Panduan Pengguna

AWS Akses Terverifikasi

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Akses Terverifikasi: Panduan Pengguna

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu Akses Terverifikasi AWS? 1
Manfaat Akses Terverifikasi 1
Mengakses Akses Terverifikasi 1
Harga 2
Cara kerja Akses Terverifikasi 3
Komponen utama dari Akses Terverifikasi 3
Mulai tutorial 6
Prasyarat 6
Buat penyedia kepercayaan 7
Buatlah sebuah instans 7
Membuat grup 8
Buat titik akhir 8
Konfigurasikan DNS untuk titik akhir 9
Uji konektivitas ke aplikasi 10
Menambahkan kebijakan akses 10
Bersihkan 11
Instans Akses Terverifikasi 12
Membuat dan mengelola instance Akses Terverifikasi 12
Buat instance Akses Terverifikasi 12
Lampirkan penyedia kepercayaan ke instans Akses Terverifikasi 13
Lepaskan penyedia kepercayaan dari instans Akses Terverifikasi 13
Tambahkan subdomain khusus 14
Menghapus instans Akses Terverifikasi 15
Integrasikan dengan AWS WAF 15
Izin IAM yang diperlukan 16
Kaitkan ACL AWS WAF web 16
Periksa status asosiasi 17
Putuskan hubungan ACL AWS WAF web 17
Kepatuhan FIPS 18
Lingkungan yang ada 19
Lingkungan baru 19
Penyedia kepercayaan 20
Identitas pengguna 20
Pusat Identitas IAM 20

Penyedia kepercayaan OIDC 22
Berbasis perangkat 26
Penyedia kepercayaan perangkat yang didukung 26
Buat penyedia kepercayaan berbasis perangkat 26
Memodifikasi penyedia kepercayaan berbasis perangkat 27
Menghapus penyedia kepercayaan berbasis perangkat 28
Grup Akses Terverifikasi 29
Membuat dan mengelola grup Akses Terverifikasi 29
Membuat grup Akses Terverifikasi 30
Memodifikasi grup Akses Terverifikasi 30
Ubah kebijakan grup Akses Terverifikasi 31
Bagikan grup dengan akun lain 31
Pertimbangan 32
Pembagian sumber daya 33
Menghapus grup Akses Terverifikasi 34
Titik akhir Akses Terverifikasi 35
Jenis titik akhir Akses Terverifikasi 35
Cara kerja Akses Terverifikasi dengan berbagi VPCs dan subnet 36
Buat titik akhir penyeimbang beban 36
Buat titik akhir antarmuka jaringan 38
Buat titik akhir CIDR jaringan 39
Membuat titik akhir Layanan Amazon Relational Database Service 40
Izinkan lalu lintas dari titik akhir Anda 42
Ubah titik akhir Akses Terverifikasi 43
Ubah kebijakan titik akhir Akses Terverifikasi 43
Menghapus titik akhir Akses Terverifikasi 44
Data kepercayaan Akses Terverifikasi 45
Konteks default 45
Permintaan HTTP 46
Aliran TCP 47
AWS IAM Identity Center konteks 48
Konteks pihak ketiga 50
Ekstensi browser 51
Jamf 51
CrowdStrike 53
JumpCloud 55

Klaim pengguna lewat ... 57
JWT untuk klaim pengguna OIDC ... 57
Klaim pengguna JWT untuk IAM Identity Center ... 58
Kunci publik ... 59
Mengambil dan mendekode JWT ... 60
Kebijakan Akses Terverifikasi ... 61
Pernyataan kebijakan ... 61
Komponen kebijakan ... 62
Komentar ... 62
Beberapa klausa ... 63
Karakter yang dipesan ... 63
Operator bawaan ... 63
Evaluasi kebijakan ... 66
Logika kebijakan korsleting ... 66
Contoh kebijakan ... 67
Berikan akses ke grup di Pusat Identitas IAM ... 67
Berikan akses ke grup di penyedia pihak ketiga ... 68
Berikan akses menggunakan CrowdStrike ... 68
Izinkan atau tolak alamat IP tertentu ... 69
Asisten kebijakan ... 69
Langkah 1: Tentukan sumber daya Anda ... 70
Langkah 2: Uji dan edit kebijakan ... 70
Langkah 3: Tinjau dan terapkan perubahan ... 71
Konektivitas Klien ... 72
Prasyarat ... 72
Unduh Klien Konektivitas ... 73
Ekspor file konfigurasi klien ... 73
Connect ke aplikasi ... 73
Copot pemasangan klien ... 74
Praktik terbaik ... 74
Pemecahan Masalah ... 75
Saat masuk, browser tidak terbuka untuk menyelesaikan otentikasi oleh iDP ... 75
Setelah otentikasi, status klien “tidak terhubung” ... 75
Tidak dapat terhubung menggunakan browser Chrome atau Edge ... 76
Riwayat versi ... 76
Keamanan ... 77

Perlindungan data 77
Enkripsi bergerak 79
Privasi lalu lintas antar jaringan 79
Enkripsi data saat istirahat 79
Manajemen identitas dan akses 94
Audiens 94
Mengautentikasi dengan identitas 95
Mengelola akses menggunakan kebijakan 99
Cara Kerja Akses Terverifikasi dengan IAM 101
Contoh kebijakan berbasis identitas 108
Pemecahan Masalah 112
Gunakan peran tertaut layanan 114
AWS kebijakan terkelola 116
Validasi kepatuhan 117
Ketahanan 119
Beberapa subnet untuk ketersediaan tinggi 119
Pemantauan 120
Log Akses Terverifikasi 120
Versi logging 121
Izin pencatatan 121
Mengaktifkan atau menonaktifkan log 122
Mengaktifkan atau menonaktifkan konteks kepercayaan 124
Contoh log OCSF versi 0.1 125
Contoh log OCSF versi 1.0.0-rc.2 137
CloudTrail log 145
Acara manajemen 146
Contoh acara 146
Kuota 149
Riwayat dokumen 151
..... cliii

Apa itu Akses Terverifikasi AWS?

Dengan Akses Terverifikasi AWS, Anda dapat memberikan akses aman ke aplikasi Anda tanpa memerlukan penggunaan jaringan pribadi virtual (VPN). Verified Access mengevaluasi setiap permintaan aplikasi dan membantu memastikan bahwa pengguna dapat mengakses setiap aplikasi hanya ketika mereka memenuhi persyaratan keamanan yang ditentukan.

Manfaat Akses Terverifikasi

- Postur keamanan yang ditingkatkan — Model keamanan tradisional mengevaluasi akses sekali dan memberi pengguna akses ke semua aplikasi. Verified Access mengevaluasi setiap permintaan akses aplikasi secara real time. Ini menyulitkan aktor jahat untuk berpindah dari satu aplikasi ke aplikasi lainnya.
- Integrasi dengan layanan keamanan — Akses Terverifikasi terintegrasi dengan layanan manajemen identitas dan perangkat, termasuk layanan pihak ketiga AWS dan layanan pihak ketiga. Menggunakan data dari layanan ini, Verified Access memverifikasi kepercayaan pengguna dan perangkat terhadap serangkaian persyaratan keamanan dan menentukan apakah pengguna harus memiliki akses ke aplikasi.
- Pengalaman pengguna yang ditingkatkan — Akses Terverifikasi menghilangkan kebutuhan pengguna untuk menggunakan VPN untuk mengakses aplikasi Anda. Ini membantu mengurangi jumlah kasus dukungan yang timbul dari masalah terkait VPN.
- Pemecahan masalah dan audit yang disederhanakan — Akses Terverifikasi mencatat semua upaya akses, memberikan visibilitas terpusat ke akses aplikasi, untuk membantu Anda merespons insiden keamanan dan permintaan audit dengan cepat.

Mengakses Akses Terverifikasi

Anda dapat menggunakan salah satu antarmuka berikut untuk bekerja dengan Akses Terverifikasi:

- AWS Management Console— Menyediakan antarmuka web yang dapat Anda gunakan untuk membuat dan mengelola sumber daya Akses Terverifikasi. Masuk ke AWS Management Console dan buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
- AWS Command Line Interface (AWS CLI) — Menyediakan perintah untuk serangkaian luas Layanan AWS, termasuk Akses Terverifikasi AWS. AWS CLI Ini didukung di Windows, macOS, dan Linux. Untuk mendapatkan AWS CLI, lihat AWS Command Line Interface.

- AWS SDKs— Menyediakan bahasa khusus APIs. AWS SDKs Mengurus banyak detail koneksi, seperti menghitung tanda tangan, dan menangani percobaan ulang permintaan dan kesalahan. Untuk informasi selengkapnya, lihat AWS SDKs.
- Query API - Menyediakan tindakan API tingkat rendah yang Anda panggil menggunakan permintaan HTTPS. Menggunakan Query API adalah cara paling langsung untuk mengakses Akses Terverifikasi. Namun, ini mengharuskan aplikasi Anda untuk menangani detail tingkat rendah seperti membuat hash untuk menandatangani permintaan dan menangani kesalahan. Untuk informasi selengkapnya, lihat Tindakan Akses Terverifikasi di Referensi Amazon EC2 API.

Panduan ini menjelaskan cara menggunakan sumber daya AWS Management Console untuk membuat, mengakses, dan mengelola sumber daya Akses Terverifikasi.

Harga

Anda dikenakan biaya per jam untuk setiap aplikasi pada Akses Terverifikasi, dan Anda dikenakan biaya untuk jumlah data yang diproses oleh Akses Terverifikasi. Untuk informasi lebih lanjut, lihat Harga Akses Terverifikasi AWS.

Cara kerja Akses Terverifikasi

Akses Terverifikasi AWS mengevaluasi setiap permintaan aplikasi dari pengguna Anda dan memungkinkan akses berdasarkan:

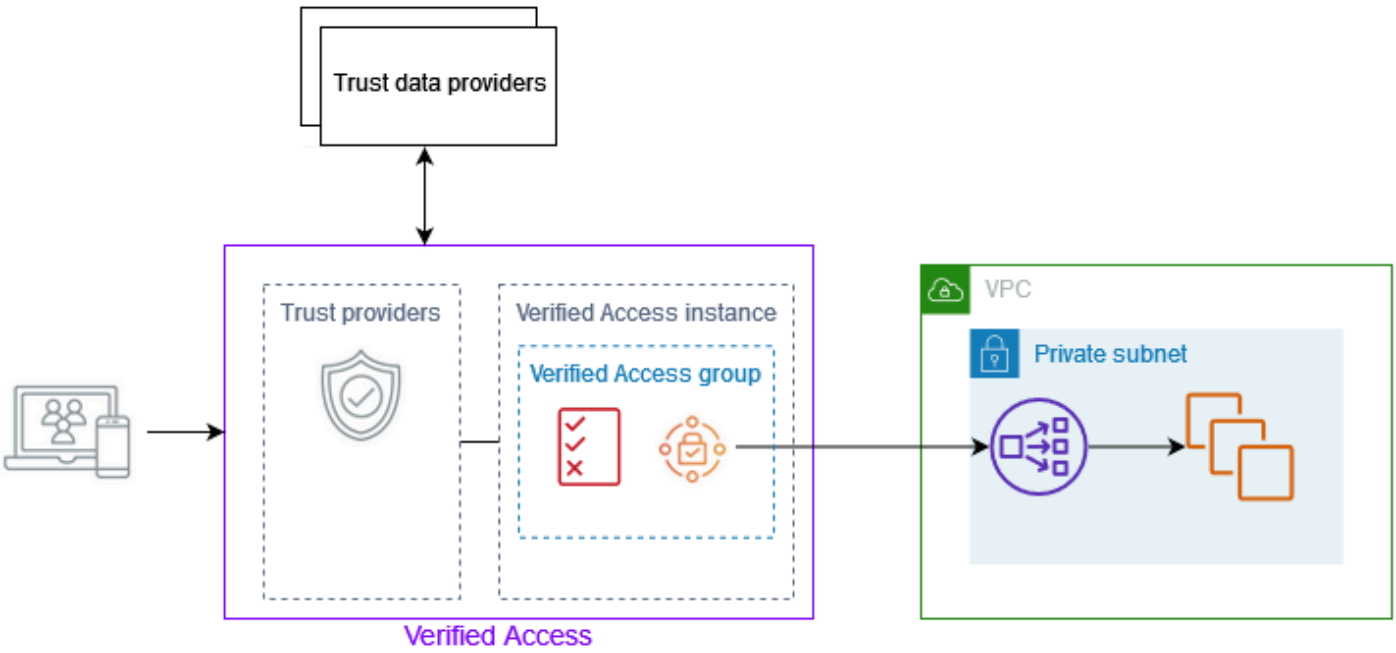
- Data kepercayaan yang dikirim oleh penyedia kepercayaan pilihan Anda (dari AWS atau pihak ketiga).
- Kebijakan akses yang Anda buat di Akses Terverifikasi.

Saat pengguna mencoba mengakses aplikasi, Verified Access mendapatkan datanya dari penyedia kepercayaan dan mengevaluasinya terhadap kebijakan yang Anda tetapkan untuk aplikasi tersebut. Akses Terverifikasi memberikan akses ke aplikasi yang diminta hanya jika pengguna memenuhi persyaratan keamanan yang Anda tentukan. Semua permintaan aplikasi ditolak secara default, hingga kebijakan ditentukan.

Selain itu, Akses Terverifikasi mencatat setiap upaya akses, untuk membantu Anda merespons insiden keamanan dan permintaan audit dengan cepat.

Komponen utama dari Akses Terverifikasi

Diagram berikut memberikan ikhtisar tingkat tinggi Akses Terverifikasi. Pengguna mengirim permintaan untuk mengakses aplikasi. Akses Terverifikasi mengevaluasi permintaan terhadap kebijakan akses untuk grup dan kebijakan titik akhir khusus aplikasi apa pun. Jika akses diizinkan, permintaan dikirim ke aplikasi melalui titik akhir.



- Instans Akses Terverifikasi — Instance mengevaluasi permintaan aplikasi dan memberikan akses hanya jika persyaratan keamanan Anda terpenuhi.
- Titik akhir Akses Terverifikasi - Setiap titik akhir mewakili aplikasi. Pada diagram di atas, aplikasi di-host pada EC2 instance yang merupakan target penyeimbang beban.
- Grup Akses Terverifikasi — Kumpulan titik akhir Akses Terverifikasi. Kami menyarankan Anda mengelompokkan titik akhir untuk aplikasi dengan persyaratan keamanan serupa untuk menyederhanakan administrasi kebijakan. Misalnya, Anda dapat mengelompokkan titik akhir untuk semua aplikasi penjualan Anda bersama-sama.
- Kebijakan akses — Seperangkat aturan yang ditentukan pengguna yang menentukan apakah akan mengizinkan atau menolak akses ke aplikasi. Anda dapat menentukan kombinasi faktor, termasuk identitas pengguna dan status keamanan perangkat. Anda membuat kebijakan akses grup untuk setiap grup Akses Terverifikasi, yang diwarisi oleh semua titik akhir dalam grup. Anda dapat secara opsional membuat kebijakan khusus aplikasi dan melampirkannya ke titik akhir tertentu.
- Penyedia kepercayaan — Layanan yang mengelola identitas pengguna atau status keamanan perangkat. Akses Terverifikasi berfungsi dengan penyedia kepercayaan pihak ketiga AWS dan penyedia kepercayaan pihak ketiga. Anda harus melampirkan setidaknya satu penyedia kepercayaan ke setiap instans Akses Terverifikasi. Anda dapat melampirkan satu penyedia kepercayaan identitas dan beberapa penyedia kepercayaan perangkat ke setiap instans Akses Terverifikasi.

