage`
Per ulteriori informazioni, consulta le [autorizzazioni dei Service-linked ruoli per Amazon Inspector](#).

[Aggiornamenti per il plug-in Amazon Inspector SSM e Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il plug-in Amazon Inspector SSM e Amazon Inspector SBOM Generator possono generare rilevazioni di vulnerabilità per CVE-2025-61728, CVE-2025-61730, and CVE-2025-61726. Queste vulnerabilità possono essere risolte aggiornando la versione del plug-in Amazon Inspector SSM a 1.0.2327.0 o Amazon Inspector SBOM Generator 1.10.1 o versione successiva.

29 gennaio 2026

[Aggiornamenti per il plug-in Amazon Inspector SSM e Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il plug-in Amazon Inspector SSM e Amazon Inspector SBOM Generator potrebbero generare risultati di vulnerabilità. CVE-2025-61729. È stato confermato che queste applicazioni non sono interessate da questo CVE. Attualmente stiamo lavorando a miglioramenti per risolvere questo rilevamento. Nel frattempo, i clienti possono tranquillamente ignorare o eliminare questa vulnerabilità.

3 dicembre 2025

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector è a conoscenza di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità per e. CVE-2025-47914 CVE-2025-58181. È stato confermato che il generatore SBOM di Amazon Inspector non è influenzato da questi CVE. Attualmente stiamo lavorando a miglioramenti per risolvere questi rilevamenti. Nel frattempo, i clienti possono tranquillamente ignorare o sopprimere queste vulnerabilità.

20 novembre 2025

[Nuova funzionalità](#)

Amazon Inspector ora supporta AWS Organizations politiche per l'abilitazione e la governance centralizzate tra gli account dell'organizzazione. Le policy organizzative consentono di abilitare automaticamente i tipi di scansione di Amazon Inspector in tutta l'organizzazione e prevenire modifiche non autorizzate. Per ulteriori informazioni, consulta il [tutorial Getting started](#) e [Managing multiple account](#).

19 novembre 2025

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector viene informato di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità. CVE-2025-47913 È stato confermato che il generatore SBOM di Amazon Inspector non è interessato da questo CVE ed è stato distribuito un aggiornamento per risolvere questo rilevamento.

14 novembre 2025

[Politiche aggiornate](#)

Amazon Inspector aggiunge nuove autorizzazioni alle politiche gestite e [AmazonInspector2FullAccess_v2](#) [AmazonInspector2ReadOnlyAccess](#). Le autorizzazioni consentono la visualizzazione delle politiche organizzative di Amazon Inspector e delle configurazioni di delega stabilite tramite politiche. AWS Organizations Per ulteriori informazioni, consulta [le politiche AWS gestite per Amazon Inspector](#).

14 novembre 2025

[Aggiornamenti per Amazon Inspector SBOM Generator](#)

Amazon Inspector aggiorna la versione di Amazon Inspector SBOM Generator. Per ulteriori informazioni, consulta [Versioni precedenti di Amazon Inspector SBOM Generator](#).

11 novembre 2025

Politica aggiornata

Amazon Inspector aggiunge nuove autorizzazioni al ruolo collegato al servizio denominato [AmazonInspector2ServiceRolePolicy](#). Le autorizzazioni consentono alla policy di Amazon AWS Organizations Inspector di imporre l'attivazione e la disabilitazione di Amazon Inspector. Per ulteriori informazioni, consulta le [autorizzazioni dei Service-linked ruoli per Amazon Inspector](#).

10 novembre 2025

Aggiornamenti per Amazon Inspector SBOM Generator

Amazon Inspector viene informato di uno scenario in cui il generatore SBOM di Amazon Inspector potrebbe generare risultati di vulnerabilità per e. CVE-2025-58188 CVE-2025-61725. È stato confermato che Amazon Inspector SBOM Generator non è influenzato da questi CVE e Amazon Inspector aggiorna la versione di Amazon Inspector SBOM Generator. Per ulteriori informazioni, consulta [Versioni precedenti di Amazon Inspector SBOM Generator](#).

