aws

Panduan Pengguna

Amazon Inspector

Amazon Inspector: Panduan Pengguna

Table of Contents

Apa itu Amazon Inspector?

Amazon Inspector adalah layanan manajemen kerentanan yang secara otomatis menemukan beban kerja dan terus memindai mereka untuk kerentanan perangkat lunak dan paparan jaringan yang tidak diinginkan. Amazon Inspector menemukan dan memindai instans Amazon EC2, gambar kontainer di Amazon ECR, dan fungsi Lambda. Ketika Amazon Inspector mendeteksi kerentanan perangkat lunak atau eksposur jaringan yang tidak diinginkan, itu menciptakan temuan, yang merupakan laporan terperinci tentang masalah tersebut. Anda dapat mengelola temuan di konsol Amazon Inspector atau API.

> **Note**
>
> Saat mengirimkan permintaan dukungan, Amazon Inspector dapat mengakses dan memproses temuan yang relevan di AWS Region tempat penyimpanannya (tetapi dalam geografi yang sama) untuk mengatasi masalah tersebut.

Topik

Fitur Amazon Inspector

Kelola beberapa akun Amazon Inspector secara terpusat

Jika AWS lingkungan Anda memiliki beberapa akun, Anda dapat mengelola lingkungan secara terpusat melalui satu akun dengan menggunakan AWS Organizations. Dengan menggunakan pendekatan ini, Anda dapat menetapkan akun sebagai akun administrator yang didelegasikan untuk Amazon Inspector.

Amazon Inspector dapat diaktifkan untuk seluruh organisasi Anda dengan satu klik. Selain itu, Anda dapat mengotomatiskan pengaktifan layanan untuk anggota future setiap kali mereka bergabung dengan organisasi Anda. Akun administrator yang didelegasikan Amazon Inspector dapat mengelola data temuan dan pengaturan tertentu untuk anggota organisasi. Ini termasuk melihat rincian temuan agregat untuk semua akun anggota, mengaktifkan atau menonaktifkan pemindaian untuk akun anggota, dan meninjau sumber daya yang dipindai dalam organisasi. AWS

Terus memindai lingkungan Anda untuk kerentanan dan eksposur jaringan

Dengan Amazon Inspector, Anda tidak perlu menjadwalkan atau mengonfigurasi pemindaian penilaian secara manual. Amazon Inspector secara otomatis menemukan dan mulai memindai sumber daya Anda yang memenuhi syarat. Amazon Inspector terus menilai lingkungan Anda sepanjang siklus hidup sumber daya Anda dengan melakukan recanning resource secara otomatis sebagai respons terhadap perubahan yang dapat menimbulkan kerentanan baru, seperti: menginstal paket baru di instans EC2, memasang tambalan, dan saat kerentanan dan eksposur umum baru (CVE) yang memengaruhi sumber daya dipublikasikan. Tidak seperti perangkat lunak pemindaian keamanan tradisional, Amazon Inspector memiliki dampak minimal pada kinerja armada Anda.

Ketika kerentanan atau jalur jaringan terbuka diidentifikasi, Amazon Inspector menghasilkan temuan yang dapat Anda selidiki. Temuan ini mencakup rincian komprehensif tentang kerentanan, sumber daya yang terpengaruh, dan rekomendasi remediasi. Jika Anda memulihkan temuan dengan tepat, Amazon Inspector secara otomatis mendeteksi remediasi dan menutup temuan tersebut.

Menilai kerentanan secara akurat dengan skor Amazon Inspector Risk

Saat Amazon Inspector mengumpulkan informasi tentang lingkungan Anda melalui pemindaian, Amazon Inspector memberikan skor keparahan yang secara khusus disesuaikan dengan lingkungan Anda. Amazon Inspector memeriksa metrik keamanan yang menyusun skor dasar National Vulnerability Database (NVD) untuk kerentanan dan menyesuaikannya sesuai dengan lingkungan komputasi Anda. Misalnya, layanan dapat menurunkan skor Amazon Inspector dari temuan untuk instans Amazon EC2 jika kerentanan dapat dieksploitasi melalui jaringan tetapi tidak ada jalur jaringan terbuka ke internet yang tersedia dari instans. Skor ini dalam format CVSS dan merupakan modifikasi dari skor Common Vulnerability Scoring System (CVSS) dasar yang disediakan oleh NVD.

Identifikasi temuan berdampak tinggi dengan dasbor Amazon Inspector

Dasbor Amazon Inspector menawarkan tampilan temuan tingkat tinggi dari seluruh lingkungan Anda. Dari dasbor, Anda dapat mengakses detail granular dari sebuah temuan. Dasbor berisi informasi yang disederhanakan tentang cakupan pemindaian di lingkungan Anda, temuan Anda yang paling penting, dan sumber daya mana yang memiliki temuan paling banyak. Panel remediasi berbasis risiko di dasbor Amazon Inspector menyajikan temuan yang memengaruhi jumlah instance dan gambar terbesar. Panel ini memudahkan untuk mengidentifikasi temuan dengan dampak terbesar pada lingkungan Anda, meninjau detail temuan, dan meninjau solusi yang disarankan.

Kelola temuan Anda menggunakan tampilan yang dapat disesuaikan

Selain dasbor, konsol Amazon Inspector menawarkan tampilan Temuan. Halaman ini mencantumkan semua temuan untuk lingkungan Anda dan memberikan rincian temuan individu. Anda dapat melihat temuan yang dikelompokkan berdasarkan kategori atau jenis kerentanan. Di setiap tampilan, Anda dapat menyesuaikan hasil lebih lanjut menggunakan filter. Anda juga dapat menggunakan filter untuk membuat aturan penekanan yang menyembunyikan temuan yang tidak diinginkan dari pandangan Anda.

Anda dapat menggunakan filter dan aturan penekanan untuk menghasilkan laporan temuan yang menunjukkan semua temuan atau pilihan temuan yang disesuaikan. Laporan dapat dibuat dalam format CSV atau JSON.

Memantau dan memproses temuan dengan layanan dan sistem lain

Untuk mendukung integrasi dengan layanan dan sistem lain, Amazon Inspector menerbitkan temuan ke Amazon EventBridge sebagai acara pencarian. EventBridge adalah layanan bus acara tanpa server yang dapat merutekan data temuan ke target seperti AWS Lambda fungsi dan topik Simple Notification Service Amazon (Amazon SNS). Dengan EventBridge, Anda dapat memantau dan memproses temuan secara nyaris real time sebagai bagian dari alur kerja keamanan dan kepatuhan yang ada.

Jika Anda telah mengaktifkan AWS Security Hub CSPM, maka Amazon Inspector juga akan mempublikasikan temuan ke Security Hub CSPM. Security Hub CSPM adalah layanan yang memberikan pandangan komprehensif tentang postur keamanan Anda di seluruh AWS lingkungan Anda dan membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik. Dengan Security Hub CSPM, Anda dapat lebih mudah memantau dan memproses temuan Anda sebagai bagian dari analisis yang lebih luas tentang postur keamanan organisasi Anda. AWS

Mengakses Amazon Inspector

Amazon Inspector tersedia di sebagian besar. Wilayah AWS Untuk daftar Wilayah tempat Amazon Inspector saat ini tersedia, lihat titik akhir dan kuota Amazon Inspector di Referensi Umum Amazon Web Services. Untuk mempelajari selengkapnya Wilayah AWS, lihat Mengelola Wilayah AWS di Referensi Umum Amazon Web Services. Di setiap Wilayah, Anda dapat bekerja dengan Amazon Inspector dengan cara berikut.

AWS Konsol Manajemen

Konsol Manajemen AWS Ini adalah antarmuka berbasis browser yang dapat Anda gunakan untuk membuat dan mengelola AWS sumber daya. Sebagai bagian dari konsol itu, konsol Amazon Inspector menyediakan akses ke akun dan sumber daya Amazon Inspector Anda. Anda dapat melakukan tugas Amazon Inspector dari konsol Amazon Inspector.

AWS alat baris perintah

Dengan alat baris AWS perintah, Anda dapat mengeluarkan perintah di baris perintah sistem Anda untuk melakukan tugas Amazon Inspector. Menggunakan baris perintah dapat lebih cepat dan lebih nyaman dibandingkan konsol. Alat baris perintah juga berguna jika Anda ingin membangun skrip yang melakukan tugas.

AWS menyediakan dua set alat baris perintah: AWS Command Line Interface (AWS CLI) dan AWS Tools for PowerShell. Untuk informasi tentang menginstal dan menggunakan AWS CLI, lihat Panduan Pengguna Antarmuka Baris AWS Perintah. Untuk informasi tentang menginstal dan menggunakan Alat untuk PowerShell, lihat Panduan AWS Tools for PowerShell Pengguna.

AWS SDKs

AWS menyediakan SDKs yang terdiri dari pustaka dan kode sampel untuk berbagai bahasa pemrograman dan platform, termasuk Java, Go, Python, C ++, dan.NET. SDKs Menyediakan akses terprogram yang nyaman ke Amazon Inspector dan lainnya. Layanan AWS SDK menangani tugas seperti menandatangani permintaan secara kriptografis, mengelola kesalahan, dan mencoba kembali permintaan secara otomatis. Untuk informasi tentang menginstal dan menggunakan AWS SDKs, lihat Alat untuk Dibangun AWS.

Amazon Inspector REST API

Amazon Inspector REST API memberi Anda akses terprogram yang komprehensif ke akun dan sumber daya Amazon Inspector Anda. Dengan API ini, Anda dapat mengirim permintaan HTTPS langsung ke Amazon Inspector. Namun, tidak seperti alat baris AWS perintah dan SDKs, penggunaan API ini mengharuskan aplikasi Anda untuk menangani detail tingkat rendah seperti membuat hash untuk menandatangani permintaan.

Memulai dengan Amazon Inspector

Bagian ini memberikan informasi yang perlu dipertimbangkan sebelum mengaktifkan Amazon Inspector dan tutorial memulai yang menjelaskan cara mengaktifkan Amazon Inspector dan melihat temuan Anda di konsol Amazon Inspector dan dengan Amazon Inspector API.

Topik

Sebelum mengaktifkan Amazon Inspector

Sebelum mengaktifkan Amazon Inspector, pertimbangkan hal berikut:

Amazon Inspector adalah layanan Regional

Data Anda disimpan di AWS Region tempat Anda mengaktifkan Amazon Inspector. Ulangi langkah-langkah di bagian pertama tutorial memulai untuk semua Wilayah AWS tempat Anda berencana menggunakan Amazon Inspector.

Amazon Inspector membuat peran terkait layanan 2 dan 2Agentless AWSService RoleForAmazonInspector AWSService RoleForAmazonInspector

Peran terkait layanan adalah peran dalam AWS Identity and Access Management (IAM) yang ditautkan ke servce. AWS AWSServiceRoleForAmazonInspector2 dan AWSServiceRoleForAmazonInspector2Agentless memungkinkan Amazon Inspector mengakses yang Layanan AWS diperlukan untuk melakukan penilaian keamanan.

Identitas IAM dengan izin administrator dapat mengaktifkan Amazon Inspector

Lindungi kredensil Anda dengan membuat pengguna dengan IAM atau. AWS IAM Identity Center Ini membantu Anda memastikan pengguna hanya memiliki izin yang diperlukan untuk mengelola Amazon Inspector. Untuk informasi selengkapnya, lihat kebijakan AWS terkelola: AmazonInspectorFullAccess.

Pemindaian hibrida diaktifkan secara otomatis

Pemindaian hibrida mencakup pemindaian berbasis agen dan pemindaian tanpa agen. Secara default, Amazon Inspector menggunakan metode pemindaian ini di semua instans Amazon EC2

yang memenuhi syarat. Untuk informasi selengkapnya, lihat Memindai EC2 instans Amazon dengan Amazon Inspector.

Pemindaian Amazon ECR dan pemindaian fungsi Lambda tidak memerlukan agen SSM

Pemindaian berbasis agen menggunakan agen SSM untuk mengumpulkan inventaris perangkat lunak. Pemindaian tanpa agen menggunakan snapshot Amazon EBS untuk mengumpulkan perangkat lunak inverntory.

> **Note**
>
> Secara default, agen SSM sudah diinstal di EC2 instans Amazon berdasarkan Amazon Machine Images. Namun, Anda mungkin perlu mengaktifkan agen SSM secara manual dalam beberapa kasus. Untuk informasi selengkapnya, lihat Bekerja dengan agen SSM di Panduan AWS Systems Manager Pengguna.

Biaya bulanan didasarkan pada beban kerja yang dipindai

Untuk informasi selengkapnya, lihat harga Amazon Inspector.

Pengaktifan multi-akun dengan AWS Organizations

Untuk organisasi yang menggunakan AWS Organizations, Amazon Inspector mendukung manajemen administrator yang didelegasikan dan pemberdayaan berbasis kebijakan organisasi. Kebijakan organisasi menyediakan tata kelola terpusat dengan pemberdayaan otomatis untuk akun baru. Untuk instruksi terperinci tentang kedua pendekatan, lihatMemulai tutorial: Mengaktifkan Amazon Inspector.

Memulai tutorial: Mengaktifkan Amazon Inspector

Topik ini menjelaskan cara mengaktifkan Amazon Inspector untuk lingkungan akun mandiri (akun anggota) dan lingkungan multi-akun (akun administrator yang didelegasikan). Saat Anda mengaktifkan Amazon Inspector, Amazon Inspector secara otomatis mulai menemukan beban kerja dan memindainya untuk mencari kerentanan perangkat lunak dan paparan jaringan yang tidak diinginkan.

Standalone account environment

Prosedur berikut menjelaskan cara mengaktifkan Amazon Inspector di konsol untuk akun anggota. Untuk mengaktifkan Amazon Inspector secara terprogram, inspector2-. enablement-with-cli

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Pilih Memulai.
3. Pilih Aktifkan Amazon Inspector.

Saat Anda mengaktifkan Amazon Inspector untuk akun mandiri, semua jenis pemindaian diaktifkan secara default. Untuk informasi tentang akun anggota, lihat Memahami akun administrator yang didelegasikan dan akun anggota di Amazon Inspector.

Multi-account (with AWS Organizations policy)

AWS Organizations kebijakan menyediakan tata kelola terpusat untuk mengaktifkan Amazon Inspector di seluruh organisasi Anda. Bila Anda menggunakan kebijakan organisasi, pengaktifan Amazon Inspector dikelola secara otomatis untuk semua akun yang tercakup dalam kebijakan, dan akun anggota tidak dapat mengubah pemindaian yang dikelola kebijakan menggunakan Amazon Inspector API.

Prasyarat

- Akun Anda harus menjadi bagian dari AWS Organizations organisasi.
- Anda harus memiliki izin untuk membuat dan mengelola kebijakan organisasi. AWS Organizations
- Akses tepercaya untuk Amazon Inspector harus diaktifkan. AWS Organizations Untuk petunjuk, lihat Mengaktifkan akses tepercaya untuk Amazon Inspector di Panduan Pengguna AWS Organizations .
- Peran terkait layanan Amazon Inspector harus ada di akun manajemen. Untuk membuatnya, aktifkan Amazon Inspector di akun manajemen atau jalankan perintah berikut dari akun manajemen:
  - `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`

- Administrator yang didelegasikan oleh Amazon Inspector harus ditunjuk.

> **Note**
>
> Tanpa peran Amazon Inspector terkait layanan dari akun manajemen dan administrator yang didelegasikan, kebijakan organisasi akan memberlakukan pemberdayaan Amazon Inspector, tetapi akun anggota tidak akan dikaitkan dengan organisasi Amazon Inspector untuk temuan terpusat dan pengelolaan akun.

Untuk mengaktifkan Amazon Inspector menggunakan kebijakan AWS Organizations

1. Tunjuk administrator yang didelegasikan untuk Amazon Inspector sebelum membuat kebijakan organisasi untuk memastikan akun anggota terkait dengan organisasi Amazon Inspector untuk visibilitas temuan terpusat. Masuk ke akun AWS Organizations manajemen, buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/v2/home, dan ikuti langkah-langkahnya. Menunjuk administrator yang didelegasikan untuk organisasi Anda AWS

   > **Note**
   >
   > Kami sangat menyarankan agar ID akun administrator yang didelegasikan AWS Organizations Amazon Inspector dan ID akun administrator yang ditunjuk Amazon Inspector tetap sama. Jika ID akun administrator yang AWS Organizations didelegasikan berbeda dari ID akun administrator yang didelegasikan Amazon Inspector, Amazon Inspector memprioritaskan ID akun yang ditunjuk Inspektur. Jika administrator yang didelegasikan Amazon Inspector tidak disetel tetapi administrator yang AWS Organizations didelegasikan disetel dan akun manajemen memiliki peran terkait layanan Amazon Inspector, Amazon Inspector secara otomatis menetapkan ID akun administrator yang didelegasikan sebagai administrator yang AWS Organizations didelegasikan Amazon Inspector.

2. Di konsol Amazon Inspector, navigasikan ke Pengaturan umum dari akun manajemen. Di bawah Kebijakan delegasi, pilih Lampirkan pernyataan. Dalam dialog Lampirkan pernyataan kebijakan, tinjau kebijakan, pilih Saya mengakui bahwa saya telah meninjau kebijakan dan memahami izin yang diberikannya, lalu pilih Lampirkan pernyataan.

> **Important**
>
> Akun manajemen harus memiliki izin berikut untuk melampirkan pernyataan kebijakan delegasi:
>
> - Izin Amazon Inspector dari kebijakan terkelola AmazonInspector2 _v2 FullAccess
> - AWS Organizations `organizations:PutResourcePolicy`izin dari kebijakan AWSOrganizationsFullAccessterkelola
>
> Jika `organizations:PutResourcePolicy` izin hilang, operasi gagal dengan kesalahan:`Failed to attach statement to the delegation policy`.

3. Selanjutnya, buat kebijakan Amazon Inspector AWS Organizations . Dari panel navigasi, pilih Manajemen, lalu pilih Konfigurasi.
4. Konfigurasikan kebijakan manajemen kerentanan. Berikan Detail dengan nama dan deskripsi (opsional) untuk kebijakan.
5. Pada halaman Configure Inspector, di bagian Detail, masukkan nama dan deskripsi untuk kebijakan tersebut. Dalam Pemilihan Kemampuan, lakukan salah satu hal berikut:
   - Pilih Konfigurasi dan aktifkan semua kemampuan (Disarankan). Ini mengaktifkan semua kemampuan Inspector termasuk EC2, ECR, standar Lambda, pemindaian kode Lambda, dan Keamanan Kode.
   - Pilih Pilih subset kemampuan. Pilih kemampuan jenis pemindaian apa pun yang harus dihidupkan.
6. Di bagian Pemilihan akun, pilih salah satu opsi berikut:
   - Pilih Semua unit dan akun organisasi jika Anda ingin menerapkan konfigurasi ke semua unit dan akun organisasi.
   - Pilih Unit dan akun organisasi tertentu jika Anda ingin menerapkan konfigurasi ke unit dan akun organisasi tertentu. Jika Anda memilih opsi ini, gunakan bilah pencarian atau pohon struktur organisasi untuk menentukan unit organisasi dan akun tempat kebijakan akan diterapkan.
   - Pilih Tidak ada unit organisasi atau akun jika Anda tidak ingin menerapkan konfigurasi ke unit organisasi atau akun mana pun.

7. Di bagian Wilayah, pilih Aktifkan semua Wilayah, Nonaktifkan semua Wilayah, atau Tentukan Wilayah.

   - Jika Anda memilih Aktifkan semua Wilayah, Anda dapat menentukan apakah akan mengaktifkan Wilayah baru secara otomatis.
   - Jika Anda memilih Nonaktifkan semua Wilayah, Anda dapat menentukan apakah akan menonaktifkan Wilayah baru secara otomatis.
   - Jika Anda memilih Tentukan Wilayah, Anda harus memilih Wilayah mana yang ingin Anda aktifkan dan nonaktifkan.

   (Opsional) Untuk pengaturan lanjutan, lihat panduan dari AWS Organizations.

   (Opsional) Untuk tag Sumber Daya, tambahkan tag sebagai pasangan nilai kunci untuk membantu Anda mengidentifikasi konfigurasi dengan mudah.

8. Pilih Berikutnya, tinjau perubahan Anda, lalu pilih Terapkan. Akun target Anda dikonfigurasi berdasarkan kebijakan. Status konfigurasi kebijakan Anda ditampilkan di bagian atas halaman Kebijakan. Setiap kemampuan memberikan status apakah itu dikonfigurasi atau di mana ada kegagalan penerapan. Untuk kegagalan apa pun, pilih tautan untuk pesan kegagalan untuk melihat detail selengkapnya. Untuk melihat kebijakan efektif di tingkat akun, Anda dapat meninjau tab Organisasi di halaman Konfigurasi tempat Anda dapat memilih akun.

Jika Amazon Inspector diaktifkan melalui kebijakan organisasi, akun yang dicakup oleh kebijakan tidak dapat menonaktifkan jenis pemindaian yang dikelola kebijakan melalui Amazon Inspector API atau konsol. Untuk informasi terperinci tentang apa yang dapat dan tidak dapat dilakukan oleh administrator yang didelegasikan dan akun anggota berdasarkan kebijakan organisasi, lihat. Mengelola beberapa akun di Amazon Inspector dengan AWS Organizations

Multi-account (without AWS Organizations policy)

> **Note**
>
> Anda harus menggunakan akun AWS Organizations manajemen untuk menyelesaikan prosedur ini. Hanya akun AWS Organizations manajemen yang dapat menunjuk administrator yang didelegasikan. Izin mungkin diperlukan untuk menunjuk administrator

yang didelegasikan. Untuk informasi selengkapnya, lihat Izin yang diperlukan untuk menetapkan administrator yang didelegasikan.

Saat Anda mengaktifkan Amazon Inspector untuk pertama kalinya, Amazon Inspector membuat `AWSServiceRoleForAmazonInspector` peran yang ditautkan layanan untuk akun tersebut. Untuk informasi tentang cara Amazon Inspector menggunakan peran terkait layanan, lihat. Menggunakan peran tertaut layanan untuk Amazon Inspector

Untuk menunjuk administrator yang didelegasikan untuk Amazon Inspector

1. Masuk ke akun AWS Organizations manajemen, lalu buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. Pilih Mulai.
3. Di bawah Administrator yang didelegasikan, masukkan ID 12 digit yang ingin Akun AWS Anda tetapkan sebagai administrator yang didelegasikan.
4. Pilih Delegasi, lalu pilih Delegasi lagi.
5. (Opsional) Jika Anda ingin mengaktifkan Amazon Inspector untuk akun AWS Organizations manajemen, pilih Aktifkan Amazon Inspector di bawah Izin layanan.

Saat Anda menunjuk administrator yang didelegasikan, semua jenis pemindaian diaktifkan untuk akun secara default. Untuk informasi tentang akun administrator yang didelegasikan, lihat Memahami akun administrator yang didelegasikan dan akun anggota di Amazon Inspector.

Jenis pemindaian otomatis di Amazon Inspector

Amazon Inspector menggunakan mesin pemindaian yang dibuat khusus yang memantau sumber daya Anda untuk kerentanan perangkat lunak yang dapat ditindaklanjuti dan paparan jaringan yang tidak diinginkan. Ketika Amazon Inspector mendeteksi kerentanan perangkat lunak atau eksposur jaringan yang tidak diinginkan, itu menciptakan temuan. Saat Anda mengaktifkan Amazon Inspector untuk pertama kalinya, akun Anda secara otomatis terdaftar di semua jenis pemindaian, yang mencakup pemindaian Amazon Amazon EC2, Pemindaian ECR Amazon, dan pemindaian standar Lambda.

> **Note**
>
> Pemindaian kode Lambda adalah lapisan opsional pemindaian fungsi Lambda yang dapat Anda aktifkan kapan saja.

Topik

Ikhtisar jenis pemindaian Amazon Inspector

Amazon Inspector menyediakan berbagai jenis pemindaian, yang berfokus pada jenis sumber daya tertentu di lingkungan Anda AWS .

Pemindaian Amazon EC2

Saat Anda mengaktifkan pemindaian Amazon EC2, Amazon Inspector memindai instans EC2 Anda untuk mencari kerentanan dan eksposur umum CVEs (), masalah paparan jaringan, masalah jangkauan jaringan, sistem operasi, dan kerentanan paket bahasa pemrograman.

Amazon Inspector melakukan pemindaian melalui penggunaan agen SSM yang diinstal pada instans Anda atau melalui snapshot instans Amazon EBS. Untuk informasi selengkapnya, lihat Memindai instans Amazon EC2 dengan Amazon Inspector. Secara default, saat Anda mengaktifkan pemindaian Amazon EC2, Anda secara otomatis mengaktifkan mode pemindaian hybrid. Untuk informasi selengkapnya, lihat Pemindaian tanpa agen.

Pemindaian ECR Amazon

Saat Anda mengaktifkan pemindaian Amazon ECR, Amazon Inspector mengonversi semua repositori di registri pribadi Anda dari repositori wadah pemindaian dasar ke repositori pemindaian yang disempurnakan. Anda dapat mengonfigurasi pengaturan ini dengan aturan penyertaan untuk memindai on-push saja atau untuk memindai repositori tertentu. Amazon Inspector hanya memindai gambar kontainer ECR yang aktif (`imageStatus`field is`ACTIVE`) di ECR. Amazon Inspector memindai semua gambar yang didorong atau dialihkan ke active (`lastActivatedAt`) di ECR dalam 30 hari terakhir atau ditarik dalam 90 hari terakhir. Amazon Inspector terus memantau gambar selama 90 hari secara default. Anda dapat mengubah pengaturan ini kapan saja. Untuk informasi selengkapnya, lihat Memindai gambar wadah Amazon Elastic Container Registry dengan Amazon Inspector.

Pemindaian standar Lambda

Saat Anda mengaktifkan pemindaian standar Lambda, Amazon Inspector menemukan semua fungsi Lambda di akun Anda dan segera memindainya untuk mencari kerentanan. Amazon Inspector memindai fungsi dan lapisan Lambda baru saat digunakan. Amazon Inspector memindainya kembali saat diperbarui atau saat baru diterbitkan. CVEs Untuk informasi lebih lanjut, pemindaian, lihatAWS Lambda Fungsi pemindaian dengan Amazon Inspector.

Pemindaian standar Lambda+pemindaian kode Lambda

Saat Anda mengaktifkan pemindaian kode Lambda, Amazon Inspector menemukan fungsi dan lapisan Lambda di akun Anda dan memindainya untuk mencari kerentanan kode. Jenis pemindaian ini mengevaluasi dependensi paket aplikasi yang digunakan dalam fungsi Lambda untuk. CVEs Saat Anda mengaktifkan jenis pemindaian ini, Anda juga mengaktifkan pemindaian standar Lambda. Untuk informasi selengkapnya, lihat AWS Lambda Fungsi pemindaian dengan Amazon Inspector.

Keamanan Kode untuk Amazon Inspector

Jenis pemindaian ini memanfaatkan mesin pemindaian Pengembang Amazon Q untuk memindai kode aplikasi pihak pertama, dependensi aplikasi pihak ketiga, dan Infrastruktur sebagai Kode untuk kerentanan Untuk informasi selengkapnya, lihat Keamanan Kode untuk Amazon Inspector.

Mengaktifkan jenis pemindaian

Anda dapat mengaktifkan jenis pemindaian kapan saja. Saat Anda mengaktifkan jenis pemindaian, Amazon Inspector mulai memindai sumber daya yang memenuhi syarat untuk jenis pemindaian.

Pemindaian Amazon EC2

Jenis pemindaian ini mengekstrak metadata dari instans Amazon EC2 sebelum membandingkan metadata dengan aturan yang dikumpulkan dari penasihat keamanan. Saat Anda mengaktifkan jenis pemindaian ini, Amazon Inspector memindai semua instans Amazon EC2 yang memenuhi syarat di akun Anda untuk mengetahui kerentanan paket dan masalah jangkauan jaringan. Setelah Anda mengaktifkan jenis pemindaian ini, Anda dapat melihat berapa banyak instance yang dipindai di tab Instans.

Pemindaian ECR Amazon

Jenis pemindaian ini memindai gambar kontainer dan repositori kontainer di Amazon ECR. Saat Anda mengaktifkan jenis pemindaian ini, Anda mengubah pengaturan konfigurasi pemindaian untuk registri pribadi Anda dari pemindaian dasar ke pemindaian yang disempurnakan. Setelah mengaktifkan pemindaian Amazon ECR, Anda dapat melihat berapa banyak gambar dan repositori yang dipindai di gambar Container dan tab repositori Container.

Pemindaian standar Lambda+pemindaian kode Lambda

Pemindaian standar Lambda adalah jenis pemindaian Lambda default. Saat Anda mengaktifkan pemindaian standar Lambda, semua fungsi Lambda Anda dipindai untuk mencari kerentanan perangkat lunak, selama mereka dipanggil atau diperbarui dalam 90 hari terakhir. Setelah Anda mengaktifkan pemindaian standar Lambda, Anda melihat berapa banyak fungsi Lambda yang dipindai di tab fungsi Lambda.

Pemindaian kode Lambda memindai kode aplikasi khusus dalam fungsi Lambda. Saat Anda mengaktifkan pemindaian kode Lambda, semua fungsi Lambda Anda akan dipindai untuk mencari kerentanan kode, selama mereka dipanggil atau diperbarui dalam 90 hari terakhir. Setelah Anda mengaktifkan pemindaian standar Lambda, Anda dapat melihat berapa banyak fungsi Lambda yang dipindai untuk kerentanan kode di tab fungsi Lambda.

> **Note**
>
> Jika Anda ingin mengaktifkan pemindaian kode Lambda, Anda harus mengaktifkan pemindaian standar Lambda terlebih dahulu.

Keamanan Kode Amazon Inspector

Jenis pemindaian ini memindai kode aplikasi pihak pertama, dependensi aplikasi pihak ketiga, dan Infrastruktur sebagai Kode untuk kerentanan. Saat Anda mengaktifkan Keamanan Kode, Amazon Inspector mulai memindai repositori kode Anda untuk kerentanan kode berdasarkan konfigurasi pemindaian Anda. Setelah mengaktifkan Amazon Inspector Code Security, Anda dapat melihat berapa banyak repositori kode yang dipindai di tab Code repositori.

Mengaktifkan pemindaian

Prosedur berikut menjelaskan cara mengaktifkan jenis pemindaian di Amazon Inspector.

> **Note**
>
> Jika Anda administrator yang didelegasikan untuk AWS organisasi, Anda dapat mengaktifkan jenis pemindaian Amazon Inspector untuk beberapa akun di beberapa Wilayah menggunakan skrip shell. Untuk informasi lebih lanjut, lihat inspector2- enablement-with-cli on. GitHub Jika tidak, selesaikan langkah-langkah berikut saat masuk sebagai administrator yang didelegasikan Amazon Inspector.

Console

Untuk mengaktifkan pemindaian

1. Buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. Menggunakan AWS Region pemilih di sudut kanan atas halaman, pilih Wilayah tempat Anda ingin mengaktifkan jenis pemindaian baru.
3. Di panel navigasi, pilih Manajemen akun.
4. Pada halaman Manajemen akun, pilih akun yang ingin Anda aktifkan jenis pemindaian.
5. Pilih Aktifkan dan pilih jenis pemindaian yang ingin Anda aktifkan.
6. (Disarankan) Ulangi langkah-langkah ini di masing-masing AWS Region yang ingin Anda aktifkan jenis pemindaian itu.

API

Jalankan operasi Aktifkan API. Dalam permintaan, berikan akun tempat IDs Anda mengaktifkan pemindaian, dan token idempotensi, dan satu atau lebih dari,`EC2`, `ECR`LAMBDA, atau `LAMBDA_CODE` `resourceTypes` untuk mengaktifkan pemindaian jenis itu.

Memindai instans Amazon EC2 dengan Amazon Inspector

Amazon Inspector Amazon EC2 scanning mengekstrak metadata dari instans EC2 Anda sebelum membandingkan metadata dengan aturan yang dikumpulkan dari penasihat keamanan. Amazon Inspector memindai instans untuk kerentanan paket dan masalah jangkauan jaringan untuk menghasilkan temuan. Amazon Inspector melakukan pemindaian jangkauan jaringan setiap 12 jam sekali dan kerentanan paket memindai pada irama variabel yang bergantung pada metode pemindaian yang terkait dengan instans EC2.

Package vulnerability scan dapat dilakukan dengan menggunakan metode pemindaian berbasis agen atau agentless. Kedua metode pemindaian ini menentukan bagaimana dan kapan Amazon Inspector mengumpulkan inventaris perangkat lunak dari instans EC2 untuk pemindaian kerentanan paket. Pemindaian berbasis agen mengumpulkan inventaris perangkat lunak menggunakan agen SSM, dan pemindaian tanpa agen mengumpulkan inventaris perangkat lunak menggunakan snapshot Amazon EBS.

Amazon Inspector menggunakan metode pemindaian yang Anda aktifkan untuk akun Anda. Saat Anda mengaktifkan Amazon Inspector untuk pertama kalinya, akun Anda secara otomatis terdaftar dalam pemindaian hibrida, yang menggunakan kedua metode pemindaian. Namun, Anda dapat mengubah pengaturan ini kapan saja. Untuk informasi tentang cara mengaktifkan jenis pemindaian, lihat Mengaktifkan jenis pemindaian. Bagian ini memberikan informasi tentang pemindaian Amazon EC2.

> **Note**
>
> Pemindaian Amazon EC2 tidak memindai direktori sistem file yang terkait dengan lingkungan virtual meskipun disediakan melalui inspeksi mendalam. Misalnya, jalur tidak `/var/lib/docker/` dipindai karena biasanya digunakan untuk waktu proses kontainer.

Pemindaian berbasis agen

Pemindaian berbasis agen dilakukan terus menerus menggunakan agen SSM pada semua instance yang memenuhi syarat. Untuk pemindaian berbasis agen, Amazon Inspector menggunakan asosiasi SSM, dan plugin yang diinstal melalui asosiasi ini, untuk mengumpulkan inventaris perangkat lunak dari instans Anda. Selain pemindaian kerentanan paket untuk paket sistem operasi, pemindaian berbasis agen Amazon Inspector juga dapat mendeteksi kerentanan paket untuk paket bahasa pemrograman aplikasi dalam instance berbasis Linux. Inspeksi mendalam Amazon Inspector untuk instans Amazon EC2 berbasis Linux

Proses berikut menjelaskan bagaimana Amazon Inspector menggunakan SSM untuk mengumpulkan inventaris dan melakukan pemindaian berbasis agen:

1. Amazon Inspector membuat asosiasi SSM di akun Anda untuk mengumpulkan inventaris dari instans Anda. Untuk beberapa jenis Instance (Windows, dan Linux), asosiasi ini menginstal plugin pada instance individual untuk mengumpulkan inventaris.
2. Menggunakan SSM, Amazon Inspector mengekstrak inventaris paket dari sebuah instance.
3. Amazon Inspector mengevaluasi inventaris yang diekstraksi dan menghasilkan temuan untuk setiap kerentanan yang terdeteksi.

> **Note**
>
> Untuk pemindaian berbasis agen, instans Amazon EC2 harus dikelola oleh SSM secara sama. Akun AWS

Contoh yang memenuhi syarat

Amazon Inspector akan menggunakan metode berbasis agen untuk memindai instance jika memenuhi ketentuan berikut:

- Instans memiliki OS yang didukung. Untuk daftar OS yang didukung, lihat kolom dukungan pemindaian berbasis agen. the section called “Sistem operasi yang didukung: Pemindaian Amazon EC2”
- Instans tidak dikecualikan dari pemindaian oleh tag pengecualian Amazon Inspector EC2.
- Instans ini dikelola SSM. Untuk petunjuk tentang memverifikasi dan mengonfigurasi agen, lihatMengkonfigurasi Agen SSM.

Perilaku pemindaian berbasis agen

Saat menggunakan metode pemindaian berbasis agen, Amazon Inspector memulai pemindaian kerentanan baru instans EC2 dalam situasi berikut:

- Saat Anda meluncurkan instans EC2 baru.
- Ketika Anda menginstal perangkat lunak baru pada instans EC2 yang ada (Linux dan Mac).
- Saat Amazon Inspector menambahkan item common vulnerabilities and exposure (CVE) baru ke database-nya, dan CVE tersebut relevan dengan instans EC2 Anda (Linux dan Mac).

Amazon Inspector memperbarui bidang yang dipindai terakhir untuk instans EC2 saat pemindaian awal selesai. Setelah ini, bidang Last scanned diperbarui saat Amazon Inspector mengevaluasi inventaris SSM (setiap 30 menit secara default), atau saat instance dipindai ulang karena CVE baru yang memengaruhi instance tersebut ditambahkan ke database Amazon Inspector.

Anda dapat memeriksa kapan instans EC2 terakhir dipindai untuk kerentanan dari tab Instans di halaman Manajemen akun atau dengan menggunakan perintah. ListCoverage

Mengkonfigurasi Agen SSM

Agar Amazon Inspector mendeteksi kerentanan perangkat lunak untuk instans Amazon EC2 menggunakan metode pemindaian berbasis agen, instans harus berupa instans terkelola di Amazon EC2 Systems Manager (SSM). Instans terkelola SSM memiliki Agen SSM yang diinstal dan dijalankan, dan SSM memiliki izin untuk mengelola instance. Jika Anda sudah menggunakan SSM untuk mengelola instans Anda, tidak ada langkah lain yang diperlukan untuk pemindaian berbasis agen.

Agen SSM diinstal secara default pada instans EC2 yang dibuat dari beberapa Amazon Machine Images (). AMIs Untuk informasi selengkapnya, lihat Tentang Agen SSM di Panduan AWS Systems Manager Pengguna. Namun, meskipun sudah diinstal, Anda mungkin perlu mengaktifkan Agen SSM secara manual, dan memberikan izin SSM untuk mengelola instans Anda.

Prosedur berikut menjelaskan cara mengonfigurasi instans Amazon EC2 sebagai instans terkelola menggunakan profil instans IAM. Prosedur ini juga menyediakan tautan ke informasi yang lebih rinci di Panduan AWS Systems Manager Pengguna.

AmazonSSMManagedInstanceCoreadalah kebijakan yang disarankan untuk digunakan saat Anda melampirkan profil instance. Kebijakan ini memiliki semua izin yang diperlukan untuk pemindaian Amazon Inspector EC2.

> **Note**
>
> Anda juga dapat mengotomatiskan manajemen SSM dari semua instans EC2 Anda, tanpa menggunakan profil instans IAM menggunakan Konfigurasi Manajemen Host Default SSM. Untuk informasi selengkapnya, lihat Konfigurasi Manajemen Host Default.

Untuk mengonfigurasi SSM untuk instans Amazon EC2

1. Jika belum diinstal oleh vendor sistem operasi Anda, instal Agen SSM. Untuk informasi lebih lanjut, lihat Bekerja dengan SSM Agent.
2. Gunakan AWS CLI untuk memverifikasi bahwa Agen SSM sedang berjalan. Untuk informasi selengkapnya, lihat Memeriksa status SSM Agent dan memulai agen.
3. Berikan izin kepada SSM untuk mengelola instans Anda. Anda dapat memberikan izin dengan membuat profil instans IAM dan melampirkannya ke instans Anda. Sebaiknya gunakan kebijakan ini, karena AmazonSSMManagedInstanceCorekebijakan ini memiliki izin untuk Distributor SSM, Inventaris SSM, dan manajer SSM State, yang dibutuhkan Amazon Inspector untuk pemindaian. Untuk petunjuk cara membuat profil instans dengan izin ini dan melampirkannya ke instance, lihat Mengonfigurasi izin instans untuk Systems Manager.
4. (Opsional) Aktifkan pembaruan otomatis untuk Agen SSM. Untuk informasi selengkapnya, lihat Mengotomatiskan pembaruan ke Agen SSM.
5. (Opsional) Konfigurasikan Systems Manager untuk menggunakan titik akhir Amazon Virtual Private Cloud (Amazon VPC). Untuk informasi selengkapnya, lihat Membuat titik akhir VPC Amazon.

> **Important**
>
> Amazon Inspector memerlukan asosiasi Manajer Negara Systems Manager di akun Anda untuk mengumpulkan inventaris aplikasi perangkat lunak. Amazon Inspector secara otomatis membuat asosiasi yang disebut `InspectorInventoryCollection-do-not-delete` jika belum ada.
>
> Amazon Inspector juga memerlukan sinkronisasi data sumber daya dan secara otomatis membuat yang dipanggil `InspectorResourceDataSync-do-not-delete` jika belum ada. Untuk informasi selengkapnya, lihat Mengonfigurasi sinkronisasi data sumber daya untuk Inventaris di Panduan AWS Systems Manager Pengguna. Setiap akun dapat memiliki sejumlah sinkronisasi data sumber daya per Wilayah. Untuk informasi selengkapnya, lihat

Jumlah maksimum sinkronisasi data sumber daya (per Akun AWS per Wilayah) di titik akhir dan kuota SSM.

Sumber daya SSM dibuat untuk pemindaian

Amazon Inspector memerlukan sejumlah sumber daya SSM di akun Anda untuk menjalankan pemindaian Amazon EC2. Sumber daya berikut dibuat saat Anda pertama kali mengaktifkan pemindaian Amazon Inspector EC2:

> **Note**
>
> Jika salah satu sumber daya SSM ini dihapus saat pemindaian Amazon Inspector Amazon EC2 diaktifkan untuk akun Anda, Amazon Inspector akan mencoba membuatnya kembali pada interval pemindaian berikutnya.

`InspectorInventoryCollection-do-not-delete`

Ini adalah asosiasi Systems Manager State Manager (SSM) yang digunakan Amazon Inspector untuk mengumpulkan inventaris aplikasi perangkat lunak dari instans Amazon EC2 Anda. Jika akun Anda sudah memiliki asosiasi SSM untuk mengumpulkan inventaris`InstanceIds*`, Amazon Inspector akan menggunakannya alih-alih membuatnya sendiri.

`InspectorResourceDataSync-do-not-delete`

Ini adalah sinkronisasi data sumber daya yang digunakan Amazon Inspector untuk mengirim data inventaris yang dikumpulkan dari instans Amazon EC2 Anda ke bucket Amazon S3 yang dimiliki oleh Amazon Inspector. Untuk informasi selengkapnya, lihat Mengonfigurasi sinkronisasi data sumber daya untuk Inventaris di Panduan AWS Systems Manager Pengguna.

`InspectorDistributor-do-not-delete`

Ini adalah asosiasi SSM yang digunakan Amazon Inspector untuk memindai instance Windows. Asosiasi ini menginstal plugin Amazon Inspector SSM pada instans Windows Anda. Jika file plugin dihapus secara tidak sengaja, asosiasi ini akan menginstalnya kembali pada interval asosiasi berikutnya.

`InvokeInspectorSsmPlugin-do-not-delete`

Ini adalah asosiasi SSM yang digunakan Amazon Inspector untuk memindai instance Windows. Asosiasi ini memungkinkan Amazon Inspector untuk memulai pemindaian menggunakan plugin,

Anda juga dapat menggunakannya untuk mengatur interval khusus untuk pemindaian instance Windows. Untuk informasi selengkapnya, lihat Mengatur jadwal khusus untuk pemindaian Windows misalnya.

`InspectorLinuxDistributor-do-not-delete`

Ini adalah asosiasi SSM yang digunakan Amazon Inspector untuk inspeksi mendalam Amazon EC2 Linux. Asosiasi ini menginstal plugin Amazon Inspector SSM pada instans Linux Anda.

`InvokeInspectorLinuxSsmPlugin-do-not-delete`

Ini adalah asosiasi SSM yang digunakan Amazon Inspector untuk inspeksi mendalam Amazon EC2 Linux. Asosiasi ini memungkinkan Amazon Inspector untuk memulai pemindaian menggunakan plugin.

> **Note**
>
> Saat Anda menonaktifkan pemindaian Amazon Inspector Amazon EC2 atau inspeksi mendalam, sumber daya SSM `InvokeInspectorLinuxSsmPlugin-do-not-delete` tidak lagi dipanggil.

Pemindaian tanpa agen

Amazon Inspector menggunakan metode pemindaian tanpa agen pada instans yang memenuhi syarat saat akun Anda dalam mode pemindaian hibrid. Mode pemindaian hibrida mencakup pemindaian berbasis agen dan tanpa agen dan diaktifkan secara otomatis saat Anda mengaktifkan pemindaian Amazon EC2.

Untuk pemindaian tanpa agen, Amazon Inspector menggunakan snapshot EBS untuk mengumpulkan inventaris perangkat lunak dari instans Anda. Pemindaian tanpa agen memindai instance untuk sistem operasi dan kerentanan paket bahasa pemrograman aplikasi..

> **Note**
>
> Saat memindai instance Linux untuk kerentanan paket bahasa pemrograman aplikasi, metode tanpa agen memindai semua jalur yang tersedia, sedangkan pemindaian berbasis agen hanya memindai jalur default dan jalur tambahan yang Anda tentukan sebagai bagian darinya. Inspeksi mendalam Amazon Inspector untuk instans Amazon EC2 berbasis Linux

Hal ini dapat mengakibatkan contoh yang sama memiliki temuan yang berbeda tergantung pada apakah itu dipindai menggunakan metode berbasis agen atau metode tanpa agen.

Proses berikut menjelaskan bagaimana Amazon Inspector menggunakan snapshot EBS untuk mengumpulkan inventaris dan melakukan pemindaian tanpa agen:

1. Amazon Inspector membuat snapshot EBS dari semua volume yang dilampirkan ke instance. Saat Amazon Inspector menggunakannya, snapshot disimpan di akun Anda dan ditandai `InspectorScan` sebagai kunci tag, dan ID pemindaian unik sebagai nilai tag.
2. Amazon Inspector mengambil data dari snapshot menggunakan EBS direct APIs dan mengevaluasinya untuk kerentanan. Temuan dihasilkan untuk setiap kerentanan yang terdeteksi.
3. Amazon Inspector menghapus snapshot EBS yang dibuatnya di akun Anda.

Contoh yang memenuhi syarat

Amazon Inspector akan menggunakan metode agentless untuk memindai instance jika memenuhi ketentuan berikut:

- Instans memiliki OS yang didukung. Untuk informasi selengkapnya, lihat kolom dukungan pemindaian berbasis agen dari. the section called “Sistem operasi yang didukung: Pemindaian Amazon EC2”
- Instance memiliki status`Unmanaged EC2 instance`,`Stale inventory`, atau`No inventory`.
- Instans ini didukung oleh Amazon EBS dan memiliki salah satu format sistem file berikut:
  - `ext3`
  - `ext4`
  - `xfs`
- Instans tidak dikecualikan dari pemindaian melalui tag pengecualian Amazon EC2.
- Jumlah volume yang melekat pada instance kurang dari 8 dan memiliki ukuran gabungan yang kurang dari atau sama dengan 1200 GB.

Perilaku pemindaian tanpa agen

Saat akun Anda dikonfigurasi untuk pemindaian Hybrid, Amazon Inspector melakukan pemindaian tanpa agen pada instans yang memenuhi syarat setiap 24 jam. Amazon Inspector mendeteksi

dan memindai instans baru yang memenuhi syarat setiap jam, yang mencakup instans baru tanpa agen SSM, atau instans yang sudah ada sebelumnya dengan status yang telah berubah menjadi. `SSM_UNMANAGED`

Amazon Inspector memperbarui bidang yang dipindai terakhir untuk instans Amazon EC2 setiap kali memindai snapshot yang diekstraksi dari instance setelah pemindaian tanpa agen.

Anda dapat memeriksa kapan instans EC2 terakhir dipindai untuk kerentanan dari tab Instans di halaman Manajemen akun atau dengan menggunakan perintah. ListCoverage

Mengelola mode pemindaian

Mode pemindaian EC2 Anda menentukan metode pemindaian yang akan digunakan Amazon Inspector saat melakukan pemindaian EC2 di akun Anda. Anda dapat melihat mode pemindaian untuk akun Anda dari halaman pengaturan pemindaian EC2 di bawah Pengaturan umum.
Akun mandiri atau administrator yang didelegasikan Amazon Inspector dapat mengubah mode pemindaian. Saat Anda menyetel mode pemindaian sebagai administrator yang didelegasikan Amazon Inspector, mode pemindaian disetel untuk semua akun anggota di organisasi Anda. Amazon Inspector memiliki mode pemindaian berikut:

Pemindaian berbasis agen — Dalam mode pemindaian ini, Amazon Inspector akan secara eksklusif menggunakan metode pemindaian berbasis agen saat memindai kerentanan paket. Mode pemindaian ini hanya memindai instans terkelola SSM di akun Anda, tetapi memiliki manfaat menyediakan pemindaian berkelanjutan sebagai respons terhadap CVE baru atau perubahan pada instans. Pemindaian berbasis agen juga menyediakan Inspeksi mendalam Amazon Inspector untuk instans yang memenuhi syarat. Ini adalah mode pemindaian default untuk akun yang baru diaktifkan.

Pemindaian hibrida — Dalam mode pemindaian ini, Amazon Inspector menggunakan kombinasi metode berbasis agen dan tanpa agen untuk memindai kerentanan paket. Untuk instans EC2 yang memenuhi syarat yang memiliki agen SSM diinstal dan dikonfigurasi, Amazon Inspector menggunakan metode berbasis agen. Untuk instans yang memenuhi syarat yang tidak dikelola SSM, Amazon Inspector akan menggunakan metode tanpa agen untuk instans yang didukung EBS yang memenuhi syarat.

Untuk mengubah mode pemindaian

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/

2. Menggunakan AWS Region pemilih di sudut kanan atas halaman, pilih Wilayah tempat Anda ingin mengubah mode pemindaian EC2 Anda.
3. Dari panel navigasi samping, di bawah Pengaturan umum, pilih pengaturan pemindaian EC2.
4. Di bawah Mode Pindai, pilih Edit.
5. Pilih mode pemindaian dan kemudian pilih Simpan perubahan.

Mengecualikan instance dari pemindaian Amazon Inspector

Anda dapat mengecualikan Linux dan Windows instans dari pemindaian Amazon Inspector dengan menandai instance ini dengan kunci. `InspectorEc2Exclusion` Kunci tag tidak peka huruf besar/kecil. Termasuk nilai tag adalah opsional. Untuk informasi tentang menambahkan tag, lihat Menandai sumber daya Amazon EC2 Anda.

Saat Anda menandai instance untuk pengecualian dari pemindaian Amazon Inspector, Amazon Inspector menandai instance sebagai dikecualikan dan tidak akan membuat temuan untuknya. Namun, plugin Amazon Inspector SSM akan terus dipanggil. Untuk mencegah plugin dipanggil, Anda harus mengizinkan akses ke tag dalam metadata instance.

> **Note**
>
> Anda tidak dikenakan biaya untuk instans yang dikecualikan.

Selain itu, Anda dapat mengecualikan volume EBS terenkripsi dari pemindaian tanpa agen dengan menandai AWS KMS kunci yang digunakan untuk mengenkripsi volume tersebut dengan tag. `InspectorEc2Exclusion` Untuk informasi selengkapnya, lihat Menandai kunci.

Sistem operasi yang didukung

Amazon Inspector memindai instans Mac, Windows, dan Linux yang mendukung kerentanan dalam paket sistem operasi. Untuk instance Linux, Amazon Inspector dapat menghasilkan temuan untuk paket bahasa pemrograman aplikasi yang digunakan. Inspeksi mendalam Amazon Inspector untuk instans Amazon EC2 berbasis Linux Untuk instance Mac dan Windows hanya paket sistem operasi yang dipindai.

Untuk informasi tentang sistem operasi yang didukung, termasuk sistem operasi mana yang dapat dipindai tanpa agen SSM, lihat. Nilai status instans Amazon EC2

Inspeksi mendalam Amazon Inspector untuk instans Amazon EC2 berbasis Linux

Amazon Inspector memperluas cakupan pemindaian Amazon EC2 untuk menyertakan inspeksi mendalam. Dengan pemeriksaan mendalam, Amazon Inspector mendeteksi kerentanan paket untuk paket bahasa pemrograman aplikasi dalam instans Amazon EC2 berbasis Linux Anda. Amazon Inspector memindai jalur default untuk pustaka paket bahasa pemrograman. Namun, Anda dapat mengonfigurasi jalur khusus selain jalur yang dipindai Amazon Inspector secara default.

> **Note**
>
> Anda dapat menggunakan inspeksi mendalam dengan pengaturan Konfigurasi Manajemen Host Default. Namun, Anda harus membuat atau menggunakan peran yang dikonfigurasi dengan `ssm:GetParameter` izin `ssm:PutInventory` dan.

Untuk melakukan pemindaian inspeksi mendalam untuk instans Amazon EC2 berbasis Linux, Amazon Inspector menggunakan data yang dikumpulkan dengan plugin Amazon Inspector SSM. Untuk mengelola plugin Amazon Inspector SSM dan melakukan inspeksi mendalam untuk Linux, Amazon Inspector secara otomatis membuat asosiasi SSM di akun Anda. `InvokeInspectorLinuxSsmPlugin-do-not-delete` Amazon Inspector mengumpulkan inventaris aplikasi yang diperbarui dari instans Amazon EC2 berbasis Linux Anda setiap 6 jam.

> **Note**
>
> Inspeksi mendalam tidak didukung untuk Windows atau instance Mac.

Bagian ini menjelaskan cara mengelola inspeksi mendalam Amazon Inspector untuk instans Amazon EC2, termasuk cara menyetel jalur khusus untuk dipindai Amazon Inspector.

Topik

Mengakses atau menonaktifkan inspeksi mendalam

> **Note**
>
> Untuk akun yang mengaktifkan Amazon Inspector setelah 17 April 2023, inspeksi mendalam diaktifkan secara otomatis sebagai bagian dari pemindaian Amazon EC2.

Untuk mengelola inspeksi mendalam

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Pengaturan umum, lalu pilih pengaturan pemindaian Amazon EC2.
3. Di bawah Inspeksi mendalam instans Amazon EC2, Anda dapat mengatur jalur khusus untuk organisasi atau akun Anda sendiri.

Anda dapat memeriksa status aktivasi secara terprogram untuk satu akun dengan GetEc2DeepInspectionConfiguration API. Anda dapat memeriksa status aktivasi secara terprogram untuk beberapa akun dengan API. BatchGetMemberEc2DeepInspectionStatus

Jika Anda mengaktifkan Amazon Inspector sebelum 17 April 2023, Anda dapat mengaktifkan inspeksi mendalam melalui spanduk konsol atau API. UpdateEc2DeepInspectionConfiguration Jika Anda adalah administrator yang didelegasikan untuk organisasi di Amazon Inspector, Anda dapat menggunakan BatchUpdateMemberEc2DeepInspectionStatusAPI untuk mengaktifkan inspeksi mendalam untuk diri sendiri dan akun anggota Anda.

Anda dapat menonaktifkan inspeksi mendalam melalui UpdateEc2DeepInspectionConfigurationAPI. Akun anggota di organisasi tidak dapat menonaktifkan inspeksi mendalam. Sebagai gantinya, akun anggota harus dinonaktifkan oleh administrator yang didelegasikan menggunakan API. BatchUpdateMemberEc2DeepInspectionStatus

Jalur khusus untuk inspeksi mendalam Amazon Inspector

Anda dapat mengatur jalur khusus untuk dipindai Amazon Inspector selama inspeksi mendalam instans Amazon EC2 Linux Anda. Saat Anda menetapkan jalur kustom, Amazon Inspector memindai paket di direktori itu dan semua sub-direktori di dalamnya.

Semua akun dapat menentukan hingga 5 jalur khusus. Administrator yang didelegasikan untuk organisasi dapat menentukan 10 jalur kustom.

Amazon Inspector memindai semua jalur kustom selain jalur default berikut, yang dipindai Amazon Inspector untuk semua akun:

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

> **Note**
>
> Jalur khusus harus berupa jalur lokal. Amazon Inspector tidak memindai jalur jaringan yang dipetakan, seperti pemasangan Sistem File Jaringan atau pemasangan sistem file Amazon S3.

Memformat jalur khusus

Jalur kustom tidak boleh lebih dari 256 karakter. Berikut ini adalah contoh bagaimana jalur kustom mungkin terlihat:

Contoh jalur

`/home/usr1/project01`

> **Note**
>
> Batas paket per instance adalah 5.000. Waktu pengumpulan persediaan paket maksimum adalah 15 menit. Amazon Inspector merekomendasikan agar Anda memilih jalur khusus untuk menghindari batasan ini.

Menyetel jalur kustom di konsol Amazon Inspector dan dengan Amazon Inspector API

Prosedur berikut menjelaskan cara menyetel jalur kustom untuk inspeksi mendalam Amazon Inspector di konsol Amazon Inspector dan dengan Amazon Inspector API. Setelah Anda menetapkan jalur khusus, Amazon Inspector menyertakan jalur dalam inspeksi mendalam berikutnya.

Console

1. Masuk ke administrator yang Konsol Manajemen AWS didelegasikan, dan buka konsol Amazon Inspector di v2/home https://console.aws.amazon.com/inspector/
2. Gunakan AWS Region pemilih untuk memilih Wilayah tempat Anda ingin mengaktifkan pemindaian standar Lambda.
3. Dari panel navigasi, pilih Pengaturan umum, lalu pilih Pengaturan pemindaian EC2.
4. Di bawah Jalur khusus untuk akun Anda sendiri, pilih Edit.
5. Di kotak teks jalur, masukkan jalur kustom Anda.
6. Pilih Simpan.

API

Jalankan perintah UpdateEc2DeepInspectionConfiguration. Untuk `packagePaths` menentukan array jalur untuk memindai.

Jadwal khusus untuk inspeksi mendalam Amazon Inspector

Secara default, Amazon Inspector mengumpulkan inventaris aplikasi dari instans Amazon EC2 setiap 6 jam. Namun, Anda dapat menjalankan perintah berikut untuk mengontrol seberapa sering Amazon Inspector melakukan ini.

Contoh perintah 1: Daftar asosiasi untuk melihat ID asosiasi dan interval saat ini

Perintah berikut menunjukkan ID asosiasi untuk asosiasi`InvokeInspectorLinuxSsmPlugin-do-not-delete`.

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

Contoh perintah 2: Perbarui asosiasi untuk menyertakan interval baru

Perintah berikut menggunakan ID asosiasi untuk asosiasi`InvokeInspectorLinuxSsmPlugin-do-not-delete`. Anda dapat mengatur tarif `schedule-expression` dari 6 jam ke interval baru, seperti 12 jam.

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

> **Note**
>
> Tergantung pada kasus penggunaan Anda, jika Anda menetapkan tarif `schedule-expression` dari 6 jam ke interval seperti 30 menit, Anda dapat melebihi batas persediaan ssm harian. Hal ini menyebabkan hasil tertunda, dan Anda mungkin menemukan instans Amazon EC2 dengan status kesalahan sebagian.

Bahasa pemrograman yang didukung

Untuk instance Linux, inspeksi mendalam Amazon Inspector dapat menghasilkan temuan untuk paket bahasa pemrograman aplikasi dan paket sistem operasi.

Untuk instance Mac dan Windows, inspeksi mendalam Amazon Inspector dapat menghasilkan temuan hanya untuk paket sistem operasi.

Untuk informasi selengkapnya tentang bahasa pemrograman yang didukung, lihat Bahasa pemrograman yang didukung: Pemeriksaan mendalam Amazon EC2.

Memindai instans Windows EC2 dengan Amazon Inspector

Amazon Inspector secara otomatis menemukan semua Windows instans yang didukung dan menyertakannya dalam pemindaian berkelanjutan tanpa tindakan tambahan apa pun. Untuk informasi tentang instans mana yang didukung, lihat Sistem operasi dan bahasa pemrograman yang didukung oleh Amazon Inspector. Amazon Inspector menjalankan Windows pemindaian secara berkala. Windowscontoh dipindai pada penemuan dan kemudian setiap 6 jam. Namun, Anda dapat menyesuaikan interval pemindaian default setelah pemindaian pertama.

Saat pemindaian Amazon EC2 diaktifkan, Amazon Inspector membuat asosiasi SSM berikut untuk sumber daya Windows `InspectorDistributor-do-not-delete` Anda:,, dan. `InspectorInventoryCollection-do-not-delete` `InvokeInspectorSsmPlugin-do-not-delete` Untuk menginstal plugin Amazon Inspector SSM pada Windows instans Anda, asosiasi SSM menggunakan dokumen `InspectorDistributor-do-not-delete`

SSM dan paket Distributor `AWS-ConfigureAWSPackage` SSM. `AmazonInspector2-InspectorSsmPlugin` Untuk informasi selengkapnya, lihat Plugin Amazon Inspector SSM untuk. Windows Untuk mengumpulkan data instans dan menghasilkan temuan Amazon Inspector, asosiasi `InvokeInspectorSsmPlugin-do-not-delete` SSM menjalankan plugin Amazon Inspector SSM dengan interval 6 jam. Namun, Anda dapat menyesuaikan pengaturan ini menggunakan ekspresi cron atau rate.

> **Note**
>
> Amazon Inspector akan memperbarui file definisi Open Vulnerability and Assessment Language (OVAL) ke bucket S3. `inspector2-oval-prod-`*`your-AWS-Region`* Bucket Amazon S3 berisi definisi OVAL yang digunakan dalam pemindaian. Definisi OVAL ini tidak boleh dimodifikasi. Jika tidak, Amazon Inspector tidak akan memindai yang baru CVEs saat dirilis.

Persyaratan pemindaian Amazon Inspector untuk instans Windows

Untuk memindai Windows instance, Amazon Inspector mengharuskan instans memenuhi kriteria berikut:

- Instans ini adalah instance terkelola SSM. Untuk petunjuk tentang pengaturan instans Anda untuk pemindaian, lihatMengkonfigurasi Agen SSM.
- Sistem operasi instance adalah salah satu sistem Windows operasi yang didukung. Untuk daftar lengkap sistem operasi yang didukung, lihatNilai status instans Amazon EC2.
- Instans memiliki plugin Amazon Inspector SSM diinstal. Amazon Inspector secara otomatis menginstal plugin Amazon Inspector SSM untuk instans terkelola setelah penemuan. Lihat topik berikutnya untuk detail tentang plugin.

> **Note**
>
> Jika host Anda berjalan di VPC Amazon tanpa akses internet keluar, Windows pemindaian mengharuskan host Anda untuk dapat mengakses titik akhir Amazon S3 Regional. Untuk mempelajari cara mengonfigurasi titik akhir Amazon S3 Amazon VPC, lihat Membuat titik akhir gateway di Panduan Pengguna Amazon Virtual Private Cloud. Jika kebijakan endpoint Amazon VPC membatasi akses ke bucket S3 eksternal, Anda harus secara khusus mengizinkan akses ke bucket yang dikelola oleh Amazon Inspector yang menyimpan definisi

OVAL AWS Region yang digunakan untuk mengevaluasi instans Anda. Bucket ini memiliki format sebagai berikut:`inspector2-oval-prod-`*`REGION`*.

Mengatur jadwal khusus untuk pemindaian Windows misalnya

Anda dapat menyesuaikan waktu antara pemindaian instans Windows Amazon EC2 dengan menyetel ekspresi cron atau ekspresi laju untuk asosiasi menggunakan SSM. `InvokeInspectorSsmPlugin-do-not-delete` Untuk informasi selengkapnya, lihat Referensi: Cron dan ekspresi nilai untuk Systems Manager di Panduan AWS Systems Manager Pengguna atau gunakan petunjuk berikut.

Pilih dari contoh kode berikut untuk mengubah irama pemindaian untuk Windows instance dari default 6 jam menjadi 12 jam menggunakan ekspresi laju atau ekspresi cron.

Contoh berikut mengharuskan Anda untuk menggunakan AssociationIduntuk asosiasi bernama`InvokeInspectorSsmPlugin-do-not-delete`. Anda dapat mengambil Anda AssociationIddengan menjalankan AWS CLI perintah berikut:

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

> **Note**
>
> AssociationIdIni Regional, jadi Anda harus terlebih dahulu mengambil ID unik untuk masing-masing AWS Region. Anda kemudian dapat menjalankan perintah untuk mengubah irama pemindaian di setiap Wilayah tempat Anda ingin mengatur jadwal pemindaian khusus untuk Windows instance.

Example rate expression

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

Example cron expression

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

Memindai gambar wadah Amazon Elastic Container Registry dengan Amazon Inspector

Amazon Inspector memindai gambar kontainer yang disimpan di Amazon Elastic Container Registry untuk mencari kerentanan perangkat lunak guna menghasilkan temuan kerentanan paket. Saat mengaktifkan pemindaian Amazon ECR, Anda menetapkan Amazon Inspector sebagai layanan pemindaian pilihan untuk registri pribadi Anda.

> **Note**
>
> Amazon ECR menggunakan kebijakan registri untuk memberikan izin kepada kepala sekolah. AWS Kepala sekolah ini memiliki izin yang diperlukan untuk memanggil Amazon APIs Inspector untuk pemindaian. Saat menyetel cakupan kebijakan registri Anda, Anda tidak boleh menambahkan `ecr:*` tindakan atau `PutRegistryScanningConfiguration` masuk`deny`. Ini menghasilkan kesalahan pada tingkat registri saat mengaktifkan dan menonaktifkan pemindaian untuk Amazon ECR.

Dengan pemindaian dasar, Anda dapat mengonfigurasi repositori Anda untuk memindai saat push atau melakukan pemindaian manual. Dengan pemindaian yang disempurnakan, Anda memindai sistem operasi dan kerentanan paket bahasa pemrograman di tingkat registri. Untuk side-by-side perbandingan perbedaan antara pemindaian dasar dan yang disempurnakan, lihat FAQ Amazon Inspector.

> **Note**
>
> Pemindaian dasar disediakan dan ditagih melalui Amazon ECR. Untuk informasi selengkapnya, lihat harga Amazon Elastic Container Registry. Pemindaian yang

disempurnakan disediakan dan ditagih melalui Amazon Inspector. Untuk informasi selengkapnya, lihat harga Amazon Inspector.

Untuk informasi tentang cara mengaktifkan pemindaian Amazon ECR, lihat Mengaktifkan jenis pemindaian. Untuk informasi tentang cara melihat temuan, lihat Melihat temuan Amazon Inspector. Untuk informasi tentang cara melihat temuan dalam Amazon ECR pada tingkat gambar, lihat Pemindaian gambar di Panduan Pengguna Amazon Elastic Container Registry. Anda dapat mengelola temuan menggunakan Layanan AWS tidak tersedia untuk pemindaian dasar, seperti AWS Security Hub CSPM dan Amazon EventBridge.

Anda dapat melihat konfigurasi pemindaian untuk setiap repositori di Amazon Inspector melalui halaman cakupan dan. APIs Namun, pengaturan konfigurasi untuk pemindaian dasar versus pemindaian berkelanjutan hanya dapat dimodifikasi di Amazon ECR. Amazon Inspector menyediakan visibilitas ke pengaturan ini tetapi tidak menawarkan kemampuan modifikasi langsung. Untuk informasi selengkapnya, lihat Memindai gambar untuk kerentanan perangkat lunak di Amazon ECR di Panduan Pengguna Amazon ECR.

Bagian ini memberikan informasi tentang pemindaian Amazon ECR dan menjelaskan cara mengonfigurasi pemindaian yang disempurnakan untuk repositori Amazon ECR.

Perilaku pemindaian untuk pemindaian Amazon ECR

Saat pertama kali mengaktifkan pemindaian Amazon ECR, Amazon Inspector mendeteksi gambar yang didorong dalam 14 hari terakhir. Amazon Inspector kemudian memindai gambar dan menetapkan status pemindaian. `ACTIVE` Amazon Inspector hanya akan memindai gambar yang aktif di ECR (`imageStatus`field is). `ACTIVE` Gambar dengan status Diarsipkan di ECR (`imageStatus`field is`ARCHIVED`) tidak dipindai oleh Amazon Inspector.

Jika pemindaian berkelanjutan diaktifkan, Amazon Inspector memantau gambar selama didorong dalam 14 hari (secara default), last-in-use tanggal dalam 14 hari (secara default), atau gambar dipindai dalam durasi pemindaian ulang yang dikonfigurasi. Untuk akun Amazon Inspector yang dibuat sebelum 16 Mei 2025, konfigurasi defaultnya adalah pemindaian ulang untuk memantau gambar jika didorong atau ditarik dalam 90 hari terakhir. Untuk informasi selengkapnya, lihat Mengonfigurasi durasi pemindaian ulang Amazon ECR.

Untuk pemindaian berkelanjutan, Amazon Inspector memulai pemindaian kerentanan baru gambar kontainer dalam situasi berikut:

- Setiap kali gambar kontainer baru didorong.
- Setiap kali Amazon Inspector menambahkan item common vulnerabilities and exposure (CVE) baru ke database-nya, dan CVE tersebut relevan dengan image container tersebut (hanya pemindaian berkelanjutan).
- Setiap kali gambar kontainer dialihkan dari diarsipkan ke aktif di ECR.

Jika Anda mengonfigurasi repositori untuk pemindaian push, gambar hanya dipindai saat Anda mendorongnya.

Anda dapat memeriksa kapan gambar kontainer terakhir diperiksa untuk kerentanan dari tab Gambar kontainer di halaman Manajemen akun atau dengan menggunakan ListCoverageAPI. Amazon Inspector memperbarui bidang Terakhir dipindai di bidang gambar Amazon ECR sebagai tanggapan atas peristiwa berikut:

- Saat Amazon Inspector menyelesaikan pemindaian awal gambar kontainer.
- Saat Amazon Inspector memindai ulang image container karena item common vulnerabilities and exposure (CVE) baru yang memengaruhi image container tersebut ditambahkan ke database Amazon Inspector.

Gambar kontainer ECR yang diarsipkan

Amazon Inspector tidak memindai gambar kontainer yang diarsipkan dalam ECR (is). `imageStatus ARCHIVED` Ketika gambar aktif di ECR dialihkan ke arsip, Amazon Inspector secara otomatis menutup temuan dan kemudian menghapus temuan setelah 3 hari. Jika gambar kontainer yang diarsipkan dialihkan ke aktif di ECR, Amazon Inspector akan memicu pemindaian baru.

Memetakan gambar kontainer ke wadah yang sedang berjalan

Amazon Inspector menyediakan manajemen keamanan kontainer yang komprehensif dengan memetakan image kontainer ke container yang sedang berjalan di Amazon Elastic Container Service (Amazon ECS) Service (Amazon ECS) dan Amazon Elastic Kubernetes Service (Amazon EKS). Pemetaan ini memberikan wawasan tentang kerentanan untuk gambar pada container yang sedang berjalan.

> **Note**
>
> Kebijakan terkelola `AWSReadOnlyAccess` saja tidak memberikan izin yang cukup untuk melihat pemetaan antara image Amazon ECR dan container yang sedang berjalan. Anda memerlukan kebijakan `AWSReadOnlyAccess` dan kebijakan `AWSInspector2ReadOnlyAccess` terkelola untuk melihat informasi pemetaan gambar kontainer.

Anda dapat memprioritaskan upaya remediasi berdasarkan risiko operasional dan menjaga cakupan keamanan di seluruh ekosistem kontainer. Anda dapat melihat berapa banyak gambar kontainer yang saat ini digunakan dan gambar kontainer mana yang terakhir digunakan pada kluster Amazon ECS atau Amazon EKS dalam 24 jam terakhir. Anda juga dapat melihat berapa banyak tugas Amazon ECS dan pod Amazon EKS yang digunakan. Informasi ini dapat ditemukan di konsol Amazon Inspector pada layar detail untuk temuan gambar kontainer dan dengan `ecrImageLastInUseAt` filter `ecrImageInUseCount` dan untuk tipe `FilterCriteria`data. Untuk gambar atau akun kontainer baru, dibutuhkan waktu hingga 36 jam agar data tersedia. Setelah itu, data ini diperbarui setiap 24 jam sekali. Untuk informasi selengkapnya, lihat Melihat temuan Amazon Inspector dan Melihat detail untuk temuan Amazon Inspector.

> **Note**
>
> Data ini secara otomatis dikirim ke temuan Amazon ECR saat Anda mengaktifkan pemindaian Amazon ECR dan mengonfigurasi repositori Anda untuk pemindaian berkelanjutan. Pemindaian berkelanjutan harus dikonfigurasi di tingkat repositori Amazon ECR. Untuk informasi selengkapnya, lihat Pemindaian yang disempurnakan di Panduan Pengguna Amazon Elastic Container Registry.

Anda juga dapat memindai ulang gambar kontainer dari cluster berdasarkan tanggalnya last-in-use.

Fitur ini juga didukung di Fargate dengan Amazon ECS dan Amazon EKS.

Sistem operasi dan jenis media yang didukung

Untuk informasi tentang sistem operasi yang didukung, lihatSistem operasi yang didukung: Pemindaian Amazon ECR dengan Amazon Inspector.

Pemindaian Amazon Inspector dari repositori Amazon ECR mencakup jenis media yang didukung berikut:

Manifes gambar

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

Konfigurasi gambar

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

Lapisan gambar

- `"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 tidak mendukung jenis `"application/vnd.docker.distribution.manifest.list.v2+json"` media untuk pemindaian repositori Amazon ECR.

Mengonfigurasi durasi pemindaian ulang Amazon ECR

Pengaturan durasi pemindaian ulang Amazon ECR menentukan berapa lama Amazon Inspector terus memantau gambar kontainer di repositori. Anda mengonfigurasi durasi pemindaian ulang

untuk last-in-use tanggal gambar, tanggal tarik terakhir, dan tanggal push. Sebagai praktik terbaik, konfigurasikan durasi pemindaian ulang agar sesuai dengan lingkungan Anda.

Jika Anda sering membuat gambar, pilih durasi pemindaian yang lebih pendek. Untuk gambar yang digunakan dalam jangka waktu yang lama, pilih durasi pemindaian yang lebih lama. Durasi pemindaian default untuk akun baru, termasuk akun baru yang ditambahkan ke organisasi, adalah 14 hari.

Amazon Inspector akan terus memantau dan memindai ulang gambar selama terakhir digunakan pada cluster atau didorong dalam 14 hari (secara default). Jika gambar belum didorong atau terakhir digunakan pada container yang sedang berjalan dalam push yang dikonfigurasi dan tanggal terakhir digunakan, Amazon Inspector berhenti memantaunya. Ada opsi untuk mengubah pengaturan untuk memantau gambar dengan tanggal tarik terakhir alih-alih tanggal penggunaan terakhir, jika diperlukan. Saat Amazon Inspector berhenti memantau gambar, Amazon Inspector akan menyetel kode status pemindaian gambar menjadi tidak aktif dan kode alasan kedaluwarsa. Amazon Inspector kemudian menjadwalkan semua temuan gambar terkait ditutup.

Jika Anda meningkatkan durasi tanggal push, Amazon Inspector menerapkan perubahan ke semua gambar yang dipindai secara aktif di repositori yang dikonfigurasi untuk pemindaian berkelanjutan. Namun, gambar yang tidak aktif tetap tidak aktif, bahkan jika Anda mendorongnya dalam durasi baru.

Saat Anda mengonfigurasi durasi pemindaian ulang dari akun administrator yang didelegasikan, Amazon Inspector menerapkan pengaturan ke semua akun anggota di organisasi. Jika akun administrator yang didelegasikan tidak mengaktifkan pemindaian Amazon ECR, akun tersebut tidak dapat melihat kluster untuk image API.

Untuk gambar multi-arsitektur, pelacakan last-in-use tanggal tidak didukung. Saat menggunakan gambar multi-arsitektur, sebaiknya Anda mengonfigurasi pemindaian berdasarkan peristiwa penarikan gambar atau push alih-alih last-in-use tanggal untuk memastikan perilaku pemindaian ulang yang tepat.

> **Note**
>
> Semua pengaturan durasi pemindaian ulang yang dikonfigurasi sebelum 16 Mei 2025, akan tetap tidak berubah. Anda dapat terus menggunakan pengaturan default yang telah dikonfigurasi sebelumnya.

Durasi pemindaian ulang gambar

Durasi pemindaian ulang gambar menentukan berapa lama Amazon Inspector akan memantau gambar. Durasi pemindaian ulang gambar mencakup dua mode: Tanggal penggunaan terakhir (default) atau Tanggal tarik terakhir. Pilih Tanggal terakhir digunakan (default) jika Anda ingin menggunakan tanggal penggunaan terakhir dari aktivitas klaster Amazon ECS/Amazon EKS Anda. Pilih Tanggal tarik terakhir jika Anda ingin menggunakan tanggal tarik terakhir dari gambar Amazon ECR Anda untuk memindai ulang gambar. Opsi berikut tersedia sebagai durasi pemindaian ulang:

- 14 hari (default)
- 30 hari
- 60 hari
- 90 hari
- 180 hari

Durasi tanggal push gambar

Durasi tanggal push image menentukan berapa lama Amazon Inspector akan terus memantau gambar setelah didorong ke repositori. Opsi berikut tersedia sebagai durasi pemindaian ulang:

- 14 hari (default)
- 30 hari
- 60 hari
- 90 hari
- 180 hari
- Seumur hidup

Untuk mengonfigurasi durasi pemindaian ulang Amazon ECR

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Pilih AWS Region tempat Anda ingin mengonfigurasi durasi pemindaian ulang Amazon ECR.
3. Dari panel navigasi, pilih Pengaturan umum, lalu pilih Pengaturan pemindaian ECR.
4. Di bawah durasi pemindaian ulang ECR, pilih mode pemindaian ulang gambar, lalu pilih durasi yang sesuai.
5. Di bawah Tanggal push Image, pilih tanggal push image.

6. Pilih Simpan.

Memahami status gambar kontainer ECR

Inspector hanya memindai `ACTIVE` gambar dalam gambar kontainer ECR. Gambar kontainer ECR dalam `ARCHIVED` status tidak dipindai. Untuk mempelajari lebih lanjut tentang perilaku pemindaian, lihatPerilaku pemindaian untuk pemindaian Amazon ECR.

Ketika status gambar kontainer ECR dalam transisi ECR, `ACTIVE` Inspector menggunakan `lastActivatedAt` bidang untuk memantau durasi pemindaian ulang.

AWS Lambda Fungsi pemindaian dengan Amazon Inspector

Dukungan Amazon Inspector untuk AWS Lambda fungsi dan lapisan menyediakan penilaian kerentanan keamanan otomatis yang berkelanjutan. Amazon Inspector menawarkan dua jenis pemindaian fungsi Lambda:

Pemindaian standar Amazon Inspector Lambda

Jenis pemindaian ini adalah jenis pemindaian Lambda default. Ini memindai dependensi aplikasi dalam fungsi dan lapisan Lambda untuk kerentanan paket.

Pemindaian kode Amazon Inspector Lambda

Jenis pemindaian ini memindai kode aplikasi khusus di fungsi dan lapisan Lambda Anda untuk kerentanan kode. Anda dapat mengaktifkan pemindaian standar Lambda atau pemindaian standar Lambda dengan pemindaian kode Lambda.

Jika Anda ingin mengaktifkan pemindaian kode Lambda, Anda harus mengaktifkan pemindaian standar Lambda terlebih dahulu. Untuk informasi selengkapnya, lihat Mengaktifkan jenis pemindaian.