- Data kepercayaan — Data terkait keamanan untuk pengguna atau perangkat yang dikirimkan oleh penyedia kepercayaan Anda ke Akses Terverifikasi. Juga disebut sebagai klaim pengguna atau konteks kepercayaan. Misalnya, alamat email pengguna atau versi sistem operasi perangkat. Akses Terverifikasi mengevaluasi data ini terhadap kebijakan akses Anda saat menerima setiap permintaan untuk mengakses aplikasi.

Tutorial: Memulai dengan Akses Terverifikasi

Gunakan tutorial ini untuk memulai Akses Terverifikasi AWS. Anda akan mempelajari cara membuat dan mengonfigurasi sumber daya Akses Terverifikasi.

Sebagai bagian dari tutorial ini, Anda akan menambahkan aplikasi ke Verified Access. Di akhir tutorial, pengguna tertentu dapat mengakses aplikasi itu melalui internet, tanpa menggunakan VPN. Sebagai gantinya, Anda akan menggunakannya AWS IAM Identity Center sebagai penyedia kepercayaan identitas. Perhatikan bahwa tutorial ini juga tidak menggunakan penyedia kepercayaan perangkat.

Tugas

- Prasyarat tutorial Akses Terverifikasi
- Langkah 1: Buat penyedia kepercayaan Akses Terverifikasi
- Langkah 2: Buat instance Akses Terverifikasi
- Langkah 3: Buat grup Akses Terverifikasi
- Langkah 4: Buat titik akhir Akses Terverifikasi
- Langkah 5: Konfigurasikan DNS untuk titik akhir Akses Terverifikasi
- Langkah 6: Uji konektivitas ke aplikasi
- Langkah 7: Tambahkan kebijakan akses tingkat grup Akses Terverifikasi
- Bersihkan sumber daya Akses Terverifikasi

Prasyarat tutorial Akses Terverifikasi

Berikut ini adalah prasyarat untuk menyelesaikan tutorial ini:

- AWS IAM Identity Center diaktifkan di Wilayah AWS tempat Anda bekerja. Anda kemudian dapat menggunakan IAM Identity Center sebagai penyedia kepercayaan dengan Akses Terverifikasi. Untuk informasi selengkapnya, lihat Mengaktifkan AWS IAM Identity Center di Panduan AWS IAM Identity Center Pengguna.
- Grup keamanan untuk mengontrol akses ke aplikasi. Izinkan semua lalu lintas masuk dari CIDR VPC dan semua lalu lintas keluar.
- Aplikasi yang berjalan di belakang penyeimbang beban internal dari Elastic Load Balancing. Kaitkan grup keamanan Anda dengan penyeimbang beban.

- Sertifikat TLS yang ditandatangani sendiri atau publik di. AWS Certificate Manager Gunakan sertifikat RSA dengan panjang kunci 1.024 atau 2.048.
- Domain yang dihosting publik dan izin yang diperlukan untuk memperbarui catatan DNS untuk domain tersebut.
- Kebijakan IAM dengan izin yang diperlukan untuk membuat instance Akses Terverifikasi AWS . Untuk informasi selengkapnya, lihat Kebijakan untuk membuat instance Akses Terverifikasi.

Langkah 1: Buat penyedia kepercayaan Akses Terverifikasi

Gunakan prosedur berikut untuk mengatur AWS IAM Identity Center sebagai penyedia kepercayaan Anda.

Untuk membuat penyedia kepercayaan Pusat Identitas IAM

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi.
3. Pilih Buat penyedia kepercayaan Akses Terverifikasi.
4. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk penyedia kepercayaan Akses Terverifikasi.
5. Masukkan pengenal kustom untuk digunakan nanti saat bekerja dengan aturan kebijakan untuk nama referensi Kebijakan. Misalnya, Anda bisa masuk**`idc`**.
6. Untuk jenis penyedia Trust, pilih Penyedia kepercayaan pengguna.
7. Untuk jenis penyedia kepercayaan pengguna, pilih Pusat Identitas IAM.
8. Pilih Buat penyedia kepercayaan Akses Terverifikasi.

Langkah 2: Buat instance Akses Terverifikasi

Gunakan prosedur berikut untuk membuat instance Akses Terverifikasi.

Untuk membuat instance Akses Terverifikasi

1. Di panel navigasi, pilih instans Akses Terverifikasi.
2. Pilih Buat instance Akses Terverifikasi.
3. (Opsional) Untuk Nama dan Deskripsi, masukkan nama dan deskripsi untuk instance Akses Terverifikasi.

4. Untuk penyedia kepercayaan Akses Terverifikasi, pilih penyedia kepercayaan Anda.
5. Pilih Buat instance Akses Terverifikasi.

Langkah 3: Buat grup Akses Terverifikasi

Gunakan prosedur berikut untuk membuat grup Akses Terverifikasi.

Untuk membuat grup Akses Terverifikasi

1. Di panel navigasi, pilih grup Akses Terverifikasi.
2. Pilih Buat grup Akses Terverifikasi.
3. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk grup.
4. Untuk instance Akses Terverifikasi, pilih instans Akses Terverifikasi Anda.
5. Biarkan definisi Kebijakan tetap kosong. Anda akan menambahkan kebijakan tingkat grup di langkah selanjutnya.
6. Pilih Buat grup Akses Terverifikasi.

Langkah 4: Buat titik akhir Akses Terverifikasi

Gunakan prosedur berikut untuk membuat titik akhir Akses Terverifikasi. Langkah ini mengasumsikan bahwa Anda memiliki aplikasi yang berjalan di belakang penyeimbang beban internal dari Elastic Load Balancing dan sertifikat domain publik di. AWS Certificate Manager

Untuk membuat titik akhir Akses Terverifikasi

1. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
2. Pilih Buat titik akhir Akses Terverifikasi.
3. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk titik akhir.
4. Untuk grup Akses Terverifikasi, pilih grup Akses Terverifikasi Anda.
5. Untuk detail Endpoint, lakukan hal berikut:
    a. Untuk Protokol, pilih HTTPS atau HTTP, tergantung pada konfigurasi penyeimbang beban Anda.
    b. Untuk jenis Lampiran, pilih VPC.

c. Untuk tipe Endpoint, pilih Load balancer.

d. Untuk Port, masukkan nomor port yang digunakan oleh pendengar penyeimbang beban Anda. Misalnya, 443 untuk HTTPS atau 80 untuk HTTP.

e. Untuk Load balancer ARN, pilih load balancer Anda.

f. Untuk Subnet, pilih subnet yang terkait dengan penyeimbang beban Anda.

g. Untuk grup Keamanan, pilih grup keamanan Anda. Menggunakan grup keamanan yang sama untuk penyeimbang beban dan titik akhir Anda memungkinkan lalu lintas di antara mereka. Jika Anda memilih untuk tidak menggunakan grup keamanan yang sama, pastikan untuk mereferensikan grup keamanan endpoint dari penyeimbang beban Anda sehingga menerima lalu lintas dari titik akhir.

h. Untuk awalan domain Endpoint, masukkan pengenal kustom. Misalnya, **`my-ava-app`**. Awalan ini ditambahkan ke nama DNS yang dihasilkan oleh Verified Access.

6. Untuk detail Aplikasi, lakukan hal berikut:

   a. Untuk domain Aplikasi, masukkan nama DNS untuk aplikasi Anda. Domain ini harus cocok dengan yang ada di sertifikat domain Anda.

   b. Untuk ARN sertifikat Domain, pilih Nama Sumber Daya Amazon (ARN) sertifikat domain Anda di. AWS Certificate Manager

7. Jauhkan detail Kebijakan tetap kosong. Anda akan menambahkan kebijakan akses tingkat grup di langkah selanjutnya.

8. Pilih Buat titik akhir Akses Terverifikasi.

Langkah 5: Konfigurasikan DNS untuk titik akhir Akses Terverifikasi

Untuk langkah ini, Anda memetakan nama domain aplikasi Anda (misalnya, www.myapp.example.com) ke nama domain titik akhir Akses Terverifikasi Anda. Untuk menyelesaikan pemetaan DNS, buat Canonical Name Record (CNAME) dengan penyedia DNS Anda. Setelah Anda membuat catatan CNAME, semua permintaan dari pengguna ke aplikasi Anda akan dikirim ke Akses Terverifikasi.

Untuk mendapatkan nama domain dari endpoint Anda

1. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
2. Pilih titik akhir Anda.
3. Pilih tab Detail.

4. Salin domain dari domain Endpoint. Berikut ini adalah contoh nama domain endpoint:`my-ava-app.edge-1a2b3c4d5e6f7g.vai-1a2b3c4d5e6f7g.prod.verified-access.us-west-2.amazonaws.com`.

Ikuti petunjuk yang diberikan oleh penyedia DNS Anda untuk membuat catatan CNAME. Gunakan nama domain aplikasi Anda sebagai nama rekaman dan nama domain titik akhir Akses Terverifikasi sebagai nilai rekaman.

Langkah 6: Uji konektivitas ke aplikasi

Anda sekarang dapat menguji konektivitas ke aplikasi Anda. Masukkan nama domain aplikasi Anda ke browser web Anda. Perilaku default Akses Terverifikasi adalah menolak semua permintaan. Karena kami tidak menambahkan kebijakan Akses Terverifikasi ke grup atau titik akhir, semua permintaan ditolak.

Langkah 7: Tambahkan kebijakan akses tingkat grup Akses Terverifikasi

Gunakan prosedur berikut untuk mengubah grup Akses Terverifikasi dan mengonfigurasi kebijakan akses yang memungkinkan konektivitas ke aplikasi Anda. Rincian kebijakan akan tergantung pada pengguna dan grup yang dikonfigurasi di Pusat Identitas IAM. Untuk informasi, lihat Kebijakan Akses Terverifikasi.

Untuk mengubah grup Akses Terverifikasi

1. Di panel navigasi, pilih grup Akses Terverifikasi.
2. Pilih grup Anda.
3. Pilih Tindakan, Ubah kebijakan grup Akses Terverifikasi.
4. Aktifkan Aktifkan kebijakan.
5. Masukkan kebijakan yang memungkinkan pengguna dari Pusat Identitas IAM Anda untuk mengakses aplikasi Anda. Sebagai contoh, lihat the section called “Contoh kebijakan ”.
6. Pilih Ubah kebijakan grup Akses Terverifikasi.
7. Setelah kebijakan grup Anda diberlakukan, ulangi pengujian dari langkah sebelumnya untuk memverifikasi bahwa permintaan diizinkan. Jika permintaan diizinkan, Anda diminta untuk masuk

melalui halaman masuk Pusat Identitas IAM. Setelah Anda memberikan nama pengguna dan kata sandi, Anda dapat mengakses aplikasi Anda.

Bersihkan sumber daya Akses Terverifikasi

Setelah Anda selesai dengan tutorial ini, gunakan prosedur berikut untuk menghapus sumber daya Akses Terverifikasi Anda.

Untuk menghapus sumber daya Akses Terverifikasi

1. Di panel navigasi, pilih titik akhir Akses Terverifikasi. Pilih endpoint dan pilih Actions, Delete Verified Access endpoint.
2. Di panel navigasi, pilih grup Akses Terverifikasi. Pilih grup dan pilih Tindakan, Hapus grup Akses Terverifikasi. Anda mungkin perlu menunggu sampai proses penghapusan titik akhir selesai.
3. Di panel navigasi, pilih instans Akses Terverifikasi. Pilih instans Anda dan pilih Tindakan, Lepaskan penyedia kepercayaan Akses Terverifikasi. Pilih penyedia kepercayaan dan pilih Lepaskan penyedia kepercayaan Akses Terverifikasi.
4. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi. Pilih penyedia kepercayaan Anda dan pilih Actions, Delete Verified Access trust provider.
5. Di panel navigasi, pilih instans Akses Terverifikasi. Pilih instans Anda dan pilih Actions, Delete Verified Access instance.

Instans Akses Terverifikasi

Akses Terverifikasi AWS Instance adalah AWS sumber daya yang membantu Anda mengatur penyedia kepercayaan dan grup Akses Terverifikasi. Sebuah instans mengevaluasi permintaan aplikasi dan memberikan akses hanya ketika persyaratan keamanan Anda terpenuhi.

Tugas

- Membuat dan mengelola instance Akses Terverifikasi
- Menghapus instans Akses Terverifikasi
- Integrasikan Akses Terverifikasi dengan AWS WAF
- Kepatuhan FIPS untuk Akses Terverifikasi

Membuat dan mengelola instance Akses Terverifikasi

Anda menggunakan instans Akses Terverifikasi untuk mengatur penyedia kepercayaan dan grup Akses Terverifikasi. Gunakan prosedur berikut untuk membuat instance Akses Terverifikasi, lalu lampirkan penyedia kepercayaan ke Akses Terverifikasi atau lepaskan penyedia kepercayaan dari Akses Terverifikasi.

Tugas

- Buat instance Akses Terverifikasi
- Lampirkan penyedia kepercayaan ke instans Akses Terverifikasi
- Lepaskan penyedia kepercayaan dari instans Akses Terverifikasi
- Tambahkan subdomain khusus

Buat instance Akses Terverifikasi

Gunakan prosedur berikut untuk membuat instance Akses Terverifikasi.

Untuk membuat instance Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instance Akses Terverifikasi, lalu Buat instance Akses Terverifikasi.
3. (Opsional) Untuk Nama dan Deskripsi, masukkan nama dan deskripsi untuk instance Akses Terverifikasi.

4. (Titik akhir CIDR Jaringan) Untuk subdomain khusus untuk titik akhir CIDR jaringan, masukkan subdomain khusus.
5. (Opsional) Pilih Aktifkan untuk Standar Proses Informasi Federal (FIPS) jika Anda memerlukan Akses Terverifikasi agar sesuai dengan FIPS.
6. (Opsional) Untuk penyedia kepercayaan Akses Terverifikasi, pilih penyedia kepercayaan untuk dilampirkan ke instance Akses Terverifikasi.
7. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
8. Pilih Buat instance Akses Terverifikasi.

Untuk membuat instance Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah create-verified-access-instance.

Lampirkan penyedia kepercayaan ke instans Akses Terverifikasi

Gunakan prosedur berikut untuk melampirkan penyedia kepercayaan ke instance Akses Terverifikasi.

Untuk melampirkan penyedia kepercayaan ke instance Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instans.
4. Pilih Tindakan, Lampirkan penyedia kepercayaan Akses Terverifikasi.
5. Untuk penyedia kepercayaan Akses Terverifikasi, pilih penyedia kepercayaan.
6. Pilih Lampirkan penyedia kepercayaan Akses Terverifikasi.

Untuk melampirkan penyedia kepercayaan ke instans Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah attach-verified-access-trust-provider.

Lepaskan penyedia kepercayaan dari instans Akses Terverifikasi

Gunakan prosedur berikut untuk melepaskan penyedia kepercayaan dari instance Akses Terverifikasi.