4 novembre 2025

[Aggiornamento per il plugin](#)

Amazon Inspector viene informato di uno scenario in cui il plug-in Amazon Inspector SSM potrebbe generare risultati di vulnerabilità per e. CVE-2025-58188 CVE-2025-61725 È stato confermato che il plug-in Amazon Inspector SSM non è interessato da questi CVE ed è stato distribuito un aggiornamento per risolvere questo rilevamento.

3 novembre 2025

[Aggiornamento per il plugin](#)

Amazon Inspector viene informato di uno scenario in cui il plug-in Amazon Inspector SSM potrebbe generare un rilevamento di vulnerabilità. CVE-2025-47907 È stato confermato che il plug-in Amazon Inspector SSM non è interessato da questi CVE ed è stato distribuito un aggiornamento per risolvere questo rilevamento.

8 agosto 2025

[Nuova policy](#)

Amazon Inspector aggiunge una nuova policy gestita che fornisce l'accesso completo ad Amazon Inspector e l'accesso ad altri servizi correlati. Per ulteriori informazioni, consulta [le politiche AWS gestite per Amazon Inspector](#).

3 luglio 2025

[Funzionalità aggiornate](#)

Amazon Inspector è ora disponibile in nuove versioni. Regioni AWS Per maggiori informazioni, consulta [Regioni ed endpoint di](#) .

1 luglio 2025

[Funzionalità aggiornate](#)

Amazon Inspector aggiorna il periodo di conservazione dei risultati chiusi. Amazon Inspector rimuove i risultati dopo 3 giorni se le risorse associate vengono eliminate , terminate o non sono più idonee per la scansione. Per ulteriori informazioni, consulta [Comprendere i risultati di Amazon Inspector](#).

25 giugno 2025

[Funzionalità aggiornate](#)

Amazon Inspector aggiorna i sistemi operativi supportati per la scansione Amazon EC2 e la scansione Amazon ECR. La scansione di Amazon EC2 ora supporta la Fedora versione 42 e Ubuntu la versione 25.04. La scansione Amazon ECR ora supporta la Alpine versione 3.22, la Fedora versione 42 e la Ubuntu versione 25.04. Per ulteriori informazioni, consulta [Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector](#).

18 giugno 2025

Nuova funzionalità

Amazon Inspector ora analizza il codice sorgente delle applicazioni di prime parti, le dipendenze delle applicazioni di terze parti e Infrastrutture as Code per individuare eventuali vulnerabilità. Per ulteriori informazioni, consulta [Amazon Inspector Code Security](#).

17 giugno 2025

Aggiornamento per il plugin

Amazon Inspector viene informato di uno scenario in cui il plug-in Amazon Inspector SSM potrebbe generare un rilevamento di vulnerabilità per e. CVE-2025-0913 CVE-2025-4673 È stato confermato che il plug-in Amazon Inspector SSM non è interessato da questi CVE ed è stato distribuito un aggiornamento per risolvere questo rilevamento.

13 giugno 2025

Nuova funzionalità

Amazon Inspector ora può mostrare le immagini dei container utilizzate attivamente e l'ultima volta che le immagini dei container sono state utilizzate in un cluster. Per ulteriori informazioni, consulta [Mappare le immagini dei container ai container in esecuzione](#).

16 maggio 2025

[Aggiornamenti ai sistemi operativi supportati](#)

Amazon Inspector aggiunge il supporto per BusyBox Per ulteriori informazioni, consulta [Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector](#).

13 maggio 2025

[Politica aggiornata](#)

Amazon Inspector aggiunge una nuova autorizzazione al ruolo collegato al servizio denominato. [AmazonInspector2ServiceRolePolicy](#) Questa autorizzazione consente di descrivere e gli indirizzi IP e i gateway Internet. Per ulteriori informazioni, consulta [le politiche AWS gestite per Amazon Inspector](#).