Saat Anda mengaktifkan pemindaian fungsi Lambda, Amazon Inspector membuat saluran terkait layanan berikut di akun Anda: dan. `cloudtrail:CreateServiceLinkedChannel` `cloudtrail:DeleteServiceLinkedChannel` Amazon Inspector mengelola saluran ini dan menggunakannya untuk memantau CloudTrail peristiwa untuk pemindaian. Saluran memungkinkan Anda untuk melihat CloudTrail acara di akun Anda seolah-olah Anda memiliki jejak CloudTrail. Sebaiknya buat jejak Anda sendiri CloudTrail untuk mengelola acara di akun Anda. Untuk informasi tentang cara melihat saluran ini, lihat Melihat saluran terkait layanan di AWS CloudTrail Panduan Pengguna.

> **Note**
>
> Amazon Inspector tidak mendukung pemindaian fungsi Lambda yang dienkripsi dengan kunci yang dikelola pelanggan. Ini berlaku untuk pemindaian standar Lambda dan pemindaian kode Lambda.

Memindai perilaku untuk pemindaian fungsi Lambda

Setelah aktivasi, Amazon Inspector memindai semua fungsi Lambda yang dipanggil atau diperbarui dalam 90 hari terakhir di akun Anda. Amazon Inspector memulai pemindaian kerentanan fungsi Lambda dalam situasi berikut:

- Segera setelah Amazon Inspector menemukan fungsi Lambda yang ada.
- Saat Anda menerapkan fungsi Lambda baru ke layanan Lambda.
- Saat Anda menerapkan pembaruan ke kode aplikasi atau dependensi fungsi Lambda yang ada atau lapisannya.
- Setiap kali Amazon Inspector menambahkan item common vulnerabilities and exposure (CVE) baru ke database-nya, dan CVE tersebut relevan dengan fungsi Anda.

Amazon Inspector memantau setiap fungsi Lambda sepanjang masa pakainya hingga dihapus atau dikecualikan dari pemindaian.

Anda dapat memeriksa kapan fungsi Lambda terakhir diperiksa untuk kerentanan dari tab fungsi Lambda di halaman Manajemen akun atau dengan menggunakan API. ListCoverage Amazon Inspector memperbarui bidang Terakhir dipindai di untuk fungsi Lambda sebagai respons terhadap peristiwa berikut:

- Saat Amazon Inspector menyelesaikan pemindaian awal fungsi Lambda.
- Saat fungsi Lambda diperbarui.
- Saat Amazon Inspector memindai ulang fungsi Lambda karena item CVE baru yang memengaruhi fungsi tersebut ditambahkan ke database Amazon Inspector.

Runtime yang didukung dan fungsi yang memenuhi syarat

Amazon Inspector mendukung runtime yang berbeda untuk pemindaian standar Lambda dan pemindaian kode Lambda. Untuk daftar runtime yang didukung untuk setiap jenis pemindaian, lihat Runtime yang didukung: Pemindaian standar Amazon Inspector Lambda danRuntime yang didukung: Pemindaian kode Amazon Inspector Lambda.

Selain memiliki runtime yang didukung, fungsi Lambda harus memenuhi kriteria berikut agar memenuhi syarat untuk pemindaian Amazon Inspector:

- Fungsi telah dipanggil atau diperbarui dalam 90 hari terakhir.
- Fungsinya ditandai`$LATEST`.
- Fungsi ini tidak dikecualikan dari pemindaian oleh tag.

> **Note**
>
> Fungsi Lambda yang belum dipanggil atau dimodifikasi dalam 90 hari terakhir secara otomatis dikecualikan dari pemindaian. Amazon Inspector akan melanjutkan pemindaian fungsi yang dikecualikan secara otomatis jika dipanggil lagi atau jika perubahan dilakukan pada kode fungsi Lambda.

Pemindaian standar Amazon Inspector Lambda

Pemindaian standar Amazon Inspector Lambda mengidentifikasi kerentanan perangkat lunak dalam dependensi paket aplikasi yang Anda tambahkan ke kode fungsi dan lapisan Lambda Anda. Misalnya, jika fungsi Lambda Anda menggunakan versi `python-jwt` paket dengan kerentanan yang diketahui, pemindaian standar Lambda akan menghasilkan temuan untuk fungsi itu.

Jika Amazon Inspector mendeteksi kerentanan dalam dependensi paket aplikasi fungsi Lambda Anda, Amazon Inspector akan menghasilkan temuan tipe Package Vulnerability yang terperinci.

Untuk petunjuk tentang mengaktifkan jenis pemindaian lihatMengaktifkan jenis pemindaian.

> **Note**
>
> Pemindaian standar Lambda tidak memindai ketergantungan AWS SDK yang diinstal secara default di lingkungan runtime Lambda. Amazon Inspector hanya memindai dependensi yang diunggah dengan kode fungsi atau diwarisi dari lapisan.

> **Note**
>
> Menonaktifkan pemindaian standar Amazon Inspector Lambda juga akan menonaktifkan pemindaian kode Amazon Inspector Lambda.

Tidak termasuk fungsi dari pemindaian standar Lambda

Anda dapat menambahkan tag ke fungsi Lambda, sehingga Anda dapat mengecualikannya dari pemindaian standar Amazon Inspector Lambda. Mengecualikan fungsi dari pemindaian dapat mencegah peringatan yang tidak dapat ditindaklanjuti. Saat Anda menandai fungsi untuk pengecualian, tag harus memiliki pasangan kunci-nilai berikut.

- Kunci: `InspectorExclusion`
- Nilai: `LambdaStandardScanning`

Topik ini menjelaskan cara menandai fungsi untuk pengecualian dari pemindaian. Untuk informasi selengkapnya tentang menambahkan tag di Lambda, lihat Menggunakan tag pada fungsi Lambda.

Untuk mengecualikan fungsi dari pemindaian

1. Masuk menggunakan kredensialmu, lalu buka konsol Lambda di. https://console.aws.amazon.com/lambda/
2. Dari panel navigasi, pilih Fungsi.
3. Pilih nama fungsi yang ingin Anda kecualikan dari pemindaian standar Amazon Inspector Lambda.
4. Pilih Konfigurasi, lalu pilih Tag.
5. Pilih Kelola tag, lalu Tambahkan tag baru.
    a. Untuk Kunci, masukkan `InspectorExclusion`.

b. Untuk Nilai, masukkan `LambdaStandardScanning`.

6. Pilih Simpan.

Pemindaian kode Amazon Inspector Lambda

> **Important**
>
> Fitur ini menangkap cuplikan fungsi Lambda untuk menyoroti kerentanan yang terdeteksi. Cuplikan ini dapat menunjukkan kredensi hardcode dan materi sensitif lainnya.

Dengan fitur ini, Amazon Inspector memindai kode aplikasi dalam fungsi Lambda untuk kerentanan kode berdasarkan praktik terbaik AWS keamanan untuk mendeteksi kebocoran data, cacat injeksi, enkripsi yang hilang, dan kriptografi yang lemah. Amazon Inspector menggunakan penalaran otomatis dan pembelajaran mesin untuk mengevaluasi kode aplikasi fungsi Lambda Anda. Ini juga menggunakan detektor internal yang dikembangkan bekerja sama dengan Amazon Q untuk mengidentifikasi pelanggaran kebijakan dan kerentanan.

Amazon Inspector menghasilkan kerentanan kode saat mendeteksi kerentanan dalam kode aplikasi fungsi Lambda Anda. Jenis temuan ini menyertakan cuplikan kode yang menunjukkan masalah dan di mana Anda dapat menemukan masalah dalam kode Anda. Ini juga menyarankan bagaimana memperbaiki masalah ini. Saran tersebut mencakup blok plug-and-play kode yang dapat Anda gunakan untuk mengganti baris kode yang rentan. Perbaikan kode ini disediakan selain panduan remediasi kode umum untuk jenis temuan ini.

Saran remediasi kode didukung oleh penalaran otomatis. Beberapa saran remediasi kode mungkin tidak berfungsi sebagaimana dimaksud. Anda bertanggung jawab atas saran remediasi kode yang Anda adopsi. Selalu tinjau saran remediasi kode sebelum mengadopsinya. Anda mungkin perlu mengeditnya untuk memastikan kode Anda berfungsi sebagaimana dimaksud. Untuk informasi selengkapnya, lihat Kebijakan AI yang Bertanggung Jawab.

Jika Anda ingin mengaktifkan pemindaian kode Lambda, Anda harus mengaktifkan pemindaian standar Lambda terlebih dahulu. Untuk informasi selengkapnya, lihat Mengaktifkan jenis pemindaian. Untuk informasi tentang yang Wilayah AWS mendukung fitur ini, lihatKetersediaan fitur khusus wilayah.

Mengenkripsi kode Anda dalam temuan kerentanan kode

Amazon Q menyimpan cuplikan kode yang terdeteksi sehubungan dengan temuan kerentanan kode menggunakan pemindaian kode Lambda. Secara default, Amazon Q mengontrol kunci yang AWS dimiliki yang digunakan untuk mengenkripsi kode Anda. Namun, Anda dapat menggunakan kunci terkelola pelanggan Anda sendiri untuk enkripsi melalui Amazon Inspector API. Lihat informasi yang lebih lengkap di Enkripsi saat istirahat untuk kode dalam temuan Anda.

Tidak termasuk fungsi dari pemindaian kode Lambda

Anda dapat menambahkan tag ke fungsi Lambda, sehingga Anda dapat mengecualikannya dari pemindaian kode Amazon Inspector Lambda. Mengecualikan fungsi dari pemindaian dapat mencegah peringatan yang tidak dapat ditindaklanjuti. Saat Anda menandai fungsi untuk pengecualian, tag harus memiliki pasangan kunci-nilai berikut.

- Kunci – `InspectorCodeExclusion`
- Nilai - `LambdaCodeScanning`

Topik ini menjelaskan cara menandai fungsi untuk pengecualian dari pemindaian kode. Untuk informasi selengkapnya tentang menambahkan tag di Lambda, lihat Menggunakan tag pada fungsi Lambda.

Untuk mengecualikan fungsi dari pemindaian kode

1. Masuk menggunakan kredensialmu, lalu buka konsol Lambda di. https://console.aws.amazon.com/lambda/
2. Dari panel navigasi, pilih Fungsi.
3. Pilih nama fungsi yang ingin Anda kecualikan dari pemindaian kode Amazon Inspector Lambda.
4. Pilih Konfigurasi, lalu pilih Tag.
5. Pilih Kelola tag, lalu Tambahkan tag baru.
    a. Untuk Kunci, masukkan `InspectorCodeExclusion`.
    b. Untuk Nilai, masukkan `LambdaCodeScanning`.
6. Pilih Simpan.

Menonaktifkan jenis pemindaian di Amazon Inspector

Saat Anda menonaktifkan jenis pemindaian, Anda kehilangan akses ke temuan apa pun yang dihasilkan oleh jenis pemindaian. Jika Anda mengaktifkan kembali jenis pemindaian, Amazon Inspector memindai semua sumber daya yang memenuhi syarat untuk menghasilkan temuan baru. Jika Anda ingin menyimpan catatan temuan Anda, Anda dapat mengekspornya ke bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) sebagai laporan temuan. Untuk informasi selengkapnya, lihat Mengekspor laporan temuan Amazon Inspector. Saat menonaktifkan jenis pemindaian, Anda mungkin mengalami perubahan berikut di AWS akun tempat Anda menonaktifkan jenis pemindaian:

Pemindaian Amazon EC2

Saat Anda menonaktifkan pemindaian Amazon Inspector Amazon EC2 untuk akun, asosiasi SSM berikut akan dihapus:

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

Selain itu, plugin Amazon Inspector SSM dihapus dari semua host. Windows Untuk informasi selengkapnya, lihat Memindai Windows contoh EC2.

Pemindaian ECR Amazon

Saat Anda menonaktifkan pemindaian Amazon ECR untuk akun, akun jenis pemindaian Amazon ECR berubah dari Pemindaian yang ditingkatkan dengan Amazon Inspector menjadi pemindaian Dasar dengan Amazon ECR.

Pemindaian standar Lambda

Saat Anda menonaktifkan pemindaian standar Lambda untuk akun, Anda menonaktifkan pemindaian kode Lambda jika jenis pemindaian diaktifkan. Anda juga menghapus saluran CloudTrail terkait layanan yang dibuat Amazon Inspector saat Anda mengaktifkan pemindaian standar Lambda.

Keamanan Kode Amazon Inspector

Saat Anda menonaktifkan Keamanan Kode untuk akun Anda, Anda menghapus semua integrasi, proyek, dan konfigurasi pemindaian yang terkait dengannya. Jika akun Anda adalah administrator yang didelegasikan untuk organisasi, Anda hanya menonaktifkan Keamanan Kode untuk akun Anda, dan akun memeber menjadi akun mandiri.

Menonaktifkan pemindaian

Menonaktifkan semua jenis pemindaian untuk akun menonaktifkan Amazon Inspector untuk akun tersebut di dalamnya. AWS Region Untuk informasi selengkapnya, lihat Menonaktifkan Amazon Inspector.

Untuk menyelesaikan prosedur ini untuk lingkungan multi-akun, ikuti langkah-langkah ini saat masuk sebagai administrator yang didelegasikan Amazon Inspector.

Console

Untuk menonaktifkan pemindaian

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dengan menggunakan AWS Region pemilih di sudut kanan atas halaman, pilih Wilayah tempat Anda ingin menonaktifkan pemindaian.
3. Di panel navigasi, pilih Manajemen akun.
4. Pilih tab Akun untuk menampilkan status pemindaian akun.
5. Pilih kotak centang setiap akun yang ingin Anda nonaktifkan pemindaian.
6. Pilih Tindakan, dan, dari opsi Nonaktifkan, pilih jenis pemindaian yang ingin Anda nonaktifkan.
7. (Disarankan) Ulangi langkah-langkah ini di masing-masing AWS Region yang ingin Anda nonaktifkan jenis pemindaian itu.

API

Jalankan operasi Nonaktifkan API. Dalam permintaan, berikan akun tempat IDs Anda menonaktifkan pemindaian, dan untuk `resourceTypes` berikan satu atau lebih,, `EC2` `ECR` `LAMBDA`, atau `LAMBDA_CODE` untuk menonaktifkan pemindaian.

Pusat Keamanan Internet (CIS) memindai sistem operasi instans Amazon EC2

Amazon Inspector CIS scan (CIS scan) benchmark sistem operasi instans Amazon EC2 Anda untuk memastikan Anda mengonfigurasinya sesuai dengan rekomendasi praktik terbaik yang ditetapkan oleh Pusat Keamanan Internet. Tolok Ukur Keamanan CIS menyediakan garis dasar konfigurasi standar industri dan praktik terbaik untuk mengonfigurasi sistem dengan aman. Anda dapat melakukan atau menjadwalkan pemindaian CIS setelah mengaktifkan pemindaian Amazon Inspector EC2 untuk akun. Untuk informasi tentang cara mengaktifkan pemindaian Amazon EC2, lihat Mengaktifkan jenis pemindaian.

> **Note**
>
> Standar CIS ditujukan untuk sistem operasi x86_64. Beberapa pemeriksaan mungkin tidak dievaluasi atau mengembalikan instruksi remediasi yang tidak valid pada sumber daya berbasis ARM.

Amazon Inspector melakukan pemindaian CIS pada instans Amazon EC2 target berdasarkan tag instans dan jadwal pemindaian yang ditentukan. Amazon Inspector melakukan serangkaian pemeriksaan instans pada setiap instans yang ditargetkan. Setiap pemeriksaan mengevaluasi apakah konfigurasi sistem Anda memenuhi rekomendasi Tolok Ukur CIS tertentu. Setiap cek memiliki ID cek CIS dan judul, yang sesuai dengan rekomendasi CIS Benchmark untuk platform tersebut. Ketika pemindaian CIS selesai, Anda dapat melihat hasilnya untuk melihat pemeriksaan instance mana yang lulus, dilewati, atau gagal untuk sistem itu.

> **Note**
>
> Untuk melakukan atau menjadwalkan pemindaian CIS, Anda harus memiliki koneksi internet yang aman. Namun, jika Anda ingin menjalankan pemindaian CIS pada instance pribadi, Anda harus menggunakan titik akhir VPC.

Topik

- Pertimbangan untuk mengelola pemindaian Amazon Inspector CIS dengan AWS Organizations
- Amazon Inspector memiliki ember Amazon S3 yang digunakan untuk pemindaian Amazon Inspector CIS
- Membuat konfigurasi pemindaian CIS
- Melihat hasil pemindaian CIS
- Mengedit konfigurasi pemindaian CIS
- Mengunduh hasil pemindaian CIS

Persyaratan instans Amazon EC2 untuk pemindaian Amazon Inspector CIS

Untuk menjalankan pemindaian CIS pada instans Amazon EC2 Anda, instans Amazon EC2 harus memenuhi kriteria berikut:

- Sistem operasi instance adalah salah satu sistem operasi yang didukung untuk pemindaian CIS. Untuk informasi selengkapnya, lihat Sistem operasi dan bahasa pemrograman yang didukung oleh Amazon Inspector.
- Instans ini adalah instans Amazon EC2 Systems Manager. Untuk informasi selengkapnya, lihat Bekerja dengan Agen SSM di Panduan AWS Systems Manager Pengguna.
- Plugin Amazon Inspector SSM diinstal pada instance. Amazon Inspector secara otomatis menginstal plugin ini pada instans yang rusak.
- Instance memiliki profil instance yang memberikan izin kepada SSM untuk mengelola instance dan Amazon Inspector untuk menjalankan pemindaian CIS untuk instance tersebut. Untuk memberikan izin ini, lampirkan ManagedCisPolicy kebijakan Amazon SSMManaged InstanceCore dan AmazonInspector2 ke peran IAM. Kemudian lampirkan peran IAM ke instance Anda sebagai profil instance. Untuk petunjuk cara membuat dan melampirkan profil instans, lihat Bekerja dengan peran IAM di Panduan Pengguna Amazon EC2.

> **Note**
>
> Anda tidak diharuskan mengaktifkan inspeksi mendalam Amazon Inspector sebelum menjalankan pemindaian CIS pada instans Amazon EC2 Anda. Jika Anda menonaktifkan inspeksi mendalam Amazon Inspector, Amazon Inspector secara otomatis menginstal Agen SSM, tetapi Agen SSM tidak akan dipanggil untuk menjalankan inspeksi mendalam lagi.

Namun, sebagai hasilnya, `InspectorLinuxDistributor-do-not-delete` asosiasi hadir di akun Anda.

Persyaratan titik akhir Amazon Virtual Private Cloud untuk menjalankan pemindaian CIS pada instans Amazon EC2 pribadi

Anda dapat menjalankan pemindaian CIS pada instans Amazon EC2 melalui jaringan Amazon. Namun, jika Anda ingin menjalankan pemindaian CIS pada instans Amazon EC2 pribadi, Anda harus membuat titik akhir Amazon VPC. Titik akhir berikut diperlukan saat Anda membuat titik akhir Amazon VPC untuk Systems Manager:

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

Untuk informasi selengkapnya, lihat Membuat titik akhir Amazon VPC untuk Systems Manager di Panduan Pengguna.AWS Systems Manager

> **Note**
>
> Saat ini, beberapa Wilayah AWS tidak mendukung `amazonaws.com.region.inspector2` titik akhir.

Menjalankan pemindaian CIS

Anda dapat menjalankan pemindaian CIS sekali sesuai permintaan atau sebagai pemindaian berulang yang dijadwalkan. Untuk menjalankan pemindaian, pertama-tama Anda membuat konfigurasi pemindaian.

Saat membuat konfigurasi pemindaian, Anda menentukan pasangan nilai kunci tag yang akan digunakan untuk menargetkan instance. Jika Anda adalah administrator yang didelegasikan Amazon Inspector untuk organisasi, Anda dapat menentukan beberapa akun dalam konfigurasi pemindaian,

dan Amazon Inspector akan mencari instance dengan tag yang ditentukan di masing-masing akun tersebut. Anda memilih level CIS Benchmark untuk pemindaian. Untuk setiap benchmark, CIS mendukung profil level 1 dan level 2 yang dirancang untuk memberikan garis dasar untuk berbagai tingkat keamanan yang mungkin diperlukan oleh lingkungan yang berbeda.

- Level 1 — merekomendasikan pengaturan keamanan dasar penting yang dapat dikonfigurasi pada sistem apa pun. Menerapkan pengaturan ini harus menyebabkan sedikit atau tidak ada gangguan layanan. Tujuan dari rekomendasi ini adalah untuk mengurangi jumlah titik masuk ke sistem Anda, mengurangi risiko keamanan siber Anda secara keseluruhan.
- Level 2 — merekomendasikan pengaturan keamanan yang lebih canggih untuk lingkungan dengan keamanan tinggi. Menerapkan pengaturan ini membutuhkan perencanaan dan koordinasi untuk meminimalkan risiko dampak bisnis. Tujuan dari rekomendasi ini adalah untuk membantu Anda mencapai kepatuhan terhadap peraturan.

Level 2 memperluas level 1. Saat Anda memilih Level 2, Amazon Inspector memeriksa semua konfigurasi yang direkomendasikan untuk level 1 dan level 2.

Setelah menentukan parameter untuk pemindaian Anda, Anda dapat memilih apakah akan menjalankannya sebagai pemindaian satu kali, yang berjalan setelah Anda menyelesaikan konfigurasi, atau pemindaian berulang. Pemindaian berulang dapat berjalan setiap hari, mingguan, atau bulanan, pada waktu pilihan Anda.

> **Tip**
>
> Sebaiknya pilih hari dan waktu yang paling tidak memengaruhi sistem Anda saat pemindaian sedang berjalan.

Pertimbangan untuk mengelola pemindaian Amazon Inspector CIS dengan AWS Organizations

Saat Anda menjalankan pemindaian CIS di suatu organisasi, administrator dan akun anggota yang didelegasikan Amazon Inspector berinteraksi dengan konfigurasi pemindaian CIS dan hasil pemindaian secara berbeda.

Bagaimana administrator yang didelegasikan Amazon Inspector dapat berinteraksi dengan konfigurasi pemindaian CIS dan hasil pemindaian

Ketika administrator yang didelegasikan membuat konfigurasi pemindaian, baik untuk semua akun atau akun anggota tertentu, organisasi memiliki konfigurasi tersebut. Konfigurasi pemindaian yang dimiliki organisasi memiliki ARN yang menentukan ID organisasi sebagai pemilik:

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

Administrator yang didelegasikan dapat mengelola konfigurasi pemindaian yang dimiliki organisasi, bahkan jika akun lain membuatnya.

Administrator yang didelegasikan dapat melihat hasil pemindaian untuk akun apa pun di organisasinya.

Jika administrator yang didelegasikan membuat konfigurasi pemindaian dan menetapkan `SELF` sebagai akun target, administrator yang didelegasikan memiliki konfigurasi pemindaian, meskipun mereka meninggalkan organisasi. Namun, administrator yang didelegasikan tidak dapat mengubah target konfigurasi pemindaian dengan `SELF` target.

> **Note**
>
> Adminstrator yang didelegasikan tidak dapat menambahkan tag ke konfigurasi pemindaian CIS yang dimiliki organisasi.

Bagaimana akun anggota Amazon Inspector dapat berinteraksi dengan konfigurasi pemindaian CIS dan hasil pemindaian

Ketika akun anggota membuat konfigurasi pemindaian CIS, ia memiliki konfigurasi. Namun, administrator yang didelegasikan dapat melihat konfigurasi. Jika akun anggota meninggalkan organisasi, administrator yang didelegasikan tidak akan dapat melihat konfigurasi.

> **Note**
>
> Administrator yang didelegasikan tidak dapat mengedit konfigurasi pemindaian yang dibuat oleh akun anggota.

Akun anggota, administrator yang didelegasikan `SELF` sebagai target, dan akun mandiri semuanya memiliki konfigurasi pemindaian yang mereka buat. Konfigurasi pemindaian ini memiliki ARN yang menunjukkan ID akun sebagai pemilik:

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

Akun anggota dapat melihat hasil pemindaian di akun mereka, termasuk hasil pemindaian dari pindaian CIS yang dijadwalkan administrator yang didelegasikan.

Amazon Inspector memiliki ember Amazon S3 yang digunakan untuk pemindaian Amazon Inspector CIS

Open Vulnerability and Assessment Language (OVAL) adalah upaya keamanan informasi yang menstandarisasi cara menilai dan melaporkan keadaan mesin sistem komputer. Tabel berikut mencantumkan semua bucket Amazon S3 milik Amazon Inspector dengan definisi OVAL yang digunakan untuk pemindaian CIS. Amazon Inspector menampilkan file definisi OVAL yang diperlukan untuk pemindaian CIS. Bucket Amazon S3 milik Amazon Inspector harus diizinkan masuk jika perlu. VPCs

> **Note**
>
> Detail untuk masing-masing bucket Amazon S3 milik Amazon Inspector berikut tidak dapat berubah. Namun, tabel mungkin diperbarui untuk mencerminkan yang baru didukung Wilayah AWS. Anda tidak dapat menggunakan bucket Amazon S3 milik Amazon Inspector untuk operasi Amazon S3 lainnya atau di bucket Amazon S3 Anda sendiri.

Ember CIS	AWS Region
`cis-datasets-prod-arn-5908f6f`	Eropa (Stockholm)
`cis-datasets-prod-bah-8f88801`	Timur Tengah (Bahrain)
`cis-datasets-prod-bjs-0f40506`	Tiongkok (Beijing)
`cis-datasets-prod-bom-435a167`	Asia Pasifik (Mumbai)
`cis-datasets-prod-cdg-f3a9c58`	Eropa (Paris)
`cis-datasets-prod-cgk-09eb12f`	Asia Pasifik (Jakarta)

Ember CIS	AWS Region
cis-datasets-prod-cmh-63030b9	AS Timur (Ohio)
cis-datasets-prod-cpt-02c5c6f	Africa (Cape Town)
cis-datasets-prod-dub-984936f	Eropa (Irlandia)
cis-datasets-prod-fra-6eb96eb	Eropa (Frankfurt)
cis-datasets-prod-gru-de69f99	Amerika Selatan (Sao Paulo)
cis-datasets-prod-hkg-8e30800	Asia Pasifik (Hong Kong)
cis-datasets-prod-iad-8438411	AS Timur (Virginia Utara)
cis-datasets-prod-icn-f4eff1c	Asia Pasifik (Seoul)
cis-datasets-prod-kix-5743b21	Asia Pasifik (Osaka)
cis-datasets-prod-lhr-8b1fbd0	Eropa (London)
cis-datasets-prod-mxp-7b1bbce	Europe (Milan)
cis-datasets-prod-nrt-464f684	Asia Pasifik (Tokyo)
cis-datasets-prod-osu-5bead6f	AWS GovCloud (AS-Timur)
cis-datasets-prod-pdt-adadf9c	AWS GovCloud (AS-Barat)
cis-datasets-prod-pdx-acfb052	AS Barat (Oregon)
cis-datasets-prod-sfo-1515ba8	AS Barat (California Utara)
cis-datasets-prod-sin-309725b	Asia Pasifik (Singapura)
cis-datasets-prod-syd-f349107	Asia Pacific (Sydney)
cis-datasets-prod-yul-5e0c95e	Kanada (Pusat)
cis-datasets-prod-zhy-5a8eacb	Tiongkok (Ningxia)

Ember CIS	AWS Region
`cis-datasets-prod-zrh-67e0e3d`	Europe (Zurich)

Membuat konfigurasi pemindaian CIS

Topik ini menjelaskan cara membuat konfigurasi pemindaian CIS.

Untuk menjalankan pemindaian CIS

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan AWS Region dropdown untuk memilih AWS Region tempat Anda ingin menjalankan pemindaian CIS.
3. Dari panel navigasi, pilih Pemindaian sesuai permintaan, lalu pilih pemindaian CIS.
4. Pilih Buat pemindaian baru.
5. Untuk nama konfigurasi Pindai, masukkan nama konfigurasi Pindai.
6. Untuk tag sumber daya Target, masukkan Kunci dan Nilai yang sesuai untuk instance yang ingin Anda pindai. Anda dapat menentukan hingga lima nilai berbeda untuk setiap kunci dan total 25 tag untuk disertakan dalam pemindaian.
7. Untuk tingkat CIS Benchmark, Anda dapat memilih Level 1 untuk konfigurasi keamanan dasar atau Level 2 untuk konfigurasi keamanan lanjutan.
8. Untuk akun Target, tentukan akun mana yang akan disertakan dalam pemindaian CIS. Untuk informasi selengkapnya, lihat Pertimbangan untuk mengelola pemindaian Amazon Inspector CIS dengan AWS Organizations.

   Jika akun Anda adalah akun administrator yang didelegasikan, Anda dapat memilih Semua akun atau Tentukan akun. Opsi Semua akun menargetkan semua akun di organisasi Anda. Tentukan akun hanya menargetkan akun individual di organisasi Anda. Jika Anda memilih opsi ini, Anda dapat menentukan lebih dari satu akun dengan memisahkan nomor akun dengan koma. Anda juga dapat memasukkan `SELF` bukan ID akun untuk membuat konfigurasi pemindaian untuk akun Anda

   Jika akun Anda adalah akun mandiri atau akun anggota dalam suatu organisasi, Anda dapat memilih Self untuk membuat konfigurasi pemindaian untuk akun Anda.

9. Untuk Jadwal, pilih Pemindaian satu kali, yang berjalan segera setelah Anda selesai membuat konfigurasi pemindaian, atau Pemindaian berulang, yang berjalan pada waktu yang Anda tentukan.
10. Konfirmasikan pilihan Anda, lalu pilih Buat.

Melihat hasil pemindaian CIS

Amazon Inspector membuat tugas pemindaian untuk setiap konfigurasi pemindaian yang berjalan dan mengumpulkan hasil pemindaian dengan ID pemindaian unik. Hasil pemindaian CIS tersedia selama 90 hari. Anda dapat melihat hasil pemindaian CIS dengan cek atau sumber daya yang dipindai:

- Hasil pemindaian dikumpulkan berdasarkan pemeriksaan — Kelompokkan hasil pemindaian oleh setiap pemeriksaan individu yang dilakukan selama pemindaian. Untuk setiap pemeriksaan, Anda mendapatkan laporan tentang berapa banyak sumber daya yang gagal, dilewati, atau dilewati.
- Hasil pemindaian dikumpulkan berdasarkan sumber daya yang dipindai - Kelompokkan hasil pemindaian oleh setiap sumber daya yang dipindai target pemindaian selama pemindaian. Untuk setiap sumber daya, Anda mendapatkan laporan berapa banyak pemeriksaan yang gagal, dilewati, atau dilewati sumber daya.

Topik ini menjelaskan cara melihat hasil untuk pemindaian CIS.

Untuk melihat hasil pemindaian

1. Masuk menggunakan kredensi Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan AWS Region dropdown untuk memilih AWS Region tempat Anda membuat konfigurasi pemindaian CIS Anda.
3. Dari panel navigasi, pilih Pemindaian sesuai permintaan, lalu pilih pemindaian CIS.
4. Pilih tab Hasil Pindai.
5. Di bawah kolom Dijadwalkan menurut, pilih ID jadwal pemindaian yang ingin Anda lihat. Atau pilih baris dengan ID jadwal pemindaian yang ingin Anda lihat, lalu pilih Lihat detail.
6. Pilih Cek untuk melihat setiap pemeriksaan yang dilakukan atau Sumber daya yang dipindai untuk melihat setiap sumber daya yang dipindai yang ditargetkan selama pemindaian.

Anda juga dapat melihat detail untuk pemindaian CIS terjadwal.

Untuk melihat detail pemindaian CIS terjadwal

1. Masuk menggunakan kredensi Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan AWS Region dropdown untuk memilih AWS Region tempat Anda membuat konfigurasi pemindaian CIS Anda.
3. Dari panel navigasi, pilih Pemindaian sesuai permintaan, lalu pilih pemindaian CIS.
4. Pilih tab Terjadwal.
5. Di bawah kolom Nama konfigurasi Pindai, pilih nama konfigurasi pemindaian yang ingin Anda lihat. Atau pilih baris dengan konfigurasi pemindaian yang ingin Anda lihat, lalu pilih Lihat detail.

Mengedit konfigurasi pemindaian CIS

Topik ini menjelaskan cara mengedit konfigurasi pemindaian CIS.

Untuk mengedit konfigurasi pemindaian CIS

1. Masuk menggunakan kredensi Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan AWS Region dropdown untuk memilih AWS Region tempat Anda membuat konfigurasi pemindaian CIS Anda.
3. Dari panel navigasi, pilih Pemindaian sesuai permintaan, lalu pilih pemindaian CIS.
4. Pilih tab Terjadwal.
5. Pilih baris dengan konfigurasi pemindaian yang ingin Anda edit, lalu pilih Edit.

Mengunduh hasil pemindaian CIS

Anda dapat mengunduh PDF atau CSV dari pemindaian CIS menggunakan konsol Amazon Inspector atau API.

> **Note**
>
> Anda hanya dapat mengunduh file CSV hasil pemindaian CIS Anda untuk pemindaian CIS yang dikumpulkan setelah 05/03/2024.

Topik ini menjelaskan cara mengunduh pemindaian CIS menggunakan konsol Amazon Inspector.

Untuk mengunduh hasil pemindaian CIS dari konsol

1. Masuk menggunakan kredensi Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan AWS Region dropdown untuk memilih AWS Region tempat Anda membuat konfigurasi pemindaian CIS Anda.
3. Dari panel navigasi, pilih Pemindaian sesuai permintaan, lalu pilih pemindaian CIS.
4. Pilih tab Hasil Pindai.
5. Di bawah kolom Dijadwalkan Berdasarkan, pilih ID jadwal pemindaian yang ingin Anda lihat. Atau pilih baris dengan ID jadwal pemindaian yang ingin Anda lihat, lalu pilih Lihat detail.
6. Pilih Unduh, lalu pilih PDF atau CSV. Jika akun Anda adalah akun administrator yang didelegasikan, Anda dapat memilih Pilih akun untuk mengunduh hasil untuk akun anggota tertentu.

Keamanan Kode Amazon Inspector

Amazon Inspector adalah layanan manajemen kerentanan yang secara otomatis menemukan beban kerja dan terus memindai mereka untuk kerentanan perangkat lunak dan paparan jaringan yang tidak diinginkan. Dengan Keamanan Kode, Amazon Inspector memindai kode sumber aplikasi pihak pertama, dependensi aplikasi pihak ketiga, dan Infrastruktur sebagai Kode untuk kerentanan. Anda dapat mengaktifkan Keamanan Kode di konsol Amazon Inspector atau dengan Amazon Inspector API. Setelah Anda mengaktifkan Keamanan Kode, Anda dapat membuat dan menerapkan konfigurasi pemindaian ke repositori kode Anda untuk menentukan seberapa sering dan kapan akan dipindai. Anda dapat melihat, mengedit, dan menghapus konfigurasi pemindaian Anda kapan saja. Untuk informasi tentang Wilayah AWS tempat Keamanan Kode tersedia, lihat Wilayah dan titik akhir. Untuk informasi tentang harga, lihat harga Amazon Inspector.

Prasyarat untuk Keamanan Kode

Sebelum Anda dapat mulai menggunakan Keamanan Kode, Anda harus mengaktifkan Keamanan Kode dan memutuskan cara mengenkripsi data Anda. Ini bisa berupa informasi seperti kredensi integrasi, kode, atau informasi lain yang terkait dengan integrasi, repositori kode, dan proyek Anda. Secara default, data Anda dienkripsi dengan kunci yang AWS dimiliki. Ini berarti kunci dibuat, dimiliki, dan dikelola oleh layanan. Jika Anda ingin memiliki dan mengelola kunci yang digunakan untuk mengenkripsi data Anda, Anda dapat membuat kunci KMS yang dikelola pelanggan.

Mengaktifkan Keamanan Kode

Anda mengaktifkan Keamanan Kode dengan cara yang sama seperti Anda mengaktifkan semua jenis pemindaian otomatis. Untuk informasi selengkapnya, lihat Mengaktifkan jenis pemindaian.

Membuat kunci yang dikelola pelanggan untuk mengakses AWS KMS

Secara default, data Anda dienkripsi dengan kunci yang AWS dimiliki. Ini berarti kunci dibuat, dimiliki, dan dikelola oleh layanan. Jika Anda ingin memiliki dan mengelola kunci yang digunakan untuk mengenkripsi data Anda, Anda dapat membuat kunci KMS yang dikelola pelanggan. Amazon Inspector tidak berinteraksi dengan data Anda. Amazon Inspector hanya menyerap metadata dari repositori di penyedia kode sumber Anda. Untuk informasi tentang cara membuat kunci KMS terkelola pelanggan, lihat Membuat kunci KMS di AWS Key Management Service Panduan Pengguna.

Contoh kebijakan

Saat Anda membuat kunci terkelola pelanggan, gunakan kebijakan contoh berikut.

> **Note**
>
> Izin FAS dalam kebijakan berikut khusus untuk Amazon Inspector, karena izin tersebut mengizinkan Amazon Inspector untuk hanya melakukan panggilan API tersebut.

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"
```

```
        }
      }
    }
  ]
}
```

Setelah Anda membuat kunci KMS Anda, Anda dapat menggunakan Amazon APIs Inspector berikut.

- UpdateEncryptionKey — Gunakan dengan `CODE_REPOSITORY` untuk `resourceType` dan `CODE` sebagai jenis pemindaian untuk mengonfigurasi penggunaan kunci KMS yang dikelola pelanggan Anda.
- GetEncryptionKey — Gunakan dengan `CODE_REPOSITORY` untuk `resourceType` dan `CODE` sebagai jenis pemindaian untuk mengonfigurasi pengambilan konfigurasi kunci KMS Anda.
- ResetEncryptionKey — Gunakan dengan `CODE_REPOSITORY` for `resourceType` dan `CODE` untuk mengatur ulang konfigurasi kunci KMS Anda dan untuk menggunakan kunci KMS yang AWS dimiliki.

Membuat integrasi antara Amazon Inspector repositori kode Anda

Bagian ini mencakup topik yang menjelaskan cara membuat integrasi antara Amazon Inspector dan repositori kode Anda. Saat Anda membuat integrasi, semua repositori kode terdaftar sebagai proyek di konsol Amazon Inspector di halaman Keamanan Kode. Topik lain di bagian ini menjelaskan cara mengakses integrasi dan proyek Anda.

Kode Keamanan hanya mengimpor hingga 100.000 proyek, dan hanya cabang default untuk setiap repositori yang dipantau. Sebuah proyek dapat dikaitkan dengan maksimal tiga konfigurasi pemindaian default.

Kode Keamanan hanya mendukung maksimal 100 integrasi per akun. Integrasi Kode Keamanan tidak memiliki konsep hubungan account/member akun administrator yang didelegasikan.

Untuk menghindari pembatasan, kami sarankan untuk tidak menggunakan host yang sama untuk integrasi lebih dari sekali.

Integrasi denganGitHub SaaS,GitHub Enterprise Cloud, dan GitHub Enterprise Server membutuhkan akses internet publik.

> **Important**
>
> Integrasi pihak ketiga mungkin dinonaktifkan sementara atau permanen tanpa pemberitahuan sebelumnya karena alasan apa pun, seperti untuk mengatasi masalah keamanan.

Membuat integrasi antara Amazon Inspector dan GitHub

Topik ini menjelaskan cara membuat integrasi antara Amazon Inspector dan. GitHub

> **Note**
>
> Jika ini adalah pertama kalinya Anda membuat integrasi, Anda diminta untuk membuat konfigurasi pemindaian default pada Langkah 2. Saat Anda membuat konfigurasi pemindaian, Anda memilih frekuensi pemindaian, analisis pemindaian, dan repositori yang akan dipindai. Membuat konfigurasi pemindaian default sama dengan membuat konfigurasi pemindaian umum. Namun, konfigurasi pemindaian default secara otomatis dikaitkan dengan proyek baru dan yang sudah ada yang diimpor ke Amazon Inspector. Jika Anda ingin membuat konfigurasi pemindaian default, pilih Lanjutkan dengan konfigurasi ini. Anda hanya dapat membuat konfigurasi pemindaian default sekali. Jika Anda membuat konfigurasi pemindaian default, Anda tidak akan diminta untuk membuat konfigurasi pemindaian default lagi. Anda hanya dapat membuat konfigurasi pemindaian default sekali per akun dan sekali per organisasi. Jika Anda tidak ingin mengonfigurasi konfigurasi pemindaian default, pilih Lewati konfigurasi. Namun, akan diminta untuk membuat konfigurasi pemindaian default saat berikutnya Anda membuat integrasi. Setelah Anda membuat konfigurasi pemindaian default atau melewatkan membuat konfigurasi pemindaian default, Anda diarahkan ke Langkah 3 dari alur kerja integrasi tempat Anda memasukkan detail integrasi.

Integrasi denganGitHub SaaS,GitHub Enterprise Cloud, dan GitHub Enterprise Server membutuhkan akses internet publik.

> **Note**
>
> Amazon Inspector hanya memindai dan memantau cabang default Anda. Jika Anda membuat cabang default baru, Amazon Inspector memindai dan memperbarui cabang default yang baru.

> **Important**
>
> Sebelum Anda selesai membuat integrasi, Anda diarahkan untuk mengotorisasi koneksi antara Amazon GitHub Inspector dan. Anda harus menyelesaikan langkah ini untuk menyelesaikan prosedur. Jika Anda menutup pop-up, Anda tidak akan dapat melanjutkan.

Untuk membuat integrasi antara Amazon Inspector dan GitHub

1. Masuk menggunakan kredensialmu. Buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. Dari panel navigasi, pilih Keamanan Kode. Pilih Connect to, dan pilihGitHub.
3. Di bawah Detail integrasi, masukkan nama integrasi Anda, dan pilih Connect to GitHub.
4. Pilih Otorisasi di pop-up untuk membuat koneksi antara Amazon Inspector dan. GitHub
5. Di spanduk sukses, pilih Buka halaman pembuatan GitHub koneksi.
6. Masukkan ID instalasi untuk GitHub aplikasi. Jika Anda menginstal GitHub aplikasi, Anda dapat menemukan ID instalasi GitHub dari halaman GitHub Aplikasi atau di akhir URL GitHub aplikasi. Jika Anda belum menginstal GitHub aplikasi, pilih Instal aplikasi baru. Ini mengarahkan Anda ke GitHub tempat Anda memilih GitHub organisasi dan menentukan ruang lingkup repositori.
7. Pilih Connect to GitHub.

Setelah Anda membuat integrasi, Anda dapat menemukan skenario di mana Amazon Inspector tidak dapat menyegarkan token akses. Hal ini dapat terjadi jika host integrasi tidak tersedia atau Amazon Inspector mengalami masalah komunikasi lainnya. Untuk mengatasi masalah ini, Anda dapat mengautentikasi ulang koneksi dari tab Integrasi pada halaman Keamanan Kode. Di bawah kolom Status, integrasi ditampilkan sebagai Tidak Aktif, dan Amazon Inspector menyediakan opsi untuk mengautentikasi ulang. Pilih Autentikasi Ulang. Anda diarahkan ke alur kerja integrasi tempat Anda dapat menyelesaikan pengaturan koneksi.

Jika Anda menghapus pengaturan sistem untuk integrasi Anda, Anda dapat kehilangan koneksi tanpa batas waktu. Jika ini terjadi, Anda harus menghapus integrasi dan membuat integrasi baru. Ketika Anda menghapus integrasi, Anda kehilangan semua proyek dan memindai konfigurasi yang terkait dengan integrasi.

Membuat integrasi antara Amazon Inspector dan GitLab Self Managed

Topik ini menjelaskan cara membuat integrasi antara Amazon Inspector dan repositori kode Anda di. GitLab Self Managed

Informasi yang diperlukan

Berikut ini diperlukan saat Anda membuat koneksi:

- Nama integrasi — Ini adalah nama yang ditambahkan ke badan integrasi Anda.
- URL Endpoint — Ini adalah URL yang digunakan untuk mengakses GitLab Self Managed instance Anda.
- Token akses pribadi — Token akses pribadi dibuat GitLab Self Managed dari akun administrator dan harus mencakup cakupan berikut:`api`,, `read_api``read_repository`, dan`write_repository`.

> **Note**
>
> Amazon Inspector hanya memindai dan memantau cabang default Anda. Jika Anda membuat cabang default baru, Amazon Inspector memindai dan memperbarui cabang default yang baru.

Membuat integrasi antara Amazon Inspector dan GitLab Self Managed

Prosedur berikut menjelaskan cara membuat koneksi antara Amazon Inspector dan repositori kode Anda di. GitLab Self Managed

> **Note**
>
> Jika ini adalah pertama kalinya Anda membuat integrasi, Anda diminta untuk membuat konfigurasi pemindaian default pada Langkah 2. Saat Anda membuat konfigurasi pemindaian, Anda memilih frekuensi pemindaian, analisis pemindaian, dan repositori yang akan dipindai. Membuat konfigurasi pemindaian default sama dengan membuat konfigurasi pemindaian umum. Namun, konfigurasi pemindaian default secara otomatis dikaitkan dengan proyek baru dan yang sudah ada yang diimpor ke Amazon Inspector. Jika Anda ingin membuat konfigurasi pemindaian default, pilih Lanjutkan dengan konfigurasi ini. Anda

hanya dapat membuat konfigurasi pemindaian default sekali. Jika Anda membuat konfigurasi pemindaian default, Anda tidak akan diminta untuk membuat konfigurasi pemindaian default lagi. Anda hanya dapat membuat konfigurasi pemindaian default sekali per akun dan sekali per organisasi. Jika Anda tidak ingin mengonfigurasi konfigurasi pemindaian default, pilih Lewati konfigurasi. Namun, Anda akan diminta untuk membuat konfigurasi pemindaian default saat berikutnya Anda membuat integrasi. Setelah Anda membuat konfigurasi pemindaian default atau melewatkan membuat konfigurasi pemindaian default, Anda diarahkan ke Langkah 3 dari alur kerja integrasi tempat Anda memasukkan detail integrasi.

Important

Sebelum Anda selesai membuat integrasi, Anda diminta untuk mengotorisasi koneksi antara Amazon Inspector GitLab dan Self Managed. Anda harus menyelesaikan langkah ini untuk menyelesaikan prosedur. Jika Anda menutup pop-up, Anda tidak akan dapat melanjutkan.

Untuk membuat koneksi dengan GitLab Self Managed

1. Masuk menggunakan kredensialmu. Buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. Dari panel navigasi, pilih Keamanan Kode. Pilih Connect to dan pilih GitLab Self Managed.
3. Di bawah Detail Integrasi, masukkan yang berikut ini:
   a. Untuk nama Integrasi, masukkan nama yang ditambahkan ke badan integrasi Anda.
   b. Untuk URL Endpoint, masukkan URL yang digunakan untuk mengakses instans yang GitLab dikelola sendiri.
   c. Untuk token akses pribadi, masukkan token akses pribadi Anda dengan cakupan yang diperlukan.
4. Pilih sambungkan keGitLab.
5. Pilih Otorisasi di jendela pop-up untuk menyelesaikan pembuatan koneksi antara Amazon Inspector dan. GitLab

Setelah Anda membuat integrasi, Anda dapat menemukan skenario di mana Amazon Inspector tidak dapat menyegarkan token akses. Hal ini dapat terjadi jika host integrasi tidak tersedia atau

Amazon Inspector mengalami masalah komunikasi lainnya. Untuk mengatasi masalah ini, Anda dapat mengautentikasi ulang koneksi dari tab Integrasi pada halaman Keamanan Kode. Di bawah kolom Status, integrasi ditampilkan sebagai Tidak Aktif, dan Amazon Inspector menyediakan opsi untuk mengautentikasi ulang. Pilih Autentikasi Ulang. Anda diarahkan ke alur kerja integrasi tempat Anda dapat menyelesaikan pengaturan koneksi.

Jika Anda menghapus pengaturan sistem untuk integrasi Anda, Anda dapat kehilangan koneksi tanpa batas waktu. Jika ini terjadi, Anda harus menghapus integrasi dan membuat integrasi baru. Ketika Anda menghapus integrasi, Anda kehilangan semua proyek dan memindai konfigurasi yang terkait dengan integrasi.

Melihat integrasi dengan repositori kode

Topik ini menjelaskan cara melihat integrasi di konsol Amazon Inspector.

Untuk melihat integrasi di konsol Amazon Inspector

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Integrasi. Dari tab ini, Anda dapat meninjau semua integrasi yang dikonfigurasi dan meninjau informasi dasar tentang semua integrasi Anda. Informasi ini mencakup nama integrasi, status integrasi, dan nama penyedia kode sumber.

Autentikasi ulang ke penyedia

Setelah Anda membuat integrasi, Anda dapat menemukan skenario di mana Amazon Inspector tidak dapat menyegarkan token akses. Hal ini dapat terjadi jika host integrasi tidak tersedia atau Amazon Inspector mengalami masalah komunikasi lainnya. Untuk mengatasi masalah ini, Anda dapat mengautentikasi ulang koneksi dari tab Integrasi pada halaman Keamanan Kode. Di bawah kolom Status, integrasi ditampilkan sebagai Tidak Aktif, dan Amazon Inspector menyediakan opsi untuk mengautentikasi ulang. Pilih Autentikasi Ulang. Anda diarahkan ke alur kerja integrasi di mana Anda dapat menyelesaikan pengaturan koneksi.

Jika Anda menghapus pengaturan sistem untuk integrasi Anda, Anda dapat kehilangan koneksi tanpa batas waktu. Jika ini terjadi, Anda harus menghapus integrasi dan membuat integrasi baru. Ketika Anda menghapus integrasi, Anda kehilangan semua proyek dan memindai konfigurasi yang terkait dengan integrasi.

Melihat repositori kode

Topik ini menjelaskan cara melihat repositori kode di konsol Amazon Inspector.

Untuk melihat repositori kode di konsol Amazon Inspector

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Repositori kode. Dari tab ini, Anda dapat meninjau semua repositori kode Anda, yang terdaftar sebagai proyek, dan meninjau informasi dasar tentang mereka. Informasi ini mencakup nama dan status pemindaian untuk setiap proyek. Anda juga dapat meninjau konfigurasi yang terkait dengan proyek Anda dan kapan proyek Anda terakhir dipindai. Anda bahkan dapat memfilter proyek Anda di bilah pencarian.

Melihat detail untuk sebuah proyek

Topik ini menjelaskan cara melihat detail proyek di konsol Amazon Inspector. Jika akun Anda adalah administrator yang didelegasikan untuk organisasi, Anda dapat melihat detail untuk proyek milik akun anggota.

Untuk melihat proyek kode di konsol Amazon Inspector

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Repositori kode. Dari tab ini, Anda dapat meninjau semua repositori kode Anda, yang terdaftar sebagai proyek, dan meninjau informasi dasar tentang mereka. Informasi ini mencakup nama dan status pemindaian untuk setiap proyek. Anda juga dapat meninjau konfigurasi yang terkait dengan proyek Anda dan kapan proyek Anda terakhir dipindai. Anda bahkan dapat memfilter proyek Anda di bilah pencarian.
4. Pilih proyek. Atau pilih proyek, dan pilih Lihat detail. Dari layar Detail proyek, Anda dapat meninjau informasi dasar tentang proyek. Informasi ini mencakup nama dan ID untuk proyek, serta integrasi ARN. Ini mencakup informasi tentang kapan proyek dipindai dan jenis penyediaan. Anda bahkan dapat meninjau temuan yang terkait dengan proyek, serta mengekspor temuan dan membuat aturan penekanan untuk temuan.

Menghapus integrasi

Prosedur berikut menjelaskan cara menghapus integrasi di konsol Amazon Inspector. Ketika Anda menghapus integrasi, Anda kehilangan semua proyek dan memindai konfigurasi yang terkait dengan integrasi.

Untuk menghapus integrasi di konsol Amazon Inspector.

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Integrasi. Dari tab ini, Anda dapat meninjau semua integrasi yang dikonfigurasi dan meninjau informasi dasar tentang semua integrasi Anda. Informasi ini mencakup nama integrasi, status integrasi, dan jenis penyedia integrasi.
4. Pilih integrasi, dan pilih Hapus.

Membuat konfigurasi pemindaian

Sebelum membuat konfigurasi pemindaian, Anda harus membuat integrasi dengan Amazon Inspector. Pertama kali Anda membuat integrasi, Anda diminta untuk membuat konfigurasi pemindaian default. Topik ini menjelaskan cara membuat konfigurasi pemindaian umum. Perbedaan antara konfigurasi pemindaian default dan konfigurasi pemindaian umum adalah bahwa konfigurasi pemindaian default secara otomatis dilampirkan ke proyek baru. Anda dapat melewatkan membuat konfigurasi pemindaian default.

Kode Keamanan hanya mendukung maksimum 500 konfigurasi pemindaian umum. Keamanan kode hanya mendukung 1 konfigurasi pemindaian default per akun dan per organisasi. Konfigurasi pemindaian hanya dapat dikaitkan dengan maksimum 100.000 proyek.

Sebuah proyek dapat dikaitkan dengan maksimal 4 konfigurasi pemindaian total. Ini termasuk konfigurasi pemindaian default jika konfigurasi pemindaian default dibuat. Konfigurasi pemindaian untuk organisasi tidak dapat diberi tag.

Jika administrator yang didelegasikan untuk organisasi membuat konfigurasi pemindaian, konfigurasi pemindaian dibuat di tingkat organisasi dan diterapkan ke semua akun anggota dalam organisasi. Hal yang sama terjadi jika administrator yang didelegasikan membuat konfigurasi pemindaian default.

Saat Anda membuat konfigurasi pemindaian, Anda memilih frekuensi pemindaian, analisis pemindaian, dan repositori yang akan dipindai. Frekuensi pemindaian dapat didasarkan perubahan dan periodik atau disesuaikan. Pemindaian berbasis perubahan dan periodik memberi Anda opsi untuk mengaktifkan pemindaian berkala. Jika Anda mengaktifkan pemindaian berkala, Anda mengatur frekuensi pemindaian ke hari dalam seminggu atau bulan saat pemindaian terjadi. Pemindaian khusus memberi Anda opsi untuk mengaktifkan pemindaian saat kode diubah dan pemindaian berkala. Jika Anda mengaktifkan pemindaian saat kode diubah, Anda menentukan pemicu pemindaian untuk disertakan dalam permintaan gabungan dan tarik.

Pemindaian dapat dilewati jika ID komit tidak berubah dalam jangka waktu tertentu. Untuk pemindaian berkala, pemindaian dilewati jika ID komit tidak berubah antara pemindaian dalam 1 minggu. Untuk pemindaian sesuai permintaan, pemindaian dilewati jika ID komit tidak berubah di antara pemindaian dalam 24 jam.

> **Note**
>
> Jika konfigurasi pemindaian hanya memiliki pemicu untuk permintaan gabungan dan permintaan tarik, hanya 25 temuan kritis atau tinggi teratas yang disajikan dan hanya di platform manajemen kode sumber. Tidak ada yang akan terlihat di Amazon Inspector.

Untuk membuat konfigurasi pemindaian umum

1. Masuk menggunakan kredensialmu. Buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Konfigurasi, lalu pilih Buat konfigurasi pemindaian.
4. Di bawah detail Pindai, lakukan hal berikut:
   - Untuk nama Konfigurasi, masukkan nama untuk konfigurasi pemindaian.
5. Di bawah Frekuensi pemindaian, tentukan seberapa sering kode dipindai dengan memilih Pemindaian berbasis perubahan dan periodik atau Jenis dan pemicu pemindaian yang disesuaikan.
   a. (Opsi 1) Jika Anda memilih Ubah pemindaian berbasis dan berkala, pilih Aktifkan pemindaian berkala atau Nonaktifkan pemindaian berkala.

- . Jika Anda memilih Aktifkan pemindaian berkala, atur frekuensi pemindaian dengan memilih minggu dan hari Anda ingin kode dipindai.

b. (Opsi 2) Jika Anda memilih Pemindaian khusus, putuskan apakah akan mengaktifkan pemindaian saat kode diubah dan pemindaian berkala.

   i. Pilih Aktifkan pemindaian saat kode diubah atau Nonaktifkan pemindaian saat kode diubah. Jika Anda memilih Aktifkan pemindaian saat kode diubah, tentukan kapan pemindaian dipicu dari dropdown.

   ii. Pilih Aktifkan pemindaian berkala atau Nonaktifkan pemindaian berkala. Jika Anda memilih Aktifkan pemindaian berkala, atur frekuensi pemindaian dengan memilih minggu dan hari Anda ingin kode dipindai. Anda juga dapat memindai pemicu berbasis peristiwa. Peristiwa ini termasuk ketika permintaan tarik baru awalnya dibuka terhadap cabang default dan ketika komit digabungkan atau didorong ke cabang default. Pemindaian tidak dipicu pada pembaruan atau revisi berikutnya untuk permintaan tarik yang ada. Untuk memicu pemindaian baru, tutup dan buka kembali permintaan tarik.

6. Di bawah Analisis pemindaian, putuskan apakah akan mengonfigurasi analisis pemindaian lengkap atau analisis pemindaian yang disesuaikan:

   a. (Opsi 1) Jika Anda memilih Analisis pemindaian lengkap, Anda menerapkan semua analisis pemindaian berikut:

      - Pengujian Keamanan Aplikasi Statis - Menganalisis kode sumber untuk kerentanan.
      - Pemindaian IAC — Menganalisis skrip dan kode yang mengkonfigurasi dan menyediakan infrastruktur.
      - Analisis komposisi perangkat lunak statis — Memeriksa paket open source dalam aplikasi.

   b. (Opsi 2) Jika Anda memilih Analisis pemindaian yang disesuaikan, Anda harus memilih setidaknya satu jenis jenis analisis pemindaian yang disebutkan sebelumnya dari menu tarik-turun:

7. (Opsional) Untuk Tag, buat pasangan kunci-nilai untuk diterapkan ke proyek Anda. Anda dapat membuat hingga 50 tag.

8. Pilih Berikutnya.

9. Di bawah Pemilihan repositori, pilih Semua repositori atau Repositori khusus.

   a. (Opsi 1) Jika Anda memilih Semua repositori, pemindaian diaktifkan untuk repositori yang ada.

b. (Opsi 2) Jika Anda memilih Repositori khusus, pemindaian diaktifkan hanya untuk repositori yang Anda tentukan.

10. Pilih Berikutnya.
11. Tinjau pilihan Anda, lalu pilih Buat konfigurasi pemindaian.

> **Note**
>
> Konfigurasi pemindaian umum diterapkan ke semua repositori kode yang ada saja. Mereka tidak akan diterapkan ke repositori kode baru.

Melihat konfigurasi pemindaian

Prosedur berikut menjelaskan cara melihat konfigurasi pemindaian di konsol Amazon Inspector.

> **Note**
>
> Saat Anda melihat konfigurasi pemindaian di tingkat organisasi, beberapa detail di layar Keamanan Kode akan berbeda untuk mencerminkan konfigurasi Anda Akun AWS.

Untuk melihat detail untuk konfigurasi pemindaian

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Konfigurasi untuk melihat daftar konfigurasi pemindaian Anda. Jika Anda administrator yang didelegasikan, daftar tersebut menyertakan konfigurasi pemindaian organisasi Anda. Anda dapat melihat nama setiap konfigurasi pemindaian dan siapa yang membuat setiap konfigurasi pemindaian (Akun AWS ID atau ID organisasi). Anda juga dapat melihat jenis pemindaian dan jenis analisis pemindaian yang diterapkan pada konfigurasi. Anda bahkan dapat memfilter konfigurasi pemindaian Anda dengan bidang yang berbeda di bilah pencarian.

Melihat detail untuk konfigurasi pemindaian

Prosedur berikut menjelaskan cara melihat detail untuk konfigurasi pemindaian di konsol Amazon Inspector.

Untuk melihat detail untuk konfigurasi pemindaian

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Konfigurasi.
4. Pilih konfigurasi yang ingin Anda lihat detailnya. Layar detail konfigurasi pemindaian memberikan gambaran umum tentang konfigurasi pemindaian. Dari layar ini, Anda dapat melihat konfigurasi pemindaian ARN, jenis frekuensi pemindaian mana yang diaktifkan, dan jenis analisis pemindaian mana yang diaktifkan. Anda juga dapat menghapus konfigurasi pemindaian dari layar ini. Jika Anda melihat konfigurasi pemindaian milik organisasi Anda, Anda juga dapat mengedit dari layar ini.

Mengedit konfigurasi pemindaian

Anda dapat mengedit konfigurasi pemindaian kapan saja. Saat mengedit konfigurasi pemindaian, Anda dapat mengubah frekuensi pemindaian, analisis pemindaian, tag, dan repositori yang akan dipindai. Misalnya, Anda mengedit konfigurasi pemindaian untuk menjeda pemindaian untuk repositori tertentu. Prosedur berikut menjelaskan cara mengedit konfigurasi pemindaian.

Untuk mengedit konfigurasi pemindaian

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan Kode.
3. Pilih Konfigurasi.
4. Pilih konfigurasi yang ingin Anda edit, lalu pilih Edit. Anda juga dapat memilih konfigurasi yang ingin Anda edit, lalu pilih Edit.

Menghapus konfigurasi pemindaian

Anda dapat menghapus konfigurasi pemindaian kapan saja. Topik ini menjelaskan cara menghapus konfigurasi pemindaian.

Untuk menghapus konfigurasi pemindaian

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan kode.
3. Pilih Konfigurasi.
4. Pilih konfigurasi yang ingin Anda hapus, lalu pilih Hapus. Atau pilih konfigurasi yang ingin Anda hapus, lalu pilih Hapus.

Melakukan pemindaian sesuai permintaan

Anda dapat melakukan on-demand untuk proyek Anda. Saat Anda melakukan pemindaian sesuai permintaan, gabungan semua konfigurasi pemindaian yang dikonfigurasi diterapkan ke proyek yang Anda pilih. Jika akun Anda adalah akun administrator yang didelegasikan untuk organisasi, Anda dapat melakukan pemindaian sesuai permintaan untuk proyek milik akun anggota. Prosedur berikut menjelaskan cara melakukan pemindaian sesuai permintaan di konsol Amazon Inspector.

Untuk melakukan pemindaian sesuai permintaan

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Keamanan kode.
3. Pilih Repositori kode.
4. Pilih proyek yang ingin Anda pindai, lalu pilih Pemindaian sesuai permintaan.

Bahasa yang didukung untuk keamanan kode Amazon Inspector

Topik ini mencakup bahasa yang didukung untuk Amazon Inspector Code Security.

Bahasa yang didukung untuk SAST

- C#(semua versi tetapi .Net 6.0 dan yang lebih baru disarankan)

- C(C11 atau sebelumnya)
- C++(C++17 atau lebih awal)
- Go(hanya Go 1,18)
- Java(Java17 atau lebih awal)
- JavaScript(EMCMAScript 2021 atau sebelumnya)
- JSX(React 17 atau sebelumnya)
- Kotlin(Kotlin2.0 atau sebelumnya)
- PHP(PHP8.2 atau sebelumnya)
- Python(Python3.11 atau lebih awal dalam seri Python 3)
- Ruby(Hanya Ruby 2,7 dan 3,2)
- Rust
- Scala(Scala3.2.2 atau sebelumnya)
- Shell
- TSX
- TypeScript (semua versi)

Bahasa yang didukung untuk analisis komposisi perangkat lunak

- Go(hanya Go 1,18)
- Java(Java17 atau lebih awal)
- JavaScript(EMCMAScript 2021 atau sebelumnya)
- PHP(PHP8.2 atau sebelumnya)
- Python(Python3.11 atau lebih awal dalam seri Python 3)
- .Net
- Ruby(Hanya Ruby 2,7 dan 3,2)
- Rust

Bahasa untuk Infrastruktur sebagai Kode

- AWS CDK (PythondanTypeScript)
- CloudFormation (2010—09—09)
- Terraform(1.6.2 atau sebelumnya)

Menonaktifkan Keamanan Kode

Untuk informasi selengkapnya tentang menonaktifkan Keamanan Kode, lihat Menonaktifkan jenis pemindaian.

Memahami temuan Amazon Inspector

Amazon Inspector menghasilkan temuan saat mendeteksi kerentanan dengan perbaikan atau perbaikan tertunda di instans Amazon EC2, image container Amazon ECR, dan fungsi Lambda. Ini juga menghasilkan temuan untuk kerentanan kode yang terdeteksi dalam kode sumber aplikasi pihak pertama, dependensi aplikasi pihak ketiga, dan Infrastruktur sebagai Kode. Temuan adalah laporan terperinci tentang kerentanan yang memengaruhi salah satu sumber daya Anda AWS .

Temuan dinamai berdasarkan kerentanan dan memberikan peringkat keparahan, informasi tentang sumber daya yang terkena dampak dan non AWS sumber AWS daya, dan detail yang menjelaskan cara memulihkan kerentanan yang terdeteksi. Amazon Inspector menyimpan semua temuan aktif Anda sampai Anda memperbaikinya.

Ketika sumber daya dihapus, dihentikan, atau tidak lagi memenuhi syarat untuk pemindaian, Amazon Inspector secara otomatis menutup temuan yang terkait dengan sumber daya dan kemudian menghapus temuan setelah 3 hari. Jika temuan ditutup karena alasan lain, mereka dihapus setelah 30 hari.

> **Note**
>
> Amazon Inspector akan membuka kembali temuan yang diperbaiki dalam waktu tujuh hari setelah menutup temuan jika masalah yang menyebabkan kerentanan terjadi kembali.

Jika Anda menonaktifkan Amazon Inspector, temuan akan dihapus setelah 24 jam. Jika sumber daya dihentikan, temuan apa pun yang terkait dengan sumber daya akan dihapus setelah 3 hari. Hal yang sama terjadi untuk setiap temuan yang melekat pada sumber daya di mana pemindaian tidak lagi memenuhi syarat. Jika AWS menangguhkan akun Anda, temuan akan dihapus setelah 90 hari. Temuan untuk contoh yang dihentikan tetap aktif.

Temuan menyatakan

Amazon Inspector mengkategorikan temuan di negara bagian berikut.

Aktif

Amazon Inspector mengkategorikan temuan yang belum diperbaiki sebagai Aktif.

Ditekan

Amazon Inspector mengkategorikan temuan yang tunduk pada satu atau lebih aturan penindasan sebagai Ditekan.

Ditutup

Ketika sebuah temuan telah diperbaiki, Amazon Inspector mengkategorikan temuan tersebut sebagai Closed.

Topik

Jenis pencarian Amazon Inspector

Bagian ini menjelaskan berbagai jenis temuan di Amazon Inspector.

Topik

Kerentanan Package

Temuan kerentanan Package mengidentifikasi paket perangkat lunak di AWS lingkungan Anda yang terkena Common Vulnerabilities and Exposures (). CVEs Penyerang dapat mengeksploitasi kerentanan yang belum ditambal ini untuk membahayakan kerahasiaan, integritas, atau ketersediaan data, atau untuk mengakses sistem lain. Sistem CVE adalah metode referensi untuk kerentanan dan eksposur keamanan informasi yang diketahui publik. Untuk informasi lebih lanjut, lihat https://www.cve.org/.

Amazon Inspector dapat menghasilkan temuan kerentanan paket untuk EC2 instance, image container ECR, dan fungsi Lambda. Temuan kerentanan paket memiliki detail tambahan yang unik untuk jenis temuan ini, ini adalah skor Inspector dan intelijen kerentanan.

Kerentanan kode

Temuan kerentanan kode membantu mengidentifikasi baris kode yang dapat dieksploitasi. Kerentanan kode termasuk enkripsi yang hilang, kebocoran data, kekurangan injeksi, dan kriptografi yang lemah. Amazon Inspector menghasilkan temuan kerentanan kode melalui pemindaian fungsi Lambda dan fitur Keamanan Kode-nya.

Amazon Inspector mengevaluasi kode aplikasi fungsi Lambda menggunakan penalaran otomatis dan pembelajaran mesin untuk menganalisis kode aplikasi untuk kepatuhan keamanan secara keseluruhan. Ini mengidentifikasi pelanggaran kebijakan dan kerentanan berdasarkan detektor internal yang dikembangkan bekerja sama dengan Amazon Q. Untuk daftar kemungkinan deteksi, lihat Amazon Q Detector Library.

Pemindaian kode menangkap cuplikan kode untuk menyoroti kerentanan yang terdeteksi. Misalnya, cuplikan kode mungkin menampilkan kredensi hardcode atau materi sensitif lainnya dalam teks biasa. Amazon Q menyimpan cuplikan kode yang terkait dengan kerentanan kode. Secara default, kode Anda dienkripsi dengan kunci yang AWS dimiliki. Namun, Anda dapat membuat kunci yang dikelola pelanggan untuk mengenkripsi kode Anda jika Anda ingin lebih banyak kontrol atas informasi ini. Untuk informasi selengkapnya, lihat Enkripsi saat istirahat untuk kode dalam temuan Anda.

> **Note**
>
> Administrator yang didelegasikan untuk organisasi tidak dapat melihat cuplikan kode milik akun anggota.

Jangkauan jaringan

Temuan jangkauan jaringan menunjukkan bahwa ada jalur jaringan terbuka ke EC2 instans Amazon di lingkungan Anda. Temuan ini muncul ketika port TCP dan UDP Anda dapat dijangkau dari tepi VPC, seperti gateway internet (termasuk contoh di belakang Application Load Balancers atau Classic Load Balancers), koneksi peering VPC, atau VPN melalui gateway virtual. Temuan ini menyoroti konfigurasi jaringan yang mungkin terlalu permisif, seperti grup keamanan yang salah kelola, Daftar Kontrol Akses, atau gateway internet, atau yang memungkinkan akses yang berpotensi berbahaya.

Amazon Inspector hanya menghasilkan temuan jangkauan jaringan untuk instans Amazon. EC2 Amazon Inspector melakukan pemindaian untuk temuan jangkauan jaringan setiap 12 jam setelah Amazon Inspector diaktifkan.

Amazon Inspector mengevaluasi konfigurasi berikut saat memindai jalur jaringan:

- EC2 Contoh Amazon
- Penyeimbang Beban Aplikasi
- Connect Langsung
- Penyeimbang Beban Elastis
- Antarmuka Jaringan Elastis
- Gerbang Internet
- Daftar Kontrol Akses Jaringan
- Tabel Rute
- Grup Keamanan
- Subnet
- Awan Pribadi Virtual
- Gateway Pribadi Virtual
- Titik akhir VPC
- Titik akhir gerbang VPC
- Koneksi peering VPC
- Koneksi VPN

Melihat temuan Amazon Inspector

Anda dapat melihat temuan di konsol Amazon Inspector dan dengan Amazon `ListFindings`Inspector API. Di konsol Amazon Inspector, Anda dapat melihat semua temuan Anda di layar Dasbor dan Temuan. Secara default, layar ini hanya menampilkan temuan aktif dan kritis Anda. Namun, Anda dapat memfilter temuan atau memilih untuk melihat temuan berdasarkan kategori. Anda juga dapat melihat beberapa temuan di Security Hub CSPM dan Amazon ECR jika Anda mengaktifkan integrasi ini. Prosedur di bagian ini menjelaskan cara melihat temuan di konsol Amazon Inspector dan dengan Amazon `ListFindings` Inspector API.

Console

Untuk melihat temuan Amazon Inspector

1. Masuk menggunakan kredensil Anda. Buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. (Opsional) Dari panel navigasi, pilih Dasbor. Dasbor menunjukkan ikhtisar cakupan untuk lingkungan Anda dan hanya temuan aktif dan kritis Anda.
3. (Opsional) Dari panel navigasi, pilih Temuan. Layar ini mencantumkan semua temuan aktif Anda. Anda dapat menggunakan kriteria filter untuk melihat temuan tertentu. Untuk mengecualikan temuan dari daftar, buat aturan penindasan. Untuk melihat detail temuan, pilih nama temuan.
4. (Opsional) Dari panel navigasi, pilih salah satu opsi berikut untuk melihat temuan Anda berdasarkan kategori:
   - Berdasarkan kerentanan - Menunjukkan kerentanan dengan temuan paling kritis.
   - Berdasarkan akun - Menampilkan akun dengan temuan paling penting. Kategori ini hanya tersedia untuk administrator yang didelegasikan.
   - Sebagai contoh - Menampilkan instans Amazon EC2 dengan temuan paling penting. Kategori ini tidak termasuk informasi tentang ketersediaan jaringan.
   - Berdasarkan gambar kontainer - Menampilkan gambar kontainer Amazon ECR dengan temuan paling penting. Kategori ini juga memberikan informasi dasar tentang gambar kontainer Anda. Bahkan mencakup detail, seperti berapa banyak tugas Amazon ECS dan pod Amazon EKS yang digunakan. Dari layar ini, Anda dapat mengetahui berapa banyak tasks/pods yang berjalan dalam 24 jam terakhir dan berhenti.
   - Dengan repositori kontainer - Menampilkan repositori kontainer dengan temuan paling penting.
   - Dengan fungsi Lambda — Menunjukkan fungsi Lambda dengan temuan paling kritis.

API

Untuk melihat temuan Amazon Inspector

- Jalankan operasi ListFindingsAPI. Dalam permintaan, tentukan FilterCriteria untuk mengembalikan temuan tertentu.

Melihat detail untuk temuan Amazon Inspector Anda

Prosedur di bagian ini menjelaskan cara melihat detail untuk temuan Amazon Inspector.

Untuk melihat detail untuk temuan

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home https://console.aws.amazon.com/inspector/
2. Pilih Wilayah untuk melihat temuan di.
3. Di panel navigasi, pilih Temuan untuk menampilkan daftar temuan
4. (Opsional) Gunakan bilah filter untuk memilih temuan tertentu. Untuk informasi selengkapnya, lihat Memfilter temuan Amazon Inspector Anda.
5. Pilih temuan untuk melihat panel detailnya.

Panel Finding details berisi fitur identifikasi dasar dari temuan tersebut. Ini termasuk judul temuan serta deskripsi dasar tentang kerentanan yang diidentifikasi, saran remediasi, dan skor keparahan. Untuk informasi tentang penilaian, lihatMemahami tingkat keparahan untuk temuan Amazon Inspector Anda.

Detail yang tersedia untuk temuan bervariasi tergantung pada jenis temuan dan Sumber Daya yang terpengaruh.

Semua temuan berisi nomor Akun AWS ID tempat temuan diidentifikasi, tingkat keparahan, Jenis temuan, tanggal penemuan dibuat, dan bagian yang terpengaruh Sumber Daya dengan detail tentang sumber daya itu.

Jenis temuan menentukan informasi intelijen remediasi dan kerentanan yang tersedia untuk temuan tersebut. Tergantung pada jenis temuan, detail temuan yang berbeda tersedia.

Package Vulnerability

Temuan kerentanan paket tersedia untuk instans EC2, gambar kontainer ECR, dan fungsi Lambda. Lihat Kerentanan Package untuk info lebih lanjut.

Temuan kerentanan Package juga termasukMelihat skor Amazon Inspector dan memahami detail intelijen kerentanan.

Jenis temuan ini memiliki detail sebagai berikut:

- Perbaiki tersedia - Menunjukkan jika kerentanan diperbaiki dalam versi yang lebih baru dari paket yang terpengaruh. Memiliki salah satu nilai berikut:
  - `YES`, yang berarti semua paket yang terpengaruh memiliki versi tetap.
  - `NO`, yang berarti tidak ada paket yang terpengaruh memiliki versi tetap.
  - `PARTIAL`, yang berarti satu atau lebih (tetapi tidak semua) paket yang terpengaruh memiliki versi tetap.
- Eksploitasi tersedia - Menunjukkan kerentanan memiliki eksploitasi yang diketahui.
  - `YES`, yang berarti kerentanan yang ditemukan di lingkungan Anda memiliki eksploitasi yang diketahui. Amazon Inspector tidak memiliki visibilitas ke dalam penggunaan eksploitasi di lingkungan.
  - `NO`, yang berarti kerentanan ini tidak memiliki eksploitasi yang diketahui.
- Paket yang terpengaruh - Daftar setiap paket yang diidentifikasi rentan dalam temuan, dan detail setiap paket:
- Filepath — ID volume EBS dan nomor partisi yang terkait dengan temuan. Bidang ini hadir dalam temuan untuk instans EC2 yang dipindai menggunakan. Pemindaian tanpa agen
- Versi terinstal/ Versi tetap - Nomor versi paket yang saat ini diinstal yang kerentanan terdeteksi. Bandingkan nomor versi yang diinstal dengan nilai setelah garis miring (/). Nilai kedua adalah nomor versi paket yang memperbaiki kerentanan yang terdeteksi seperti yang disediakan oleh Common Vulnerabilities and Exposures (CVEs) atau advisory yang terkait dengan temuan. Jika kerentanan telah diperbaiki dalam beberapa versi, bidang ini mencantumkan versi terbaru yang menyertakan perbaikan. Jika perbaikan tidak tersedia, nilai ini adalah`None available`.

  > **Note**
  >
  > Jika temuan terdeteksi sebelum Amazon Inspector mulai memasukkan bidang ini dalam temuan, nilai untuk bidang ini kosong. Namun, perbaikan mungkin tersedia.

- Package manager — Manajer paket yang digunakan untuk mengkonfigurasi paket ini.
- Remediasi — Jika perbaikan tersedia melalui paket atau pustaka pemrograman yang diperbarui, bagian ini menyertakan perintah yang dapat Anda jalankan untuk melakukan pembaruan. Anda dapat menyalin perintah yang disediakan dan menjalankannya di lingkungan Anda.

> **Note**
>
> Perintah remediasi disediakan dari umpan data vendor dan dapat bervariasi tergantung pada konfigurasi sistem Anda. Tinjau referensi penemuan atau dokumentasi sistem operasi untuk panduan yang lebih spesifik.

- Detail kerentanan - menyediakan tautan ke sumber pilihan Amazon Inspector untuk CVE yang diidentifikasi dalam temuan, seperti National Vulnerability Database (NVD), REDHAT, atau vendor OS lainnya. Selain itu, Anda akan menemukan skor keparahan untuk temuan tersebut. Untuk informasi lebih lanjut tentang penilaian tingkat keparahan seperti, lihatMemahami tingkat keparahan untuk temuan Amazon Inspector Anda. Skor berikut disertakan, termasuk vektor penilaian untuk masing-masing:
  - Skor Exploit Prediction Scoring System (EPSS)
  - Skor Inspector
  - CVSS 3.1 dari Amazon CVE
  - CVSS 3.1 dari NVD
  - CVSS 2.0 dari NVD (jika berlaku, untuk yang lebih tua) CVEs
- Kerentanan terkait - Menentukan kerentanan lain yang terkait dengan temuan. Biasanya ini adalah CVEs hal lain yang berdampak pada versi paket yang sama, atau lainnya CVEs dalam grup yang sama dengan CVE temuan, sebagaimana ditentukan oleh vendor.
- Sumber daya yang terpengaruh - Termasuk informasi tentang registri, repositori, jenis sumber daya, ID gambar, dan sistem operasi gambar. Ini juga mencakup informasi, seperti kapan gambar terakhir kali didorong, berapa banyak tugas Amazon ECS dan pod Amazon EKS yang digunakan, dan kapan gambar terakhir digunakan dalam 24 jam terakhir. Jika Anda memiliki tugas Amazon ECS dan pod Amazon EKS yang diterapkan, Anda dapat melihat detail dengan memilih nilai untuk bidang tersebut. Ini mengarahkan Anda ke layar tempat Anda dapat melihat informasi, seperti ARN cluster, saat sumber daya terakhir digunakan dalam 24 jam terakhir, hitungan sumber daya yang berjalan dan berhenti, serta nama dan jenis beban kerja.

Kerentanan kode

Temuan kerentanan kode hanya tersedia untuk fungsi Lambda. Lihat Kerentanan kode untuk info lebih lanjut. Jenis temuan ini memiliki detail sebagai berikut:

- Perbaiki yang tersedia - Untuk kerentanan kode, nilai ini selaluYES.

- Nama detektor — Nama detektor Amazon Q yang digunakan untuk mendeteksi kerentanan kode. Untuk daftar kemungkinan deteksi, lihat Perpustakaan Detektor Q.
- Tag detektor — Tag Amazon Q yang terkait dengan detektor, Amazon Q menggunakan tag untuk mengkategorikan deteksi.
- CWE yang relevan — IDs dari Common Weakness Enumeration (CWE) yang terkait dengan kerentanan kode.
- Jalur file — Lokasi file kerentanan kode.
- Lokasi kerentanan — Untuk kerentanan kode pemindaian kode Lambda, bidang ini menunjukkan baris kode yang tepat di mana Amazon Inspector menemukan kerentanan.
- Remediasi yang disarankan — Ini menunjukkan bagaimana kode dapat diedit untuk memulihkan temuan.

Jangkauan jaringan

Temuan jangkauan jaringan hanya tersedia untuk instans EC2. Lihat Jangkauan jaringan untuk info lebih lanjut. Jenis temuan ini memiliki detail sebagai berikut:

- Rentang port terbuka — Rentang port yang melaluinya instans EC2 dapat diakses.
- Jalur jaringan terbuka - Menunjukkan jalur akses terbuka ke instans EC2. Pilih item di jalur untuk informasi lebih lanjut.
- Remediasi — Merekomendasikan metode untuk menutup jalur jaringan terbuka.

Melihat skor Amazon Inspector dan memahami detail intelijen kerentanan

Amazon Inspector membuat skor untuk temuan instans Amazon Elastic Compute Cloud (Amazon EC2). Anda dapat melihat skor Amazon Inspector dan detail intelijen kerentanan di konsol Amazon Inspector. Skor Amazon Inspector memberi Anda detail yang dapat Anda bandingkan dengan metrik di Sistem Penilaian Kerentanan Umum. Detail ini hanya tersedia untuk temuan kerentanan paket. Bagian ini menjelaskan cara menafsirkan skor Amazon Inspector dan memahami detail intelijen kerentanan.

Skor Amazon Inspector

Amazon Inspector membuat skor untuk setiap temuan Amazon EC2. Amazon Inspector menentukan skor dengan mengkorelasikan informasi skor dasar CVSS dengan informasi dari lingkungan

komputasi Anda, seperti data jangkauan jaringan dan data eksploitabilitas. Amazon Inspector mendukung vendor Amazon, Debian, dan RHEL. Setiap vendor memberikan skor dasar CVSS v3.1. Untuk vendor lain, Amazon Inspector menggunakan skor dasar CVSS yang disediakan oleh National Vulnerability Database (NVD).

Karena persyaratan FedRAMP, Amazon Inspector menggunakan skor dasar CVSS v3.1 sebagai skor default. Namun, skor dasar CVSS 4.0 akan disertakan dalam metadata kerentanan Anda saat tersedia. Skor dasar CVSS 4.0 memberikan metrik tambahan untuk meningkatkan penilaian kerentanan. Anda dapat menemukan sumber dan versi skor dasar CVSS dalam detail kerentanan untuk temuan dan temuan yang diekspor.

> **Note**
>
> Skor Amazon Inspector tidak tersedia untuk instance Linux yang menjalankan Ubuntu. Ubuntu menggunakan sistem peringkat keparahan khusus yang berbeda dari skor CVSS.

Rincian skor Amazon Inspector

Saat Anda membuka halaman detail temuan, Anda dapat memilih Inspector score and vulnerability intelligence Tab. Panel ini menunjukkan perbedaan antara skor dasar dan skor Inspector. Bagian ini menjelaskan bagaimana Amazon Inspector menetapkan peringkat keparahan berdasarkan kombinasi skor Amazon Inspector dan skor vendor untuk paket perangkat lunak. Jika skor berbeda panel ini menunjukkan penjelasan mengapa.

Di bagian metrik skor CVSS Anda dapat melihat tabel dengan perbandingan antara metrik skor dasar CVSS dan skor Inspector. Metrik yang dibandingkan adalah metrik dasar yang ditentukan dalam dokumen spesifikasi CVSS yang dikelola oleh. first.org Berikut ini adalah ringkasan metrik dasar:

Serangan Vektor

Konteks dimana kerentanan dapat dieksploitasi. Untuk temuan Amazon Inspector, ini bisa berupa Jaringan, Jaringan Berdekatan, atau Lokal.

Kompleksitas Serangan

Ini menggambarkan tingkat kesulitan yang akan dihadapi penyerang saat mengeksploitasi kerentanan. Skor rendah berarti bahwa penyerang harus memenuhi sedikit atau tidak ada kondisi tambahan untuk mengeksploitasi kerentanan. Skor tinggi berarti bahwa penyerang akan perlu

menginvestasikan sejumlah besar upaya untuk melakukan serangan yang sukses dengan kerentanan ini.

Hak Istimewa Diperlukan

Ini menggambarkan tingkat hak istimewa yang dibutuhkan penyerang untuk mengeksploitasi kerentanan.

Interaksi Pengguna

Metrik ini menyatakan jika serangan yang berhasil menggunakan kerentanan ini membutuhkan pengguna manusia, selain penyerang.

Lingkup

Ini menyatakan apakah kerentanan dalam satu komponen yang rentan berdampak pada sumber daya dalam komponen di luar lingkup keamanan komponen yang rentan. Jika nilai ini Tidak berubah, sumber daya yang terpengaruh dan sumber daya yang terkena dampak adalah sama. Jika nilai ini diubah maka komponen yang rentan dapat dieksploitasi untuk mempengaruhi sumber daya yang dikelola oleh otoritas keamanan yang berbeda.

Kerahasiaan

Ini mengukur tingkat dampak terhadap kerahasiaan data dalam sumber daya ketika kerentanan dieksploitasi. Ini berkisar dari None, di mana tidak ada kerahasiaan yang hilang, ke High di mana semua informasi dalam sumber daya diungkapkan atau informasi rahasia seperti kata sandi atau kunci enkripsi dapat diungkapkan.

Integritas

Ini mengukur tingkat dampak terhadap integritas data dalam sumber daya yang terkena dampak jika kerentanan dieksploitasi. Integritas berisiko ketika penyerang memodifikasi file dalam sumber daya yang terkena dampak. Skor berkisar dari None, di mana eksploitasi tidak memungkinkan penyerang untuk memodifikasi informasi apa pun, ke Tinggi, di mana jika dieksploitasi, kerentanan akan memungkinkan penyerang untuk memodifikasi salah satu atau semua file, atau file yang dapat dimodifikasi memiliki konsekuensi serius.

Ketersediaan

Ini mengukur tingkat dampak terhadap ketersediaan sumber daya yang terkena dampak ketika kerentanan dieksploitasi. Skor berkisar dari None, ketika kerentanan tidak memengaruhi ketersediaan sama sekali, hingga Tinggi, di mana jika dieksploitasi, penyerang dapat sepenuhnya menolak ketersediaan sumber daya, atau menyebabkan layanan menjadi tidak tersedia.

Kecerdasan Kerentanan

Bagian ini merangkum intelijen yang tersedia tentang CVE dari Amazon serta sumber intelijen keamanan standar industri seperti Cybersecurity and Infrastructure Security Agency (CISA).

> **Note**
>
> Intel dari CISA atau Amazon tidak akan tersedia untuk semua CVEs.

Anda dapat melihat detail intelijen kerentanan di konsol atau dengan menggunakan BatchGetFindingDetailsAPI. Rincian berikut tersedia di konsol:

ATT&CK

Bagian ini menunjukkan taktik, teknik, dan prosedur MITRE (TTPs) yang terkait dengan CVE. Yang terkait TTPs ditampilkan, jika ada lebih dari dua yang berlaku, TTPs Anda dapat memilih tautan untuk melihat daftar lengkap. Memilih taktik atau teknik membuka informasi tentangnya di situs web MITRE.

CISA

Bagian ini mencakup tanggal yang relevan yang terkait dengan kerentanan. Tanggal Cybersecurity and Infrastructure Security Agency (CISA) menambahkan kerentanan ke Katalog Kerentanan Tereksploitasi yang Diketahui, berdasarkan bukti eksploitasi aktif, dan tanggal jatuh tempo CISA mengharapkan sistem untuk ditambal. Informasi ini bersumber dari CISA.

Malware yang dikenal

Bagian ini mencantumkan kit dan alat eksploitasi yang dikenal yang mengeksploitasi kerentanan ini.

Terakhir kali dilaporkan

Bagian ini menunjukkan tanggal eksploitasi publik terakhir yang diketahui untuk kerentanan ini.

Memahami tingkat keparahan untuk temuan Amazon Inspector Anda

Ketika Amazon Inspector menghasilkan temuan, ia memberikan peringkat keparahan pada temuan tersebut. Peringkat keparahan membantu Anda menilai dan memprioritaskan temuan Anda.

Peringkat keparahan untuk temuan sesuai dengan skor numerik dan tingkat: informasi, rendah, sedang, tinggi, dan kritis. Amazon Inspector menentukan peringkat keparahan untuk temuan berdasarkan jenis temuan. Bagian ini menjelaskan bagaimana Amazon Inspector menentukan peringkat keparahan untuk setiap jenis temuan.

Tingkat keparahan kerentanan paket perangkat lunak

Amazon Inspector menggunakan NVD/CVSS skor sebagai dasar penilaian tingkat keparahan untuk kerentanan paket perangkat lunak. NVD/CVSS Skor adalah skor keparahan kerentanan yang diterbitkan oleh NVD dan ditentukan oleh CVSS. NVD/CVSS Skor adalah komposisi metrik keamanan, seperti kompleksitas serangan, kematangan kode eksploitasi, dan hak istimewa yang diperlukan. Amazon Inspector menghasilkan skor numerik dari 1 hingga 10 yang mencerminkan tingkat keparahan kerentanan. Amazon Inspector mengkategorikan ini sebagai skor dasar karena mencerminkan tingkat keparahan kerentanan menurut karakteristik intrinsiknya, yang konstan dari waktu ke waktu. Skor ini juga mengasumsikan dampak kasus terburuk yang wajar di berbagai lingkungan yang diterapkan. Standar CVSS v3 memetakan skor CVSS ke peringkat keparahan berikut.

Skor	Peringkat
0	Informasi
0,1—3,9	Rendah
4.0—6.9	Sedang
7.0—8.9	Tinggi
9.0—10.0	Kritis

Temuan Package vulnerability juga dapat memiliki tingkat keparahan Untriaged. Ini berarti bahwa vendor belum menetapkan skor kerentanan untuk kerentanan yang terdeteksi. Dalam hal ini, kami merekomendasikan menggunakan referensi URLs untuk temuan untuk meneliti kerentanan itu dan merespons sesuai dengan itu.

Temuan kerentanan Package mencakup skor berikut dan vektor penilaian terkait sebagai bagian dari detail temuan mereka:

- Skor EPSS

- Skor Inspector
- CVSS 3.1 dari Amazon CVE
- CVSS 3.1 dari NVD
- CVSS 2.0 dari NVD (jika berlaku)

Tingkat keparahan kerentanan kode

Untuk temuan kerentanan kode, Amazon Inspector menggunakan tingkat keparahan yang ditentukan oleh detektor Amazon Q yang menghasilkan temuan. Setiap detektor diberi tingkat keparahan menggunakan sistem penilaian CVSS v3.?

Tingkat keparahan jangkauan jaringan

Amazon Inspector menentukan tingkat keparahan kerentanan jangkauan jaringan berdasarkan layanan, port, dan protokol yang diekspos dan berdasarkan jenis jalur terbuka. Tabel berikut mendefinisikan peringkat keparahan ini. Nilai dalam kolom Peringkat jalur terbuka mewakili jalur terbuka dari gateway virtual, peered VPCs, dan jaringan. AWS Direct Connect Semua layanan, port, dan protokol lain yang terpapar memiliki peringkat tingkat keparahan informasi.

Layanan	Port TCP	Port UDP	Peringkat jalur internet	Peringkat jalur terbuka
DHCP	67, 68, 546, 547	67, 68, 546, 547	Sedang	Informasi
Elasticsearch	9300, 9200	TA	Sedang	Informasi
FTP	21	21	Tinggi	Sedang
Katalog global LDAP	3268	TA	Sedang	Informasi
Katalog global LDAP melalui TLS	3269	TA	Sedang	Informasi
HTTP	80	80	Rendah	Informasi
HTTPS	443	443	Rendah	Informasi

Kerberos	88, 464, 543, 544, 749, 751	88, 464, 749, 750, 751, 752	Sedang	Informasi
LDAP	389	389	Sedang	Informasi
LDAP melalui TLS	636	TA	Sedang	Informasi
MongoDB	27017, 27018, 27019, 28017	TA	Sedang	Informasi
MySQL	3306	TA	Sedang	Informasi
NetBIOS	137, 139	137, 138	Sedang	Informasi
NFS	111, 2049, 4045, 1110	111, 2049, 4045, 1110	Sedang	Informasi
Oracle	1521, 1630	TA	Sedang	Informasi
PostgreSQL	5432	TA	Sedang	Informasi
Layanan cetak	515	TA	Tinggi	Sedang
RDP	3389	3389	Sedang	Rendah
RPC	111, 135, 530	111, 135, 530	Sedang	Informasi
SMB	445	445	Sedang	Informasi
SSH	22	22	Sedang	Rendah
SQL Server	1433	1434	Sedang	Informasi
Syslog	601	514	Sedang	Informasi
Telnet	23	23	Tinggi	Sedang
WINS	1512, 42	1512, 42	Sedang	Informasi

Mengelola temuan di Amazon Inspector

Dengan Amazon Inspector, Anda dapat mengelola temuan Anda dengan berbagai cara. Anda dapat memfilter temuan Anda berdasarkan statusnya. Anda dapat mencari temuan Anda berdasarkan kriteria filter. Anda dapat membuat aturan penindasan untuk mengecualikan temuan dari daftar temuan Anda. Anda juga dapat mengekspor temuan ke AWS Security Hub CSPM, Amazon EventBridge, dan Amazon Simple Storage Service (Amazon S3).

Topik

Memfilter temuan Amazon Inspector Anda

Anda dapat memfilter temuan Amazon Inspector menggunakan kriteria filter. Jika temuan tidak sesuai dengan kriteria filter Anda, Amazon Inspector mengecualikan temuan dari tampilan. Bagian ini menjelaskan cara memfilter temuan Amazon Inspector Anda menggunakan kriteria filter.

Membuat filter di konsol Amazon Inspector

Di setiap tampilan temuan, Anda dapat menggunakan fungsionalitas filter untuk menemukan temuan dengan karakteristik tertentu. Filter akan dihapus ketika Anda pindah ke tampilan tab yang berbeda.

Filter terdiri dari kriteria filter, yang terdiri dari atribut filter yang dipasangkan dengan nilai filter. Temuan yang tidak sesuai dengan kriteria filter Anda dikecualikan dari daftar temuan. Misalnya, untuk melihat semua temuan yang terkait dengan akun administrator Anda, Anda dapat memilih atribut ID AWS akun dan memasangkannya dengan nilai ID AWS akun dua belas digit Anda.

Beberapa kriteria filter berlaku untuk semua temuan, sementara yang lain tersedia untuk jenis sumber daya tertentu atau jenis pencarian saja.

Untuk menerapkan filter ke tampilan temuan

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/

2. Di panel navigasi, pilih Temuan. Tampilan default menampilkan semua temuan dengan status Aktif.
3. Untuk memfilter temuan berdasarkan kriteria, pilih bilah Tambahkan filter untuk melihat daftar semua kriteria filter yang berlaku untuk tampilan tersebut. Kriteria filter yang berbeda tersedia dalam tampilan yang berbeda.
4. Pilih kriteria yang ingin Anda filter dari daftar.
5. Dari panel input kriteria masukkan nilai filter yang diinginkan untuk menentukan kriteria itu.
6. Pilih Terapkan untuk menerapkan kriteria filter tersebut ke hasil Anda saat ini. Anda dapat terus menambahkan kriteria filter lainnya dengan memilih bilah input filter lagi.
7. (Opsional) Untuk melihat temuan Anda yang ditekan atau ditutup, pilih Aktif di bilah filter, lalu pilih Ditekan atau Ditutup. Pilih Tampilkan semua untuk melihat temuan aktif, ditekan, dan tertutup dalam tampilan yang sama.

Menekan temuan Amazon Inspector

Anda dapat membuat aturan penekanan untuk menyembunyikan temuan yang sesuai dengan kriteria. Misalnya, Anda dapat membuat aturan penekanan untuk menyembunyikan temuan berdasarkan peringkat tingkat keparahannya. Jika Amazon Inspector menghasilkan temuan yang cocok dengan aturan penindasan Anda, Amazon Inspector menekan temuan tersebut dan menyembunyikannya dari pandangan. Toko Amazon Inspector menekan temuan sampai mereka diperbaiki. Setelah temuan yang ditekan diperbaiki, Amazon Inspector menutup temuan tersebut. Anda dapat melihat temuan yang ditekan di konsol.

Anda membuat aturan penindasan untuk memprioritaskan temuan Anda yang paling penting. Aturan penindasan tidak berdampak pada temuan Anda, karena hanya menyembunyikan temuan dari pandangan. Anda tidak dapat membuat aturan penindasan yang menutup atau memulihkan temuan. Anda juga dapat menekan temuan yang tidak diinginkan AWS Security Hub CSPM dengan EventBridge aturan Amazon. Prosedur di bagian ini menjelaskan cara membuat, melihat, mengedit, dan menghapus aturan penekanan.

> **Note**
>
> Hanya administrator yang didelegasikan untuk organisasi yang dapat membuat dan mengelola aturan penindasan.

Membuat aturan penindasan

Anda dapat membuat aturan penekanan untuk memfilter daftar temuan yang ditampilkan secara default. Anda dapat membuat aturan penekanan secara terprogram dengan menggunakan CreateFilterAPI dan menentukan SUPPRESS sebagai nilai untuk. `action`

> **Note**
>
> Hanya akun yang berdiri sendiri dan administrator yang didelegasikan Amazon Inspector yang dapat membuat dan mengelola aturan penindasan. Anggota dalam organisasi tidak akan melihat opsi untuk aturan penindasan di panel navigasi.

Untuk membuat aturan penindasan (konsol)

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Di panel navigasi, pilih Aturan penindasan. Kemudian, pilih Buat aturan.
3. Untuk setiap kriteria, lakukan hal berikut:
   - Pilih bilah filter untuk melihat daftar kriteria filter yang dapat Anda tambahkan ke aturan penekanan Anda.
   - Pilih kriteria filter untuk aturan penekanan Anda.
4. Setelah selesai menambahkan kriteria, masukkan nama untuk aturan dan deskripsi opsional.
5. Pilih Simpan aturan. Amazon Inspector segera menerapkan aturan penindasan baru dan menyembunyikan temuan apa pun yang sesuai dengan kriteria.

Melihat temuan yang ditekan

Secara default, Amazon Inspector tidak menampilkan temuan yang ditekan di konsol Amazon Inspector. Namun, Anda dapat melihat temuan yang ditekan oleh aturan tertentu.

Untuk melihat temuan yang ditekan

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Di panel navigasi, pilih Aturan penindasan.

3. Dalam daftar aturan penindasan, pilih judul aturan.

Mengedit aturan penindasan

Anda dapat membuat perubahan pada aturan penindasan kapan saja.

Untuk memodifikasi aturan penindasan

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Aturan penindasan.
3. Pilih nama aturan penekanan yang ingin Anda ubah, lalu pilih Edit.
4. Buat perubahan yang Anda inginkan, lalu pilih Simpan.

Menghapus aturan penindasan

Anda dapat menghapus aturan penindasan. Jika Anda menghapus aturan penindasan, Amazon Inspector berhenti menekan kemunculan temuan baru dan yang sudah ada yang memenuhi kriteria aturan dan yang tidak ditekan oleh aturan lain.

Setelah Anda menghapus aturan penekanan, kemunculan temuan baru dan yang sudah ada yang memenuhi kriteria aturan memiliki status Aktif. Ini berarti bahwa mereka muncul secara default di konsol Amazon Inspector. Selain itu, Amazon Inspector menerbitkan temuan ini ke AWS Security Hub CSPM dan Amazon sebagai acara. EventBridge

Untuk menghapus aturan penindasan

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Di panel navigasi, pilih Aturan penindasan.
3. Pilih kotak centang di sebelah judul aturan penekanan yang ingin Anda hapus.
4. Pilih Hapus, lalu konfirmasikan pilihan Anda untuk menghapus aturan secara permanen.

Mengekspor laporan temuan Amazon Inspector

Laporan temuan adalah file CSV atau JSON yang memberikan snapshot rinci dari temuan Anda. Anda dapat mengekspor laporan temuan ke AWS Security Hub CSPM, Amazon EventBridge, dan Amazon Simple Storage Service (Amazon S3). Saat Anda mengonfigurasi laporan temuan, Anda menentukan temuan mana yang akan disertakan di dalamnya. Secara default, laporan temuan Anda mencakup data untuk semua temuan aktif Anda. Jika Anda adalah administrator yang didelegasikan untuk organisasi, laporan temuan Anda menyertakan data untuk semua akun anggota di organisasi. Untuk menyesuaikan laporan temuan, buat dan terapkan filter ke dalamnya.

Saat Anda mengekspor laporan temuan, Amazon Inspector mengenkripsi data temuan Anda dengan yang AWS KMS key Anda tentukan. Setelah Amazon Inspector mengenkripsi data temuan Anda, Amazon Inspector menyimpan laporan temuan Anda di bucket Amazon S3 yang Anda tentukan. AWS KMS Kunci Anda harus digunakan AWS Region sama dengan bucket Amazon S3 Anda. Kebijakan AWS KMS utama Anda harus mengizinkan Amazon Inspector untuk menggunakannya, dan kebijakan bucket Amazon S3 Anda harus mengizinkan Amazon Inspector untuk menambahkan objek ke dalamnya. Setelah mengekspor laporan temuan, Anda dapat mengunduhnya dari bucket Amazon S3 atau mentransfernya ke lokasi baru. Anda juga dapat menggunakan bucket Amazon S3 sebagai repositori untuk laporan temuan yang diekspor lainnya.

Bagian ini menjelaskan cara mengekspor laporan temuan di konsol Amazon Inspector. Tugas berikut mengharuskan Anda memverifikasi izin, mengonfigurasi bucket Amazon S3, mengonfigurasi, dan mengonfigurasi AWS KMS key serta mengekspor laporan temuan.

> **Note**
>
> Jika Anda mengekspor laporan temuan dengan Amazon Inspector CreateFindingsReportAPI, Anda hanya dapat melihat temuan aktif Anda. Jika Anda ingin melihat temuan Anda yang ditekan atau tertutup, Anda harus menentukan `SUPPRESSED` atau `CLOSED` sebagai bagian dari kriteria filter Anda.

Tugas

- Langkah 1: Verifikasi izin Anda
- Langkah 2: Konfigurasikan bucket S3
- Langkah 3: Konfigurasikan AWS KMS key
- Langkah 4: Konfigurasikan dan ekspor laporan temuan

- Memecahkan masalah kesalahan ekspor

Langkah 1: Verifikasi izin Anda

> **Note**
>
> Setelah Anda mengekspor laporan temuan untuk pertama kalinya, langkah 1-3 bersifat opsional. Mengikuti langkah-langkah ini didasarkan pada apakah Anda ingin menggunakan bucket Amazon S3 yang sama dan AWS KMS key untuk laporan temuan yang diekspor lainnya. Jika Anda ingin mengekspor laporan temuan secara terprogram setelah menyelesaikan langkah 1-3, gunakan CreateFindingsReportpengoperasian Amazon Inspector API.

Sebelum mengekspor laporan temuan dari Amazon Inspector, verifikasi bahwa Anda memiliki izin yang Anda perlukan untuk mengekspor laporan temuan dan mengonfigurasi sumber daya untuk mengenkripsi dan menyimpan laporan. Untuk memverifikasi izin Anda, gunakan AWS Identity and Access Management (IAM) untuk meninjau kebijakan IAM yang dilampirkan pada identitas IAM Anda. Kemudian bandingkan informasi dalam kebijakan tersebut dengan daftar tindakan berikut yang harus diizinkan untuk dilakukan untuk mengekspor laporan temuan.

Amazon Inspector

Untuk Amazon Inspector, verifikasi bahwa Anda diizinkan untuk melakukan tindakan berikut:

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

Tindakan ini memungkinkan Anda untuk mengambil data temuan untuk akun Anda dan mengekspor data tersebut dalam laporan temuan.

Jika Anda berencana untuk mengekspor laporan besar secara terprogram, Anda juga dapat memverifikasi bahwa Anda diizinkan untuk melakukan tindakan berikut:`inspector2:GetFindingsReportStatus`, untuk memeriksa status laporan, dan`inspector2:CancelFindingsReport`, untuk membatalkan ekspor yang sedang berlangsung.

AWS KMS

Untuk AWS KMS, verifikasi bahwa Anda diizinkan untuk melakukan tindakan berikut:

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

Tindakan ini memungkinkan Anda untuk mengambil dan memperbarui kebijakan kunci untuk AWS KMS key yang Anda inginkan Amazon Inspector gunakan untuk mengenkripsi laporan Anda.

Untuk menggunakan konsol Amazon Inspector untuk mengekspor laporan, pastikan juga bahwa Anda diizinkan melakukan tindakan berikut: AWS KMS

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

Tindakan ini memungkinkan Anda untuk mengambil dan menampilkan informasi tentang AWS KMS keys untuk akun Anda. Anda kemudian dapat memilih salah satu kunci ini untuk mengenkripsi laporan Anda.

Jika Anda berencana untuk membuat kunci KMS baru untuk enkripsi laporan Anda, Anda juga harus diizinkan untuk melakukan `kms:CreateKey` tindakan.

Amazon S3

Untuk Amazon S3, verifikasi bahwa Anda diizinkan untuk melakukan tindakan berikut:

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

Tindakan ini memungkinkan Anda membuat dan mengonfigurasi bucket S3 tempat Amazon Inspector menyimpan laporan Anda. Mereka juga memungkinkan Anda untuk menambah dan menghapus objek dari ember.

Jika Anda berencana menggunakan konsol Amazon Inspector untuk mengekspor laporan, pastikan juga bahwa Anda diizinkan untuk melakukan tindakan `s3:ListAllMyBuckets` dan `s3:GetBucketLocation` tindakan. Tindakan ini memungkinkan Anda untuk mengambil dan menampilkan informasi tentang bucket S3 untuk akun Anda. Anda kemudian dapat memilih salah satu ember ini untuk menyimpan laporan.

Jika Anda tidak diizinkan untuk melakukan satu atau beberapa tindakan yang diperlukan, mintalah bantuan AWS administrator Anda sebelum melanjutkan ke langkah berikutnya.

Langkah 2: Konfigurasikan bucket S3

Setelah memverifikasi izin, Anda siap mengonfigurasi bucket S3 tempat Anda ingin menyimpan laporan temuan. Ini bisa berupa bucket yang sudah ada untuk akun Anda sendiri, atau bucket yang sudah ada yang dimiliki oleh orang lain Akun AWS dan Anda diizinkan untuk mengaksesnya. Jika Anda ingin menyimpan laporan Anda di bucket baru, buat bucket sebelum melanjutkan.

Bucket S3 harus AWS Region sama dengan data temuan yang ingin Anda ekspor. Misalnya, jika Anda menggunakan Amazon Inspector di Wilayah AS Timur (Virginia N.) dan Anda ingin mengekspor data temuan untuk Wilayah tersebut, bucket tersebut juga harus berada di Wilayah AS Timur (Virginia N.).

Selain itu, kebijakan bucket harus mengizinkan Amazon Inspector untuk menambahkan objek ke bucket. Topik ini menjelaskan cara memperbarui kebijakan bucket dan memberikan contoh pernyataan untuk ditambahkan ke kebijakan. Untuk informasi mendetail tentang menambahkan dan memperbarui kebijakan bucket, lihat Menggunakan kebijakan bucket di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Jika Anda ingin menyimpan laporan di bucket S3 yang dimiliki oleh akun lain, bekerjalah dengan pemilik bucket untuk memperbarui kebijakan bucket. Dapatkan juga URI untuk bucket. Anda harus memasukkan URI ini saat mengekspor laporan.

Untuk memperbarui kebijakan bucket

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon S3 di /s3. https://console.aws.amazon.com
2. Di panel navigasi, pilih Bucket.
3. Pilih bucket S3 tempat Anda ingin menyimpan laporan temuan.
4. Pilih tab Izin.
5. Di bagian Kebijakan bucket, pilih Edit.
6. Salin pernyataan contoh berikut ke clipboard Anda:

   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. Di editor kebijakan Bucket di konsol Amazon S3, tempelkan pernyataan sebelumnya ke dalam kebijakan untuk menambahkannya ke kebijakan.

   Ketika Anda menambahkan pernyataan, pastikan bahwa sintaksnya valid. Kebijakan bucket menggunakan format JSON. Ini berarti Anda perlu menambahkan koma sebelum atau sesudah pernyataan, tergantung di mana Anda menambahkan pernyataan ke kebijakan. Jika Anda menambahkan pernyataan sebagai pernyataan terakhir, tambahkan koma setelah tanda kurung kurung penutup untuk pernyataan sebelumnya. Jika Anda menambahkannya sebagai pernyataan pertama atau di antara dua pernyataan yang ada, tambahkan koma setelah tanda kurung kurung penutup untuk pernyataan tersebut.

8. Perbarui pernyataan dengan nilai yang benar untuk lingkungan Anda, di mana:

   - *amzn-s3-demo-bucket*adalah nama ember.
   - *111122223333*adalah ID akun untuk Anda Akun AWS.

- *Region*adalah AWS Region tempat Anda menggunakan Amazon Inspector dan ingin mengizinkan Amazon Inspector menambahkan laporan ke ember. Misalnya, `us-east-1` untuk Wilayah AS Timur (Virginia N.).

> **Note**
>
> Jika Anda menggunakan Amazon Inspector secara manual diaktifkan AWS Region, tambahkan juga kode Region yang sesuai ke nilai untuk bidang tersebut`Service`. Bidang ini menentukan prinsipal layanan Amazon Inspector.
> Misalnya, jika Anda menggunakan Amazon Inspector di Wilayah Timur Tengah (Bahrain), yang memiliki kode Wilayah`me-south-1`, ganti `inspector2.amazonaws.com` dengan `inspector2.me-south-1.amazonaws.com` dalam pernyataan.

Perhatikan bahwa pernyataan contoh mendefinisikan kondisi yang menggunakan dua kunci kondisi global IAM:

- aws: SourceAccount — Kondisi ini memungkinkan Amazon Inspector untuk menambahkan laporan ke bucket hanya untuk akun Anda. Ini mencegah Amazon Inspector menambahkan laporan ke bucket untuk akun lain. Lebih khusus lagi, kondisi menentukan akun mana yang dapat menggunakan bucket untuk sumber daya dan tindakan yang ditentukan oleh `aws:SourceArn` kondisi.

  Untuk menyimpan laporan akun tambahan di bucket, tambahkan ID akun untuk setiap akun tambahan ke kondisi ini. Contoh:

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

- aws: SourceArn — Kondisi ini membatasi akses ke bucket berdasarkan sumber objek yang ditambahkan ke bucket. Ini mencegah orang lain Layanan AWS menambahkan objek ke ember. Ini juga mencegah Amazon Inspector menambahkan objek ke bucket saat melakukan tindakan lain untuk akun Anda. Lebih khusus lagi, kondisi ini memungkinkan Amazon Inspector untuk menambahkan objek ke bucket hanya jika objek adalah laporan temuan, dan hanya jika laporan tersebut dibuat oleh akun dan di Wilayah yang ditentukan dalam kondisi.

Untuk mengizinkan Amazon Inspector melakukan tindakan yang ditentukan untuk akun tambahan, tambahkan Amazon Resource Names (ARNs) untuk setiap akun tambahan ke kondisi ini. Contoh:

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

Akun yang ditentukan oleh `aws:SourceAccount` dan `aws:SourceArn` kondisi harus cocok.

Kedua kondisi tersebut membantu mencegah Amazon Inspector digunakan sebagai wakil yang bingung selama transaksi dengan Amazon S3. Meskipun kami tidak merekomendasikannya, Anda dapat menghapus kondisi ini dari kebijakan bucket.

9. Setelah Anda selesai memperbarui kebijakan bucket, pilih Simpan perubahan.

Langkah 3: Konfigurasikan AWS KMS key

Setelah memverifikasi izin dan mengonfigurasi bucket S3, tentukan yang ingin digunakan Amazon Inspector untuk mengenkripsi laporan temuan AWS KMS key Anda. Kuncinya harus berupa kunci KMS enkripsi simetris yang dikelola pelanggan. Selain itu, kuncinya harus AWS Region sama dengan bucket S3 yang Anda konfigurasikan untuk menyimpan laporan.

Kuncinya dapat berupa kunci KMS yang ada dari akun Anda sendiri, atau kunci KMS yang ada yang dimiliki akun lain. Jika Anda ingin menggunakan kunci KMS baru, buat kunci sebelum melanjutkan. Jika Anda ingin menggunakan kunci yang ada yang dimiliki akun lain, dapatkan Nama Sumber Daya Amazon (ARN) dari kunci tersebut. Anda harus memasukkan ARN ini saat mengekspor laporan dari Amazon Inspector. Untuk informasi tentang membuat dan meninjau pengaturan kunci KMS, lihat Mengelola kunci di Panduan AWS Key Management Service Pengembang.

Setelah Anda menentukan kunci KMS mana yang ingin Anda gunakan, berikan izin kepada Amazon Inspector untuk menggunakan kunci tersebut. Jika tidak, Amazon Inspector tidak akan dapat mengenkripsi dan mengekspor laporan. Untuk memberikan izin kepada Amazon Inspector untuk menggunakan kunci, perbarui kebijakan kunci untuk kunci tersebut. Untuk informasi terperinci tentang kebijakan utama dan mengelola akses ke kunci KMS, lihat Kebijakan utama AWS KMS di Panduan AWS Key Management Service Pengembang.

> **Note**
>
> Prosedur berikut adalah memperbarui kunci yang ada untuk memungkinkan Amazon Inspector menggunakannya. Jika Anda tidak memiliki kunci yang ada, lihat Membuat kunci di Panduan AWS Key Management Service Pengembang.

Untuk memperbarui kebijakan utama

1. Masuk menggunakan kredensil Anda, lalu buka AWS KMS konsol di https://console.aws.amazon.com /kms.
2. Di panel navigasi, pilih Kunci yang dikelola pelanggan.
3. Pilih kunci KMS yang ingin Anda gunakan untuk mengenkripsi laporan. Kuncinya harus berupa kunci enkripsi simetris (SYMMETRIC_DEFAULT).
4. Di tab Kebijakan kunci, pilih Edit. Jika Anda tidak melihat kebijakan kunci dengan tombol Edit, Anda harus terlebih dahulu memilih Beralih ke tampilan kebijakan.
5. Salin pernyataan contoh berikut ke clipboard Anda:

```
{
    "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. Di editor kebijakan kunci di AWS KMS konsol, tempelkan pernyataan sebelumnya ke kebijakan kunci untuk menambahkannya ke kebijakan.

   Ketika Anda menambahkan pernyataan, pastikan bahwa sintaksnya valid. Kebijakan kunci menggunakan format JSON. Ini berarti Anda perlu menambahkan koma sebelum atau sesudah pernyataan, tergantung di mana Anda menambahkan pernyataan ke kebijakan. Jika Anda menambahkan pernyataan sebagai pernyataan terakhir, tambahkan koma setelah tanda kurung kurung penutup untuk pernyataan sebelumnya. Jika Anda menambahkannya sebagai pernyataan pertama atau di antara dua pernyataan yang ada, tambahkan koma setelah tanda kurung kurung penutup untuk pernyataan tersebut.

7. Perbarui pernyataan dengan nilai yang benar untuk lingkungan Anda, di mana:

   - *111122223333*adalah ID akun untuk Anda Akun AWS.
   - *Region*adalah AWS Region di mana Anda ingin mengizinkan Amazon Inspector untuk mengenkripsi laporan dengan kunci. Misalnya, `us-east-1` untuk Wilayah AS Timur (Virginia N.).

   > **Note**
   >
   > Jika Anda menggunakan Amazon Inspector secara manual diaktifkan AWS Region, tambahkan juga kode Region yang sesuai ke nilai untuk bidang tersebut`Service`. Misalnya, jika Anda menggunakan Amazon Inspector di Wilayah Timur Tengah (Bahrain), ganti dengan. `inspector2.amazonaws.com` `inspector2.me-south-1.amazonaws.com`

   Seperti pernyataan contoh untuk kebijakan bucket pada langkah sebelumnya, `Condition` bidang dalam contoh ini menggunakan dua kunci kondisi global IAM:

   - aws: SourceAccount — Kondisi ini memungkinkan Amazon Inspector untuk melakukan tindakan yang ditentukan hanya untuk akun Anda. Lebih khusus lagi, ini menentukan akun mana yang dapat melakukan tindakan yang ditentukan untuk sumber daya dan tindakan yang ditentukan oleh `aws:SourceArn` kondisi.

     Untuk mengizinkan Amazon Inspector melakukan tindakan yang ditentukan untuk akun tambahan, tambahkan ID akun untuk setiap akun tambahan ke kondisi ini. Contoh:

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

- aws: SourceArn — Kondisi ini Layanan AWS mencegah other melakukan tindakan yang ditentukan. Ini juga mencegah Amazon Inspector menggunakan kunci saat melakukan tindakan lain untuk akun Anda. Dengan kata lain, ini memungkinkan Amazon Inspector untuk mengenkripsi objek S3 dengan kunci hanya jika objek adalah laporan temuan, dan hanya jika laporan tersebut dibuat oleh akun dan di Wilayah yang ditentukan dalam kondisi.

  Untuk mengizinkan Amazon Inspector melakukan tindakan yang ditentukan untuk akun tambahan, tambahkan ARNs untuk setiap akun tambahan ke kondisi ini. Contoh:

  ```
  "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/*"
  ]
  ```

  Akun yang ditentukan oleh `aws:SourceAccount` dan `aws:SourceArn` kondisi harus cocok.

Kondisi ini membantu mencegah Amazon Inspector digunakan sebagai wakil yang bingung selama transaksi dengan. AWS KMS Meskipun kami tidak merekomendasikannya, Anda dapat menghapus kondisi ini dari pernyataan.

8. Setelah Anda selesai memperbarui kebijakan kunci, pilih Simpan perubahan.

Langkah 4: Konfigurasikan dan ekspor laporan temuan

> **Note**
>
> Anda hanya dapat mengekspor hanya satu laporan temuan satu kali. Jika ekspor sedang berlangsung, Anda harus menunggu hingga ekspor selesai sebelum mengekspor laporan temuan lain.

Setelah memverifikasi izin dan mengonfigurasi sumber daya untuk mengenkripsi dan menyimpan laporan temuan, Anda siap mengonfigurasi dan mengekspor laporan.

Untuk mengonfigurasi dan mengekspor laporan temuan

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Di panel navigasi, di bawah Temuan, pilih Semua temuan.
3. (Opsional) Dengan menggunakan bilah filter di atas tabel Temuan, tambahkan kriteria filter yang menentukan temuan mana yang akan disertakan dalam laporan. Saat Anda menambahkan kriteria, Amazon Inspector memperbarui tabel untuk menyertakan hanya temuan yang sesuai dengan kriteria. Tabel menyediakan pratinjau data yang akan berisi laporan Anda.

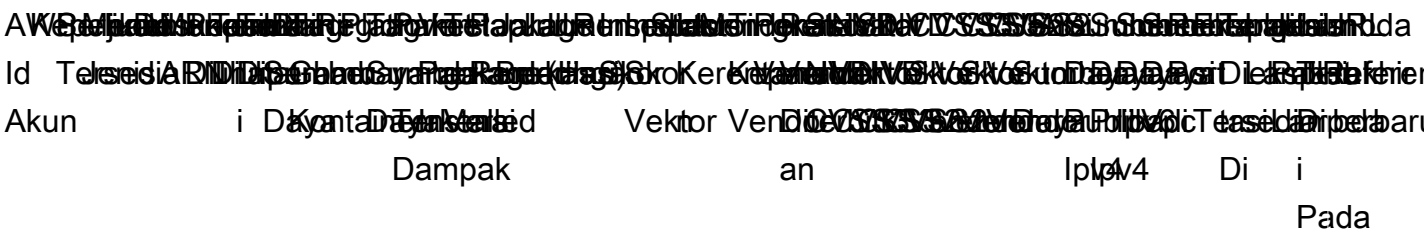
   > **Note**
   >
   > Kami menyarankan Anda menambahkan kriteria filter. Jika tidak, laporan akan menyertakan data untuk semua temuan Anda saat ini AWS Region yang berstatus Aktif. Jika Anda administrator Amazon Inspector untuk suatu organisasi, ini termasuk data temuan untuk semua akun anggota di organisasi Anda.
   > Jika laporan menyertakan data untuk semua atau banyak temuan, perlu waktu lama untuk menghasilkan dan mengekspor laporan, dan Anda hanya dapat mengekspor satu laporan pada satu waktu.

4. Pilih temuan Ekspor.
5. Di bagian Pengaturan ekspor, untuk Ekspor jenis file, tentukan format file untuk laporan:

   - Untuk membuat file JavaScript Object Notation (.json) yang berisi data, pilih JSON.

     Jika Anda memilih opsi JSON, laporan akan menyertakan semua bidang untuk setiap temuan. Untuk daftar kemungkinan bidang JSON, lihat tipe data Finding di referensi Amazon Inspector API.
   - Untuk membuat file nilai dipisahkan koma (.csv) yang berisi data, pilih CSV.

     Jika Anda memilih opsi CSV, laporan hanya akan menyertakan subset bidang untuk setiap temuan, kira-kira 45 bidang yang melaporkan atribut kunci dari temuan. Bidang meliputi: Jenis Penemuan, Judul, Tingkat Keparahan, Status, Deskripsi, Pertama Dilihat, Terakhir Terlihat, Perbaiki Tersedia, ID AWS akun, ID Sumber Daya, Tag Sumber Daya, dan Remediasi. Ini adalah tambahan untuk bidang yang menangkap detail penilaian dan referensi URLs untuk setiap temuan. Berikut ini adalah contoh header CSV dalam laporan temuan:



6. Di bawah Lokasi ekspor, untuk URI S3, tentukan bucket S3 tempat Anda ingin menyimpan laporan:

   - Untuk menyimpan laporan dalam bucket yang dimiliki akun Anda, pilih Browse S3. Amazon Inspector menampilkan tabel bucket S3 untuk akun Anda. Pilih baris untuk ember yang Anda inginkan, lalu pilih Pilih.

     > **Tip**
     >
     > Untuk juga menentukan awalan jalur Amazon S3 untuk laporan, tambahkan garis miring (/) dan awalan ke nilai di kotak URI S3. Amazon Inspector kemudian menyertakan awalan saat menambahkan laporan ke bucket, dan Amazon S3 menghasilkan jalur yang ditentukan oleh awalan.
     > Misalnya, jika Anda ingin menggunakan Akun AWS ID Anda sebagai awalan dan ID akun Anda adalah 111122223333, tambahkan **`/111122223333`** nilai di kotak URI S3. Awalan mirip dengan jalur direktori dalam bucket S3. Ini memungkinkan Anda untuk mengelompokkan objek serupa bersama-sama dalam ember, seperti Anda mungkin menyimpan file serupa bersama-sama dalam folder pada sistem file. Untuk informasi selengkapnya, lihat Mengatur objek di konsol Amazon S3 menggunakan folder di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

   - Untuk menyimpan laporan dalam bucket yang dimiliki akun lain, masukkan URI untuk bucket—misalnya, di mana DOC-EXAMPLE_BUCKET adalah nama bucket. **`s3://DOC-EXAMPLE_BUCKET`** Pemilik ember dapat menemukan informasi ini untuk Anda di properti ember.

7. Untuk kunci KMS, tentukan AWS KMS key yang ingin Anda gunakan untuk mengenkripsi laporan:

   - Untuk menggunakan kunci dari akun Anda sendiri, pilih kunci dari daftar. Daftar ini menampilkan kunci KMS enkripsi simetris yang dikelola pelanggan untuk akun Anda.

- Untuk menggunakan kunci yang dimiliki akun lain, masukkan Nama Sumber Daya Amazon (ARN) dari kunci tersebut. Pemilik kunci dapat menemukan informasi ini untuk Anda di properti kunci. Untuk informasi selengkapnya, lihat Menemukan ID kunci dan kunci ARN di Panduan AWS Key Management Service Pengembang.

8. Pilih Ekspor.

Amazon Inspector membuat laporan temuan, mengenkripsinya dengan kunci KMS yang Anda tentukan, dan menambahkannya ke bucket S3 yang Anda tentukan. Bergantung pada jumlah temuan yang Anda pilih untuk dimasukkan dalam laporan, proses ini dapat memakan waktu beberapa menit atau jam. Ketika ekspor selesai, Amazon Inspector menampilkan pesan yang menunjukkan bahwa laporan temuan Anda berhasil diekspor. Secara opsional pilih Lihat laporan dalam pesan untuk menavigasi ke laporan di Amazon S3.

Perhatikan bahwa Anda hanya dapat mengekspor satu laporan dalam satu kali. Jika ekspor sedang berlangsung, tunggu hingga ekspor selesai sebelum Anda mencoba mengekspor laporan lain.

Memecahkan masalah kesalahan ekspor

Jika terjadi kesalahan saat Anda mencoba mengekspor laporan temuan, Amazon Inspector menampilkan pesan yang menjelaskan kesalahan tersebut. Anda dapat menggunakan informasi dalam topik ini sebagai panduan untuk mengidentifikasi kemungkinan penyebab dan solusi untuk kesalahan tersebut.

Misalnya, verifikasi bahwa bucket S3 ada di bucket saat ini AWS Region dan kebijakan bucket memungkinkan Amazon Inspector untuk menambahkan objek ke bucket. Juga verifikasi bahwa AWS KMS key diaktifkan di Wilayah saat ini, dan pastikan bahwa kebijakan kunci memungkinkan Amazon Inspector untuk menggunakan kunci.

Setelah Anda mengatasi kesalahan, coba ekspor laporan lagi.

Tidak dapat memiliki beberapa laporan kesalahan

Jika Anda mencoba membuat laporan tetapi Amazon Inspector sudah membuat laporan, Anda akan menerima kesalahan yang menyatakan Alasan: Tidak dapat memiliki beberapa laporan yang sedang berlangsung. Kesalahan ini terjadi karena Amazon Inspector hanya dapat menghasilkan satu laporan untuk akun pada satu waktu.

Untuk mengatasi kesalahan, Anda dapat menunggu laporan lain selesai atau membatalkannya sebelum meminta laporan baru.

Anda dapat memeriksa status laporan dengan menggunakan GetFindingsReportStatusoperasi, operasi ini mengembalikan ID laporan dari setiap laporan yang sedang dibuat.

Jika perlu, Anda dapat menggunakan ID laporan yang diberikan oleh `GetFindingsReportStatus` operasi untuk membatalkan ekspor yang sedang berlangsung dengan menggunakan CancelFindingsReportoperasi.

Membuat tanggapan khusus terhadap temuan Amazon Inspector dengan Amazon EventBridge

Amazon Inspector membuat acara di Amazon EventBridge untuk temuan yang baru dihasilkan dan temuan agregat. Amazon Inspector juga membuat acara untuk setiap perubahan pada status temuan. Ini berarti Amazon Inspector membuat acara baru untuk temuan ketika Anda mengambil tindakan seperti memulai ulang sumber daya atau mengubah tag yang terkait dengan sumber daya. Saat Amazon Inspector membuat acara baru untuk temuan yang diperbarui, temuannya `id` tetap sama.

> **Note**
>
> Jika akun Anda adalah akun administrator yang didelegasikan oleh Amazon Inspector, EventBridge menerbitkan acara ke akun Anda dan akun anggota tempat acara tersebut berasal.

Saat menggunakan EventBridge peristiwa dengan Amazon Inspector, Anda dapat mengotomatiskan tugas untuk membantu Anda menanggapi masalah keamanan yang diungkapkan temuan Anda. Untuk menerima pemberitahuan tentang temuan Amazon Inspector berdasarkan EventBridge peristiwa, Anda harus membuat EventBridge aturan dan menentukan target untuk Amazon Inspector. EventBridge Aturan ini memungkinkan EventBridge untuk mengirim pemberitahuan untuk temuan Amazon Inspector, dan target menentukan ke mana harus mengirim notifikasi.

Amazon Inspector memancarkan peristiwa ke bus acara default di AWS Region tempat Anda saat ini menggunakan Amazon Inspector. Ini berarti Anda harus mengonfigurasi aturan acara untuk setiap AWS Region tempat Anda mengaktifkan Amazon Inspector dan mengonfigurasi Amazon Inspector untuk menerima acara. EventBridge Amazon Inspector memancarkan acara dengan upaya terbaik.

Bagian ini memberi Anda contoh skema acara dan menjelaskan cara membuat EventBridge aturan.

Skema peristiwa

Berikut ini adalah contoh format acara Amazon Inspector untuk acara pencarian EC2. Misalnya skema jenis temuan dan jenis acara lainnya, lihatEventBridge skema.


```
{
    "version": "0",
    "id": "66a7a279-5f92-971c-6d3e-c92da0950992",
    "detail-type": "Inspector2 Finding",
    "source": "aws.inspector2",
    "account": "111122223333",
    "time": "2023-01-19T22:46:15Z",
    "region": "us-east-1",
    "resources": ["i-0c2a343f1948d5205"],
    "detail": {
        "awsAccountId": "111122223333",
        "description": "\n It was discovered that the sound subsystem in the Linux
 kernel contained a\n race condition in some situations. A local attacker could use
 this to cause\n a denial of service (system crash).",
        "exploitAvailable": "YES",
        "exploitabilityDetails": {
            "lastKnownExploitAt": "Oct 24, 2022, 11:08:59 PM"
        },
        "findingArn": "arn:aws:inspector2:us-east-1:111122223333:finding/FINDING_ID",
        "firstObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "fixAvailable": "YES",
        "lastObservedAt": "Jan 19, 2023, 10:46:15 PM",
        "packageVulnerabilityDetails": {
            "cvss": [{
                "baseScore": 4.7,
                "scoringVector": "CVSS:3.1/AV:L/AC:H/PR:L/UI:N/S:U/C:N/I:N/A:H",
                "source": "NVD",
                "version": "3.1"
            }],
            "referenceUrls": ["https://lore.kernel.org/all/
CAFcO6XN7JDM4xSXGhtusQfS2mSBcx50VJKwQpCq=WeLt57aaZA@mail.gmail.com/", "https://
ubuntu.com/security/notices/USN-5792-1", "https://ubuntu.com/security/notices/
USN-5791-2", "https://ubuntu.com/security/notices/USN-5791-1", "https://ubuntu.com/
security/notices/USN-5793-2", "https://git.kernel.org/pub/scm/linux/kernel/git/
torvalds/linux.git/commit/?id=8423f0b6d513b259fdab9c9bf4aaa6188d054c2d", "https://
ubuntu.com/security/notices/USN-5793-1", "https://ubuntu.com/security/notices/
USN-5792-2", "https://ubuntu.com/security/notices/USN-5791-3", "https://ubuntu.com/
```

```
security/notices/USN-5793-4", "https://ubuntu.com/security/notices/USN-5793-3",
 "https://git.kernel.org/linus/8423f0b6d513b259fdab9c9bf4aaa6188d054c2d(6.0-rc5)",
 "https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-3303"],
            "relatedVulnerabilities": [],
            "source": "UBUNTU_CVE",
            "sourceUrl": "https://people.canonical.com/~ubuntu-security/cve/2022/
CVE-2022-3303.html",
            "vendorCreatedAt": "Sep 27, 2022, 11:15:00 PM",
            "vendorSeverity": "medium",
            "vulnerabilityId": "CVE-2022-3303",
            "vulnerablePackages": [{
                "arch": "X86_64",
                "epoch": 0,
                "fixedInVersion": "0:5.15.0.1027.31~20.04.16",
                "name": "linux-image-aws",
                "packageManager": "OS",
                "remediation": "apt update && apt install --only-upgrade linux-image-
aws",
                "version": "5.15.0.1026.30~20.04.16"
            }]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [{
            "details": {
                "awsEc2Instance": {
                    "iamInstanceProfileArn": "arn:aws:iam::111122223333:instance-
profile/AmazonSSMRoleForInstancesQuickSetup",
                    "imageId": "ami-0b7ff1a8d69f1bb35",
                    "ipV4Addresses": ["172.31.85.212", "44.203.45.27"],
                    "ipV6Addresses": [],
                    "launchedAt": "Jan 19, 2023, 7:53:14 PM",
                    "platform": "UBUNTU_20_04",
                    "subnetId": "subnet-8213f2a3",
                    "type": "t2.micro",
                    "vpcId": "vpc-ab6650d1"
                }
            },
            "id": "i-0c2a343f1948d5205",
            "partition": "aws",
            "region": "us-east-1",
```

```
            "type": "AWS_EC2_INSTANCE"
        }],
        "severity": "MEDIUM",
        "status": "ACTIVE",
        "title": "CVE-2022-3303 - linux-image-aws",
        "type": "PACKAGE_VULNERABILITY",
        "updatedAt": "Jan 19, 2023, 10:46:15 PM"
    }
}
```

Membuat EventBridge aturan untuk memberi tahu Anda tentang temuan Amazon Inspector

Untuk meningkatkan visibilitas temuan Amazon Inspector, Anda dapat EventBridge menggunakan untuk mengatur peringatan pencarian otomatis yang dikirim ke pusat pesan. Topik ini menunjukkan cara mengirim peringatan `CRITICAL` dan temuan `HIGH` tingkat keparahan ke email, Slack, atau Amazon Chime. Anda akan mempelajari cara menyiapkan topik Amazon Simple Notification Service dan kemudian menghubungkan topik tersebut ke aturan EventBridge acara.

Langkah 1. Siapkan topik dan titik akhir Amazon SNS

Untuk mengatur peringatan otomatis, Anda harus terlebih dahulu menyiapkan topik di Amazon Simple Notification Service dan menambahkan titik akhir. Untuk informasi lebih lanjut, lihat panduan SNS.

Prosedur ini menetapkan di mana Anda ingin mengirim data temuan Amazon Inspector. Topik SNS dapat ditambahkan ke aturan EventBridge acara selama atau setelah pembuatan aturan acara.

Email setup

Membuat topik SNS

1. Masuk ke konsol Amazon SNS di https://console.aws.amazon.com/sns/ v3/home.
2. Dari panel navigasi, pilih Topik, lalu pilih Buat Topik.
3. Di bagian Buat topik, pilih Standar. Selanjutnya, masukkan nama topik, seperti**`Inspector_to_Email`**. Detail lainnya bersifat opsional.
4. Pilih Buat Topik. Ini membuka panel baru dengan detail untuk topik baru Anda.
5. Di bagian Langganan, pilih Buat Langganan.

6. a. Dari menu Protokol, pilih Email.
   b. Di bidang Endpoint, masukkan alamat email yang ingin Anda terima notifikasi.

   > **Note**
   >
   > Anda akan diminta untuk mengkonfirmasi langganan Anda melalui klien email Anda setelah membuat langganan.

   c. Pilih Buat langganan.
7. Cari pesan berlangganan di kotak masuk Anda dan pilih Konfirmasi Langganan.

Slack setup

Membuat topik SNS

1. Masuk ke konsol Amazon SNS di https://console.aws.amazon.com/sns/ v3/home.
2. Dari panel navigasi, pilih Topik, lalu pilih Buat Topik.
3. Di bagian Buat topik, pilih Standar. Selanjutnya, masukkan nama topik, seperti**`Inspector_to_Slack`**. Detail lainnya bersifat opsional. Pilih Buat topik untuk menyelesaikan pembuatan titik akhir.

Mengkonfigurasi Pengembang Amazon Q di klien aplikasi obrolan

1. Arahkan ke Pengembang Amazon Q di konsol aplikasi obrolan dihttps://console.aws.amazon.com/chatbot/.
2. Dari panel Klien yang dikonfigurasi, pilih Konfigurasi klien baru.
3. Pilih Slack, lalu pilih Konfigurasi untuk mengonfirmasi.

   > **Note**
   >
   > Saat memilih Slack, Anda harus mengonfirmasi izin untuk Pengembang Amazon Q di aplikasi obrolan untuk mengakses saluran Anda dengan memilih izinkan.

4. Pilih Konfigurasi saluran baru untuk membuka panel detail konfigurasi.
   a. Masukkan nama untuk saluran.
   b. Untuk saluran Slack, pilih saluran yang ingin Anda gunakan.

c. Di Slack, salin ID saluran dari saluran pribadi dengan mengklik kanan pada nama saluran dan memilih Salin Tautan.

d. Di Konsol Manajemen AWS jendela Amazon Q Developer di aplikasi obrolan, tempel ID saluran yang Anda salin dari Slack ke bidang ID saluran pribadi.

e. Di Izin, pilih untuk membuat peran IAM menggunakan templat jika Anda belum memiliki peran.

f. Untuk templat Kebijakan, pilih Izin pemberitahuan. Ini adalah template kebijakan IAM untuk Pengembang Amazon Q dalam aplikasi obrolan. Kebijakan ini menyediakan izin baca dan daftar yang diperlukan untuk CloudWatch alarm, peristiwa, dan log, serta untuk topik Amazon SNS.

g. Untuk kebijakan pagar pembatas Saluran, pilih AmazonInspector 2. ReadOnlyAccess

h. Pilih Wilayah tempat Anda sebelumnya membuat topik SNS, lalu pilih topik Amazon SNS yang Anda buat untuk mengirim pemberitahuan ke saluran Slack.

5. Pilih Konfigurasi.

Amazon Chime setup

Membuat topik SNS

1. Masuk ke konsol Amazon SNS di https://console.aws.amazon.com/sns/ v3/home.
2. Pilih Topik dari panel navigasi, lalu pilih Buat Topik.
3. Di bagian Buat topik, pilih Standar. Selanjutnya, masukkan nama topik, seperti**`Inspector_to_Chime`**. Detail lainnya bersifat opsional. Pilih Buat topik untuk diselesaikan.

Mengkonfigurasi Pengembang Amazon Q di klien aplikasi obrolan

1. Arahkan ke Pengembang Amazon Q di konsol aplikasi obrolan dihttps://console.aws.amazon.com/chatbot/.
2. Dari panel Klien yang dikonfigurasi, pilih Konfigurasikan klien baru.
3. Pilih Chime, lalu pilih Konfigurasi untuk mengonfirmasi.
4. Dari panel Detail konfigurasi, masukkan nama untuk saluran.
5. Di Amazon Chime, buka ruang obrolan yang diinginkan.

   a. Pilih ikon roda gigi di sudut kanan atas dan pilih Kelola webhook dan bot.

b. Pilih Salin URL untuk menyalin URL webhook ke clipboard Anda.

6. Di jendela Konsol Manajemen AWS Amazon Q Developer di aplikasi obrolan, tempel URL yang Anda salin ke bidang URL Webhook.
7. Di Izin, pilih untuk membuat peran IAM menggunakan templat jika Anda belum memiliki peran.
8. Untuk templat Kebijakan, pilih Izin pemberitahuan. Ini adalah template kebijakan IAM untuk Pengembang Amazon Q dalam aplikasi obrolan. Ini memberikan izin baca dan daftar yang diperlukan untuk CloudWatch alarm, peristiwa, dan log, dan untuk topik Amazon SNS.
9. Pilih Wilayah tempat Anda membuat topik SNS sebelumnya, lalu pilih topik Amazon SNS yang Anda buat untuk mengirim notifikasi ke ruang Amazon Chime.
10. Pilih Konfigurasi.

Langkah 2. Buat EventBridge aturan untuk temuan Amazon Inspector

1. Masuk menggunakan kredensil Anda.
2. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
3. Pilih Aturan dari panel navigasi, lalu pilih Buat aturan.
4. Masukkan nama dan deskripsi opsional untuk aturan Anda.
5. Pilih Aturan dengan pola acara dan kemudian Berikutnya.
6. Di panel Pola Acara, pilih Pola kustom (editor JSON).
7. Tempelkan JSON berikut ke editor.

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

> **Note**
>
> Pola ini mengirimkan pemberitahuan untuk setiap temuan aktif `CRITICAL` atau `HIGH` tingkat keparahan yang terdeteksi oleh Amazon Inspector.

Pilih Berikutnya ketika Anda selesai memasukkan pola acara.

8. Pada halaman Pilih target, pilih Layanan AWS. Kemudian, untuk Pilih jenis target, pilih topik SNS.
9. Untuk Topik, pilih nama topik SNS yang Anda buat di langkah 1. Lalu pilih Selanjutnya.
10. Tambahkan tag opsional jika diperlukan dan pilih Berikutnya.
11. Tinjau aturan Anda dan kemudian pilih Buat aturan.

EventBridge untuk lingkungan multi-akun Amazon Inspector

Jika Anda administrator yang didelegasikan Amazon Inspector, EventBridge aturan akan muncul di akun Anda berdasarkan temuan yang berlaku dari akun anggota Anda. Jika Anda mengatur pemberitahuan temuan melalui EventBridge akun administrator, seperti yang dijelaskan di bagian sebelumnya, Anda akan menerima pemberitahuan tentang beberapa akun. Dengan kata lain, Anda akan diberi tahu tentang temuan dan peristiwa yang dihasilkan oleh akun anggota Anda selain yang dihasilkan oleh akun Anda sendiri.

Anda dapat menggunakan rincian JSON `accountId` dari temuan untuk mengidentifikasi akun anggota dari mana temuan Amazon Inspector berasal.

Bekerja dengan dasbor di Amazon Inspector

Dasbor menyediakan snapshot statistik agregat untuk sumber daya yang dipindai Amazon Inspector. Gunakan dasbor untuk mempelajari tentang cakupan untuk lingkungan Anda dan temuan penting.

> **Note**
>
> Jika akun Anda adalah akun administrator yang didelegasikan untuk organisasi, dasbor menampilkan informasi untuk akun Anda dan setiap akun lain di organisasi.

Topik ini menjelaskan cara melihat dasbor dan memahami komponen yang membentuk dasbor.

Topik

Melihat dasbor

Dasbor menunjukkan ikhtisar cakupan untuk lingkungan Anda dan temuan penting. Dasbor menyegarkan data secara otomatis setiap lima menit. Anda dapat menyegarkan data secara manual dengan memilih ikon penyegaran di dekat sudut kanan atas layar. Anda dapat melihat data pendukung untuk suatu item dengan memilih item.

> **Note**
>
> Jika akun Anda adalah akun administrator yang didelegasikan untuk organisasi, Anda dapat melihat statistik agregat untuk akun anggota dengan memasukkan ID akun anggota di bidang Akun.

Untuk melihat dasbor:

1. Masuk menggunakan kredensi Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Dasbor.

Memahami komponen dasbor dan menafsirkan data

Setiap bagian dasbor memberikan wawasan tentang metrik utama dan data temuan, sehingga Anda dapat memahami postur kerentanan AWS sumber daya Anda saat ini. AWS Region

Cakupan lingkungan

Bagian cakupan Lingkungan menyediakan statistik tentang sumber daya yang dipindai oleh Amazon Inspector. Di bagian ini, Anda dapat melihat hitungan dan persentase EC2 instans Amazon, gambar Amazon ECR, dan AWS Lambda fungsi yang dipindai oleh Amazon Inspector. Jika Anda mengelola beberapa akun melalui AWS Organizations administrator yang didelegasikan Amazon Inspector, Anda juga akan melihat jumlah total akun organisasi, nomor dengan Amazon Inspector diaktifkan, dan persentase cakupan yang dihasilkan untuk organisasi. Anda juga dapat menggunakan bagian ini untuk menentukan sumber daya mana yang tidak dicakup oleh Amazon Inspector. Sumber daya ini mungkin mengandung kerentanan yang dapat dieksploitasi untuk membahayakan organisasi Anda. Untuk detail selengkapnya, lihat Menilai cakupan Amazon Inspector dari lingkungan Anda AWS.

Memilih grup cakupan akan membawa Anda ke halaman Manajemen akun untuk pengelompokan yang Anda pilih. Halaman manajemen akun menunjukkan rincian tentang akun, EC2 instans Amazon, dan repositori Amazon ECR yang dicakup oleh Amazon Inspector.

Grup cakupan berikut tersedia:

- Akun
- Instans
- Repositori kontainer
- Image kontainer
- Lambda

Temuan kritis

Bagian Temuan Kritis memberikan hitungan kerentanan kritis di lingkungan Anda dan jumlah total semua temuan di lingkungan Anda. Di bagian ini, jumlah ditampilkan per sumber daya dan jenis penilaian. Untuk informasi lebih lanjut tentang temuan penting dan bagaimana Amazon Inspector menentukan kekritisan, lihat. Memahami temuan Amazon Inspector

Memilih grup temuan kritis akan membawa Anda ke halaman Semua temuan dan secara otomatis menerapkan filter untuk menampilkan semua temuan penting yang cocok dengan pengelompokan yang Anda pilih.

Kelompok temuan penting berikut tersedia:

- Temuan pemindaian kode Amazon Inspector
- Temuan EC2 contoh Amazon
- Temuan gambar wadah Amazon ECR
- Temuan fungsi Lambda

Remediasi berbasis risiko

Bagian remediasi berbasis risiko menunjukkan lima paket perangkat lunak teratas dengan kerentanan kritis yang memengaruhi sebagian besar sumber daya di lingkungan Anda. Remediasi paket-paket ini dapat secara signifikan mengurangi jumlah risiko kritis terhadap lingkungan Anda. Pilih nama paket perangkat lunak untuk melihat detail kerentanan terkait dan sumber daya yang terpengaruh.

Akun dengan temuan paling kritis

Bagian Akun dengan temuan paling kritis menunjukkan lima AWS akun teratas di lingkungan Anda dengan temuan paling kritis, dan jumlah total temuan untuk akun itu. Bagian ini hanya dapat dilihat dari akun administrator yang didelegasikan saat Amazon Inspector dikonfigurasi untuk pemindaian multi-akun. AWS Organizations Tampilan ini membantu administrator yang didelegasikan memahami akun mana yang paling berisiko dalam organisasi.

Pilih ID Akun untuk melihat informasi selengkapnya tentang akun anggota yang terpengaruh.

Repositori Amazon ECR dengan temuan paling penting

Repositori Elastic Container Registry (ECR) dengan bagian temuan paling kritis menunjukkan lima repositori ECR Amazon teratas di lingkungan Anda dengan temuan gambar kontainer paling penting. Tampilan menunjukkan nama repositori, pengenal AWS akun, tanggal pembuatan repositori, jumlah kerentanan kritis, dan jumlah total kerentanan. Pandangan ini membantu Anda mengidentifikasi repositori mana yang paling berisiko.

Pilih nama Repositori untuk melihat informasi lebih lanjut tentang repositori yang terpengaruh.

Gambar kontainer dengan temuan paling kritis

Gambar Container dengan bagian temuan paling kritis menunjukkan lima gambar kontainer teratas di lingkungan Anda dengan temuan paling kritis. Tampilan menampilkan data tag gambar, nama repositori, intisari gambar, pengenal AWS akun, jumlah kerentanan kritis, dan jumlah total kerentanan. Tampilan ini membantu pemilik aplikasi mengidentifikasi gambar kontainer mana yang mungkin perlu dibangun kembali dan diluncurkan kembali.

Pilih gambar Container untuk melihat informasi selengkapnya tentang gambar kontainer yang terpengaruh.

Contoh dengan temuan paling kritis

Bagian Instans dengan temuan paling kritis menunjukkan lima EC2 contoh Amazon teratas dengan temuan paling kritis. Tampilan menampilkan pengenal instans, pengenal AWS akun, pengidentifikasi Amazon Machine Image (AMI), jumlah kerentanan kritis, dan jumlah total kerentanan. Tampilan ini membantu pemilik infrastruktur mengidentifikasi instance mana yang mungkin memerlukan penambalan.

Pilih ID Instance untuk melihat informasi selengkapnya tentang EC2 instans Amazon yang terpengaruh.

Amazon Machine Images (AMI) dengan temuan paling kritis

Gambar Mesin Amazon (AMIs) dengan bagian temuan paling kritis menunjukkan lima teratas AMIs di lingkungan Anda dengan temuan paling kritis. Tampilan menunjukkan pengenal AMI, pengenal AWS akun, jumlah EC2 instans yang terpengaruh yang berjalan di lingkungan, tanggal pembuatan AMI, platform sistem operasi AMI, jumlah kerentanan kritis, dan jumlah total kerentanan. Pandangan ini membantu pemilik infrastruktur mengidentifikasi mana yang AMIs mungkin memerlukan pembangunan kembali.

Pilih Instans yang terpengaruh untuk melihat informasi selengkapnya tentang instans yang diluncurkan dari AMI yang terpengaruh.

AWS Lambda berfungsi dengan temuan paling kritis

AWS Lambda Fungsi dengan bagian temuan paling kritis menunjukkan lima fungsi Lambda teratas di lingkungan Anda dengan temuan paling kritis. Tampilan menunjukkan nama fungsi Lambda, pengenal AWS akun, lingkungan runtime, jumlah kerentanan kritis, jumlah kerentanan tinggi, dan jumlah total kerentanan. Tampilan ini membantu pemilik infrastruktur mengidentifikasi fungsi Lambda mana yang mungkin memerlukan perbaikan.

Pilih Nama fungsi untuk melihat informasi selengkapnya tentang AWS Lambda fungsi yang terpengaruh.

Pemindaian kode Amazon Inspector dengan temuan paling kritis

Proyek dengan bagian kerentanan kode paling kritis menunjukkan lima proyek teratas dengan temuan penting. Anda dapat memilih proyek untuk melihat detail tentang temuan. Ketika Anda memilih proyek, Anda diarahkan ke repositori tempat temuan berada. Tab temuan menunjukkan

nama-nama temuan Anda dan peringkat keparahannya. Ini menunjukkan jenis analisis apa yang digunakan untuk menghasilkan temuan Anda. Ini juga menunjukkan berapa usia temuan Anda dan statusnya.

Mencari database kerentanan Amazon Inspector

Anda dapat mencari database kerentanan Amazon Inspector untuk mencari kerentanan dan eksposur umum (CVE). Amazon Inspector menggunakan informasi dari database kerentanan untuk menghasilkan detail yang terkait dengan ID CVE. Anda dapat melihat detail ini di layar detail CVE. Amazon Inspector melacak dan menghasilkan temuan untuk kerentanan perangkat lunak dalam database kerentanan. Amazon Inspector hanya mendukung CVEs dengan platform yang tercantum di bagian Platform Deteksi pada layar detail CVE. Bagian ini menjelaskan cara mencari database vulnerability Amazon Inspector menggunakan ID CVE.

> **Note**
>
> Saat ini, pencarian CVE tidak mendukungMicrosoft Windows.

Mencari database kerentanan

Bagian ini menjelaskan cara mencari database kerentanan di konsol dan dengan Amazon Inspector API.

> **Note**
>
> Anda harus mengaktifkan Amazon Inspector di saat ini AWS Region sebelum Anda dapat mencari database kerentanan.

Console

1. Masuk menggunakan kredensi Anda, lalu buka konsol Amazon Inspector di v2/home https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Pencarian basis data kerentanan.
3. Di bilah pencarian, masukkan ID CVE, dan pilih Cari.

API

Jalankan Amazon Inspector SearchVulnerabilitiesAPI, dan berikan satu ID CVE seperti `filterCriteria` dalam format berikut:. `CVE-<year>-<ID>`

Memahami detail CVE

Bagian ini menjelaskan cara menginterpet halaman detail CVE.

Rincian CVE

Bagian detail CVE mencakup informasi berikut:

- Deskripsi dan ID CVE
- Keparahan CVE
- Skor Common Vulnerability Scoring System (CVSS) dan Exploit Prediction Scoring System (EPSS)
- Platform deteksi

> **Note**
>
> Jika bidang ini kosong, Amazon Inspector tidak mendukung deteksi untuk ID CVE Anda.

- Pencacahan Kelemahan Umum (CWE)
- Tanggal dibuat dan diperbarui vendor

Kecerdasan kerentanan

Bagian intelijen kerentanan menyediakan data intelijen ancaman seperti target eksploitasi dan tanggal eksploitasi publik terakhir yang diketahui.

Ini juga menyediakan data dari Cybersecurity and Infrastructure Security Agency (CISA), yang mencakup tindakan remediasi, tanggal CVE ditambahkan ke katalog Known Exploited Vulnerability, dan tanggal waktu CISA mengharapkan agen federal untuk memulihkan CVE.

Referensi

Bagian referensi menyediakan tautan ke sumber daya untuk informasi lebih lanjut tentang CVE.

Mengekspor SBOMs dengan Amazon Inspector

Software bill of materials (SBOM) adalah inventaris bersarang dari semua komponen perangkat lunak open-source dan pihak ketiga dalam basis kode Anda. Amazon Inspector menyediakan SBOMs sumber daya individual di lingkungan Anda. Anda dapat menggunakan konsol Amazon Inspector atau Amazon Inspector API untuk SBOMs menghasilkan sumber daya Anda. Anda dapat mengekspor SBOMs semua sumber daya yang didukung dan dipantau oleh Amazon Inspector. Diekspor SBOMs memberikan informasi tentang pasokan perangkat lunak Anda. Anda dapat meninjau status sumber daya Anda dengan menilai cakupan AWS lingkungan Anda. Bagian ini menjelaskan cara mengkonfigurasi dan mengekspor SBOMs.

Beberapa komponen perangkat lunak dan manajer paket menggunakan rentang versi atau referensi dinamis alih-alih versi tetap untuk dependensi. Praktik ini membuat hash yang belum terselesaikan, di mana Amazon Inspector mengidentifikasi file hash atau jar tetapi tidak dapat memetakannya ke nama dan versi tertentu untuk deteksi kerentanan. Amazon Inspector sekarang menyertakan hash yang belum terselesaikan ini dalam ekspor Software Bill of Materials (SBOM). Meskipun paket-paket ini tidak dapat dipindai untuk kerentanan, nilai hash mereka tersedia dalam daftar komponen yang diekspor.

> **Note**
>
> Saat ini, Amazon Inspector tidak mendukung ekspor untuk instans SBOMs Windows Amazon. EC2

Format Amazon Inspector

Amazon Inspector mendukung ekspor SBOMs dalam format yang kompatibel dengan CycloneDX 1.4 dan SPDX 2.3. Amazon Inspector mengekspor SBOMs sebagai `JSON` file ke bucket Amazon S3 yang Anda pilih.

> **Note**
>
> Ekspor format SPDX dari Amazon Inspector kompatibel dengan sistem yang menggunakan SPDX 2.3, namun tidak mengandung bidang Creative Commons Zero (CC0). Ini karena menyertakan bidang ini akan memungkinkan pengguna untuk mendistribusikan ulang atau mengedit materi.