Untuk melepaskan penyedia kepercayaan dari instance Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instans.
4. Pilih Tindakan, Lepaskan penyedia kepercayaan Akses Terverifikasi.
5. Untuk penyedia kepercayaan Akses Terverifikasi, pilih penyedia kepercayaan.
6. Pilih Lepaskan penyedia kepercayaan Akses Terverifikasi.

Untuk melepaskan penyedia kepercayaan dari instans Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah detach-verified-access-trust-provider.

Tambahkan subdomain khusus

Gunakan prosedur berikut untuk menambah atau memperbarui subdomain kustom. Subdomain ini hanya digunakan ketika Anda membuat titik akhir CIDR jaringan.

Untuk menambahkan subdomain kustom menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instans.
4. Pilih Tindakan, Ubah instance Akses Terverifikasi.
5. Untuk subdomain kustom untuk titik akhir CIDR jaringan, masukkan subdomain kustom.
6. Pilih Modify Verified Access instance.
7. Perbarui server nama untuk subdomain Anda, masukkan server nama yang disediakan oleh Akses Terverifikasi. Daftar ini tersedia di bawah Nameservers pada tab Detail untuk instance.

Untuk menambahkan subdomain kustom menggunakan AWS CLI

Gunakan perintah modify-verified-access-instance.

Menghapus instans Akses Terverifikasi

Setelah selesai dengan instance Akses Terverifikasi, Anda dapat menghapusnya. Sebelum menghapus instans, Anda harus menghapus penyedia kepercayaan terkait atau grup Akses Terverifikasi.

Untuk menghapus instance Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi.
4. Pilih Tindakan, Hapus instans Akses Terverifikasi.
5. Saat diminta konfirmasi, masukkan**`delete`**, lalu pilih Hapus.

Untuk menghapus instance Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah delete-verified-access-instance.

Integrasikan Akses Terverifikasi dengan AWS WAF

Selain aturan otentikasi dan otorisasi yang diberlakukan oleh Akses Terverifikasi, Anda mungkin juga ingin menerapkan perlindungan perimeter. Ini dapat membantu Anda melindungi aplikasi Anda dari ancaman tambahan. Anda dapat melakukannya dengan mengintegrasikan AWS WAF ke dalam penerapan Akses Terverifikasi Anda. AWS WAF adalah firewall aplikasi web yang memungkinkan Anda memantau permintaan HTTP yang diteruskan ke sumber daya aplikasi web Anda yang dilindungi. Lihat informasi selengkapnya di Panduan Developer AWS WAF.

Anda dapat mengintegrasikan AWS WAF dengan Akses Terverifikasi dengan mengaitkan daftar kontrol akses AWS WAF web (ACL) dengan instans Akses Terverifikasi. ACL web adalah AWS WAF sumber daya yang memberi Anda kontrol halus atas semua permintaan web HTTP yang ditanggapi oleh sumber daya terlindungi Anda. Saat permintaan AWS WAF asosiasi atau disasosiasi sedang diproses, status titik akhir Akses Terverifikasi yang dilampirkan ke instance ditampilkan sebagai. `updating` Setelah permintaan selesai, status kembali k`eactive`. Anda dapat melihat status di AWS Management Console atau dengan menjelaskan titik akhir dengan. AWS CLI

Penyedia kepercayaan identitas pengguna menentukan kapan AWS WAF memeriksa lalu lintas. Jika Anda menggunakan IAM Identity Center, AWS WAF periksa lalu lintas sebelum autentikasi

pengguna. Jika Anda menggunakan OpenID Connect (OIDC), AWS WAF periksa lalu lintas setelah otentikasi pengguna.

Daftar Isi

- Izin IAM yang diperlukan
- Kaitkan ACL AWS WAF web
- Periksa status asosiasi
- Putuskan hubungan ACL AWS WAF web

Izin IAM yang diperlukan

Mengintegrasikan AWS WAF dengan Akses Terverifikasi mencakup tindakan khusus izin yang tidak berhubungan langsung dengan operasi API. Tindakan ini ditunjukkan dalam Referensi Otorisasi AWS Identity and Access Management Layanan dengan`[permission only]`. Lihat Kunci tindakan, sumber daya, dan kondisi untuk Amazon EC2 di Referensi Otorisasi Layanan.

Untuk bekerja dengan ACL web, AWS Identity and Access Management kepala sekolah Anda harus memiliki izin berikut.

- `ec2:AssociateVerifiedAccessInstanceWebAcl`
- `ec2:DisassociateVerifiedAccessInstanceWebAcl`
- `ec2:DescribeVerifiedAccessInstanceWebAclAssociations`
- `ec2:GetVerifiedAccessInstanceWebAcl`

Kaitkan ACL AWS WAF web

Langkah-langkah berikut menunjukkan cara mengaitkan daftar kontrol akses AWS WAF web (ACL) dengan instance Akses Terverifikasi menggunakan konsol Akses Terverifikasi.

Prasyarat

Sebelum Anda mulai, buat ACL AWS WAF web. Untuk informasi selengkapnya, lihat Membuat ACL web di Panduan AWS WAF Pengembang.

Untuk mengaitkan ACL AWS WAF web ke instans Akses Terverifikasi

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/

2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi.
4. Pilih tab Integrasi.
5. Pilih Actions, lalu Associate Web ACL.
6. Untuk Web ACL, pilih ACL web yang ada, lalu pilih Associate Web ACL.

Atau, Anda dapat menggunakan AWS WAF konsol. Jika Anda menggunakan AWS WAF konsol atau API, Anda memerlukan Nama Sumber Daya Amazon (ARN) dari instance Akses Terverifikasi. AVA ARN memiliki format berikut:. `arn:${Partition}:ec2:${Region}:${Account}:verified-access-instance/${VerifiedAccessInstanceId}` Untuk informasi selengkapnya, lihat Mengaitkan ACL web dengan AWS sumber daya di Panduan AWS WAF Pengembang.

Periksa status asosiasi

Anda dapat memverifikasi apakah daftar kontrol akses AWS WAF web (ACL) dikaitkan dengan instans Akses Terverifikasi atau tidak dengan menggunakan konsol Akses Terverifikasi.

Untuk melihat status AWS WAF integrasi dengan instans Akses Terverifikasi

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi.
4. Pilih tab Integrasi.
5. Periksa detail yang tercantum di bawah status integrasi WAF. Status akan ditampilkan sebagai Terkait atau Tidak terkait, bersama dengan pengenal ACL web, jika dalam status Terkait.

Putuskan hubungan ACL AWS WAF web

Langkah-langkah berikut menunjukkan cara memisahkan daftar kontrol akses AWS WAF web (ACL) dari instance Akses Terverifikasi menggunakan konsol Akses Terverifikasi.

Untuk memisahkan ACL AWS WAF web dari instans Akses Terverifikasi

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.

3. Pilih instance Akses Terverifikasi.
4. Pilih tab Integrasi.
5. Pilih Actions, lalu Disassociate Web ACL.
6. Konfirmasikan dengan memilih Disassociate Web ACL.

Atau, Anda dapat menggunakan AWS WAF konsol. Untuk informasi selengkapnya, lihat Memutuskan hubungan ACL web dari AWS sumber daya di Panduan AWS WAF Pengembang.

Kepatuhan FIPS untuk Akses Terverifikasi

Federal Information Processing Standard (FIPS) adalah standar pemerintah AS dan Kanada yang menetapkan persyaratan keamanan untuk modul kriptografi yang melindungi informasi sensitif. Akses Terverifikasi AWS menyediakan opsi untuk mengonfigurasi lingkungan Anda agar mematuhi FIPS Publication 140-2. Kepatuhan FIPS untuk Akses Terverifikasi tersedia di AWS Wilayah berikut:

- AS Timur (Ohio)
- AS Timur (Virginia Utara)
- AS Barat (California Utara)
- AS Barat (Oregon)
- Kanada (Pusat)
- AWS GovCloud (US) Barat
- AWS GovCloud (US) Timur

Halaman ini menunjukkan cara mengonfigurasi lingkungan Akses Terverifikasi baru, atau yang sudah ada, agar sesuai dengan FIPS.

Daftar Isi

- Konfigurasikan lingkungan Akses Terverifikasi yang ada untuk kepatuhan FIPS
- Konfigurasikan lingkungan Akses Terverifikasi baru untuk kepatuhan FIPS

Konfigurasikan lingkungan Akses Terverifikasi yang ada untuk kepatuhan FIPS

Jika Anda memiliki lingkungan Akses Terverifikasi yang ada dan Anda ingin mengonfigurasinya agar sesuai dengan FIPS, beberapa sumber daya perlu dihapus dan dibuat ulang untuk mengaktifkan kepatuhan FIPS.

Untuk mengonfigurasi ulang Akses Terverifikasi AWS lingkungan yang ada agar sesuai dengan FIPS, ikuti langkah-langkah di bawah ini.

1. Hapus titik akhir, grup, dan instans Akses Terverifikasi asli Anda. Penyedia kepercayaan Anda yang dikonfigurasi dapat digunakan kembali.
2. Buat instance Akses Terverifikasi, pastikan untuk mengaktifkan Standar Proses Informasi Federal (FIPS) selama pembuatan. Juga selama pembuatan, lampirkan penyedia kepercayaan Akses Terverifikasi yang ingin Anda gunakan, dengan memilihnya dari daftar drop-down.
3. Buat grup Akses Terverifikasi. Selama pembuatan grup, Anda mengaitkannya dengan instance Akses Terverifikasi yang baru saja dibuat.
4. Buat satu atau lebihTitik akhir Akses Terverifikasi. Selama pembuatan titik akhir Anda, Anda mengaitkannya dengan grup yang dibuat pada langkah sebelumnya.

Konfigurasikan lingkungan Akses Terverifikasi baru untuk kepatuhan FIPS

Untuk mengonfigurasi Akses Terverifikasi AWS lingkungan baru yang sesuai dengan FIPS, ikuti langkah-langkah di bawah ini.

1. Konfigurasikan penyedia kepercayaan. Anda perlu membuat penyedia kepercayaan identitas pengguna dan (opsional) penyedia kepercayaan berbasis perangkat, tergantung pada kebutuhan Anda.
2. Buat instance Akses Terverifikasi, pastikan untuk mengaktifkan Standar Proses Informasi Federal (FIPS) selama proses berlangsung. Juga selama pembuatan, lampirkan penyedia kepercayaan Akses Terverifikasi yang Anda buat di langkah sebelumnya, dengan memilihnya dari daftar drop-down.
3. Buat grup Akses Terverifikasi. Selama pembuatan grup, Anda mengaitkannya dengan instance Akses Terverifikasi yang baru saja dibuat.
4. Buat satu atau lebihTitik akhir Akses Terverifikasi. Selama pembuatan titik akhir Anda, Anda mengaitkannya dengan grup yang dibuat pada langkah sebelumnya.

Penyedia kepercayaan untuk Akses Terverifikasi

Penyedia kepercayaan adalah layanan yang mengirimkan informasi tentang pengguna dan perangkat ke Akses Terverifikasi AWS. Informasi ini disebut konteks kepercayaan. Ini dapat mencakup atribut berdasarkan identitas pengguna, seperti alamat email atau keanggotaan dalam organisasi “penjualan”, atau informasi perangkat seperti patch keamanan yang diinstal atau versi perangkat lunak anti-virus.

Akses Terverifikasi mendukung kategori penyedia kepercayaan berikut:

- Identitas pengguna — Layanan penyedia identitas (iDP) yang menyimpan dan mengelola identitas digital untuk pengguna.
- Manajemen perangkat — Sistem manajemen perangkat untuk perangkat seperti laptop, tablet, dan smartphone.

Daftar Isi

- Penyedia kepercayaan identitas pengguna untuk Akses Terverifikasi
- Penyedia kepercayaan berbasis perangkat untuk Akses Terverifikasi

Penyedia kepercayaan identitas pengguna untuk Akses Terverifikasi

Anda dapat memilih untuk menggunakan salah satu AWS IAM Identity Center atau penyedia kepercayaan identitas pengguna yang kompatibel dengan OpenID Connect.

Daftar Isi

- Menggunakan IAM Identity Center sebagai penyedia kepercayaan
- Menggunakan penyedia kepercayaan OpenID Connect

Menggunakan IAM Identity Center sebagai penyedia kepercayaan

Anda dapat menggunakan AWS IAM Identity Center sebagai penyedia kepercayaan identitas pengguna dengan Akses AWS Terverifikasi.

Prasyarat dan pertimbangan

- Instance IAM Identity Center Anda harus berupa sebuah AWS Organizations instance. Instans Pusat Identitas IAM AWS akun mandiri tidak akan berfungsi.
- Instance Pusat Identitas IAM Anda harus diaktifkan di AWS Wilayah yang sama tempat Anda ingin membuat penyedia kepercayaan Akses Terverifikasi.
- Akses Terverifikasi dapat menyediakan akses ke pengguna di Pusat Identitas IAM yang ditetapkan hingga 1.000 grup.

Lihat Mengelola instans organisasi dan akun Pusat Identitas IAM di Panduan AWS IAM Identity Center Pengguna untuk detail tentang berbagai jenis instans.

Buat penyedia kepercayaan Pusat Identitas IAM

Setelah Pusat Identitas IAM diaktifkan di AWS akun Anda, Anda dapat menggunakan prosedur berikut untuk menyiapkan Pusat Identitas IAM sebagai penyedia kepercayaan Anda untuk Akses Terverifikasi.

Untuk membuat penyedia kepercayaan Pusat Identitas IAM (AWS konsol)

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi, lalu Buat penyedia kepercayaan Akses Terverifikasi.
3. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk penyedia kepercayaan.
4. Untuk nama referensi Kebijakan, masukkan pengenal yang akan digunakan nanti saat bekerja dengan aturan kebijakan.
5. Di bawah Jenis penyedia Trust, pilih Penyedia kepercayaan pengguna.
6. Di bawah Jenis penyedia kepercayaan pengguna, pilih Pusat Identitas IAM.
7. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
8. Pilih Buat penyedia kepercayaan Akses Terverifikasi.

Untuk membuat penyedia kepercayaan Pusat Identitas IAM (AWS CLI)

- create-verified-access-trust-penyedia ()AWS CLI

Hapus penyedia kepercayaan Pusat Identitas IAM

Sebelum Anda dapat menghapus penyedia kepercayaan, Anda harus menghapus semua konfigurasi titik akhir dan grup dari instance yang dilampirkan penyedia kepercayaan.

Untuk menghapus penyedia kepercayaan Pusat Identitas IAM (AWS konsol)

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi, lalu pilih penyedia kepercayaan yang ingin Anda hapus di bawah Penyedia kepercayaan Akses Terverifikasi.
3. Pilih Tindakan, lalu Hapus penyedia kepercayaan Akses Terverifikasi.
4. Konfirmasikan penghapusan dengan memasukkan `delete` ke dalam kotak teks.
5. Pilih Hapus.

Untuk menghapus penyedia kepercayaan Pusat Identitas IAM (AWS CLI)

- delete-verified-access-trust-penyedia ()AWS CLI

Menggunakan penyedia kepercayaan OpenID Connect

Akses Terverifikasi AWS mendukung penyedia identitas yang menggunakan metode OpenID Connect (OIDC) standar. Anda dapat menggunakan penyedia yang kompatibel dengan OIDC sebagai penyedia kepercayaan identitas pengguna dengan Akses Terverifikasi. Namun, karena beragam penyedia OIDC potensial, AWS tidak dapat menguji setiap integrasi OIDC dengan Akses Terverifikasi.