29 aprile 2025

[Aggiornamento per il plugin](#)

Amazon Inspector viene informato di uno scenario in cui il plug-in Amazon Inspector SSM potrebbe generare un rilevamento di vulnerabilità. CVE-2025-22871 È stato confermato che il plug-in Amazon Inspector SSM non è interessato da questi CVE ed è stato distribuito un aggiornamento per risolvere questo rilevamento.

21 aprile 2025

[Aggiornamento per il plugin](#)

Amazon Inspector viene informato di uno scenario in cui il plug-in Amazon Inspector SSM potrebbe generare un rilevamento di vulnerabilità per, e. CVE-2020-8911 CVE-2020-8912 CVE-2024-45337 È stato confermato che Amazon Inspector non è interessato da questi CVE ed è stato distribuito un aggiornamento per risolvere questo rilevamento.

18 aprile 2025

[Aggiornamenti al capitolo Amazon Inspector SBOM Generator](#)

Amazon Inspector aggiorna la versione di Amazon Inspector SBOM Generator. Per ulteriori informazioni, consulta [Versioni precedenti di Amazon Inspector SBOM Generator](#).

16 aprile 2025

[Aggiornamenti al capitolo Amazon Inspector SBOM Generator](#)

Amazon Inspector aggiunge un nuovo argomento al capitolo Amazon Inspector SBOM Generator. Questo argomento descrive come tenere Sbmngen traccia delle informazioni sulla licenza in una distinta base del software. Per ulteriori informazioni, consulta la raccolta di [licenze Amazon Inspector SBOM Generator](#).

16 aprile 2025

[Aggiornamenti alle politiche gestite](#)

Amazon Inspector aggiunge autorizzazioni che consentono l'accesso in sola lettura alle azioni di Amazon ECS e Amazon EKS. Per ulteriori informazioni, consulta le [autorizzazioni dei Service-linked ruoli per Amazon Inspector](#).

25 marzo 2025

[Aggiornamenti ai sistemi operativi supportati](#)

Amazon Inspector non supporta più SUSE Linux Enterprise Server 12.5 come parte della scansione per Amazon EC2 e Amazon ECR. Per ulteriori informazioni, consulta [Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector](#).

21 marzo 2025

[Aggiornamenti ai sistemi operativi supportati](#)

Amazon Inspector aggiunge il supporto per Chainguard e alla scansione Wolfi Amazon ECR. Per ulteriori informazioni, consulta [Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector](#).

21 marzo 2025

[Aggiornamenti al sommario](#)

Amazon Inspector aggiunge un capitolo sull'etichettatura delle risorse di Amazon Inspector. Per ulteriori informazioni, consulta [Taggare le risorse di Amazon Inspector](#).

25 febbraio 2025

Aggiornamenti al sommario	Amazon Inspector aggiunge un nuovo argomento al capitolo Amazon Inspector SBOM Generator. Per ulteriori informazioni, consulta la raccolta completa di sistemi operativi Amazon Inspector SBOM Generator .	28 gennaio 2025
Funzionalità aggiornate	Amazon Inspector aggiunge nodejs202.x e python3.13 amplia il suo elenco di runtime supportati per la scansione standard Lambda. Per ulteriori informazioni, consulta Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector .	24 gennaio 2025
Funzionalità aggiornate	Amazon Inspector rimuove Oracle Linux (Oracle) 7 e SUSE Linux Enterprise Server (SLES) 15.5 dall'elenco dei sistemi operativi supportati per Amazon EC2 e Amazon ECR. Per ulteriori informazioni, consulta Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector .	31 dicembre 2024