Contoh format CycloneDX 1.4 SBOM dari Amazon Inspector


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

```
      "bom-ref": "2f4d199d4ef9e2ae639b4f8d04a813a2"
    },
    {
      "type": "application",
      "name": "liblz4-1",
      "purl": "pkg:dpkg/liblz4-1@1.8.3-1+deb10u1?
arch=ARM64&epoch=0&upstream=liblz4-1-1.8.3-1+deb10u1.src.dpkg",
      "bom-ref": "9a6be8907ead891b070e60f5a7b7aa9a"
    },
    {
      "type": "application",
      "name": "mawk",
      "purl": "pkg:dpkg/mawk@1.3.3-17+b3?
arch=ARM64&epoch=0&upstream=mawk-1.3.3-17+b3.src.dpkg",
      "bom-ref": "c2015852a729f97fde924e62a16f78a5"
    },
    {
      "type": "application",
      "name": "libgmp10",
      "purl": "pkg:dpkg/libgmp10@6.1.2+dfsg-4+deb10u1?
arch=ARM64&epoch=2&upstream=libgmp10-6.1.2+dfsg-4+deb10u1.src.dpkg",
      "bom-ref": "52907290f5beef00dff8da77901b1085"
    },
    {
      "type": "application",
      "name": "ncurses-bin",
      "purl": "pkg:dpkg/ncurses-bin@6.1+20181013-2+deb10u3?
arch=ARM64&epoch=0&upstream=ncurses-bin-6.1+20181013-2+deb10u3.src.dpkg",
      "bom-ref": "cd20cfb9ebeeadba3809764376f43bce"
    }
  ],
  "vulnerabilities": [
    {
      "id": "CVE-2022-40897",
      "affects": [
        {
          "ref": "a74a4862cc654a2520ec56da0c81cdb3"
        },
        {
          "ref": "0119eb286405d780dc437e7dbf2f9d9d"
        }
      ]
    }
  ]
```

```
}
```

Contoh format SPDX 2.3 SBOM dari 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"
}
```

Filter untuk SBOMs

Saat Anda mengekspor, SBOMs Anda dapat menyertakan filter untuk membuat laporan untuk subset sumber daya tertentu. Jika Anda tidak menyediakan filter SBOMs untuk semua sumber daya aktif yang didukung akan diekspor. Dan jika Anda adalah administrator yang didelegasikan, ini termasuk sumber daya untuk semua anggota juga. Filter berikut tersedia:

- AccountID — Filter ini dapat digunakan untuk SBOMs mengekspor sumber daya apa pun yang terkait dengan ID Akun tertentu.
- EC2 tag instance - Filter ini dapat digunakan SBOMs untuk mengekspor EC2 instance dengan tag tertentu.
- Nama fungsi - Filter ini dapat digunakan SBOMs untuk mengekspor fungsi Lambda tertentu.
- Tag gambar - Filter ini dapat digunakan SBOMs untuk mengekspor gambar kontainer dengan tag tertentu.
- Tag fungsi Lambda - Filter ini dapat digunakan untuk mengekspor fungsi SBOMs Lambda dengan tag tertentu.
- Jenis sumber daya - Filter ini dapat digunakan untuk memfilter jenis sumber daya: EC2 /ECR/ Lambda.
- ID Sumber Daya — Filter ini dapat digunakan untuk mengekspor SBOM untuk sumber daya tertentu.
- Nama repositori —Filter ini dapat digunakan SBOMs untuk menghasilkan gambar kontainer di repositori tertentu.

Konfigurasikan dan ekspor SBOMs

Untuk mengekspor SBOMs, Anda harus terlebih dahulu mengonfigurasi bucket Amazon S3 dan AWS KMS kunci yang diizinkan untuk digunakan oleh Amazon Inspector. Anda dapat menggunakan filter SBOMs untuk mengekspor subset tertentu dari sumber daya Anda. SBOMs Untuk mengekspor beberapa akun di AWS Organisasi, ikuti langkah-langkah ini saat masuk sebagai administrator yang didelegasikan Amazon Inspector.

Prasyarat

- Sumber daya yang didukung yang sedang dipantau secara aktif oleh Amazon Inspector.
- Bucket Amazon S3 yang dikonfigurasi dengan kebijakan yang memungkinkan Amazon Inspector menambahkan objek ke. Untuk informasi tentang mengonfigurasi kebijakan, lihat Mengonfigurasi izin ekspor.
- AWS KMS Kunci yang dikonfigurasi dengan kebijakan yang memungkinkan Amazon Inspector digunakan untuk mengenkripsi laporan Anda. Untuk informasi tentang mengonfigurasi kebijakan, lihat Mengonfigurasi AWS KMS kunci untuk ekspor.

> **Note**
>
> Jika sebelumnya Anda telah mengonfigurasi bucket Amazon S3 dan AWS KMS kunci untuk ekspor temuan, Anda dapat menggunakan bucket dan kunci yang sama untuk ekspor SBOM.

Pilih metode akses pilihan Anda untuk mengekspor SBOM.

Console

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Menggunakan AWS Region pemilih di sudut kanan atas halaman, pilih Wilayah dengan sumber daya yang ingin Anda ekspor SBOM.
3. Di panel navigasi, pilih Ekspor SBOMs.
4. (Opsional) Di SBOMs halaman Ekspor, gunakan menu Tambahkan filter untuk memilih subset sumber daya untuk membuat laporan. Jika tidak ada filter yang disediakan, Amazon Inspector akan mengekspor laporan untuk semua sumber daya aktif. Jika Anda adalah administrator yang didelegasikan, ini akan mencakup semua sumber daya aktif di organisasi Anda.
5. Di bawah Pengaturan Ekspor pilih format yang Anda inginkan untuk SBOM.
6. Masukkan URI Amazon S3 atau pilih Jelajahi Amazon S3 untuk memilih lokasi Amazon S3 untuk menyimpan SBOM.
7. Masukkan AWS KMS kunci yang dikonfigurasi untuk Amazon Inspector untuk digunakan untuk mengenkripsi laporan Anda.

API

- SBOMs Untuk mengekspor sumber daya Anda secara terprogram, gunakan CreateSbomExportpengoperasian Amazon Inspector API.

  Dalam permintaan Anda, gunakan `reportFormat` parameter untuk menentukan format output SBOM, pilih `CYCLONEDX_1_4` atau`SPDX_2_3`. `s3Destination`Parameter diperlukan dan Anda harus menentukan bucket S3 yang dikonfigurasi dengan kebijakan yang memungkinkan Amazon Inspector menulis ke sana. Secara opsional gunakan

`resourceFilterCriteria` parameter untuk membatasi ruang lingkup laporan ke sumber daya tertentu.

AWS CLI

- SBOMs Untuk mengekspor sumber daya Anda menggunakan AWS Command Line Interface jalankan perintah berikut:

  ```
  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
  ```

  Dalam permintaan Anda, ganti *FORMAT* dengan format pilihan Anda, `CYCLONEDX_1_4` atau`SPDX_2_3`. Kemudian ganti *user input placeholders* untuk tujuan s3 dengan nama bucket S3 untuk diekspor, awalan yang akan digunakan untuk output di S3, dan ARN untuk kunci KMS yang Anda gunakan untuk mengenkripsi laporan.

Skema EventBridge acara Amazon untuk acara Amazon Inspector

Amazon EventBridge memberikan aliran data real-time dari aplikasi dan lainnya Layanan AWS ke target, seperti AWS Lambda fungsi, topik Layanan Pemberitahuan Sederhana Amazon, dan aliran data di Amazon Kinesis Data Streams. Untuk mendukung integrasi dengan aplikasi, layanan, dan sistem lain, Amazon Inspector secara otomatis menerbitkan temuan sebagai peristiwa. EventBridge Anda dapat menggunakan Amazon Inspector untuk mempublikasikan acara untuk temuan, cakupan, dan pemindaian. Bagian ini memberikan contoh skema untuk EventBridge acara.

Topik

Skema EventBridge dasar Amazon untuk Amazon Inspector

Berikut ini adalah contoh skema dasar untuk EventBridge acara Amazon Inspector. Detail acara berbeda berdasarkan jenis acara.

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

```
}
```

Amazon Inspector menemukan contoh skema acara

Berikut ini mencakup contoh skema untuk EventBridge acara untuk temuan Amazon Inspector. Menemukan peristiwa dibuat saat Amazon Inspector mengidentifikasi kerentanan perangkat lunak atau masalah jaringan di salah satu sumber daya Anda. Untuk panduan membuat notifikasi sebagai respons terhadap jenis acara ini, lihatMembuat tanggapan khusus terhadap temuan Amazon Inspector dengan Amazon EventBridge.

Bidang berikut mengidentifikasi peristiwa temuan:

- `detail-type`diatur ke`Inspector2 Finding`.
- `detail`menjelaskan temuan tersebut.
- `detail.resources.tags`adalah tempat data nilai kunci disimpan.

Anda dapat memfilter tab untuk melihat skema pencarian acara untuk sumber daya yang berbeda dan jenis pencarian.

Amazon EC2 package vulnerability finding

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

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

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

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

Amazon EC2 network reachability finding

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

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

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

Amazon ECR package vulnerability finding

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

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

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

Lambda package vulnerability finding

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

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

```
                    "version": "3.1"
                }
            ],
            "referenceUrls": [
                "https://www.debian.org/security/2023/dsa-5442",
                "https://lists.debian.org/debian-lts-announce/2023/08/msg00024.html"
            ],
            "relatedVulnerabilities": [],
            "source": "NVD",
            "sourceUrl": "https://nvd.nist.gov/vuln/detail/CVE-2023-30861",
            "vendorCreatedAt": "Tue May 02 18:15:52.000 UTC 2023",
            "vendorSeverity": "HIGH",
            "vendorUpdatedAt": "Sun Aug 20 21:15:09.000 UTC 2023",
            "vulnerabilityId": "CVE-2023-30861",
            "vulnerablePackages": [
                {
                    "epoch": 0,
                    "filePath": "requirements.txt",
                    "fixedInVersion": "2.3.2",
                    "name": "flask",
                    "packageManager": "PIP",
                    "version": "2.0.0"
                }
            ]
        },
        "remediation": {
            "recommendation": {
                "text": "None Provided"
            }
        },
        "resources": [
            {
                "details": {
                    "awsLambdaFunction": {
                        "architectures": [
                            "X86_64"
                        ],
                        "codeSha256": "O7jkFEmfPB+CK3Y6Pby5zW9gjG
+zusAaqRRMGS8B27c=",
                        "executionRoleArn": "arn:aws:iam::123456789012:role/service-
role/VulnerableFunction-role-f9vs5mq8",
                        "functionName": "VulnerableFunction",
                        "lastModifiedAt": "Wed Sep 04 16:50:20.000 UTC 2024",
                        "packageType": "ZIP",
```

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

Lambda code vulnerability finding

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

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

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


> **Note**
>
> Nilai detail mengembalikan rincian JSON dari temuan tunggal sebagai objek. Itu tidak mengembalikan seluruh sintaks respons temuan, yang mendukung beberapa temuan dalam array.

Contoh skema acara lengkap pemindaian awal Amazon Inspector

Berikut ini adalah contoh skema EventBridge acara untuk acara Amazon Inspector untuk menyelesaikan pemindaian awal. Acara ini dibuat saat Amazon Inspector menyelesaikan pemindaian awal salah satu sumber daya Anda.

Bidang berikut mengidentifikasi peristiwa lengkap pemindaian awal:

- Bidang `detail-type` diatur ke `Inspector2 Scan`.
- `detail`Objek berisi `finding-severity-counts` objek yang merinci jumlah temuan dalam kategori keparahan yang berlaku, seperti`CRITICAL`,`HIGH`, dan`MEDIUM`.

Pilih dari opsi untuk melihat skema peristiwa pemindaian awal yang berbeda menurut jenis sumber daya.

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"
  }
}
```

Contoh skema acara cakupan Amazon Inspector

Berikut ini adalah contoh skema EventBridge acara untuk acara Amazon Inspector untuk liputan. Acara ini dibuat saat cakupan pemindaian Amazon Inspector untuk sumber daya diubah. Bidang berikut mengidentifikasi peristiwa cakupan:

- Bidang `detail-type` diatur ke `Inspector2 Coverage`.
- `detail`Objek berisi `scanStatus` objek yang menunjukkan status pemindaian baru untuk sumber daya.

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

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

Amazon Inspector auto mengaktifkan contoh skema

Acara aktifkan otomatis dikirim ke admin yang didelegasikan saat Amazon Inspector tidak dapat mendukung jumlah anggota dalam suatu organisasi. Bidang berikut mengidentifikasi peristiwa aktifkan otomatis:

- Bidang `detail-type` diatur ke `Inspector2 AutoEnable`.
- `detail`Objek menjelaskan mengapa peristiwa auto enable gagal.

```
{
    "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"
        }
}
```

Plugin Amazon Inspector SSM untuk Linux and Windows

Topik ini menjelaskan plugin Amazon Inspector SSM untuk Linux and Windows contoh.

Plugin Amazon Inspector SSM untuk Linux

Amazon Inspector menggunakan plugin Amazon Inspector SSM untuk melakukan pemindaian inspeksi mendalam pada instance Linux. Plugin Amazon Inspector SSM secara otomatis diinstal pada instance Linux di direktori. `/opt/aws/inspector/bin` Nama executable adalah. `inspectorssmplugin`

Amazon Inspector menggunakan Systems Manager Distributor untuk menyebarkan plugin pada instans Anda. Untuk melakukan pemindaian inspeksi mendalam, Systems Manager Distributor dan Amazon Inspector harus mendukung sistem operasi instans EC2 Amazon Anda. Untuk informasi tentang sistem operasi yang didukung oleh Distributor Systems Manager, lihat Platform dan arsitektur paket yang didukung di Panduan AWS Systems Manager Pengguna.

Amazon Inspector membuat direktori file untuk mengelola data yang dikumpulkan untuk pemeriksaan mendalam oleh plugin Amazon Inspector SSM. Direktori file ini termasuk `/opt/aws/inspector/var/input` dan`/opt/aws/inspector/var/output`.

`packages.txt`File di `/opt/aws/inspector/var/output` menyimpan jalur lengkap ke paket yang ditemukan oleh inspeksi mendalam. Jika Amazon Inspector mendeteksi paket yang sama beberapa kali pada instance Anda, `packages.txt` file tersebut akan mencantumkan setiap lokasi tempat paket ditemukan.

Amazon Inspector menyimpan log untuk plugin di direktori. `/var/log/amazon/inspector`

Menghapus instalasi plugin Amazon Inspector SSM

Jika `inspectorssmplugin` file dihapus secara tidak sengaja, asosiasi SSM `InspectorLinuxDistributor-do-not-delete` akan mencoba menginstal ulang `inspectorssmplugin` file pada interval pemindaian berikutnya.

Jika Anda menonaktifkan EC2 pemindaian Amazon, plugin akan dihapus secara otomatis dari semua host Linux.

Plugin Amazon Inspector SSM untuk Windows

Plugin Amazon Inspector SSM diperlukan untuk Amazon Inspector untuk memindai Windows contoh. Plugin Amazon Inspector SSM secara otomatis diinstal pada Anda Windows contoh di`C:\Program Files\Amazon\Inspector`, dan file biner yang dapat dieksekusi diberi nama. `InspectorSsmPlugin.exe`

Lokasi file berikut dibuat untuk menyimpan data yang dikumpulkan oleh plugin Amazon Inspector SSM:

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

> **Note**
>
> Secara default, plugin Amazon Inspector SSM berjalan di bawah prioritas normal.

> **Note**
>
> Anda dapat menggunakan Windows instance dengan pengaturan Konfigurasi Manajemen Host Default. Namun, Anda harus membuat atau menggunakan peran yang dikonfigurasi dengan `ssm:GetParameter` izin `ssm:PutInventory` dan.

Menghapus instalasi plugin Amazon Inspector SSM

Jika `InspectorSsmPlugin.exe` file dihapus secara tidak sengaja, `InspectorDistributor-do-not-delete` asosiasi akan menginstal ulang file di berikutnya `InspectorSsmPlugin.exe` Windows interval pemindaian. Jika Anda ingin menghapus plugin Amazon Inspector SSM, Anda dapat menggunakan tindakan Uninstall dalam dokumen. `AmazonInspector2-ConfigureInspectorSsmPlugin` Namun, plugin Amazon Inspector SSM akan dihapus secara otomatis dari semua Windows host jika Anda menonaktifkan EC2 pemindaian Amazon.

> **Note**
>
> Jika Anda menghapus instalan Agen SSM sebelum menonaktifkan Amazon Inspector, plugin Amazon Inspector SSM akan tetap ada di Windows host, tetapi tidak akan mengirim data ke plugin Amazon Inspector SSM. Lihat informasi yang lebih lengkap di Menonaktifkan Amazon Inspector.

Amazon Inspector SBOM Generator

Software Bill of Materials (SBOM) adalah daftar komponen, pustaka, dan modul yang terstruktur secara formal yang diperlukan untuk membangun perangkat lunak. Amazon Inspector SBOM Generator (Sbomgen) adalah alat yang menghasilkan SBOM untuk arsip, gambar kontainer, direktori, sistem lokal, dan kompilasi dan binari. Go Rust Sbomgenmemindai file yang berisi informasi tentang paket yang diinstal. Ketika Sbomgen menemukan file yang relevan, ia mengekstrak nama paket, versi, dan metadata lainnya. Sbomgenkemudian mengubah metadata paket menjadi SBOM. CycloneDX Anda dapat menggunakan Sbomgen untuk menghasilkan CycloneDX SBOM sebagai file atau di STDOUT dan mengirim ke Amazon SBOMs Inspector untuk deteksi kerentanan. Anda juga dapat menggunakan Sbomgen sebagai bagian dari CI/CD integrasi, yang memindai gambar kontainer secara otomatis sebagai bagian dari pipeline penerapan Anda.

Jenis paket yang didukung

Sbomgenmengumpulkan inventaris untuk jenis paket berikut:

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

Pemeriksaan konfigurasi gambar kontainer yang didukung

Sbomgendapat memindai Dockerfiles mandiri dan membangun riwayat dari gambar yang ada untuk masalah keamanan. Untuk informasi selengkapnya, lihat pemeriksaan Amazon Inspector Dockerfile.

Menginstal Sbomgen

Sbomgenhanya tersedia untuk sistem operasi Linux.

Anda harus Docker menginstal jika Anda Sbomgen ingin menganalisis gambar yang di-cache secara lokal. Dockertidak diperlukan untuk menganalisis gambar yang diekspor sebagai `.tar` file atau gambar yang dihosting di pendaftar kontainer jarak jauh.

Amazon Inspector merekomendasikan agar Anda menjalankan Sbomgen dari sistem dengan setidaknya spesifikasi perangkat keras berikut:

- CPU inti 4x
- 8 GB RAM

Untuk menginstal Sbomgen

1. Unduh file Sbomgen zip terbaru dari URL yang benar untuk arsitektur Anda:

   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

   Atau, Anda dapat men-download versi sebelumnya dari Amazon Inspector SBOM Generator file zip.
2. Buka zip unduhan menggunakan perintah berikut:

   ```
   unzip inspector-sbomgen.zip
   ```
3. Periksa file-file berikut di direktori yang diekstrak:

   - `inspector-sbomgen`— Ini adalah alat yang akan Anda jalankan untuk menghasilkan SBOMs.
   - `README.txt`- Ini adalah dokumentasi untuk digunakanSbomgen.
   - `LICENSE.txt`— File ini berisi lisensi perangkat lunak untukSbomgen.
   - `licenses`— Folder ini berisi info lisensi untuk paket pihak ketiga yang digunakan olehSbomgen.
   - `checksums.txt`— File ini menyediakan hash Sbomgen alat.

- `sbom.json`— Ini adalah CycloneDX SBOM untuk Sbomgen alat ini.
- `WhatsNew.txt`— File ini berisi log perubahan yang dirangkum, sehingga Anda dapat melihat perubahan besar dan peningkatan antar Sbomgen versi dengan cepat.

4. (Opsional) Verifikasi keaslian dan integritas alat menggunakan perintah berikut:

   ```
   sha256sum < inspector-sbomgen
   ```

   - Bandingkan hasilnya dengan isi `checksums.txt` file.

5. Berikan izin yang dapat dieksekusi ke alat menggunakan perintah berikut:

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

6. Verifikasi bahwa Sbomgen berhasil diinstal menggunakan perintah berikut:

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

   Anda akan melihat output yang mirip dengan yang berikut ini:

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

Menggunakan Sbomgen

Bagian ini menjelaskan berbagai cara yang dapat Anda gunakanSbomgen. Anda dapat mempelajari lebih lanjut tentang cara menggunakan Sbomgen melalui contoh bawaan. Untuk melihat contoh ini, jalankan `list-examples` perintah:

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

Hasilkan SBOM untuk gambar kontainer dan output hasilnya

Anda dapat menggunakan Sbomgen SBOMs untuk menghasilkan gambar kontainer dan menampilkan hasilnya ke file. Kemampuan ini dapat diaktifkan menggunakan `container` subperintah.

Perintah contoh

Dalam cuplikan berikut, Anda dapat mengganti *`image:tag`* dengan ID gambar Anda dan *`output_path.json`* dengan jalur ke output yang ingin Anda simpan.

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

> **Note**
>
> Waktu dan kinerja pemindaian tergantung pada ukuran gambar dan seberapa kecil jumlah lapisannya. Gambar yang lebih kecil tidak hanya meningkatkan Sbomgen kinerja, tetapi juga mengurangi potensi permukaan serangan. Gambar yang lebih kecil juga meningkatkan waktu pembuatan, unduhan, dan unggah gambar.

Saat menggunakan Sbomgen with `ScanSbom`, Amazon Inspector Scan API tidak akan memproses SBOMs yang berisi lebih dari 5.000 paket. Dalam skenario ini, Amazon Inspector Scan API mengembalikan respons HTTP 400.

Jika gambar menyertakan file media massal atau direktori, pertimbangkan untuk mengecualikan mereka dari Sbomgen menggunakan argumen. `--skip-files`

Contoh: Kasus kesalahan umum

Pemindaian gambar kontainer dapat gagal karena kesalahan berikut:

- `InvalidImageFormat`— Terjadi saat memindai gambar kontainer yang salah bentuk dengan header TAR yang rusak, file manifes, atau file konfigurasi.
- `ImageValidationFailure`— Terjadi ketika validasi checksum atau panjang konten gagal untuk komponen gambar kontainer, seperti header Panjang Konten yang tidak cocok, intisari manifes yang salah, atau verifikasi checksum yang gagal. SHA256
- `ErrUnsupportedMediaType`— Terjadi ketika komponen gambar menyertakan jenis media yang tidak didukung. Untuk informasi tentang jenis media yang didukung, lihat Sistem operasi dan jenis media yang didukung.

Amazon Inspector tidak mendukung jenis `application/vnd.docker.distribution.manifest.list.v2+json` media. Namun, Amazon Inspector mendukung daftar manifes. Saat memindai gambar yang menggunakan daftar manifes, Anda dapat secara eksplisit menentukan platform mana yang akan digunakan dengan argumen tersebut`--platform`. Jika `--platform` argumen tidak ditentukan, Amazon Inspector SBOM Generator secara otomatis memilih manifes berdasarkan platform tempat perjalanannya.

Menghasilkan SBOM dari direktori dan arsip

Anda dapat menggunakan Sbomgen untuk menghasilkan SBOMs dari direktori dan arsip. Kemampuan ini dapat diaktifkan menggunakan `directory` atau `archive` subperintah. Amazon Inspector merekomendasikan penggunaan fitur ini ketika Anda ingin membuat SBOM dari folder proyek, seperti repositori git yang diunduh.

Contoh perintah 1

Cuplikan berikut menunjukkan subperintah yang menghasilkan SBOM dari file direktori.

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

Contoh perintah 2

Cuplikan berikut menunjukkan subperintah yang menghasilkan SBOM dari file arsip. Satu-satunya format arsip yang didukung adalah `.zip`, `.tar`, dan `.tar.gz`.

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

Hasilkan SBOM dari Go atau binari yang Rust dikompilasi

Anda dapat menggunakan Sbomgen untuk menghasilkan SBOMs dari kompilasi Go dan Rust binari. Anda dapat mengaktifkan cabapility ini melalui subperintah: `binary`

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

Hasilkan SBOM dari volume yang dipasang

Anda dapat menggunakan Amazon Inspector SBOM Generator untuk menghasilkan SBOMs dari volume yang dipasang. Kemampuan ini dapat diaktifkan menggunakan `volume` subperintah. Sebaiknya gunakan fitur ini saat Anda ingin menganalisis volume penyimpanan, seperti volume Amazon EBS yang telah dipasang ke sistem Anda. Berbeda dengan subperintah direktori, pemindaian volume yang dipasang mendeteksi paket OS dan informasi OS.

Anda dapat memindai volume Amazon EBS dengan melampirkannya ke instans Amazon tempat Amazon Inspector SBOM Generator diinstal dan memasangnya pada EC2 instance itu. Untuk volume

Amazon EBS yang saat ini digunakan oleh EC2 instans Amazon lainnya, Anda dapat membuat snapshot Amazon EBS dari volume tersebut dan kemudian membuat volume Amazon EBS baru dari snapshot tersebut untuk tujuan pemindaian. Untuk informasi selengkapnya tentang Amazon EBS, lihat Apa itu Amazon EBS? di Panduan Pengguna Amazon Elastic Block Store.

Perintah contoh

Cuplikan berikut menunjukkan subperintah yang menghasilkan SBOM dari volume yang dipasang. `--path`Argumen harus menentukan direktori root tempat volume dipasang.

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

Perintah contoh

Cuplikan berikut menunjukkan subperintah yang menghasilkan SBOM dari volume yang dipasang sementara mengecualikan jalur file tertentu dengan argumen. `--exclude-suffix` `--exclude-suffix`Argumen ini sangat berguna ketika volume berisi file massal (seperti file log atau file media). File dan direktori yang jalurnya diakhiri dengan sufiks yang ditentukan akan dikecualikan dari pemindaian, yang dapat mengurangi waktu pemindaian dan penggunaan memori.

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

Semua jalur file dalam volume target dinormalisasi ke jalur aslinya. Misalnya, saat memindai volume yang dipasang `/mnt/volume` yang berisi file di`/mnt/volume/var/lib/rpm/rpmdb.sqlite`, jalur akan dinormalisasi ke `/var/lib/rpm/rpmdb.sqlite` dalam SBOM yang dihasilkan.

Kirim SBOM ke Amazon Inspector untuk identifikasi kerentanan

Selain menghasilkan SBOM, Anda dapat mengirim SBOM untuk pemindaian dengan satu perintah dari Amazon Inspector Scan API. Amazon Inspector mengevaluasi konten SBOM untuk kerentanan sebelum mengembalikan temuan ke. Sbomgen Tergantung pada masukan Anda, temuan dapat ditampilkan atau ditulis ke file.

> **Note**
>
> Anda harus memiliki izin baca aktif Akun AWS `InspectorScan-ScanSbom` untuk menggunakan kemampuan ini.

Untuk mengaktifkan kemampuan ini, Anda meneruskan `--scan-sbom` argumen ke Sbomgen CLI. Anda juga dapat meneruskan `--scan-sbom` argumen ke salah satu Sbomgen subperintah berikut:`archive`,,`binary`, `containerdirectory`,`localhost`.

> **Note**
>
> Amazon Inspector Scan API tidak memproses SBOMs lebih dari 5.000 paket. Dalam skenario ini, Amazon Inspector Scan API mengembalikan respons HTTP 400.

Anda dapat melakukan autentikasi ke Amazon Inspector melalui AWS profil atau peran IAM dengan argumen berikut: AWS CLI

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

Anda juga dapat mengautentikasi ke Amazon Inspector dengan menyediakan variabel lingkungan berikut. Sbomgen

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

Untuk menentukan format respons, gunakan `--scan-sbom-output-format cyclonedx` argumen atau `--scan-sbom-output-format inspector` argumen.

Contoh perintah 1

Perintah ini membuat SBOM untuk Alpine Linux rilis terbaru, memindai SBOM, dan menulis hasil kerentanan ke 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
```

Contoh perintah 2

Perintah ini mengautentikasi ke Amazon Inspector AWS menggunakan kredensyal sebagai variabel lingkungan.

```
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
```

Contoh perintah 3

Perintah ini mengautentikasi ke Amazon Inspector menggunakan ARN untuk peran IAM.

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

Gunakan pemindai tambahan untuk meningkatkan kemampuan deteksi

Amazon Inspector SBOM Generator menerapkan pemindai yang telah ditentukan berdasarkan perintah yang digunakan.

Grup pemindai default

Setiap subperintah Amazon Inspector SBOM Generator menerapkan grup pemindai default berikut secara otomatis.

- Untuk `directory` subperintah: biner, programming-language-packages, grup pemindai dockerfile
- Untuk `localhost` subperintah: os, programming-language-packages, grup pemindai ekosistem ekstra
- Untuk `container` subperintah: os,, extra-ecosystem programming-language-packages, dockerfile, grup pemindai biner

Pemindai khusus

Untuk menyertakan pemindai di luar grup pemindai default, gunakan `--additional-scanners` opsi diikuti dengan nama pemindai yang akan ditambahkan. Berikut ini adalah contoh perintah yang menunjukkan bagaimana melakukan ini.

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

Berikut ini adalah contoh perintah yang menunjukkan cara menambahkan beberapa pemindai dengan daftar yang dipisahkan koma.

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

Optimalkan pemindaian kontainer dengan menyesuaikan ukuran file maksimum untuk memindai

Saat Anda menganalisis dan memproses gambar kontainer, Sbomgen memindai file yang berukuran 200 MB atau kurang secara default. File yang lebih besar dari 200 MB jarang berisi paket metadata. Anda dapat menemukan kesalahan ketika Anda inventaris Go atau Rust biner yang melebihi 200MB. Untuk menyesuaikan batas ukuran, gunakan `--max-file-size` argumen. Ini memungkinkan Anda untuk meningkatkan batas untuk menyertakan file besar dan mengurangi batas untuk mengurangi penggunaan sumber daya dengan mengecualikan file besar.

Contoh

Contoh berikut menunjukkan bagaimana menggunakan `--max-file-size` argumen untuk meningkatkan ukuran 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
```

Menyesuaikan pengaturan ini membantu mengontrol penggunaan disk, konsumsi memori, dan durasi pemindaian keseluruhan.

Nonaktifkan indikator kemajuan

Sbomgenmenampilkan indikator kemajuan berputar yang dapat menghasilkan karakter garis miring yang berlebihan di CI/CD lingkungan.

```
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
```

Anda dapat menonaktifkan indikator kemajuan menggunakan `--disable-progress-bar` argumen:

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

Mengautentikasi ke pendaftar pribadi dengan Sbomgen

Dengan memberikan kredensi otentikasi registri pribadi Anda, Anda dapat menghasilkan SBOMs dari kontainer yang di-host di pendaftar pribadi. Anda dapat memberikan kredensi ini melalui metode berikut:

Otentikasi menggunakan kredensi cache (disarankan)

Untuk metode ini, Anda mengautentikasi ke registri kontainer Anda. Misalnya, jika menggunakanDocker, Anda dapat mengautentikasi ke registri kontainer Anda menggunakan perintah Docker loging:. `docker login`

1. Otentikasi ke registri kontainer Anda. Misalnya, jika menggunakanDocker, Anda dapat mengautentikasi ke registri Anda menggunakan Docker `login` perintah:
2. Setelah Anda mengautentikasi ke registri kontainer Anda, gunakan Sbomgen pada gambar kontainer yang ada di registri. Untuk menggunakan contoh berikut, ganti *`image:tag`* dengan nama gambar yang akan dipindai:

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

Otentikasi menggunakan metode interaktif

Untuk metode ini, berikan nama pengguna Anda sebagai parameter, dan Sbomgen akan meminta Anda untuk entri kata sandi yang aman bila diperlukan.

Untuk menggunakan contoh berikut, ganti *`image:tag`* dengan nama gambar yang ingin Anda pindai dan *`your_username`* dengan nama pengguna yang memiliki akses ke gambar:

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

Otentikasi menggunakan metode non-interaktif

Untuk metode ini, simpan kata sandi atau token registri Anda dalam `.txt` file.

> **Note**
>
> Pengguna saat ini seharusnya hanya dapat membaca file ini. File juga harus berisi kata sandi atau token Anda pada satu baris.

Untuk menggunakan contoh berikut, ganti *`your_username`* dengan nama pengguna Anda, *`password.txt`* dengan `.txt` file yang menyertakan kata sandi atau token Anda pada satu baris, dan *`image:tag`* dengan nama gambar untuk dipindai:

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

Contoh output dari Sbomgen

Berikut ini adalah contoh SBOM untuk gambar kontainer yang diinventarisasi menggunakan. Sbomgen

Gambar kontainer 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"
        }
      ]
    }
  ]
}
```

Versi sebelumnya dari Amazon Inspector SBOM Generator

Topik ini mencakup tautan ke versi terbaru dan sebelumnya dari Amazon Inspector SBOM Generator. Untuk informasi tentang menginstalSbomgen, lihat Menginstal Sbomgen.

Platform	Versi	SHA-256 checksum	
Linux AMD64	1.11.2	bef68671bc532e4fb5 29500b62d7af836012 3cd967308d41ad0ce8 f43f7762fb	
Linux ARM64	1.11.2		

Platform	Versi	SHA-256 checksum
		4f11d7037efa443f44 2c4edf7ba28774c4fa 706fb7622e4fba645b b3ad3958c9
Linux AMD64 Linux ARM64	1.11.1 1.11.1	809eb7cb80d24fbf6f fdd124438d53a90763 2c222e924913ebd610 44ca949490 057f9e4c9970aeda4b da0685e7e02436fd52 23fbe81cec65138551 c63ed77ba0
Linux AMD64 Linux ARM64	1.11.0 1.11.0	5172a5556cf46f9fbc 5cf1d35bd382919fb6 b41aca1ec938db3a75 530060b0cf c9e2da7b076dc89dc3 9a962a7dd9c7d1fd29 230a4eec7eb95f951d 6a179093d0
Linux AMD64 Linux ARM64	1.10.1 1.10.1	9e33622a7874adfe71 9ab7db75a1e44f4b5f ae3573374068b501c8 9f0accfcfe 78d5a7f800fc26ba86 adab5b634431a91c00 7075e06d6ce46e5068 7d5156184e

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.10.0 1.10.0	0b7a553d7d2d17c40a62f1a11013bc46fa2c3814f407c11130e15af3fe313769 5ce9e315a4f8f90ff5eed7ab058efc8dbff6593d66d3fc455f1c37e882ec6466
Linux AMD64 Linux ARM64	1.9.1 1.9.1	d0ef4c14fec6c42e70ae55b3e44d17d027132947596e8ef861c0efc3c0e5a871 2d8145011c13f5611fc30f4510785d53e98b911717f6dbe69616af4d4b0df61f
Linux AMD64 Linux ARM64	1.9.0 1.9.0	78b377b2730eb15476173e40885454ae191e953663af3e0928dddfb8608f4655 985bdc06d25eccb87c4a81995c8a2d3c78e1c02beea309a620b2de4954767591

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.8.3 1.8.3	54eed5a772f68320f3906bec5920e3a19da904abdace10f985b87859015eef89 febd74a397fb0cdd3356072503f08465ab872d1620d59a2ab7d83bdb076c929d
Linux AMD64 Linux ARM64	1.8.2 1.8.2	2e4e3c754e230046349dd975feb48fa953ea5a2de190cbbc17c1c85043936b5a 449a49e222a2bdffe0353435d7b04b0556b35a391c7b9714ce46d1a5382bc3e2
Linux AMD64 Linux ARM64	1.8.1 1.8.1	9ff7958e298d2b228b0c7617f0a9a873254587fc26aee9826c37273650b389e9 6737584fd2c7d24b56777d02846d1737f47d0121344baea217a3e5368fd98fcc

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.8.0 1.8.0	ef32e7fb4ee0af1e47d6b528b47293fc7127c7a7539f7354e84452626a4c204d 0b82ddc691a517bb8fc6ccd67b80ca566b117a1bb410c05764c9b7e3ba76c510
Linux AMD64 Linux ARM64	1.7.3 1.7.3	3fba95d44aaea55ad06d3c7635a671662c483474578376d3f11e84474f8de25f 1f4b52e3d80de87b92b563a78bac4a2d898e7af82db5b6791d899d516e97cfbb
Linux AMD64 Linux ARM64	1.7.2 1.7.2	c44ba9bf1cfeb3ea2d6d0b15d25816800a5045a438474f2f77c390bac41ae4cb d37c5b1605bf82260da0b0f36311c83b1646a4327c3fd8169ba4b3a978470c9c

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.7.1 1.7.1	b0beb602a6ae439d4e307bd99682bc8a419fd7d5e78a278bfc718eb18e00b05e 95ff2d9df2fcd1982dd705df1e763f57a0b499b6fe06801e9a80869e2e464831
Linux AMD64 Linux ARM64	1.7.0 1.7.0	a6316c2ecd5fde7091d1099335f45f0e2400b3977c92ee4d72bd1eb359320e61 9751ba5e5c6c0aef7d29b1c4adbd4088da3a07bb77eaa7de3f04aa33ad8562
Linux AMD64 Linux ARM64	1.6.3 1.6.3	b6a309e879aaa78d7d8e224eb5214df5fd415244d370885e6c8876db5a4181d2 59ed0b7eb7d1eadadb691f058d32634a03a856ba03ac2ddb8cd3599ceb55cb9a

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.6.2 1.6.2	8d8ba0653 5be614a4d44b1bd74c 66d1fd4874ff9ab788 ad5e23aa5229db9c68 7 2bd7b4a88b9c6b041a 6ff82f7f9bc116b76c f410bf6eb896fc8d68 e717b55f2a
Linux AMD64 Linux ARM64	1.6.1 1.6.1	3e3d62dc794b31d9d2 de1904592cf42f25e9 f42c30eb90cc53385a 60b42f1a63 ad89f670908fb0b48b ca0242f3ac58e7179f 6fabfcc9a2b3fd0e5c 3d79e27539
Linux AMD64 Linux ARM64	1.6.0 1.6.0	ffe671c2c1d1c2142a 4af056d1c179eaffbc 3925f5afaaa6f3d655 bd495ce5e1c a733c0b00c7225369c 68ad47c57846b4546e 2c9f47580ab98394ba efc765c134

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.5.5 1.5.5	ebcfbe565631de5bc61b1d55d70a2d15b965f628678a2b60cffd01cd0c3443f1 a8e018ceee3a76dd4271f966015c216438b11ee807fcd970753e786baa335b56
Linux AMD64 Linux ARM64	1.5.4 1.5.4	aa8c1ffacc563b87975497f53eddec0b29397a898fac19f4902b8acb7eeb347b c6ba98d441aa88d3d3150449c098cd13ce3baeccee45ad4c9a1326f8bb8f87fc
Linux AMD64 Linux ARM64	1.5.3 1.5.3	d493c23121101c9c3df888e717bf81d7f7b81809754f3492e1ae52f02b089b68 8dfa5c97b3bd45da487706e95d1894290f53b113247bbb89b9fac16dab8184b6

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.5.2 1.5.2	ff6233d7da9f7e963589a0eb8f07bee2ca375360365cb6b6e354585cf1371910 fd31efb6031754b2bc8414d7fe9dd14a067767704145af0559b3500cc437c7ee
Linux AMD64 Linux ARM64	1.5.1 1.5.1	391fcc52117fed79cae6e92a9e225732166a6df2582aa7f6b5230149761f6732 f9bc90d18724f93db0f5ca3b79136adb7b49fa33fa179a5e87b4d512f256b56b
Linux AMD64 Linux ARM64	1.5.0 1.5.0	d7b6cb84053358e462d76488d019140ecd05ad405217a60a96b727fb062880fe 067dcf5c302160a5270f89aed3f941bb0571dcb8a59f75dddb1b7747c2a82ec7

Platform	Versi	SHA-256 checksum	
Linux AMD64 Linux ARM64	1.4.0 1.4.0	c8ca73761afd742e1deb98b04eb5714c9c2a574b652a763b18e23560e66aea24 188d9757782278653e65605aaf186feda104345ba2f9de438873e568f1ff6204	
Linux AMD64 Linux ARM64	1.3.2 1.3.2	57dd5d135600e84690706cfe95860e78149988d37cf81429ce97b9256d179fb4 91526ecdafc6cc3718fabe75b2693ace5effb9c0af3327b484b7f5a154929997	
Linux AMD64 Linux ARM64	1.3.1 1.3.1	097ec83907c459a36de11c92d016fffd64f1c33fd4bcbf2af465e0979b0d9237 aa93a3d402abc4a986a9ad9d3de8fcca81ee25a55596ac6dc4502ed1d6819502	

Platform	Versi	SHA-256 checksum
Linux AMD64 Linux ARM64	1.3.0 1.3.0	21439f92c314daf136832ca6676a65d288768aa69fc6dcd2014a3038b2701eeb 4a41779b0c3b32242eedef288de6c1bf40fda0d4246b32fd0cd8d4e51e58f94b
Linux AMD64 Linux ARM64	1.2.1 1.2.1	e022e95e59f1790949bca8dbbb6478a5d3fb677ccd45aa4ba30ebd91ae86ad65 824acc5bb5b0210954fe9ab089d9461453a4975d34292cc0c676837c3a7279b4
Linux AMD64 Linux ARM64	1.2.0 1.2.0	9625b1a8ae1937ca2179c2535a0ffceca934138e0b66feac9ba3e34ffaa22ec5 7f387e560b41571fb52efd9e620bf2b9e3a067ca781e88aaa977b2b8acdebf35

Platform	Versi	SHA-256 checksum	
Linux AMD64 Linux ARM64	1.1.1 1.1.1	6809b7e46675c66e3af354c53433dc46c4d1ddaf258e05ba15e38e784ea0285e 6361e59fb2448c66c4698ea33979ecaaaefc2af4420034aabbbe741242f60dbdd	
Linux AMD64 Linux ARM64	1.1.0 1.1.0	f84c8815413d451490b38509950235f88713c0c61c7259a4831934995664bd8f aaffefb5e44195dc55d5fd3289e511720f64c130644cbd58103cf7f36e96f058	
Linux AMD64 Linux ARM64	1.0.0 1.0.0	cc126e24962f1a6497cf17679b3e3b73be68963c47e3968a56e73caacf045b5c 5d5bf97a4acfeaaa73ad6c918738188e0c822e475ef37a334e49d77ba907b08a	

Koleksi sistem operasi komprehensif Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator memindai sistem operasi yang berbeda untuk menjamin analisis komponen sistem yang kuat dan terperinci. Menghasilkan SBOM membantu Anda memahami

komposisi sistem operasi Anda, sehingga Anda dapat mengidentifikasi kerentanan dalam paket yang dikelola sistem. Topik ini menjelaskan fitur utama dari koleksi paket sistem operasi yang berbeda yang didukung Amazon Inspector SBOM Generator. Untuk informasi tentang sistem operasi yang didukung Amazon Inspector, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.

Artefak sistem operasi yang didukung

Amazon Inspector SBOM Generator mendukung artefak sistem operasi berikut:

Platform	Biner	Sumber	Streaming
Alma Linux	N/A	Ya	Ya
Alpine Linux	Ya	Ya	N/A
Amazon Linux	N/A	Ya	N/A
CentOS	N/A	Ya	N/A
Chainguard	Ya	Ya	N/A
Debian	Ya	Ya	N/A
Distroless	Ya	Ya	N/A
Fedora	N/A	Ya	N/A
MinimOS	Ya	Ya	N/A
OpenSUSE	N/A	Ya	N/A
Oracle Linux	N/A	Ya	N/A
Photon OS	N/A	Ya	N/A
RHEL	N/A	Ya	Ya
Rocky Linux	N/A	Ya	Ya
SLES	N/A	Ya	N/A

Platform	Biner	Sumber	Streaming
Ubuntu	Ya	Ya	N/A
Windows	N/A	N/A	N/A

Koleksi paket OS berbasis APK

Bagian ini mencakup platform yang didukung dan fitur utama untuk koleksi paket OS APK berbasis. Untuk informasi lebih lanjut, lihat Alpine Package Keeper di situs web. Alpine Linux

Platform yang didukung

Berikut ini adalah platform yang didukung.

- Alpine Linux

> **Note**
>
> Untuk sistem APK berbasis, Amazon Inspector SBOM Generator mengumpulkan metadata paket dari file. `/lib/apk/db/`

Fitur utama

- Package name collection - Mengekstrak nama setiap paket yang diinstal
- Koleksi versi - Ekstrak versi dari setiap paket yang diinstal
- Identifikasi paket sumber - Mengidentifikasi paket sumber untuk setiap paket yang diinstal

Contoh

Cuplikan berikut adalah contoh dari file APK database.

```
C:Q1JlboSJkrN4qkDcokr4zenpcWEXQ=
P:zlib
V:1.2.13-r1
```

```
A:x86_64
S:54253
I:110592
T:A compression/decompression Library
U:https://zlib.net/
L:Zlib
o:zlib
```

Koleksi paket OS berbasis DPKG

Bagian ini mencakup platform yang didukung dan fitur utama untuk koleksi paket OS DPKG berbasis. Untuk informasi selengkapnya, lihat Paket Debian di Debian situs web.

Platform yang didukung

Platform berikut didukung.

- Debian
- Ubuntu

> **Note**
>
> Untuk sistem DPKG berbasis, Amazon Inspector SBOM Generator mengumpulkan metadata paket dari file. `/var/lib/dpkg/status`

Fitur utama

Berikut ini adalah fitur utama untuk paket OS DPKG berbasis.

- Package name collection - Mengekstrak nama setiap paket yang diinstal
- Koleksi versi - Ekstrak versi dari setiap paket yang diinstal
- Identifikasi paket sumber - Mengidentifikasi paket sumber untuk setiap paket yang diinstal

Contoh

Cuplikan berikut adalah contoh 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/
```

Koleksi paket OS berbasis RPM

Bagian ini mencakup platform yang didukung dan fitur utama untuk koleksi paket OS RPM berbasis. Untuk informasi selengkapnya, lihat RPM Package Manager di RPM situs web.

Platform yang didukung

Platform berikut didukung.

- Alma Linux
- Amazon Linux
- CentOS
- Fedora
- OpenSUSE
- Oracle Linux
- PhotonOS
- RedHat Enterprise Linux
- Rocky Linux

- SUSE Linux Enterprise Server

> **Note**
>
> Untuk sistem RPM berbasis, Amazon Inspector SBOM Generator mengumpulkan metadata paket dari file. `/var/lib/rpm`

Fitur utama

Berikut ini adalah fitur utama untuk koleksi paket OS RPM berbasis.

- Package name collection - Mengekstrak nama setiap paket yang diinstal
- Koleksi versi - Ekstrak versi dari setiap paket yang diinstal
- Identifikasi paket sumber - Mengidentifikasi paket sumber untuk setiap paket yang diinstal
- Dukungan aliran - Ekstrak metadata aliran dari setiap paket yang diinstal

Contoh

Berikut ini adalah contoh cuplikan file 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
```

Koleksi versi OS Windows

Tidak seperti sistem operasi berbasis Linux, Windows tidak menggunakan sistem manajemen paket untuk sistem operasi itu sendiri. Amazon Inspector SBOM Generator hanya mengumpulkan informasi versi OS Windows. Untuk pemindaian aplikasi Windows, gunakan windows-apps pemindai sebagai gantinya. windows-appsPemindai mengumpulkan informasi tentang aplikasi yang diinstal pada sistem Windows. Untuk informasi lebih lanjut, LihatMicrosoft applicationspengumpulan ekosistem.

Fitur utama

- Koleksi versi OS - Mengekstrak versi OS Windows dari Windows Registry. Versi OS yang diekstraksi digunakan untuk deteksi kerentanan untuk OS Windows.

Kunci dan nilai registri

Kunci dan nilai Windows Registry berikut digunakan untuk mengumpulkan nama OS dan informasi versi.

- Kunci Registri

```
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion
```

- Nilai Registri
  - ProductName— Nama dan edisi OS (mis.,"Windows Server 2025 Datacenter")
  - CurrentMajorVersionNumber— versi utama OS
  - CurrentMinorVersionNumber— Versi minor OS
  - CurrentBuild— Jumlah build OS
  - UBR— Jumlah revisi OS

Koleksi paket gambar Chainguard

Bagian ini mencakup platform yang didukung dan fitur utama untuk koleksi paket Chainguard gambar. Untuk informasi lebih lanjut, lihat Gambar di Chainguard situs web.

Platform yang didukung

Platform berikut didukung

- Wolfi Linux

> **Note**
>
> Untuk Chainguard gambar, Amazon Inspector SBOM Generator mengumpulkan metadata paket dari file. `/lib/apk/db/installed`

Fitur utama

Berikut ini adalah fitur utama.

- Package name collection - Mengekstrak nama setiap paket yang diinstal
- Koleksi versi - Ekstrak versi dari setiap paket yang diinstal
- Identifikasi paket sumber - Mengidentifikasi paket sumber untuk setiap paket yang diinstal

Contoh

Cuplikan berikut adalah contoh file Chainguard gambar.

```
P:wolfi-keys
V:1-r8
A:x86_64
L:MIT
T:Wolfi signing keyring
o:wolfi-keys
```

Koleksi paket gambar distroless

Distrolesskontainer adalah gambar kontainer yang mengecualikan manajer paket, shell, dan utilitas lain dalam Linux distribusi. Distrolesscontainer hanya menyertakan dependensi penting yang diperlukan untuk menjalankan aplikasi dan meningkatkan kinerja dan keamanan.

> **Note**
>
> Untuk Distrolessgambar, Amazon Inspector SBOM Generator mengumpulkan metadata paket dari file. `/var/lib/dpkg/status.d` Distribusi hanya Debian dan Ubuntu berbasis yang didukung. Ini dapat diidentifikasi oleh `NAME` bidang dalam sistem `/etc/os-release` file, yang menunjukkan "Debian" atau "Ubuntu."

Fitur utama

- Package name collection - Mengekstrak nama setiap paket yang diinstal
- Koleksi versi - Ekstrak versi dari setiap paket yang diinstal

Contoh

Berikut ini adalah contoh file Distroless gambar.

```
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.
```

Koleksi paket miniMOS

Bagian ini mencakup platform yang didukung dan fitur utama untuk koleksi paket Minimus gambar. Untuk informasi lebih lanjut, lihat situs web Minimus.

Platform yang didukung

Platform berikut didukung.

- MinimOS

> **Note**
>
> Untuk Minimus gambar, Amazon Inspector SBOM Generator mengumpulkan metadata paket dari file. `/lib/apk/db/installed`

Fitur utama

Berikut ini adalah fitur utama.

- Package name collection - Mengekstrak nama setiap paket yang diinstal
- Koleksi versi - Ekstrak nama setiap paket yang diinstal
- Identifikasi paket sumber - Mengidentifikasi paket sumber untuk setiap paket yang diinstal

Berikut ini adalah cuplikan file Minimus gambar.

```
P:ca-certificates-bundle
V:20241121-r1
A:aarch64
L:MPL-2.0 AND MIT
T:
o:ca-certificates
```

Koleksi ketergantungan bahasa pemrograman

Amazon Inspector SBOM Generator mendukung berbagai bahasa dan kerangka kerja pemrograman, yang membentuk kumpulan dependensi yang kuat dan terperinci. Membuat SBOM membantu Anda memahami komposisi perangkat lunak Anda, sehingga Anda dapat mengidentifikasi kerentanan dan menjaga kepatuhan terhadap standar keamanan. Amazon Inspector SBOM Generator mendukung bahasa pemrograman berikut dan format file.

Pergi pemindaian ketergantungan

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
Go	Go	go.mod	N/A	N/A	N/A	N/A	Ya
		go.sum	N/A	N/A	N/A	N/A	Ya
			Ya	N/A	N/A	N/A	Ya

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
		Go Binaries GOMODCACHE	N/A	N/A	N/A	N/A	Tidak

go.mod/go.sum

Gunakan `go.mod` dan `go.sum` file untuk menentukan dan mengunci dependensi dalam Go proyek. Amazon Inspector SBOM Generator mengelola file-file ini secara berbeda berdasarkan versi toolchain. Go

Fitur utama

- Mengumpulkan dependensi dari `go.mod` (jika versi Go toolchain adalah 1.17 atau lebih tinggi)
- Mengumpulkan dependensi dari `go.sum` (jika versi Go toolchain 1.17 atau lebih rendah)
- Parses `go.mod` untuk mengidentifikasi semua dependensi dan versi dependensi yang dideklarasikan

Contoh file **`go.mod`**

Berikut ini adalah contoh `go.mod` file.

```
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
)
```

Contoh file **go.sum**

Berikut ini adalah contoh `go.sum` file.

```
github.com/gin-gonic/gin v1.7.2 h1:VZ7DdRl0sghbA6lVGSkX+UXO2+J0aH7RbsNugG+FA8Q=
github.com/gin-gonic/gin v1.7.2/go.mod h1:ILZ1Ngh2f1pL1ASUj7gGk8lGFeNC8cRTaN2ZhsBNbXU=
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123 h1:b6rCu+qHze
+BUsmC3CZzH8aNu8LzPZTVsNTo64OypSc=
golang.org/x/crypto v0.0.0-20210616213533-5cf6c0f8e123/go.mod h1:K5Dkpb0Q4ewZW/
EzWlQphgJcUMBCzoWrLfDOVzpTGVQ=
```

> **Note**
>
> Masing-masing file ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Go Binari

Amazon Inspector SBOM Generator mengekstrak dependensi dari Go binari yang dikompilasi untuk memberikan jaminan tentang kode yang digunakan.

> **Note**
>
> Amazon Inspector SBOM Generator mendukung pengambilan dan evaluasi versi toolchain dari Go binari yang dibuat menggunakan kompiler resmi. Go Untuk informasi selengkapnya, lihat Unduh dan instal di Go situs web. Jika Anda menggunakan Go rantai alat dari vendor lain, sepertiRed Hat, evaluasi mungkin tidak akurat karena potensi perbedaan dalam distribusi dan ketersediaan metadata.

Fitur utama

- Mengekstrak informasi ketergantungan langsung dari binari Go
- Mengumpulkan dependensi yang tertanam dalam biner

- Mendeteksi dan mengekstrak versi Go toolchain yang digunakan untuk mengkompilasi biner.

GOMODCACHE

Amazon Inspector SBOM Generator memindai cache Go modul untuk mengumpulkan informasi tentang dependensi yang diinstal. Cache ini menyimpan modul yang diunduh untuk memastikan versi yang sama digunakan di berbagai build.

Fitur utama

- Memindai `GOMODCACHE` direktori untuk mengidentifikasi modul yang di-cache
- Mengekstrak metadata terperinci, termasuk nama modul, versi, dan sumber URLs

Contoh struktur

Berikut ini adalah contoh `GOMODCACHE` strukturnya.

```
~/go/pkg/mod/
### github.com/gin-gonic/gin@v1.7.2
### golang.org/x/crypto@v0.0.0-20210616213533-5cf6c0f8e123
```

> **Note**
>
> Struktur ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pemindaian ketergantungan Java

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
`Java`	`Maven`	JavaAplikasi yang dikompilasi (.jar/.war/.ear) `pom.xml`	N/A N/A	N/A N/A	Ya Ya	N/A N/A	Ya Ya

> **Note**
>
> Fitur evaluasi kerentanan kami hanya mendukung repositori Maven Central. Repositori pihak ketiga, sepertiJBoss Enterprise Maven Repository, saat ini tidak didukung.

Amazon Inspector SBOM Generator melakukan pemindaian Java ketergantungan dengan menganalisis aplikasi dan file yang dikompilasiJava. `pom.xml` Saat memindai aplikasi yang dikompilasi, pemindai menghasilkan hash SHA-1 untuk verifikasi integritas, mengekstrak file yang disematkan, dan mem-parsing `pom.properties` file bersarang. `pom.xml`

Koleksi hash SHA—1 (untuk file.jar, .war, .ear yang dikompilasi)

Amazon Inspector SBOM Generator mencoba mengumpulkan hash SHA—1 untuk semua `.ear` `.jar`, dan `.war` file dalam proyek untuk menjamin integritas dan keterlacakan artefak yang dikompilasi. Java

Fitur utama

- Menghasilkan hash SHA—1 untuk semua artefak yang dikompilasi Java

Contoh artefak

Berikut ini adalah contoh artefak 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**
>
> Artefak ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

pom.properti

`pom.properties`File ini digunakan dalam Maven proyek untuk menyimpan metadata proyek, termasuk nama paket dan versi paket. Amazon Inspector SBOM Generator mem-parsing file ini untuk mengumpulkan informasi proyek.

Fitur utama

- Mem-parsing dan mengekstrak artefak paket, grup paket, dan versi paket

Contoh file **pom.properties**

Berikut ini adalah contoh `pom.properties` file.

```
#Generated by Maven
#Tue Mar 16 15:44:02 UTC 2021

version=1.6.0
groupId=net.datafaker
artifactId=datafaker
```

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Tidak termasuk parsing bersarang **pom.xml**

Jika Anda ingin mengecualikan `pom.xml` parsing saat memindai Java aplikasi yang dikompilasi, gunakan `--skip-nested-pomxml` argumen.

pom.xml

`pom.xml`File tersebut adalah file konfigurasi inti untuk Maven proyek. Ini berisi informasi tentang proyek dan dependensi proyek. Amazon Inspector SBOM Generator mem-parsing pom.xml file untuk mengumpulkan dependensi, memindai file mandiri di repositori dan file di dalam file yang dikompilasi. .jar

Fitur utama

- Mem-parsing dan mengekstrak artefak paket, grup paket, dan versi paket dari `pom.xml` file.

MavenLingkup dan tag yang didukung

Dependensi dikumpulkan dengan cakupan berikut: Maven

- mengompilasikan
- provided
- runtime
- pengujian
- sistem
- impor

Dependensi dikumpulkan dengan Maven tag berikut:. `<optional>true</optional>`

Contoh **`pom.xml`** file dengan cakupan

Berikut ini adalah contoh `pom.xml` file dengan cakupan.

```
<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>
```

Contoh **`pom.xml`** file tanpa ruang lingkup

Berikut ini adalah contoh `pom.xml` file tanpa ruang lingkup.

```
<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**
>
> Masing-masing file ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

JavaScript pemindaian ketergantungan

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
`Javascript`	`Node Modules`	`node_modules/*/package.json`	N/A	N/A	Ya	Ya	Ya
	`NPM`		N/A	Ya	N/A	N/A	Tidak
	`PNPM`		N/A	Ya	N/A	N/A	Tidak
	`YARN`	`package-lock.json`	N/A	Ya	N/A	N/A	Tidak

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
		(v1, v2, and v3) / npm-shrinkwrap.json pnpm-lock.yaml yarn.lock					

package.json

`package.json`File adalah komponen inti dari Node.js proyek. Ini berisi metadata tentang paket yang diinstal. Amazon Inspector SBOM Generator memindai file ini untuk mengidentifikasi nama paket dan versi paket.

Fitur utama

- Mem-parsing struktur file JSON untuk mengekstrak nama dan versi paket
- Mengidentifikasi paket pribadi dengan nilai pribadi

Contoh file **`package.json`**

Berikut ini adalah contoh `package.json` file.

```
{
"name": "arrify",
```

```
"private": true,
"version": "2.0.1",
"description": "Convert a value to an array",
"license": "MIT",
"repository": "sindresorhus/arrify"
}
```

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

package-lock.json

`package-lock.json`File secara otomatis dihasilkan oleh npm untuk mengunci versi dependensi yang tepat yang diinstal untuk sebuah proyek. Ini memastikan konsistensi dalam lingkungan dengan menyimpan versi yang tepat dari semua dependensi dan sub-dependensinya. File ini dapat membedakan antara dependensi reguler dan dependensi pengembangan.

Fitur utama

- Mem-parsing struktur file JSON untuk mengekstrak nama paket dan versi paket
- Mendukung deteksi ketergantungan dev

Contoh file **`package-lock.json`**

Berikut ini adalah contoh `package-lock.json` file.

```
"verror": {
"version": "1.10.0",
"resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
"integrity": "sha1-OhBcoXBTr1XW4nDB+CiGguGNpAA=",
"requires": {
    "assert-plus": "^1.0.0",
```

```
    "core-util-is": "1.0.2",
    "extsprintf": "^1.2.0"
}
},
"wrappy": {
"version": "1.0.2",
"resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",
"integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",
"dev": true
},
"yallist": {
"version": "3.0.2",
"resolved": "https://registry.npmjs.org/yallist/-/yallist-3.0.2.tgz",
"integrity": "sha1-hFK0u36Dx8GI2AQcGoN8dz1ti7k="
}
```

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

npm-shrinkwrap.json

npmsecara otomatis menghasilkan `package-lock.json` dan `npm-shrinkwrap.json` file untuk mengunci versi dependensi yang tepat yang diinstal untuk sebuah proyek. Ini menjamin konsistensi dalam lingkungan dengan menyimpan versi yang tepat dari semua dependensi dan sub-dependensi. File membedakan antara dependensi reguler dan dependensi pengembangan.

Fitur utama

- Parse `package-lock` versi 1 ,2, dan 3 dari struktur JSON file untuk mengekstrak nama paket dan versi
- Deteksi ketergantungan pengembang didukung (`package-lock.json`menangkap dependensi produksi dan pengembangan, memungkinkan alat untuk mengidentifikasi paket mana yang digunakan dalam lingkungan pengembangan)
- `npm-shrinkwrap.json`File diprioritaskan di atas file `package-lock.json`

Contoh

Berikut ini adalah contoh `package-lock.json` file.

```
"verror": {
            "version": "1.10.0",
            "resolved": "https://registry.npmjs.org/verror/-/verror-1.10.0.tgz",
            "integrity": "sha1-OhBcoXBTr1XW4nDB+CiGguGNpAA=",
            "requires": {
                "assert-plus": "^1.0.0",
                "core-util-is": "1.0.2",
                "extsprintf": "^1.2.0"
            }
        },
        "wrappy": {
            "version": "1.0.2",
            "resolved": "https://registry.npmjs.org/wrappy/-/wrappy-1.0.2.tgz",
            "integrity": "sha1-tSQ9jz7BqjXxNkYFvA0QNuMKtp8=",
            "dev": true
        },
        "yallist": {
            "version": "3.0.2",
            "resolved": "https://registry.npmjs.org/yallist/-/yallist-3.0.2.tgz",
            "integrity": "sha1-hFK0u36Dx8GI2AQcGoN8dz1ti7k="
}
```

pnpm-yaml.lock

`pnpm-lock.yaml`File dihasilkan oleh pnpm untuk mempertahankan catatan versi ketergantungan yang diinstal. Ini juga melacak dependensi pengembangan secara terpisah.

Fitur utama

- Mem-parsing struktur file YAMM untuk mengekstrak nama dan versi paket
- Mendukung deteksi ketergantungan dev

Contoh

Berikut ini adalah contoh `pnpm-lock.yaml` file.

```
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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

yarn.lock

Amazon Inspector SBOM Generator mencoba mengumpulkan hash SHA—1 untuk `.ear`, `.jar`, dan `.war` file dalam proyek untuk menjamin integritas dan keterlacakan artefak yang dikompilasi. Java

Fitur utama

- Menghasilkan hash SHA—1 untuk semua artefak yang dikompilasi Java

Contoh artefak SHA—1

Berikut ini adalah contoh artefak 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**
>
> Artefak ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pemindaian ketergantungan.NET

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
.NET	.NET Core	*.deps.json	N/A	N/A	N/A	N/A	Ya
	Nuget	Packages.config	N/A	N/A	N/A	N/A	Ya
	Nuget	packages.lock.json	N/A	N/A	Ya	N/A	Ya
	.NET	.csproj	N/A	N/A	N/A	N/A	Ya

Packages.config

`Packages.config`File ini adalah file XMLyang digunakan oleh versi lama Nuget untuk mengelola dependensi proyek. Ini mencantumkan semua paket yang direferensikan oleh proyek, termasuk versi tertentu.

Fitur utama

- Mem-parsing struktur XMLuntuk mengekstrak paket IDs dan versi

Contoh

Berikut ini adalah contoh `Packages.config` file.

```
<?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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

*.deps.json

`*.deps.json`File dihasilkan oleh .NET Core proyek dan berisi informasi terperinci tentang semua dependensi, termasuk jalur, versi, dan dependensi runtime. File ini memastikan runtime memiliki informasi yang diperlukan untuk memuat versi dependensi yang benar.

Fitur utama

- Mem-parsing struktur JSON untuk detail ketergantungan yang komprehensif
- Mengekstrak nama dan versi paket dalam `libraries` daftar.

Contoh file `.deps.json`

Berikut ini adalah contoh `.deps.json` file.

```
{
"runtimeTarget": {
    "name": ".NETCoreApp,Version=v7.0",
    "signature": ""
},
"libraries": {
    "sample-Nuget/1.0.0": {
        "type": "project",
        "serviceable": false,
```

```
        "sha512": ""
    },
    "Microsoft.EntityFrameworkCore/7.0.5": {
        "type": "package",
        "serviceable": true,
        "sha512": "sha512-
RXbRLHHWP2Z3pq8qcL5nQ6LPeoOyp8hasM5bd0Te8PiQi3RjWQR4tcbdY5XMqQ+oTO9wA8/RLhZRn/
hnxlTDnQ==",
        "path": "microsoft.entityframeworkcore/7.0.5",
        "hashPath": "microsoft.entityframeworkcore.7.0.5.nupkg.sha512"
    },
}
```

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

packages.lock.json

`packages.lock.json`File ini digunakan oleh versi yang lebih baru Nuget untuk mengunci versi dependensi yang tepat untuk .NET proyek guna menjamin versi yang sama digunakan secara konsisten di lingkungan yang berbeda.

Fitur utama

- Mem-parsing struktur JSON untuk membuat daftar dependensi terkunci
- Mendukung dependensi langsung dan transitif
- Ekstrak nama paket dan versi yang diselesaikan

Contoh file **`packages.lock.json`**

Berikut ini adalah contoh `packages.lock.json` file.

```
{
"version": 1,
"dependencies": {
"net7.0": {
  "Microsoft.EntityFrameworkCore": {
    "type": "Direct",
    "requested": "[7.0.5, )",
    "resolved": "7.0.5",
    "contentHash": "RXbRLHHWP2Z3pq8qcL5nQ6LPeoOyp8hasM5bd0Te8PiQi3RjWQR4tcbdY5XMqQ
+oTO9wA8/RLhZRn/hnxlTDnQ==",
    "dependencies": {
      "Microsoft.EntityFrameworkCore.Abstractions": "7.0.5",
      "Microsoft.EntityFrameworkCore.Analyzers": "7.0.5",
      "Microsoft.Extensions.Caching.Memory": "7.0.0",
      "Microsoft.Extensions.DependencyInjection": "7.0.0",
      "Microsoft.Extensions.Logging": "7.0.0"
    }
  },
  "Newtonsoft.Json": {
    "type": "Direct",
    "requested": "[13.0.3, )",
    "resolved": "13.0.3",
    "contentHash": "HrC5BXdl00IP9zeV+0Z848QWPAoCr9P3bDEZguI+gkLcBKAOxix/tLEAAHC
+UvDNPv4a2d18lOReHMOagPa+zQ=="
  },
  "Microsoft.Extensions.Primitives": {
    "type": "Transitive",
    "resolved": "7.0.0",
    "contentHash": "um1KU5kxcRp3CNuI8o/GrZtD4AIOXDk
+RLsytjZ9QPok3ttLUelLKpilVPuaFT3TFjOhSibUAso0odbOaCDj3Q=="
  }
}
}
}
```


> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

.csproj

`.csproj`File ini ditulis dalam XHTML dan file proyek untuk .NET proyek. Ini termasuk referensi ke Nuget paket, properti proyek, dan konfigurasi build.

Fitur utama

- Mem-parsing XMLstruktur untuk mengekstrak referensi paket

Contoh file **`.csproj`**

Berikut ini adalah contoh `.csproj` file.

```
<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>
```

Contoh file **`.csproj`**

Berikut ini adalah contoh `.csproj` file.

```
<PackageReference Include="ExamplePackage" Version="6.*" />
<PackageReferencePackageReference Include="ExamplePackage" Version="(4.1.3,)" />
<PackageReference Include="ExamplePackage" Version="(,5.0)" />
<PackageReference Include="ExamplePackage" Version="[1,3)" />
<PackageReference Include="ExamplePackage" Version="[1.3.2,1.5)" />
```

> **Note**
>
> Masing-masing file ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pemindaian ketergantungan PHP

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
PHP	Composer	`composer.lock`	N/A	N/A	Ya	N/A	Ya
		`/vendor/composer/installed.json`	N/A	N/A	Ya	N/A	Ya

composer.lock

`composer.lock`File secara otomatis dihasilkan saat menjalankan perintah composer install atau composer update. File ini menjamin versi dependensi yang sama diinstal di setiap lingkungan. Ini memberikan proses pembuatan yang konsisten dan andal.

Fitur utama

- Mem-parsing format JSON untuk data terstruktur

- Mengekstrak nama dan versi ketergantungan

Contoh file **`composer.lock`**

Berikut ini adalah contoh `composer.lock` file.

```
{
"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**
>
> Ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

/vendor/composer/installed.json

`/vendor/composer/installed.json`File ini terletak di `vendor/composer` direktori dan menyediakan daftar lengkap semua paket yang diinstal dan versi paket.

Fitur utama

- Mem-parsing format JSON untuk data terstruktur
- Mengekstrak nama dan versi ketergantungan

Contoh file **`/vendor/composer/installed.json`**

Berikut ini adalah contoh `/vendor/composer/installed.json` file.

```
{
"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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pemindaian ketergantungan Python

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
Python	pip	requirements.txt	N/A	N/A	N/A	N/A	Ya
	Poetry	Poetry.lock	N/A	N/A	N/A	N/A	Ya
	Pipenv	Pipfile.lock	N/A	N/A	N/A	N/A	Ya
	Egg/Wheel	.egg-info/PKG-INFO	N/A	N/A	N/A	N/A	Ya
		.dist-info/METADATA	N/A	N/A	N/A	N/A	Ya

requirements.txt

`requirements.txt`File ini adalah format yang banyak digunakan dalam Python proyek untuk menentukan dependensi proyek. Setiap baris dalam file ini menyertakan paket dengan batasan versinya. Amazon Inspector SBOM Generator mem-parsing file ini untuk mengidentifikasi dan katalog dependensi secara akurat.

Fitur utama

- Mendukung penentu versi (== dan ~=)
- Mendukung komentar dan garis ketergantungan yang kompleks

> **Note**
>
> Penentu versi <= dan => tidak didukung.

Contoh file **`requirements.txt`**

Berikut ini adalah contoh `requirements.txt` file.

```
flask==1.1.2
requests==2.24.0
numpy==1.18.5
foo~=1.2.0
# Comment about a dependency
scipy. # invalid
```

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pipfile.lock

Pipenvadalah alat yang menghadirkan yang terbaik dari semua dunia pengemasan (dibundel, disematkan, dan tidak disematkan). `Pipfile.lock`Mengunci versi dependensi yang tepat untuk memfasilitasi build deterministik. Amazon Inspector SBOM Generator membaca file ini untuk mencantumkan dependensi dan versi yang diselesaikan.

Fitur utama

- Mem-parsing format JSON untuk resolusi ketergantungan
- Mendukung dependensi default dan pengembangan

Contoh file **`Pipfile.lock`**

Berikut ini adalah contoh `Pipfile.lock` file.

```
{
"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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Puisi.lock

Poetryadalah manajemen ketergantungan dan alat pengemasan untuk Python. `Poetry.lock`File mengunci versi dependensi yang tepat untuk memfasilitasi lingkungan yang konsisten. Amazon Inspector SBOM Generator mengekstrak informasi ketergantungan terperinci dari file ini.

Fitur utama

- Mem-parsing format TOMM untuk data terstruktur
- Mengekstrak nama ketergantungan, dan versi

Contoh file **`Poetry.lock`**

Berikut ini adalah contoh `Poetry.lock` file.

```
[[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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Telur/Roda

Untuk paket Python yang diinstal secara global, Amazon Inspector SBOM Generator mendukung penguraian file metadata yang ditemukan di direktori dan. `.egg-info/PKG-INFO` `.dist-info/METADATA` File-file ini menyediakan metadata terperinci tentang paket yang diinstal.

Fitur utama

- Ekstrak nama paket, dan versi
- Mendukung format telur dan roda

Contoh file **`PKG-INFO/METADATA`**

Berikut ini adalah contoh `PKG-INFO/METADATA` file.

```
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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pemindaian ketergantungan Ruby

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchaing	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
`Ruby`	`Bundler`	`Gemfile.lock`	N/A	N/A	Ya	N/A	Ya
			N/A	N/A	N/A	N/A	Ya
		`.gemspec`	N/A	N/A	N/A	N/A	Ya

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchaing	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif
		`globall installed Gems`					

Gemfile.lock

`Gemfile.lock`File mengunci versi yang tepat dari semua dependensi untuk memastikan versi yang sama digunakan di setiap lingkungan.

Fitur utama

- Mem-parsing `Gemfile.lock` file ke dependensi identitas dan versi dependensi
- Mengekstrak nama paket rinci dan versi paket

Contoh file **`Gemfile.lock`**

Berikut ini adalah contoh `Gemfile.lock` file.

```
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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

.gemspec

`.gemspec`File tersebut adalah RubyGem file yang berisi metadata tentang permata. Amazon Inspector SBOM Generator mem-parsing file ini untuk mengumpulkan informasi rinci tentang permata.

Fitur utama

- Mem-parsing dan mengekstrak nama permata dan versi permata

> **Note**
>
> Spesifikasi referensi tidak didukung.

Contoh file `.gemspec`

Berikut ini adalah contoh `.gemspec` file.

```
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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Permata yang dipasang secara global

Amazon Inspector SBOM Generator mendukung pemindaian permata yang diinstal secara global, yang terletak di direktori standar, seperti di Amazon `/usr/local/lib/ruby/gems/<ruby_version>/gems/` EC2/Amazon ECR dan di Lambda. `ruby/gems/<ruby_version>/gems/` Ini memastikan semua dependensi yang diinstal secara global diidentifikasi dan dikatalogkan.

Fitur utama

- Mengidentifikasi dan memindai semua permata yang diinstal secara global di direktori standar
- Mengekstrak metadata dan informasi versi untuk setiap permata yang diinstal secara global

Contoh struktur direktori

Berikut ini adalah contoh struktur direktori.

```
.
### /usr/local/lib/ruby/3.5.0/gems/
```

```
### actrivesupport-6.1.4
### concurrent-ruby-1.1.9
### i18n-1.8.10
```

> **Note**
>
> Struktur ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Pemindaian ketergantungan karat

Bahasa pemrograman	Manajer Package	Artefak yang didukung	Dukungan Toolchain	Ketergantungan pengembangan	Ketergantungan transitif	Bendera pribadi	Secara rekursif

`Rust`	`Cargo.toml`	`Cargo.toml`
		`Cargo.lock`
		`Rust binary (built with cargo-auditable)`
			N/A
			N/A
			Ya
				N/A
				N/A
				N/A
					N/A
					Ya
					N/A
						N/A
						N/A
						N/A
							Ya
							Ya
							Ya

Cargo.toml

`Cargo.toml`File adalah file manifes untuk Rust proyek.

Fitur utama

- Mem-parsing dan mengekstrak `Cargo.toml` file untuk mengidentifikasi nama paket proyek dan versi.

Contoh file **`Cargo.toml`**

Berikut ini adalah contoh `Cargo.toml` file.

```
[package]
name = "wait-timeout"
version = "0.2.0"
description = "A crate to wait on a child process with a timeout specified across Unix
 and\nWindows platforms.\n"
homepage = "https://github.com/alexcrichton/wait-timeout"
documentation = "https://docs.rs/wait-timeout"
readme = "README.md"
categories = ["os"]
license = "MIT/Apache-2.0"
repository = "https://github.com/alexcrichton/wait-timeout"
[target."cfg(unix)".dependencies.libc]
version = "0.2"
[badges.appveyor]
repository = "alexcrichton/wait-timeout"
```

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Cargo.lock

`Cargo.lock`File mengunci versi dependensi untuk memastikan versi yang sama digunakan setiap kali proyek dibangun.

Fitur utama

- Mem-parsing `Cargo.lock` file untuk mengidentifikasi semua dependensi dan versi dependensi.

Contoh file **`Cargo.lock`**

Berikut ini adalah contoh `Cargo.lock` file.

```
# 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**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Biner karat dengan cargo-auditable

Amazon Inspector SBOM Generator mengumpulkan dependensi dari Rust binari yang dibangun dengan perpustakaan. `cargo-auditable` Ini memberikan informasi ketergantungan tambahan dengan mengaktifkan ekstraksi ketergantungan dari binari yang dikompilasi.

Fitur utama

- Mengekstrak informasi ketergantungan langsung dari Rust binari yang dibangun dengan perpustakaan `cargo-auditable`

- Mengambil metadata dan informasi versi untuk dependensi yang disertakan dalam binari

> **Note**
>
> File ini menghasilkan output yang berisi URL paket. URL ini dapat digunakan untuk menentukan informasi tentang paket perangkat lunak saat membuat tagihan materi perangkat lunak dan dapat dimasukkan dalam ScanSbomAPI. Untuk informasi selengkapnya, lihat url paket di Situs Web. GitHub

Artefak yang tidak didukung

Bagian ini menjelaskan artefak yang tidak didukung.

Java

Generator Amazon Inspector SBOM Generator hanya mendukung deteksi kerentanan untuk dependensi yang bersumber dari repositori arus utama. Maven MavenRepositori pribadi atau kustom, seperti Red Hat Maven danJenkins, tidak didukung. Untuk deteksi kerentanan yang akurat, pastikan Java dependensi ditarik dari repositori arus utama. Maven Dependensi dari repositori lain tidak akan tercakup dalam pemindaian kerentanan.

JavaScript

bundel esbuild

Untuk bundel esbuild yang diperkecil, Amazon Inspector SBOM Generator tidak mendukung pemindaian dependensi untuk proyek yang digunakan. esbuild Peta sumber yang dihasilkan oleh esbuild tidak menyertakan metadata yang memadai (nama dan versi ketergantungan) yang diperlukan untuk pembuatan yang akurat. Sbomgen Untuk hasil yang andal, pindai file proyek asli, seperti `node_modules/directory` dan`package-lock.json`, sebelum proses bundling.

package.json

Amazon Inspector SBOM Generator tidak mendukung pemindaian file package.json tingkat root untuk informasi ketergantungan. File ini hanya menentukan nama paket dan rentang versi, tetapi tidak menyertakan versi paket yang sepenuhnya diselesaikan. Untuk hasil pemindaian yang akurat, gunakan `package.json` atau file kunci lainnya, seperti `yarn.lock` dan`pnpm.lock`, yang menyertakan versi yang diselesaikan.

Dotnet

Saat menggunakan versi mengambang atau rentang versi`PackageReference`, menjadi lebih menantang untuk menentukan versi paket yang tepat yang digunakan dalam proyek tanpa melakukan resolusi paket. Versi mengambang dan rentang versi memungkinkan pengembang untuk menentukan rentang versi paket yang dapat diterima daripada versi tetap.

Pergi binari

Amazon Inspector SBOM Generator tidak memindai Go binari yang dibuat dengan flag build yang dikonfigurasi untuk mengecualikan ID build. Bendera build ini mencegah Amazon Inspector SBOM Generator memetakan biner secara akurat ke sumber aslinya. GoBinari yang tidak jelas tidak didukung karena ketidakmampuan untuk mengekstrak informasi paket. Untuk pemindaian dependensi yang akurat, pastikan Go binari dibuat dengan setelan default, termasuk ID build.

Biner karat

Amazon Inspector SBOM Generator hanya memindai Rust binari jika binari dibuat menggunakan pustaka yang dapat diaudit kargo. Rustbinari yang tidak menggunakan perpustakaan ini tidak memiliki metadata yang diperlukan untuk ekstraksi ketergantungan yang akurat. Amazon Inspector SBOM Generator mengekstrak versi Rust toolchain yang dikompilasi mulai dari Rust 1.7.3, tetapi hanya untuk binari di lingkungan. Linux Untuk pemindaian komprehensif, buat Rust binari Linux menggunakan cargo-auditable.

> **Note**
>
> Deteksi kerentanan untuk Rust rantai alat itu sendiri tidak didukung, bahkan jika versi toolchain diekstraksi.

Koleksi ekosistem komprehensif Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator adalah alat untuk membuat tagihan bahan perangkat lunak (SBOM) dan melakukan pemindaian kerentanan untuk paket yang didukung dari sistem operasi dan bahasa pemrograman. Ini mendukung pemindaian berbagai ekosistem di luar sistem operasi inti, memastikan analisis komponen infrastruktur yang kuat dan terperinci. Dengan membuat SBOM,

Anda dapat memahami komposisi tumpukan teknologi modern, mengidentifikasi kerentanan dalam komponen ekosistem, dan mendapatkan visibilitas ke perangkat lunak pihak ketiga.

Ekosistem yang didukung

Koleksi ekosistem memperluas generasi SBOM di luar paket yang diinstal melalui manajer paket OS. Ini dilakukan melalui pengumpulan aplikasi yang digunakan dalam metode alternatif, seperti instalasi manual. Amazon Inspector SBOM Generator mendukung pemindaian untuk ekosistem berikut:

Ekosistem	Aplikasi
7-Zip	7-Zippengarsipan (versi 21.07 dan lebih tinggi)
Apache	Apache httpd Apache tomcat
Atlassian	Jira Core Confluence Jira Software Jira Service Management
Curl	Curl Libcurl
Elasticsearch	Elasticsearch
Google	Chrome
Java	JDK JRE Amazon Corretto
Jenkins	Jenkins(versi 2.400.* dan lebih tinggi)
MariaDB dan MySQL	MariaDB Server(10.6+, 11.x, 12.x)

Ekosistem	Aplikasi
	Oracle MySQL Server Server(8.0, 8.4, 9.4+)
Microsoft applications	PowerShell NuGet CLI Visual Studio Code Microsoft Edge SharePoint Server Microsoft Defender Exchange Server Visual Studio .NET Runtime ASP.NET Core Runtime Microsoft Teams Outlook for Windows Microsoft Office Microsoft 365
Nginx	Nginx
Node	Node
Node.JS	node
OpenSSH	OpenSSH(versi 9 dan 10)
OpenSSL	OpenSSL
Oracle	Oracle Database Server

Ekosistem	Aplikasi
PHP	PHP(versi 8.1 dan lebih tinggi)
WordPress	core plugin theme

7-Zippengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- 7 Zip archiver (versi 21.07 atau lebih tinggi)

Fitur utama

- Memeriksa 7-Zip binari untuk mengekstrak informasi versi tertanam.