Akses Terverifikasi memperoleh data kepercayaan yang dievaluasi dari penyedia OIDC. `UserInfo Endpoint` `Scope`Parameter ini digunakan untuk menentukan kumpulan data kepercayaan mana yang akan diambil. Setelah data kepercayaan diterima, kebijakan Akses Terverifikasi dievaluasi terhadapnya.

Klaim token ID dari penyedia kepercayaan OIDC disertakan dalam `addition_user_context` kunci, untuk penyedia kepercayaan yang dibuat setelah 24 Februari 2025.

Dengan penyedia kepercayaan yang dibuat pada atau sebelum 24 Februari 2025, Akses Terverifikasi tidak menggunakan data kepercayaan dari yang `ID token` dikirim oleh penyedia OIDC. Hanya data kepercayaan dari yang `UserInfo Endpoint` dievaluasi terhadap kebijakan.

Dengan penyedia kepercayaan yang dibuat pada atau sebelum 24 Februari 2025, durasi sesi default adalah satu hari. Dengan penyedia kepercayaan yang dibuat sebelum 24 Februari 2025, durasi sesi default adalah tujuh hari.

Jika token penyegaran ditentukan, Akses Terverifikasi menggunakan kedaluwarsa token penyegaran sebagai durasi sesi. Jika tidak ada token penyegaran, durasi sesi default digunakan.

Daftar Isi

- Prasyarat untuk membuat penyedia kepercayaan OIDC
- Buat penyedia kepercayaan OIDC
- Memodifikasi penyedia kepercayaan OIDC
- Hapus penyedia kepercayaan OIDC

Prasyarat untuk membuat penyedia kepercayaan OIDC

Anda perlu mengumpulkan informasi berikut dari layanan penyedia kepercayaan Anda secara langsung:

- Penerbit
- Titik akhir otorisasi
- Titik akhir token
- UserInfo titik akhir
- ID Klien
- Rahasia klien
- Cakupan

Buat penyedia kepercayaan OIDC

Gunakan prosedur berikut untuk membuat OIDC sebagai penyedia kepercayaan Anda.

Untuk membuat penyedia kepercayaan OIDC (konsol)AWS

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi, lalu Buat penyedia kepercayaan Akses Terverifikasi.

3. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk penyedia kepercayaan.
4. Untuk nama referensi Kebijakan, masukkan pengenal yang akan digunakan nanti saat bekerja dengan aturan kebijakan.
5. Di bawah Jenis penyedia Trust, pilih Penyedia kepercayaan pengguna.
6. Di bawah Jenis penyedia kepercayaan pengguna, pilih OIDC (OpenID Connect).
7. Untuk OIDC (OpenID Connect), pilih penyedia kepercayaan.
8. Untuk Penerbit, masukkan pengenal penerbit OIDC.
9. Untuk titik akhir Otorisasi, masukkan URL lengkap titik akhir otorisasi.
10. Untuk titik akhir Token, masukkan URL lengkap titik akhir token.
11. Untuk titik akhir Pengguna, masukkan URL lengkap titik akhir pengguna.
12. (Native Application OIDC) Untuk URL kunci penandatanganan publik, masukkan URL lengkap dari titik akhir kunci penandatanganan publik.
13. Masukkan pengenal klien OAuth 2.0 untuk ID Klien.
14. Masukkan rahasia klien OAuth 2.0 untuk rahasia Klien.
15. Masukkan daftar cakupan yang dibatasi spasi yang ditentukan dengan penyedia identitas Anda. Minimal, ruang openid lingkup diperlukan untuk Lingkup.
16. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
17. Pilih Buat penyedia kepercayaan Akses Terverifikasi.
18. Anda harus menambahkan URI pengalihan ke daftar izin untuk penyedia OIDC Anda.
    - Aplikasi HTTP — Gunakan URI berikut:**`https://application_domain/oauth2/idpresponse`**. Di konsol, Anda dapat menemukan domain aplikasi pada tab Detail untuk titik akhir Akses Terverifikasi. Menggunakan AWS CLI atau AWS SDK, domain aplikasi disertakan dalam output saat Anda menjelaskan titik akhir Akses Terverifikasi.
    - Aplikasi TCP — Gunakan URI berikut:**`http://localhost:8000`**.

Untuk membuat penyedia kepercayaan OIDC (CLI AWS)

- create-verified-access-trust-penyedia ()AWS CLI

Memodifikasi penyedia kepercayaan OIDC

Setelah Anda membuat penyedia kepercayaan, Anda dapat memperbarui konfigurasinya.

Untuk memodifikasi penyedia kepercayaan OIDC (konsol)AWS

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi, lalu pilih penyedia kepercayaan yang ingin Anda ubah di bawah Penyedia kepercayaan Akses Terverifikasi.
3. Pilih Tindakan, lalu Ubah penyedia kepercayaan Akses Terverifikasi.
4. Ubah opsi yang ingin Anda ubah.
5. Pilih Ubah penyedia kepercayaan Akses Terverifikasi.

Untuk memodifikasi penyedia kepercayaan OIDC (CLI AWS)

- modify-verified-access-trust-penyedia ()AWS CLI

Hapus penyedia kepercayaan OIDC

Sebelum dapat menghapus penyedia kepercayaan pengguna, pertama-tama Anda harus menghapus semua konfigurasi titik akhir dan grup dari contoh penyedia kepercayaan yang dilampirkan.

Untuk menghapus penyedia kepercayaan OIDC (konsol)AWS

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi, lalu pilih penyedia kepercayaan yang ingin Anda hapus di bawah Penyedia kepercayaan Akses Terverifikasi.
3. Pilih Tindakan, lalu Hapus penyedia kepercayaan Akses Terverifikasi.
4. Konfirmasikan penghapusan dengan memasukkan `delete` ke dalam kotak teks.
5. Pilih Hapus.

Untuk menghapus penyedia kepercayaan OIDC (CLI AWS)

- delete-verified-access-trust-penyedia ()AWS CLI

Penyedia kepercayaan berbasis perangkat untuk Akses Terverifikasi

Anda dapat menggunakan penyedia kepercayaan perangkat dengan Akses AWS Terverifikasi. Anda dapat menggunakan satu atau beberapa penyedia kepercayaan perangkat dengan instans Akses Terverifikasi.

Daftar Isi

- Penyedia kepercayaan perangkat yang didukung
- Buat penyedia kepercayaan berbasis perangkat
- Memodifikasi penyedia kepercayaan berbasis perangkat
- Menghapus penyedia kepercayaan berbasis perangkat

Penyedia kepercayaan perangkat yang didukung

Penyedia kepercayaan perangkat berikut dapat diintegrasikan dengan Akses Terverifikasi:

- CrowdStrike — Mengamankan aplikasi pribadi dengan CrowdStrike dan Akses AWS Terverifikasi
- Jamf - Mengintegrasikan Akses Terverifikasi dengan Identitas Perangkat Jamf
- JumpCloud — Mengintegrasikan JumpCloud dan Akses AWS Terverifikasi

Buat penyedia kepercayaan berbasis perangkat

Ikuti langkah-langkah berikut untuk membuat dan mengonfigurasi penyedia kepercayaan perangkat untuk digunakan dengan Akses Terverifikasi.

Untuk membuat penyedia kepercayaan perangkat Akses Terverifikasi (AWS konsol)

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi, lalu Buat penyedia kepercayaan Akses Terverifikasi.
3. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk penyedia kepercayaan.
4. Masukkan pengenal yang akan digunakan nanti saat bekerja dengan aturan kebijakan untuk nama referensi Kebijakan.

5. Untuk jenis penyedia Trust, pilih Identitas perangkat.
6. Untuk jenis identitas Perangkat, pilih Jamf, CrowdStrike, atau JumpCloud.
7. Untuk ID Penyewa, masukkan pengenal aplikasi penyewa.
8. (Opsional) Untuk URL kunci penandatanganan publik, masukkan URL kunci unik yang dibagikan oleh penyedia kepercayaan perangkat Anda. (Parameter ini tidak diperlukan untuk Jamf, CrowdStrike atau Jumpcloud.)
9. Pilih Buat penyedia kepercayaan Akses Terverifikasi.

> **Note**
>
> Anda perlu menambahkan URI pengalihan ke daftar izin penyedia OIDC Anda.
> Anda akan ingin menggunakan titik akhir Akses Terverifikasi untuk tujuan ini.
> `DeviceValidationDomain` Ini dapat ditemukan di AWS Management Console, di bawah tab Detail untuk titik akhir Akses Terverifikasi Anda atau dengan menggunakan AWS CLI untuk menggambarkan titik akhir. Tambahkan yang berikut ini ke daftar izin penyedia OIDC Anda: https:///oauth2/idpresponse `DeviceValidationDomain`

Untuk membuat penyedia kepercayaan perangkat Akses Terverifikasi (AWS CLI)

- create-verified-access-trust-penyedia ()AWS CLI

Memodifikasi penyedia kepercayaan berbasis perangkat

Setelah Anda membuat penyedia kepercayaan, Anda dapat memperbarui konfigurasinya.

Untuk mengubah penyedia kepercayaan perangkat Akses Terverifikasi (AWS konsol)

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi.
3. Pilih penyedia kepercayaan.
4. Pilih Tindakan, lalu pilih Ubah penyedia kepercayaan Akses Terverifikasi.
5. Ubah deskripsi sesuai kebutuhan.

6. (Opsional) Untuk URL kunci penandatanganan publik, ubah URL kunci unik yang dibagikan oleh penyedia kepercayaan perangkat Anda. (Parameter ini tidak diperlukan jika penyedia kepercayaan perangkat Anda adalah Jamf, CrowdStrike atau Jumpcloud.)
7. Pilih Ubah penyedia kepercayaan Akses Terverifikasi.

Untuk mengubah penyedia kepercayaan perangkat Akses Terverifikasi (AWS CLI)

- modify-verified-access-trust-penyedia ()AWS CLI

Menghapus penyedia kepercayaan berbasis perangkat

Setelah selesai dengan penyedia kepercayaan, Anda dapat menghapusnya.

Untuk menghapus penyedia kepercayaan perangkat Akses Terverifikasi (AWS konsol)

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih Penyedia kepercayaan Akses Terverifikasi.
3. Pilih penyedia kepercayaan yang ingin Anda hapus di bawah Penyedia kepercayaan Akses Terverifikasi.
4. Pilih Tindakan, lalu pilih Hapus penyedia kepercayaan Akses Terverifikasi.
5. Saat diminta konfirmasi, masukkan**`delete`**, lalu pilih Hapus.

Untuk menghapus penyedia kepercayaan perangkat Akses Terverifikasi (AWS CLI)

- delete-verified-access-trust-penyedia ()AWS CLI

Grup Akses Terverifikasi

Grup Akses Terverifikasi terdiri dari titik akhir Akses Terverifikasi dan kebijakan Akses Terverifikasi yang berlaku untuk semua titik akhir dalam grup. Dengan mengelompokkan titik akhir yang memiliki persyaratan keamanan umum, Anda dapat menentukan kebijakan grup tunggal yang memenuhi persyaratan keamanan minimum beberapa titik akhir. Oleh karena itu, Anda tidak perlu membuat dan memelihara kebijakan untuk setiap titik akhir.

Misalnya, Anda dapat mengelompokkan semua aplikasi penjualan bersama-sama dan menetapkan kebijakan akses seluruh grup. Anda kemudian dapat menggunakan kebijakan ini untuk menentukan serangkaian persyaratan keamanan minimum yang umum untuk semua aplikasi penjualan. Pendekatan ini membantu menyederhanakan administrasi kebijakan.

Saat membuat grup, Anda harus mengaitkan grup dengan instance Akses Terverifikasi. Selama proses pembuatan titik akhir, Anda akan mengaitkan titik akhir dengan grup.

Fitur lain dari grup Akses Terverifikasi adalah kemampuan untuk membagikannya dengan AWS akun lain menggunakan AWS RAM. Ini memungkinkan Anda untuk membuat dan mengelola grup secara terpusat dalam satu akun, lalu membagikannya dengan beberapa akun.

Tugas

- Membuat dan mengelola grup Akses Terverifikasi
- Ubah kebijakan grup Akses Terverifikasi
- Bagikan grup Akses Terverifikasi dengan grup lain Akun AWS
- Menghapus grup Akses Terverifikasi

Membuat dan mengelola grup Akses Terverifikasi

Anda menggunakan grup Akses Terverifikasi untuk mengatur titik akhir berdasarkan persyaratan keamanannya. Saat membuat titik akhir Akses Terverifikasi, Anda mengaitkan titik akhir dengan grup.

Tugas

- Membuat grup Akses Terverifikasi
- Memodifikasi grup Akses Terverifikasi

Membuat grup Akses Terverifikasi

Gunakan prosedur berikut untuk membuat grup Akses Terverifikasi. Sebelum membuat grup Akses Terverifikasi, Anda harus membuat instance Akses Terverifikasi. Untuk informasi selengkapnya, lihat the section called “ Buat instance Akses Terverifikasi”.

Untuk membuat grup Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih grup Akses Terverifikasi, lalu Buat grup Akses Terverifikasi.
3. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk grup.
4. Untuk instance Akses Terverifikasi, pilih instance Akses Terverifikasi untuk dikaitkan dengan grup.
5. (Opsional) Untuk definisi Kebijakan, masukkan kebijakan Akses Terverifikasi untuk diterapkan ke grup.
6. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
7. Pilih Buat grup Akses Terverifikasi.

Untuk membuat grup Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah create-verified-access-group.

Memodifikasi grup Akses Terverifikasi

Gunakan prosedur berikut untuk memodifikasi grup Akses Terverifikasi.

Untuk mengubah grup Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih grup Akses Terverifikasi, lalu Buat grup Akses Terverifikasi.
3. Pilih grup, lalu pilih Actions, Modify Verified Access group.
4. (Opsional) Perbarui deskripsi.
5. Pilih Buat grup Akses Terverifikasi.
6. Pilih instance Akses Terverifikasi untuk diasosiasikan dengan grup.

Untuk mengubah grup Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-group.

Ubah kebijakan grup Akses Terverifikasi

Akses Terverifikasi AWS memungkinkan akses ke aplikasi Anda berdasarkan kebijakan akses yang Anda buat. Kebijakan Akses Terverifikasi yang Anda lampirkan ke grup diwarisi oleh semua titik akhir dalam grup. Anda dapat secara opsional melampirkan kebijakan khusus aplikasi ke titik akhir tertentu.

Gunakan prosedur berikut untuk mengubah kebijakan grup Akses Terverifikasi. Setelah Anda melakukan perubahan, dibutuhkan beberapa menit sebelum diterapkan.

Untuk mengubah kebijakan grup Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih grup Akses Terverifikasi.
3. Pilih grup .
4. Pilih Tindakan, Ubah kebijakan grup Akses Terverifikasi.
5. (Opsional) Aktifkan atau nonaktifkan Aktifkan kebijakan sesuai kebutuhan.
6. (Opsional) Untuk Kebijakan, masukkan kebijakan Akses Terverifikasi untuk diterapkan ke grup.
7. Pilih Ubah kebijakan grup Akses Terverifikasi.