Funzionalità aggiornate	Amazon Inspector aggiunge Ubuntu 24.10 al suo elenco di sistemi operativi supportati per Amazon EC2 e Amazon ECR. Per ulteriori informazioni, consulta Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector .	12 dicembre 2024
Aggiornamenti al sommario	Amazon Inspector aggiunge nuovi argomenti al capitolo Amazon Inspector SBOM Generator. Per ulteriori informazioni, consulta Amazon Inspector SBOM Generator .	9 dicembre 2024
Funzionalità aggiornate	Amazon Inspector aggiorna la <code>amazon:inspector:sbom_generator</code> tabella per aggiungere e rimuovere namespace. Per ulteriori informazioni, consulta Utilizzo degli spazi dei nomi CyclonedX con Amazon Inspector .	9 dicembre 2024
Funzionalità aggiornate	Amazon Inspector aggiorna la sua funzionalità di CI/CD integrazione per supportare le azioni di scansione con CodePipeline. Per ulteriori informazioni, consulta Usare le azioni di scansione di Amazon Inspector con CodePipeline .	26 novembre 2024

Aggiornamenti al sommario	Amazon Inspector riorganizza il sommario per includere un capitolo per Amazon Inspector SBOM Generator. Per ulteriori informazioni, consulta Amazon Inspector SBOM Generator .	22 novembre 2024
Funzionalità aggiornate	Amazon Inspector rimuove Fedora 39 dal suo elenco di sistemi operativi supportati per Amazon EC2 e Amazon ECR. Per ulteriori informazioni, consulta Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector .	22 novembre 2024
Funzionalità aggiornate	Amazon Inspector rimuove Alpine 3.17 dal suo elenco di sistemi operativi supportati per Amazon ECR. Per ulteriori informazioni, consulta Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector .	22 novembre 2024
Funzionalità aggiornate	Amazon Inspector aggiunge Sbomgen versioni alle versioni precedenti di Amazon Inspector SBOM Generator .	19 novembre 2024

Funzionalità aggiornate	Amazon Inspector aggiunge AL2 come runtime supportato. Per ulteriori informazioni, consulta Sistemi operativi e linguaggi di programmazione supportati per Amazon Inspector .	26 agosto 2024
Funzionalità aggiornate	Amazon Inspector ha aggiunto una nuova dichiarazione alla AmazonInspector2ServiceRolePolicy policy . La nuova istruzione consente ad Amazon Inspector di restituire i tag di funzione. AWS Lambda	31 luglio 2024
Funzionalità aggiornate	Amazon Inspector rilascia nuovi controlli di sicurezza. Per ulteriori informazioni, consulta i controlli di Amazon Inspector nella Guida per l'AWS Security Hub CSPM utente.	11 luglio 2024
Funzionalità aggiornate	Amazon Inspector SBOM Generator ora analizza i file Dockerfile e le immagini dei container Docker alla ricerca di configurazioni errate che possono introdurre vulnerabilità di sicurezza. Per ulteriori informazioni, consulta Amazon Inspector Dockerfile checks .	10 giugno 2024

Funzionalità aggiornate

Amazon Inspector aggiorna la sua [funzionalità di CI/CD integrazione](#) per supportar e CodeCatalyst le azioni, in modo da poter aggiungere e scansioni di vulnerabilità di Amazon Inspector ai flussi di lavoro. CodeCatalyst [Per ulteriori informazioni, consulta Utilizzo delle azioni. CodeCatalyst](#)

7 giugno 2024

Funzionalità aggiornate

Amazon Inspector include un'opzione per scaricare un file CSV dei risultati delle scansioni CIS. Per ulteriori informazioni, consulta [Visualizzazione e download dei risultati delle scansioni CIS nelle scansioni del Center for Internet Security \(CIS\) per le istanze Amazon EC2.](#)

3 maggio 2024

Funzionalità aggiornate

Amazon Inspector aggiorna la sua [funzionalità di CI/CD integrazione](#) per supportar eGitHub Actions, in modo da poter aggiungere scansioni di vulnerabilità di Amazon Inspector ai flussi di lavoro. GitHub Per ulteriori informazioni, consulta [Usare Amazon Inspector](#) con. GitHub Actions