> **Note**
>
> Secara khusus, ia mencari nilai versi produk dari biner.

Platform yang didukung — 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`

Contoh PURL

Berikut ini adalah contoh URL paket untuk7-Zip.

```
pkg:generic/7zip/7zip@25.01
```

Apachepengumpulan ekosistem

Bagian ini memberikan detail tentang aplikasi Apache httpd dan Apache tomcat.

Apache httpd

Aplikasi-aplikasi yang didukung

- Apache httpd

> **Note**
>
> Evaluasi kerentanan hanya berlaku untuk Apache httpd versi 2.0 dan yang lebih tinggi.

Fitur utama

- Mem-parsing `/include/ap_release.h` file untuk mengekstrak makro instalasi, yang berisi string pengenal utama, string pengidentifikasi minor, dan string pengidentifikasi patch.

Platform yang didukung

Amazon Inspector SBOM Generator memindai instalasi di jalur instalasi umum di seluruh platform:

Unix

- `/usr/local/apache2/include/`

Windows

- `/Apache24/include/`
- `/Program Files/Apache24/include/`

- `/Program Files (x86)/Apache24/include/`

Contoh file **`ap_release.h`**

Berikut ini adalah contoh konten di dalam `ap_release.h` file.

```
//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
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk `Apache httpd` aplikasi.

```
Sample PURL: pkg:generic/apache/httpd@2.4.1
```

Apache tomcat

Aplikasi-aplikasi yang didukung

- Apache tomcat

> **Note**
>
> Evaluasi kerentanan hanya berlaku untuk Apache tomcat versi 9.0 dan lebih tinggi.

Fitur utama

- Membongkar `catalina.jar` file untuk mengekstrak makro instalasi di dalam `META-INF/MANIFEST.MF` file, yang berisi string versi.

Platform yang didukung

Amazon Inspector SBOM Generator memindai instalasi di jalur instalasi umum di seluruh platform:

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/`

Contoh file **`catalina.jar/META-INF/MANIFEST.MF`**

Berikut ini adalah contoh konten di dalam `catalina.jar/META-INF/MANIFEST.MF` file.

```
//truncated

Implementation-Title: Apache Tomcat
Implementation-Vendor: Apache Software Foundation
Implementation-Version: 10.1.31

//truncated
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk `Apache tomcat` aplikasi.

```
Sample PURL: pkg:generic/apache/tomcat@10.1.31
```

Atlassianpengumpulan ekosistem

Bagian ini memberikan rincian tentang produk dan aplikasi Atlassian server.

Atlassian Server Products

Aplikasi-aplikasi yang didukung

- Jira Core
- Confluence

Fitur utama

- Jira Core— Mem-parsing properti POM Maven dari `atlassian-jira-webapp` untuk mengekstrak informasi versi.
- Confluence— Mem-parsing properti POM Maven dari `confluence-webapp` untuk mengekstrak informasi versi.

Platform yang didukung

Amazon Inspector SBOM Generator memindai instalasi di jalur instalasi umum:

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`

Contoh PURL

Berikut ini adalah contoh paket URLs untuk produk Atlassian server.

```
// Jira Core
pkg:generic/atlassian/jira-core@10.0.1?distro=linux

// Confluence
pkg:generic/atlassian/confluence@9.2.7?distro=linux
```

Atlassian Applications

Aplikasi-aplikasi yang didukung

- Jira Software
- Jira Service Management

Fitur utama

- Jira Software— Mendeteksi melalui `jira-software-application` JAR dan mengekstrak versi dari properti Maven POM.
- Jira Service Management— Mendeteksi melalui `jira-servicedesk-application` JAR dan mengekstrak versi dari properti Maven POM.

Platform yang didukung

Amazon Inspector SBOM Generator memindai instalasi di jalur instalasi umum:

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`

Contoh PURL

Berikut ini adalah contoh paket URLs untuk Atlassian aplikasi.

```
// 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
```

Curlpengumpulan ekosistem

Bagian ini memberikan rincian tentang Curl dan Libcurl aplikaton.

Curl

Aplikasi-aplikasi yang didukung

- Curl

Platform yang didukung

- Unix— Linux dan macOS
  - /usr/local/bin/curl

Fitur utama - Curl

- Memeriksa curl binari untuk mengekstrak informasi versi tertanam.

> **Note**
>
> Secara khusus, ia mencari string versi di bagian biner yang dapat dieksekusi (untuk binari ELF di Linux), `.rodata` bagian (untuk binari PE di Windows), atau `.rdata` bagian __cstring (untuk binari MaCho di macOS).

Curl version string

Berikut ini adalah contoh string versi yang disematkan dalam Curl biner:

```
curl/8.14.1
```

Versi `8.14.1` diekstraksi dari string untuk mengidentifikasi `Curl` versi.

Contoh PURL (Curl)

Berikut ini adalah contoh URL paket untuk file `Curl` versi.

```
Sample PURL: pkg:generic/curl/curl@8.14.1
```

Libcurl

Aplikasi-aplikasi yang didukung

- Libcurl

Platform yang didukung

- Unix— Linux dan macOS
  - /usr/local/bin/curl/curlver.h

Fitur utama - Libcurl

- Memeriksa curlver.h untuk mengekstrak informasi versi tertanam untukLibcurl.

> **Note**
>
> Secara khusus, ia mengekstrak versi dari yang ditentukan`LIBCURL_VERSION_MAJOR`,`LIBCURL_VERSION_MINOR`, dan `LIBCURL_VERSION_PATCH` variabel.

Libcurl version string

Berikut ini adalah contoh variabel versi dalam `curlver.h` file:

```
#define LIBCURL_VERSION_MAJOR 8
    #define LIBCURL_VERSION_MINOR 14
    #define LIBCURL_VERSION_PATCH 1
```

Versi `8.14.1` diekstraksi dari baris ini untuk mengidentifikasi `Libcurl` versi.

Contoh PURL (Libcurl)

Berikut ini adalah contoh URL paket untuk file `Libcurl` versi.

```
Sample PURL: pkg:generic/curl/libcurl@8.14.1
```

Elasticsearchpengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- Elasticsearch

> **Note**
>
> Evaluasi kerentanan hanya berlaku untuk Elasticsearch versi 7.17.0.

Fitur utama

- Version— Membongkar `elasticsearch-<specific.version>.jar` file untuk mengekstrak makro instalasi di dalam `META-INF/MANIFEST.MF` file, yang berisi string Elasticsearch versi.

Platform yang didukung

- Linux—`/etc/elasticsearch/lib`,`/opt/elasticsearch/lib/`, dan `/usr/share/elasticsearch/lib/`
- macOS – `/usr/local/var/lib/elasticsearch/lib/`
- Windows—`/elasticsearch/`,`/Program Files (x86)/Elastic/elasticsearch/lib/`, dan `/Program Files/Elastic/elasticsearch/lib/`

Contoh file **`elasticsearch-<specific.version>.jar/META-INF/MANIFEST.MF`**

Berikut ini adalah contoh `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
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk sebuah `elasticsearch-<specific.version>.jar/META-INF/MANIFEST.MF` file.

```
pkg:generic/elastic/elasticsearch@8.19.0-SNAPSHOT
```

Googlepengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- Google Chrome
- Puppeteer(mendukung perpustakaan dalang; dalang inti tidak termasuk)

> **Note**
>
> Puppeteermendukung perpustakaan dalang. Puppeteerinti tidak termasuk.

Artefak yang didukung

Amazon Inspector mengumpulkan Google Chrome informasi dari berikut ini:

- `chrome/VERSION`File (sumber build)
- `chrome.exe`File (Windows Chromeinstalasi)
- `puppeteer`File (instalasi)

Untuk setiap artefak yang didukung, Sbomgen mem-parsing dan mengumpulkan chrome file atau file. puppeteer Untuk puppeteer instalasi, Chromium versi yang sesuai dikumpulkan berdasarkan puppeteer versi. Untuk informasi selengkapnya, lihat Browser yang didukung di situs web Puppeteer.

Ketika variabel `PUPPETEER_SKIP_CHROMIUM_DOWNLOAD` lingkungan diatur ke`true`, evaluasi dilewati, dan `skip_chromium_download=true` qualifier ditambahkan ke URL Puppeteer paket.

Contoh file **`chrome/VERSION`** versi

Berikut ini adalah contoh file `chrome/VERSION` versi.

```
MAJOR=130
MINOR=0
BUILD=6723
PATCH=58
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk file `chrome/VERSION` versi.

```
Sample PURL: pkg:generic/google/chrome@131.0.6778.87
```

Contoh file **`puppeteer`** versi

Berikut ini adalah contoh file `puppeteer` versi.

```
{
"name": "puppeteer",
"version": "23.9.0",
"description": "A high-level API to control headless Chrome over the DevTools
 Protocol",
"keywords": [
  "puppeteer",
  "chrome",
  "headless",
  "automation"
]
}
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk file `puppeteer` versi.

```
Sample PURL: pkg:generic/google/puppeteer@23.9.0
```

Contoh PURL

Berikut ini adalah contoh URL paket dengan skip qualifier untuk file `puppeteer` versi.

```
pkg:generic/google/puppeteer@22.15.0?distro=linux&skip_chromium_download=true
```

Javapengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- Oracle JDK
- Oracle JRE
- Amazon Corretto

Fitur utama

- Ekstrak string Java instalasi.
- Mengidentifikasi jalur direktori yang berisi Java runtime.
- Mengidentifikasi vendor sebagaiOracle JDK,Oracle JRE, danAmazon Corretto.

Amazon Inspector SBOM Generator memindai Java instalasi di jalur dan platform penginstalan berikut:

- macOS: `/Library/Java/JavaVirtualMachines`
- Linux 32-bit: `/usr/lib/jvm`
- Linux 64-bit: `/usr/lib64/jvm`
- Linux (generic): `/usr/java and /opt/java`

Contoh informasi Java versi

Folllowing adalah contoh Oracle Java rilis.

```
// Amazon Corretto
IMPLEMENTOR="Amazon.com Inc."
IMPLEMENTOR_VERSION="Corretto-17.0.11.9.1"
JAVA_RUNTIME_VERSION="17.0.11+9-LTS"
JAVA_VERSION="17.0.11"
JAVA_VERSION_DATE="2024-04-16"
LIBC="default"
MODULES="java.base java.compiler java.datatransfer java.xml java.prefs java.desktop
 java.instrument java.logging java.management java.security.sasl java.naming
 java.rmi java.management.rmi java.net.http java.scripting java.security.jgss
```

```
 java.transaction.xa java.sql java.sql.rowset java.xml.crypto java.se java.smartcardio
 jdk.accessibility jdk.internal.jvmstat jdk.attach jdk.charsets jdk.compiler
 jdk.crypto.ec jdk.crypto.cryptoki jdk.dynalink jdk.internal.ed jdk.editpad
 jdk.hotspot.agent jdk.httpserver jdk.incubator.foreign jdk.incubator.vector
 jdk.internal.le jdk.internal.opt jdk.internal.vm.ci jdk.internal.vm.compiler
 jdk.internal.vm.compiler.management jdk.jartool jdk.javadoc jdk.jcmd jdk.management
 jdk.management.agent jdk.jconsole jdk.jdeps jdk.jdwp.agent jdk.jdi jdk.jfr jdk.jlink
 jdk.jpackage jdk.jshell jdk.jsobject jdk.jstatd jdk.localedata jdk.management.jfr
 jdk.naming.dns jdk.naming.rmi jdk.net jdk.nio.mapmode jdk.random jdk.sctp
 jdk.security.auth jdk.security.jgss jdk.unsupported jdk.unsupported.desktop
 jdk.xml.dom jdk.zipfs"
OS_ARCH="x86_64"
OS_NAME="Darwin"
SOURCE=".:git:7917f11551e8+"

// JDK
IMPLEMENTOR="Oracle Corporation"
JAVA_VERSION="19"
JAVA_VERSION_DATE="2022-09-20"
LIBC="default"
MODULES="java.base java.compiler java.datatransfer java.xml java.prefs java.desktop
 java.instrument java.logging java.management java.security.sasl java.naming
 java.rmi java.management.rmi java.net.http java.scripting java.security.jgss
 java.transaction.xa java.sql java.sql.rowset java.xml.crypto java.se java.smartcardio
 jdk.accessibility jdk.internal.jvmstat jdk.attach jdk.charsets jdk.zipfs jdk.compiler
 jdk.crypto.ec jdk.crypto.cryptoki jdk.dynalink jdk.internal.ed jdk.editpad
 jdk.hotspot.agent jdk.httpserver jdk.incubator.concurrent jdk.incubator.vector
 jdk.internal.le jdk.internal.opt jdk.internal.vm.ci jdk.internal.vm.compiler
 jdk.internal.vm.compiler.management jdk.jartool jdk.javadoc jdk.jcmd jdk.management
 jdk.management.agent jdk.jconsole jdk.jdeps jdk.jdwp.agent jdk.jdi jdk.jfr jdk.jlink
 jdk.jpackage jdk.jshell jdk.jsobject jdk.jstatd jdk.localedata jdk.management.jfr
 jdk.naming.dns jdk.naming.rmi jdk.net jdk.nio.mapmode jdk.random jdk.sctp
 jdk.security.auth jdk.security.jgss jdk.unsupported jdk.unsupported.desktop
 jdk.xml.dom"
OS_ARCH="x86_64"
OS_NAME="Darwin"
SOURCE=".:git:53b4a11304b0 open:git:967a28c3d85f"
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk Oracle Java rilis.

```
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
```

Jenkinspengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- Jenkins Core

> **Note**
>
> Evaluasi kerentanan berlaku untuk Jenkins versi 2.400.* dan lebih tinggi.

Fitur utama

- Mengekstrak informasi versi dari `jenkins.war` file dengan membaca `META-INF/MANIFEST.M` file, yang berisi string Jenkins versi.

Amazon Inspector SBOM Generator mencari instalasi Jenkins di jalur instalasi umum di seluruh platform:

Linux

- `/usr/share/jenkins/jenkins.war`
- /usr/share/java/jenkins.perang

macOS

- `/opt/homebrew/opt/jenkins-lts/libexec/jenkins.war`

Windows

- `/Program Files/Jenkins/Jenkins.war`
- `/Program Files (x86)/Jenkins/Jenkins.war`

Contoh file

Berikut ini adalah contoh `jenkins.war/META-INF/MANIFEST.MF` file untuk rilis yang berbeda.

```
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
```

Sampel PURLs

Berikut ini adalah paket URLs untuk versi 2.516.2 dari rilis Jenkins LTS dan versi 2.414 dari rilis server otomatisasi. Jenkins

```
LTS: pkg:generic/jenkins/jenkins-core-lts@2.516.2.1
Regular: pkg:generic/jenkins/jenkins-core@2.414
```

MariaDBdan pengumpulan MySQL ekosistem

MariaDB

Aplikasi-aplikasi yang didukung

- MariaDB Server(10.6+, 11.x, 12.x)

Fitur utama

- Mengekstrak informasi versi dari binari server database dan file header menggunakan pola khusus database.
- Mengidentifikasi jalur direktori yang berisi instalasi server database.
- Secara otomatis membedakan antara MariaDB dan MySQL instalasi menggunakan deteksi tipe file berbasis data.

Generator SBOM mencari MariaDB instalasi di jalur instalasi umum di seluruh platform:

Linux

- `/usr/bin/mariadbd`
- `/usr/sbin/mariadbd`
- `/usr/local/bin/mariadbd`

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)`

Contoh PURL

Berikut ini adalah contoh URL paket untuk MariaDB server.

```
# MariaDB Server

pkg:generic/mysql/mariadb-server@10.11.8
```

MySQLpengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- Oracle MySQL Server Server(8.0, 8.4, 9.4+)

Fitur utama

- Mengekstrak informasi versi dari binari server database dan file header menggunakan pola khusus database.
- Mengidentifikasi jalur direktori yang berisi instalasi server database.
- Secara otomatis membedakan antara MySQL dan MariaDB instalasi menggunakan deteksi tipe file berbasis data.

Generator SBOM mencari MySQL instalasi di jalur instalasi umum di seluruh platform:

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)`

Contoh PURL

Berikut ini adalah contoh URL paket untuk MySQL server.

```
# Oracle MySQL Server
```

```
pkg:generic/mysql/mysql-server@8.0.43
```

Microsoft applicationspengumpulan ekosistem

Aplikasi Microsoft yang didukung

- PowerShell
- NuGet CLI
- Visual Studio Code
- Microsoft Edge
- SharePoint Server
- Microsoft Defender
- Exchange Server
- Visual Studio
- .NET Runtime
- ASP.NET Core Runtime
- Microsoft Teams
- Outlook for Windows
- Microsoft Office
- Microsoft 365

Fitur utama

- PowerShell— Memeriksa `pwsh.exe` file untuk mengekstrak informasi versi tertanam.
- NuGet CLI— Memeriksa `nuget.exe` file untuk mengekstrak informasi versi tertanam.
- Visual Studio Code— Memeriksa `Code.exe` file untuk mengekstrak informasi versi tertanam.
- Microsoft Edge— Memeriksa `msedge.exe` file untuk mengekstrak informasi versi tertanam.
- SharePoint Server— Memeriksa `Microsoft.SharePoint.dll` file untuk mengekstrak informasi versi tertanam.
- Microsoft Defender— Memeriksa `MsMpEng.exe` file untuk mengekstrak informasi versi tertanam.
- Exhange Server— Memeriksa `Exsetup.exe` file untuk mengekstrak informasi versi tertanam.
- Visual Studio— `state.json` Mem-parsing file untuk mengambil string versi dari bidang. `catalogInfo.productDisplayVersion`

- .NET Runtime— Mencari `Microsoft.NETCore.App.deps.json` file di jalur instalasi dan mengekstrak string versi dari pola jalur file berikut.

```
Microsoft.NETCore.App/<VERSION>/Microsoft.NETCore.App.deps.json
```

- ASP.NET Runtime— Mencari `Microsoft.AspNetCore.App.deps.json` file di jalur instalasi dan mengekstrak string versi dari pola jalur file berikut.

```
Microsoft.AspNetCore.App/<VERSION>/Microsoft.AspNetCore.App.deps.json
```

- Outlook for Windows— Mem-parsing Windows Registry, dan mengekstrak versi dari kunci registri berikut.

```
HKLM\SOFTWARE\Classes\Local Settings\Software\Microsoft
\Windows\CurrentVersion\AppModel\PackageRepository\Packages
\Microsoft.OutlookForWindows_<VERSION>_<ARCH>__8wekyb3d8bbwe
```

- Microsoft Teams— Mem-parsing Windows Registry, dan mengekstrak versi dari kunci registri berikut.

```
HKLM\SOFTWARE\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion
\AppModel\PackageRepository\Packages\MSTeams_<VERSION>_<ARCH>__8wekyb3d8bbwee
```

- Microsoft Office 365 / Microsoft 365— Parsing Windows Registry, dan ekstrak versi dari kunci registri berikut dan nilai.
  - Kunci Registri

    ```
    KEY_LOCAL_MACHINES\SOFTWARE\Microsoft\Office\ClickToRun\Configuration
    ```

  - Nilai Registri
    - VersionToReport— Versi Microsoft Office
    - ProductReleaseIds— Daftar produk IDs. Ini digunakan untuk mengidentifikasi produk Office yang diinstal. Untuk informasi lebih lanjut tentang produk IDs, lihat product IDsdi Microsoft situs web.
- Microsoft Office Suite— Mengumpulkan setiap aplikasi Office yang diinstal dengan memeriksa file yang dapat dieksekusi berikut:
  - `EXCEL.EXE` – Microsoft Excel
  - `WINWORD.EXE` – Microsoft Word

- `POWERPNT.EXE` – Microsoft PowerPoint
- `OUTLOOK.EXE` – Microsoft Outlook

Nomor versi di Windows Registry digunakan sebagai nomor versi otoritatif untuk setiap aplikasi Office yang diinstal.

Contoh file **`state.json`**

Berikut ini adalah contoh `state.json` file yang akan digunakan untuk mengumpulkan Visual Studio versi yang diinstal.

```
{
    "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
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk masing-masingMicrosoft 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 Runtime
Sample PURL: pkg:generic/microsoft/dotnet@8.0.18

// 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=O365HomePremRetail

// 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
```

Nginxpengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- Nginx

Platform yang didukung

Berikut ini adalah platform yang didukung.

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 adalah versi arbitrer)

macOS

- /usr/local/etc/nginx/nginx

Fitur utama

Koleksi ini memeriksa binari untuk mengekstrak informasi versi tertanam. Ini mencari string versi di bagian biner yang dapat dieksekusi (untuk binari ELF aktif), `.rodata` bagian (untuk binari PE aktifLinux), atau `.rdata` bagian (untuk binariWindows). `__ctring` MachO

Contoh string versi

Berikut ini adalah contoh dari string versi tertanam dalam Nginx biner.

```
nginx version: nginx/1.27.5
```

Versi `1.27.5` diekstraksi untuk mengidentifikasi Nginx versi.

Contoh PURL

Berikut ini adalah contoh URL paket untuk Nginx.

```
Sample PURL: pkg:generic/nginx/nginx@1.27.5
```

Node.JSkoleksi runtime

Aplikasi-aplikasi yang didukung

- biner runtime node untuk Node.JS

Platform yang didukung

Berikut ini adalah platform yang didukung. (* adalah versi arbitrer)

Linux

- /usr/local/bin/node
- /usr/bin/node
- /nodejs/bin/node
- ~/.nvm/versions/node/*/bin/node
- ~/.local/share/fnm/node-versions/*/installation/bin/node
- ~/.asdf/installs/nodejs/*/bin/node
- ~/.local/share/mise/installs/node/*/bin/node
- ~/.volta/tools/image/node/*/bin/node

Windows

- C:\Program File\nodejs\node.exe

- C:\Program File (x86)\nodejs\node.exe
- ~\AppData\ Roaming\ fnm\node-versi\ *\ instalasi\node.exe

macOS

- /opt/homebrew/Cellar/node/*/bin/node

Fitur utama

Koleksi ini memeriksa binari untuk mengekstrak informasi versi tertanam. Ini mencari string versi di bagian biner yang dapat dieksekusi (untuk binari ELF aktif), `.rodata` bagian (untuk binari PE aktifLinux), atau `.rdata` bagian (untuk binariWindows). `__ctring` MachO

Contoh string versi

Berikut ini adalah contoh string versi yang disematkan dalam biner Node.JS runtime.

```
node.js/v24.11.1
```

Versi `24.11.1` diekstraksi untuk mengidentifikasi versi Node.JS runtime.

Contoh PURL

Berikut ini adalah contoh URL paket untukNode.JS.

```
Sample PURL: pkg:generic/nodejs/node@24.11.1
```

Koleksi ekosistem OpenSSH

Aplikasi-aplikasi yang didukung

- OpenSSH(Versi 9)
- OpenSSH(Versi 10)

Platform yang didukung Linux/macOS

- `/usr/sbin/sshd`
- `/usr/local/sbin/sshd`

Platform yang didukung Windows

- `C:/Windows/System32/OpenSSH/sshd.exe`
- `C:/Program Files/OpenSSH/sshd.exe`
- `C:/Program Files (x86)/OpenSSH/sshd.exe`
- `C:/OpenSSH/sshd.exe`

Fitur utama

- Memeriksa `sshd` binari untuk mengekstrak informasi verion yang disematkan.
- Mencari string versi di bagian biner yang dapat dieksekusi (untuk binari ELF padaLinux, `.rodata` bagian (untuk binari Mach-O aktifMacOs), atau `__cstring` `.rdata` bagian (untuk binari PE aktif). Windows

Contoh string versi

Berikut ini adalah contoh dari string versi tertanam dalam OpenSSH biner.

```
OpenSSH_9.9p2
```

Versi `9.9p2` diekstraksi untuk mengidentifikasi OpenSSH versi.

Contoh PURL

Berikut ini adalah contoh URL paket untukOpenSSH.

```
Sample PURL: pkg:generic/openssh/openssh@9.9p2
```

Koleksi ekosistem OpenSSL

Aplikasi-aplikasi yang didukung

Support untuk pustaka OpenSSL dan paket pengembangan terbatas pada perangkat lunak yang dibangun dengan OpenSSL resmi untuk rilis 3.0.0 dan di atasnya. Perangkat lunak ini juga harus mengikuti versi semantik. Varian dan versi OpenSSL khusus atau bercabang yang lebih rendah dari 3.0.0 tidak didukung.

Amazon Inspector SBOM Generator mengekstrak informasi paket kunci untuk setiap instance OpenSSL yang diinstal.

Fitur utama

- Mengekstrak string versi SEMVER dasar dari file header OpenSSL
- Mengidentifikasi jalur direktori yang berisi instalasi OpenSSL

Amazon Inspector SBOM Generator mencari instalasi OpenSSL dengan memindai file di jalur instalasi umum di `openssl.h` seluruh platform.

Contoh jalur instalasi untuk Linux/Unix

Berikut ini adalah contoh jalur instalasi untuk 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 mengekstrak informasi versi dengan mengurai `openssl.h` file dan mencari definisi versi.

```
# define OPENSSL_VERSION_MAJOR  3
# define OPENSSL_VERSION_MINOR  4
# define OPENSSL_VERSION_PATCH  0
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk versi OpenSSL.

```
Sample PURL: pkg:generic/openssl/openssl@3.4.0
```

Koleksi Server Database Oracle

Aplikasi-aplikasi yang didukung

- Oracle Database

Platform yang didukung Linux

- `/opt/oracle`

- `/u01/app/oracle`

> **Note**
>
> Evaluasi kerentanan hanya berlaku untuk Oracle Database Server versi 19 dan lebih tinggi.

Fitur utama

- Memeriksa Oracle binari untuk mengekstrak informasi versi tertanam.
- Mencari string versi di `.rodata` bagian biner yang dapat dieksekusi (untuk binari ELF aktif). Linux
- Informasi versi mengikuti format tertentu yang mencakup string versi RDBMS.

Contoh string versi

Berikut ini adalah contoh string versi yang disematkan dalam Oracle Database biner:

```
RDBMS_23.7.0.25.01DBRU_LINUX.X64_240304
```

Versi `23.7.0.25.01` diekstraksi untuk mengidentifikasi Oracle Database versi.

Contoh PURL

Berikut ini adalah contoh URL paket untukOracle Database.

```
Sample PURL: pkg:generic/oracle/database@23.7.0.25.01
```

PHPpengumpulan ekosistem

Aplikasi-aplikasi yang didukung

- PHP(versi 8.1 dan lebih tinggi)

Fitur utama

- Mengekstrak informasi versi dari executable PHP biner menggunakan string versi tertanam.
- Mengidentifikasi jalur direktori yang berisi PHP biner.

- Secara otomatis mendeteksi PHP binari standar dan instalasi berversi, seperti,, dan. `php8.1` `php8.2` `php8.3`

Amazon Inspector SBOM Generator mencari PHP instalasi di jalur instalasi umum di seluruh platform:

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`(binari berversi)

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`(direktori berversi)

Contoh ekstraksi PHP versi

Amazon Inspector SBOM Generator mengekstrak informasi versi dari PHP binari dengan mencari string versi tertanam menggunakan pola berikut.

```
X-Powered-By: PHP/8.4.12
```

`8.4.12`diekstraksi dari pola ini untuk mengidentifikasi PHP versinya.

Contoh PURL

Berikut ini adalah contoh URL paket untuk sebuah PHP pola.

```
pkg:generic/php/php@8.4.12
```

WordPresspengumpulan ekosistem

Komponen yang didukung

- WordPressinti
- WordPressplugin
- WordPresstema

Fitur utama

- WordPresscore - mem-parsing `/wp-includes/version.php` file untuk mengekstrak nilai versi dari variabel $wp_version.
- WordPressplugin — mem-parsing `/wp-content/plugins/<WordPress Plugin>/readme.txt` file atau `/wp-content/plugins/<WordPress Plugin>/readme.md` file untuk mengekstrak `Stable` tag sebagai string versi.
- WordPresstema — mem-parsing `/wp-content/themes/<WordPress Theme>/style.css` file untuk mengekstrak versi dari metadata versi.

Contoh file **`version.php`**

Berikut ini adalah contoh `version.php` file WordPress inti.

```
// 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
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk WordPress inti.

```
Sample PURL: pkg:generic/wordpress/core/wordpress@6.5.5
```

Contoh file **`readme.txt`**

Berikut ini adalah contoh `readme.txt` file 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
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk sebuah WordPress plugin.

```
Sample PURL: pkg:generic/wordpress/plugin/exclusive-addons-for-elementor@1.0.0
```

Contoh file **`style.css`**

Berikut ini adalah contoh `style.css` file WordPress tema.

```
/*
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
*/
```

Contoh PURL

Berikut ini adalah contoh URL paket untuk WordPress tema.

```
Sample PURL: pkg:generic/wordpress/theme/avada@1.0.0
```

Pemindaian sertifikat Amazon Inspector SBOM Generator SSL/TLS

Bagian ini menjelaskan cara menggunakan Amazon Inspector SBOM Generator untuk inventaris sertifikat. SSL/TLS SbomgenInventaris SSL/TLS sertifikat dengan mencari sertifikat di lokasi yang telah ditentukan serta direktori yang disediakan oleh pengguna. Fitur ini dimaksudkan untuk memungkinkan pengguna untuk inventaris SSL/TLS sertifikat serta mengidentifikasi sertifikat kedaluwarsa. Sertifikat CA juga akan muncul di inventaris keluaran.

Menggunakan pemindaian Sbomgen sertifikat

Anda dapat mengaktifkan pengumpulan inventaris SSL/TLS sertifikat menggunakan `--scanners certificates` argumen. Pemindaian sertifikat dapat dikombinasikan dengan pemindai lainnya. Secara default, pemindaian sertifikat tidak diaktifkan.

SbomgenPencarian lokasi yang berbeda untuk sertifikat tergantung pada artefak yang dipindai. Dalam semua kasus, Sbomgen upaya untuk mengekstrak sertifikat dalam file dengan ekstensi berikut.

```
.pem
.crt
.der
.p7b
.p7m
.p7s
.p12
.pfx
```

Jenis artefak localhost

Jika pemindai sertifikat diaktifkan dan jenis artefak adalah localhost, Sbomgen secara rekursif mencari sertifikat di`/etc/*/ssl`,,, dan `/opt/*/ssl/certs` `/usr/local/*/ssl/var/lib/*/certs`, where * tidak kosong. Direktori yang disediakan pengguna akan dicari secara rekursif, terlepas dari direktori apa yang dinamai. Biasanya, CA/system sertifikat tidak ditempatkan di jalur ini. Sertifikat ini sering dalam folder bernama`pki`,`ca-certs`, atau`CA`. Mereka juga dapat muncul di jalur pemindaian localhost default.

Direktori dan artefak kontainer

Saat memindai direktori atau artefak kontainer, Sbomgen pencarian sertifikat terletak di mana saja pada artefak.

Contoh perintah pemindaian sertifikat

Berikut ini berisi contoh perintah pemindaian sertifikat. Satu menghasilkan SBOM yang hanya berisi sertifikat di direktori lokal. Lain menghasilkan SBOM yang berisi sertifikat dan`Alpine`,`Debian`, dan `Rhel` paket dalam direktori lokal. Lain menghasilkan SBOM yang berisi sertifikat ditemukan di lokasi sertifikat umum.

```
# 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
```

Contoh komponen file

Berikut ini berisi dua contoh komponen pencarian sertifikat. Ketika sertifikat kedaluwarsa, Anda dapat melihat properti tambahan yang mengidentifikasi tanggal kedaluwarsa.

```
{
      "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"
            }
      ]
```

```
}
```

Contoh komponen respons kerentanan

Menjalankan Amazon Inspector SBOM Generator dengan `--scan-sbom` flag mengirimkan SBOM yang dihasilkan ke Amazon Inspector untuk pemindaian kerentanan. Berikut ini adalah contoh penemuan sertifikat untuk komponen respons kerentanan.


```
{
    "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"
}
```

Koleksi lisensi Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator membantu melacak informasi lisensi dalam tagihan bahan perangkat lunak (SBOM). Ini mengumpulkan informasi lisensi dari paket yang didukung di seluruh sistem operasi dan bahasa pemrograman. Dengan ekspresi lisensi standar dalam SBOM yang Anda hasilkan, Anda dapat memahami kewajiban lisensi Anda.

Kumpulkan informasi lisensi

Perintah contoh

Contoh berikut menunjukkan cara mengumpulkan informasi lisensi dari direktori.

```
./inspector-sbomgen directory --path /path/to/your/directory/ --collect-licenses
```

Contoh komponen SBOM

Contoh berikut menunjukkan entri komponen dalam SBOM dihasilkan.

```
"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",
    }
  ]
```

Paket yang didukung

Bahasa pemrograman berikut dan paket sistem operasi didukung untuk pengumpulan lisensi.

Target	Manajer Package	Sumber informasi lisensi	Jenis
Alma Linux	RPM	• /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	OS
Amazon Linux	RPM	• /usr/lib/sysimage/rpm/rpmdb.sqlite • /usr/lib/sysimage/rpm/Packages	OS

Target	Manajer Package	Sumber informasi lisensi	Jenis
		• /usr/lib/sysimage/rpm/Packages.db • /var/lib/rpm/rpmdb.sqlite • /var/lib/rpm/Packages • /var/lib/rpm/Packages.db	
CentOS	RPM	• /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	OS

Target	Manajer Package	Sumber informasi lisensi	Jenis
Fedora	RPM	• /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	OS
OpenSUSE	RPM	• /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	OS

Target	Manajer Package	Sumber informasi lisensi	Jenis
Oracle Linux	RPM	• /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	OS
Photon OS	RPM	• /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	OS

Target	Manajer Package	Sumber informasi lisensi	Jenis
RHEL	RPM	• /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	OS
Rocky Linux	RPM	• /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	OS

Target	Manajer Package	Sumber informasi lisensi	Jenis
SLES	RPM	• /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	OS
Alpine Linux	APK	`/lib/apk/db/installed`	OS
Chainguard	APK	`/lib/apk/db/installed`	OS
Debian	DPKG	`/usr/share/doc/*/copyright`	OS
Ubuntu	DPKG	`/usr/share/doc/*/copyright`	OS
Node.js	Javascript	`node_modules/*/package.json`	Bahasa pemrograman
PHP	Komposer Paket	• `composer.lock` • `/vendor/composer/installed.json`	Bahasa pemrograman

Target	Manajer Package	Sumber informasi lisensi	Jenis
Go	Go	`LICENSE`	Bahasa pemrograman
Python	Python/Egg/Wheel	• `.dist-info/METADATA` • `.egg-info` • `.egg-info/PKG-INFO`	Bahasa pemrograman
Ruby	RubyGem	`*.gemspec`	Bahasa pemrograman
Rust	crate	`Cargo.toml`	Bahasa pemrograman

Standardisasi ekspresi lisensi

Format ekspresi lisensi SPDX memberikan representasi akurat dari istilah lisensi yang ditemukan dalam perangkat lunak sumber terbuka. Amazon Inspector SBOM Generator menstandarisasi semua informasi lisensi ke dalam ekspresi lisensi SPDX melalui aturan yang dijelaskan di bagian ini. Aturan memberikan konsistensi dan kompatibilitas di seluruh informasi perizinan.

Pemetaan pengenal bentuk pendek SPDX

Semua nama lisensi dipetakan ke pengidentifikasi formulir pendek SPDX. Sebagai contoh, `MIT License` disingkat menjadi`MIT`.

Beberapa kombinasi lisensi

Anda dapat menggabungkan lebih dari satu lisensi dengan `AND` operator. Berikut ini adalah contoh perintah yang menunjukkan cara memformat perintah Anda.

```
MIT AND Apache-2.0
```

Awalan lisensi kustom

Lisensi khusus diawali dengan`LicenseRef`, seperti. `LicenseRef-CompanyPrivate`

Awalan pengecualian kustom

Pengecualian khusus diawali dengan`AdditionRef-`, seperti. `AdditionRef-CustomException`

Apa itu URL paket?

URL paket atau PURL adalah format standar yang digunakan untuk mengidentifikasi paket perangkat lunak, komponen, dan perpustakaan di berbagai sistem manajemen paket. Format ini memudahkan untuk melacak, menganalisis, dan mengelola dependensi dalam proyek perangkat lunak, terutama saat membuat Software Bill of Materials ()SBOMs.

Struktur PURL

Struktur PURL mirip dengan URL dan terdiri dari beberapa komponen:

- `pkg`— Awalan literal
- `type`— Jenis paket
- `namespace`— Pengelompokan
- `name`— Nama paket
- `version`— Versi paket
- `qualifiers`— Pasangan nilai kunci ekstra
- `subpath`— Filepath dalam paket

Contoh PURL

Berikut ini adalah contoh bagaimana PURL mungkin terlihat.

```
pkg:<type>/<namespace>/<name>@<version>?<qualifiers>#<subpath>
```

PURL generik

PURL generik digunakan untuk mewakili paket perangkat lunak dan komponen yang tidak sesuai dengan ekosistem paket yang sudah mapan, sepertinpm,, atau. pypi maven Ini mengidentifikasi komponen perangkat lunak dan menangkap metadata yang mungkin tidak selaras dengan sistem manajemen paket tertentu. PURL generik berguna untuk berbagai proyek perangkat lunak, dari binari yang dikompilasi hingga platform, seperti dan. Apache WordPress Ini memungkinkannya untuk

diterapkan di berbagai kasus penggunaan, termasuk binari yang dikompilasi, platform web, dan distribusi perangkat lunak khusus.

Kasus penggunaan kunci

- Mendukung binari yang dikompilasi dan berguna untuk Go dan Rust
- Mendukung platform web, seperti Apache danWordPress, di mana paket mungkin tidak terkait dengan manajer paket tradisional.
- Mendukung perangkat lunak warisan kustom dengan memungkinkan organisasi untuk referensi perangkat lunak atau sistem yang dikembangkan secara internal yang tidak memiliki paket formal.

Contoh format

Berikut ini adalah contoh format PURL generik.

```
pkg:generic/<namespace>/<name>@<version>?<qualifiers>
```

Contoh tambahan dari format PURL generik

Berikut ini adalah contoh tambahan dari format PURL generik.

GoBiner yang dikompilasi

Berikut ini mewakili yang `inspector-sbomgen binary` dikompilasi dengan aGo.

```
pkg:generic/inspector-sbomgen?go_toolchain=1.22.5
```

RustBiner yang dikompilasi

Berikut ini merupakan `myrustapp` biner yang dikompilasi denganRust.

```
pkg:generic/myrustapp?rust_toolchain=1.71.0
```

Proyek Apache

Berikut ini mengacu pada proyek http di bawah Apache namespace.

```
pkg:generic/apache/httpd@1.0.0
```

WordPressperangkat lunak

Berikut ini mengacu pada WordPress perangkat lunak inti.

```
pkg:generic/wordpress/core/wordpress@6.0.0
```

WordPresstema

Berikut ini mengacu pada WordPress tema khusus.

```
pkg:generic/wordpress/theme/mytheme@1.0.0
```

WordPress plugin

Berikut ini mengacu pada WordPress plugin khusus.

```
pkg:generic/wordpress/plugin/myplugin@1.0.0
```

Menangani referensi versi yang belum terselesaikan atau tidak standar di Amazon Inspector SBOM Generator

Amazon Inspector SBOM Generator menemukan dan mem-parsing artefak yang didukung dalam sistem dengan mengidentifikasi dependensi langsung dari file sumber. Ini bukan manajer paket dan tidak menyelesaikan rentang versi, menyimpulkan versi berdasarkan referensi dinamis, atau menangani pencarian registri. Ini mengumpulkan dependensi hanya karena mereka didefinisikan dalam artefak sumber proyek. Dalam banyak kasus, dependensi dalam manifes paket, seperti,, atau `package.json` `pom.xml` `requirements.txt`, ditentukan menggunakan versi yang belum terselesaikan atau berbasis rentang. Topik ini mencakup contoh bagaimana dependensi ini mungkin terlihat.

Rekomendasi

Amazon Inspector SBOM Generator mengekstrak dependensi dari artefak sumber, tetapi tidak menyelesaikan atau menafsirkan rentang versi atau referensi dinamis. Untuk pemindaian kerentanan yang lebih akurat dan SBOMs, sebaiknya gunakan pengidentifikasi versi semantik yang diselesaikan dalam dependensi proyek.

Java

UntukJava, Maven proyek dapat menggunakan rentang versi untuk menentukan dependensi dalam file. `pom.xml`

```
<dependency>
    <groupId>org.inspector</groupId>
    <artifactId>inspector-api</artifactId>
    <version>(,1.0]</version>
</dependency>
```

Rentang menentukan bahwa versi apa pun hingga dan termasuk 1.0 dapat diterima. Namun, jika versi bukan versi yang diselesaikan, Amazon Inspector SBOM Generator tidak akan mengumpulkannya karena tidak dapat dipetakan ke rilis tertentu.

JavaScript

UntukJavaScript, `package.json` file dapat menyertakan rentang versi yang menyerupai berikut ini:

```
"dependencies": {
    "ky": "^1.2.0",
    "registry-auth-token": "^5.0.2",
    "registry-url": "^6.0.1",
    "semver": "^7.6.0"
}
```

^Operator menentukan bahwa versi apa pun yang lebih besar dari atau sama dengan versi yang ditentukan dapat diterima. Namun, jika versi yang ditentukan bukan versi yang diselesaikan, Amazon Inspector SBOM Generator tidak akan mengumpulkannya karena hal itu dapat menyebabkan positif palsu selama deteksi kerentanan.

Python

UntukPython, `requirements.txt` file dapat menyertakan entri dengan ekspresi boolean.

```
requests>=1.0.0
```

>=Operator menentukan bahwa versi apa pun yang lebih besar dari atau sama dengan dapat `1.0.0` diterima. Karena ekspresi khusus ini tidak menentukan versi yang tepat, Amazon Inspector SBOM Generator tidak dapat mengumpulkan versi untuk analisis kerentanan dengan andal.

Amazon Inspector SBOM Generator tidak mendukung pengidentifikasi versi non-standar atau ambigu, seperti beta, terbaru, atau snapshot.

```
pkg:maven/org.example.com/testmaven@1.0.2%20Beta-RC-1_Release
```

> **Note**
>
> Penggunaan akhiran non-standar, seperti, tidak sesuai dengan versi semantik standar dan tidak dapat dinilai untuk kerentanan dalam mesin deteksi Amazon Inspector. Beta-RC-1_Release

Menggunakan CycloneDX ruang nama dengan Amazon Inspector

Amazon Inspector memberi Anda CycloneDX ruang nama dan nama properti yang dapat Anda gunakan. SBOMs Bagian ini menjelaskan semua key/value properti kustom yang mungkin ditambahkan ke komponen di CycloneDX SBOMs. Untuk informasi lebih lanjut, lihat Taksonomi properti CycloneDX di situs web. GitHub

`amazon:inspector:sbom_scanner` taksonomi namespace

Amazon Inspector Scan API menggunakan `amazon:inspector:sbom_scanner` namespace dan memiliki properti berikut:

Properti	Deskripsi
`amazon:inspector:sbom_scanner:cisa_kev_date_added`	Menunjukkan kapan kerentanan ditambahkan ke katalog CISA Known Exploited Vulnerabilities.
`amazon:inspector:sbom_scanner:cisa_kev_date_due`	Menunjukkan kapan perbaikan kerentanan jatuh tempo sesuai dengan katalog CISA Known Exploited Vulnerabilities.

Properti	Deskripsi
`amazon:inspector:sbom_scanner:critical_vulnerabilities`	Hitungan jumlah total kerentanan keparahan kritis yang ditemukan di SBOM.
`amazon:inspector:sbom_scanner:exploit_available`	Menunjukkan apakah eksploitasi tersedia untuk kerentanan yang diberikan.
`amazon:inspector:sbom_scanner:exploit_last_seen_in_public`	Menunjukkan kapan eksploitasi terakhir terlihat di depan umum untuk kerentanan yang diberikan.
`amazon:inspector:sbom_scanner:fixed_version: component_bom_ref`	Menyediakan versi tetap dari komponen yang ditunjukkan untuk kerentanan yang diberikan.
`amazon:inspector:sbom_scanner:high_vulnerabilities`	Hitungan jumlah total kerentanan tingkat keparahan tinggi yang ditemukan di SBOM.
`amazon:inspector:sbom_scanner:info`	Menyediakan konteks pemindaian untuk komponen tertentu, misalnya: “Komponen dipindai: tidak ada kerentanan yang ditemukan .”
`amazon:inspector:sbom_scanner:is_malicious`	Menunjukkan jika OpenSSF mengident ifikasi komponen yang terpengaruh sebagai berbahaya.
`amazon:inspector:sbom_scanner:low_vulnerabilities`	Hitungan jumlah total kerentanan tingkat keparahan rendah yang ditemukan di SBOM.
`amazon:inspector:sbom_scanner:medium_vulnerabilities`	Hitungan jumlah total kerentanan tingkat keparahan sedang yang ditemukan di SBOM.
`amazon:inspector:sbom_scanner:path`	Jalur ke file yang menghasilkan informasi paket subjek.

Properti	Deskripsi
`amazon:inspector:sbom_scanner:priority`	Prioritas yang disarankan untuk memperbaiki kerentanan yang diberikan. Nilai dalam urutan menurun adalah “SEGERA”, “URGENT”, “MODERATE”, dan “STANDARD”.
`amazon:inspector:sbom_scanner:priority_intelligence`	Kualitas kecerdasan yang digunakan untuk menentukan prioritas kerentanan tertentu. Nilainya termasuk “TERVERIFIKASI” atau “TIDAK DIVERIFIKASI”.
`amazon:inspector:sbom_scanner:warning`	Memberikan konteks mengapa komponen tertentu tidak dipindai, misalnya: “Komponen dilewati: tidak ada purl yang disediakan.”

`amazon:inspector:sbom_generator` taksonomi namespace

Amazon Inspector SBOM Generator menggunakan `amazon:inspector:sbom_generator` namespace dan memiliki properti berikut:

Properti	Deskripsi
`amazon:inspector:sbom_generator:cpu_architecture`	Arsitektur CPU dari sistem yang diinventarisasi (x86_64).
`amazon:inspector:sbom_generator:ec2:instance_id`	ID EC2 instans Amazon.
`amazon:inspector:sbom_generator:ec2:instance_type`	Jenis EC2 Instans Amazon
`amazon:inspector:sbom_generator:live_patching_enabled`	Nilai boolean yang menunjukkan apakah penambalan langsung diaktifkan di Amazon Amazon EC2 . Linux

Properti	Deskripsi
`amazon:inspector:sbom_generator:live_patched_cves`	Daftar CVEs tambalan melalui penambalan langsung di Amazon Amazon EC2 . Linux
`amazon:inspector:sbom_generator:dockerfile_finding: inspector_finding_id`	Menunjukkan bahwa temuan Amazon Inspector dalam komponen terkait Dockerfile dengan pemeriksaan.
`amazon:inspector:sbom_generator:image_id`	Hash milik file konfigurasi gambar kontainer (juga dikenal sebagai ID Gambar).
`amazon:inspector:sbom_generator:image_arch`	Arsitektur gambar kontainer.
`amazon:inspector:sbom_generator:image_author`	Penulis gambar kontainer.
`amazon:inspector:sbom_generator:image_docker_version`	Versi docker digunakan untuk membangun image container.
`amazon:inspector:sbom_generator:is_duplicate_package`	Menunjukkan bahwa paket subjek ditemukan oleh lebih dari satu pemindai file.
`amazon:inspector:sbom_generator:duplicate_purl`	Menunjukkan paket duplikat PURL ditemukan oleh pemindai lain.
`amazon:inspector:sbom_generator:kernel_name`	Nama kernel dari sistem yang sedang diinventa risasi.
`amazon:inspector:sbom_generator:kernel_version`	Versi kernel dari sistem yang sedang diinventa risasi.
`amazon:inspector:sbom_generator:kernel_component`	Nilai boolean yang menunjukkan apakah paket subjek adalah komponen kernel
`amazon:inspector:sbom_generator:running_kernel`	Nilai boolean yang menunjukkan apakah paket subjek adalah kernel yang sedang berjalan

Properti	Deskripsi
`amazon:inspector:sbom_generator:layer_diff_id`	Hash dari layer gambar kontainer yang tidak terkompresi.
`amazon:inspector:sbom_generator:replaced_by`	Nilai yang menggantikan Go modul saat ini.
`amazon:inspector:sbom_generator:os_hostname`	Nama host dari sistem yang sedang diinventa risasi.
`amazon:inspector:sbom_generator:source_file_scanner`	Pemindai yang menemukan file yang berisi informasi paket, misalnya:`/var/lib/dpkg/status` .
`amazon:inspector:sbom_generator:source_package_collector`	Kolektor yang mengekstrak nama paket dan versi dari file tertentu.
`amazon:inspector:sbom_generator:source_path`	Jalur ke file tempat informasi paket subjek diekstraksi.
`amazon:inspector:sbom_generator:file_size_bytes`	Menunjukkan ukuran file dari artefak yang diberikan.
`amazon:inspector:sbom_generator:unresolved_version`	Menunjukkan string versi yang belum diselesai kan oleh manajer paket..
`amazon:inspector:sbom_generator:experimental:transitive_dependency`	Menunjukkan dependensi tidak langsung dari manajer paket.
`amazon:inspector:sbom_generator:metadata:host:hostname`	Nama host dari sistem yang dipindai.
`amazon:inspector:sbom_generator:metadata:host:kernel_name`	Nama kernel dari sistem operasi (misalnya, Linux, Darwin, Windows_NT).

Properti	Deskripsi
`amazon:inspector:sbom_generator:metadata:host:kernel_version`	String versi kernel dari sistem operasi.
`amazon:inspector:sbom_generator:metadata:host:cpu_architecture`	Arsitektur CPU sistem (misalnya, x86_64, arm64).
`amazon:inspector:sbom_generator:metadata:host:bootdisk_id`	Pengidentifikasi unik dari disk boot.
`amazon:inspector:sbom_generator:metadata:host:boot_id`	Pengidentifikasi unik untuk sesi boot saat ini.
`amazon:inspector:sbom_generator:metadata:host:boot_time`	Waktu boot sistem dalam format ISO 8601.
`amazon:inspector:sbom_generator:metadata:host:system_id`	Pengidentifikasi sistem persisten (id mesin di Linux, MachineGuid di Windows).
`amazon:inspector:sbom_generator:metadata:host:system_serial`	Nomor seri perangkat keras dari firmware sistem.
`amazon:inspector:sbom_generator:metadata:host:network_interfaces:` *`name`*`:hardware`	Alamat MAC dari antarmuka jaringan.
`amazon:inspector:sbom_generator:metadata:host:network_interfaces:` *`name`*`:ipv4`	IPv4 alamat (es) ditugaskan ke antarmuka.
`amazon:inspector:sbom_generator:metadata:host:network_interfaces:` *`name`*`:ipv6`	IPv6 alamat (es) ditugaskan ke antarmuka.

Properti	Deskripsi
`amazon:inspector:sbom_generator:metadata:host:sbomgen_tag: key`	Tag khusus yang ditentukan pengguna diteruskan melalui argumen CLI --tag.
`amazon:inspector:sbom_generator:metadata:imds:provider`	Penyedia cloud terdeteksi melalui IMDS (aws, azure).
`amazon:inspector:sbom_generator:metadata:imds:instance_id`	ID EC2 instans Amazon atau nama Azure VM.
`amazon:inspector:sbom_generator:metadata:imds:instance_type`	Jenis instance (misalnya, t3.micro, Standard_D2S_v3).
`amazon:inspector:sbom_generator:metadata:imds:instance_location`	Contoh. region/location
`amazon:inspector:sbom_generator:metadata:imds:instance_partition`	Partisi cloud (aws, aws-cn, aws-us-gov for AWS, atau AzurePublicCloud untuk Azure).
`amazon:inspector:sbom_generator:metadata:imds:instance_managed_id`	ID instans terkelola Amazon EC2 Systems Manager (AWS hanya).
`amazon:inspector:sbom_generator:metadata:imds:tenant_id`	ID penyewa Azure (hanya Azure).
`amazon:inspector:sbom_generator:metadata:imds:vm_id`	Pengidentifikasi unik Azure VM (hanya Azure).
`amazon:inspector:sbom_generator:metadata:host:open_port: port:protocol`	Menunjukkan port terbuka dari sumber daya runtime (mis. EC2)

Properti	Deskripsi
`amazon:inspector:sbom_generator:hardened_image:vendor`	Vendor gambar kontainer yang diperkeras

Mengintegrasikan pemindaian Amazon Inspector ke dalam pipeline Anda CI/CD

CI/CD Integrasi Amazon Inspector menggunakan Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan kerentanan untuk gambar kontainer. Amazon Inspector SBOM Generator membuat tagihan bahan perangkat lunak (SBOM) untuk arsip, gambar kontainer, direktori, sistem lokal, dan kompilasi dan binari. Go Rust Amazon Inspector Scan API memindai SBOM untuk membuat laporan dengan detail tentang kerentanan yang terdeteksi.
Anda dapat mengintegrasikan pemindaian gambar penampung Amazon Inspector dengan CI/CD pipeline untuk memindai kerentanan perangkat lunak dan menghasilkan laporan kerentanan, yang memungkinkan Anda menyelidiki dan memulihkan risiko sebelum penerapan. Untuk mengatur CI/CD integrasi, Anda dapat menggunakan plugin atau membuat CI/CD integrasi khusus menggunakan Amazon Inspector SBOM Generator dan Amazon Inspector Scan API.

Topik

Integrasi plugin

Amazon Inspector menyediakan plugin untuk solusi yang didukung. CI/CD Anda dapat menginstal plugin ini dari pasar masing-masing dan kemudian menggunakannya untuk menambahkan Amazon Inspector Scan sebagai langkah pembuatan dalam pipeline Anda. Langkah pembuatan plugin

menjalankan generator Amazon Inspector SBOM pada gambar yang Anda berikan, dan kemudian menjalankan Amazon Inspector Scan API pada SBOM yang dihasilkan.

Berikut ini adalah ikhtisar tentang cara kerja CI/CD integrasi Amazon Inspector melalui plugin:

1. Anda mengonfigurasi Akun AWS untuk mengizinkan akses ke Amazon Inspector Scan API. Untuk petunjuk, lihat Menyiapkan AWS akun untuk menggunakan integrasi Amazon Inspector CI/CD.
2. Anda menginstal plugin Amazon Inspector dari marketplace.
3. Anda menginstal dan mengkonfigurasi biner Amazon Inspector SBOM Generator. Untuk petunjuk, lihat Amazon Inspector SBOM Generator.
4. Anda menambahkan Amazon Inspector Scan sebagai langkah build di CI/CD pipeline Anda dan mengonfigurasi pemindaian.
5. Saat Anda menjalankan build, plugin mengambil image container Anda sebagai input dan kemudian menjalankan Amazon Inspector SBOM Generator pada image untuk menghasilkan SBOM yang CycloneDX kompatibel.
6. Dari sana, plugin mengirimkan SBOM yang dihasilkan ke titik akhir Amazon Inspector Scan API yang menilai setiap komponen SBOM untuk kerentanan.
7. Respons API Amazon Inspector Scan diubah menjadi laporan kerentanan dalam format CSV, SBOM JSON, dan HTML. Laporan tersebut berisi rincian tentang kerentanan apa pun yang ditemukan Amazon Inspector.

CI/CD Solusi yang didukung

Amazon Inspector saat ini mendukung solusi berikut CI/CD . Untuk instruksi lengkap tentang pengaturan CI/CD integrasi menggunakan plugin, pilih plugin untuk solusi CI/CD Anda:

- Plugin Jenkins
- TeamCity plugin
- GitHub tindakan

Integrasi kustom

Jika Amazon Inspector tidak menyediakan plugin untuk CI/CD solusi Anda, Anda dapat membuat CI/CD integrasi kustom Anda sendiri menggunakan kombinasi Amazon Inspector SBOM Generator dan Amazon Inspector Scan API. Anda juga dapat menggunakan integrasi khusus untuk

menyempurnakan pemindaian menggunakan opsi yang tersedia melalui Amazon Inspector SBOM Generator.

Berikut ini adalah ikhtisar tentang cara kerja CI/CD integrasi Amazon Inspector khusus:

1. Anda mengonfigurasi Akun AWS untuk mengizinkan akses ke Amazon Inspector Scan API. Untuk petunjuk, lihat Menyiapkan AWS akun untuk menggunakan integrasi Amazon Inspector CI/CD.
2. Anda menginstal dan mengkonfigurasi biner Amazon Inspector SBOM Generator. Untuk petunjuk, lihat Amazon Inspector SBOM Generator.
3. Anda menggunakan Amazon Inspector SBOM Generator untuk menghasilkan SBOM yang CycloneDX kompatibel untuk image container Anda.
4. Anda menggunakan Amazon Inspector Scan API pada SBOM yang dihasilkan untuk menghasilkan laporan kerentanan.

Untuk petunjuk tentang menyiapkan integrasi kustom, lihatMembuat integrasi CI/CD pipeline khusus dengan Amazon Inspector Scan.

Menyiapkan AWS akun untuk menggunakan integrasi Amazon Inspector CI/CD

Untuk menggunakan CI/CD integrasi Amazon Inspector, Anda harus mendaftar untuk file. Akun AWS Peran IAM Akun AWS harus memiliki yang memberikan akses CI/CD pipleline Anda ke Amazon Inspector Scan API. Selesaikan tugas dalam topik berikut untuk mendaftar Akun AWS, membuat pengguna administrator, dan mengonfigurasi peran IAM untuk CI/CD integrasi.

> **Note**
>
> Jika Anda sudah mendaftar untuk Akun AWS, Anda dapat melompat keKonfigurasikan peran IAM untuk integrasi CI/CD.

Topik

- Mendaftar untuk Akun AWS
- Buat pengguna dengan akses administratif
- Konfigurasikan peran IAM untuk integrasi CI/CD

Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka https://portal.aws.amazon.com/billing/pendaftaran.
2. Ikuti petunjuk online.

   Bagian dari prosedur pendaftaran melibatkan menerima panggilan telepon atau pesan teks dan memasukkan kode verifikasi pada keypad telepon.

   Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWSdibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan tugas yang memerlukan akses pengguna root.

AWS mengirimi Anda email konfirmasi setelah proses pendaftaran selesai. Kapan saja, Anda dapat melihat aktivitas akun Anda saat ini dan mengelola akun Anda dengan masuk https://aws.amazon.comke/ dan memilih Akun Saya.

Buat pengguna dengan akses administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

Amankan Anda Pengguna root akun AWS

1. Masuk ke Konsol Manajemen AWSsebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di laman berikutnya, masukkan kata sandi.

   Untuk bantuan masuk dengan menggunakan pengguna root, lihat Masuk sebagai pengguna root di AWS Sign-In Panduan Pengguna.
2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

   Untuk petunjuk, lihat Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root (konsol) Anda di Panduan Pengguna IAM.

Buat pengguna dengan akses administratif

1. Aktifkan Pusat Identitas IAM.

   Untuk mendapatkan petunjuk, silakan lihat Mengaktifkan AWS IAM Identity Center di Panduan Pengguna AWS IAM Identity Center .

2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna.

   Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna dengan akses administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email saat Anda membuat pengguna Pusat Identitas IAM.

  Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat Masuk ke portal AWS akses di Panduan AWS Sign-In Pengguna.

Tetapkan akses ke pengguna tambahan

1. Di Pusat Identitas IAM, buat set izin yang mengikuti praktik terbaik menerapkan izin hak istimewa paling sedikit.

   Untuk petunjuknya, lihat Membuat set izin di Panduan AWS IAM Identity Center Pengguna.

2. Tetapkan pengguna ke grup, lalu tetapkan akses masuk tunggal ke grup.

   Untuk petunjuk, lihat Menambahkan grup di Panduan AWS IAM Identity Center Pengguna.

Konfigurasikan peran IAM untuk integrasi CI/CD

Untuk mengintegrasikan pemindaian Amazon Inspector ke dalam CI/CD pipeline Anda, Anda perlu membuat kebijakan IAM yang memungkinkan akses ke Amazon Inspector Scan API yang memindai tagihan perangkat lunak materi (). SBOMs Kemudian, Anda dapat melampirkan kebijakan tersebut ke peran IAM yang dapat diasumsikan akun Anda untuk menjalankan Amazon Inspector Scan API.

1. Masuk ke Konsol Manajemen AWS dan buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi konsol IAM, Kebijakan lalu pilih Buat Kebijakan.
3. Di Editor Kebijakan pilih JSON dan tempel pernyataan berikut:

   JSON

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

4. Pilih Berikutnya.
5. Beri kebijakan nama, misalnya`InspectorCICDscan-policy`, dan tambahkan deskripsi opsional, lalu pilih Buat Kebijakan. Kebijakan ini akan dilampirkan pada peran yang akan Anda buat di langkah selanjutnya.
6. Di panel navigasi konsol IAM, pilih Peran dan kemudian pilih Buat Peran Baru.
7. Untuk jenis entitas Tepercaya pilih Kebijakan kepercayaan khusus dan tempel kebijakan berikut:

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "sts:AssumeRole",
            "Condition": {}
```

```
        }
    ]
}
```

8. Pilih Berikutnya.
9. Di Tambahkan izin, cari dan pilih kebijakan yang Anda buat sebelumnya, lalu pilih Berikutnya.
10. Beri nama peran, misalnya`InspectorCICDscan-role`, dan tambahkan deskripsi opsional, lalu pilih`Create Role`.

Pemeriksaan Amazon Inspector Dockerfile

Bagian ini menjelaskan cara menggunakan Amazon Inspector SBOM Generator untuk memindai Dockerfiles dan Docker menampung gambar untuk kesalahan konfigurasi yang menyebabkan kerentanan keamanan.

Topik

- Menggunakan pemeriksaan Sbomgen Dockerfile
- Pemeriksaan Dockerfile yang didukung

Menggunakan pemeriksaan Sbomgen Dockerfile

Pemeriksaan Dockerfile dilakukan secara otomatis ketika file bernama `Dockerfile` atau `*.Dockerfile` ditemukan dan ketika gambar Docker dipindai.

Anda dapat menonaktifkan pemeriksaan Dockerfile menggunakan argumen. `--skip-scanners dockerfile` Anda juga dapat menggabungkan pemeriksaan Dockerfile dengan pemindai yang tersedia, seperti OS atau paket pihak ketiga.

Contoh perintah cek Docker

Contoh perintah berikut menunjukkan cara menghasilkan SBOMs gambar kontainer Dockerfiles dan Docker, serta untuk OS dan paket pihak ketiga.

```
# 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
```

Contoh komponen file

Berikut ini adalah contoh temuan Dockerfile untuk komponen 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"
    },
```

Contoh komponen respons kerentanan

Berikut ini adalah contoh temuan Dockerfile untuk komponen respons kerentanan.

```
{
      "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**
>
> Jika Anda memanggil Sbomgen tanpa `--scan-sbom` tanda, Anda hanya dapat melihat temuan Dockerfile mentah.

Pemeriksaan Dockerfile yang didukung

SbomgenPemeriksaan Dockerfile didukung untuk hal-hal berikut:

- Paket biner Sudo
- Utilitas APT Debian
- Rahasia hardcode
- Wadah akar
- Bendera perintah yang melemahkan runtime
- Variabel lingkungan yang melemah runtime

Masing-masing pemeriksaan Dockerfile ini memiliki peringkat keparahan yang sesuai, yang dicatat di bagian atas topik berikut.

> **Note**
>
> Rekomendasi yang dijelaskan dalam topik berikut didasarkan pada praktik terbaik industri.

Paket biner Sudo

> **Note**
>
> Peringkat keparahan untuk pemeriksaan ini adalah Info.

Kami merekomendasikan untuk tidak menginstal atau menggunakan paket biner Sudo karena memiliki perilaku TTY dan penerusan sinyal yang tidak dapat diprediksi. Untuk informasi selengkapnya, lihat Pengguna di situs web Docker Docs. Jika kasus penggunaan Anda memerlukan fungsionalitas yang mirip dengan paket biner Sudo, kami sarankan menggunakan Gosu.

DebianUtilitas APT

> **Note**
>
> Peringkat keparahan untuk pemeriksaan ini adalah Tinggi.

Berikut ini adalah praktik terbaik untuk menggunakan utilitas Debian APT.

Menggabungkan **`apt-get`** perintah dalam satu **`Run`** pernyataan untuk menghindari masalah caching

Sebaiknya gabungkan `apt-get` perintah dalam satu pernyataan RUN di dalam wadah Docker Anda. Menggunakan `apt-get update` dengan sendirinya menghasilkan masalah caching dan `apt-get install` instruksi selanjutnya gagal. Untuk informasi selengkapnya, lihat apt-get di situs web Docker Docs.

> **Note**
>
> Perilaku caching yang dijelaskan juga dapat terjadi di dalam Docker wadah Anda jika perangkat lunak kontainer Docker kedaluwarsa.

Menggunakan utilitas baris perintah APT dengan cara non-interaktif

Sebaiknya gunakan utilitas baris perintah APT secara interaktif. Utilitas baris perintah APT dirancang sebagai alat pengguna akhir, dan perilakunya berubah antar versi. Untuk informasi selengkapnya, lihat Penggunaan Skrip dan perbedaan dari alat APT lainnya di situs web Debian.

Rahasia kode keras

> **Note**
>
> Peringkat keparahan untuk pemeriksaan ini sangat penting.

Informasi rahasia di Dockerfile Anda dianggap sebagai rahasia hard-code. Rahasia hard-code berikut dapat diidentifikasi melalui pemeriksaan file Sbomgen Docker:

- AWS kunci akses IDs - `AKIAIOSFODNN7EXAMPLE`
- AWS kunci rahasia — `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`
- DockerHub token akses pribadi — `dckr_pat_thisisa27charexample1234567`
- GitHub token akses pribadi — `ghp_examplev61wY7Pj1YnotrealUoY123456789`
- GitLab token akses pribadi — `glpat-12345example12345678`

Wadah akar

> **Note**
>
> Penanda keparahan untuk pemeriksaan ini adalah Info.

Kami merekomendasikan menjalankan kontainer Docker tanpa hak akses root. Untuk beban kerja kontainer yang tidak dapat berjalan tanpa hak akses root, sebaiknya buat aplikasi Anda

menggunakan prinsip dengan jumlah hak istimewa paling sedikit. Untuk informasi selengkapnya, lihat Pengguna di situs web Docker Docs.

Variabel lingkungan yang melemah runtime

> **Note**
>
> Peringkat keparahan untuk pemeriksaan ini adalah Tinggi.

Beberapa utilitas baris perintah atau runtime bahasa pemrograman mendukung melewati default aman, yang memungkinkan eksekusi melalui metode yang tidak aman.

NODE_TLS_REJECT_UNAUTHORIZED=0

Ketika Node.js proses berjalan dengan `NODE_TLS_REJECT_UNAUTHORIZED` set to`0`, validasi sertifikat TLS dinonaktifkan. Untuk informasi selengkapnya, lihat NODE_TLS_REJECT_UNAUTHORIZED=0 di situs web Node.js.

GIT_SSL_NO_VERIFY=*

Ketika proses baris perintah git berjalan dengan `GIT_SSL_NO_VERIFY` set, Git melewatkan verifikasi sertifikat TLS. Untuk informasi selengkapnya, lihat Variabel lingkungan di situs web Git.

PIP_TRUSTED_HOST=*

Saat proses baris perintah Python pip berjalan dengan `PIP_TRUSTED_HOST` set, Pip melewatkan verifikasi sertifikat TLS pada domain yang ditentukan. Untuk informasi selengkapnya, lihat --trusted-host di situs web Pip.

NPM_CONFIG_STRICT_SSL=Salah

Ketika proses baris perintah Node.js npm berjalan dengan `NPM_CONFIG_STRICT_SSL` set ke false, utilitas Node Package Manager (npm) akan terhubung ke registri NPM tanpa memvalidasi sertifikat TLS. Untuk informasi selengkapnya, lihat strict-ssl di situs web npm Docs.

Bendera perintah yang melemahkan runtime

> **Note**
>
> Peringkat keparahan untuk pemeriksaan ini adalah Tinggi.

Mirip dengan variabel lingkungan pelemahan runtime, beberapa utilitas baris perintah atau runtime bahasa pemrograman mendukung melewati default aman, yang memungkinkan eksekusi melalui metode yang tidak aman.

`npm --strict-ssl=false`

Ketika proses baris perintah Node.js npm dijalankan dengan `--strict-ssl=false` flag, utilitas Node Package Manager (npm) terhubung ke registri NPM tanpa memvalidasi sertifikat TLS. Untuk informasi selengkapnya, lihat strict-ssl di situs web npm Docs.

`apk --allow-untrusted`

Ketika Alpine Package Keeper utilitas dijalankan dengan `--allow-untrusted` bendera, `apk` akan menginstal paket tanpa atau tidak terpercaya tanda tangan. Untuk informasi selengkapnya, lihat repositori berikut di situs web Aline.

`apt-get --allow-unauthenticated`

Ketika utilitas `apt-get` paket Debian dijalankan dengan `--allow-unauthenticated` flag, `apt-get` tidak memeriksa validitas paket. Untuk informasi selengkapnya, lihat apt-get (8) di situs web Debian.

`pip --trusted-host`

Saat utilitas Python pip dijalankan dengan `--trusted-host` flag, nama host yang ditentukan akan melewati validasi sertifikat TLS. Untuk informasi selengkapnya, lihat --trusted-host di situs web Pip.

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

Ketika manajer paket berbasis RPM `rpm` dijalankan dengan,,, dan `--nofiledigest` flag `--nodigest` `--nosignature--noverify`, manajer paket RPM tidak memvalidasi header paket, tanda tangan, atau file saat menginstal paket. Untuk informasi lebih lanjut, lihat halaman manual RPM berikut di situs web RPM.

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

Ketika manajer paket berbasis RPM dijalankan dengan `--setopt=sslverify` flag disetel ke false, manajer `yum-config-manager` paket YUM tidak memvalidasi sertifikat TLS. Untuk informasi lebih lanjut, lihat halaman manual YUM berikut di situs web Man7.

`yum --nogpgcheck`

Ketika manajer paket berbasis RPM `yum` dijalankan dengan `--nogpgcheck` flag, manajer paket YUM melewatkan memeriksa tanda tangan GPG pada paket. Untuk informasi lebih lanjut, lihat yum (8) di situs web Man7.

`curl --insecure, curl -k`

Ketika `curl` dijalankan dengan `-k` tanda `--insecure` atau, validasi sertifikat TLS dinonaktifkan. Secara default, setiap koneksi aman yang `curl` dibuat diverifikasi agar aman sebelum transfer dilakukan. Opsi ini membuat `curl` melewati langkah verifikasi dan melanjutkan tanpa memeriksa. Untuk informasi lebih lanjut, lihat halaman manual Curl berikut di situs web Curl.

`wget --no-check-certificate`

Ketika `wget` dijalankan dengan `--no-check-certificate` bendera, validasi sertifikat TLS dinonaktifkan. Untuk informasi lebih lanjut, lihat halaman manual Wget berikut di situs web GNU.

Pemeriksaan penghapusan untuk database paket OS dalam kontainer

> **Note**
>
> Peringkat keparahan untuk pemeriksaan ini adalah Info.

Penghapusan database paket sistem operasi mengurangi kemampuan untuk memindai inventaris lengkap perangkat lunak gambar kontainer. Database ini harus dibiarkan utuh selama langkah-langkah pembuatan kontainer.

Pemeriksaan penghapusan untuk database paket OS didukung untuk manajer paket berikut:

Alpine Package Keeper (APK)

Gambar kontainer yang menggunakan pengelola paket APK untuk perangkat lunak yang diinstal harus memastikan file sistem APK tidak dihapus selama pembuatan. Untuk informasi selengkapnya, lihat dokumentasi file sistem manpages APK di Arch Linux situs web.

Debian Package Manager (DPKG)

Container yang menggunakan manajer paket DPKG, seperti Debian, Ubuntu, atau gambar berbasis Distroless, harus memastikan database DPKG tidak dihapus selama pembuatan kontainer. Untuk informasi selengkapnya, lihat dokumentasi file sistem manpages DPKG di situs web. Ubuntu

Manajer Paket RPM (RPM)

Container yang menggunakan RPM Package Manager (yum/dnf), seperti Amazon Linux atau Red Hat Enterprise Linux, harus memastikan database RPM tidak dihapus selama pembuatan container. Untuk informasi selengkapnya, lihat dokumentasi file sistem manpages RPM di situs web RPM.

Membuat integrasi CI/CD pipeline khusus dengan Amazon Inspector Scan

Kami menyarankan Anda menggunakan plugin Amazon Inspector jika CI/CD plugin Amazon Inspector tersedia untuk solusi CI/CD Anda. CI/CD Jika CI/CD plugin Amazon Inspector tidak tersedia untuk CI/CD solusi Anda, Anda dapat menggunakan kombinasi Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk membuat integrasi kustom. CI/CD Langkah-langkah berikut menjelaskan cara membuat integrasi CI/CD pipeline khusus dengan Amazon Inspector Scan.

> **Tip**
>
> Anda dapat menggunakan Amazon Inspector SBOM Generator (Sbomgen) untuk melewati Langkah 3 dan Langkah 4 jika Anda ingin menghasilkan dan memindai SBOM Anda dalam satu perintah.

Langkah 1. Mengkonfigurasi Akun AWS

Konfigurasikan Akun AWS yang menyediakan akses ke Amazon Inspector Scan API. Untuk informasi selengkapnya, lihat Menyiapkan AWS akun untuk menggunakan integrasi Amazon Inspector CI/CD.

Langkah 2. Menginstal Sbomgen biner

Instal dan konfigurasikan Sbomgen biner. Untuk informasi lebih lanjut, lihat Memasang Sbomgen.

Langkah 3. Menggunakan Sbomgen

Gunakan Sbomgen untuk membuat file SBOM untuk gambar kontainer yang ingin Anda pindai.

Anda dapat menggunakan contoh berikut. Ganti *image:id* dengan nama gambar yang akan Anda pindai. Ganti *sbom_path.json* dengan lokasi tempat Anda ingin menyimpan output SBOM.

Contoh

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

Langkah 4. Memanggil Amazon Inspector Scan API

Panggil `inspector-scan` API untuk memindai SBOM yang dihasilkan dan memberikan laporan kerentanan.

Anda dapat menggunakan contoh berikut. Ganti *`sbom_path.json`* dengan lokasi file SBOM kompatibel CycloneDX yang valid. Ganti *`ENDPOINT`* dengan titik akhir API untuk AWS Region tempat Anda saat ini diautentikasi. Ganti *`REGION`* dengan Wilayah yang sesuai.

Contoh

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

Untuk daftar lengkap Wilayah AWS dan titik akhir, lihat Wilayah dan titik akhir.

(Opsional) Langkah 5. Hasilkan dan pindai SBOM dalam satu perintah

> **Note**
>
> Hanya selesaikan langkah ini jika Anda melewati Langkah 3 dan Langkah 4.

Hasilkan dan pindai SBOM Anda dalam satu perintah menggunakan `--scan-bom` bendera.

Anda dapat menggunakan contoh berikut. Ganti *`image:id`* dengan nama gambar yang ingin Anda pindai. Ganti *`profile`* dengan profil yang sesuai. Ganti *`REGION`* dengan Wilayah yang sesuai. Ganti */tmp/scan.json* dengan lokasi file scan.json di direktori tmp.

Contoh

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

Untuk daftar lengkap Wilayah AWS dan titik akhir, lihat Wilayah dan titik akhir.

Format keluaran API

Amazon Inspector Scan API dapat menampilkan laporan kerentanan dalam format CycloneDX 1.5 atau Amazon Inspector menemukan JSON. Default dapat diubah menggunakan `--output-format` bendera.

Contoh output format CycloneDX 1,5


```
{
  "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/M5CSVUNV4HWZZXGOKNSK6L7RPM7BOKIB/"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf"
          },
          {
            "url": "https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf"
          },
          {
            "url": "https://lists.fedoraproject.org/archives/list/package-
announce@lists.fedoraproject.org/message/VU57UJDCFIASIO35GC55JMKSRXJMCDFM/"
          },
          {
            "url": "https://www.oracle.com/security-alerts/cpuapr2022.html"
          },
          {
            "url": "https://twitter.com/kurtseifried/status/1469345530182455296"
          },
          {
            "url": "https://tools.cisco.com/security/center/content/
CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd"
```

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

Contoh output format Inspector


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

Menggunakan plugin Amazon Inspector Jenkins

JenkinsPlugin ini memanfaatkan biner Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan terperinci di akhir build, sehingga Anda dapat menyelidiki dan memulihkan risiko sebelum penerapan. Dengan Jenkins plugin Amazon Inspector, Anda dapat menambahkan pemindaian kerentanan Amazon Inspector ke pipeline Anda. Jenkins Pemindaian kerentanan Amazon Inspector dapat dikonfigurasi untuk lulus atau gagal eksekusi pipeline berdasarkan jumlah dan tingkat keparahan kerentanan yang terdeteksi. Anda dapat melihat versi terbaru dari Jenkins plugin di Jenkins pasar di https://plugins.jenkins.io/amazon-inspector-image-scanner/. Langkah-langkah berikut menjelaskan cara mengatur plugin Amazon Inspector Jenkins.

> **Important**
>
> Sebelum menyelesaikan langkah-langkah berikut, Anda harus memutakhirkan Jenkins ke versi 2.387.3 atau lebih tinggi agar plugin dapat berjalan.

Langkah 1. Mengatur sebuah Akun AWS

Konfigurasikan Akun AWS dengan peran IAM yang memungkinkan akses ke Amazon Inspector Scan API. Untuk petunjuk, lihat Menyiapkan AWS akun untuk menggunakan integrasi Amazon Inspector CI/CD.

Langkah 2. Instal Plugin Amazon Inspector Jenkins

Prosedur berikut menjelaskan cara menginstal plugin Amazon Inspector Jenkins dari dasbor. Jenkins

1. Dari dasbor Jenkins, pilih Kelola Jenkins, lalu pilih Kelola Plugin.
2. Pilih Tersedia.
3. Dari tab Tersedia, cari Amazon Inspector Scan, lalu instal plugin.

(Opsional) Langkah 3. Tambahkan kredensi docker ke Jenkins

> **Note**
>
> Hanya tambahkan kredensyal docker jika image docker ada di repositori pribadi. Jika tidak, lewati langkah ini.