Untuk mengubah kebijakan grup Akses Terverifikasi menggunakan AWS CLI

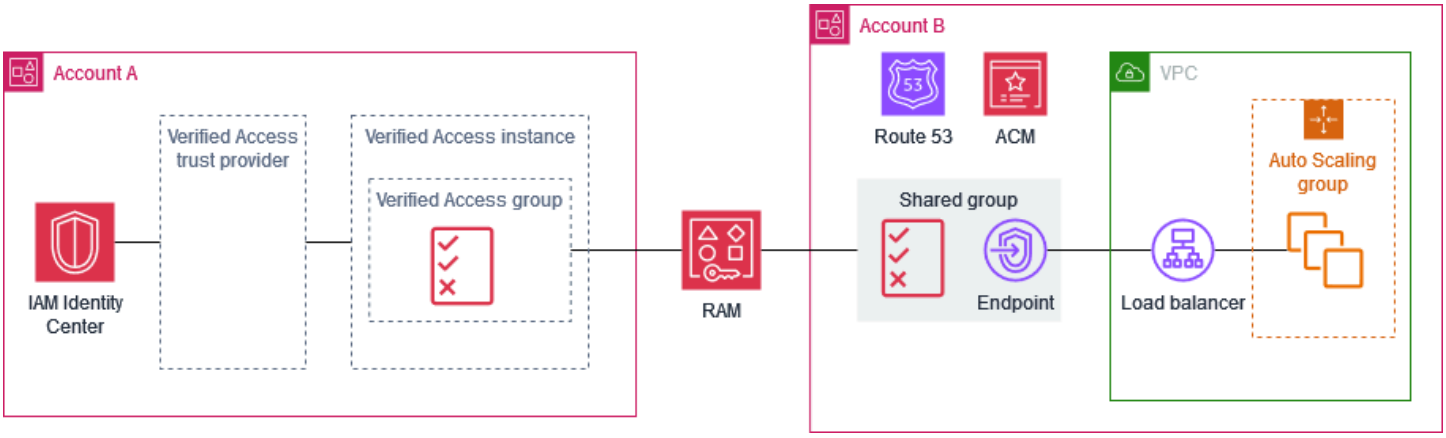
Gunakan perintah modify-verified-access-group-policy.

Bagikan grup Akses Terverifikasi dengan grup lain Akun AWS

Saat membagikan grup Akses Terverifikasi yang Anda miliki dengan AWS akun lain, Anda mengaktifkan akun tersebut untuk membuat titik akhir Akses Terverifikasi di grup Anda. Akun yang membuat grup Akses Terverifikasi disebut sebagai akun pemilik. Akun yang menggunakan grup bersama disebut sebagai akun konsumen.

Diagram berikut menggambarkan manfaat berbagi grup Akses Terverifikasi. Tim keamanan pusat memiliki Akun A. Mereka mengelola pengguna dan grup AWS IAM Identity Center, dan mengelola sumber daya Akses Terverifikasi yang diperlukan untuk menyediakan akses ke aplikasi internal, seperti penyedia kepercayaan Akses Terverifikasi, instans Akses Terverifikasi, grup Akses

Terverifikasi, dan kebijakan Akses Terverifikasi. Tim aplikasi memiliki Akun B. Mereka mengelola sumber daya yang diperlukan untuk menjalankan aplikasi internal mereka, seperti load balancer, grup Auto Scaling, konfigurasi DNS di Amazon Route 53, dan sertifikat TLS AWS Certificate Manager dari (ACM). Setelah tim keamanan pusat membagikan grup Akses Terverifikasi dengan Akun B, tim aplikasi dapat membuat titik akhir Akses Terverifikasi menggunakan grup bersama. Akses ke aplikasi diizinkan atau ditolak berdasarkan kebijakan yang dibuat oleh tim keamanan pusat untuk grup Akses Terverifikasi.



Pertimbangan

Pertimbangan berikut berlaku untuk grup Akses Terverifikasi bersama.

Pemilik

- Untuk berbagi grup Akses Terverifikasi, pengguna harus memiliki izin berikut: `ec2:PutResourcePolicy` dan`ec2:DeleteResourcePolicy`.
- Untuk berbagi grup Akses Terverifikasi, Anda harus memilikinya. Anda tidak dapat membagikan grup Akses Terverifikasi yang dibagikan dengan Anda.
- Jika mengaktifkan berbagi dengan akun di organisasi, Anda dapat berbagi sumber daya, seperti grup Akses Terverifikasi, tanpa menggunakan undangan. Jika tidak, konsumen menerima undangan dan harus menerimanya untuk mengakses grup bersama. Untuk mengaktifkan berbagi, dari akun manajemen untuk organisasi Anda, buka halaman Pengaturan di AWS RAM konsol dan pilih Aktifkan berbagi dengan AWS Organizations.
- Anda tidak dapat menghapus grup jika ada titik akhir Akses Terverifikasi terkait. Anda dapat melihat titik akhir yang dibuat oleh akun konsumen di halaman titik akhir Akses Terverifikasi di akun Anda. ID akun pemilik titik akhir tercermin dalam Nama Sumber Daya Amazon (ARN) sertifikat untuk titik akhir.

Konsumen

- Untuk melihat grup Akses Terverifikasi yang dibagikan dengan Anda, buka halaman grup Akses Terverifikasi di konsol, atau hubungi describe-verified-access-groups. ID akun pemilik tercermin di bidang Pemilik dan Nama Sumber Daya Amazon (ARN) grup.
- Saat membuat titik akhir Akses Terverifikasi, Anda dapat menentukan grup Akses Terverifikasi yang dibagikan dengan Anda.
- Anda tidak dapat melihat titik akhir yang terkait dengan grup bersama tetapi tidak dimiliki oleh Anda.
- Jika pemilik grup Akses Terverifikasi menghapus pembagian sumber daya, Anda tidak dapat membuat titik akhir Akses Terverifikasi baru di grup. Setiap titik akhir Akses Terverifikasi yang Anda buat sebelum penghapusan pembagian sumber daya tidak terpengaruh oleh penghapusan pembagian sumber daya. Namun, pemilik grup bersama dapat menghapus titik akhir Anda.

Pembagian sumber daya

Untuk membagikan grup Akses Terverifikasi, Anda harus menambahkannya ke pembagian sumber daya. Pembagian sumber daya menentukan sumber daya untuk dibagikan dan konsumen yang dapat menggunakan sumber daya bersama.

Untuk berbagi grup Akses Terverifikasi menggunakan konsol

1. Buka AWS RAM konsol di https://console.aws.amazon.com/ram/rumah.
2. Jika Anda tidak memiliki pembagian sumber daya untuk organisasi Anda, buatlah satu. Untuk kepala sekolah, Anda dapat memilih seluruh organisasi, unit organisasi, atau AWS akun tertentu.
3. Pilih pembagian sumber daya Anda dan pilih Ubah.
4. Untuk`Resources`, pilih Grup Akses Terverifikasi sebagai jenis sumber daya, lalu pilih grup sumber daya yang akan dibagikan.
5. Pilih Lewati ke: Tinjau dan perbarui.
6. Pilih Perbarui berbagi sumber daya.

Untuk informasi selengkapnya, lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.

Menghapus grup Akses Terverifikasi

Setelah selesai dengan grup Akses Terverifikasi, Anda dapat menghapusnya. Anda tidak dapat menghapus grup jika ada titik akhir Akses Terverifikasi terkait.

Untuk menghapus grup Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih grup Akses Terverifikasi.
3. Pilih grup .
4. Pilih Tindakan, Hapus grup Akses Terverifikasi.
5. Saat diminta konfirmasi, masukkan**`delete`**, lalu pilih Hapus.

Untuk menghapus grup Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah delete-verified-access-group.

Titik akhir Akses Terverifikasi

Titik akhir Akses Terverifikasi mewakili aplikasi. Setiap titik akhir dikaitkan dengan grup Akses Terverifikasi dan mewarisi kebijakan akses untuk grup. Anda dapat melampirkan kebijakan endpoint khusus aplikasi secara opsional ke setiap titik akhir.

Daftar Isi

- Jenis titik akhir Akses Terverifikasi
- Cara kerja Akses Terverifikasi dengan berbagi VPCs dan subnet
- Membuat titik akhir penyeimbang beban untuk Akses Terverifikasi
- Membuat titik akhir antarmuka jaringan untuk Akses Terverifikasi
- Buat titik akhir CIDR jaringan untuk Akses Terverifikasi
- Membuat endpoint Layanan Amazon Relational Database Service untuk Akses Terverifikasi
- Izinkan lalu lintas yang berasal dari titik akhir Akses Terverifikasi Anda
- Ubah titik akhir Akses Terverifikasi
- Ubah kebijakan titik akhir Akses Terverifikasi
- Menghapus titik akhir Akses Terverifikasi

Jenis titik akhir Akses Terverifikasi

Berikut ini adalah kemungkinan jenis titik akhir Akses Terverifikasi:

- Load balancer — Permintaan aplikasi dikirim ke penyeimbang beban untuk didistribusikan ke aplikasi Anda. Untuk informasi selengkapnya, lihat Buat titik akhir penyeimbang beban.
- Antarmuka jaringan — Permintaan aplikasi dikirim ke antarmuka jaringan menggunakan protokol dan port yang ditentukan. Untuk informasi selengkapnya, lihat Buat titik akhir antarmuka jaringan.
- CIDR Jaringan — Permintaan aplikasi dikirim ke blok CIDR yang ditentukan. Untuk informasi selengkapnya, lihat Buat titik akhir CIDR jaringan.
- Amazon Relational Database Service (RDS) - Permintaan aplikasi dikirim ke instans RDS, klaster RDS, atau proxy RDS DB. Untuk informasi selengkapnya, lihat Membuat titik akhir Layanan Amazon Relational Database Service.

Cara kerja Akses Terverifikasi dengan berbagi VPCs dan subnet

Berikut ini adalah perilaku terkait subnet VPC bersama:

- Titik akhir Akses Terverifikasi didukung oleh berbagi subnet VPC. Peserta dapat membuat titik akhir Akses Terverifikasi di subnet bersama.
- Peserta yang membuat endpoint akan menjadi pemilik endpoint, dan satu-satunya pihak yang diizinkan untuk memodifikasi endpoint. Pemilik VPC tidak akan diizinkan untuk memodifikasi titik akhir.
- Titik akhir Akses Terverifikasi tidak dapat dibuat di Zona AWS Lokal dan oleh karena itu berbagi melalui Local Zones tidak dimungkinkan.

Untuk informasi selengkapnya, lihat, Bagikan VPC Anda dengan akun lain di Panduan Pengguna Amazon VPC.

Membuat titik akhir penyeimbang beban untuk Akses Terverifikasi

Gunakan prosedur berikut untuk membuat titik akhir penyeimbang beban untuk Akses Terverifikasi. Untuk informasi selengkapnya tentang load balancer, lihat Panduan Pengguna Elastic Load Balancing.

Persyaratan

- Hanya IPv4 lalu lintas yang didukung.
- Koneksi HTTPS yang berumur panjang, seperti WebSocket koneksi, hanya didukung melalui TCP.
- Load balancer harus berupa Application Load Balancer atau Network Load Balancer, dan harus merupakan penyeimbang beban internal.
- Penyeimbang beban dan subnet harus dimiliki oleh virtual private cloud (VPC) yang sama.
- Penyeimbang beban HTTPS dapat menggunakan sertifikat TLS yang ditandatangani sendiri atau publik. Gunakan sertifikat RSA dengan panjang kunci 1.024 atau 2.048.
- Sebelum membuat titik akhir Akses Terverifikasi, Anda harus membuat grup Akses Terverifikasi. Untuk informasi selengkapnya, lihat the section called “Membuat grup Akses Terverifikasi”.
- Anda harus memberikan nama domain untuk aplikasi Anda. Ini adalah nama DNS publik yang akan digunakan pengguna Anda untuk mengakses aplikasi Anda. Anda juga perlu memberikan sertifikat SSL publik dengan CN yang cocok dengan nama domain ini. Anda dapat membuat atau mengimpor sertifikat menggunakan AWS Certificate Manager.

Untuk membuat titik akhir penyeimbang beban menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih Buat titik akhir Akses Terverifikasi.
4. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk titik akhir.
5. Untuk grup Akses Terverifikasi, pilih grup Akses Terverifikasi.
6. Untuk detail Endpoint, lakukan hal berikut:
    a. Untuk Protokol, pilih protokol.
    b. Untuk jenis Lampiran, pilih VPC.
    c. Untuk tipe Endpoint, pilih Load balancer.
    d. (HTTP/HTTPS) Untuk Port, masukkan nomor port. (TCP) Untuk rentang Port, masukkan rentang port dan pilih Tambahkan port.
    e. Untuk Load balancer ARN, pilih load balancer.
    f. Untuk Subnet, pilih subnet. Anda hanya dapat menentukan satu subnet per Availability Zone.
    g. Untuk grup Keamanan, pilih grup keamanan untuk titik akhir. Grup keamanan ini mengontrol lalu lintas masuk dan keluar untuk titik akhir Akses Terverifikasi.
    h. Untuk awalan domain Endpoint, masukkan pengenal kustom untuk menambahkan nama DNS yang dihasilkan Akses Terverifikasi untuk titik akhir.
7. (HTTP/HTTPS) Untuk detail Aplikasi, lakukan hal berikut:
    a. Untuk domain Aplikasi, masukkan nama DNS untuk aplikasi Anda.
    b. Di bawah Sertifikat domain ARN, pilih sertifikat TLS publik.
8. (Opsional) Untuk definisi Kebijakan, masukkan kebijakan Akses Terverifikasi untuk titik akhir.
9. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
10. Pilih Buat titik akhir Akses Terverifikasi.

Untuk membuat titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah create-verified-access-endpoint.

Membuat titik akhir antarmuka jaringan untuk Akses Terverifikasi

Gunakan prosedur berikut untuk membuat titik akhir antarmuka jaringan.

Persyaratan

- Hanya IPv4 lalu lintas yang didukung.
- Antarmuka jaringan harus termasuk dalam virtual private cloud (VPC) yang sama dengan grup keamanan.
- Kami menggunakan IP pribadi pada antarmuka jaringan untuk meneruskan lalu lintas.
- Sebelum membuat titik akhir Akses Terverifikasi, Anda harus membuat grup Akses Terverifikasi. Untuk informasi selengkapnya, lihat the section called “Membuat grup Akses Terverifikasi”.
- Anda harus memberikan nama domain untuk aplikasi Anda. Ini adalah nama DNS publik yang akan digunakan pengguna Anda untuk mengakses aplikasi Anda. Anda juga perlu memberikan sertifikat SSL publik dengan CN yang cocok dengan nama domain ini. Anda dapat membuat atau mengimpor sertifikat menggunakan AWS Certificate Manager.