29 aprile 2024

Funzionalità aggiornate	Amazon Inspector aggiorna la policy gestita AmazonInspector2FullAccess , quindi crea il ruolo collegato al servizio. AWSServiceRoleForAmazonInspector2Agentless . Ciò consente agli utenti di eseguire scansioni basate su agenti e scansioni senza agenti quando abilitano Amazon Inspector.	24 aprile 2024
Funzionalità aggiornate	Amazon Inspector aggiorna il periodo di conservazione dei risultati chiusi da 30 giorni a 7 giorni. Per ulteriori informazioni, consulta Comprendere i risultati in Amazon Inspector .	12 febbraio 2024
Funzionalità aggiornate	Amazon Inspector ha aggiunto una nuova dichiarazione alla AmazonInspector2ServiceRolePolicy policy . La nuova istruzione consente ad Amazon Inspector di avviare scansioni CIS per la tua istanza.	23 gennaio 2024
Nuova politica	Amazon Inspector ha aggiunto una nuova policy, la AmazonInspector2ManagedCisPolicypolicy , che puoi utilizzare come parte di un profilo di istanza per consentire le scansioni CIS su un'istanza.	23 gennaio 2024

Nuova funzionalità	Amazon Inspector ora aggiornerà la durata della nuova scansione ECR delle immagini dei container quando le estrai. Per modificare la durata della nuova scansione in base alle date di push o pull, consulta Configurazione della durata della nuova scansione ECR.	23 gennaio 2024
Nuova funzionalità	Amazon Inspector ora può eseguire scansioni del Center for Internet Security (CIS) su istanze EC2. Per ulteriori informazioni, consulta Scansioni CIS di Amazon Inspector .	23 gennaio 2024
Nuova funzionalità	Amazon Inspector ora può scansionare le immagini dei container nelle tue CI/CD pipeline. Per ulteriori informazioni, consulta CI/CD l'integrazione con Amazon Inspector .	30 novembre 2023
Nuova politica	Amazon Inspector ha aggiunto una nuova policy che consente ad Amazon Inspector di scansionare le istantanee di Amazon EBS dall'istanza EC2 per una scansione senza agenti. Per ulteriori informazioni sulla politica, consulta Agentless scanning .	27 novembre 2023

Nuova funzionalità	Amazon Inspector ora supporta la scansione delle istanze Linux Amazon EC2 supportate senza agenti SSM tramite la scansione senza agenti. Per ulteriori informazioni, consulta la sezione Scansione senza agente.	27 novembre 2023
Nuove risorse supportate	Amazon Inspector ora supporta la scansione di istanze Amazon EC2 per macOS. Vedi Sistemi operativi supportati: scansione di Amazon EC2 per le versioni macOS supportate.	5 ottobre 2023
Nuove regioni	Amazon Inspector è ora disponibile in Asia Pacifico (Giacarta), Africa (Città del Capo), Asia Pacifico (Osaka) ed Europa (Zurigo).	29 settembre 2023
Nuova funzionalità	Ora puoi escludere le istanze EC2 dalle scansioni di Amazon Inspector utilizzando i tag di esclusione.	14 settembre 2023
Nuova funzionalità	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono ad Amazon Inspector di scansionare le configurazioni di rete delle istanze Amazon EC2 che fanno parte dei gruppi target Elastic Load Balancing.	31 agosto 2023

Nuova funzionalità	Amazon Inspector ora fornisce dettagli di intelligence sulle vulnerabilità per rilevare le vulnerabilità dei pacchetti.	31 luglio 2023
Funzionalità aggiornate	Amazon Inspector ha aggiunto nuove autorizzazioni che consentono agli utenti di sola lettura di esportare Software Bill of Materials (SBOM) per le proprie risorse.	29 giugno 2023
Nuova funzionalità	Ora puoi esportare SBOM per le risorse scansionate da Amazon Inspector.	13 giugno 2023
Nuova funzionalità	La scansione del codice Lambda è ora disponibile a livello generale. Sono state aggiunte nuove funzionalità che consentono di crittografare il codice identificato nei risultati della scansione del codice Lambda. Inoltre, la scansione del codice Lambda ora fornisce suggerimenti per correggere le riscritture del codice.	13 giugno 2023