Prosedur berikut menjelaskan cara menambahkan kredensyal docker dari dasbor. Jenkins Jenkins

1. Dari dasbor Jenkins, pilih Manage Jenkins, Credentials, dan kemudian System.
2. Pilih Kredensial global lalu Tambahkan kredensyal.
3. Untuk Jenis, pilih Nama pengguna dengan kata sandi.
4. Untuk Lingkup, pilih Global (Jenkins, node, item, semua item anak, dll).
5. Masukkan detail Anda, lalu pilih OK.

(Opsional) Langkah 4. Tambahkan AWS kredensi

> **Note**
>
> Hanya tambahkan AWS kredensyal jika Anda ingin mengautentikasi berdasarkan pengguna IAM. Jika tidak, lewati langkah ini.

Prosedur berikut menjelaskan cara menambahkan AWS kredensyal dari dasbor. Jenkins

1. Dari dasbor Jenkins, pilih Manage Jenkins, Credentials, dan kemudian System.
2. Pilih Kredensial global lalu Tambahkan kredensyal.

3. Untuk Jenis, pilih AWS Credentials.
4. Masukkan detail Anda, termasuk ID Kunci Akses dan Kunci Akses Rahasia, lalu pilih OK.

Langkah 5. Tambahkan dukungan CSS dalam Jenkins skrip

Prosedur berikut menjelaskan cara menambahkan dukungan CSS dalam Jenkins skrip.

1. Mulai ulang Jenkins.
2. Dari Dashboard, pilih Manage Jenkins, Nodes, Built-in Node, dan kemudian Script Console.
3. Di kotak teks, tambahkan barisSystem.setProperty("hudson.model.DirectoryBrowserSupport.CSP", ""), lalu pilih Jalankan.

Langkah 6. Tambahkan Amazon Inspector Scan ke build Anda

Anda dapat menambahkan Amazon Inspector Scan ke build dengan menambahkan langkah build dalam project Anda atau dengan menggunakan pipeline Jenkins deklaratif.

Amazon Inspector Scan ke build Anda dengan menambahkan langkah build dalam proyek Anda

1. Pada halaman konfigurasi, gulir ke bawah ke Build Steps, dan pilih Add build step. Kemudian pilih Amazon Inspector Scan.
2. Pilih antara dua metode instalasi inspector-sbomgen: Otomatis atau Manual. Opsi otomatis memungkinkan plugin untuk mengunduh versi terbaru. Ini juga memastikan Anda selalu memiliki fitur terbaru, pembaruan keamanan, dan perbaikan bug.

    a. (Opsi 1) Pilih Otomatis untuk mengunduh versi terbaru dari inspector-sbomgen. Opsi ini secara otomatis mendeteksi sistem operasi dan arsitektur CPU yang sedang digunakan.

    b. (Opsi 2) Pilih Manual jika Anda ingin mengatur biner Amazon Inspector SBOM Generator untuk pemindaian. Jika Anda memilih metode ini, pastikan untuk memberikan jalur lengkap ke versi inspector-sbomgen yang diunduh sebelumnya.

    Untuk informasi selengkapnya, lihat Menginstal Amazon Inspector SBOM Generator (Sbomgen) di Amazon Inspector SBOM Generator.

3. Selesaikan yang berikut ini untuk menyelesaikan konfigurasi langkah pembuatan Amazon Inspector Scan:

   a. Masukkan Id Gambar Anda. Gambar dapat berupa lokal, jarak jauh, atau diarsipkan. Nama gambar harus mengikuti konvensi Docker penamaan. Jika menganalisis gambar yang diekspor, berikan jalur ke file tar yang diharapkan. Lihat contoh jalur Id Gambar berikut:

      i. Untuk kontainer lokal atau jarak jauh: `NAME[:TAG|@DIGEST]`

      ii. Untuk file tar: `/path/to/image.tar`

   b. Pilih AWS Regionuntuk mengirim permintaan pemindaian melalui.

   c. (Opsional) Untuk Laporkan Nama Artifact, masukkan nama kustom untuk artefak yang dihasilkan selama proses pembuatan. Ini membantu mengidentifikasi dan mengelolanya secara unik.

   d. (Opsional) Untuk Lewati file, tentukan satu atau beberapa direktori yang ingin Anda kecualikan dari pemindaian. Pertimbangkan opsi ini untuk direktori yang tidak perlu dipindai karena ukurannya.

   e. (Opsional) Untuk kredensyal Docker, pilih nama pengguna Anda. Docker Lakukan ini hanya jika gambar kontainer Anda ada di repositori pribadi.

   f. (Opsional) Anda dapat memberikan metode AWS otentikasi yang didukung berikut ini:

      i. (Opsional) Untuk peran IAM, berikan peran ARN (arn:aws:iam: ::role/). *`AccountNumber RoleName`*

      ii. (Opsional) Untuk kredensyal AWS, tentukan AWS kredensyal yang akan diautentikasi berdasarkan pengguna IAM.

      iii. (Opsional) Untuk nama AWS profil, berikan nama profil untuk diautentikasi menggunakan nama profil.

   g. (Opsional) Pilih Aktifkan ambang kerentanan. Dengan opsi ini, Anda dapat menentukan apakah build gagal jika kerentanan yang dipindai melebihi nilai. Jika semua nilai sama`0`, build berhasil, terlepas dari berapa banyak kerentanan yang dipindai. Untuk skor EPSS, nilainya bisa dari 0 hingga 1. Jika kerentanan yang dipindai melebihi nilai, build gagal, dan semua CVEs dengan skor EPSS di atas nilai ditampilkan di konsol.

4. Pilih Simpan.

Tambahkan Amazon Inspector Scan ke build Anda menggunakan pipeline deklaratif Jenkins

Anda dapat menambahkan Amazon Inspector Scan ke build menggunakan pipeline deklaratif Jenkins secara otomatis atau manual.

Untuk mengunduh pipeline SBOMGen deklaratif secara otomatis

- Untuk menambahkan Amazon Inspector Scan ke build, gunakan sintaks contoh berikut. Ganti *IMAGE_PATH* dengan jalur ke gambar Anda (seperti*alpine:latest*), *IAM_ROLE* dengan ARN dari peran IAM yang Anda konfigurasikan pada langkah 1, dan *ID* dengan ID Docker kredenal Anda jika Anda menggunakan repositori pribadi. Anda dapat mengaktifkan ambang kerentanan secara opsional dan menentukan nilai untuk setiap tingkat keparahan.

```
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
```

```
                isEpssThresholdEnabled: false, // Enable/disable EPSS-based failure
 threshold
                epssThreshold: 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
                ])
            }
        }
    }
}
```

Untuk mengunduh pipeline SBOMGen deklaratif secara manual

- Untuk menambahkan Amazon Inspector Scan ke build, gunakan sintaks contoh berikut. Ganti *SBOMGEN_PATH* dengan jalur ke Amazon Inspector SBOM Generator yang Anda instal di langkah 3, *IMAGE_PATH* dengan jalur ke gambar Anda (seperti*alpine:latest*), *IAM_ROLE* dengan ARN peran IAM yang Anda konfigurasikan pada langkah 1, dan *ID* dengan ID Docker kredenal Anda jika Anda menggunakan repositori pribadi. Anda dapat mengaktifkan ambang kerentanan secara opsional dan menentukan nilai untuk setiap tingkat keparahan.

> **Note**
>
> Tempatkan Sbomgen di direktori Jenkins, dan berikan jalur ke direktori Jenkins di plugin (seperti*/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
                isEpssThresholdEnabled: false, // Enable/disable EPSS-based failure
threshold
                epssThreshold: 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
                ])
            }
        }
    }
}
```

Plugin ini mencakup fitur untuk mengelola kerentanan keamanan.

Daftar CVE yang Ditekan

Pemindaian terkadang dapat mendeteksi kerentanan yang bukan ancaman aktual. Untuk mencegah kesalahan positif ini menghentikan build, Anda dapat menambahkannya ke daftar yang ditekan.

```
isSuppressedCveEnabled: true,
suppressedCveList: 'CVE-2023-1234,CVE-2023-5678'
```

Ini mengabaikan spesifik CVEs saat memeriksa apakah build Anda gagal. Anda hanya harus menambahkan positif palsu ke daftar yang ditekan jika Anda mengatasinya. Setelah Anda menambahkan kerentanan ini ke daftar yang ditekan, kerentanan tersebut CVEs masih muncul di laporan keamanan Anda, tetapi tidak akan menyebabkan kegagalan build.

Daftar CVE Gagal Otomatis

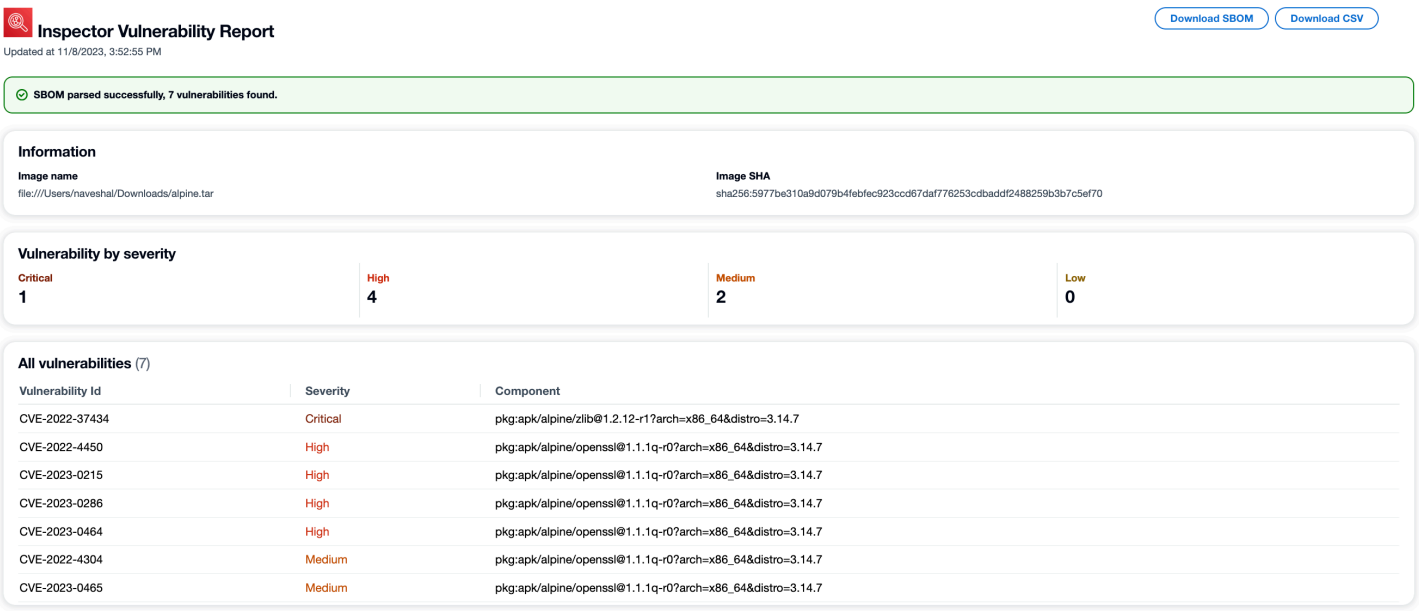
Untuk kerentanan keamanan kritis, Anda dapat membuat daftar yang selalu menyebabkan build Anda gagal.

```
isAutoFailCveEnabled: true,
autoFailCveList: 'CVE-2024-9999'
```

Ini selalu menyebabkan build Anda gagal, apa pun pengaturan yang Anda aktifkan. Anda hanya harus membuat daftar ini untuk masalah keamanan prioritas tinggi yang tidak boleh digunakan. Daftar ini mengesampingkan semua pengaturan ambang batas lainnya untuk keamanan maksimum.

Langkah 7. Lihat laporan kerentanan Amazon Inspector Anda

1. Selesaikan pembangunan baru proyek Anda.
2. Setelah build selesai, pilih format keluaran dari hasil. Jika Anda memilih HTML, Anda memiliki opsi untuk mengunduh laporan versi JSON SBOM atau CSV. Berikut ini menunjukkan contoh laporan HTML:



> **Note**
>
> Anda dapat menggunakan skrip yang lebih lama, karena plugin mendukung nama parameter lama. Namun, Anda akan menemukan peringatan di konsol yang menyarankan Anda memperbarui parameter ini ke yang lebih baru. Misalnya, jika Anda menggunakan`isThresholdEnabled`, Anda akan menemukan peringatan yang menyarankan Anda memperbarui parameter ke`isSeverityThresholdEnabled`.

Pemecahan masalah

Berikut ini adalah kesalahan umum yang dapat Anda temui saat menggunakan plugin Amazon Inspector Scan untuk. Jenkins

Gagal memuat kredensi atau kesalahan pengecualian sts

Kesalahan:

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

Resulltion

Dapatkan `aws_access_key_id` dan `aws_secret_access_key` untuk AWS akun Anda. Siapkan `aws_access_key_id` dan `aws_secret_access_key` masuk`~/.aws/credentials`.

Gagal memuat gambar dari tarball, lokal, atau sumber jarak jauh

Kesalahan:

```
2024/10/16 02:25:17 [ImageDownloadFailed]: failed to load image from
tarball, local, or remote sources.
```

> **Note**
>
> Kesalahan ini dapat terjadi jika plugin Jenkins tidak dapat membaca gambar kontainer, gambar kontainer tidak ditemukan di Docker mesin, dan gambar kontainer tidak ditemukan di registri kontainer jarak jauh.

Penyelesaian:

Verifikasi hal berikut;

- Pengguna plugin Jenkins telah membaca izin untuk gambar yang ingin Anda pindai.
- Gambar yang ingin Anda pindai ada di Docker mesin.
- URL gambar jarak jauh Anda benar.
- Anda diautentikasi ke registri jarak jauh (jika ada).

Kesalahan jalur inspektor-sbomgen

Kesalahan:

```
Exception:com.amazon.inspector.jenkins.amazoninspectorbuildstep.exception.Sbomge
There was an issue running inspector-sbomgen, is /opt/inspector/inspector-
sbomgen the correct path?
```

Penyelesaian:

Selesaikan prosedur berikut untuk menyelesaikan masalah.

1. Tempatkan arsitektur OS Inspector-SBOMGEN yang benar di Jenkins direktori Untuk informasi selengkapnya, lihat Amazon Inspector SBOM Generator.
2. Berikan izin yang dapat dieksekusi ke biner menggunakan perintah berikut:. `chmod +x inspector-sbomgen`

3. Berikan jalur Jenkins mesin yang benar di plugin, seperti`/opt/folder/arm64/inspector-sbomgen`.
4. Simpan konfigurasi, dan jalankan Jenkins pekerjaan.

Menggunakan plugin Amazon Inspector TeamCity

TeamCityPlugin Amazon Inspector memanfaatkan biner Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan terperinci di akhir build, sehingga Anda dapat menyelidiki dan memulihkan risiko sebelum penerapan. Dengan TeamCity plugin Amazon Inspector, Anda dapat menambahkan pemindaian kerentanan Amazon Inspector ke pipeline Anda. TeamCity Pemindaian kerentanan Amazon Inspector dapat dikonfigurasi untuk lulus atau gagal eksekusi pipeline berdasarkan jumlah dan tingkat keparahan kerentanan yang terdeteksi. Anda dapat melihat versi terbaru TeamCity plugin Amazon Inspector di TeamCity pasar di https://plugins.jetbrains.com/plugin/23236 -. amazon-inspector-scanner Untuk informasi tentang cara mengintegrasikan Amazon Inspector Scan ke dalam CI/CD pipeline, lihat Mengintegrasikan pemindaian Amazon Inspector ke dalam pipeline Anda. CI/CD Untuk daftar sistem operasi dan bahasa pemrograman yang didukung Amazon Inspector, lihat Sistem operasi yang didukung dan bahasa pemrograman. Langkah-langkah berikut menjelaskan cara mengatur plugin Amazon Inspector TeamCity.

1. Mengatur sebuah Akun AWS.
   - Konfigurasikan Akun AWS dengan peran IAM yang memungkinkan akses ke Amazon Inspector Scan API. Untuk petunjuk, lihat Menyiapkan AWS akun untuk menggunakan integrasi Amazon Inspector CI/CD.
2. Instal plugin Amazon InspectorTeamCity.
   a. Dari dasbor Anda, buka Administrasi > Plugin.
   b. Cari Amazon Inspector Scan.
   c. Instal plugin.
3. Instal Amazon Inspector SBOM Generator.
   - Instal biner Amazon Inspector SBOM Generator di direktori server Teamcity Anda. Untuk petunjuk, lihat Menginstal Sbomgen.
4. Tambahkan langkah pembuatan Amazon Inspector Scan ke proyek Anda.

a. Pada halaman konfigurasi, gulir ke bawah ke Build Steps, pilih Add build step, lalu pilih Amazon Inspector Scan.

b. Konfigurasikan langkah pembuatan Amazon Inspector Scan dengan mengisi detail berikut:

- Tambahkan nama Langkah.
- Pilih di antara dua metode instalasi Amazon Inspector SBOM Generator: Otomatis atau Manual.
  - Otomatis mengunduh versi terbaru Amazon Inspector SBOM Generator berdasarkan sistem dan arsitektur CPU Anda.
  - Manual mengharuskan Anda menyediakan jalur lengkap ke versi Amazon Inspector SBOM Generator yang diunduh sebelumnya.

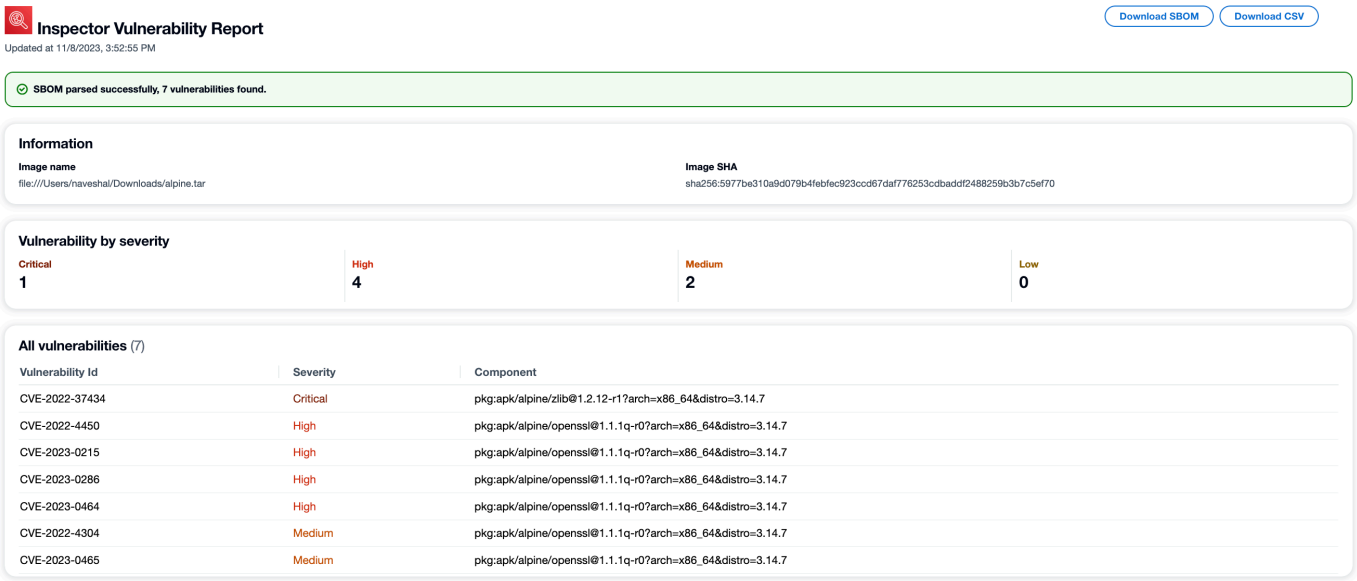
  Untuk informasi lebih lanjut, lihat Menginstal Amazon Inspector SBOM Generator (Sbomgen) di Amazon Inspector SBOM Generator.
- Masukkan Id Gambar Anda. Gambar Anda dapat berupa lokal, jarak jauh, atau diarsipkan. Nama gambar harus mengikuti konvensi Docker penamaan. Jika menganalisis gambar yang diekspor, berikan jalur ke file tar yang diharapkan. Lihat contoh jalur Id Gambar berikut:
  - Untuk kontainer lokal atau jarak jauh: `NAME[:TAG|@DIGEST]`
  - Untuk file tar: `/path/to/image.tar`
- Untuk Peran IAM, masukkan ARN untuk peran yang Anda konfigurasikan pada langkah 1.
- Pilih AWS Regionuntuk mengirim permintaan pemindaian melalui.
- (Opsional) Untuk Otentikasi Docker masukkan Nama Pengguna Docker dan Kata Sandi Docker Anda. Lakukan ini hanya jika gambar kontainer Anda ada di repositori pribadi.
- (Opsional) Untuk AWS Otentikasi, masukkan ID kunci AWS akses dan kunci AWS rahasia Anda. Lakukan ini hanya jika Anda ingin mengautentikasi berdasarkan AWS kredensyal.
- (Opsional) Tentukan ambang kerentanan per tingkat keparahan. Jika jumlah yang Anda tentukan terlampaui selama pemindaian, build gambar akan gagal. Jika nilainya semua `0` build akan berhasil terlepas dari jumlah kerentanan yang ditemukan.

c. Pilih Simpan.

5. Lihat laporan kerentanan Amazon Inspector Anda.

a. Selesaikan pembangunan baru proyek Anda.

b. Saat build selesai pilih format keluaran dari hasil. Saat Anda memilih HTML, Anda memiliki opsi untuk mengunduh laporan versi JSON SBOM atau CSV. Berikut ini adalah contoh dari laporan 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: file:///Users/naveshal/Downloads/alpine.tar

Image SHA: sha256:5977be310a9d079b4febfec923ccd67daf776253cdbaddf2488259b3b7c5ef70

Vulnerability by severity

Critical	High	Medium	Low
1	4	2	0

All vulnerabilities (7)

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

Menggunakan Amazon Inspector dengan tindakan GitHub

Anda dapat menggunakan Amazon Inspector GitHub actionsuntuk menambahkan pemindaian kerentanan Amazon Inspector ke alur kerja Anda. GitHub Ini memanfaatkan Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan terperinci di akhir build, sehingga Anda dapat menyelidiki dan memulihkan risiko sebelum penerapan. Pemindaian kerentanan Amazon Inspector dapat dikonfigurasi untuk meneruskan atau gagal alur kerja berdasarkan jumlah dan tingkat keparahan kerentanan yang terdeteksi. Anda dapat melihat versi terbaru dari tindakan Amazon Inspector di situs web. GitHub Untuk informasi tentang cara mengintegrasikan Amazon Inspector Scan ke dalam CI/CD pipeline, lihat Mengintegrasikan pemindaian Amazon Inspector ke dalam pipeline Anda. CI/CD Untuk daftar sistem operasi dan bahasa pemrograman yang didukung Amazon Inspector, lihat Sistem operasi yang didukung dan bahasa pemrograman.

Menggunakan Amazon Inspector dengan komponen GitLab

Anda dapat menggunakan Amazon Inspector dengan komponen GitLab CI/CD untuk menambahkan pemindaian kerentanan Amazon Inspector ke proyek Anda. GitLab Ini memanfaatkan Amazon

Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan terperinci di akhir build, sehingga Anda dapat menyelidiki dan memulihkan risiko sebelum penerapan. Pemindaian kerentanan Amazon Inspector dapat dikonfigurasi untuk meneruskan atau gagal alur kerja berdasarkan jumlah dan tingkat keparahan kerentanan yang terdeteksi. Anda dapat melihat versi terbaru komponen Amazon Inspector di situs web. GitLab Untuk informasi tentang cara mengintegrasikan Amazon Inspector Scan ke dalam CI/CD pipeline, lihat Mengintegrasikan pemindaian Amazon Inspector ke dalam pipeline Anda. CI/CD Untuk daftar sistem operasi dan bahasa pemrograman yang didukung Amazon Inspector, lihat Sistem operasi yang didukung dan bahasa pemrograman.

Menggunakan CodeCatalyst tindakan dengan Amazon Inspector

Anda dapat menggunakan Amazon Inspector dengan Amazon CodeCatalyst untuk menambahkan pemindaian kerentanan Amazon Inspector ke alur kerja Anda. CodeCatalyst Ini memanfaatkan Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan terperinci di akhir build, sehingga Anda dapat menyelidiki dan memulihkan risiko sebelum penerapan. Pemindaian kerentanan Amazon Inspector dapat dikonfigurasi untuk meneruskan atau gagal alur kerja berdasarkan jumlah dan tingkat keparahan kerentanan yang terdeteksi. Untuk informasi tentang cara mengintegrasikan Amazon Inspector Scan ke dalam CI/CD pipeline, lihat Mengintegrasikan pemindaian Amazon Inspector ke dalam pipeline Anda. CI/CD Untuk daftar sistem operasi dan bahasa pemrograman yang didukung Amazon Inspector, lihat Sistem operasi yang didukung dan bahasa pemrograman.

Menggunakan tindakan Amazon Inspector Scan dengan CodePipeline

Anda dapat menggunakan Amazon Inspector AWS CodePipeline dengan menambahkan pemindaian kerentanan ke alur kerja Anda. Integrasi ini memanfaatkan Amazon Inspector SBOM Generator dan Amazon Inspector Scan API untuk menghasilkan laporan terperinci di akhir build Anda. Integrasi ini membantu Anda menyelidiki dan memulihkan risiko sebelum penerapan. `InspectorScan`Tindakan ini adalah tindakan komputasi terkelola CodePipeline yang mengotomatiskan mendeteksi dan memperbaiki kerentanan keamanan dalam kode sumber terbuka Anda. Anda dapat menggunakan tindakan ini dengan kode sumber aplikasi di repositori pihak ketiga Anda, seperti GitHub atau Bitbucket Cloud, atau dengan gambar untuk aplikasi kontainer. Untuk informasi selengkapnya, lihat InspectorScan memanggil referensi tindakan di Panduan AWS CodePipeline Pengguna.

Menilai cakupan Amazon Inspector dari lingkungan Anda AWS

Anda dapat menilai cakupan Amazon Inspector dari AWS lingkungan Anda dari layar Manajemen akun di konsol Amazon Inspector, yang menampilkan detail dan statistik tentang status pemindaian Amazon Inspector untuk akun dan sumber daya Anda.

> **Note**
>
> Jika Anda adalah administrator yang didelegasikan untuk organisasi, Anda dapat melihat detail dan statistik untuk semua akun di organisasi.

Prosedur berikut menjelaskan cara menilai cakupan lingkungan Amazon Inspector Anda.

Untuk menilai cakupan Amazon Inspector dari lingkungan Anda AWS

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dari panel navigasi, pilih Manajemen akun.
3. Untuk meninjau cakupan, pilih salah satu tab berikut:
   - Pilih Akun untuk meninjau cakupan tingkat akun.
   - Pilih Instans untuk meninjau cakupan instans Amazon Elastic Compute Cloud (Amazon EC2).
   - Pilih repositori Container untuk meninjau cakupan repositori Amazon Elastic Container Registry (Amazon ECR).
   - Pilih gambar Container untuk meninjau cakupan gambar kontainer Amazon ECR.
   - Pilih fungsi Lambda untuk meninjau cakupan fungsi Lambda.

Topik berikut menjelaskan informasi yang diberikan masing-masing tab ini.

Topik

- Menilai cakupan gambar kontainer Amazon ECR
- Menilai cakupan fungsi AWS Lambda

Menilai cakupan tingkat akun

Jika akun Anda bukan bagian dari organisasi atau bukan akun administrator Amazon Inspector yang didelegasikan untuk organisasi, tab Akun memberikan informasi tentang akun Anda dan status pemindaian sumber daya untuk akun Anda. Pada tab ini, Anda dapat mengaktifkan atau menonaktifkan pemindaian untuk semua atau hanya jenis sumber daya tertentu untuk akun Anda. Untuk informasi selengkapnya, lihat Jenis pemindaian otomatis di Amazon Inspector.

Jika akun Anda adalah akun administrator Amazon Inspector yang didelegasikan untuk organisasi, tab Akun menyediakan setelan aktivasi otomatis untuk akun di organisasi Anda, dan mencantumkan semua akun di organisasi Anda. Untuk setiap akun, daftar menunjukkan apakah Amazon Inspector diaktifkan untuk akun dan, jika demikian, jenis pemindaian sumber daya yang diaktifkan untuk akun tersebut. Sebagai administrator yang didelegasikan, Anda dapat menggunakan tab ini untuk mengubah pengaturan aktivasi otomatis untuk organisasi Anda. Anda juga dapat mengaktifkan atau menonaktifkan jenis pemindaian sumber daya tertentu untuk akun anggota individu. Untuk informasi selengkapnya, lihat Mengaktifkan pemindaian Amazon Inspector untuk akun anggota.

Menilai cakupan instans Amazon EC2

Tab Instans menampilkan instans Amazon EC2 di lingkungan Anda. AWS Daftar disusun ke dalam kelompok-kelompok pada tab berikut:

- Semua - Menunjukkan semua contoh di lingkungan Anda. Kolom Status menunjukkan status pemindaian saat ini untuk sebuah instance.
- Scanning - Menunjukkan semua instance yang Amazon Inspector secara aktif memantau dan memindai di lingkungan Anda.
- Tidak memindai - Menunjukkan semua contoh yang Amazon Inspector tidak memantau dan memindai di lingkungan Anda. Kolom Alasan menunjukkan mengapa Amazon Inspector tidak memantau dan memindai instance.

  Instans EC2 dapat muncul di tab Not scanning karena beberapa alasan. Amazon Inspector menggunakan AWS Systems Manager (SSM) dan Agen SSM untuk secara otomatis memantau dan memindai instans EC2 Anda untuk kerentanan. Jika instans tidak menjalankan Agen SSM, tidak memiliki peran AWS Identity and Access Management (IAM) yang mendukung Systems

Manager, atau tidak menjalankan sistem operasi atau arsitektur yang didukung, Amazon Inspector tidak dapat memantau dan memindai instance. Untuk informasi selengkapnya, lihat Pemindaian instans Amazon EC2.

Pada setiap tab, kolom Account menentukan Akun AWS yang memiliki instance.

Tag instans EC2 - Kolom ini menunjukkan tag yang terkait dengan instance dan dapat digunakan untuk menentukan apakah instance Anda telah dikecualikan dari pemindaian oleh tag.

Sistem operasi — Kolom ini menunjukkan kepada Anda jenis sistem operasi, yang dapat berupa`WINDOWS`,`MAC`,`LINUX`, atau`UNKNOWN`.

Dimonitor menggunakan - Kolom ini menunjukkan apakah Amazon Inspector menggunakan metode pemindaian berbasis agen atau tanpa agen pada instance ini.

Terakhir dipindai - Kolom ini menunjukkan kepada Anda kapan Amazon Inspector terakhir memeriksa sumber daya tersebut untuk kerentanan. Frekuensi Amazon Inspector melakukan pemindaian bergantung pada metode pemindaian yang digunakannya untuk memindai instance.

Untuk meninjau detail tambahan tentang instans EC2, pilih tautan di kolom instans EC2. Amazon Inspector kemudian menampilkan detail tentang instance dan temuan saat ini untuk instans tersebut. Untuk meninjau detail temuan, pilih tautan di kolom Judul. Untuk informasi tentang detail ini, lihatMelihat detail untuk temuan Amazon Inspector Anda.

Memindai nilai status untuk instans Amazon EC2

Untuk instans Amazon Elastic Compute Cloud (Amazon EC2), nilai Status yang mungkin adalah:

- Pemantauan aktif - Amazon Inspector terus memantau dan memindai instans.
- Batas penyimpanan instans tanpa agen terlampaui — Amazon Inspector menggunakan status ini ketika ukuran gabungan dari semua volume yang dilampirkan ke instans lebih besar dari 1200 GB, atau instans memiliki lebih dari 8 volume yang melekat padanya.
- Batas waktu pengumpulan instans tanpa agen terlampaui — Amazon Inspector habis waktu saat mencoba menjalankan pemindaian tanpa agen pada sebuah instance.
- Instans EC2 dihentikan — Amazon Inspector menghentikan pemindaian untuk instance karena instance dalam status berhenti. Setiap temuan yang ada akan bertahan sampai instance dihentikan. Jika instance dimulai ulang, Amazon Inspector akan secara otomatis melanjutkan pemindaian untuk instance tersebut.

- Kesalahan internal - Terjadi kesalahan internal saat Amazon Inspector mencoba memindai instance. Amazon Inspector akan secara otomatis mengatasi kesalahan dan melanjutkan pemindaian sesegera mungkin.
- Tidak ada inventaris — Amazon Inspector tidak dapat menemukan inventaris aplikasi perangkat lunak untuk memindai instance. Asosiasi Amazon Inspector untuk instance tersebut mungkin telah dihapus atau mungkin gagal dijalankan.

  Untuk mengatasi masalah ini, gunakan AWS Systems Manager untuk memastikan bahwa `InspectorInventoryCollection-do-not-delete` asosiasi ada dan status asosiasinya berhasil. Selain itu, gunakan AWS Systems Manager Fleet Manager untuk memverifikasi inventaris aplikasi perangkat lunak untuk instance tersebut.
- Menunggu penonaktifan - Amazon Inspector telah berhenti memindai instance. Instance sedang dinonaktifkan, menunggu penyelesaian tugas pembersihan.
- Pemindaian awal yang tertunda - Amazon Inspector telah mengantri instance untuk pemindaian awal.
- Sumber daya dihentikan - Instance dihentikan. Amazon Inspector saat ini sedang membersihkan temuan dan data cakupan yang ada untuk instance tersebut.
- Inventaris basi — Amazon Inspector tidak dapat mengumpulkan inventaris aplikasi perangkat lunak yang diperbarui yang ditangkap dalam 7 hari terakhir untuk instance tersebut.

  Untuk mengatasi masalah ini, gunakan AWS Systems Manager untuk memastikan bahwa asosiasi Amazon Inspector yang diperlukan ada dan berjalan untuk instans. Selain itu, gunakan AWS Systems Manager Fleet Manager untuk memverifikasi inventaris aplikasi perangkat lunak untuk instance tersebut.
- Instans EC2 yang tidak dikelola — Amazon Inspector tidak memantau atau memindai instans. Instance tidak dikelola oleh AWS Systems Manager.

  Untuk mengatasi masalah ini, Anda dapat menggunakan yang AWSSupport-TroubleshootManagedInstance runbookdisediakan oleh AWS Systems Manager Automation. Setelah Anda mengonfigurasi AWS Systems Manager untuk mengelola instans, Amazon Inspector akan secara otomatis mulai memantau dan memindai instans secara otomatis.
- OS yang tidak didukung - Amazon Inspector tidak memantau atau memindai instans. Instans menggunakan sistem operasi atau arsitektur yang tidak didukung Amazon Inspector. Untuk daftar sistem operasi yang didukung Amazon Inspector, lihat. Nilai status instans Amazon EC2

- Memantau secara aktif dengan kesalahan sebagian — Status ini berarti bahwa pemindaian EC2 aktif, tetapi ada kesalahan yang terkait dengannyaInspeksi mendalam Amazon Inspector untuk instans Amazon EC2 berbasis Linux. Kemungkinan kesalahan inspeksi mendalam adalah:
  - Batas pengumpulan paket inspeksi mendalam terlampaui - Instance telah melampaui batas paket 5000 untuk inspeksi mendalam Amazon Inspector. Untuk melanjutkan pemeriksaan mendalam untuk contoh ini, Anda dapat mencoba menyesuaikan jalur kustom yang terkait dengan akun.
  - Batas inventaris ssm harian inspeksi mendalam terlampaui — Agen SSM tidak dapat mengirim inventaris ke Amazon Inspector karena kuota SSM untuk data Inventaris yang dikumpulkan per instans per hari telah tercapai untuk contoh ini. Untuk informasi selengkapnya, lihat titik akhir dan kuota Amazon EC2 Systems Manager.
  - Batas waktu pengumpulan inspeksi mendalam terlampaui - Amazon Inspector gagal mengekstrak inventaris paket karena waktu pengumpulan paket melebihi ambang batas maksimum 15 menit.
  - Inspeksi mendalam tidak memiliki inventaris - Plugin Amazon Inspector SSM belum dapat mengumpulkan inventaris paket untuk contoh ini. Ini biasanya merupakan hasil dari pemindaian yang tertunda, namun, jika status ini berlanjut setelah 6 jam, gunakan Amazon EC2 Systems Manager untuk memastikan bahwa asosiasi Amazon Inspector yang diperlukan ada dan berjalan untuk instance.

Untuk detail tentang mengonfigurasi pengaturan pemindaian untuk instans EC2, lihat. Pemindaian instans Amazon EC2

Menilai cakupan repositori Amazon ECR

Tab Repositori menunjukkan repositori Amazon ECR di lingkungan Anda. AWS Daftar disusun ke dalam kelompok-kelompok pada tab berikut:

- Semua - Menampilkan semua repositori di lingkungan Anda. Kolom Status menunjukkan status pemindaian saat ini untuk repositori.
- Diaktifkan - Menampilkan semua repositori yang Amazon Inspector dikonfigurasi untuk memantau dan memindai di lingkungan Anda. Kolom Status menunjukkan status pemindaian saat ini untuk repositori.

- Tidak diaktifkan - Menampilkan semua repositori yang Amazon Inspector tidak memantau dan memindai di lingkungan Anda. Kolom Alasan menunjukkan mengapa Amazon Inspector tidak memantau dan memindai repositori.

Pada setiap tab, kolom Account menentukan Akun AWS yang memiliki repositori.

Untuk meninjau detail tambahan tentang repositori, pilih nama repositori. Amazon Inspector kemudian menampilkan daftar gambar kontainer di repositori dan detail untuk setiap gambar. Detailnya termasuk tag gambar, intisari gambar, dan status pemindaian. Mereka juga termasuk statistik temuan kunci, seperti jumlah temuan kritis untuk gambar. Untuk menelusuri dan meninjau data pendukung untuk menemukan statistik, pilih tag gambar untuk gambar.

> **Note**
>
> Gambar Amazon ECR tanpa pemindaian berkelanjutan tidak termasuk dalam widget cakupan.

Memindai nilai status untuk repositori Amazon ECR

Untuk repositori Amazon Elastic Container Registry (Amazon ECR), nilai Status yang mungkin adalah:

- Activated (Continuous) - Untuk repositori, Amazon Inspector terus memantau gambar di repositori ini. Pengaturan pemindaian yang disempurnakan untuk repositori diatur ke pemindaian berkelanjutan. Amazon Inspector awalnya memindai gambar baru ketika mereka didorong dan memindai ulang gambar jika CVE baru yang relevan dengan gambar itu diterbitkan. Amazon Inspector akan terus memantau gambar di repositori ini untuk durasi pemindaian ulang Amazon ECR yang Anda konfigurasikan.
- Diaktifkan (On push) - Amazon Inspector secara otomatis memindai gambar kontainer individual di repositori saat gambar baru didorong. Pemindaian yang ditingkatkan diaktifkan untuk repositori dan diatur untuk memindai saat push.
- Akses ditolak - Amazon Inspector tidak diizinkan mengakses repositori atau gambar kontainer apa pun di repositori.

  Untuk mengatasi masalah ini, pastikan bahwa kebijakan AWS Identity and Access Management (IAM) untuk repositori memungkinkan Amazon Inspector mengakses repositori.

- Dinonaktifkan (Manual) - Amazon Inspector tidak memantau atau memindai gambar kontainer apa pun di repositori. Pengaturan pemindaian Amazon ECR untuk repositori diatur ke pemindaian manual dasar.

  Untuk mulai memindai gambar di repositori dengan Amazon Inspector, ubah pengaturan pemindaian untuk repositori menjadi pemindaian yang disempurnakan, lalu pilih apakah akan memindai gambar secara terus menerus atau hanya ketika gambar baru didorong.
- Diaktifkan (On push) - Amazon Inspector secara otomatis memindai gambar kontainer individual di repositori saat gambar baru didorong. Pengaturan pemindaian yang disempurnakan untuk repositori diatur untuk memindai saat push.
- Kesalahan internal - Terjadi kesalahan internal saat Amazon Inspector mencoba memindai repositori. Amazon Inspector akan secara otomatis mengatasi kesalahan dan melanjutkan pemindaian sesegera mungkin.

Untuk detail tentang mengkonfigurasi pengaturan pemindaian untuk Pemindaian gambar wadah Amazon ECR repositori.

Menilai cakupan gambar kontainer Amazon ECR

Tab Gambar menunjukkan gambar kontainer Amazon ECR di AWS lingkungan Anda. Daftar disusun ke dalam kelompok-kelompok pada tab berikut:

- Semua - Menampilkan semua gambar kontainer di lingkungan Anda. Kolom Status menunjukkan status pemindaian saat ini untuk gambar.
- Scanning - Menampilkan semua gambar kontainer yang Amazon Inspector dikonfigurasi untuk memantau dan memindai di lingkungan Anda. Kolom Status menunjukkan status pemindaian saat ini untuk gambar.
- Tidak memindai - Menampilkan semua gambar kontainer yang Amazon Inspector tidak memantau dan memindai di lingkungan Anda. Kolom Alasan menunjukkan mengapa Amazon Inspector tidak memantau dan memindai gambar.

  Gambar kontainer dapat muncul di tab Tidak diaktifkan karena beberapa alasan. Gambar mungkin disimpan dalam repositori yang tidak diaktifkan oleh pemindaian Amazon Inspector, atau aturan pemfilteran Amazon ECR mencegah repositori tersebut dipindai. Atau gambar belum didorong atau ditarik dalam jumlah hari yang Anda konfigurasi untuk durasi pemindaian ulang ECR. Untuk informasi selengkapnya, lihat Mengonfigurasi durasi pemindaian ulang Amazon ECR.

Pada setiap tab, kolom nama Repositori menentukan nama repositori yang menyimpan gambar kontainer. Kolom Akun menentukan Akun AWS yang memiliki repositori. Kolom terakhir yang dipindai menunjukkan kepada Anda kapan Amazon Inspector terakhir memeriksa sumber daya tersebut untuk kerentanan. Ini dapat mencakup pemeriksaan ketika ada pembaruan untuk menemukan metadata, ketika ada pembaruan ke inventaris aplikasi sumber daya, atau ketika pemindaian ulang dilakukan sebagai respons terhadap CVE baru. Untuk informasi selengkapnya, lihat Perilaku pemindaian untuk pemindaian Amazon ECR.

Untuk meninjau detail tambahan tentang gambar kontainer, pilih tautan di kolom gambar kontainer ECR. Amazon Inspector kemudian menampilkan detail tentang gambar dan temuan terkini untuk gambar tersebut. Untuk meninjau detail temuan, pilih tautan di kolom Judul. Untuk informasi tentang detail ini, lihatMelihat detail untuk temuan Amazon Inspector Anda.

Memindai nilai status untuk gambar kontainer Amazon ECR

Untuk image container Amazon Elastic Container Registry, nilai Status yang mungkin adalah:

- Pemantauan aktif (Berkelanjutan) - Amazon Inspector terus memantau dan gambar serta pemindaian baru dilakukan di atasnya setiap kali CVE baru yang relevan diterbitkan. Durasi pemindaian ulang Amazon ECR untuk gambar disegarkan setiap kali gambar didorong atau ditarik. Pemindaian yang disempurnakan diaktifkan untuk repositori yang menyimpan gambar, dan pengaturan pemindaian yang disempurnakan untuk repositori diatur ke pemindaian berkelanjutan.
- Diaktifkan (On push) - Amazon Inspector secara otomatis memindai gambar setiap kali gambar baru didorong. Pemindaian yang disempurnakan diaktifkan untuk repositori yang menyimpan gambar, dan pengaturan pemindaian yang disempurnakan untuk repositori diatur untuk memindai saat push.
- Kesalahan internal - Terjadi kesalahan internal saat Amazon Inspector mencoba memindai gambar kontainer. Amazon Inspector akan secara otomatis mengatasi kesalahan dan melanjutkan pemindaian sesegera mungkin.
- Pemindaian awal yang tertunda - Amazon Inspector telah mengantri gambar untuk pemindaian awal.
- Kelayakan pemindaian kedaluwarsa (Berkelanjutan) - Amazon Inspector menangguhkan pemindaian untuk gambar. Gambar belum diperbarui dalam durasi yang Anda tentukan untuk pemindaian ulang otomatis gambar di repositori. Anda dapat mendorong atau menarik gambar untuk melanjutkan pemindaian.
- Kelayakan pemindaian kedaluwarsa (On push) - Amazon Inspector menangguhkan pemindaian untuk gambar. Gambar belum diperbarui dalam durasi yang Anda tentukan untuk pemindaian

ulang otomatis gambar di repositori. Anda dapat mendorong gambar untuk melanjutkan pemindaian.

- Manual frekuensi pemindaian (Manual) - Amazon Inspector tidak memindai gambar wadah Amazon ECR. Pengaturan pemindaian Amazon ECR untuk repositori yang menyimpan gambar diatur ke pemindaian manual dasar. Untuk mulai memindai gambar secara otomatis dengan Amazon Inspector, ubah pengaturan repositori menjadi pemindaian yang disempurnakan, lalu pilih apakah akan memindai gambar secara terus menerus atau hanya ketika gambar baru didorong.
- OS yang tidak didukung - Amazon Inspector tidak memantau atau memindai gambar. Gambar didasarkan pada sistem operasi yang Amazon Inspector tidak mendukung, atau menggunakan jenis media yang Amazon Inspector tidak mendukung.

  Untuk daftar sistem operasi yang didukung Amazon Inspector, lihat. Sistem operasi yang didukung: Pemindaian Amazon ECR dengan Amazon Inspector Untuk daftar jenis media yang didukung Amazon Inspector, lihat Jenis media yang didukung.

Untuk detail tentang mengonfigurasi pengaturan pemindaian untuk repositori dan gambar, lihat. Pemindaian gambar wadah Amazon ECR

Menilai cakupan fungsi AWS Lambda

Tab Lambda menunjukkan fungsi Lambda di lingkungan Anda. AWS Halaman ini dua tabel, satu yang menunjukkan detail cakupan fungsi untuk pemindaian standar Lambda dan satu lagi untuk pemindaian kode Lambda. Anda dapat mengelompokkan fungsi berdasarkan tab berikut:

- Semua - Menampilkan semua fungsi Lambda di lingkungan Anda. Kolom Status menunjukkan status pemindaian saat ini untuk fungsi Lambda.
- Scanning - Menunjukkan fungsi Lambda yang Amazon Inspector dikonfigurasi untuk memindai. Kolom Status menunjukkan status pemindaian saat ini untuk setiap fungsi Lambda.
- Tidak memindai - Menunjukkan fungsi Lambda yang Amazon Inspector tidak dikonfigurasi untuk memindai. Kolom Alasan menunjukkan mengapa Amazon Inspector tidak memantau dan memindai suatu fungsi.

  Fungsi Lambda dapat muncul di tab Tidak memindai karena beberapa alasan. Fungsi Lambda mungkin milik akun yang belum ditambahkan ke Amazon Inspector atau aturan pemfilteran mencegah fungsi ini dipindai. Untuk informasi selengkapnya, lihat Pemindaian fungsi Lambda.

Pada setiap tab, kolom nama Fungsi menentukan nama fungsi Lambda. Kolom Akun menentukan Akun AWS yang memiliki fungsi. Runtime menentukan runtime fungsi. Kolom Status menunjukkan status pemindaian saat ini untuk setiap fungsi Lambda. Tag sumber daya menunjukkan tag yang telah diterapkan ke fungsi. Kolom terakhir yang dipindai menunjukkan kepada Anda kapan Amazon Inspector terakhir memeriksa sumber daya tersebut untuk kerentanan. Ini dapat mencakup pemeriksaan ketika ada pembaruan untuk menemukan metadata, ketika ada pembaruan ke inventaris aplikasi sumber daya, atau ketika pemindaian ulang dilakukan sebagai respons terhadap CVE baru. Untuk informasi selengkapnya, lihat Memindai perilaku untuk pemindaian fungsi Lambda.

Memindai nilai status untuk AWS Lambda fungsi

Untuk fungsi Lambda, nilai Status yang mungkin adalah:

- Pemantauan aktif - Amazon Inspector terus memantau dan memindai fungsi Lambda. Pemindaian berkelanjutan mencakup pemindaian awal fungsi baru saat didorong ke repositori dan pemindaian ulang fungsi otomatis saat diperbarui atau saat Common Vulnerabilities and Exposures () baru dirilis. CVEs
- Dikecualikan oleh tag - Amazon Inspector tidak memindai fungsi ini karena telah dikecualikan dari pemindaian oleh tag.
- Kelayakan pemindaian kedaluwarsa - Amazon Inspector tidak memantau fungsi ini karena sudah 90 hari atau lebih sejak terakhir dipanggil atau diperbarui.
- Kesalahan internal —Terjadi kesalahan internal saat Amazon Inspector mencoba memindai fungsi. Amazon Inspector akan secara otomatis mengatasi kesalahan dan melanjutkan pemindaian sesegera mungkin.
- Pemindaian awal yang tertunda - Amazon Inspector telah mengantri fungsi untuk pemindaian awal.
- Tidak didukung - Fungsi Lambda memiliki runtime yang tidak didukung.

Mengelola beberapa akun di Amazon Inspector dengan AWS Organizations

Anda dapat menggunakan Amazon Inspector untuk mengelola beberapa akun dalam suatu organisasi. Amazon Inspector mendukung dua pendekatan untuk manajemen multi-akun:

- Administrator yang didelegasikan untuk AWS Organizations kebijakan - Menyediakan tata kelola terpusat kepada administrator yang didelegasikan dengan pengaktifan otomatis Amazon Inspector di seluruh akun organisasi di seluruh wilayah. Kebijakan organisasi memberlakukan jenis pemindaian mana yang diaktifkan dan diutamakan daripada administrator yang didelegasikan dan akun anggota yang tidak dikelola kebijakan.
- Administrator yang didelegasikan untuk non AWS Organizations kebijakan - Akun yang ditunjuk untuk mengelola Amazon Inspector untuk organisasi tanpa menggunakan kebijakan organisasi. Administrator yang didelegasikan dapat mengaktifkan Amazon Inspector untuk akun anggota dan mengonfigurasi pengaturan pemindaian.

Pendekatan-pendekatan ini dapat digunakan bersama. Ketika kebijakan organisasi diberlakukan, mereka mengontrol pemberdayaan jenis sumber daya (yang jenis pemindaian diaktifkan), sementara administrator yang didelegasikan mempertahankan kontrol atas pengaturan konfigurasi pemindaian seperti mode pemindaian dan jalur inspeksi mendalam. Topik berikut menjelaskan pendekatan manajemen ini, cara menunjuk administrator yang didelegasikan, dan cara mengelola akun anggota.

Topik

Memahami akun administrator dan akun anggota yang didelegasikan di Amazon Inspector

Saat menggunakan Amazon Inspector di lingkungan multi-akun, akun administrator yang didelegasikan memiliki akses ke metadata tertentu. Metadata mencakup pemindaian standar untuk Amazon EC2, Amazon ECR, dan Lambda, dan pemindaian kode Lambda. Ini juga mencakup hasil pencarian keamanan untuk akun anggota. Bagian ini memberikan informasi tentang tindakan yang dapat dilakukan oleh akun admin yang didelegasikan dan akun anggota dapat dilakukan.

Model tata kelola kebijakan organisasi

Ketika AWS Organizations kebijakan digunakan untuk mengaktifkan Amazon Inspector, model tata kelola diberlakukan yang menentukan tindakan mana yang diizinkan:

Sumber daya yang dikelola kebijakan

Sumber daya yang diaktifkan atau dinonaktifkan secara eksplisit oleh kebijakan organisasi tidak dapat diubah oleh administrator atau akun anggota yang didelegasikan. Permintaan API untuk mengaktifkan atau menonaktifkan jenis pemindaian yang dikelola kebijakan akan gagal dengan kesalahan yang jelas yang menunjukkan sumber daya dikelola oleh kebijakan organisasi.

Non-policy-managed sumber daya

Sumber daya yang tidak ditentukan dalam kebijakan organisasi dapat dikelola secara normal oleh administrator dan akun anggota yang didelegasikan menggunakan konsol Amazon Inspector atau API.

Manajemen konfigurasi pindai

Administrator yang didelegasikan selalu dapat mengonfigurasi pengaturan pemindaian seperti mode pemindaian EC2, jalur inspeksi mendalam, dan durasi pemindaian ulang ECR, terlepas dari apakah jenis sumber daya dikelola oleh kebijakan. Kebijakan organisasi hanya mengontrol apakah pemindaian diaktifkan, bukan cara operasinya.

Untuk informasi selengkapnya tentang membuat dan mengelola kebijakan organisasi Amazon Inspector, lihat AWS Organizations dokumentasi untuk kebijakan Amazon Inspector.

Tindakan administrator yang didelegasikan

Umumnya, ketika administrator yang didelegasikan menerapkan pengaturan ke akun mereka, pengaturan tersebut diterapkan ke semua akun lain di organisasi. Administrator yang didelegasikan juga dapat melihat dan mengambil informasi untuk akun mereka sendiri dan anggota terkait. Akun administrator yang didelegasikan Amazon Inspector dapat melakukan tindakan berikut:

- Hanya akun AWS Organizations manajemen yang dapat menunjuk dan menghapus administrator yang didelegasikan.
- Saat menunjuk administrator yang didelegasikan, Anda harus berada di organisasi yang sama dengan akun anggota yang ingin Anda kelola.

- Melihat dan mengelola status Amazon Inspector untuk akun terkait, termasuk mengaktifkan dan menonaktifkan Amazon Inspector.
- Aktifkan atau nonaktifkan jenis pemindaian untuk semua akun anggota di organisasi.
- Lihat data temuan gabungan di seluruh organisasi dan temukan detail untuk semua akun anggota dalam organisasi.
- Buat dan kelola aturan penindasan yang berlaku untuk temuan untuk semua akun di organisasi.
- Aktifkan pemindaian Amazon ECR yang ditingkatkan untuk semua anggota organisasi.
- Lihat cakupan sumber daya untuk seluruh organisasi.
- Tentukan durasi pemindaian ulang otomatis gambar kontainer ECR untuk semua akun anggota di organisasi. Pengaturan durasi pemindaian administrator yang didelegasikan akan mengesampingkan setelan apa pun yang sebelumnya ditetapkan oleh akun anggota. Semua akun di organisasi berbagi durasi pemindaian ulang otomatis Amazon ECR dari administrator yang didelegasikan. Anda tidak dapat mengatur durasi pemindaian ulang yang berbeda untuk masing-masing akun.
- Tentukan lima jalur khusus untuk inspeksi mendalam Amazon Inspector untuk Amazon EC2 yang akan digunakan di semua akun di organisasi. Ini merupakan tambahan dari lima jalur kustom yang dapat ditetapkan oleh administrator yang didelegasikan untuk akun individu mereka. Untuk informasi selengkapnya tentang mengonfigurasi jalur kustom inspeksi mendalam, lihatJalur khusus untuk inspeksi mendalam Amazon Inspector.
- Aktifkan dan nonaktifkan inspeksi mendalam Amazon Inspector untuk akun anggota.
- Ekspor SBOMs untuk akun anggota mana pun di organisasi.
- Setel mode pemindaian Amazon EC2 untuk semua akun anggota di organisasi. Untuk informasi selengkapnya, lihat Mengelola mode pemindaian.
- Buat dan kelola konfigurasi pemindaian CIS untuk semua akun di organisasi, kecuali untuk konfigurasi pemindaian apa pun yang dibuat oleh akun anggota.

> **Note**
>
> Jika akun anggota meninggalkan organisasi, administrator yang didelegasikan tidak akan lagi dapat melihat konfigurasi pemindaian yang dijadwalkan oleh akun tersebut.

- Lihat hasil pemindaian CIS untuk semua akun di organisasi.
- Saat kebijakan organisasi sedang digunakan, konfigurasikan setelan pemindaian untuk sumber daya yang dikelola kebijakan tetapi tidak dapat mengaktifkan atau menonaktifkan jenis pemindaian yang dikelola kebijakan itu sendiri.

Tindakan akun anggota

Akun anggota dapat melihat dan mengambil informasi tentang akun mereka di Amazon Inspector, sementara pengaturan untuk akun mereka dikelola oleh administrator yang didelegasikan. Akun anggota dalam organisasi dapat melakukan tindakan berikut di Amazon Inspector:

- Aktifkan Amazon Inspector untuk akun mereka sendiri.
- Lihat cakupan sumber daya untuk akun mereka sendiri.
- Lihat detail temuan untuk akun mereka sendiri.
- Lihat pengaturan durasi pemindaian ulang otomatis gambar kontainer ECR untuk akun mereka sendiri.
- Tentukan lima jalur kustom untuk inspeksi mendalam Amazon Inspector untuk EC2 yang akan digunakan untuk akun masing-masing. Jalur ini dipindai selain jalur kustom apa pun yang telah ditentukan oleh administrator yang didelegasikan untuk organisasi. Untuk informasi selengkapnya tentang mengonfigurasi jalur inspeksi mendalam, lihatJalur khusus untuk inspeksi mendalam Amazon Inspector.
- Lihat jalur kustom yang ditetapkan oleh administrator yang didelegasikan untuk inspeksi mendalam Amazon Inspector.
- Ekspor SBOMs untuk sumber daya apa pun yang terkait dengan akun mereka.
- Lihat mode pemindaian untuk akun mereka.
- Buat dan kelola konfigurasi pemindaian CIS untuk akun mereka.
- Lihat hasil pemindaian CIS untuk sumber daya di akun mereka, termasuk yang dijadwalkan oleh administrator yang didelegasikan.
- Aktifkan jenis pemindaian yang tidak dikelola oleh kebijakan organisasi. Jenis pemindaian yang dikelola kebijakan tidak dapat diaktifkan atau dinonaktifkan oleh akun anggota.

> **Note**
>
> Setelah aktivasi, Amazon Inspector hanya dapat dinonaktifkan oleh akun administrator yang didelegasikan.

Menunjuk akun administrator yang didelegasikan untuk Amazon Inspector

Administrator yang didelegasikan adalah akun yang mengelola layanan untuk organisasi. Topik ini menjelaskan cara menunjuk administrator yang didelegasikan untuk Amazon Inspector.

Pertimbangan-pertimbangan

Sebelum menunjuk administrator yang didelegasikan, perhatikan hal berikut:

Administrator yang didelegasikan dapat mengelola maksimal 10.000 anggota.

Jika melebihi 10.000 akun anggota, Anda menerima pemberitahuan melalui Dashboard CloudWatch Personal Health Amazon dan email ke akun administrator yang didelegasikan.

> **Note**
>
> Jika Amazon Inspector diaktifkan melalui AWS Organizations kebijakan untuk organisasi dengan lebih dari 10.000 akun (hingga 50.000), kebijakan ini berlaku untuk semua akun. Namun, hanya 10.000 akun yang akan dikaitkan dengan organisasi Amazon Inspector. yaitu administrator yang didelegasikan dapat melihat temuan dan status akun hanya untuk 10.000 akun ini di konsol Amazon Inspector.

Administrator yang didelegasikan adalah Regional.

Amazon Inspector adalah layanan Regional. Anda harus mengulangi langkah-langkah dalam prosedur di setiap AWS Region tempat Anda berencana untuk menggunakan Amazon Inspector.

Sebuah organisasi hanya dapat memiliki satu administrator yang didelegasikan.

Jika menetapkan akun sebagai administrator yang didelegasikan dalam satu akun AWS Region, akun tersebut harus menjadi administrator yang didelegasikan di semua akun lainnya. Wilayah AWS

Mengubah administrator yang didelegasikan tidak menonaktifkan Amazon Inspector untuk akun anggota.

Jika Anda menghapus administrator yang didelegasikan, akun anggota menjadi akun mandiri dan pengaturan pemindaian tidak terpengaruh.

AWS Organisasi Anda harus mengaktifkan semua fitur.

> Ini adalah pengaturan default untuk AWS Organizations. Jika tidak diaktifkan, lihat Mengaktifkan semua fitur di organisasi Anda.

Kebijakan organisasi lebih diutamakan daripada pengaturan administrator yang didelegasikan.

> Jika organisasi Anda menggunakan AWS Organizations kebijakan untuk mengaktifkan Amazon Inspector, setelan kebijakan menentukan jenis pemindaian yang diaktifkan. Sebaiknya tentukan administrator yang didelegasikan sebelum membuat kebijakan organisasi untuk memastikan tata kelola yang konsisten. Untuk informasi selengkapnya, lihat Model tata kelola kebijakan organisasi.

Izin yang diperlukan untuk menetapkan administrator yang didelegasikan

Anda harus memiliki izin untuk mengaktifkan Amazon Inspector dan menunjuk administrator yang didelegasikan Amazon Inspector. Tambahkan pernyataan berikut di akhir kebijakan IAM Anda untuk memberikan izin ini. Untuk informasi selengkapnya, lihat Mengelola kebijakan IAM.

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

Menunjuk administrator yang didelegasikan untuk organisasi Anda AWS

Prosedur berikut menjelaskan cara menunjuk administrator yang didelegasikan untuk organisasi Anda. Sebelum Anda menyelesaikan prosedur, pastikan Anda berada di organisasi yang sama dengan akun anggota yang ingin dikelola oleh administrator yang didelegasikan.

> **Note**
>
> Anda harus menggunakan akun AWS Organizations manajemen untuk menyelesaikan prosedur ini. Hanya akun AWS Organizations manajemen yang dapat menunjuk administrator yang didelegasikan. Izin mungkin diperlukan untuk menunjuk administrator yang didelegasikan. Untuk informasi selengkapnya, lihat Izin yang diperlukan untuk menetapkan administrator yang didelegasikan.

Saat Anda mengaktifkan Amazon Inspector untuk pertama kalinya, Amazon Inspector membuat `AWSServiceRoleForAmazonInspector` peran yang ditautkan layanan untuk akun tersebut. Untuk informasi tentang cara Amazon Inspector menggunakan peran terkait layanan, lihat. Menggunakan peran tertaut layanan untuk Amazon Inspector

Console

Untuk menunjuk administrator yang didelegasikan untuk Amazon Inspector

1. Masuk ke akun AWS Organizations manajemen, lalu buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
2. Gunakan AWS Region pemilih untuk menentukan AWS Region di mana Anda ingin menunjuk administrator yang didelegasikan.
3. Dari panel navigasi, pilih Pengaturan umum.
4. Di bawah Administrator yang didelegasikan, masukkan ID 12 digit yang ingin Akun AWS Anda tetapkan sebagai administrator yang didelegasikan.
5. Pilih Delegasi, lalu pilih Delegasi lagi.

Saat Anda menunjuk administrator yang didelegasikan, semua jenis pemindaian diaktifkan untuk akun secara default. Jika Anda ingin mengaktifkan Amazon Inspector untuk akun AWS Organizations manajemen, selesaikan prosedur berikut.

Untuk mengaktifkan Amazon Inspector untuk akun manajemen AWS Organizations

1. Masuk ke akun administrator yang didelegasikan, lalu buka konsol https://console.aws.amazon.com/inspector/ Amazon Inspector di v2/home.
2. Dari panel navigasi, pilih Manajemen akun.
3. Di bawah Akun, pilih akun AWS Organizations manajemen, lalu pilih Aktifkan.

4. Pilih jenis pemindaian yang ingin Anda aktifkan untuk akun AWS Organizations manajemen, lalu pilih Kirim.

API

Menunjuk administrator yang didelegasikan menggunakan API

- Jalankan operasi EnableDelegatedAdminAccountAPI menggunakan kredensi akun manajemen Organizations. Akun AWS Anda juga dapat menggunakan AWS Command Line Interface untuk melakukan ini dengan menjalankan perintah CLI berikut:. `aws inspector2 enable-delegated-admin-account --delegated-admin-account-id 11111111111`

> **Note**
>
> Pastikan untuk menentukan ID akun akun yang ingin Anda jadikan administrator delegasi Amazon Inspector.

Mengaktifkan pemindaian Amazon Inspector untuk akun anggota

Anda dapat mengaktifkan Amazon Inspector untuk akun anggota di organisasi Anda melalui beberapa metode. Metode yang Anda pilih tergantung pada persyaratan tata kelola dan struktur organisasi Anda.

AWS Organizations kebijakan (Direkomendasikan untuk tata kelola terpusat)

Gunakan AWS Organizations kebijakan untuk mengaktifkan Amazon Inspector secara otomatis di seluruh organisasi Anda dengan kontrol terpusat. Pendekatan ini memastikan cakupan pemindaian yang konsisten dan secara otomatis berlaku untuk akun baru. Untuk petunjuk terperinci, lihat AWS Organizations dokumentasi untuk membuat kebijakan Amazon Inspector.

Aktivasi administrator yang didelegasikan

Sebagai administrator yang didelegasikan, Anda dapat mengaktifkan Amazon Inspector secara manual untuk akun anggota tertentu atau semua akun anggota melalui konsol Amazon Inspector atau API. Pendekatan ini memberikan fleksibilitas ketika kebijakan organisasi tidak digunakan.

Aktivasi mandiri akun anggota

Akun anggota dapat mengaktifkan Amazon Inspector untuk akun mereka sendiri jika tidak dibatasi oleh kebijakan organisasi. Setelah diaktifkan, akun menjadi terkait dengan administrator yang didelegasikan.

Aktifkan pemindaian untuk akun anggota

Prosedur berikut menjelaskan cara mengaktifkan pemindaian akun anggota menggunakan metode administrator dan akun anggota yang didelegasikan. Untuk informasi tentang jenis pemindaian Amazon Inspector, lihat. Jenis pemindaian otomatis di Amazon Inspector

Untuk secara otomatis mengaktifkan pemindaian untuk semua akun anggota

1. Masuk menggunakan kredensil akun administrator yang didelegasikan, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan pemilih wilayah untuk memilih AWS Region tempat Anda ingin mengaktifkan pemindaian untuk semua akun anggota.
3. Dari panel navigasi, pilih Manajemen akun. Tab Akun menampilkan semua akun anggota yang terkait dengan akun AWS Organizations manajemen.
4. Di bawah Organisasi, pilih kotak di sebelah Nomor akun. Kemudian pilih Aktifkan untuk memilih opsi pemindaian mana yang ingin Anda terapkan ke akun anggota. Anda dapat memilih jenis pemindaian berikut:

   - Pemindaian Amazon EC2
   - Pemindaian ECR Amazon
   - Pemindaian standar Lambda
   - Pemindaian kode Lambda

   - Setelah Anda memilih jenis pemindaian yang Anda inginkan, pilih Simpan.

> **Note**
>
> Jika Anda memiliki beberapa halaman akun, Anda harus mengulangi langkah ini di setiap halaman. Anda dapat memilih ikon roda gigi untuk mengubah jumlah akun yang ditampilkan di setiap halaman.

5. Aktifkan pengaturan Aktifkan Inspector secara otomatis untuk akun anggota baru, dan pilih opsi pemindaian yang ingin Anda terapkan ke akun anggota baru yang ditambahkan ke organisasi Anda. Anda dapat memilih jenis pemindaian berikut:

   - Pemindaian Amazon EC2
   - Pemindaian ECR Amazon
   - Pemindaian standar Lambda
   - Pemindaian kode Lambda

   - Setelah Anda memilih jenis pemindaian pilihan Anda, pilih Aktifkan.

> **Note**
>
> Pengaturan Automatic activate Inspector for new member accounts mengaktifkan Amazon Inspector untuk semua anggota organisasi Anda yang akan datang.
> Jika jumlah akun anggota lebih dari 5.000, pengaturan ini secara otomatis dimatikan.
> Jika jumlah total akun anggota berkurang menjadi kurang dari 5.000, pengaturan secara otomatis diaktifkan kembali.

6. (Disarankan) Ulangi setiap langkah ini di setiap AWS Region tempat Anda ingin mengaktifkan pemindaian akun anggota.

Untuk mengaktifkan pemindaian akun anggota tertentu

1. Masuk menggunakan kredensil akun administrator yang didelegasikan, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan pemilih wilayah untuk memilih AWS Region tempat Anda ingin mengaktifkan pemindaian untuk semua akun anggota.

3. Dari panel navigasi, pilih Manajemen akun. Tab Akun menampilkan semua akun anggota yang terkait dengan akun AWS Organizations manajemen.
4. Di bawah Organisasi, pilih kotak di samping setiap nomor akun anggota yang ingin Anda aktifkan pemindaian. Kemudian pilih Aktifkan untuk memilih opsi pemindaian mana yang ingin Anda terapkan ke akun anggota. Anda dapat memilih jenis pemindaian berikut:

   - Pemindaian Amazon EC2
   - Pemindaian ECR Amazon
   - Pemindaian standar Lambda
   - Pemindaian kode Lambda

   - Setelah Anda memilih jenis pemindaian yang Anda inginkan, pilih Simpan.

   > **Note**
   >
   > Jika Anda memiliki beberapa halaman akun, Anda harus mengulangi langkah ini di setiap halaman. Anda dapat memilih ikon roda gigi untuk mengubah jumlah akun yang ditampilkan di setiap halaman.

5. (Disarankan) Ulangi setiap langkah ini di masing-masing AWS Region tempat Anda ingin mengaktifkan pemindaian untuk anggota tertentu.

Untuk mengaktifkan pemindaian sebagai akun anggota

1. Masuk menggunakan kredensional Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan pemilih wilayah untuk memilih AWS Region tempat Anda ingin mengaktifkan pemindaian untuk semua akun anggota.
3. Dari panel navigasi, pilih Manajemen akun. Tab Akun menampilkan semua akun anggota yang terkait dengan akun AWS Organizations manajemen.
4. Di bawah Organisasi, pilih kotak di sebelah nomor akun Anda. Kemudian pilih Aktifkan untuk memilih opsi pemindaian mana yang ingin Anda terapkan. Anda dapat memilih jenis pemindaian berikut:

   - Pemindaian Amazon EC2

- Pemindaian ECR Amazon
- Pemindaian standar Lambda
- Pemindaian kode Lambda

- Setelah Anda memilih jenis pemindaian yang Anda inginkan, pilih Simpan.

5. (Disarankan) Ulangi langkah-langkah ini di setiap Wilayah tempat Anda ingin mengaktifkan pemindaian untuk akun anggota Anda.

> **Note**
>
> Jika akun AWS Organizations manajemen Anda memiliki akun administrator yang didelegasikan untuk Amazon Inspector, Anda dapat mengaktifkan akun Anda sebagai akun anggota untuk melihat detail pemindaian.

> **Penting:**
>
> Jika kebijakan organisasi mengelola pengaktifan Amazon Inspector untuk akun Anda, administrator yang didelegasikan dan akun anggota tidak dapat mengubah jenis pemindaian yang dikelola kebijakan menggunakan Amazon Inspector. enablement/disablement APIs Permintaan API akan gagal dengan kesalahan yang menunjukkan sumber daya dikelola oleh kebijakan organisasi. Anda masih dapat mengaktifkan jenis pemindaian tambahan yang tidak dikelola oleh kebijakan.

Memutus akun anggota di Amazon Inspector

Sebagai administrator yang didelegasikan, Anda mungkin perlu memisahkan akun anggota dari akun Anda. Saat Anda memisahkan akun anggota, Amazon Inspector masih diaktifkan di akun tersebut, dan akun tersebut menjadi akun mandiri. Anda juga tidak memiliki izin untuk mengelola Amazon Inspector untuk akun tersebut lagi. Namun, Anda dapat mengaitkan akun anggota yang sebelumnya tidak terkait dengan akun Anda kapan saja. Bagian ini menjelaskan cara memisahkan akun anggota sebagai administrator yang didelegasikan.

> **Note**
>
> Untuk memisahkan akun yang dikelola kebijakan, seharusnya tidak ada kebijakan organisasi Amazon Inspector yang dilampirkan ke akun tersebut untuk jenis pemindaian.

Console

Untuk memisahkan akun anggota menggunakan konsol

1. Masuk menggunakan kredensil akun administrator yang didelegasikan, lalu buka konsol Amazon Inspector di v2/home https://console.aws.amazon.com/inspector/
2. Gunakan pemilih wilayah untuk memilih AWS Region tempat Anda ingin memisahkan akun anggota.
3. Dari panel navigasi, pilih Manajemen akun.
4. Di bawah Organisasi, pilih kotak di samping setiap nomor akun yang ingin Anda pisahkan.
5. Pilih menu Tindakan, lalu pilih Disassociate account.

API

Untuk memisahkan akun anggota menggunakan API

Jalankan operasi DisassociateMemberAPI. Dalam permintaan, berikan akun yang IDs Anda lepaskan.

Menghapus administrator yang didelegasikan di Amazon Inspector

Anda mungkin perlu menghapus akun administrator yang didelegasikan Amazon Inspector. Anda dapat melakukan ini dari akun AWS Organizations manajemen. Saat Anda menghapus akun administrator yang didelegasikan Amazon Inspector, Amazon Inspector masih diaktifkan di akun dan di semua akun anggotanya. Akun administrator yang didelegasikan dan semua akun anggotanya menjadi akun mandiri dan mempertahankan pengaturan pemindaian asli mereka.

> **Note**
>
> Jika AWS Organizations kebijakan mengelola pengaktifan Amazon Inspector, menghapus administrator yang didelegasikan tidak akan memengaruhi penegakan kebijakan. Akun akan

tetap diaktifkan sesuai dengan pengaturan kebijakan organisasi, meskipun temuan akun anggota tidak akan lagi terlihat di konsol administrator yang didelegasikan pusat hingga administrator yang didelegasikan baru ditunjuk.

Bagian ini menjelaskan cara menghapus akun administrator yang didelegasikan.

Hapus administrator yang didelegasikan oleh Amazon Inspector

Prosedur berikut menjelaskan cara menghapus administrator delegasi Amazon Inspector dan cara mengaitkan akun anggota dari akun administrator yang didelegasikan.

Untuk selengkapnya tentang cara menetapkan admninistrator yang didelegasikan Amazon Inspector, lihat Menunjuk akun administrator yang didelegasikan untuk Amazon Inspector.

Note

Setelah Anda menetapkan administrator yang didelegasikan Amazon Inspector, administrator yang didelegasikan Amazon Inspector harus mengaitkan akun anggota secara manual.

Untuk menghapus administrator yang didelegasikan

1. Masuk ke Konsol Manajemen AWS menggunakan akun AWS Organizations manajemen.
2. Buka konsol Amazon Inspector di https://console.aws.amazon.com/inspector/ v2/home.
3. Gunakan pemilih wilayah untuk memilih AWS Region tempat Anda ingin menghapus administrator yang didelegasikan.
4. Dari panel navigasi, pilih Pengaturan umum.
5. Di bawah Administrator yang didelegasikan, pilih Hapus, lalu konfirmasikan tindakan Anda.

Untuk mengasosiasikan anggota dengan administrator baru yang didelegasikan

1. Masuk menggunakan kredensil akun administrator yang didelegasikan, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Gunakan pemilih wilayah untuk memilih AWS Region tempat Anda ingin mengaitkan anggota.
3. Dari panel navigasi, pilih Manajemen akun.
4. Di bawah Organisasi, pilih kotak di sebelah Nomor akun.

5. Pilih Tindakan, lalu pilih Tambah anggota.

Menandai sumber daya Amazon Inspector

Tag adalah label yang Anda tambahkan ke sumber AWS daya. Tag membantu Anda mengkategorikan AWS sumber daya berdasarkan kriteria tertentu. Tag terdiri dari pasangan kunci-nilai. Kunci tag adalah label umum. Nilai tag adalah deskripsi dari kunci tag. Dengan Amazon Inspector, Anda dapat menandai aturan penekanan dan konfigurasi pemindaian CIS. Anda dapat menambahkan sebanyak 50 tag ke setiap sumber daya Amazon Inspector Anda.

Menandai dasar-dasar

Satu tag terdiri dari pasangan nilai kunci. Kunci tag adalah label umum. Nilai tag adalah deskripsi dari kunci tag. Topik ini menjelaskan dasar-dasar penandaan sumber daya Amazon Inspector. Saat menandai sumber daya Amazon Inspector, pertimbangkan hal berikut:

- Anda dapat menandai aturan penekanan dan konfigurasi pemindaian CIS.
- Anda dapat menambahkan sebanyak 50 tag ke setiap sumber daya Amazon Inspector Anda.
- Kunci tanda harus unik.
- Kunci tag hanya dapat memiliki satu nilai tag.
- Kunci tag dan nilai tag dapat memiliki maksimum 128 karakter UTF-8. Karakter dapat berupa huruf, angka, spasi, atau simbol berikut: `_ . : / = + -@`.
- Anda tidak dapat menggunakan `aws` awalan di salah satu tag Anda atau memodifikasi tag dengan awalan ini. Tag dengan `aws` awalan dicadangkan untuk digunakan oleh AWS.
- Tag yang ditetapkan ke sumber daya Amazon Inspector hanya tersedia di AWS akun Anda dan di AWS Region tempat Anda membuatnya.
- Saat Anda menghapus sumber daya, semua tag yang terkait dengannya juga akan dihapus.

Untuk informasi selengkapnya tentang tag, lihat Praktik dan strategi terbaik di Panduan Pengguna AWS Sumber Daya Tag dan Editor Tag.

> **Note**
>
> Tag tidak dimaksudkan untuk menyimpan informasi rahasia atau sensitif. Jangan pernah menggunakan tag untuk menyimpan jenis data ini. Tag dapat diakses dari AWS layanan lain.

Menambahkan tanda

Anda dapat menambahkan tag ke sumber daya Amazon Inspector. Sumber daya ini mencakup aturan penekanan dan konfigurasi pemindaian CIS. Tag membantu Anda mengkategorikan AWS sumber daya berdasarkan kriteria tertentu. Topik ini menjelaskan cara menambahkan tag ke sumber daya Amazon Inspector.

Menambahkan tag ke sumber daya Amazon Inspector

Anda dapat menandai aturan penekanan dan konfigurasi pemindaian CIS. Prosedur berikut menjelaskan cara menambahkan tag di konsol dan dengan Amazon Inspector API.

Menambahkan tag di konsol

Anda dapat menambahkan tag ke sumber daya Amazon Inspector di konsol.

Menambahkan tag ke aturan penindasan

Anda dapat menambahkan tag ke aturan penekanan selama pembuatan. Untuk informasi selengkapnya, lihat Membuat aturan penindasan.

Anda juga dapat mengedit aturan penindasan untuk menyertakan tag. Untuk informasi selengkapnya, lihat Mengedit aturan penindasan.

Menambahkan tag ke konfigurasi pemindaian CIS

Anda dapat menambahkan tag ke konfigurasi pemindaian CIS selama pembuatan. Untuk informasi selengkapnya, lihat Membuat konfigurasi pemindaian CIS.

Anda juga dapat mengedit konfigurasi pemindaian CIS untuk menyertakan tag. Untuk informasi selengkapnya, lihat Mengedit konfigurasi pemindaian CIS.

Menambahkan tag dengan Amazon Inspector API

Anda dapat menambahkan tag ke sumber daya Amazon Inspector dengan Amazon Inspector API.

Menambahkan tag ke sumber daya Amazon Inspector

Gunakan `TagResource` API untuk menambahkan tag ke sumber daya Amazon Inspector. Anda harus menyertakan ARN sumber daya dan pasangan kunci-nilai untuk tag dalam perintah. Contoh perintah berikut menggunakan ARN sumber daya kosong untuk filter penindasan. Kuncinya adalah

`CostAllocation` dan nilainya adalah`dev`. Untuk informasi tentang jenis sumber daya untuk Amazon Inspector, lihat Tindakan, sumber daya, dan kunci kondisi untuk Amazon Inspector2 di Referensi Otorisasi Layanan.