Untuk membuat endpoint antarmuka jaringan menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih Buat titik akhir Akses Terverifikasi.
4. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk titik akhir.
5. Untuk grup Akses Terverifikasi, pilih grup Akses Terverifikasi.
6. Untuk detail Endpoint, lakukan hal berikut:
    a. Untuk Protokol, pilih protokol.
    b. Untuk jenis Lampiran, pilih VPC.
    c. Untuk tipe Endpoint, pilih Network interface.
    d. (HTTP/HTTPS) Untuk Port, masukkan nomor port. (TCP) Untuk rentang Port, masukkan rentang port dan pilih Tambahkan port.
    e. Untuk antarmuka Jaringan, pilih antarmuka jaringan.
    f. Untuk grup Keamanan, pilih grup keamanan untuk titik akhir. Grup keamanan ini mengontrol lalu lintas masuk dan keluar untuk titik akhir Akses Terverifikasi.

g. Untuk awalan domain Endpoint, masukkan pengenal kustom untuk menambahkan nama DNS yang dihasilkan Akses Terverifikasi untuk titik akhir.

7. (HTTP/HTTPS) Untuk detail Aplikasi, lakukan hal berikut:

   a. Untuk domain Aplikasi, masukkan nama DNS untuk aplikasi Anda.

   b. Di bawah Sertifikat domain ARN, pilih sertifikat TLS publik.

8. (Opsional) Untuk definisi Kebijakan, masukkan kebijakan Akses Terverifikasi untuk titik akhir.
9. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
10. Pilih Buat titik akhir Akses Terverifikasi.

Untuk membuat titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah create-verified-access-endpoint.

Buat titik akhir CIDR jaringan untuk Akses Terverifikasi

Gunakan prosedur berikut untuk membuat titik akhir CIDR jaringan. Misalnya, Anda dapat menggunakan titik akhir CIDR jaringan untuk mengaktifkan akses ke EC2 instance di subnet tertentu melalui port 22 (SSH).

Persyaratan

- Hanya protokol TCP yang didukung.
- Akses Terverifikasi menyediakan catatan DNS untuk setiap alamat IP dalam rentang CIDR yang digunakan oleh sumber daya. Jika Anda menghapus sumber daya, alamat IPnya tidak lagi digunakan dan Akses Terverifikasi menghapus catatan DNS yang sesuai.
- Jika Anda menentukan subdomain kustom, Akses Terverifikasi menyediakan catatan DNS untuk setiap alamat IP yang digunakan di subdomain dan memberi Anda alamat IP server DNS-nya. Anda dapat mengonfigurasi aturan penerusan untuk subdomain Anda untuk mengarah ke server DNS Akses Terverifikasi. Setiap permintaan yang dibuat ke catatan dalam domain diselesaikan oleh server DNS Akses Terverifikasi ke alamat IP sumber daya yang diminta.
- Sebelum membuat titik akhir Akses Terverifikasi, Anda harus membuat grup Akses Terverifikasi. Untuk informasi selengkapnya, lihat the section called “Membuat grup Akses Terverifikasi”.
- Buat titik akhir dan kemudian sambungkan ke aplikasi menggunakan file. Konektivitas Klien

Untuk membuat titik akhir CIDR jaringan menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih Buat titik akhir Akses Terverifikasi.
4. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk titik akhir.
5. Untuk grup Akses Terverifikasi, pilih grup Akses Terverifikasi untuk titik akhir.
6. Untuk detail Endpoint, lakukan hal berikut:
    a. Untuk Protokol, pilih TCP.
    b. Untuk jenis Lampiran, pilih VPC.
    c. Untuk tipe Endpoint, pilih Network CIDR.
    d. Untuk rentang Port, masukkan rentang port dan pilih Tambah port.
    e. Untuk Subnet, pilih subnet.
    f. Untuk grup Keamanan, pilih grup keamanan untuk titik akhir. Grup keamanan ini mengontrol lalu lintas masuk dan keluar untuk titik akhir Akses Terverifikasi.
    g. (Opsional) Untuk awalan domain Endpoint, masukkan pengenal kustom untuk menambahkan nama DNS yang dihasilkan Akses Terverifikasi untuk titik akhir.
7. (Opsional) Untuk definisi Kebijakan, masukkan kebijakan Akses Terverifikasi untuk titik akhir.
8. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
9. Pilih Buat titik akhir Akses Terverifikasi.

Untuk membuat titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah create-verified-access-endpoint.

Membuat endpoint Layanan Amazon Relational Database Service untuk Akses Terverifikasi

Gunakan prosedur berikut untuk membuat titik akhir Amazon Relational Database Service (RDS).

Persyaratan

- Hanya protokol TCP yang didukung.

- Buat instance RDS, cluster RDS, atau proxy RDS DB.
- Sebelum membuat titik akhir Akses Terverifikasi, Anda harus membuat grup Akses Terverifikasi. Untuk informasi selengkapnya, lihat the section called “Membuat grup Akses Terverifikasi”.
- Buat titik akhir dan kemudian sambungkan ke aplikasi menggunakan file. Konektivitas Klien

Untuk membuat endpoint Amazon Relational Database Service menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih Buat titik akhir Akses Terverifikasi.
4. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk titik akhir.
5. Untuk grup Akses Terverifikasi, pilih grup Akses Terverifikasi untuk titik akhir.
6. Untuk detail Endpoint, lakukan hal berikut:

    a. Untuk Protokol, pilih TCP.

    b. Untuk jenis Lampiran, pilih VPC.

    c. Untuk tipe Endpoint, pilih Amazon Relational Database Service (RDS).

    d. Untuk tipe target RDS, lakukan salah satu hal berikut:

        - Pilih contoh RDS, dan kemudian pilih instance RDS dari instance RDS.
        - Pilih klaster RDS, lalu pilih cluster RDS dari cluster RDS.
        - Pilih proxy RDS DB, lalu pilih proxy RDS DB dari proxy RDS DB.

    e. Untuk titik akhir RDS, pilih titik akhir RDS yang terkait dengan sumber daya RDS yang Anda pilih pada langkah sebelumnya.

    f. Untuk Port, masukkan nomor port.

    g. Untuk Subnet, pilih subnet. Anda hanya dapat menentukan satu subnet per Availability Zone.

    h. Untuk grup Keamanan, pilih grup keamanan untuk titik akhir. Grup keamanan ini mengontrol lalu lintas masuk dan keluar untuk titik akhir Akses Terverifikasi.

    i. (Opsional) Untuk awalan domain Endpoint, masukkan pengenal kustom untuk menambahkan nama DNS yang dihasilkan Akses Terverifikasi untuk titik akhir.
7. (Opsional) Untuk definisi Kebijakan, masukkan kebijakan Akses Terverifikasi untuk titik akhir.
8. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.

9. Pilih Buat titik akhir Akses Terverifikasi.

Untuk membuat titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah create-verified-access-endpoint.

Izinkan lalu lintas yang berasal dari titik akhir Akses Terverifikasi Anda

Anda dapat mengonfigurasi grup keamanan untuk aplikasi Anda sehingga memungkinkan lalu lintas yang berasal dari titik akhir Akses Terverifikasi Anda. Anda melakukannya dengan menambahkan aturan masuk yang menentukan grup keamanan untuk titik akhir sebagai sumbernya. Kami menyarankan Anda menghapus aturan masuk tambahan, sehingga aplikasi Anda hanya menerima lalu lintas dari titik akhir Akses Terverifikasi Anda.

Kami menyarankan Anda untuk mempertahankan aturan keluar yang ada.

Untuk memperbarui aturan grup keamanan untuk aplikasi Anda menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih titik akhir Akses Terverifikasi, temukan grup Keamanan IDs di tab Detail, dan salin ID grup keamanan untuk titik akhir Anda.
4. Pada panel navigasi, pilih Grup keamanan.
5. Pilih kotak centang untuk grup keamanan yang terkait dengan target Anda, lalu pilih Tindakan, Edit aturan masuk.
6. Untuk menambahkan aturan grup keamanan yang mengizinkan lalu lintas yang berasal dari titik akhir Akses Terverifikasi, lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Jenis, pilih Semua lalu lintas atau lalu lintas tertentu yang akan diizinkan.
   c. Untuk Sumber, pilih Kustom dan tempel ID grup keamanan untuk titik akhir Anda.
7. (Opsional) Untuk mewajibkan lalu lintas hanya berasal dari titik akhir Akses Terverifikasi Anda, hapus aturan grup keamanan masuk lainnya.
8. Pilih Simpan aturan.

Untuk memperbarui aturan grup keamanan untuk aplikasi Anda menggunakan AWS CLI

Gunakan describe-verified-access-endpointsperintah untuk mendapatkan ID grup keamanan dan kemudian gunakan authorize-security-group-ingressperintah untuk menambahkan aturan masuk.

Ubah titik akhir Akses Terverifikasi

Gunakan prosedur berikut untuk mengubah titik akhir Akses Terverifikasi.

Untuk mengubah titik akhir Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih titik akhir.
4. Pilih Tindakan, Ubah titik akhir Akses Terverifikasi.
5. Ubah detail titik akhir sesuai kebutuhan.
6. Pilih Ubah titik akhir Akses Terverifikasi.

Untuk mengubah titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-endpoint.

Ubah kebijakan titik akhir Akses Terverifikasi

Gunakan prosedur berikut untuk mengubah kebijakan titik akhir Akses Terverifikasi. Setelah Anda melakukan perubahan, dibutuhkan beberapa menit sebelum diterapkan.

Untuk mengubah kebijakan titik akhir Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih titik akhir.
4. Pilih Tindakan, Ubah kebijakan titik akhir Akses Terverifikasi.
5. (Opsional) Aktifkan atau nonaktifkan Aktifkan kebijakan sesuai kebutuhan.
6. (Opsional) Untuk Kebijakan, masukkan kebijakan Akses Terverifikasi untuk diterapkan pada titik akhir.

7. Pilih Ubah kebijakan titik akhir Akses Terverifikasi.

Untuk mengubah kebijakan titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-endpoint-policy.

Menghapus titik akhir Akses Terverifikasi

Setelah selesai dengan titik akhir Akses Terverifikasi, Anda dapat menghapusnya.

Untuk menghapus titik akhir Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih titik akhir Akses Terverifikasi.
3. Pilih titik akhir.
4. Pilih Tindakan, Hapus titik akhir Akses Terverifikasi.
5. Saat diminta konfirmasi, masukkan **`delete`**, lalu pilih Hapus.

Untuk menghapus titik akhir Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah delete-verified-access-endpoint.

Data kepercayaan dikirim ke Akses Terverifikasi dari penyedia kepercayaan

Data kepercayaan adalah data yang dikirim Akses Terverifikasi AWS dari penyedia kepercayaan. Data kepercayaan juga disebut sebagai “klaim pengguna” atau “konteks kepercayaan.” Data umumnya mencakup informasi tentang pengguna atau perangkat. Contoh data kepercayaan termasuk email pengguna, keanggotaan grup, versi sistem operasi perangkat, status keamanan perangkat, dan sebagainya. Informasi yang dikirim bervariasi tergantung pada penyedia kepercayaan, jadi Anda harus merujuk ke dokumentasi penyedia kepercayaan Anda untuk daftar data kepercayaan yang lengkap dan diperbarui.

Namun, dengan menggunakan kemampuan pencatatan Akses Terverifikasi, Anda juga dapat melihat data kepercayaan apa yang dikirim dari penyedia kepercayaan Anda. Ini dapat berguna saat menentukan kebijakan yang mengizinkan atau menolak akses ke aplikasi Anda. Untuk informasi tentang menyertakan konteks kepercayaan di log Anda, lihatMengaktifkan atau menonaktifkan konteks kepercayaan Akses Terverifikasi.

Bagian ini berisi contoh data kepercayaan dan contoh untuk membantu Anda memulai penulisan kebijakan. Informasi yang diberikan di sini dimaksudkan untuk tujuan ilustrasi saja dan bukan sebagai referensi resmi.

Daftar Isi

- Konteks default untuk data kepercayaan Akses Terverifikasi
- AWS IAM Identity Center konteks untuk data kepercayaan Akses Terverifikasi
- Konteks penyedia kepercayaan pihak ketiga untuk data kepercayaan Akses Terverifikasi
- Klaim pengguna lulus dan verifikasi tanda tangan di Akses Terverifikasi

Konteks default untuk data kepercayaan Akses Terverifikasi

Akses Terverifikasi AWS menyertakan beberapa elemen tentang permintaan saat ini secara default di semua evaluasi Cedar terlepas dari penyedia kepercayaan Anda yang dikonfigurasi. Anda dapat menulis kebijakan yang mengevaluasi data jika Anda memilih.

Berikut ini adalah contoh data yang termasuk dalam evaluasi.

Contoh

- Permintaan HTTP
- Aliran TCP

Permintaan HTTP

Saat kebijakan dievaluasi, Akses Terverifikasi menyertakan data tentang permintaan HTTP saat ini dalam konteks Cedar di bawah kunci. `context.http_request`

```
{
    "title": "HTTP Request data included by Verified Access",
    "type": "object",
    "properties": {
        "http_method": {
            "type": "string",
            "description": "The HTTP method",
            "example": "GET"
        },
        "hostname": {
            "type": "string",
            "description": "The host subcomponent of the authority component of the
 URI",
            "example": "example.com"
        },
        "path": {
            "type": "string",
            "description": "The path component of the URI",
            "example": "app/images"
        },
        "query": {
            "type": "string",
            "description": "The query component of the URI",
            "example": "value1=1&value2=2"
        },
        "x_forwarded_for": {
            "type": "string",
            "description": "The value of the X-Forwarded-For request header",
            "example": "17.7.7.1"
        },
        "port": {
           "type": "integer",
           "description": "The endpoint port",
           "example": 443
```

```
        },
        "user_agent": {
            "type": "string",
            "description": "The value of the User-Agent request header",
            "example": "Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:47.0)
 Gecko/20100101 Firefox/47.0"
        },
        "client_ip": {
            "type": "string",
            "description": "The IP address connecting to the endpoint",
            "example": "15.248.6.6"
        }
    }
}
```

Contoh kebijakan

Berikut ini adalah contoh kebijakan Cedar yang menggunakan data permintaan HTTP.

```
forbid(principal, action, resource) when {
   context.http_request.http_method == "POST"
   && !(context.identity.roles.contains("Administrator"))
 };
```

Aliran TCP

Saat kebijakan dievaluasi, Akses Terverifikasi menyertakan data tentang aliran TCP saat ini dalam konteks Cedar di bawah kunci. `context.tcp_flow`

```
{
    "title": "TCP flow data included by Verified Access",
    "type": "object",
    "properties": {
        "destination_ip": {
            "type": "string",
            "description": "The IP address of the target",
            "example": "192.100.1.3"
        },
        "destination_port": {
            "type": "string",
            "description": "The target port",
            "example": 22
        },
```

```
        "client_ip": {
            "type": "string",
            "description": "The IP address connecting to the endpoint",
            "example": "172.154.16.9"
        }
    }
}
```

AWS IAM Identity Center konteks untuk data kepercayaan Akses Terverifikasi

Saat kebijakan dievaluasi, jika Anda mendefinisikan AWS IAM Identity Center sebagai penyedia kepercayaan, Akses Terverifikasi AWS sertakan data kepercayaan dalam konteks Cedar di bawah kunci yang Anda tentukan sebagai “Nama Referensi Kebijakan” pada konfigurasi penyedia kepercayaan. Anda dapat menulis kebijakan yang mengevaluasi terhadap data kepercayaan jika Anda memilih.