Funzionalità aggiornate	Amazon Inspector ha aggiunto una nuova dichiarazione alla AmazonInspector2ReadOnlyAccess policy. Le nuove istruzioni consentono o agli utenti di sola lettura di recuperare i dettagli dello stato e dei risultati della scansione del codice Lambda per il proprio account.	2 maggio 2023
Nuova funzionalità	Amazon Inspector ha aggiunto la ricerca nel database delle vulnerabilità che consente di verificare se Amazon Inspector copre un CVE specifico.	1 maggio 2023
Funzionalità aggiornate	Amazon Inspector ha aggiunto nuove autorizzazioni alla AmazonInspector2ServiceRole Policy che consentono o ad Amazon Inspector di creare canali AWS CloudTrail collegati ai servizi nel tuo account quando attivi la scansione Lambda. Ciò consente ad Amazon Inspector di monitorare CloudTrail gli eventi nel tuo account.	30 aprile 2023

Funzionalità aggiornate	Amazon Inspector ha aggiunto una nuova dichiarazione alla <code>AmazonInspector2FullAccess</code> policy. La nuova dichiarazione consente agli utenti di recuperare i dettagli delle vulnerabilità del codice rilevate dalla scansione del codice Lambda.	17 aprile 2023
Funzionalità aggiornate	Amazon Inspector ha aggiunto una nuova dichiarazione alla <code>AmazonInspector2ServiceRolePolicy</code> policy. La nuova dichiarazione consente ad Amazon Inspector di inviare informazioni ad Amazon EC2 Systems Manager sui percorsi personalizzati che hai definito per l'ispezione approfondita di Amazon EC2.	17 aprile 2023
Nuova funzionalità	Amazon Inspector aggiunge supporto aggiuntivo per le istanze Linux EC2 sotto forma di Amazon Inspector deep inspection, che analizza le istanze alla ricerca di vulnerabilità dei pacchetti nei pacchetti di linguaggi di programmazione delle applicazioni.	17 aprile 2023

Funzionalità aggiornate

[Amazon Inspector ha aggiunto una nuova dichiarazione alla AmazonInspector2ServiceRolePolicy policy](#). Le nuove istruzioni consentono ad Amazon Inspector di richiedere scansioni del codice di sviluppo nelle AWS Lambda funzioni e ricevere dati di scansione da Amazon Security. CodeGuru Inoltre, Amazon Inspector ha aggiunto le autorizzazioni per la revisione delle politiche IAM. Amazon Inspector utilizza queste informazioni per scansionare le funzioni Lambda alla ricerca di vulnerabilità del codice.

28 febbraio 2023

Nuova funzionalità

Amazon Inspector aggiunge un supporto aggiuntivo per le funzioni Lambda sotto forma di [scansione del codice Lambda, che scansiona il codice](#) sviluppatore delle tue funzioni Lambda alla ricerca di vulnerabilità di sicurezza.

28 febbraio 2023

Funzionalità aggiornate

[Amazon Inspector ha aggiunto una nuova dichiarazione alla AmazonInspector2ServiceRolePolicy policy.](#) La nuova istruzione consente ad Amazon Inspector di recuperare informazioni CloudWatch sull'ultima volta che una AWS Lambda funzione è stata richiamata. Utilizza queste informazioni per concentrare le scansioni sulle funzioni Lambda del tuo ambiente che sono state attive negli ultimi 90 giorni.

20 febbraio 2023

Funzionalità aggiornate

[Amazon Inspector ha aggiunto una nuova dichiarazione alla AmazonInspector2ServiceRolePolicy policy.](#) La nuova dichiarazione consente ad Amazon Inspector di recuperare informazioni sulle tue funzioni. AWS Lambda Amazon Inspector utilizza queste informazioni per scansionare le funzioni Lambda alla ricerca di vulnerabilità di sicurezza.