```
aws inspector2 tag-resource \
--resource-arn "arn:${Partition}:inspector2:${Region}:${Account}:owner/${OwnerId}/
filter/${FilterId}" \
--tags CostAllocation=dev \
--region us-west-2
```

Menambahkan tag ke aturan penekanan selama pembuatan

Gunakan `CreateFilter` API untuk menambahkan tag ke aturan penekanan selama pembuatan.

```
aws inspector2 create-filter \
--name "ExampleSuppressionRuleECR" \
--action SUPPRESS \
--filter-criteria 'resourceType=[{comparison="EQUALS", value="AWS_ECR_IMAGE"}]' \
--tags Owner=ApplicationSecurity \
--region us-west-2
```

Menambahkan tag ke konfigurasi pemindaian CIS

Gunakan `CreateCisScanConfiguration` API untuk menambahkan tag ke konfigurasi pemindaian 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
```

Menghapus tanda

Anda dapat menghapus tag dari sumber daya Amazon Inspector. Sumber daya ini mencakup aturan penekanan dan konfigurasi pemindaian CIS. Tag membantu Anda mengkategorikan AWS sumber daya berdasarkan kriteria tertentu. Topik ini menjelaskan cara menghapus tag dari sumber daya Amazon Inspector.

Menghapus tag dari sumber daya Amazon Inspector

Anda dapat menghapus tag dari aturan penekanan dan konfigurasi pemindaian CIS. Prosedur berikut menjelaskan cara menghapus tag di konsol dan dengan Amazon Inspector API.

Menghapus tag di konsol

Anda dapat menghapus tag dari sumber daya Amazon Inspector di konsol.

Menghapus tag dari aturan penindasan

Anda dapat menghapus tag dari aturan penekanan dengan mengedit aturan penekanan agar tidak lagi menyertakan tag. Untuk informasi selengkapnya, lihat Mengedit aturan penindasan.

Menghapus tag dari konfigurasi pemindaian CIS

Anda dapat menghapus tag dari konfigurasi pemindaian CIS dengan mengedit konfigurasi pemindaian CIS agar tidak lagi menyertakan tag. Untuk informasi selengkapnya, lihat Mengedit konfigurasi pemindaian CIS.

Menghapus tag dengan Amazon Inspector API

Anda dapat menghapus tag dari sumber daya Amazon Inspector dengan Amazon Inspector API.

Menghapus tag dari sumber daya Amazon Inspector

Gunakan `UntagResource` API untuk menghapus tag dari sumber daya Amazon Inspector.

Cuplikan berikut menunjukkan contoh cara menghapus tag dari sumber daya Amazon Inspector menggunakan. `UntagResource` Anda harus menyertakan ARN sumber daya dan kunci untuk tag dalam perintah. Contoh berikut menggunakan ARN sumber daya kosong untuk filter penindasan. Kuncinya adalah `CostAllocation`. Untuk informasi tentang jenis sumber daya untuk Amazon Inspector, lihat Tindakan, sumber daya, dan kunci kondisi untuk Amazon Inspector2 di Referensi Otorisasi Layanan.

```
aws inspector2 untag-resource \
--resource-arn "arn:${Partition}:inspector2:${Region}:${Account}:owner/${OwnerId}/cis-configuration/${CISScanConfigurationId}" \
--tag-keys CostAllocation \
--region us-west-2
```

Pemantauan Penggunaan dan Biaya di Amazon Inspector

Anda dapat menggunakan konsol Amazon Inspector dan API untuk memproyeksikan biaya Amazon Inspector bulanan untuk lingkungan Anda. Jika Anda administrator Amazon Inspector untuk lingkungan beberapa akun, Anda dapat melihat total biaya untuk lingkungan dan metrik biaya untuk semua akun anggota. Bagian ini menjelaskan cara mengakses statistik penggunaan dan menghitung biaya penggunaan.

Menggunakan konsol penggunaan

Anda dapat menilai penggunaan dan biaya yang diproyeksikan untuk Amazon Inspector dari konsol.

Untuk mengakses statistik penggunaan

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dengan menggunakan AWS Region pemilih di sudut kanan atas halaman, pilih Wilayah yang ingin Anda pantau biaya.
3. Pada panel navigasi, pilih Penggunaan.

Di tab Berdasarkan akun Anda akan melihat total biaya yang diproyeksikan berdasarkan periode 30 hari yang tercantum dalam penggunaan Akun. Dalam tabel di bawah kolom Biaya yang diproyeksikan, pilih nilai untuk melihat rincian penggunaan berdasarkan jenis pemindaian untuk akun tersebut. Di panel detail ini Anda juga dapat melihat jenis pemindaian mana yang memiliki uji coba gratis yang aktif untuk akun tersebut.

Jika Anda adalah administrator yang didelegasikan untuk organisasi, Anda akan melihat baris dalam tabel untuk setiap akun dalam organisasi Anda. Jika akun di organisasi Anda dipisahkan, konsol menunjukkan biaya yang diproyeksikan sebagai -.

Di tab By scan type Anda dapat melihat rincian penggunaan aktual sejauh ini dalam periode 30 hari saat ini berdasarkan jenis pemindaian. Ini adalah informasi yang digunakan untuk menghitung biaya yang diproyeksikan di tab By account.

Jika Anda adalah administrator yang didelegasikan untuk organisasi, Anda dapat melihat penggunaan untuk setiap akun di organisasi Anda.

Di tab ini, Anda dapat memperluas salah satu panel berikut untuk statistik penggunaan:

EC2 Pemindaian Amazon

Konsol penggunaan Amazon Inspector melacak metrik berikut untuk pemindaian berbasis agen dan pemindaian tanpa agen:

- Instances (Avg) — Amazon Inspector menggunakan jam cakupan untuk menghitung jumlah rata-rata sumber daya EC2 untuk pemindaian instans. Rata-rata adalah total jam pertanggungan dibagi 720 jam (jumlah jam dalam periode 30 hari).
- Jam cakupan — untuk EC2 pemindaian Amazon, ini adalah jumlah total jam dalam 30 hari terakhir yang Amazon Amazon Inspector berikan cakupan aktif untuk setiap EC2 instans di akun. Misalnya EC2 , jam pertanggungan adalah jam dari saat Amazon Inspector menemukan instans hingga dihentikan atau dihentikan, atau dikecualikan dari pemindaian berdasarkan tag. (saat Anda memulai ulang instance yang dihentikan atau menghapus tag pengecualian, Amazon Inspector melanjutkan cakupan dan jam cakupan untuk instance tersebut akan terus bertambah).

Pemindaian Instans CIS — Jumlah total pemindaian CIS yang dilakukan untuk instance di akun.

Pemindaian ECR Amazon

Pemindaian awal — Jumlah total pemindaian gambar pertama kali di akun dalam 30 hari terakhir.

Rescan — Jumlah total pemindaian ulang untuk gambar di akun dalam 30 hari terakhir. Pemindaian ulang adalah pemindaian apa pun yang dilakukan pada gambar ECR yang sebelumnya dipindai Amazon Inspector. Jika Anda telah mengonfigurasi repositori ECR untuk pemindaian berkelanjutan, pemindaian ulang terjadi secara otomatis saat Amazon Inspector menambahkan Common Vulnerabilities and Exposures (CVE) baru ke database-nya.

Pemindaian Lambda

Konsol penggunaan Amazon Inspector melacak metrik berikut untuk pemindaian standar Lambda dan pemindaian kode Lambda:

- Jumlah fungsi Lambda (Rata-rata) - Amazon Inspector menggunakan jam cakupan untuk menghitung jumlah rata-rata fungsi untuk pemindaian fungsi Lambda. Rata-rata adalah total jam pertanggungan dibagi 720 jam (jumlah jam dalam periode 30 hari).
- Jam cakupan - Untuk pemindaian fungsi Lambda, ini adalah jumlah total jam dalam 30 hari terakhir Amazon Inspector menyediakan cakupan aktif untuk setiap fungsi Lambda dalam sebuah akun. Untuk AWS Lambda fungsi, jam cakupan dihitung dari saat Amazon Inspector

menemukan fungsi hingga saat dihapus atau dikecualikan dari pemindaian. Jika fungsi yang dikecualikan disertakan lagi, jam cakupan untuk fungsi tersebut akan terus bertambah.

Memahami bagaimana Amazon Inspector menghitung biaya penggunaan

Biaya yang disediakan oleh Amazon Inspector adalah perkiraan, bukan biaya aktual, sehingga mungkin berbeda dari yang ada di konsol Anda AWS Billing .

Perhatikan hal berikut tentang cara Amazon Inspector menghitung biaya di halaman Penggunaan:

- Biaya penggunaan hanya mencerminkan wilayah saat ini. Harga per jenis pemindaian bervariasi menurut AWS Wilayah, untuk meninjau harga pasti per wilayah, lihat Harga untuk Amazon Inspector
- Semua proyeksi penggunaan dibulatkan ke dolar AS terdekat.
- Diskon tidak termasuk dalam biaya yang diproyeksikan.
- Biaya yang diproyeksikan mewakili total biaya untuk periode penggunaan 30 hari per jenis pemindaian. Jika ada kurang dari 30 hari penggunaan untuk akun, Amazon Inspector memproyeksikan biaya setelah 30 hari seolah-olah ada sumber daya yang saat ini tercakup akan tetap ditanggung selama sisa periode 30 hari.
- Biaya per jenis pemindaian dihitung berdasarkan hal berikut:
  - EC2 pemindaian: biaya mencerminkan jumlah rata-rata EC2 instans yang dicakup oleh Amazon Inspector dalam 30 hari terakhir.
  - Pemindaian kontainer ECR: biaya mencerminkan jumlah pemindaian gambar awal+pemindaian ulang gambar dalam 30 hari terakhir.
  - Pemindaian standar Lambda: biaya mencerminkan jumlah rata-rata fungsi Lambda yang dicakup oleh Amazon Inspector dalam 30 hari terakhir.
  - Pemindaian kode Lambda: biaya mencerminkan jumlah rata-rata fungsi Lambda yang dicakup oleh Amazon Inspector dalam 30 hari terakhir.

Tentang uji coba gratis Amazon Inspector

Di Amazon Inspector, setiap jenis pemindaian memiliki jejak gratis. Saat Anda mengaktifkan jenis pemindaian, Anda secara otomatis mendaftar dalam uji coba gratis 15 hari untuk jenis pemindaian

tersebut. Setelah uji coba gratis dimulai, secara otomatis kedaluwarsa dalam 15 hari, bahkan jika Anda menonaktifkan jenis pemindaian.

> **Note**
>
> Uji coba gratis tidak berlaku untuk pemindaian CIS.

Keamanan di Amazon Inspector

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Model tanggung jawab bersama menjelaskan hal ini sebagai keamanan dari cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program AWS Kepatuhan Program AWS Kepatuhan . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon Inspector, lihat AWS Layanan dalam Lingkup berdasarkan AWS Layanan Program Kepatuhan dalam Lingkup oleh Program .
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain, yang mencakup sensitivitas data Anda, persyaratan perusahaan Anda, serta undang-undang dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan Amazon Inspector. Topik berikut menunjukkan kepada Anda cara mengonfigurasi Amazon Inspector untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya Amazon Inspector Anda.

Topik

Perlindungan data di Amazon Inspector

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di Amazon Inspector. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredensil dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang penggunaan CloudTrail jejak untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di AWS CloudTrail Panduan Pengguna.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan Amazon Inspector atau lainnya Layanan AWS menggunakan konsol, API AWS CLI, atau. AWS SDKs Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan

atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Topik

Enkripsi saat diam

Secara default, Amazon Inspector menyimpan data saat istirahat menggunakan solusi AWS enkripsi. Amazon Inspector mengenkripsi data, seperti berikut ini:

- Inventaris sumber daya dikumpulkan dengan AWS Systems Manager.
- Inventaris sumber daya diuraikan dari gambar Amazon Elastic Container Registry
- Temuan keamanan yang dihasilkan menggunakan kunci enkripsi yang AWS dimiliki dari AWS Key Management Service

Anda tidak dapat mengelola, menggunakan, atau melihat kunci AWS yang dimiliki. Namun, Anda tidak perlu mengambil tindakan atau mengubah program untuk melindungi kunci yang mengenkripsi data Anda. Untuk informasi selengkapnya, lihat kunci AWS yang dimiliki.

Jika Anda menonaktifkan Amazon Inspector, Amazon Inspector akan menghapus secara permanen semua sumber daya yang disimpan atau dipelihara untuk Anda, seperti inventaris yang dikumpulkan dan temuan keamanan.

Enkripsi saat istirahat untuk kode dalam temuan Anda

Untuk pemindaian kode Amazon Inspector Lambda, Amazon Inspector bermitra dengan Amazon Q untuk memindai kode Anda dari kerentanan. Ketika kerentanan terdeteksi, Amazon Q mengekstrak cuplikan kode Anda yang berisi kerentanan dan menyimpan kode tersebut hingga Amazon Inspector meminta akses. Secara default, Amazon Q menggunakan kunci yang AWS dimiliki untuk mengenkripsi kode yang diekstrak. Namun, Anda dapat mengonfigurasi Amazon Inspector untuk menggunakan kunci yang dikelola pelanggan AWS KMS Anda sendiri untuk enkripsi.

Alur kerja berikut menjelaskan cara Amazon Inspector menggunakan kunci yang Anda konfigurasikan untuk mengenkripsi kode Anda:

1. Anda menyediakan AWS KMS kunci ke Amazon Inspector menggunakan Amazon UpdateEncryptionKeyInspector API.
2. Amazon Inspector meneruskan informasi tentang kunci Anda AWS KMS ke Amazon Q, dan Amazon Q menyimpan informasi untuk digunakan di masa mendatang.
3. Amazon Q menggunakan kunci KMS yang Anda konfigurasikan di Amazon Inspector melalui kebijakan kunci.
4. Amazon Q membuat kunci data terenkripsi dari AWS KMS kunci Anda dan menyimpannya. Kunci data ini digunakan untuk mengenkripsi data kode Anda yang disimpan oleh Amazon Q.
5. Saat Amazon Inspector meminta data dari pemindaian kode, Amazon Q menggunakan kunci KMS untuk mendekripsi kunci data. Saat Anda menonaktifkan Pemindaian Kode Lambda, Amazon Q menghapus kunci data terkait.

Izin untuk enkripsi kode dengan kunci yang dikelola pelanggan

Untuk enkripsi, Anda harus membuat kunci KMS dengan kebijakan yang menyertakan pernyataan yang memungkinkan Amazon Inspector dan Amazon Q untuk melakukan tindakan berikut.

- `kms:Decrypt`
- `kms:DescribeKey`
- `kms:Encrypt`
- `kms:GenerateDataKey`
- `kms:GenerateDataKeyWithoutPlainText`

Pernyataan kebijakan

Anda dapat menggunakan pernyataan kebijakan berikut saat membuat kunci KMS.

> **Note**
>
> Ganti *`account-id`* dengan Akun AWS ID 12 digit Anda. Ganti *`Region`* dengan AWS Region tempat Anda mengaktifkan pemindaian kode Amazon Inspector dan Lambda. Ganti *`role-ARN`* dengan Nama Sumber Daya Amazon untuk peran IAM Anda.

```
{
```

```
  "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"
      }
    }
  }
```

Pernyataan kebijakan diformat dalam JSON. Setelah Anda menyertakan pernyataan, tinjau kebijakan untuk memastikan sintaksnya valid. Jika pernyataan tersebut adalah pernyataan terakhir dalam kebijakan, tempatkan koma setelah tanda kurung kurung penutup untuk pernyataan sebelumnya. Jika pernyataan tersebut adalah pernyataan pertama atau di antara dua pernyataan yang ada dalam kebijakan, tempatkan koma setelah tanda kurung kurung penutup untuk pernyataan tersebut.

> **Note**
>
> Amazon Inspector tidak lagi mendukung hibah untuk mengenkripsi cuplikan kode yang diekstrak dari paket. Jika Anda menggunakan kebijakan berbasis hibah, Anda masih dapat

mengakses temuan Anda. Namun, jika Anda memperbarui atau mengatur ulang kunci KMS atau menonaktifkan Pemindaian Kode Lambda, Anda harus menggunakan kebijakan kunci KMS yang dijelaskan di bagian ini.

Jika Anda menyetel, memperbarui, atau mengatur ulang kunci enkripsi untuk akun Anda, Anda harus menggunakan kebijakan administrator Amazon Inspector, seperti kebijakan AWS terkelola. `AmazonInspector2FullAccess`

Mengkonfigurasi enkripsi dengan kunci yang dikelola pelanggan

Untuk mengonfigurasi enkripsi akun menggunakan kunci terkelola pelanggan, Anda harus menjadi administrator Amazon Inspector dengan izin yang diuraikan. Izin untuk enkripsi kode dengan kunci yang dikelola pelanggan Selain itu, Anda akan memerlukan AWS KMS kunci di AWS Wilayah yang sama dengan temuan Anda, atau kunci multi-wilayah. Anda dapat menggunakan kunci simetris yang ada di akun Anda atau membuat kunci terkelola pelanggan simetris dengan menggunakan AWS Management Console, atau. AWS KMS APIs Untuk informasi selengkapnya, lihat Membuat AWS KMS kunci enkripsi simetris di panduan AWS KMS pengguna.

> **Note**
>
> Efektif 13 Juni 2025, prinsip layanan dalam AWS KMS permintaan yang masuk CloudTrail selama cuplikan encryption/decryption kode berubah dari “codeguru-reviewer” menjadi “q”.

Menggunakan Amazon Inspector API untuk mengonfigurasi enkripsi

Untuk menyetel kunci enkripsi, UpdateEncryptionKeypengoperasian Amazon Inspector API saat masuk sebagai administrator Amazon Inspector. Dalam permintaan API, gunakan `kmsKeyId` bidang untuk menentukan ARN AWS KMS kunci yang ingin Anda gunakan. Untuk `scanType` masuk `CODE` dan `resourceType` masuk`AWS_LAMBDA_FUNCTION`.

Anda dapat menggunakan UpdateEncryptionKeyAPI untuk memeriksa tampilan AWS KMS kunci mana yang digunakan Amazon Inspector untuk enkripsi.

> **Note**
>
> Jika Anda mencoba menggunakan `GetEncryptionKey` ketika Anda belum menetapkan kunci terkelola pelanggan, operasi mengembalikan `ResourceNotFoundException` kesalahan yang berarti bahwa kunci yang AWS dimiliki sedang digunakan untuk enkripsi.

Jika Anda menghapus kunci atau mengubah kebijakan untuk menolak akses ke Amazon Inspector atau Amazon Q, Anda tidak akan dapat mengakses temuan kerentanan kode Anda dan pemindaian kode Lambda akan gagal untuk akun Anda.

Anda dapat menggunakan `ResetEncryptionKey` untuk melanjutkan menggunakan kunci yang AWS dimiliki untuk mengenkripsi kode yang diekstraksi sebagai bagian dari temuan Amazon Inspector Anda.

Enkripsi saat bergerak

AWS mengenkripsi semua data dalam perjalanan antara sistem AWS internal dan layanan lainnya AWS . AWS Systems Manager mengumpulkan data telemetri dari instans EC2 milik pelanggan yang dikirimkannya ke saluran yang dilindungi Transport Layer Security (TLS) untuk AWS penilaian. Temuan pemindaian fungsi Amazon ECR dan AWS Lambda yang dikirim ke Security Hub CSPM dienkripsi menggunakan saluran yang dilindungi TLS. Untuk informasi selengkapnya, lihat Perlindungan Data di Systems Manager untuk memahami cara SSM mengenkripsi data saat transit.

Identity and Access Management untuk Amazon Inspector

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat terautentikasi (masuk) dan berwenang (memiliki izin) untuk menggunakan sumber daya Amazon Inspector. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- Audiens
- Mengautentikasi dengan identitas
- Mengelola akses menggunakan kebijakan
- Cara kerja Amazon Inspector dengan IAM

- Contoh kebijakan berbasis identitas untuk Amazon Inspector
- AWS kebijakan terkelola untuk Amazon Inspector
- Menggunakan peran tertaut layanan untuk Amazon Inspector
- Pemecahan masalah identitas dan akses Amazon Inspector

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda berdasarkan peran Anda:

- Pengguna layanan - minta izin dari administrator Anda jika Anda tidak dapat mengakses fitur (lihat Pemecahan masalah identitas dan akses Amazon Inspector)
- Administrator layanan - tentukan akses pengguna dan mengirimkan permintaan izin (lihat Cara kerja Amazon Inspector dengan IAM)
- Administrator IAM - tulis kebijakan untuk mengelola akses (lihat Contoh kebijakan berbasis identitas untuk Amazon Inspector)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi sebagai Pengguna root akun AWS, pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk sebagai identitas federasi menggunakan kredensyal dari sumber identitas seperti AWS IAM Identity Center (Pusat Identitas IAM), autentikasi masuk tunggal, atau kredensyal. Google/ Facebook Untuk informasi selengkapnya tentang cara masuk, lihat Cara masuk ke Akun AWS Anda dalam Panduan Pengguna AWS Sign-In .

Untuk akses terprogram, AWS sediakan SDK dan CLI untuk menandatangani permintaan secara kriptografis. Untuk informasi selengkapnya, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang disebut pengguna Akun AWS root yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya. Kami

sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Untuk tugas yang memerlukan kredensial pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas terfederasi

Sebagai praktik terbaik, mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori perusahaan Anda, penyedia identitas web, atau Directory Service yang mengakses Layanan AWS menggunakan kredensyal dari sumber identitas. Identitas terfederasi mengambil peran yang memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami menyarankan AWS IAM Identity Center. Untuk informasi selengkapnya, lihat Apa itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dengan izin khusus untuk satu orang atau aplikasi. Sebaiknya gunakan kredensial sementara alih-alih pengguna IAM dengan kredensial jangka panjang. Untuk informasi selengkapnya, lihat Mewajibkan pengguna manusia untuk menggunakan federasi dengan penyedia identitas untuk mengakses AWS menggunakan kredensi sementara di Panduan Pengguna IAM.

Grup IAM menentukan kumpulan pengguna IAM dan mempermudah pengelolaan izin untuk pengguna dalam jumlah besar. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dengan izin khusus yang menyediakan kredensial sementara. Anda dapat mengambil peran dengan beralih dari pengguna ke peran IAM (konsol) atau dengan memanggil operasi AWS CLI atau AWS API. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM berguna untuk akses pengguna terfederasi, izin pengguna IAM sementara, akses lintas akun, akses lintas layanan, dan aplikasi yang berjalan di Amazon EC2. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan menentukan izin saat dikaitkan dengan identitas atau sumber daya. AWS mengevaluasi kebijakan ini ketika kepala sekolah membuat permintaan. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Menggunakan kebijakan, administrator menentukan siapa yang memiliki akses ke apa dengan mendefinisikan principal mana yang dapat melakukan tindakan pada sumber daya apa, dan dalam kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Administrator IAM membuat kebijakan IAM dan menambahkannya ke peran, yang kemudian dapat diambil oleh pengguna. Kebijakan IAM mendefinisikan izin terlepas dari metode yang Anda gunakan untuk melakukan operasinya.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang Anda lampirkan ke identitas (pengguna, grup, atau peran). Kebijakan ini mengontrol tindakan apa yang bisa dilakukan oleh identitas tersebut, terhadap sumber daya yang mana, dan dalam kondisi apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan yang dikelola pelanggan dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat berupa kebijakan inline (disematkan langsung ke dalam satu identitas) atau kebijakan terkelola (kebijakan mandiri yang dilampirkan pada banyak identitas). Untuk mempelajari cara memilih antara kebijakan terkelola dan kebijakan inline, lihat Pilih antara kebijakan terkelola dan kebijakan inline dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contohnya termasuk kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Anda harus menentukan principal dalam kebijakan berbasis sumber daya.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola dari IAM dalam kebijakan berbasis sumber daya.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang dapat menetapkan izin maksimum yang diberikan oleh jenis kebijakan yang lebih umum:

- Batasan izin – Menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM. Untuk informasi selengkapnya, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCPs) — Tentukan izin maksimum untuk organisasi atau unit organisasi di AWS Organizations. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam Panduan Pengguna AWS Organizations .
- Kebijakan kontrol sumber daya (RCPs) — Tetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda. Untuk informasi selengkapnya, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan lanjutan yang diteruskan sebagai parameter saat membuat sesi sementara untuk peran atau pengguna terfederasi. Untuk informasi selengkapnya, lihat Kebijakan sesi dalam Panduan Pengguna IAM.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan Pengguna IAM.

Cara kerja Amazon Inspector dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke Amazon Inspector, pelajari fitur IAM apa yang tersedia untuk digunakan dengan Amazon Inspector.

Fitur IAM yang dapat Anda gunakan dengan Amazon Inspector

Fitur IAM	Dukungan Amazon Inspector
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya

Fitur IAM	Dukungan Amazon Inspector
Sumber daya kebijakan	Ya
kunci-kunci persyaratan kebijakan (spesifik layanan)	Ya
ACLs	Tidak
ABAC (tanda dalam kebijakan)	Parsial
Kredensial sementara	Ya
Izin principal	Ya
Peran layanan	Tidak
Peran terkait layanan	Ya

Untuk mendapatkan tampilan tingkat tinggi tentang cara kerja Amazon Inspector dan Layanan AWS lainnya dengan sebagian besar fitur IAM, Layanan AWS lihat fitur tersebut bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk Amazon Inspector

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Contoh kebijakan berbasis identitas untuk Amazon Inspector

Untuk melihat contoh kebijakan berbasis identitas Amazon Inspector, lihat Contoh kebijakan berbasis identitas untuk Amazon Inspector.

Kebijakan berbasis sumber daya dalam Amazon Inspector

Mendukung kebijakan berbasis sumber daya: Tidak

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh principal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan principal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan secara spesifik seluruh akun atau entitas IAM di akun lain sebagai principal dalam kebijakan berbasis sumber daya. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk Amazon Inspector

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan Amazon Inspector, lihat Tindakan yang ditentukan oleh Amazon Inspector di Referensi Otorisasi Layanan.

Tindakan kebijakan di Amazon Inspector menggunakan awalan berikut sebelum tindakan:

```
inspector2
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [
      "inspector2:action1",
      "inspector2:action2"
         ]
```

Untuk melihat contoh kebijakan berbasis identitas Amazon Inspector, lihat Contoh kebijakan berbasis identitas untuk Amazon Inspector.

Sumber daya kebijakan untuk Amazon Inspector

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

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

Untuk melihat daftar jenis sumber daya Amazon Inspector dan jenisnya ARNs, lihat Sumber daya yang ditentukan oleh Amazon Inspector di Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat Tindakan yang ditentukan oleh Amazon Inspector.

Untuk melihat contoh kebijakan berbasis identitas Amazon Inspector, lihat Contoh kebijakan berbasis identitas untuk Amazon Inspector.

Kunci kondisi kebijakan untuk Amazon Inspector

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen `Condition` menentukan ketika pernyataan dieksekusi berdasarkan kriteria yang ditetapkan. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci kondisi Amazon Inspector, lihat Kunci kondisi untuk Amazon Inspector di Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang ditentukan oleh Amazon Inspector.

Untuk melihat contoh kebijakan berbasis identitas Amazon Inspector, lihat Contoh kebijakan berbasis identitas untuk Amazon Inspector.

ACLs di Amazon Inspector

Mendukung ACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan JSON.

ABAC dengan Amazon Inspector

Mendukung ABAC (tag dalam kebijakan): Sebagian

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut tanda. Anda dapat melampirkan tag ke entitas dan AWS sumber daya IAM, lalu merancang kebijakan ABAC untuk mengizinkan operasi saat tag prinsipal cocok dengan tag pada sumber daya.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Menggunakan kredensial sementara dengan Amazon Inspector

Mendukung kredensial sementara: Ya

Kredensi sementara menyediakan akses jangka pendek ke AWS sumber daya dan secara otomatis dibuat saat Anda menggunakan federasi atau beralih peran. AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensi sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM dan Layanan AWS yang berfungsi dengan IAM dalam Panduan Pengguna IAM.

Izin utama lintas layanan untuk Amazon Inspector

Mendukung sesi akses terusan (FAS): Ya

Sesi akses teruskan (FAS) menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses terusan.

Peran layanan untuk Amazon Inspector

Mendukung peran layanan: Tidak

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak fungsionalitas Amazon Inspector. Edit peran layanan hanya jika Amazon Inspector memberikan panduan untuk melakukannya.

Peran terkait layanan untuk Amazon Inspector

Mendukung peran terkait layanan: Ya

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul

di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk detail tentang membuat atau mengelola peran terkait layanan, lihat Layanan AWS bahwa bekerja dengan IAM. Cari layanan dalam tabel yang memiliki `Yes` di kolom Peran terkait layanan. Pilih tautan Ya untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Contoh kebijakan berbasis identitas untuk Amazon Inspector

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya Amazon Inspector. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh Amazon Inspector, termasuk format ARNs untuk setiap jenis sumber daya, lihat Kunci tindakan, sumber daya, dan kondisi untuk Amazon Inspector dalam Referensi Otorisasi Layanan.

Topik

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya Amazon Inspector di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan

yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.

- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, lihat Praktik terbaik keamanan di IAM dalam Panduan Pengguna IAM.

Menggunakan konsol Amazon Inspector

Untuk mengakses konsol Amazon Inspector tersebut, Anda harus memiliki rangkaian izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang sumber daya

Amazon Inspector di Anda. Akun AWS Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna atau peran) dengan kebijakan tersebut.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau AWS API. Sebagai gantinya, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang coba mereka lakukan.

Untuk memastikan bahwa pengguna dan peran masih dapat menggunakan konsol Amazon Inspector, lampirkan juga Amazon *ConsoleAccess* Inspector *ReadOnly* AWS atau kebijakan terkelola ke entitas. Untuk informasi selengkapnya, lihat Menambah izin untuk pengguna dalam Panduan Pengguna IAM.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara membuat kebijakan yang mengizinkan pengguna IAM melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau menggunakan API atau secara terprogram. 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": "*"
        }
    ]
}
```

Izinkan akses hanya-baca ke semua sumber daya Amazon Inspector

Contoh ini menunjukkan kebijakan yang memungkinkan akses hanya-baca ke semua sumber daya 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": "*"
        }
```

```
    ]
}
```

Izinkan akses penuh ke semua sumber daya Amazon Inspector

Contoh ini menunjukkan kebijakan yang memungkinkan akses penuh ke semua sumber daya 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 kebijakan terkelola untuk Amazon Inspector

Kebijakan AWS terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS Kebijakan terkelola dirancang untuk memberikan izin bagi banyak kasus penggunaan umum sehingga Anda dapat mulai menetapkan izin kepada pengguna, grup, dan peran.

Perlu diingat bahwa kebijakan AWS terkelola mungkin tidak memberikan izin hak istimewa paling sedikit untuk kasus penggunaan spesifik Anda karena tersedia untuk digunakan semua pelanggan. AWS Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda.

Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan AWS terkelola. Jika AWS memperbarui izin yang ditentukan dalam kebijakan AWS terkelola, pembaruan akan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui kebijakan AWS terkelola saat baru Layanan AWS diluncurkan atau operasi API baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat Kebijakan terkelola AWS dalam Panduan Pengguna IAM.

AWS kebijakan terkelola: AmazonInspector2FullAccess_v2

Anda dapat melampirkan kebijakan `AmazonInspector2FullAccess_v2` ke identitas IAM Anda.

Kebijakan ini memberikan akses penuh ke Amazon Inspector dan akses ke layanan terkait lainnya.

Detail izin

Kebijakan ini mencakup izin berikut.

- `inspector2`— Memungkinkan akses lengkap ke Amazon Inspector APIs.
- `codeguru-security`— Memungkinkan administrator untuk mengambil temuan keamanan dan pengaturan konfigurasi untuk akun.
- `iam`— Memungkinkan Amazon Inspector untuk membuat peran terkait layanan dan. `AWSServiceRoleForAmazonInspector2` `AWSServiceRoleForAmazonInspector2Agentless` `AWSServiceRoleForAmazonInspector2`Amazon Inspector diperlukan untuk melakukan operasi seperti mengambil informasi tentang instans Amazon EC2, repositori Amazon ECR, dan gambar kontainer Amazon ECR. Ini juga diperlukan untuk mendekripsi snapshot Amazon EBS yang dienkripsi dengan kunci. AWS KMS Untuk informasi selengkapnya, lihat Menggunakan peran tertaut layanan untuk Amazon Inspector.
- `organizations`— hanya `AllowServicePrincipalBasedAccessToOrganizationApis` mengizinkan prinsipal layanan untuk membuat peran terkait layanan Akun AWS, mendaftarkan Akun AWS sebagai administrator yang didelegasikan untuk organisasi, dan daftar administrator yang didelegasikan dalam suatu organisasi. `AllowOrganizationalBasedAccessToOrganizationApis`memungkinkan pemegang polis untuk mengambil informasi, khususnya tingkat sumber daya ARNs, tentang unit organisasi. `AllowAccountsBasedAccessToOrganizationApis`memungkinkan pemegang polis untuk mengambil informasi, khususnya tingkat sumber daya ARNs, tentang suatu. Akun `AWSAllowAccessToOrganizationApis`memungkinkan pemegang polis untuk melihat informasi yang Layanan AWS terintegrasi dengan organisasi dan organisasi. Kebijakan ini memungkinkan daftar kebijakan organisasi Inspector dengan memfilter menurut jenis kebijakan Inspector, melihat kebijakan sumber daya delegasi yang ditetapkan oleh akun manajemen, dan melihat kebijakan Inspector efektif yang diterapkan pada akun.

> **Note**
>
> Amazon Inspector tidak lagi menggunakan CodeGuru untuk melakukan pemindaian Lambda. AWS akan menghentikan dukungan untuk CodeGuru pada 20 November 2025. Untuk informasi selengkapnya, lihat Akhir dukungan untuk CodeGuru Keamanan. Amazon Inspector sekarang menggunakan Amazon Q untuk melakukan pemindaian Lambda dan tidak memerlukan izin yang dijelaskan di bagian ini.

Untuk meninjau izin kebijakan ini, lihat AmazonInspector2 FullAccess _v2 di Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSInspector2OrganizationsAccess

Anda dapat melampirkan kebijakan `AWSInspector2OrganizationsAccess` ke identitas IAM Anda.

Kebijakan ini memberikan izin administratif untuk mengaktifkan dan mengelola Amazon Inspector untuk organisasi di. AWS Organizations Izin untuk kebijakan ini memungkinkan akun manajemen organisasi menunjuk akun administrator yang didelegasikan untuk Amazon Inspector. Mereka juga mengizinkan akun administrator yang didelegasikan untuk mengaktifkan akun organisasi sebagai akun anggota.

Kebijakan ini hanya memberikan izin untuk AWS Organizations. Akun manajemen organisasi dan akun administrator yang didelegasikan juga memerlukan izin untuk tindakan terkait. Izin ini dapat diberikan menggunakan kebijakan `AmazonInspector2FullAccess_v2` terkelola.

Detail izin

Kebijakan ini mencakup izin berikut.

- `organizations:ListAccounts`— Memungkinkan kepala sekolah untuk mengambil daftar akun yang merupakan bagian dari organisasi.
- `organizations:DescribeOrganization`— Memungkinkan kepala sekolah untuk mengambil informasi tentang organisasi.
- `organizations:ListRoots`— Memungkinkan kepala sekolah untuk membuat daftar akar organisasi.
- `organizations:ListDelegatedAdministrators`— Memungkinkan kepala sekolah untuk membuat daftar administrator yang didelegasikan dari suatu organisasi.
- `organizations:ListAWSServiceAccessForOrganization`— Memungkinkan kepala sekolah untuk membuat daftar Layanan AWS yang digunakan organisasi.
- `organizations:ListOrganizationalUnitsForParent`— Memungkinkan kepala sekolah untuk membuat daftar unit organisasi anak (OU) dari OU orang tua.
- `organizations:ListAccountsForParent`— Memungkinkan kepala sekolah untuk membuat daftar akun anak dari OU orang tua.
- `organizations:ListParents`— Daftar root atau unit organisasi (OUs) yang berfungsi sebagai induk langsung dari OU atau akun anak yang ditentukan.
- `organizations:DescribeAccount` – Memungkinkan principal mengambil informasi tentang akun di organisasi.

- `organizations:DescribeOrganizationalUnit`— Memungkinkan kepala sekolah untuk mengambil informasi tentang OU dalam organisasi.
- `organizations:ListPolicies`— Mengambil daftar semua kebijakan dalam organisasi dari jenis tertentu.
- `organizations:ListPoliciesForTarget`— Daftar kebijakan yang secara langsung dilampirkan ke root target yang ditentukan, unit organisasi (OU), atau akun.
- `organizations:ListTargetsForPolicy`— Daftar semua akar, unit organisasi (OUs), dan akun yang dilampirkan kebijakan yang ditentukan.
- `organizations:DescribeResourcePolicy`— Mengambil informasi tentang kebijakan sumber daya.
- `organizations:EnableAWSServiceAccess`— Memungkinkan prinsipal untuk mengaktifkan integrasi dengan Organizations.
- `organizations:RegisterDelegatedAdministrator`— Memungkinkan kepala sekolah untuk menunjuk akun administrator yang didelegasikan.
- `organizations:DeregisterDelegatedAdministrator`— Memungkinkan kepala sekolah untuk menghapus akun administrator yang didelegasikan.
- `organizations:DescribePolicy`— Mengambil informasi tentang kebijakan.
- `organizations:DescribeEffectivePolicy`— Mengembalikan isi kebijakan efektif untuk jenis kebijakan dan akun tertentu.
- `organizations:CreatePolicy`— Membuat kebijakan dari jenis tertentu yang dapat Anda lampirkan ke root, unit organisasi (OU), atau individu Akun AWS.
- `organizations:UpdatePolicy`— Memperbarui kebijakan yang ada dengan nama, deskripsi, atau konten baru.
- `organizations:DeletePolicy`— Menghapus kebijakan yang ditentukan dari organisasi Anda.
- `organizations:AttachPolicy`— Melampirkan kebijakan ke root, unit organisasi (OU), atau akun individu.
- `organizations:DetachPolicy`Melepaskan kebijakan dari akar target, unit organisasi (OU), atau akun.
- `organizations:EnablePolicyType`— Mengaktifkan jenis kebijakan di root.
- `organizations:DisablePolicyType`— Menonaktifkan jenis kebijakan organisasi di root.
- `organizations:TagResource`— Menambahkan satu atau lebih tag ke sumber daya tertentu.
- `organizations:UntagResource`— Menghapus tag apa pun dengan kunci yang ditentukan dari sumber daya tertentu.

- `organizations:ListTagsForResource`— Daftar tag yang dilampirkan ke sumber daya tertentu.

Untuk meninjau izin kebijakan ini, lihat AWSInspector2OrganizationsAccessdi Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AmazonInspector2FullAccess

Anda dapat melampirkan kebijakan `AmazonInspector2FullAccess` ke identitas IAM Anda.

Kebijakan ini memberikan izin administratif yang memungkinkan akses penuh ke Amazon Inspector.

> **Important**
>
> Untuk keamanan yang ditingkatkan dan izin terbatas pada prinsipal layanan Inspector 2, kami sarankan Anda menggunakan 2 _v2. AmazonInspector FullAccess

Detail izin

Kebijakan ini mencakup izin berikut.

- `inspector2`— Memungkinkan akses penuh ke fungsionalitas Amazon Inspector.
- `iam`— Memungkinkan Amazon Inspector untuk membuat peran terkait layanan dan. `AWSServiceRoleForAmazonInspector2` `AWSServiceRoleForAmazonInspector2Agentless` `AWSServiceRoleForAmazonInspector2`Amazon Inspector diperlukan untuk melakukan operasi seperti mengambil informasi tentang instans Amazon EC2, repositori Amazon ECR, dan gambar kontainer. Amazon Inspector juga diperlukan untuk menganalisis jaringan VPC Anda dan menjelaskan akun yang terkait dengan organisasi Anda. `AWSServiceRoleForAmazonInspector2Agentless`Amazon Inspector diperlukan untuk melakukan operasi, seperti mengambil informasi tentang instans Amazon EC2 Anda dan snapshot Amazon EBS. Ini juga diperlukan untuk mendekripsi snapshot Amazon EBS yang dienkripsi dengan kunci. AWS KMS Untuk informasi selengkapnya, lihat Menggunakan peran tertaut layanan untuk Amazon Inspector.

- `organizations`— Memungkinkan administrator menggunakan Amazon Inspector untuk organisasi di. AWS Organizations Saat Anda mengaktifkan akses tepercaya untuk Amazon Inspector AWS Organizations, anggota akun administrator yang didelegasikan dapat mengelola setelan dan melihat temuan di seluruh organisasi mereka.
- `codeguru-security`— Memungkinkan administrator menggunakan Amazon Inspector untuk mengambil cuplikan kode informasi dan mengubah pengaturan enkripsi untuk kode yang disimpan Security. CodeGuru Untuk informasi selengkapnya, lihat Enkripsi saat istirahat untuk kode dalam temuan Anda.

Untuk meninjau izin kebijakan ini, lihat AmazonInspector2 FullAccess di Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AmazonInspector2ReadOnlyAccess

Anda dapat melampirkan kebijakan `AmazonInspector2ReadOnlyAccess` ke identitas IAM Anda.

Kebijakan ini memberikan izin yang memungkinkan akses hanya-baca ke Amazon Inspector.

Detail izin

Kebijakan ini mencakup izin berikut.

- `inspector2`— Memungkinkan akses read-only ke fungsionalitas Amazon Inspector.
- `organizations`— Memungkinkan detail tentang cakupan Amazon Inspector untuk organisasi yang akan AWS Organizations dilihat. Selain itu memungkinkan melihat kebijakan organisasi Inspector melalui `ListPolicies` penyaringan menurut jenis kebijakan Inspector, melihat kebijakan sumber daya delegasi melalui, `DescribeResourcePolicy` dan melihat kebijakan Inspector efektif yang diterapkan ke akun melalui. `DescribeEffectivePolicy` Hal ini memungkinkan pengguna untuk memahami pemberdayaan inspektur terpusat yang ditetapkan melalui kebijakan organisasi tanpa kemampuan untuk memodifikasinya.
- `codeguru-security`— Memungkinkan cuplikan kode diambil dari Keamanan. CodeGuru Juga memungkinkan pengaturan enkripsi untuk kode Anda yang disimpan di CodeGuru Keamanan untuk dilihat.

Untuk meninjau izin kebijakan ini, lihat AmazonInspector2 ReadOnlyAccess di Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AmazonInspector2ManagedCisPolicy

Anda dapat melampirkan `AmazonInspector2ManagedCisPolicy` kebijakan ke entitas IAM Anda. Kebijakan ini harus dilampirkan ke peran yang memberikan izin ke instans Amazon EC2 Anda untuk menjalankan pemindaian CIS instance. Anda dapat menggunakan peran IAM untuk mengelola kredensyal sementara untuk aplikasi yang berjalan pada instans EC2 dan membuat AWS CLI atau permintaan API. AWS Cara ini lebih dianjurkan daripada menyimpan kunci akses dalam instans EC2. Untuk menetapkan AWS peran ke instans EC2 dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat Gunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2 dalam Panduan Pengguna IAM.

Detail izin

Kebijakan ini mencakup izin berikut.

- `inspector2`— Memungkinkan akses ke tindakan yang digunakan untuk menjalankan pemindaian CIS.

Untuk meninjau izin kebijakan ini, lihat AmazonInspector2 ManagedCisPolicy di Panduan Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AmazonInspector2ServiceRolePolicy

Anda tidak dapat melampirkan kebijakan `AmazonInspector2ServiceRolePolicy` ke entitas IAM Anda. Kebijakan ini dilampirkan pada peran terkait layanan yang memungkinkan Amazon Inspector melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan peran tertaut layanan untuk Amazon Inspector.

AWS kebijakan terkelola: AmazonInspector2AgentlessServiceRolePolicy

Anda tidak dapat melampirkan kebijakan `AmazonInspector2AgentlessServiceRolePolicy` ke entitas IAM Anda. Kebijakan ini dilampirkan pada peran terkait layanan yang memungkinkan Amazon Inspector melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan peran tertaut layanan untuk Amazon Inspector.

AWS kebijakan terkelola: AmazonInspector2ManagedTelemetryPolicy

Anda dapat melampirkan `AmazonInspector2ManagedTelemetryPolicy` kebijakan ke entitas IAM Anda. Kebijakan ini memberikan izin untuk operasi telemetri Amazon Inspector, yang memungkinkan layanan mengumpulkan dan mengirimkan data inventaris paket untuk pemindaian kerentanan.

Detail izin

Kebijakan ini mencakup izin berikut.

- `inspector2-telemetry`— Memungkinkan akses ke tindakan untuk transmisi data invetory paket.

Untuk melihat detail selengkapnya tentang kebijakan, termasuk versi terbaru dari dokumen kebijakan JSON, lihat AmazonInspector2 ManagedTelemetryPolicy di Panduan Referensi Kebijakan AWS Terkelola.

Amazon Inspector memperbarui kebijakan terkelola AWS

Lihat detail tentang pembaruan kebijakan AWS terkelola untuk Amazon Inspector sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS di halaman riwayat Dokumen Amazon Inspector.

Ubah	Deskripsi	Date
AWSInspector2Organizations Access – Kebijakan baru	Amazon Inspector telah menambahkan kebijakan terkelola baru yang memberikan izin yang diperlukan untuk mengaktifkan dan mengelola Amazon Inspector melalui kebijakan. AWS Organizations	Maret 3, 2026

Ubah	Deskripsi	Date
AmazonInspector2 ManagedTelemetryPolicy — Kebijakan baru	Amazon Inspector telah menambahkan kebijakan terkelola baru yang memberikan izin untuk operasi telemetri Amazon Inspector, yang memungkinkan layanan mengumpulkan dan mengirimkan data inventaris paket untuk pemindaian kerentanan.	Februari 5, 2026
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector untuk menjelaskan metadata firewall untuk analisis jangkauan jaringan. Selain itu, Amazon Inspector telah menambahkan pelingkupan sumber daya tambahan untuk memungkinkan Amazon Inspector membuat, memperbarui, dan memulai asosiasi SSM dengan dokumen SSM. `AWS-ConfigureAWSPackage`	Februari 3, 2026

Ubah	Deskripsi	Date
AmazonInspector2 FullAccess _v2 dan AmazonInspector2 ReadOnlyAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pemegang kebijakan untuk melihat kebijakan organisasi Inspector dan konfigurasi delegasi. Ini mendukung manajemen terpusat dan visibilitas pemberdayaan Inspector melalui kebijakan. AWS Organizations	November 14, 2025
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan AWS Organizations kebijakan Amazon Inspector untuk menegakkan pemberdayaan dan penonaktifan Amazon Inspector.	November 10, 2025
AmazonInspector2 FullAccess _v2 - Kebijakan baru	Amazon Inspector telah menambahkan kebijakan terkelola baru yang menyediakan akses penuh ke Amazon Inspector dan akses ke layanan terkait lainnya.	Juli 03, 2025
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector untuk menjelaskan alamat IP dan gateway internet.	April 29, 2025

Ubah	Deskripsi	Date
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan akses hanya-baca ke Amazon ECS dan tindakan Amazon EKS.	Maret 25, 2025
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector mengembalikan tag fungsi. AWS Lambda	Juli 31, 2024
AmazonInspector2 FullAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin yang memungkinkan Amazon Inspector membuat peran terkait layanan. `AWSServiceRoleForAmazonInspector2Agentless` Ini memungkinkan pengguna untuk melakukan pemindaian berbasis agen dan pemindaian tanpa agen saat mereka mengaktifkan Amazon Inspector.	April 24, 2024
AmazonInspector2 ManagedCisPolicy — Kebijakan baru	Amazon Inspector telah menambahkan kebijakan terkelola baru yang dapat Anda gunakan sebagai bagian dari profil instans untuk mengizinkan pemindaian CIS pada instans.	23 Januari 2024

Ubah	Deskripsi	Date
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector untuk memulai pemindaian CIS pada instance target.	23 Januari 2024
AmazonInspector2 AgentlessServiceRolePolicy — Kebijakan baru	Amazon Inspector telah menambahkan kebijakan peran terkait layanan baru untuk memungkinkan pemindaian instans EC2 tanpa agen.	27 November 2023
AmazonInspector2 ReadOnlyAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna hanya-baca untuk mengambil detail intelijen kerentanan untuk temuan kerentanan paket.	September 22, 2023
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector memindai konfigurasi jaringan instans Amazon EC2 yang merupakan bagian dari grup target Elastic Load Balancing.	31 Agustus 2023

Ubah	Deskripsi	Date
AmazonInspector2 ReadOnlyAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna read-only untuk mengekspo r Software Bill of Materials (SBOM) untuk sumber daya mereka.	29 Juni 2023
AmazonInspector2 ReadOnlyAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna hanya-baca untuk mengambil detail pengaturan enkripsi untuk temuan pemindaian kode Lambda untuk akun mereka.	13 Juni 2023
AmazonInspector2 FullAcces s - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna mengonfigurasi kunci KMS yang dikelola pelanggan untuk mengenkripsi kode dalam temuan dari pemindaian kode Lambda.	13 Juni 2023
AmazonInspector2 ReadOnlyAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna hanya-baca untuk mengambil detail status pemindaian kode Lambda dan temuan untuk akun mereka.	02 Mei 2023

Ubah	Deskripsi	Date
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector membuat saluran AWS CloudTrail terkait layanan di akun Anda saat Anda mengaktifkan pemindaian Lambda. Ini memungkinkan Amazon Inspector untuk memantau CloudTrail peristiwa di akun Anda.	April 30, 2023
AmazonInspector2 FullAccess - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna untuk mengambil rincian temuan kerentanan kode dari pemindaian kode Lambda.	April 21, 2023
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector mengirim informasi ke Amazon EC2 Systems Manager tentang jalur khusus yang telah ditentukan pelanggan untuk inspeksi mendalam Amazon EC2.	17 April 2023

Ubah	Deskripsi	Date
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector membuat saluran AWS CloudTrail terkait layanan di akun Anda saat Anda mengaktifkan pemindaian Lambda. Ini memungkinkan Amazon Inspector untuk memantau CloudTrail peristiwa di akun Anda.	April 30, 2023
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector untuk meminta pemindaian kode pengembang dalam AWS Lambda fungsi, dan menerima data pemindaian dari Amazon Security. CodeGuru Selain itu, Amazon Inspector telah menambahkan izin untuk meninjau kebijakan IAM. Amazon Inspector menggunakan informasi ini untuk memindai fungsi Lambda untuk kerentanan kode.	28 Februari 2023

Ubah	Deskripsi	Date
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan pernyataan baru yang memungkinkan Amazon Inspector untuk mengambil informasi CloudWatch dari tentang kapan AWS Lambda fungsi terakhir dipanggil. Amazon Inspector menggunakan informasi ini untuk memfokuskan pemindaian pada fungsi Lambda di lingkungan Anda yang telah aktif dalam 90 hari terakhir.	Februari 20, 2023
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan pernyataan baru yang memungkinkan Amazon Inspector untuk mengambil informasi AWS Lambda tentang fungsi, termasuk setiap versi lapisan yang terkait dengan setiap fungsi. Amazon Inspector menggunakan informasi ini untuk memindai fungsi Lambda untuk kerentanan keamanan.	28 November 2022

Ubah	Deskripsi	Date
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah menambahkan tindakan baru untuk memungkinkan Amazon Inspector menggambarkan eksekusi asosiasi SSM. Selain itu, Amazon Inspector telah menambahkan pelingkupan sumber daya tambahan untuk memungkinkan Amazon Inspector membuat, memperbarui, menghapus, dan memulai asosiasi SSM dengan dokumen SSM yang dimiliki. `AmazonInspector2`	31 Agustus 2022
AmazonInspector2 ServiceRolePolicy Pembaruan kebijakan yang ada	Amazon Inspector telah memperbarui pelingkupan sumber daya kebijakan untuk memungkinkan Amazon Inspector mengumpulkan inventaris perangkat lunak di partisi lain. AWS	12 Agustus 2022
AmazonInspector2 ServiceRolePolicy - Pembaruan kebijakan yang ada	Amazon Inspector telah merestrukturisasi pelingkupan sumber daya dari tindakan yang memungkinkan Amazon Inspector membuat, menghapus, dan memperbarui asosiasi SSM.	Agustus 10, 2022

Ubah	Deskripsi	Date
AmazonInspector2 ReadOnlyAccess — Kebijakan baru	Amazon Inspector menambahkan kebijakan baru untuk mengizinkan akses hanya-baca ke fungsionalitas Amazon Inspector.	Januari 21, 2022
AmazonInspector2 FullAccess — Kebijakan baru	Amazon Inspector menambahkan kebijakan baru untuk memungkinkan akses penuh ke fungsionalitas Amazon Inspector.	29 November 2021
AmazonInspector2 ServiceRolePolicy — Kebijakan baru	Amazon Inspector menambahkan kebijakan baru untuk mengizinkan Amazon Inspector melakukan tindakan di layanan lain atas nama Anda.	29 November 2021
Amazon Inspector mulai melacak perubahan	Amazon Inspector mulai melacak perubahan untuk kebijakan yang AWS dikelola.	29 November 2021

Menggunakan peran tertaut layanan untuk Amazon Inspector

Amazon Inspector menggunakan peran terkait layanan AWS Identity and Access Management (IAM) bernama. `AWSServiceRoleForAmazonInspector2` Peran terkait layanan ini adalah peran IAM yang ditautkan langsung ke Amazon Inspector. Ini telah ditentukan sebelumnya oleh Amazon Inspector dan mencakup semua izin yang diperlukan oleh Amazon Inspector untuk memanggil orang lain atas nama Anda. Layanan AWS

Peran tertaut layanan mempermudah pengaturan Amazon Inspector karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. Amazon Inspector mendefinisikan izin peran terkait layanan dan, kecuali ditentukan lain, hanya Amazon Inspector yang dapat mengambil peran

tersebut. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, serta bahwa kebijakan izin tidak dapat dilampirkan ke entitas IAM lainnya.

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (seperti grup atau peran) membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat Izin peran terkait layanan dalam Panduan Pengguna IAM. Anda dapat menghapus peran terkait layanan hanya setelah menghapus sumber daya terkait. Ini melindungi sumber daya Amazon Inspector karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, silakan lihat layanan AWS yang bisa digunakan dengan IAM dan carilah layanan yang memiliki opsi Ya di kolom Peran terkait layanan. Pilih Ya dengan tautan untuk meninjau dokumentasi peran terkait layanan untuk layanan tersebut.

Izin peran tertaut layanan untuk Amazon Inspector

Amazon Inspector menggunakan kebijakan terkelola bernama. `AWSServiceRoleForAmazonInspector2` Peran terkait layanan ini mempercayai `inspector2.amazonaws.com` layanan untuk mengambil peran tersebut.

Kebijakan izin untuk peran, yang diberi nama `AmazonInspector2ServiceRolePolicy`, memungkinkan Amazon Inspector untuk melakukan tugas-tugas seperti:

- Gunakan tindakan Amazon Elastic Compute Cloud (Amazon EC2) untuk mengambil informasi tentang instans dan jalur jaringan Anda.
- Gunakan AWS Systems Manager tindakan untuk mengambil inventaris dari instans Amazon EC2 Anda, dan untuk mengambil informasi tentang paket pihak ketiga dari jalur khusus.
- Gunakan AWS Systems Manager `SendCommand` tindakan untuk memanggil pemindaian CIS untuk instance target.
- Gunakan tindakan Amazon Elastic Container Registry untuk mengambil informasi tentang gambar kontainer Anda.
- Gunakan AWS Lambda tindakan untuk mengambil informasi tentang fungsi Lambda Anda.
- Gunakan AWS Organizations tindakan untuk mendeskripsikan akun terkait.
- Gunakan CloudWatch tindakan untuk mengambil informasi tentang terakhir kali fungsi Lambda Anda dipanggil.
- Gunakan tindakan IAM tertentu untuk mengambil informasi tentang kebijakan IAM Anda yang dapat membuat kerentanan keamanan dalam kode Lambda Anda.

- Gunakan tindakan Amazon Q untuk melakukan pemindaian kode di fungsi Lambda Anda. Amazon Inspector menggunakan tindakan Amazon Q berikut:
  - codeguru-security: CreateScan — Memberikan izin untuk membuat Amazon Q; memindai.
  - codeguru-security: GetScan — Memberikan izin untuk mengambil metadata pemindaian Amazon Q.
  - codeguru-security: ListFindings — Memberikan izin untuk mengambil temuan yang dihasilkan oleh Amazon Q.
  - codeguru-security: DeleteScansByCategory - Memberikan izin kepada Amazon Q untuk menghapus pemindaian yang diprakarsai oleh Amazon Inspector.
  - codeguru-security: BatchGetFindings — Memberikan izin untuk mengambil sekumpulan temuan spesifik yang dihasilkan oleh Amazon Q.
- Gunakan tindakan Elastic Load Balancing tertentu untuk membentuk pemindaian jaringan instans EC2 yang merupakan bagian dari kelompok target Elastic Load Balancing.
- Gunakan tindakan Amazon ECS dan Amazon EKS untuk mengizinkan akses hanya-baca untuk melihat kluster dan tugas serta menjelaskan tugas.
- Gunakan AWS Organizations tindakan untuk mencantumkan administrator yang didelegasikan untuk Amazon Inspector di seluruh organisasi.
- Gunakan tindakan Amazon Inspector untuk mengaktifkan dan menonaktifkan Amazon Inspector di seluruh organisasi.
- Gunakan tindakan Amazon Inspector untuk menunjuk akun administrator yang didelegasikan dan akun anggota asosiasi di seluruh organisasi.

> **Note**
>
> Amazon Inspector tidak lagi menggunakan CodeGuru untuk melakukan pemindaian Lambda. AWS akan menghentikan dukungan untuk CodeGuru pada 20 November 2025. Untuk informasi selengkapnya, lihat Akhir dukungan untuk CodeGuru Keamanan. Amazon Inspector sekarang menggunakan Amazon Q untuk melakukan pemindaian Lambda dan tidak memerlukan izin yang dijelaskan di bagian ini.

Untuk meninjau izin kebijakan ini, lihat AmazonInspector2 ServiceRolePolicy di Panduan Referensi Kebijakan AWS Terkelola.

Membuat peran tertaut layanan untuk Amazon Inspector

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda mengaktifkan Amazon Inspector di, API Konsol Manajemen AWS, atau AWS API AWS CLI, Amazon Inspector membuat peran terkait layanan untuk Anda.

Mengedit peran tertaut layanan untuk Amazon Inspector

Amazon Inspector tidak mengizinkan Anda untuk mengedit peran tertaut layanan `AWSServiceRoleForAmazonInspector2`. Setelah peran terkait layanan dibuat, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengubah deskripsi peran dengan menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran terkait layanan dalam Panduan Pengguna IAM.

Menghapus peran tertaut layanan untuk Amazon Inspector

Jika Anda tidak perlu lagi menggunakan Amazon Inspector, sebaiknya hapus peran terkait `AWSServiceRoleForAmazonInspector2` layanan. Sebelum Anda dapat menghapus peran, Anda harus menonaktifkan Amazon Inspector di AWS Region setiap tempat itu diaktifkan. Saat Anda menonaktifkan Amazon Inspector, itu tidak menghapus peran untuk Anda. Oleh karena itu, jika Anda mengaktifkan Amazon Inspector lagi, itu dapat menggunakan peran yang ada. Dengan begitu Anda dapat menghindari entitas yang tidak terpakai yang tidak dipantau atau dipelihara secara aktif. Tetapi, Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.

Jika Anda menghapus peran tertaut layanan ini dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda mengaktifkan Amazon Inspector, Amazon Inspector membuat ulang peran terkait layanan untuk Anda.

> **Note**
>
> Jika layanan Amazon Inspector menggunakan peran saat Anda mencoba menghapus sumber daya, penghapusan mungkin gagal. Jika itu terjadi, tunggu beberapa menit dan kemudian coba operasi lagi.

Anda dapat menggunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran `AWSServiceRoleForAmazonInspector2` terkait layanan. Untuk informasi selengkapnya, lihat Menghapus peran terkait layanan dalam Panduan Pengguna IAM.

Izin peran terkait layanan untuk pemindaian tanpa agen Amazon Inspector

Pemindaian tanpa agen Amazon Inspector menggunakan peran terkait layanan bernama. `AWSServiceRoleForAmazonInspector2Agentless` SLR ini memungkinkan Amazon Inspector untuk membuat snapshot volume Amazon EBS di akun Anda, lalu mengakses data dari snapshot tersebut. Peran terkait layanan ini mempercayai `agentless.inspector2.amazonaws.com` layanan untuk mengambil peran tersebut.

> **Important**
>
> Pernyataan dalam peran terkait layanan ini mencegah Amazon Inspector melakukan pemindaian tanpa agen pada instans EC2 apa pun yang telah Anda kecualikan dari pemindaian menggunakan tag. `InspectorEc2Exclusion` Selain itu, pernyataan mencegah Amazon Inspector mengakses data terenkripsi dari volume ketika kunci KMS yang digunakan untuk mengenkripsi memiliki tag. `InspectorEc2Exclusion` Untuk informasi selengkapnya, lihat Mengecualikan instance dari pemindaian Amazon Inspector.

Kebijakan izin untuk peran, yang diberi nama`AmazonInspector2AgentlessServiceRolePolicy`, memungkinkan Amazon Inspector untuk melakukan tugas-tugas seperti:

- Gunakan tindakan Amazon Elastic Compute Cloud (Amazon EC2) untuk mengambil informasi tentang instans, volume, dan snapshot EC2 Anda.
  - Gunakan tindakan penandaan Amazon EC2 untuk menandai snapshot untuk pemindaian dengan kunci tag. `InspectorScan`
  - Gunakan tindakan snapshot Amazon EC2 untuk membuat snapshot, beri tag dengan kunci `InspectorScan` tag, lalu hapus snapshot volume Amazon EBS yang telah ditandai dengan kunci tag. `InspectorScan`
- Gunakan tindakan Amazon EBS untuk mengambil informasi dari snapshot yang ditandai dengan kunci tag. `InspectorScan`
- Gunakan tindakan AWS KMS dekripsi pilih untuk mendekripsi snapshot yang dienkripsi dengan kunci yang dikelola pelanggan. AWS KMS Amazon Inspector tidak mendekripsi snapshot ketika kunci KMS yang digunakan untuk mengenkripsi mereka ditandai dengan tag. `InspectorEc2Exclusion`

Peran dikonfigurasi dengan kebijakan izin berikut.

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/*"
   }
  ]
}
```

Membuat peran terkait layanan untuk pemindaian tanpa agen

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda mengaktifkan Amazon Inspector di, API Konsol Manajemen AWS, atau AWS API AWS CLI, Amazon Inspector membuat peran terkait layanan untuk Anda.

Mengedit peran terkait layanan untuk pemindaian tanpa agen

Amazon Inspector tidak mengizinkan Anda untuk mengedit peran tertaut layanan `AWSServiceRoleForAmazonInspector2Agentless`. Setelah peran terkait layanan dibuat, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengubah deskripsi peran dengan menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran terkait layanan dalam Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk pemindaian tanpa agen

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, sebaiknya hapus peran tersebut. Dengan begitu, Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dikelola secara aktif.

> **Important**
>
> Untuk menghapus `AWSServiceRoleForAmazonInspector2Agentless` peran, Anda harus mengatur mode pemindaian Anda ke berbasis agen di semua Wilayah di mana pemindaian tanpa agen tersedia.

Untuk menghapus peran tertaut layanan secara manual menggunakan IAM

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran terkait layanan AWSService RoleForAmazonInspector 2Agentless. Untuk informasi selengkapnya, lihat Menghapus peran terkait layanan dalam Panduan Pengguna IAM.

Pemecahan masalah identitas dan akses Amazon Inspector

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan mengatasi masalah umum yang mungkin Anda temui saat bekerja menggunakan Amazon Inspector dan IAM.

Topik

- Saya tidak berwenang untuk melakukan tindakan di Amazon Inspector
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses sumber daya Amazon Inspector saya

Saya tidak berwenang untuk melakukan tindakan di Amazon Inspector

Jika Anda menerima pesan kesalahan bahwa Anda tidak memiliki otorisasi untuk melakukan tindakan, kebijakan Anda harus diperbarui agar Anda dapat melakukan tindakan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM `mateojackson` mencoba menggunakan konsol untuk melihat detail tentang suatu sumber daya *`my-example-widget`* rekaan, tetapi tidak memiliki izin `inspector2:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 inspector2:GetWidget on resource: my-example-widget
```

Dalam hal ini, kebijakan untuk pengguna `mateojackson` harus diperbarui untuk mengizinkan akses ke sumber daya *`my-example-widget`* dengan menggunakan tindakan `inspector2:`*`GetWidget`*.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke Amazon Inspector.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi saat pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di Amazon Inspector . Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses sumber daya Amazon Inspector saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mempelajari apakah Amazon Inspector mendukung fitur ini, lihat Cara kerja Amazon Inspector dengan IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat Menyediakan akses ke pengguna IAM di pengguna lain Akun AWS yang Anda miliki di Panduan Pengguna IAM.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga dalam Panduan Pengguna IAM.
- Untuk mempelajari cara memberikan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna terautentikasi eksternal (federasi identitas) dalam Panduan Pengguna IAM.
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM di Panduan Pengguna IAM.

Memantau Amazon Inspector

Pemantauan adalah bagian penting dalam menjaga ketersediaan, keandalan, dan kinerja Amazon Inspector dan solusi lainnya AWS . AWS menyediakan alat untuk memantau Amazon Inspector, melaporkan masalah yang terjadi, dan mengambil tindakan untuk memperbaiki masalah ini:

- Amazon EventBridge adalah AWS layanan yang menggunakan peristiwa untuk menghubungkan komponen aplikasi bersama-sama, sehingga memudahkan Anda untuk membangun aplikasi berbasis peristiwa yang dapat diskalakan. EventBridge memberikan aliran data real-time dari aplikasi Anda, aplikasi Software-as-a-Service (SaaS), AWS dan layanan dan rute, sehingga Anda dapat memantau peristiwa yang terjadi dalam layanan dan membangun arsitektur berbasis peristiwa.
- AWS CloudTrailadalah AWS layanan yang menangkap panggilan API dan peristiwa terkait yang dibuat oleh atau atas nama Anda Akun AWS. CloudTrail mengirimkan file log ke bucket Amazon S3 yang Anda tentukan, sehingga Anda dapat mengidentifikasi pengguna dan akun mana yang AWS dipanggil, alamat IP sumber dari tempat panggilan dilakukan, dan kapan panggilan terjadi.

Mencatat panggilan Amazon Inspector API menggunakan AWS CloudTrail

Amazon Inspector terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna atau peran IAM, atau, di Amazon Inspector. Layanan AWS CloudTrail

menangkap semua panggilan API untuk Amazon Inspector sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari konsol Amazon Inspector dan panggilan ke operasi Amazon Inspector API. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon S3, termasuk acara untuk Amazon Inspector. Jika Anda tidak membuat konfigurasi jejak, Anda masih dapat melihat kejadian terbaru dalam konsol CloudTrail di Riwayat peristiwa. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan:

- Permintaan yang diajukan ke Amazon Inspector.
- Alamat IP dari mana permintaan dibuat.
- Siapa yang membuat permintaan.
- Saat permintaan dibuat.

Untuk mempelajari selengkapnya CloudTrail, lihat Panduan AWS CloudTrail Pengguna.

Informasi Amazon Inspector di CloudTrail

CloudTrail diaktifkan pada Akun AWS saat Anda membuat akun. Ketika aktivitas terjadi di Amazon Inspector, aktivitas tersebut dicatat dalam suatu CloudTrail peristiwa bersama dengan Layanan AWS peristiwa lain dalam riwayat Acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di situs Anda Akun AWS. Untuk informasi selengkapnya, lihat Melihat peristiwa dengan Riwayat CloudTrail acara.

Untuk catatan acara yang sedang berlangsung di Anda Akun AWS, termasuk acara untuk Amazon Inspector, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah di partisi AWS dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi lainnya Layanan AWS untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat topik berikut:

- Ikhtisar untuk membuat jejak
- CloudTrail layanan dan integrasi yang didukung
- Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail
- Menerima file CloudTrail log dari beberapa akun
- Menerima file CloudTrail log dari berbagai wilayah

Semua tindakan Amazon Inspector dicatat oleh. CloudTrail Semua tindakan yang dapat dilakukan Amazon Inspector didokumentasikan dalam Referensi API Amazon Inspector. Misalnya, panggilan untuk tindakan `CreateFindingsReport`, `ListCoverage`, dan `UpdateOrganizationConfiguration` menghasilkan entri dalam file log CloudTrail .

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas tersebut membantu Anda menentukan hal berikut:

- Apakah permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna IAM.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara atau tidak untuk peran atau pengguna gabungan.
- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

Untuk informasi selengkapnya, lihat Elemen userIdentity CloudTrail .

Memahami entri file log Amazon Inspector

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa menunjukkan satu permintaan dari sumber mana pun. Acara mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Informasi Amazon Inspector Scan di CloudTrail

Amazon Inspector Scan terintegrasi dengan. CloudTrail Semua operasi Amazon Inspector Scan API dicatat sebagai peristiwa manajemen. Untuk daftar operasi API Amazon Inspector Scan yang dicatat oleh Amazon Inspector, lihat Amazon Inspector CloudTrail Scan di Referensi Amazon Inspector API.

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ScanSbom` tindakan:


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROA123456789EXAMPLE:akua_mansa",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/akua_mansa",
        "accountId": "111122223333",
```

```
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROA123456789EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-10-17T15:22:59Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2023-10-17T16:02:34Z",
    "eventSource": "gamma-inspector-scan.amazonaws.com",
    "eventName": "ScanSbom",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "203.0.113.0",
    "userAgent": "aws-sdk-java/2.20.162 Mac_OS_X/13.5.2 OpenJDK_64-
Bit_Server_VM/17.0.8+7-LTS Java/17.0.8 vendor/Amazon.com_Inc. io/sync http/
UrlConnection cfg/retry-mode/legacy",
    "requestParameters": {
        "sbom": {
            "specVersion": "1.5",
            "metadata": {
                "component": {
                    "name": "debian",
                    "type": "operating-system",
                    "version": "9"
                }
            },
            "components": [
                {
                    "name": "packageOne",
                    "purl": "pkg:deb/debian/packageOne@1.0.0?arch=x86_64&distro=9",
                    "type": "application"
                }
            ],
            "bomFormat": "CycloneDX"
        }
    },
```

```
    "responseElements": null,
    "requestID": "f041a27f-f33e-4f70-b09b-5fbc5927282a",
    "eventID": "abc8d1e4-d214-4f07-bc56-8a31be6e36fe",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Validasi Kepatuhan untuk Amazon Inspector

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program AWS Kepatuhan Program AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. Untuk informasi selengkapnya tentang tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS, lihat Dokumentasi AWS Keamanan.

Ketahanan di Amazon Inspector

Infrastruktur AWS global dibangun di sekitar Wilayah AWS dan Availability Zones. Wilayah AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung ke jaringan latensi rendah, throughput tinggi, dan sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Keamanan Infrastruktur di Amazon Inspector

Sebagai layanan terkelola, Amazon Inspector dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat Keamanan

AWS Cloud. Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat Perlindungan Infrastruktur dalam Kerangka Kerja yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses Amazon Inspector melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Respons insiden di Amazon Inspector

Keamanan adalah prioritas tertinggi di AWS. Seperti disebutkan dalam model tanggung jawab AWS bersama di bawah “Keamanan Cloud,” AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan semua layanan di AWS Cloud. AWS Juga bertanggung jawab atas setiap respons insiden yang terkait dengan layanan Amazon Inspector.

Sebagai AWS pelanggan, Anda berbagi tanggung jawab untuk menjaga keamanan di AWS Cloud. Ini berarti Anda mengontrol keamanan yang Anda pilih untuk diterapkan, yang mencakup semua AWS alat dan fitur yang Anda akses. Ini juga berarti Anda bertanggung jawab atas respons insiden di pihak Anda dari model tanggung jawab bersama.

Dengan menetapkan garis dasar keamanan yang memenuhi semua tujuan untuk aplikasi Anda yang berjalan di AWS Cloud, Anda dapat mendeteksi penyimpangan yang dapat Anda tanggapi. Karena respons insiden adalah topik yang kompleks, tinjau sumber daya berikut untuk lebih memahami dampak respons insiden dan bagaimana pilihan Anda dapat memengaruhi tujuan perusahaan Anda: Panduan Respons Insiden AWS Keamanan, Praktik Terbaik AWS Keamanan, dan Kerangka Adopsi AWS Cloud: Perspektif Keamanan.

Akses Amazon Inspector menggunakan titik akhir antarmuka (AWS PrivateLink

Anda dapat menggunakan AWS PrivateLink untuk membuat koneksi pribadi antara VPC Anda dan Amazon Inspector. Anda dapat mengakses Amazon Inspector seolah-olah berada di VPC Anda,

tanpa menggunakan gateway internet, perangkat NAT, koneksi VPN, atau koneksi. Direct Connect Instans di VPC Anda tidak memerlukan alamat IP publik untuk mengakses Amazon Inspector.

Anda membuat koneksi pribadi ini dengan membuat titik akhir antarmuka, yang didukung oleh AWS PrivateLink. Kami membuat antarmuka jaringan endpoint di setiap subnet yang Anda aktifkan untuk titik akhir antarmuka. Ini adalah antarmuka jaringan yang dikelola pemohon yang berfungsi sebagai titik masuk untuk lalu lintas yang ditujukan untuk Amazon Inspector.

Untuk informasi selengkapnya, lihat Akses Layanan AWS melalui AWS PrivateLink di AWS PrivateLink Panduan.

Pertimbangan untuk Amazon Inspector

Sebelum Anda menyiapkan titik akhir antarmuka untuk Amazon Inspector, tinjau Pertimbangan dalam Panduan.AWS PrivateLink

Amazon Inspector mendukung panggilan ke semua tindakan API-nya melalui titik akhir antarmuka.

Kebijakan titik akhir VPC tidak didukung untuk Amazon Inspector. Secara default, akses penuh ke Amazon Inspector diizinkan melalui titik akhir antarmuka. Atau, Anda dapat mengaitkan grup keamanan dengan antarmuka jaringan titik akhir untuk mengontrol lalu lintas ke Amazon Inspector melalui titik akhir antarmuka.

Buat titik akhir antarmuka untuk Amazon Inspector

Anda dapat membuat titik akhir antarmuka untuk Amazon Inspector menggunakan konsol Amazon VPC atau (). AWS Command Line Interface AWS CLI Untuk informasi selengkapnya, lihat Membuat titik akhir antarmuka di AWS PrivateLink Panduan.

Saat Anda membuat titik akhir antarmuka untuk Amazon Inspector, gunakan salah satu nama layanan berikut:

```
com.amazonaws.region.inspector2
```

```
com.amazonaws.region.inspector-scan
```

Ganti *region* dengan AWS Region kode untuk yang berlaku AWS Region.

Jika Anda mengaktifkan DNS pribadi untuk titik akhir antarmuka, Anda dapat membuat permintaan API ke Amazon Inspector menggunakan nama DNS Regional default, misalnya`service-name.us-`

`east-1.amazonaws.com` , `service-name.us-east-1.api.aws.com` atau untuk US East (Virginia N.).

Integrasi Amazon Inspector

Amazon Inspector terintegrasi dengan layanan lain. AWS Layanan ini dapat menyerap data dari Amazon Inspector, sehingga Anda dapat melihat temuan Anda dengan berbagai cara. Tinjau opsi integrasi berikut untuk mempelajari lebih lanjut.

Menggunakan Amazon Inspector dengan AWS Organizations

AWS Organizationsmembantu Anda mengelola dan mengatur lingkungan Anda AWS secara terpusat. Anda dapat menggunakan AWS Organizations kebijakan untuk mengaktifkan dan mengelola Amazon Inspector di beberapa akun di organisasi Anda secara otomatis.

Kebijakan organisasi Amazon Inspector memungkinkan Anda untuk:

- Aktifkan jenis pemindaian Amazon Inspector secara terpusat (EC2, ECR, Lambda, Code Repository) di seluruh organisasi Anda
- Secara otomatis menerapkan pengaktifan Amazon Inspector ke akun baru yang bergabung dengan organisasi
- Menegakkan cakupan pemindaian yang konsisten di seluruh unit organisasi
- Mencegah akun anggota menonaktifkan pemindaian yang diperlukan

Kebijakan organisasi mengontrol pengaktifan jenis sumber daya, sementara administrator yang didelegasikan mempertahankan kendali atas setelan konfigurasi pemindaian. Untuk informasi tentang cara kebijakan organisasi berinteraksi dengan izin administrator dan akun anggota yang didelegasikan, lihat. Mengelola beberapa akun di Amazon Inspector dengan AWS Organizations Untuk petunjuk mendetail tentang cara membuat kebijakan Amazon Inspector, lihat AWS Organizations dokumentasi untuk kebijakan Amazon Inspector.

Mengintegrasikan Amazon Inspector dengan Amazon ECR

Amazon Elastic Container Registry (Amazon ECR) adalah registri gambar kontainer terkelola AWS yang mendukung pendaftar pribadi. Pendaftar pribadi Amazon ECR meng-host gambar kontainer dalam arsitektur yang sangat tersedia dan dapat diskalakan. Anda dapat menggunakan Amazon Inspector untuk memindai gambar kontainer yang berada di repositori Amazon ECR Anda untuk paket sistem operasi yang rentan dan paket bahasa pemrograman. Untuk informasi selengkapnya, lihat Integrasi Amazon Inspector dengan Amazon Elastic Container Registry (Amazon ECR).

Integrasi Amazon Inspector dengan AWS Security Hub CSPM

AWS Security Hub CSPMmemberikan pandangan komprehensif tentang status keamanan Anda AWS dan membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik Security Hub CSPM mengumpulkan data keamanan dari AWS akun, layanan, dan produk yang didukung. Anda dapat menggunakan Security Hub CSPM untuk menyerap data temuan Amazon Inspector dan membuat lokasi terpusat untuk temuan di semua AWS layanan terintegrasi dan produk Jaringan Mitra Anda. AWS Untuk informasi selengkapnya, lihat Integrasi Amazon Inspector dengan AWS Security Hub CSPM.