> **Note**
>
> Kunci konteks untuk penyedia kepercayaan Anda berasal dari nama referensi kebijakan yang Anda konfigurasikan saat membuat penyedia kepercayaan. Misalnya, jika Anda mengonfigurasi nama referensi kebijakan sebagai “idp123", kunci konteksnya adalah “context.idp123". Periksa apakah Anda menggunakan kunci konteks yang benar saat membuat kebijakan.

Skema JSON berikut menunjukkan data mana yang termasuk dalam evaluasi.

```
{
   "title": "AWS IAM Identity Center context specification",
   "type": "object",
   "properties": {
     "user": {
       "type": "object",
       "properties": {
         "user_id": {
           "type": "string",
           "description": "a unique user id generated by AWS IdC"
         },
```

```
        "user_name": {
          "type": "string",
          "description": "username provided in the directory"
        },
        "email": {
          "type": "object",
          "properties": {
            "address": {
              "type": "email",
              "description": "email address associated with the user"
            },
            "verified": {
              "type": "boolean",
              "description": "whether the email address has been verified by AWS IdC"
            }
          }
        }
      },
      "groups": {
        "type": "object",
        "description": "A list of groups the user is a member of",
        "patternProperties": {
          "^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]{4}-[a-fA-F0-9]
{12}$": {
            "type": "object",
            "description": "The Group ID of the group",
            "properties": {
              "group_name": {
                "type": "string",
                "description": "The customer-provided name of the group"
              }
            }
          }
        }
      }
    }
  }
```

Berikut ini adalah contoh kebijakan yang mengevaluasi terhadap data kepercayaan yang diberikan oleh AWS IAM Identity Center.

```
permit(principal, action, resource) when {
```

```
  context.idc.user.email.verified == true
  // User is in the "sales" group with specific ID
  && context.idc.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
};
```

> **Note**
>
> Karena nama grup dapat diubah, IAM Identity Center mengacu pada grup yang menggunakan ID grup mereka. Ini membantu menghindari melanggar pernyataan kebijakan saat mengubah nama grup.

Konteks penyedia kepercayaan pihak ketiga untuk data kepercayaan Akses Terverifikasi

Bagian ini menjelaskan data kepercayaan yang diberikan Akses Terverifikasi AWS oleh penyedia kepercayaan pihak ketiga.

> **Note**
>
> Kunci konteks untuk penyedia kepercayaan Anda berasal dari nama referensi kebijakan yang Anda konfigurasikan saat membuat penyedia kepercayaan. Misalnya, jika Anda mengonfigurasi nama referensi kebijakan sebagai “idp123", kunci konteksnya adalah “context.idp123". Pastikan Anda menggunakan kunci konteks yang benar saat membuat kebijakan.

Daftar Isi

- Ekstensi browser
- Jamf
- CrowdStrike
- JumpCloud

Ekstensi browser

Jika Anda berencana untuk memasukkan konteks kepercayaan perangkat ke dalam kebijakan akses Anda, maka Anda akan memerlukan ekstensi browser Akses AWS Terverifikasi, atau ekstensi browser mitra lain. Akses Terverifikasi saat ini mendukung browser Google Chrome dan Mozilla Firefox.

Saat ini kami mendukung tiga penyedia kepercayaan perangkat: Jamf (yang mendukung perangkat macOS) CrowdStrike , (yang mendukung perangkat Windows 11 dan Windows 10), JumpCloud dan (yang mendukung Windows dan macOS).

- Jika Anda menggunakan data kepercayaan Jamf dalam kebijakan Anda, pengguna Anda harus mengunduh dan menginstal ekstensi Akses Terverifikasi AWS browser dari toko web Chrome atau situs Add-on Firefox di perangkat mereka.
- Jika Anda menggunakan data CrowdStrikekepercayaan dalam kebijakan Anda, pertama-tama pengguna Anda harus menginstal Host Pesan Akses Terverifikasi AWS Asli (tautan unduhan langsung). Komponen ini diperlukan untuk mendapatkan data kepercayaan dari CrowdStrike agen yang berjalan di perangkat pengguna. Kemudian, setelah menginstal komponen ini, pengguna harus menginstal ekstensi Akses Terverifikasi AWS browser dari toko web Chrome atau situs Add-on Firefox di perangkat mereka.
- Jika Anda menggunakan JumpCloud, pengguna Anda harus memiliki ekstensi JumpCloud browser dari toko web Chrome atau situs Add-on Firefox yang diinstal pada perangkat mereka.

Jamf

Jamf adalah penyedia kepercayaan pihak ketiga. Ketika kebijakan dievaluasi, jika Anda mendefinisikan Jamf sebagai penyedia kepercayaan, Akses Terverifikasi menyertakan data kepercayaan dalam konteks Cedar di bawah kunci yang Anda tentukan sebagai “Nama Referensi Kebijakan” pada konfigurasi penyedia kepercayaan. Anda dapat menulis kebijakan yang mengevaluasi terhadap data kepercayaan jika Anda memilih. Skema JSON berikut menunjukkan data mana yang termasuk dalam evaluasi.

Untuk informasi selengkapnya tentang penggunaan Jamf dengan Akses Terverifikasi, lihat Mengintegrasikan AWS Verified Access dengan Jamf Device Identity di situs web Jamf.

```
{
    "title": "Jamf device data specification",
    "type": "object",
```

```
   "properties": {
       "iss": {
           "type": "string",
           "description": "\"Issuer\" - the Jamf customer ID"
       },
       "iat": {
           "type": "integer",
           "description": "\"Issued at Time\" - a unixtime (seconds since epoch) value
of when the device information data was generated"
       },
       "exp": {
           "type": "integer",
           "description": "\"Expiration\" - a unixtime (seconds since epoch) value for
when this device information is no longer valid"
       },
       "sub": {
           "type": "string",
           "description": "\"Subject\" - either the hardware UID or a value generated
based on device location"
       },
       "groups": {
           "type": "array",
           "description": "Group IDs from UEM connector sync",
           "items": {
               "type": "string"
           }
       },
       "risk": {
           "type": "string",
           "enum": [
               "HIGH",
               "MEDIUM",
               "LOW",
               "SECURE",
               "NOT_APPLICABLE"
           ],
           "description": "a Jamf-reported level of risk associated with the device."
       },
       "osv": {
           "type": "string",
           "description": "The version of the OS that is currently running, in Apple
version number format (https://support.apple.com/en-us/HT201260)"
       }
   }
```

```
}
```

Berikut ini adalah contoh kebijakan yang mengevaluasi terhadap data kepercayaan yang diberikan oleh Jamf.

```
permit(principal, action, resource) when {
   context.jamf.risk == "LOW"
};
```

Cedar menyediakan `.contains()` fungsi yang berguna untuk membantu dengan enum seperti skor risiko Jamf.

```
permit(principal, action, resource) when {
   ["LOW", "SECURE"].contains(context.jamf.risk)
};
```

CrowdStrike

CrowdStrike adalah penyedia kepercayaan pihak ketiga. Ketika kebijakan dievaluasi, jika Anda mendefinisikan CrowdStrike sebagai penyedia kepercayaan, Akses Terverifikasi menyertakan data kepercayaan dalam konteks Cedar di bawah kunci yang Anda tentukan sebagai “Nama Referensi Kebijakan” pada konfigurasi penyedia kepercayaan. Anda dapat menulis kebijakan yang mengevaluasi terhadap data kepercayaan jika Anda memilih. Skema JSON berikut menunjukkan data mana yang termasuk dalam evaluasi.

Untuk informasi selengkapnya tentang penggunaan CrowdStrike dengan Akses Terverifikasi, lihat Mengamankan aplikasi pribadi dengan CrowdStrike dan Akses Terverifikasi AWS di GitHub situs web.

```
{
  "title": "CrowdStrike device data specification",
  "type": "object",
  "properties": {
    "assessment": {
      "type": "object",
      "description": "Data about CrowdStrike's assessment of the device",
      "properties": {
        "overall": {
          "type": "integer",
          "description": "A single metric, between 1-100, that accounts as a weighted
 average of the OS and and Sensor Config scores"
```

```
        },
        "os": {
          "type": "integer",
          "description": "A single metric, between 1-100, that accounts for the OS-
specific settings monitored on the host"
        },
        "sensor_config": {
          "type": "integer",
          "description": "A single metric, between 1-100, that accounts for the
 different sensor policies monitored on the host"
        },
        "version": {
          "type": "string",
          "description": "The version of the scoring algorithm being used"
        }
      }
    },
    "cid": {
      "type": "string",
      "description": "Customer ID (CID) unique to the customer's environment"
    },
    "exp": {
      "type": "integer",
      "description": "unixtime, The expiration time of the token"
    },
    "iat": {
      "type": "integer",
      "description": "unixtime, The issued time of the token"
    },
    "jwk_url": {
      "type": "string",
      "description": "URL that details the JWT signing"
    },
    "platform": {
      "type": "string",
      "enum": ["Windows 10", "Windows 11", "macOS"],
      "description": "Operating system of the endpoint"
    },
    "serial_number": {
      "type": "string",
      "description": "The serial number of the device derived by unique system
 information"
    },
    "sub": {
```

```
      "type": "string",
      "description": "Unique CrowdStrike Agent ID (AID) of machine"
    },
    "typ": {
      "type": "string",
      "enum": ["crowdstrike-zta+jwt"],
      "description": "Generic name for this JWT media. Client MUST reject any other
 type"
    }
  }
}
```

Berikut ini adalah contoh kebijakan yang mengevaluasi terhadap data kepercayaan yang diberikan oleh CrowdStrike.

```
permit(principal, action, resource) when {
   context.crowdstrike.assessment.overall > 50
};
```

JumpCloud

JumpCloud adalah penyedia kepercayaan pihak ketiga. Ketika kebijakan dievaluasi, jika Anda mendefinisikan JumpCloud sebagai penyedia kepercayaan, Akses Terverifikasi menyertakan data kepercayaan dalam konteks Cedar di bawah kunci yang Anda tentukan sebagai “Nama Referensi Kebijakan” pada konfigurasi penyedia kepercayaan. Anda dapat menulis kebijakan yang mengevaluasi terhadap data kepercayaan jika Anda memilih. Skema JSON berikut menunjukkan data mana yang termasuk dalam evaluasi.

Untuk informasi selengkapnya tentang penggunaan JumpCloud dengan Akses AWS Terverifikasi, lihat Mengintegrasikan JumpCloud dan Akses AWS Terverifikasi di JumpCloud situs web.

```
{
  "title": "JumpCloud device data specification",
  "type": "object",
  "properties": {
    "device": {
      "type": "object",
      "description": "Properties of the device",
      "properties": {
        "is_managed": {
          "type": "boolean",
```

```
          "description": "Boolean to indicate if the device is under management"
        }
      }
    },
    "exp": {
      "type": "integer",
      "description": "Expiration. Unixtime of the token's expiration."
    },
    "durt_id": {
      "type": "string",
      "description": "Device User Refresh Token ID. Unique ID that represents the
 device + user."
    },
    "iat": {
      "type": "integer",
      "description": "Issued At. Unixtime of the token's issuance."
    },
    "iss": {
      "type": "string",
      "description": "Issuer. This will be 'go.jumpcloud.com'"
    },
    "org_id": {
      "type": "string",
      "description": "The JumpCloud Organization ID"
    },
    "sub": {
      "type": "string",
      "description": "Subject. The managed JumpCloud user ID on the device."
    },
    "system": {
      "type": "string",
      "description": "The JumpCloud system ID"
    }
  }
}
```

Berikut ini adalah contoh kebijakan yang mengevaluasi terhadap konteks kepercayaan yang diberikan oleh JumpCloud.

```
permit(principal, action, resource) when {
   context.jumpcloud.org_id == 'Unique_organization_identifier'
};
```

Klaim pengguna lulus dan verifikasi tanda tangan di Akses Terverifikasi

Setelah Akses Terverifikasi AWS instans berhasil mengautentikasi pengguna, instans mengirimkan klaim pengguna yang diterima dari iDP ke titik akhir Akses Terverifikasi. Klaim pengguna ditandatangani sehingga aplikasi dapat memverifikasi tanda tangan dan juga memverifikasi bahwa klaim dikirim oleh Akses Terverifikasi. Selama proses ini, header HTTP berikut ditambahkan:

`x-amzn-ava-user-context`

Header ini berisi klaim pengguna dalam format token web JSON (JWT). Format JWT mencakup header, payload, dan tanda tangan yang dikodekan URL base64. Akses Terverifikasi menggunakan ES384 (algoritma tanda tangan ECDSA menggunakan algoritma hash SHA-384) untuk menghasilkan tanda tangan JWT.

Aplikasi dapat menggunakan klaim ini untuk personalisasi atau pengalaman khusus pengguna lainnya. Pengembang aplikasi harus mendidik diri mereka sendiri mengenai tingkat keunikan dan verifikasi setiap klaim yang diberikan oleh penyedia identitas sebelum digunakan. Secara umum, `sub` klaim adalah cara terbaik untuk mengidentifikasi pengguna tertentu.

Daftar Isi

- Contoh: Menandatangani JWT untuk klaim pengguna OIDC
- Contoh: Menandatangani JWT untuk klaim pengguna IAM Identity Center
- Kunci publik
- Contoh: Mengambil dan mendekode JWT

Contoh: Menandatangani JWT untuk klaim pengguna OIDC

Contoh berikut menunjukkan seperti apa header dan payload untuk klaim pengguna OIDC dalam format JWT.

Contoh header:

```
{
   "alg": "ES384",
   "kid": "12345678-1234-1234-1234-123456789012",
```

```
   "signer": "arn:aws:ec2:us-east-1:123456789012:verified-access-instance/vai-
abc123xzy321a2b3c",
   "iss": "OIDC Issuer URL",
   "exp": "expiration" (120 secs)
}
```

Contoh muatan:

```
{
   "sub": "xyzsubject",
   "email": "xxx@amazon.com",
   "email_verified": true,
   "groups": [
      "Engineering",
      "finance"
   ],
   "additional_user_context": {
      "aud": "xxx",
      "exp": 1000000000,
      "groups": [
         "group-id-1",
         "group-id-2"
      ],
      "iat": 1000000000,
      "iss": "https://oidc-tp.com/",
      "sub": "xyzsubject",
      "ver": "1.0"
   }
}
```

Contoh: Menandatangani JWT untuk klaim pengguna IAM Identity Center

Contoh berikut menunjukkan seperti apa header dan payload untuk klaim pengguna IAM Identity Center dalam format JWT.

> **Note**
>
> Untuk IAM Identity Center, hanya informasi pengguna yang akan dimasukkan dalam klaim.

Contoh header:

```
{
   "alg": "ES384",
   "kid": "12345678-1234-1234-1234-123456789012",
   "signer": "arn:aws:ec2:us-east-1:123456789012:verified-access-instance/vai-
abc123xzy321a2b3c",
   "iss": "arn:aws:ec2:us-east-1:123456789012:verified-access-trust-provider/vatp-
abc123xzy321a2b3c",
   "exp": "expiration" (120 secs)
}
```

Contoh muatan:

```
{
    "user": {
        "user_id": "f478d4c8-a001-7064-6ea6-12423523",
        "user_name": "test-123",
        "email": {
            "address": "test@amazon.com",
            "verified": false
        }
    }
}
```

Kunci publik

Karena instans Akses Terverifikasi tidak mengenkripsi klaim pengguna, sebaiknya Anda mengonfigurasi titik akhir Akses Terverifikasi untuk menggunakan HTTPS. Jika Anda mengonfigurasi titik akhir Akses Terverifikasi untuk menggunakan HTTP, pastikan untuk membatasi lalu lintas ke titik akhir menggunakan grup keamanan.