28 novembre 2022

Nuova funzionalità

Amazon Inspector aggiunge il supporto per le funzioni di [scansione AWS Lambda](#).

28 novembre 2022

Contenuti aggiornati	<p>Sono state aggiunte procedure , esempi di policy e suggerimenti per esportare i report dei risultati da Amazon Inspector a un bucket Amazon Simple Storage Service (Amazon S3).</p>	14 ottobre 2022
Nuovo contenuto	<p>Sono state aggiunte informazioni sulla valutazione della copertura di Amazon Inspector del AWS tuo ambiente utilizzando la console Amazon Inspector. Le informazioni includono descrizioni dei valori Status per le singole risorse del tuo ambiente.</p>	7 ottobre 2022
Nuova funzionalità	<p>Amazon Inspector ora fornisce ulteriori dettagli su come correggere le vulnerabilità dei pacchetti. Sono stati aggiunti nuovi campi per trovare i dettagli. I nuovi campi forniscono informazioni sulla disponibilità di una correzione e tramite un aggiornamento del pacchetto. Se è disponibile una correzione, la sezione Correzione consigliata di un risultato mostra i comandi che è possibile eseguire per apportare la correzione.</p>	2 settembre 2022

Funzionalità aggiornate

[Amazon Inspector ha aggiunto una nuova azione alla AmazonInspector2ServiceRolePolicy policy](#). La nuova azione consente ad Amazon Inspector di descrivere le esecuzioni delle associazioni SSM. Amazon Inspector ha inoltre aggiunto un ulteriore ambito delle risorse per consentire ad Amazon Inspector di creare, aggiornare, eliminare e avviare associazioni SSM con documenti SSM di proprietà. AmazonInspector2

31 agosto 2022

Nuova funzionalità

[Amazon Inspector ora supporta le scansioni delle istanze Windows](#). Amazon Inspector ora può scansionare le istanze gestite SSM che eseguono sistemi operativi supportati. Le scansioni degli Windows host vengono eseguite dal plug-in Amazon Inspector SSM, che viene installato e richiamato tramite nuove associazioni SSM create automaticamente da Amazon Inspector.

31 agosto 2022

Funzionalità aggiornate

Amazon Inspector ha aggiornato l'ambito delle risorse della [AmazonInspector2ServiceRolePolicy](#) per consentire ad Amazon Inspector di raccogliere l'inventario del software in altre partizioni. AWS

12 agosto 2022

Funzionalità aggiornate

Nella [AmazonInspector2ServiceRolePolicy](#), Amazon Inspector ha ristrutturato l'ambito delle risorse delle azioni che consentono ad Amazon Inspector di creare, eliminare e aggiornare le associazioni SSM.

10 agosto 2022

Nuova funzionalità

[Amazon Inspector ora supporta la modifica dell'impostazione della durata della nuova scansione automatica a ECR.](#) L'impostazione della durata della nuova scansione automatica di Amazon ECR determina per quanto tempo Amazon Inspector monitora continuamente le immagini inserite nei repository. Quando un'immagine è più vecchia della durata della scansione , Amazon Inspector non scansionerà più l'immagine e chiuderà tutti i risultati esistenti . A tutti i nuovi account verrà automaticamente impostata la durata della nuova scansione automatica ECR su Durata. Gli account creati in precedenza avevano una durata di scansione automatica ECR di 30 giorni, ma ora puoi scegliere tra durate di 30 giorni, 180 giorni o a vita per le scansioni.

25 giugno 2022

Nuove funzionalità

Amazon Inspector ha aggiunto una nuova policy AWS gestita, la [AmazonInspector2ReadOnlyAccesspolicy](#), per consentire l'accesso in sola lettura alle funzionalità di Amazon Inspector.

21 gennaio 2022

[Disponibilità generale](#)

Questa è la versione pubblica 29 novembre 2021
iniziale della Amazon
Inspector User Guide.

AWS Glossario

Per la AWS terminologia più recente, consultate il [AWS glossario](#) nella sezione Reference. Glossario AWS

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.