Integrasi Amazon Inspector dengan Amazon Elastic Container Registry (Amazon ECR)

Amazon Elastic Container Registry adalah registri kontainer yang dikelola sepenuhnya yang mendukung gambar dan artefak Docker dan AWS OCI. Jika Anda menggunakan Amazon ECR, Anda dapat mengaktifkan Enhanced Scanning untuk registri kontainer Anda. Saat Anda mengaktifkan pemindaian yang disempurnakan, Amazon Inspector secara otomatis mendeteksi dan memindai gambar kontainer Anda untuk sistem operasi dan paket bahasa pemrograman yang rentan. Integrasi ini memungkinkan Anda untuk melihat temuan Amazon Inspector untuk gambar kontainer dan mengelola frekuensi dan cakupan pemindaian di konsol Amazon ECR. Untuk informasi selengkapnya, lihat Memindai gambar kontainer Amazon ECR dengan Amazon Inspector.

Mengaktifkan integrasi

Anda dapat mengaktifkan integrasi dengan mengaktifkan pemindaian Amazon Inspector melalui konsol Amazon Inspector atau API, atau dengan mengonfigurasi repositori Anda untuk menggunakan pemindaian yang ditingkatkan dengan Amazon Inspector melalui konsol Amazon ECR atau API.

Untuk informasi selengkapnya tentang mengaktifkan integrasi melalui Amazon Inspector, lihat. Jenis pemindaian otomatis di Amazon Inspector

Untuk informasi tentang mengaktifkan dan mengonfigurasi Pemindaian yang disempurnakan di Amazon ECR, lihat Pemindaian yang Ditingkatkan di panduan pengguna Amazon ECR.

Menggunakan integrasi dengan lingkungan multi-akun

Jika Anda adalah anggota di lingkungan multi-akun, Anda dapat mengaktifkan pemindaian yang disempurnakan melalui Amazon ECR. Namun, setelah diaktifkan, itu hanya dapat dinonaktifkan oleh

administrator delegasi Amazon Inspector Anda. Jika dinonaktifkan, itu kembali ke pemindaian dasar. Untuk informasi selengkapnya, lihat Menonaktifkan Amazon Inspector.

Integrasi Amazon Inspector dengan AWS Security Hub CSPM

Security Hub CSPM memberikan pandangan komprehensif tentang status keamanan Anda di. AWS Ini membantu Anda memeriksa lingkungan Anda terhadap standar industri keamanan dan praktik terbaik. Security Hub CSPM mengumpulkan data keamanan dari AWS akun, layanan, dan produk yang didukung. Anda dapat menggunakan informasi ini untuk menganalisis tren keamanan dan mengidentifikasi masalah keamanan. Saat Anda mengaktifkan integrasi Amazon Inspector dengan Security Hub CSPM, Amazon Inspector dapat mengirimkan temuan ke Security Hub CSPM, dan Security Hub CSPM dapat menganalisis temuan tersebut sebagai bagian dari postur keamanan Anda.

Security Hub CSPM melacak masalah keamanan sebagai temuan. Beberapa temuan dapat menjadi hasil dari masalah keamanan yang terdeteksi di AWS layanan lain atau produk pihak ketiga. Security Hub CSPM menggunakan seperangkat aturan untuk mendeteksi masalah keamanan dan menghasilkan temuan serta menyediakan alat, sehingga Anda dapat mengelola temuan. Security Hub CSPM mengarsipkan temuan Amazon Inspector setelah temuan ditutup di Amazon Inspector. Anda juga dapat melihat riwayat temuan Anda dan menemukan detail, serta melacak status penyelidikan ke dalam sebuah temuan.

Security Hub CSPM memproses temuan dalam AWS Security Finding Format (ASFF). Format ini mencakup detail seperti pengidentifikasi unik, tingkat keparahan, sumber daya yang terpengaruh, panduan remediasi, status alur kerja, dan informasi kontekstual.

> **Note**
>
> Temuan keamanan yang dihasilkan oleh Amazon Inspector Code Security tidak tersedia untuk integrasi ini. Namun, Anda dapat mengakses temuan khusus ini di konsol Amazon Inspector dan melalui Amazon Inspector API.

Topik

- Melihat temuan Amazon Inspector di AWS Security Hub CSPM
- Mengaktifkan dan mengonfigurasi integrasi Amazon Inspector dengan Security Hub CSPM
- Mengaktifkan Amazon Inspector dari Security Hub CSPM menggunakan kebijakan organisasi

- Menonaktifkan aliran temuan dari integrasi
- Melihat kontrol keamanan untuk Amazon Inspector di Security Hub CSPM

Melihat temuan Amazon Inspector di AWS Security Hub CSPM

Anda dapat melihat temuan Amazon Inspector Classic dan Amazon Inspector di Security Hub CSPM.

> **Note**
>
> Untuk memfilter hanya temuan Amazon Inspector, tambahkan `"aws/inspector/ProductVersion": "2"` ke bilah filter. Filter ini mengecualikan temuan Amazon Inspector Classic dari dasbor CSPM Security Hub.

Contoh temuan dari 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"
}
```


Mengaktifkan dan mengonfigurasi integrasi Amazon Inspector dengan Security Hub CSPM

Anda dapat mengaktifkan integrasi Amazon Inspector AWS Security Hub CSPM dengan mengaktifkan Security Hub CSPM. Setelah mengaktifkan Security Hub CSPM, integrasi Amazon Inspector AWS Security Hub CSPM dengan otomatis diaktifkan, dan Amazon Inspector mulai mengirimkan semua temuannya ke Security Hub CSPM menggunakan AWS Security Finding Format (ASFF).

Mengaktifkan Amazon Inspector dari Security Hub CSPM menggunakan kebijakan organisasi

Anda dapat mengelola aktivasi Amazon Inspector di seluruh AWS organisasi menggunakan kebijakan Organizations langsung dari konsol CSPM Security Hub. Pendekatan terpusat ini memungkinkan Anda mengaktifkan pemindaian Amazon Inspector untuk beberapa akun secara bersamaan melalui manajemen kebijakan tingkat organisasi.

Untuk petunjuk mendetail tentang mengelola aktivasi Amazon Inspector melalui CSPM Security Hub menggunakan kebijakan organisasi, lihat Mengelola akun administrator yang didelegasikan untuk CSPM Security Hub di Panduan Pengguna.AWS Security Hub CSPM

Menonaktifkan aliran temuan dari integrasi

Untuk menghentikan Amazon Inspector mengirimkan temuan ke Security Hub CSPM, Anda dapat menggunakan konsol CSPM Security Hub atau API dan.. AWS CLI

Melihat kontrol keamanan untuk Amazon Inspector di Security Hub CSPM

Security Hub CSPM menganalisis temuan dari produk yang didukung AWS dan pihak ketiga dan menjalankan pemeriksaan keamanan otomatis dan berkelanjutan terhadap aturan untuk menghasilkan temuannya sendiri. Aturan diwakili oleh kontrol keamanan, yang membantu Anda menentukan apakah persyaratan dalam standar terpenuhi.

Amazon Inspector menggunakan kontrol keamanan untuk memeriksa apakah fitur Amazon Inspector telah atau harus diaktifkan. Fitur-fitur ini mencakup hal-hal berikut:

- Pemindaian Amazon EC2
- Pemindaian ECR Amazon
- Pemindaian standar Lambda
- Pemindaian kode Lambda

Untuk informasi selengkapnya, lihat kontrol Amazon Inspector di AWS Security Hub CSPM Panduan Pengguna.

Sistem operasi dan bahasa pemrograman yang didukung untuk Amazon Inspector

Amazon Inspector dapat memindai aplikasi perangkat lunak yang diinstal pada berikut ini:

- Instans Amazon Elastic Compute Cloud (Amazon EC2)

  > **Note**
  >
  > Untuk instans Amazon EC2, Amazon Inspector dapat memindai kerentanan paket di sistem operasi yang mendukung pemindaian berbasis agen. Amazon Inspector juga dapat memindai kerentanan paket dalam sistem operasi dan bahasa pemrograman yang mendukung pemindaian hibrida. Amazon Inspector tidak memindai kerentanan toolchain. Versi kompiler bahasa pemrograman yang digunakan untuk membangun aplikasi memperkenalkan kerentanan ini.

- Gambar kontainer disimpan di repositori Amazon Elastic Container Registry (Amazon ECR)

  > **Note**
  >
  > Untuk gambar kontainer ECR, Amazon Inspector dapat memindai sistem operasi dan kerentanan paket bahasa pemrograman. Amazon Inspector juga mendukung gambar yang diperkeras yang disediakan oleh Chainguard dan Minimus. Amazon Inspector tidak memindai kerentanan toolchain di Rust ——versi kompiler bahasa pemrograman yang digunakan untuk membangun aplikasi memperkenalkan kerentanan ini.

- AWS Lambda fungsi

  > **Note**
  >
  > Untuk fungsi Lambda, Amazon Inspector dapat memindai kerentanan paket bahasa pemrograman dan kerentanan kode. Amazon Inspector tidak memindai kerentanan toolchain. Versi kompiler bahasa pemrograman yang digunakan untuk membangun aplikasi memperkenalkan kerentanan ini.

Saat Amazon Inspector memindai sumber daya, Amazon Inspector memberikan lebih dari 50 feed data untuk menghasilkan temuan untuk kerentanan dan eksposur umum (). CVEs Contoh sumber ini termasuk penasihat keamanan vendor, umpan data, dan umpan intelijen ancaman, serta National Vulnerability Database (NVD) dan MITRE. Amazon Inspector memperbarui data kerentanan dari umpan sumber setidaknya sekali sehari.

Agar Amazon Inspector dapat memindai sumber daya, sumber daya harus menjalankan sistem operasi yang didukung atau menggunakan bahasa pemrograman yang didukung. Topik di bagian ini mencantumkan sistem operasi, bahasa pemrograman, dan runtime yang didukung Amazon Inspector untuk berbagai sumber daya dan jenis pemindaian. Mereka juga mencantumkan sistem operasi yang dihentikan.

> **Note**
>
> Amazon Inspector hanya dapat memberikan dukungan terbatas untuk sistem operasi setelah vendor menghentikan dukungan untuk sistem operasi.

Topik

Sistem operasi yang didukung

Bagian ini mencantumkan sistem operasi yang didukung Amazon Inspector.

Sistem operasi yang didukung: Pemindaian Amazon EC2

Tabel berikut mencantumkan sistem operasi yang didukung Amazon Inspector untuk pemindaian instans Amazon EC2. Ini menentukan penasihat keamanan vendor untuk setiap sistem operasi dan sistem operasi mana yang mendukung pemindaian berbasis agen dan pemindaian tanpa agen.

Saat menggunakan metode pemindaian berbasis agen, Anda mengonfigurasi agen SSM untuk melakukan pemindaian berkelanjutan pada semua instance yang memenuhi syarat. Amazon

Inspector merekomendasikan agar Anda mengonfigurasi versi agen SSM yang lebih besar dari 3.2.2086.0. Untuk informasi selengkapnya, lihat Bekerja dengan Agen SSM di Panduan Pengguna Amazon EC2 Systems Manager.

Deteksi sistem operasi Linux hanya didukung untuk repositori manajer paket default (rpm dan dpkg) dan tidak termasuk aplikasi pihak ketiga, repositori dukungan yang diperluas (RHEL EUS, E4S, AUS, dan TUS), dan repositori opsional (aliran aplikasi). Amazon Inspector memindai kernel yang sedang berjalan untuk mencari kerentanan. Untuk beberapa sistem operasi, sepertiUbuntu, reboot diperlukan untuk peningkatan agar ditampilkan dalam temuan aktif.

Sistem operasi	Versi	Penasihat keamanan vendor	Dukungan pemindaian tanpa agen	Dukungan pemindaian berbasis agen
AlmaLinux	8	Errata CVE	Ya	Ya
AlmaLinux	9	Errata CVE	Ya	Ya
AlmaLinux	10	Errata CVE	Tidak	Ya
Amazon Linux (AL2)	AL2	ALAS Errata CVE	Ya	Ya
Amazon Linux 2023 () AL2023	AL2023	ALAS Errata CVE	Ya	Ya
Bottlerocket	1.7.0 dan kemudian	Errata CVE	Tidak	Ya
Server Debian (Bullseye)	11	DSA CVE	Ya	Ya
Server Debian (Kutu Buku)	12	DSA CVE	Ya	Ya
Server Debian (Trixie)	13	DSA CVE	Ya	Ya
Fedora	42	Errata CVE	Ya	Ya

Sistem operasi	Versi	Penasihat keamanan vendor	Dukungan pemindaian tanpa agen	Dukungan pemindaian berbasis agen
Lompatan openSUSE	15.6	Errata CVE	Ya	Ya
Oracle Linux (Oracle)	8	Errata CVE	Ya	Ya
Oracle Linux (Oracle)	9	Errata CVE	Ya	Ya
Oracle Linux (Oracle)	10	Errata CVE	Tidak	Ya
Red Hat Enterprise Linux (RHEL)	8	RHEL VEX CVE	Ya	Ya
Red Hat Enterprise Linux (RHEL)	9	RHEL VEX CVE	Ya	Ya
Red Hat Enterprise Linux (RHEL)	10	RHEL VEX CVE	Tidak	Ya
Linux Rocky	8	Errata CVE	Ya	Ya
Linux Rocky	9	Errata CVE	Ya	Ya
Linux Rocky	10	Errata CVE	Tidak	Ya
Server Perusahaan SUSE Linux (SLES)	15.7	SUSE CVE	Ya	Ya

Sistem operasi	Versi	Penasihat keamanan vendor	Dukungan pemindaian tanpa agen	Dukungan pemindaian berbasis agen
Ubuntu (Xenial)	16.04	USN, Ubuntu Pro (esm-infra & esm aplikasi)	Ya	Ya
Ubuntu (Bionik)	18.04	USN, Ubuntu Pro (esm-infra & esm aplikasi)	Ya	Ya
Ubuntu (Fokus)	20.04	USN, Ubuntu Pro (esm-infra & esm aplikasi)	Ya	Ya
Ubuntu (Jammy)	22.04	USN, Ubuntu Pro (esm-infra & esm aplikasi)	Ya	Ya
Ubuntu (Noble Numbat)	24.04	USN, Ubuntu Pro (esm-infra & esm-apps)	Ya	Ya
Windows Server	2016	MSKB	Tidak	Ya
Windows Server	2019	MSKB	Tidak	Ya
Windows Server	2022	MSKB	Tidak	Ya
Windows Server	2025	MSKB	Tidak	Ya
macOS (Mojave)	10.14	APEL-SA	Tidak	Ya
macOS (Catalina)	10.15	APEL-SA	Tidak	Ya
macOS (Big Sur)	11	APEL-SA	Tidak	Ya

Sistem operasi	Versi	Penasihat keamanan vendor	Dukungan pemindaian tanpa agen	Dukungan pemindaian berbasis agen
macOS (Monterey)	12	APEL-SA	Tidak	Ya
macOS (Ventura)	13	APEL-SA	Tidak	Ya
macOS (Sonoma)	14	APEL-SA	Tidak	Ya
macOS (Sequoia)	15	APEL-SA	Tidak	Ya

Sistem operasi yang didukung: Pemindaian Amazon ECR dengan Amazon Inspector

Tabel berikut mencantumkan sistem operasi yang didukung Amazon Inspector untuk pemindaian gambar kontainer di repositori Amazon ECR. Ini juga menentukan penasihat keamanan vendor untuk setiap sistem operasi.

Sistem operasi	Versi	Penasihat keamanan vendor
AlmaLinux	8	Errata CVE
AlmaLinux	9	Errata CVE
AlmaLinux	10	Errata CVE
Alpine Linux (Alpine)	3.20	Errata CVE
Alpine Linux (Alpine)	3.21	Errata CVE
Alpine Linux (Alpine)	3.22	Errata CVE
Alpine Linux (Alpine)	3.23	Errata CVE

Sistem operasi	Versi	Penasihat keamanan vendor
Amazon Linux (AL2)	AL2	CVE
Amazon Linux 2023 (AL2023)	AL2023	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	42	Errata CVE
Minimus	–	Errata CVE
OpenSUSE Leap	15.6	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 VEX CVE
Red Hat Enterprise Linux (RHEL)	9	RHEL VEX CVE
Red Hat Enterprise Linux (RHEL)	10	RHEL VEX CVE

Sistem operasi	Versi	Penasihat keamanan vendor
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 (Xenial)	16.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Bionic)	18.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Focal)	20.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Jammy)	22.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Ubuntu (Noble Numbat)	24.04	USN, Ubuntu Pro (esm-infra & esm-apps)
Wolfi	–	Errata CVE

Sistem operasi yang didukung: pemindaian CIS

Tabel berikut mencantumkan sistem operasi yang didukung Amazon Inspector untuk pemindaian CIS. Ini juga menentukan versi benchmark CIS untuk setiap sistem operasi.

> **Note**
>
> Standar CIS ditujukan untuk sistem operasi x86_64. Beberapa pemeriksaan mungkin tidak dievaluasi atau mengembalikan instruksi remediasi yang tidak valid pada sumber daya berbasis ARM.

Sistem operasi	Versi	Versi 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
Linux Rocky	8	2.0.0
Linux Rocky	9	1.0.0
Server Perusahaan SUSE Linux	15	2.0.1
Ubuntu (Bionik)	18.04	2.2.0
Ubuntu (Fokus)	20.04	3.0.0
Ubuntu (Jammy)	22.04	2.0.0
Ubuntu (Numbat Mulia)	24.04	1.0.0
Windows Server	2016	3.0.0
Windows Server	2019	4.0.0
Windows Server	2022	4.0.0
Windows Server	2025	1.0.0

Sistem operasi yang didukung: Amazon Inspector Scan API

Tabel berikut mencantumkan sistem operasi yang didukung untuk Amazon Inspector Scan API. Untuk informasi selengkapnya, lihat ScanSbomdi Referensi API Amazon Inspector V2.

Sistem operasi	Versi
AlmaLinux 8	8
AlmaLinux	9
AlmaLinux	10
Alpine Linux	3.20
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
Debian Sid	–
Echo	2
Fedora	42
Fedora	43
macOS	11+

Sistem operasi	Versi
MinimOS	–
OpenSUSE	15.6
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
Ubuntu	16.04
Ubuntu	18.04
Ubuntu	20.04
Ubuntu	22.04
Ubuntu	24.04
Ubuntu	25.10

Sistem operasi	Versi
Wolfi Linux	–

Sistem operasi yang dihentikan

Tabel berikut mencantumkan sistem operasi yang telah dihentikan dan ketika mereka dihentikan.

Meskipun Amazon Inspector tidak memberikan dukungan penuh untuk sistem operasi yang dihentikan, Amazon Inspector terus memindai instans Amazon EC2 dan image container Amazon ECR yang menjalankannya. Sebagai praktik keamanan terbaik, kami sarankan untuk beralih ke versi yang didukung. Temuan yang dihasilkan Amazon Inspector untuk sistem operasi yang dihentikan harus digunakan hanya untuk tujuan informasi.

Sesuai dengan kebijakan vendor, sistem operasi yang dihentikan tidak lagi menerima pembaruan tambalan. Penasihat keamanan baru mungkin tidak dirilis untuk sistem operasi yang dihentikan. Vendor dapat menghapus saran dan deteksi keamanan yang ada dari feed mereka untuk sistem operasi yang mencapai akhir dukungan standar. Akibatnya, Amazon Inspector dapat berhenti menghasilkan temuan untuk diketahui. CVEs

Sistem operasi	Versi	Dihentikan
Alpine Linux (Alpine)	3.2	1 Mei 2017
Alpine Linux (Alpine)	3.3	1 November 2017
Alpine Linux (Alpine)	3.4	1 Mei 2018
Alpine Linux (Alpine)	3.5	1 November 2018
Alpine Linux (Alpine)	3.6	1 Mei 2019
Alpine Linux (Alpine)	3.7	1 November 2019
Alpine Linux (Alpine)	3.8	Selasa, 01 Mei 2020
Alpine Linux (Alpine)	3.9	1 November 2020
Alpine Linux (Alpine)	3.10	Mei 1, 2021

Sistem operasi	Versi	Dihentikan
Alpine Linux (Alpine)	3.11	November 1, 2021
Alpine Linux (Alpine)	3.12	1 Mei 2022
Alpine Linux (Alpine)	3.13	1 November 2022
Alpine Linux (Alpine)	3.14	1 Mei 2023
Alpine Linux (Alpine)	3.15	1 November 2023
Alpine Linux (Alpine)	3.16	23 Mei 2024
Alpine Linux (Alpine)	3.17	November 22, 2024
Alpine Linux (Alpine)	3.18	9 Mei 2025
Alpine Linux (Alpine)	3.19	November 1, 2025
Amazon Linux (AL1)	2012	Desember 31, 2021
CentOS Linux (CentOS)	7	30 Juni 2024
CentOS Linux (CentOS)	8	Desember 31, 2021
Server Debian (Jessie)	8	30 Juni 2020
Server Debian (Peregangan)	9	30 Juni 2022
Server Debian (Buster)	10	30 Juni 2024
Fedora	33	30 November 2021
Fedora	34	7 Juni 2022
Fedora	35	13 Desember 2022
Fedora	36	16 Mei 2023
Fedora	37	15 Desember 2023

Sistem operasi	Versi	Dihentikan
Fedora	38	21 Mei 2024
Fedora	39	November 26, 2024
Fedora	40	13 Mei 2025
Fedora	41	November 19, 2025
Lompatan openSUSE	15.2	1 Desember 2021
Lompatan openSUSE	15.3	Desember 1, 2022
Lompatan openSUSE	15.4	Desember 7, 2023
Lompatan openSUSE	15.5	Desember 31, 2024
Oracle Linux (Oracle)	6	1 Maret 2021
Oracle Linux (Oracle)	7	Desember 31, 2024
Foton OS	2	2 Desember 2021
Foton OS	3	1 Maret 2024
Red Hat Enterprise Linux (RHEL)	6	30 Juni 2020
Red Hat Enterprise Linux (RHEL)	7	30 Juni 2024
Server Perusahaan SUSE Linux (SLES)	12	30 Juni 2016
Server Perusahaan SUSE Linux (SLES)	12.1	31 Mei 2017
Server Perusahaan SUSE Linux (SLES)	12.2	Maret 31, 2018
Server Perusahaan SUSE Linux (SLES)	12.3	Juni 30, 2019

Sistem operasi	Versi	Dihentikan
Server Perusahaan SUSE Linux (SLES)	12.4	30 Juni 2020
Server Perusahaan SUSE Linux (SLES)	12,5	Oktober 31, 2024
Server Perusahaan SUSE Linux (SLES)	15	Desember 31, 2019
Server Perusahaan SUSE Linux (SLES)	15.1	Januari 31, 2021
Server Perusahaan SUSE Linux (SLES)	15.2	Desember 31, 2021
Server Perusahaan SUSE Linux (SLES)	15.3	Desember 31, 2022
Server Perusahaan SUSE Linux (SLES)	15.4	Desember 31, 2023
Server Perusahaan SUSE Linux (SLES)	15.5	Desember 31, 2024
Server Perusahaan SUSE Linux (SLES)	15.6	Desember 31, 2025
Ubuntu (Terpercaya)	12.04	28 April 2017
Ubuntu (Terpercaya)	14.04	April 1, 2024
Ubuntu (Groovy)	20.10	22 Juli 2021
Ubuntu (Berrambut)	21.04	20 Januari 2022
Ubuntu (Jahat)	21.10	Juli 31, 2022
Ubuntu (Kinetik)	22.10	Juli 20, 2023

Sistem operasi	Versi	Dihentikan
Ubuntu (Lobster Bulan)	23.04	Januari 25, 2024
Ubuntu (Mantic Minotaur)	23.10	Juli 11, 2024
Ubuntu (Oriole Orakular)	24.10	Juli 10, 2025
Ubuntu (Puffin yang kuat)	25.04	Januari 15, 2026
Windows Server	2012	10 Oktober 2023
Windows Server	2012 R2	10 Oktober 2023

Bahasa pemrograman yang didukung

Bagian ini mencantumkan bahasa pemrograman yang didukung Amazon Inspector.

Bahasa pemrograman yang didukung: Amazon EC2 pemindaian tanpa agen

Amazon Inspector saat ini mendukung bahasa pemrograman berikut saat melakukan pemindaian tanpa agen pada instans Amazon EC2 yang memenuhi syarat. Untuk informasi selengkapnya, lihat pemindaian tanpa agen.

> **Note**
>
> Amazon Inspector tidak memindai kerentanan toolchain di dalam dan. Go Rust Versi kompiler bahasa pemrograman yang digunakan untuk membangun aplikasi memperkenalkan kerentanan ini.

- C#
- Go
- Java
- JavaScript
- PHP

- Python
- Ruby
- Rust

Bahasa pemrograman yang didukung: Inspeksi mendalam Amazon EC2

Amazon Inspector saat ini mendukung bahasa pemrograman berikut saat melakukan pemindaian inspeksi mendalam pada instans Amazon EC2 Linux. Untuk informasi selengkapnya, lihat inspeksi mendalam Amazon Inspector untuk instans Amazon EC2 berbasis Linux.

- Java(format arsip.ear, .jar, .par, dan .war)
- JavaScript
- Python

Amazon Inspector menggunakan Systems Manager Distributor untuk menyebarkan plugin untuk pemeriksaan mendalam instans Amazon EC2 Anda.

> **Note**
>
> Inspeksi mendalam tidak didukung untuk sistem operasi Bottlerocket.

Untuk melakukan pemindaian inspeksi mendalam, Systems Manager Distributor dan Amazon Inspector harus mendukung sistem operasi instans Amazon EC2 Anda. Untuk informasi tentang sistem operasi yang didukung di Distributor Systems Manager, lihat Platform dan arsitektur paket yang didukung di Panduan Pengguna Systems Manager.

Bahasa pemrograman yang didukung: Pemindaian Amazon ECR

Amazon Inspector saat ini mendukung bahasa pemrograman berikut saat memindai gambar kontainer di repositori Amazon ECR:

> **Note**
>
> Amazon Inspector tidak memindai kerentanan toolchain di. Rust Versi kompiler bahasa pemrograman yang digunakan untuk membangun aplikasi memperkenalkan kerentanan ini.

Untuk Python aplikasi yang menggunakan ChainguardLibrary, Amazon Inspector mengenali perbaikan keamanan back-ported dan mengecualikannya dari temuan.

- C#
- Go
- Gorantai alat
- Java
- JavaJDK
- JavaScript
- PHP
- Python(termasuk Chainguard Perpustakaan)
- Ruby
- Rust

Waktu aktif yang didukung

Bagian ini mencantumkan runtime yang didukung Amazon Inspector.

Runtime yang didukung: Pemindaian standar Amazon Inspector Lambda

Pemindaian standar Amazon Inspector Lambda saat ini mendukung runtime berikut untuk bahasa pemrograman yang dapat digunakan saat memindai fungsi Lambda untuk kerentanan dalam paket perangkat lunak pihak ketiga:

Note

Amazon Inspector tidak memindai kerentanan toolchain di. Rust Versi kompiler bahasa pemrograman yang digunakan untuk membangun aplikasi memperkenalkan kerentanan ini.

- Go
  - go1.x
- Java
  - java8

  - java8.al2
  - java11
  - java17
  - java21
- .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 yang didukung: Pemindaian kode Amazon Inspector Lambda

Pemindaian kode Amazon Inspector Lambda saat ini mendukung runtime berikut untuk bahasa pemrograman yang dapat digunakan saat memindai fungsi Lambda untuk kerentanan dalam kode:

- Java
  - java8
  - java8.al2
  - java11
  - java17
- .NET
  - .NET 6
  - .NET 8
- Node.js
  - nodejs12.x
  - nodejs14.x
  - nodejs16.x
  - nodejs18.x
  - nodejs20.x
- Python
  - python3.7
  - python3.8
  - python3.9
  - python3.10
  - python3.11
  - python3.12
- Ruby
  - ruby2.7
  - ruby3.2
  - ruby3.3

Menonaktifkan Amazon Inspector

Anda dapat menonaktifkan Amazon Inspector di konsol Amazon Inspector atau dengan Amazon Inspector API. Jika Anda menonaktifkan semua jenis pemindaian untuk akun; Amazon Inspector dinonaktifkan untuk akun tersebut secara otomatis.

Jika Anda menonaktifkan Amazon Inspector untuk akun, semua jenis pemindaian dinonaktifkan untuk akun tersebut. Selain itu, semua pengaturan pemindaian Amazon Inspector, termasuk filter, aturan penekanan, dan temuan akan dihapus untuk akun tersebut.

Saat Anda menonaktifkan pemindaian Amazon Inspector EC2 Amazon, Amazon Inspector menghapus asosiasi SSM berikut:

- `InspectorDistributor-do-not-delete`
- `InspectorInventoryCollection-do-not-delete`
- `InvokeInspectorSsmPlugin-do-not-delete`. Selain itu, plugin Amazon Inspector SSM yang diinstal melalui asosiasi ini dihapus dari semua host Anda. Windows Untuk informasi selengkapnya, lihat Memindai Windows contoh EC2.

> **Note**
>
> Setelah Anda menonaktifkan Amazon Inspector, Anda tidak lagi dikenakan biaya layanan. Namun, Anda dapat mengaktifkan kembali Amazon Inspector kapan saja.

Untuk informasi tentang cara menonaktifkan jenis pemindaian untuk sumber daya yang berbeda, lihat Menonaktifkan jenis pemindaian.

Prasyarat

Tergantung pada jenis akun, pertimbangkan hal berikut:

- Jika akun Anda adalah akun Amazon Inspector mandiri, Anda dapat menonaktifkan Amazon Inspector kapan saja.
- Jika akun Anda adalah akun anggota di lingkungan multi-akun, Anda tidak dapat menonaktifkan Amazon Inspector. Anda harus menghubungi administrator yang didelegasikan untuk organisasi Anda untuk menonaktifkan Amazon Inspector.

- Jika Anda adalah administrator yang didelegasikan untuk organisasi, Anda harus memisahkan semua akun anggota sebelum menonaktifkan Amazon Inspector.
- Jika pengaktifan Amazon Inspector akun Anda dikelola oleh AWS Organizations kebijakan, Anda tidak dapat menonaktifkan jenis pemindaian yang dikelola kebijakan melalui konsol Amazon Inspector atau API. Untuk menonaktifkan jenis pemindaian Amazon Inspector, Anda harus mengubah kebijakan organisasi untuk menonaktifkannya secara eksplisit melalui AWS Organizations konsol atau API. Anda dapat menonaktifkan jenis pemindaian yang tidak dikelola oleh kebijakan organisasi melalui konsol Amazon Inspector atau API.

> **Note**
>
> Saat menonaktifkan Amazon Inspector sebagai administrator yang didelegasikan, Anda menonaktifkan fitur aktivasi otomatis untuk organisasi Anda.

Menonaktifkan Amazon Inspector yang dikelola oleh kebijakan organisasi

Jika Amazon Inspector diaktifkan di akun Anda melalui AWS Organizations kebijakan, Anda harus menggunakan AWS Organizations konsol atau API untuk menonaktifkan Inspector. Akun anggota dan administrator yang didelegasikan tidak dapat menonaktifkan jenis pemindaian yang dikelola kebijakan melalui konsol Amazon Inspector atau API.

Untuk menonaktifkan Amazon Inspector untuk akun yang dikelola kebijakan:

Untuk menonaktifkan pemberdayaan Amazon Inspector yang dikelola kebijakan

1. Masuk ke akun AWS Organizations manajemen atau akun administrator kebijakan.
2. Ubah kebijakan organisasi untuk secara eksplisit menyetel jenis pemindaian ke dinonaktifkan di wilayah tempat Anda ingin menonaktifkan Inspector. Anda harus memperbarui konten kebijakan untuk menentukan wilayah yang dinonaktifkan untuk jenis pemindaian yang ingin dinonaktifkan.
3. AWS Organizations akan secara otomatis menerapkan perubahan kebijakan, dan Amazon Inspector akan menonaktifkan jenis pemindaian yang ditentukan di akun yang terpengaruh.

Untuk petunjuk mendetail tentang memodifikasi atau melepaskan kebijakan organisasi, lihat AWS Organizations dokumentasi untuk kebijakan Amazon Inspector.

> **Note**
>
> Saat Anda melepaskan kebijakan organisasi dari akun, akun tersebut akan mempertahankan setelan Amazon Inspector mereka saat ini (diaktifkan atau dinonaktifkan berdasarkan kebijakan terakhir yang diterapkan). Akun tidak lagi dikelola oleh kebijakan dan kemudian dapat mengelola pengaturan Amazon Inspector mereka secara independen atau melalui administrator yang didelegasikan.

Nonaktifkan Amazon Inspector

> **Note**
>
> Sebelum Anda menonaktifkan Amazon Inspector, pertimbangkan untuk mengekspor temuan Anda.

Console

Untuk menonaktifkan Amazon Inspector

1. Masuk menggunakan kredensil Anda, lalu buka konsol Amazon Inspector di v2/home. https://console.aws.amazon.com/inspector/
2. Dengan menggunakan AWS Region pemilih di sudut kanan atas halaman, pilih Wilayah tempat Anda ingin menonaktifkan Amazon Inspector.
3. Di panel navigasi, pilih Pengaturan umum.
4. Pilih Nonaktifkan Inspector.
5. Saat diminta konfirmasi, masukkan nonaktifkan di kotak teks, lalu pilih Nonaktifkan Inspector.
6. (Disarankan) Ulangi langkah-langkah ini di setiap Wilayah yang ingin Anda nonaktifkan Amazon Inspector.

API

Jalankan operasi Nonaktifkan API. Dalam permintaan, berikan akun yang IDs Anda nonaktifkan, dan `EC2, ECR, LAMBDA` `resourceTypes` untuk menonaktifkan semua pemindaian, yang akan menonaktifkan akun.

Kuota Amazon Inspector

Bagian ini mencantumkan kuota Amazon Inspector per. AWS Region

Sumber Daya	Default	Komentar
Akun anggota	10.000	Jumlah maksimum akun anggota yang terkait dengan akun administrator yang didelegasikan Amazon Inspector . Batas didasarkan pada Kuota untuk AWS Organizations.
Aturan penekanan	500	Jumlah maksimum aturan penekanan yang disimpan per AWS akun per Wilayah. Anda tidak dapat meminta kenaikan kuota.
Temuan EC2 jaringan Amazon	10.000	Jumlah maksimum temuan EC2 jaringan Amazon per AWS akun. Anda tidak dapat meminta kenaikan kuota.
Konfigurasi pemindaian CIS	500	Jumlah maksimum konfigurasi pemindaian CIS. Anda tidak dapat meminta kenaikan kuota.

Untuk daftar kuota yang terkait dengan Amazon Inspector Classic, lihat Kuota layanan Amazon Inspector Classic di. Referensi Umum AWS Untuk daftar kuota yang terkait dengan AWS Organizations, lihat kuota AWS Organizations layanan di. Referensi Umum AWS

Wilayah dan titik akhir

Topik ini mencakup tabel yang menunjukkan titik akhir untuk Amazon Inspector dan Amazon Inspector Scan. Ini juga mencakup tabel yang menunjukkan yang Wilayah AWS mendukung fitur Amazon Inspector. Untuk melihat di Wilayah AWS mana Amazon Inspector tersedia, lihat titik akhir Amazon Inspector dan kuota di. Referensi Umum Amazon Web Services

Titik akhir layanan untuk Amazon Inspector

Tabel berikut menunjukkan titik akhir layanan untuk Amazon Inspector. Konvensi penamaan untuk titik akhir Amazon Inspector adalah. `inspector2.Region.amazonaws.com`

Nama Wilayah	Wilayah	Titik Akhir	Protokol
AS Timur (Ohio)	us–east-2	inspector2.us-east-2.amazonaws.com inspector2-fips.us-east-2.amazonaws.com	HTTPS HTTPS
AS Timur (Virginia Utara)	us-east-1	inspector2.us-east-1.amazonaws.com inspector2-fips.us-east-1.amazonaws.com	HTTPS HTTPS
US West (N. California)	us-west-1	inspector2.us-west-1.amazonaws.com inspector2-fips.us-west-1.amazonaws.com	HTTPS HTTPS
AS Barat (Oregon)	us-west-2	inspector2.us-west-2.amazonaws.com inspector2-fips.us-west-2.amazonaws.com	HTTPS HTTPS
Afrika (Cape Town)	af-south-1	inspector2.af-south-1.amazonaws.com	HTTPS
Asia Pasifik	ap-east-1	inspector2.ap-east-1.amazonaws.com	HTTPS

Nama Wilayah	Wilayah	Titik Akhir	Protokol	
(Hong Kong)				
Asia Pasifik (Hyderabad)	ap-south-2	inspector2.ap-south-2.amazonaws.com	HTTPS	
Asia Pasifik (Jakarta)	ap-southeast-3	inspector2.ap-southeast-3.amazonaws.com	HTTPS	
Asia Pasifik (Malaysia)	ap-southeast-5	inspector2.ap-southeast-5.amazonaws.com	HTTPS	
Asia Pasifik (Melbourne)	ap-southeast-4	inspector2.ap-southeast-4.amazonaws.com	HTTPS	
Asia Pasifik (Mumbai)	ap-south-1	inspector2.ap-south-1.amazonaws.com	HTTPS	
Asia Pasifik (Osaka)	ap-northeast-3	inspector2.ap-northeast-3.amazonaws.com	HTTPS	
Asia Pasifik (Seoul)	ap-northeast-2	inspector2.ap-northeast-2.amazonaws.com	HTTPS	

Nama Wilayah	Wilayah	Titik Akhir	Protokol	
Asia Pasifik (Singapura)	ap-southeast-1	inspector2.ap-southeast-1.amazonaws.com	HTTPS	
Asia Pasifik (Sydney)	ap-southeast-2	inspector2.ap-southeast-2.amazonaws.com	HTTPS	
Asia Pasifik (Thailand)	ap-tenggara-7	inspector2.ap-southeast-7.amazonaws.com	HTTPS	
Asia Pacific (Tokyo)	ap-northeast-1	inspector2.ap-northeast-1.amazonaws.com	HTTPS	
Kanada (Pusat)	ca-central-1	inspector2.ca-central-1.amazonaws.com	HTTPS	
Kanada Barat (Calgary)	ca-west-1	inspector2.ca-west-1.amazonaws.com	HTTPS	
Eropa (Frankfurt)	eu-central-1	inspector2.eu-central-1.amazonaws.com	HTTPS	
Eropa (Irlandia)	eu-west-1	inspector2.eu-west-1.amazonaws.com	HTTPS	
Eropa (London)	eu-west-2	inspector2.eu-west-2.amazonaws.com	HTTPS	

Nama Wilayah	Wilayah	Titik Akhir	Protokol	
Eropa (Milan)	eu-south-1	inspector2.eu-south-1.amazonaws.com	HTTPS	
Eropa (Paris)	eu-west-3	inspector2.eu-west-3.amazonaws.com	HTTPS	
Eropa (Spanyol)	eu-south-2	inspector2.eu-south-2.amazonaws.com	HTTPS	
Eropa (Stockholm)	eu-north-1	inspector2.eu-north-1.amazonaws.com	HTTPS	
Eropa (Zürich)	eu-central-2	inspector2.eu-central-2.amazonaws.com	HTTPS	
Israel (Tel Aviv)	il-central-1	inspector2.il-central-1.amazonaws.com	HTTPS	
Meksiko (Tengah)	mx-pusat-1	inspector2.mx-central-1.amazonaws.com	HTTPS	
Timur Tengah (Bahrain)	me-south-1	inspector2.me-south-1.amazonaws.com	HTTPS	
Timur Tengah (UAE)	me-central-1	inspector2.me-central-1.amazonaws.com	HTTPS	
Amerika Selatan (Sao Paulo)	sa-east-1	inspector2.sa-east-1.amazonaws.com	HTTPS	

Nama Wilayah	Wilayah	Titik Akhir	Protokol
AWS GovCloud (AS-Timur)	us-gov-east-1	inspector2.us-gov-east-1.amazonaws.com inspector2-fips.us-gov-east-1.amazonaws.com	HTTPS HTTPS
AWS GovCloud (AS-Barat)	us-gov-west-1	inspector2.us-gov-west-1.amazonaws.com inspector2-fips.us-gov-west-1.amazonaws.com	HTTPS HTTPS

Titik akhir untuk Amazon Inspector Scan API

Tabel berikut menunjukkan titik akhir Regional yang dapat digunakan saat memanggil Amazon Inspector Scan API. Saat menggunakan API, Anda harus menyediakan titik akhir dan itu adalah Wilayah yang sesuai untuk Wilayah yang saat ini Anda autentikasi. AWS

Konvensi penamaan untuk titik akhir Amazon Inspector Scan adalah. `inspector-scan.`*`region`*`.amazonaws.com` Misalnya, jika Anda diautentikasi`us-west-2`, Anda akan menggunakan titik akhir `inspector-scan.us-west-2.amazonaws.com` untuk memanggil API. `inspector-scan`

Nama Wilayah	Wilayah	Titik Akhir	Protokol
AS Timur (Ohio)	us–east-2	inspector-scan.us-east-2.amazonaws.com inspector-scan-fips.us-east-2.amazonaws.com	HTTPS HTTPS
AS Timur (Virginia Utara)	us-east-1	inspector-scan.us-east-1.amazonaws.com inspector-scan-fips.us-east-1.amazonaws.com	HTTPS HTTPS

Nama Wilayah	Wilayah	Titik Akhir	Protokol
AS Barat (California Utara)	us-west-1	inspector-scan.us-west-1.amazonaws.com inspector-scan-fips.us-west-1.amazonaws.com	HTTPS HTTPS
AS Barat (Oregon)	us-west-2	inspector-scan.us-west-2.amazonaws.com inspector-scan-fips.us-west-2.amazonaws.com	HTTPS HTTPS
Africa (Cape Town)	af-south-1	inspector-scan.af-south-1.amazonaws.com	HTTPS
Asia Pasifik (Hong Kong)	ap-east-1	inspector-scan.ap-east-1.amazonaws.com	HTTPS
Asia Pasifik (Hyderabad)	ap-south-2	inspector-scan.ap-south-2.amazonaws.com	HTTPS
Asia Pasifik (Jakarta)	ap-southeast-3	inspector-scan.ap-southeast-3.amazonaws.com	HTTPS
Asia Pasifik (Malaysia)	ap-southeast-5	inspector-scan.ap-southeast-5.amazonaws.com	HTTPS
Asia Pasifik (Melbourne)	ap-southeast-4	inspector-scan.ap-southeast-4.amazonaws.com	HTTPS

Nama Wilayah	Wilayah	Titik Akhir	Protokol	
Asia Pasifik (Mumbai)	ap-south-1	inspector-scan.ap-south-1.amazonaws.com	HTTPS	
Asia Pasifik (Osaka)	ap-northeast-3	inspector-scan.ap-northeast-3.amazonaws.com	HTTPS	
Asia Pasifik (Seoul)	ap-northeast-2	inspector-scan.ap-northeast-2.amazonaws.com	HTTPS	
Asia Pasifik (Singapura)	ap-southeast-1	inspector-scan.ap-southeast-1.amazonaws.com	HTTPS	
Asia Pasifik (Sydney)	ap-southeast-2	inspector-scan.ap-southeast-2.amazonaws.com	HTTPS	
Asia Pasifik (Thailand)	ap-tenggara-7	inspector-scan.ap-southeast-7.amazonaws.com	HTTPS	
Asia Pacific (Tokyo)	ap-northeast-1	inspector-scan.ap-northeast-1.amazonaws.com	HTTPS	
Kanada (Pusat)	ca-central-1	inspector-scan.ca-central-1.amazonaws.com	HTTPS	
Kanada Barat (Calgary)	ca-west-1	inspector-scan.ca-west-1.amazonaws.com	HTTPS	

Nama Wilayah	Wilayah	Titik Akhir	Protokol	
Eropa (Frankfurt)	eu-central-1	inspector-scan.eu-central-1.amazonaws.com	HTTPS	
Eropa (Irlandia)	eu-west-1	inspector-scan.eu-west-1.amazonaws.com	HTTPS	
Eropa (London)	eu-west-2	inspector-scan.eu-west-2.amazonaws.com	HTTPS	
Eropa (Milan)	eu-south-1	inspector-scan.eu-south-1.amazonaws.com	HTTPS	
Eropa (Paris)	eu-west-3	inspector-scan.eu-west-3.amazonaws.com	HTTPS	
Eropa (Spanyol)	eu-south-2	inspector-scan.eu-south-2.amazonaws.com	HTTPS	
Eropa (Stockholm)	eu-north-1	inspector-scan.eu-north-1.amazonaws.com	HTTPS	
Eropa (Zürich)	eu-central-2	inspector-scan.eu-central-2.amazonaws.com	HTTPS	
Israel (Tel Aviv)	il-central-1	inspector-scan.il-central-1.amazonaws.com	HTTPS	
Meksiko (Tengah)	mx-pusat-1	inspector-scan.mx-central-1.amazonaws.com	HTTPS	
Timur Tengah (Bahrain)	me-south-1	inspector-scan.me-south-1.amazonaws.com	HTTPS	

Nama Wilayah	Wilayah	Titik Akhir	Protokol
Timur Tengah (UAE)	me-central-1	inspector-scan.me-central-1.amazonaws.com	HTTPS
Amerika Selatan (Sao Paulo)	sa-east-1	inspector-scan.sa-east-1.amazonaws.com	HTTPS
AWS GovCloud (AS-Timur)	us-gov-east-1	inspector-scan.us-gov-east-1.amazonaws.com inspector-scan-fips.us-gov-east-1.amazonaws.com	HTTPS HTTPS
AWS GovCloud (AS-Barat)	us-gov-west-1	inspector-scan.us-gov-west-1.amazonaws.com inspector-scan-fips.us-gov-west-1.amazonaws.com	HTTPS HTTPS

Ketersediaan fitur khusus wilayah

Bagian ini menjelaskan ketersediaan fitur Amazon Inspector oleh. AWS Region

Pemindaian EC2 tanpa agen untuk Wilayah Amazon EC2

Tabel berikut menunjukkan Wilayah AWS tempat pemindaian tanpa agen untuk Amazon EC2 saat ini tersedia.

Nama wilayah	Kode Wilayah
US East (Northern Virginia)	us-east-1
US East (Ohio)	us-east-2
AS Barat (California Utara)	us-west-1

Nama wilayah	Kode Wilayah
US West (Oregon)	as-barat-2
Afrika (Cape Town)	af-selatan-1
Asia Pasifik (Hong Kong)	ap-east-1
Asia Pasifik (Tokyo)	ap-northeast-1
Asia Pasifik (Seoul)	ap-northeast-2
Asia Pacific (Osaka)	ap-northeast-3
Asia Pasifik (Mumbai)	ap-south-1
Asia Pasifik (Hyderabad)	ap-south-2
Asia Pasifik (Singapura)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pasifik (Jakarta)	ap-southeast-3
Asia Pacific (Melbourne)	ap-southeast-4
Asia Pasifik (Malaysia)	ap-southeast-5
Asia Pasifik (Thailand)	ap-tenggara 7
Kanada (Pusat)	ca-central-1
Kanada Barat (Calgary)	ca-west-1
Eropa (Stockholm)	eu-north-1
Eropa (Frankfurt)	eu-central-1
Europe (Zurich)	eu-central-2
Eropa (Irlandia)	eu-west-1

Nama wilayah	Kode Wilayah
Europe (London)	eu-west-2
Europe (Paris)	eu-west-3
Europe (Milan)	eu-south-1
Eropa (Spanyol)	eu-south-2
Israel (Tel Aviv)	il-central-1
Timur Tengah (UAE)	me-central-1
Timur Tengah (Bahrain)	me-south-1
Meksiko (Tengah)	mx-pusat-1
Amerika Selatan (Sao Paulo)	sa-east-1
AWS GovCloud (AS-Timur)	us-gov-east-1
AWS GovCloud (AS-Barat)	us-gov-west-1

Wilayah pemindaian kode Lambda

Tabel berikut menunjukkan Wilayah AWS di mana pemindaian kode Lambda saat ini tersedia.

Nama wilayah	Kode Wilayah
US East (Northern Virginia)	us-east-1
AS Barat (Oregon)	us-west-2
AS Timur (Ohio)	us-east-2
Asia Pasifik (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1
Eropa (Frankfurt)	eu-central-1

Nama wilayah	Kode Wilayah
Europe (Ireland)	eu-west-1
Europe (London)	eu-west-2
Eropa (Stockholm)	eu-north-1
Asia Pasifik (Singapura)	ap-southeast-1

Important

Jika Anda mencoba mengaktifkan pemindaian kode Lambda dengan Amazon Inspector Enable API AWS Region di mana pemindaian kode Lambda tidak tersedia, Anda menerima kesalahan akses ditolak berikut:

```
An error occurred (AccessDeniedException) when calling the Enable operation:
 Lambda code scanning is not supported in unsupported-AWS Region
```

Wilayah Keamanan Kode Amazon Inspector

Tabel berikut menunjukkan Wilayah AWS tempat Amazon Inspector Code Security saat ini tersedia.

Nama wilayah	Kode Wilayah
US East (Northern Virginia)	us-east-1
AS Barat (Oregon)	us-west-2
AS Timur (Ohio)	us-east-2
Asia Pasifik (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1
Eropa (Frankfurt)	eu-central-1

Nama wilayah	Kode Wilayah
Europe (Ireland)	eu-west-1
Europe (London)	eu-west-2
Eropa (Stockholm)	eu-north-1
Asia Pasifik (Singapura)	ap-southeast-1

AWS GovCloud (US) Daerah

Untuk informasi terbaru, lihat Amazon Inspector di AWS GovCloud (US) Panduan Pengguna.

Riwayat dokumen

Tabel berikut menjelaskan perubahan penting dalam setiap rilis Panduan Pengguna Amazon Inspector, mulai November 2021. Untuk menerima pemberitahuan tentang pembaruan dokumentasi, Anda dapat berlangganan umpan RSS.

Pembaruan Produk Amazon Inspector

Perubahan	Deskripsi	Tanggal
Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector mengetahui skenario di mana Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk CVE-2026-25679, CVE-2026-27142, dan CVE-2026-27139. Dikonfirmasi bahwa Amazon Inspector SBOM Generator tidak terpengaruh oleh kerentanan ini. Kerentanan ini dapat diatasi dengan memutakhirkan versi Amazon Inspector SBOM Generator ke 1.11.2 atau yang lebih baru.	Maret 11, 2026
Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector mengetahui skenario di mana Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk. `CVE-2025-15558` Telah dikonfirmasi bahwa Amazon Inspector SBOM Generator tidak terpengaruh oleh CVE-2025-15558. Kerentanan ini dapat	Maret 5, 2026

	diatasi dengan memutakhirkan versi Amazon Inspector SBOM Generator ke 1.11.1 atau yang lebih baru.	
Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector mengetahui skenario di mana Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk `CVE-2025-68121` Telah dikonfirmasi bahwa Amazon Inspector SBOM Generator tidak terpengaruh oleh CVE-2025-68121. Kerentanan ini dapat diatasi dengan memutakhirkan versi Amazon Inspector SBOM Generator ke 1.11.0 atau yang lebih baru.	Maret 2, 2026
Kebijakan terkelola baru	Amazon Inspector telah merilis kebijakan terkelola baru `AmazonInspector2ManagedTelemetryPolicy` yang memberikan izin untuk operasi telemetri Amazon Inspector, yang memungkinkan layanan mengumpulkan dan mengirimkan data inventaris paket untuk pemindaian kerentanan. Untuk selengkapnya, lihat Amazon Inspector memperbarui kebijakan AWS terkelola.	Februari 5, 2026

Kebijakan yang diperbarui	Amazon Inspector menambahkan izin baru ke peran terkait layanan bernama. `AmazonInspector2ServiceRolePolicy` Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector untuk menjelaskan metadata firewall untuk analisis jangkauan jaringan. Selain itu, Amazon Inspector telah menambahkan pelingkupan sumber daya tambahan untuk memungkinkan Amazon Inspector membuat, memperbarui, dan memulai asosiasi SSM dengan dokumen SSM. `AWS-ConfigureAWSPackage` Untuk informasi selengkapnya, lihat Izin peran terkait layanan untuk Amazon Inspector.	Februari 3, 2026

Pembaruan untuk plugin Amazon Inspector SSM dan Amazon Inspector SBOM Generator	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM dan Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk. `CVE-2025-61728`, `CVE-2025-61730`, and `CVE-2025-61726` Kerentanan ini dapat diatasi dengan memutakhirkan versi plugin Amazon Inspector SSM ke 1.0.2327.0, atau Amazon Inspector SBOM Generator 1.10.1 atau yang lebih baru.	Januari 29, 2026
Pembaruan untuk plugin Amazon Inspector SSM dan Amazon Inspector SBOM Generator	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM dan Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk. `CVE-2025-61729` Dikonfirmasi bahwa aplikasi ini tidak terpengaruh oleh CVE ini. Saat ini kami sedang mengerjakan perbaikan untuk menyelesaikan deteksi ini. Sementara itu, pelanggan dapat dengan aman mengabaikan atau menekan kerentanan ini.	Desember 3, 2025

Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector mengetahui skenario di mana Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk dan. `CVE-2025-47914` `CVE-2025-58181` Telah dikonfirmasi Amazon Inspector SBOM Generator tidak terpengaruh oleh ini. CVEs Saat ini kami sedang mengerjakan perbaikan untuk menyelesaikan deteksi ini. Sementara itu, pelanggan dapat dengan aman mengabaikan atau menekan kerentanan ini.	November 20, 2025
Fitur baru	Amazon Inspector sekarang mendukung AWS Organizations kebijakan untuk pemberdayaan dan tata kelola terpusat di seluruh akun organisasi. Kebijakan organisasi memungkinkan Anda mengaktifkan jenis pemindaian Amazon Inspector secara otomatis di seluruh organisasi Anda dan mencegah modifikasi yang tidak sah. Untuk informasi selengkapnya, lihat Memulai tutorial dan Mengelola beberapa akun.	November 19, 2025

Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector dibuat sadar akan skenario di mana Amazon Inspector SBOM Generator dapat menghasil kan temuan kerentanan untuk. `CVE-2025-47913` Telah dikonfirmasi bahwa Amazon Inspector SBOM Generator tidak terpengaruh oleh CVE ini, dan pembaruan telah diterapkan untuk menyelesa ikan deteksi ini.	November 14, 2025
Kebijakan yang diperbarui	Amazon Inspector menambahkan izin baru ke kebijakan terkelola dan. `AmazonInspector2Fu llAccess_v2` `AmazonIns pector2ReadOnlyAcc ess` Izin memungkinkan melihat kebijakan organisas i Amazon Inspector dan konfigurasi delegasi yang dibuat melalui kebijakan. AWS Organizations Untuk informasi selengkapnya, lihat kebijakan AWS terkelola untuk Amazon Inspector.	November 14, 2025
Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector memperbar ui versi Amazon Inspector SBOM Generator. Untuk informasi selengkapnya, lihat Amazon Inspector SBOM Generator versi sebelumnya.	November 11, 2025

Kebijakan yang diperbarui	Amazon Inspector menambahkan izin baru ke peran terkait layanan bernama. `AmazonInspector2ServiceRolePolicy` Izin memungkinkan kebijakan Amazon AWS Organizations Inspector untuk menegakkan pemberdayaan dan penonaktifan Amazon Inspector. Untuk informasi selengkapnya, lihat Izin peran terkait layanan untuk Amazon Inspector.	November 10, 2025
Pembaruan untuk Amazon Inspector SBOM Generator	Amazon Inspector dibuat sadar akan skenario di mana Amazon Inspector SBOM Generator dapat menghasilkan temuan kerentanan untuk dan. `CVE-2025-58188` `CVE-2025-61725` Telah dikonfirmasi bahwa Amazon Inspector SBOM Generator tidak terpengaruh oleh ini, dan Amazon Inspector memperbarui CVEs Amazon Inspector SBOM Generator versi. Untuk informasi selengkapnya, lihat Amazon Inspector SBOM Generator versi sebelumnya.	November 4, 2025

Update untuk plugin	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM dapat menghasilkan temuan kerentanan untuk dan. `CVE-2025-58188` `CVE-2025-61725` Telah dikonfirmasi bahwa plugin Amazon Inspector SSM tidak terpengaruh oleh ini CVEs, dan pembaruan telah digunakan untuk menyelesaikan deteksi ini.	November 3, 2025
Update untuk plugin	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM mungkin menghasilkan temuan kerentanan. `CVE-2025-47907` Telah dikonfirmasi bahwa plugin Amazon Inspector SSM tidak terpengaruh oleh ini CVEs, dan pembaruan telah digunakan untuk menyelesaikan deteksi ini.	Agustus 8, 2025
Kebijakan baru	Amazon Inspector menambahkan kebijakan terkelola baru yang menyediakan akses penuh ke Amazon Inspector dan akses ke layanan terkait lainnya. Untuk informasi selengkapnya, lihat kebijakan AWS terkelola untuk Amazon Inspector.	Juli 3, 2025

Fungsionalitas diperbarui	Amazon Inspector sekarang tersedia dalam versi baru. Wilayah AWS Untuk informasi selengkapnya, lihat Wilayah dan titik akhir.	Juli 1, 2025
Fungsionalitas diperbarui	Amazon Inspector memperbarui periode retensi untuk temuan tertutup. Amazon Inspector menghapus temuan setelah 3 hari jika sumber daya terkait dihapus, dihentikan, atau tidak lagi layak untuk pemindaian. Untuk informasi selengkapnya, lihat Memahami temuan Amazon Inspector.	Juni 25, 2025
Fungsionalitas diperbarui	Amazon Inspector memperbarui sistem operasi yang didukung untuk pemindaian Amazon EC2 dan pemindaian Amazon ECR. Pemindaian Amazon EC2 sekarang mendukung Fedora versi 42 dan Ubuntu versi 25.04. Pemindaian Amazon ECR sekarang mendukung Alpine versi 3.22, Fedora versi 42, dan Ubuntu versi 25.04. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Juni 18, 2025

Fitur baru	Amazon Inspector sekarang memindai kode sumber aplikasi pihak pertama, dependensi aplikasi pihak ketiga, dan Infrastruktur sebagai Kode untuk kerentanan. Untuk informasi selengkapnya, lihat Amazon Inspector Code Security.	Juni 17, 2025
Update untuk plugin	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM dapat menghasilkan temuan kerentanan untuk dan. `CVE-2025-0913` `CVE-2025-4673` Telah dikonfirmasi bahwa plugin Amazon Inspector SSM tidak terpengaruh oleh ini CVEs, dan pembaruan telah digunakan untuk menyelesaikan deteksi ini.	Juni 13, 2025
Fitur baru	Amazon Inspector sekarang dapat menampilkan gambar kontainer yang digunakan secara aktif dan kapan gambar kontainer terakhir digunakan pada sebuah cluster. Untuk informasi selengkapnya, lihat Memetakan gambar kontainer ke container yang sedang berjalan.	16 Mei 2025

Pembaruan untuk sistem operasi yang didukung	Amazon Inspector menambahkan dukungan untuk BusyBox Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	13 Mei 2025
Kebijakan yang diperbarui	Amazon Inspector menambahkan izin baru ke peran terkait layanan bernama. `AmazonInspector2ServiceRolePolicy` Izin ini memungkinkan Anda untuk menggambarkan alamat IP dan gateway internet. Untuk informasi selengkapnya, lihat kebijakan AWS terkelola untuk Amazon Inspector.	April 29, 2025
Update untuk plugin	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM mungkin menghasilkan temuan kerentanan. `CVE-2025-22871` Telah dikonfirmasi bahwa plugin Amazon Inspector SSM tidak terpengaruh oleh ini CVEs, dan pembaruan telah digunakan untuk menyelesaikan deteksi ini.	April 21, 2025

Update untuk plugin	Amazon Inspector mengetahui skenario di mana plugin Amazon Inspector SSM dapat menghasilkan temuan kerentanan untuk,, dan. `CVE-2020-8911` `CVE-2020-8912` `CVE-2024-45337` Telah dikonfirmasi bahwa Amazon Inspector tidak terpengaruh oleh ini CVEs dan pembaruan telah diterapkan untuk menyelesaikan deteksi ini.	April 18, 2025
Pembaruan untuk Amazon Inspector SBOM Generator chapter	Amazon Inspector memperbarui versi Amazon Inspector SBOM Generator. Untuk informasi selengkapnya, lihat Amazon Inspector SBOM Generator versi sebelumnya.	April 16, 2025
Pembaruan untuk Amazon Inspector SBOM Generator chapter	Amazon Inspector menambahkan topik baru ke Amazon Inspector SBOM Generator chapter. Topik ini menjelaskan bagaimana Sbomgen melacak informasi lisensi dalam tagihan materi perangkat lunak. Untuk informasi selengkapnya, lihat koleksi lisensi Amazon Inspector SBOM Generator.	April 16, 2025

Pembaruan kebijakan terkelola	Amazon Inspector menambahkan izin yang memungkinkan akses hanya-baca ke Amazon ECS dan tindakan Amazon EKS. Untuk informasi selengkapnya, lihat Izin peran terkait layanan untuk Amazon Inspector.	25 Maret 2025
Pembaruan untuk sistem operasi yang didukung	Amazon Inspector tidak lagi mendukung SUSE Linux Enterprise Server 12.5 sebagai bagian dari pemindaian untuk Amazon EC2 dan Amazon ECR. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Maret 21, 2025
Pembaruan untuk sistem operasi yang didukung	Amazon Inspector menambahkan dukungan untuk Chainguard dan ke pemindaian Wolfi Amazon ECR. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Maret 21, 2025
Pembaruan untuk daftar isi	Amazon Inspector menambahkan bagian tentang penandaan sumber daya Amazon Inspector. Untuk informasi selengkapnya, lihat Menandai sumber daya Amazon Inspector.	Februari 25, 2025

Pembaruan untuk daftar isi	Amazon Inspector menambahkan topik baru ke Amazon Inspector SBOM Generator chapter. Untuk informasi selengkapnya, lihat koleksi sistem operasi komprehensif Amazon Inspector SBOM Generator.	Januari 28, 2025
Fungsionalitas diperbarui	Amazon Inspector menambahkan nodejs202 .x dan python3.13 ke daftar runtime yang didukung untuk pemindaian standar Lambda. Untuk informasi selengkap nya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Januari 24, 2025
Fungsionalitas diperbarui	Amazon Inspector menghapus Oracle Linux (Oracle) 7 dan SUSE Linux Enterpris e Server (SLES) 15.5 dari daftar sistem operasi yang didukung untuk Amazon EC2 dan Amazon ECR. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Desember 31, 2024

Fungsionalitas diperbarui	Amazon Inspector menambahkan Ubuntu 24.10 ke daftar sistem operasi yang didukung untuk Amazon EC2 dan Amazon ECR. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Desember 12, 2024
Pembaruan untuk daftar isi	Amazon Inspector menambahkan topik baru ke bagian Amazon Inspector SBOM Generator. Untuk informasi selengkapnya, lihat Amazon Inspector SBOM Generator.	Desember 9, 2024
Fungsionalitas diperbarui	Amazon Inspector memperbarui `amazon:inspector:sbom_generator` tabel untuk menambah dan menghapus ruang nama. Untuk informasi selengkapnya, lihat Menggunakan ruang nama CycloneDX dengan Amazon Inspector.	Desember 9, 2024
Fungsionalitas diperbarui	Amazon Inspector memperbarui fitur integrasi CI/CD untuk mendukung tindakan pemindaian dengan. CodePipeline Untuk informasi selengkapnya, lihat Menggunakan tindakan Amazon Inspector Scan dengan. CodePipeline	November 26, 2024

Pembaruan untuk daftar isi	Amazon Inspector mengatur ulang daftar isi untuk menyertakan chapter untuk Amazon Inspector SBOM Generator. Untuk informasi selengkapnya, lihat Amazon Inspector SBOM Generator.	November 22, 2024
Fungsionalitas diperbarui	Amazon Inspector menghapus Fedora 39 dari daftar sistem operasi yang didukung untuk Amazon EC2 dan Amazon ECR. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	November 22, 2024
Fungsionalitas diperbarui	Amazon Inspector menghapus Alpine 3.17 dari daftar sistem operasi yang didukung untuk Amazon ECR. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	November 22, 2024
Fungsionalitas diperbarui	Amazon Inspector menambahkan Sbomgen versi ke versi sebelumnya dari Amazon Inspector SBOM Generator.	November 19, 2024

Fungsionalitas diperbarui	Amazon Inspector menambahkan AL2 sebagai runtime yang didukung. Untuk informasi selengkapnya, lihat Sistem operasi yang didukung dan bahasa pemrograman untuk Amazon Inspector.	Agustus 26, 2024
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ServiceRolePolicy Pernyataan baru memungkinkan Amazon Inspector untuk mengembalikan tag fungsi di. AWS Lambda	Juli 31, 2024
Fungsionalitas diperbarui	Amazon Inspector merilis kontrol keamanan baru. Untuk informasi selengkapnya, lihat kontrol Amazon Inspector di AWS Security Hub CSPM Panduan Pengguna.	Juli 11, 2024
Fungsionalitas diperbarui	Amazon Inspector SBOM Generator sekarang memindai gambar kontainer Dockerfiles dan Docker untuk kesalahan konfigurasi yang dapat menimbulkan kerentanan keamanan. Untuk informasi selengkapnya, lihat pemeriksaan Amazon Inspector Dockerfile.	Juni 10, 2024

Fungsionalitas diperbarui	Amazon Inspector memperbarui fitur integrasi CI/CD untuk mendukung CodeCatalyst tindakan, sehingga Anda dapat menambahkan pemindaian kerentanan Amazon Inspector ke alur kerja Anda. CodeCatalyst Untuk informasi selengkapnya, lihat Menggunakan CodeCatalyst tindakan.	Juni 7, 2024
Fungsionalitas diperbarui	Amazon Inspector menyertakan opsi untuk mengunduh file CSV hasil pemindaian CIS. Untuk informasi selengkapnya, lihat Melihat dan mengunduh hasil pemindaian CIS di pemindaian Pusat Keamanan Internet (CIS) untuk instans Amazon EC2.	3 Mei 2024
Fungsionalitas diperbarui	Amazon Inspector memperbarui fitur integrasi CI/CD untuk mendukungGitHub Actions, sehingga Anda dapat menambahkan pemindaian kerentanan Amazon Inspector ke alur kerja Anda. GitHub Untuk informasi selengkapnya, lihat Menggunakan Amazon Inspector dengan. GitHub Actions	April 29, 2024

Fungsionalitas diperbarui	Amazon Inspector memperbarui kebijakan terkelola `AmazonInspector2FullAccess`, sehingga membuat peran terkait layanan. `AWSServiceRoleForAmazonInspector2Agentless` Hal ini memungkinkan pengguna untuk melakukan pemindaian berbasis agen dan pemindaian tanpa agen saat mereka mengaktifkan Amazon Inspector.	April 24, 2024
Fungsionalitas diperbarui	Amazon Inspector memperbarui periode retensi untuk temuan tertutup dari 30 hari hingga 7 hari. Untuk informasi selengkapnya, lihat Memahami temuan di Amazon Inspector.	Februari 12, 2024
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ServiceRolePolicy Pernyataan baru ini memungkinkan Amazon Inspector untuk memulai pemindaian CIS untuk instans Anda.	23 Januari 2024

Kebijakan Baru	Amazon Inspector telah menambahkan kebijakan baru, AmazonInspector2ManagedCisPolicykebijakan, yang dapat Anda gunakan sebagai bagian dari dalam profil instans untuk mengizinkan pemindaian CIS pada sebuah instans.	23 Januari 2024
Fitur Baru	Amazon Inspector sekarang akan menyegarkan durasi pemindaian ulang ECR dari gambar kontainer saat Anda menariknya. Untuk mengubah durasi pemindaian ulang berdasarkan tanggal push atau pull, lihat Mengonfigurasi durasi pemindaian ulang ECR.	23 Januari 2024
Fitur Baru	Amazon Inspector sekarang dapat menjalankan pemindaian Center for Internet Security (CIS) pada instans EC2. Untuk informasi selengkapnya, lihat pemindaian Amazon Inspector CIS.	23 Januari 2024
Fitur Baru	Amazon Inspector sekarang dapat memindai gambar kontainer di pipeline Anda CI/CD . Untuk informasi selengkapnya, lihat Integrasi CI/CD dengan Amazon Inspector.	30 November 2023

Kebijakan Baru	Amazon Inspector telah menambahkan kebijakan baru yang memungkinkan Amazon Inspector memindai snapshot Amazon EBS dari instans EC2 Anda untuk pemindaian tanpa agen. Untuk informasi selengkap nya tentang kebijakan ini, lihat Pemindaian tanpa agen.	27 November 2023
Fitur Baru	Amazon Inspector sekarang mendukung pemindaian instans Amazon EC2 Linux yang didukung tanpa agen SSM melalui pemindaian tanpa agen. Untuk informasi lebih lanjut, lihat Pemindaian tanpa agen.	27 November 2023
Sumber daya baru yang didukung	Amazon Inspector sekarang mendukung pemindaian instans macOS Amazon EC2. Lihat Sistem operasi yang didukung: Pemindaia n Amazon EC2 untuk versi macOS yang didukung.	5 Oktober 2023
Daerah Baru	Amazon Inspector sekarang tersedia di Asia Pasifik (Jakarta), Afrika (Cape Town), Asia Pasifik (Osaka), dan Eropa (Zurich).	September 29, 2023

Fitur baru	Anda sekarang dapat mengecualikan instans EC2 dari pemindaian Amazon Inspector menggunakan tag pengecualian.	14 September 2023
Fitur baru	Amazon Inspector telah menambahkan izin baru yang memungkinkan Amazon Inspector memindai konfigurasi jaringan instans Amazon EC2 yang merupakan bagian dari grup target Elastic Load Balancing.	31 Agustus 2023
Fitur baru	Amazon Inspector sekarang memberikan rincian intelijen kerentanan untuk temuan kerentanan paket.	31 Juli 2023
Fungsionalitas diperbarui	Amazon Inspector telah menambahkan izin baru yang memungkinkan pengguna read-only untuk mengekspor Software Bill of Materials (SBOM) untuk sumber daya mereka.	29 Juni 2023
Fitur baru	Anda sekarang dapat mengekspor SBOM untuk sumber daya yang dipindai oleh Amazon Inspector.	13 Juni 2023

Fitur baru	Pemindaian kode Lambda sekarang tersedia secara umum. Fitur baru telah ditambahkan yang memungkinkan Anda mengenkripsi kode yang diidentifikasi dalam temuan pemindaian kode Lambda Anda. Selain itu, pemindaian kode Lambda sekarang menyediakan penulisan ulang remediasi yang disarankan untuk kode Anda.	13 Juni 2023
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ReadOnlyAccess Pernyataan baru ini memungkinkan pengguna hanya-baca untuk mengambil detail status pemindaian kode Lambda dan temuan untuk akun mereka.	2 Mei 2023
Fitur baru	Amazon Inspector telah menambahkan pencarian database Vulnerability yang memungkinkan Anda memeriksa apakah Amazon Inspector mencakup CVE tertentu.	1 Mei 2023

Fungsionalitas diperbarui	Amazon Inspector telah menambahkan izin baru ke AmazonInspector2ServiceRolePolicykebijakan yang memungkinkan Amazon Inspector membuat saluran AWS CloudTrail terkait layanan di akun Anda saat Anda mengaktifkan pemindaian Lambda. Hal ini memungkinkan Amazon Inspector untuk memantau CloudTrail peristiwa di akun Anda.	April 30, 2023
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2FullAccess Pernyataan baru ini memungkinkan pengguna untuk mengambil rincian temuan kerentanan kode dari pemindaian kode Lambda.	17 April 2023
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ServiceRolePolicy Pernyataan baru ini memungkinkan Amazon Inspector untuk mengirim informasi ke Amazon EC2 Systems Manager tentang jalur kustom yang telah Anda tentukan untuk inspeksi mendalam Amazon EC2.	17 April 2023

Fitur baru	Amazon Inspector menambahkan dukungan tambahan untuk instans Linux EC2 dalam bentuk inspeksi mendalam Amazon Inspector , yang memindai instans Anda untuk kerentanan paket dalam paket bahasa pemrograman aplikasi.	17 April 2023
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ServiceRole Policy Pernyataan baru memungkinkan Amazon Inspector untuk meminta pemindaian kode pengembang dalam AWS Lambda fungsi, dan menerima data pemindaian dari Amazon Security. CodeGuru Selain itu Amazon Inspector telah menambahkan izin untuk meninjau kebijakan IAM. Amazon Inspector menggunakan informasi ini untuk memindai fungsi Lambda untuk kerentanan kode.	28 Februari 2023

Fitur baru	Amazon Inspector menambahkan dukungan tambahan untuk fungsi Lambda dalam bentuk pemindaian kode Lambda, yang memindai kode pengembang fungsi Lambda Anda untuk kerentanan keamanan.	28 Februari 2023
Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ServiceRole Policy Pernyataan baru ini memungkinkan Amazon Inspector untuk mengambil informasi dari CloudWatch tentang kapan AWS Lambda fungsi terakhir digunakan . menggunakan informasi ini untuk memfokuskan pemindaian pada fungsi Lambda di lingkungan Anda yang telah aktif dalam 90 hari terakhir.	Februari 20, 2023

Fungsionalitas diperbarui	Amazon Inspector menambahkan pernyataan baru pada kebijakan tersebut. AmazonInspector2ServiceRolePolicy Pernyataan baru ini memungkinkan Amazon Inspector untuk mengambil informasi tentang fungsi Anda. AWS Lambda Amazon Inspector menggunakan informasi ini untuk memindai fungsi Lambda Anda untuk mencari kerentanan keamanan.	28 November 2022
Fitur baru	Amazon Inspector menambahkan dukungan untuk fungsi Scanning AWS Lambda.	28 November 2022
Konten yang diperbarui	Menambahkan prosedur, contoh kebijakan, dan tip untuk mengekspor laporan temuan dari Amazon Inspector ke bucket Amazon Simple Storage Service (Amazon S3).	14 Oktober 2022
Konten baru	Menambahkan informasi tentang menilai cakupan Amazon Inspector lingkungan AWS Anda dengan menggunakan konsol Amazon Inspector. Informasi tersebut mencakup deskripsi nilai Status untuk sumber daya individu di lingkungan Anda.	Oktober 7, 2022

Fitur baru	Amazon Inspector sekarang memberikan rincian tambahan tentang cara memulihkan kerentanan paket. Bidang baru telah ditambahkan untuk menemukan detail. Bidang baru menyediakan konteks tentang apakah perbaikan tersedia melalui pembaruan paket. Jika perbaikan tersedia, bagian Remediasi yang disarankan dari temuan menunjukkan perintah yang dapat Anda jalankan untuk melakukan perbaikan.	September 2, 2022
Fungsionalitas diperbarui	Amazon Inspector menambahkan tindakan baru ke kebijakan tersebut. AmazonInspector2ServiceRolePolicy Tindakan baru ini memungkinkan Amazon Inspector untuk menggambarkan eksekusi asosiasi SSM. Amazon Inspector juga menambahkan pelingkupan sumber daya tambahan untuk memungkinkan Amazon Inspector membuat, memperbarui, menghapus, dan memulai asosiasi SSM dengan dokumen SSM yang dimiliki. `AmazonInspector2`	31 Agustus 2022

Fitur baru	Amazon Inspector sekarang mendukung pemindaian untuk instance. Windows Amazon Inspector sekarang dapat memindai instans terkelola SSM yang menjalankan sistem operasi yang didukung. Windows Pemindaian Windows host dilakukan oleh plugin Amazon Inspector SSM, yang diinstal dan dipanggil melalui asosiasi SSM baru yang secara otomatis dibuat oleh Amazon Inspector.	31 Agustus 2022
Fungsionalitas diperbarui	Amazon Inspector memperbarui pelingkupan sumber daya AmazonInspector2ServiceRolePolicykebijakan untuk memungkinkan Amazon Inspector mengumpulkan inventaris perangkat lunak di partisi lain. AWS	12 Agustus 2022
Fungsionalitas diperbarui	Dalam AmazonInspector2ServiceRolePolicykebijakan tersebut, Amazon Inspector merestrukturisasi pelingkupan sumber daya dari tindakan yang memungkinkan Amazon Inspector membuat, menghapus, dan memperbarui asosiasi SSM.	Agustus 10, 2022

Fitur baru	Amazon Inspector sekarang mendukung perubahan pengaturan durasi pemindaian ulang otomatis ECR Anda. Pengaturan durasi pemindaian ulang otomatis Amazon ECR menentukan berapa lama Amazon Inspector terus memantau gambar yang didorong ke repositori. Ketika gambar lebih tua dari durasi pemindaian, Amazon Inspector tidak akan lagi memindai gambar dan menutup semua temuan yang ada untuknya. Semua akun baru akan secara otomatis memiliki durasi pemindaian ulang otomatis ECR yang disetel ke seumur hidup. Akun yang dibuat sebelumnya memiliki durasi pemindaian ulang otomatis ECR 30 hari, tetapi sekarang Anda dapat memilih dari 30 hari, 180 hari, atau durasi seumur hidup untuk pemindaian.	Juni 25, 2022
Fungsionalitas baru	Amazon Inspector menambahkan kebijakan AWS terkelola baru, AmazonInspector2ReadOnlyAccesskebijakan, untuk mengizinkan akses hanya-baca ke fungsionalitas Amazon Inspector.	Januari 21, 2022

Ketersediaan umum	Ini adalah rilis publik awal dari Panduan Pengguna Amazon Inspector.	29 November 2021

Riset Keamanan Amazon Inspector

Amazon Inspector terus memantau dan mengidentifikasi paket berbahaya dari registri NPM untuk melindungi aplikasi Anda dari serangan rantai pasokan.

Update terakhir: 2026-02-06 12:00:00 UTC

Ringkasan Deteksi

- Total Seumur Hidup: 191.801 paket berbahaya diidentifikasi
- Bulan Ini: 147 paket berbahaya baru diidentifikasi
- Bulan Lalu: 527 paket berbahaya baru diidentifikasi
- Minggu ini: 147 paket berbahaya baru diidentifikasi
- Minggu Lalu: 96 paket berbahaya baru diidentifikasi

Laporan Paket Berbahaya Terbaru (10 Terakhir)

Nama Package	MAL-ID	Tanggal Deteksi
web3-sinon	MAL-2026-807	2026-02-06
web3-rantai-sinon	MAL-2026-806	2026-02-06
larik sejajar	MAL-2026-805	2026-02-06
layanan remah roti	MAL-2026-804	2026-02-06
@sbseg -plugin/ qbo-web-app-ui	MAL-2026-802	2026-02-06
@rsgweb /utils	MAL-2026-801	2026-02-06
@rsgweb /tina	MAL-2026-800	2026-02-06

Nama Package	MAL-ID	Tanggal Deteksi
@rsgweb /akun rockstar	MAL-2026-799	2026-02-06
@rsgweb/modules-core-www-page	MAL-2026-798	2026-02-06
@rsgweb/modules-core-feedback	MAL-2026-797	2026-02-06

AWS Glosarium

Untuk AWS terminologi terbaru, lihat AWS glosarium di Referensi.Glosarium AWS

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.