Untuk memastikan keamanan, Anda harus memverifikasi tanda tangan sebelum melakukan otorisasi berdasarkan klaim, dan memvalidasi bahwa `signer` bidang di header JWT berisi ARN instance Akses Terverifikasi yang diharapkan.

Untuk mendapatkan kunci publik, dapatkan ID kunci dari header JWT dan gunakan untuk mencari kunci publik dari titik akhir.

Titik akhir untuk masing-masing Wilayah AWS adalah sebagai berikut:

`https://public-keys.prod.verified-access.<region>.amazonaws.com/<key-id>`

Contoh: Mengambil dan mendekode JWT

Contoh kode berikut menunjukkan cara mendapatkan ID kunci, kunci publik, dan payload di Python 3.9.

```
import jwt
import requests
import base64
import json

# Step 1: Validate the signer
expected_verified_access_instance_arn = 'arn:aws:ec2:region-code:account-id:verified-
access-instance/verified-access-instance-id'

encoded_jwt = headers.dict['x-amzn-ava-user-context']
jwt_headers = encoded_jwt.split('.')[0]
decoded_jwt_headers = base64.b64decode(jwt_headers)
decoded_jwt_headers = decoded_jwt_headers.decode("utf-8")
decoded_json = json.loads(decoded_jwt_headers)
received_verified_access_instance_arn = decoded_json['signer']

assert expected_verified_access_instance_arn == received_verified_access_instance_arn,
 "Invalid Signer"

# Step 2: Get the key id from JWT headers (the kid field)
kid = decoded_json['kid']

# Step 3: Get the public key from regional endpoint
url = 'https://public-keys.prod.verified-access.' + region + '.amazonaws.com/' + kid
req = requests.get(url)
pub_key = req.text

# Step 4: Get the payload
payload = jwt.decode(encoded_jwt, pub_key, algorithms=['ES384'])
```

Kebijakan Akses Terverifikasi

Akses Terverifikasi AWS kebijakan memungkinkan Anda untuk menentukan aturan untuk mengakses aplikasi yang dihosting. AWS Mereka ditulis dalam Cedar, bahasa AWS kebijakan. Menggunakan Cedar, Anda dapat membuat kebijakan yang dievaluasi terhadap data kepercayaan yang dikirim dari identitas atau penyedia kepercayaan berbasis perangkat yang Anda konfigurasikan untuk digunakan dengan Akses Terverifikasi.

Untuk informasi lebih rinci tentang bahasa kebijakan Cedar, lihat Panduan Referensi Cedar.

Saat membuat grup Akses Terverifikasi atau membuat titik akhir Akses Terverifikasi, Anda memiliki opsi untuk menentukan kebijakan Akses Terverifikasi. Anda dapat membuat grup atau titik akhir tanpa menentukan kebijakan Akses Terverifikasi, tetapi semua permintaan akses akan diblokir hingga Anda menentukan kebijakan. Atau, Anda dapat menambahkan atau mengubah kebijakan pada grup atau titik akhir Akses Terverifikasi yang ada setelah dibuat.

Daftar Isi

- Struktur pernyataan kebijakan Akses Terverifikasi
- Operator bawaan untuk kebijakan Akses Terverifikasi
- Evaluasi kebijakan Akses Terverifikasi
- Logika kebijakan Akses Terverifikasi hubung singkat
- Kebijakan contoh Akses Terverifikasi
- Asisten kebijakan Akses Terverifikasi

Struktur pernyataan kebijakan Akses Terverifikasi

Tabel berikut menunjukkan struktur kebijakan Akses Terverifikasi.

Komponen	Sintaks
efek	`permit | forbid`
cakupan	`(principal, action, resource)`
klausa kondisi	`when {`

Komponen	Sintaks
	context.*policy-reference-name* .*attribute-name* };

Komponen kebijakan

Kebijakan Akses Terverifikasi berisi komponen-komponen berikut:

- Efek - Baik `permit` (izinkan) atau `forbid` (tolak) akses.
- Lingkup — Prinsip, tindakan, dan sumber daya yang efeknya berlaku. Anda dapat membiarkan ruang lingkup di Cedar tidak terdefinisi dengan tidak mengidentifikasi prinsip, tindakan, atau sumber daya tertentu. Dalam hal ini, kebijakan berlaku untuk semua prinsip, tindakan, dan sumber daya yang mungkin.
- Klausul kondisi — Konteks di mana efek berlaku.

> **Important**
>
> Untuk Akses Terverifikasi, kebijakan diungkapkan sepenuhnya dengan mengacu pada data kepercayaan dalam klausul kondisi. Ruang lingkup kebijakan harus selalu tetap tidak terdefinisi. Anda kemudian dapat menentukan akses menggunakan identitas dan konteks kepercayaan perangkat dalam klausa kondisi.

Komentar

Anda dapat memasukkan komentar dalam Akses Terverifikasi AWS kebijakan Anda. Komentar didefinisikan sebagai baris yang dimulai dengan `//` dan diakhiri dengan karakter baris baru.

Contoh berikut menunjukkan komentar dalam kebijakan.

```
// grants access to users in a specific domain using trusted devices
permit(principal, action, resource)
when {
  // the user's email address is in the @example.com domain
  context.idc.user.email.address.contains("@example.com")
```

```
  // Jamf thinks the user's computer is low risk or secure.
  && ["LOW", "SECURE"].contains(context.jamf.risk)
};
```

Beberapa klausa

Anda dapat menggunakan lebih dari satu klausa kondisi dalam pernyataan kebijakan menggunakan && operator.

```
permit(principal,action,resource)
when{
 context.policy-reference-name.attribute1 &&
 context.policy-reference-name.attribute2
};
```

Untuk contoh tambahan, lihat Kebijakan contoh Akses Terverifikasi.

Karakter yang dipesan

Contoh berikut menunjukkan cara menulis kebijakan jika properti context menggunakan : (titik koma), yang merupakan karakter cadangan dalam bahasa kebijakan.

```
permit(principal, action, resource)
when {
    context.policy-reference-name["namespace:groups"].contains("finance")
};
```

Operator bawaan untuk kebijakan Akses Terverifikasi

Saat membuat konteks Akses Terverifikasi AWS kebijakan menggunakan berbagai kondisi, seperti yang dibahas dalamStruktur pernyataan kebijakan Akses Terverifikasi, Anda dapat menggunakan && operator untuk menambahkan kondisi tambahan. Ada juga banyak operator bawaan lainnya yang dapat Anda gunakan untuk menambahkan kekuatan ekspresif tambahan pada kondisi kebijakan Anda. Tabel berikut berisi semua operator bawaan untuk referensi.

Operator	Jenis dan kelebihan beban	Deskripsi
!	Boolean → Boolean	Logis tidak.

Operator	Jenis dan kelebihan beban	Deskripsi
==	apa saja → apa saja	Kesetaraan. Bekerja pada argumen jenis apa pun, bahkan jika tipenya tidak cocok. Nilai dari berbagai jenis tidak pernah sama satu sama lain.
!=	apa saja → apa saja	Ketimpangan; kebalikan dari kesetaraan (lihat di atas).
<	(panjang, panjang) → Boolean	Bilangan bulat panjang kurang dari.
<=	(panjang, panjang) → Boolean	Bilangan bulat panjang less-than-or-equal -ke.
>	(panjang, panjang) → Boolean	Bilangan bulat panjang lebih besar dari.
>=	(panjang, panjang) → Boolean	Bilangan bulat panjang greater-than-or-equal -ke.
in	(entitas, entitas) → Boolean	Keanggotaan hierarki (refleksi f: A dalam A selalu benar).
	(entitas, set (entitas)) → Boolean	Keanggotaan hierarki: A di [B, C,...] benar jika (A dan B) || (A dalam C) ||... kesalahan jika himpunan berisi non-entitas.
&&	(Boolean, Boolean) → Boolean	Logis dan (hubungan arus pendek).
||	(Boolean, Boolean) → Boolean	Logis atau (hubungan arus pendek).
.ada ()	entitas → Boolean	Keberadaan entitas.

Operator	Jenis dan kelebihan beban	Deskripsi
memiliki	(entitas, atribut) → Boolean	Operator infix. `e has f`menguji apakah catatan atau entitas `e` memiliki pengikata n untuk atribut`f`. Mengembal ikan `false` jika `e` tidak ada atau jika `e` memang ada tetapi tidak memiliki atribut`f`. Atribut dapat dinyatakan sebagai pengidentifikasi atau string literal.
suka	(string, string) → Boolean	Operator infix. `t like p`memeriksa apakah teks `t` cocok dengan polap, yang mungkin termasuk karakter wildcard `*` yang cocok dengan 0 atau lebih dari karakter apa pun. Untuk mencocokkan karakter bintang literal`t`, Anda dapat menggunakan urutan karakter lolos khusus `\*` dip.
.berisi ()	(set, apa saja) → Boolean	Tetapkan keanggotaan (adalah B elemen A).
.containsAll ()	(set, atur) → Boolean	Tes jika set A berisi semua elemen dalam himpunan B.
.containsAny ()	(set, atur) → Boolean	Pengujian jika set A berisi salah satu elemen dalam himpunan B.

Evaluasi kebijakan Akses Terverifikasi

Dokumen kebijakan adalah satu set dari satu atau lebih pernyataan kebijakan (`permit`atau `forbid` pernyataan). Kebijakan ini berlaku jika klausa kondisional (`when`pernyataan) benar. Agar dokumen kebijakan memungkinkan akses, setidaknya satu kebijakan izin dalam dokumen harus berlaku dan tidak ada kebijakan larangan yang dapat diterapkan. Jika tidak ada kebijakan izin yang berlaku dan/atau satu atau lebih kebijakan larangan berlaku, maka dokumen kebijakan tersebut menolak akses. Jika Anda telah menetapkan dokumen kebijakan untuk grup Akses Terverifikasi dan titik akhir Akses Terverifikasi, kedua dokumen harus mengizinkan akses. Jika Anda belum menetapkan dokumen kebijakan untuk titik akhir Akses Terverifikasi, hanya kebijakan grup Akses Terverifikasi yang perlu diakses.

Akses Terverifikasi AWS memvalidasi sintaks saat Anda membuat kebijakan, tetapi tidak memvalidasi data yang Anda masukkan ke dalam klausa bersyarat.

Logika kebijakan Akses Terverifikasi hubung singkat

Anda mungkin ingin menulis Akses Terverifikasi AWS kebijakan yang mengevaluasi data yang mungkin atau mungkin tidak ada dalam konteks tertentu. Jika Anda mereferensikan data dalam konteks yang tidak ada, Cedar akan menghasilkan kesalahan dan mengevaluasi kebijakan untuk menolak akses, terlepas dari maksud Anda. Misalnya, ini akan menghasilkan penolakan, karena `fake_provider` dan `bogus_key` tidak ada dalam konteks ini.

```
permit(principal, action, resource) when {
  context.fake_provider.bogus_key > 42
};
```

Untuk menghindari situasi ini, Anda dapat memeriksa untuk melihat apakah ada kunci dengan menggunakan `has` operator. Jika `has` operator mengembalikan false, evaluasi lebih lanjut dari pernyataan berantai berhenti, dan Cedar tidak menghasilkan kesalahan saat mencoba mereferensikan item yang tidak ada.

```
permit(principal, action, resource) when {
  context.identity.user has "some_key" && context.identity.user.some_key > 42
};
```

Ini sangat berguna ketika menentukan kebijakan yang mereferensikan dua penyedia kepercayaan yang berbeda.

```
permit(principal, action, resource) when {
  // user is in an allowed group
  context.aws_idc.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
  &&(
    (
      // if CrowdStrike data is present,
      // permit if CrowdStrike's overall assessment is over 50
      context has "crowdstrike" && context.crowdstrike.assessment.overall > 50
    )
    ||
    (
      // if Jamf data is present,
      // permit if Jamf's risk score is acceptable
      context has "jamf" && ["LOW", "NOT_APPLICABLE", "MEDIUM",
 "SECURE"].contains(context.jamf.risk)
    )
  )
};
```

Kebijakan contoh Akses Terverifikasi

Anda dapat menggunakan kebijakan Akses Terverifikasi untuk memberikan akses ke aplikasi Anda kepada pengguna dan perangkat tertentu.

Contoh kebijakan

- Contoh 1: Berikan akses ke grup di Pusat Identitas IAM
- Contoh 2: Berikan akses ke grup di penyedia pihak ketiga
- Contoh 3: Berikan akses menggunakan CrowdStrike
- Contoh 4: Izinkan atau tolak alamat IP tertentu

Contoh 1: Berikan akses ke grup di Pusat Identitas IAM

Saat menggunakan AWS IAM Identity Center, lebih baik merujuk ke grup dengan menggunakan mereka IDs. Ini membantu menghindari pelanggaran pernyataan kebijakan jika Anda mengubah nama grup.

Kebijakan contoh berikut hanya mengizinkan akses ke pengguna dalam grup yang ditentukan dengan alamat email terverifikasi. ID grup adalah c242c5b0-6081-1845-6fa8-6e0d9513c107.

```
permit(principal,action,resource)
when {
    context.policy-reference-name.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
    && context.policy-reference-name.user.email.verified == true
};
```

Contoh kebijakan berikut mengizinkan akses hanya ketika pengguna berada dalam grup yang ditentukan, pengguna memiliki alamat email terverifikasi, dan skor risiko perangkat Jamf adalah`LOW`.

```
permit(principal,action,resource)
when {
    context.policy-reference-name.groups has "c242c5b0-6081-1845-6fa8-6e0d9513c107"
    && context.policy-reference-name.user.email.verified == true
    && context.jamf.risk == "LOW"
};
```

Untuk informasi selengkapnya tentang data kepercayaan, lihatthe section called “AWS IAM Identity Center konteks”.

Contoh 2: Berikan akses ke grup di penyedia pihak ketiga

Contoh kebijakan berikut mengizinkan akses hanya ketika pengguna berada dalam grup yang ditentukan, pengguna memiliki alamat email terverifikasi, dan skor risiko perangkat Jamf RENDAH. Nama grup adalah “keuangan”.

```
permit(principal,action,resource)
when {
     context.policy-reference-name.groups.contains("finance")
     && context.policy-reference-name.email_verified == true
     && context.jamf.risk == "LOW"
};
```

Untuk informasi selengkapnya tentang data kepercayaan, lihatthe section called “Konteks pihak ketiga”.

Contoh 3: Berikan akses menggunakan CrowdStrike

Contoh kebijakan berikut memungkinkan akses ketika skor penilaian keseluruhan lebih besar dari 50.

```
permit(principal,action,resource)
```

```
when {
    context.crwd.assessment.overall > 50
};
```

Contoh 4: Izinkan atau tolak alamat IP tertentu

Contoh kebijakan berikut memungkinkan permintaan hanya dari alamat IP yang ditentukan.

```
permit(principal, action, resource)
when {
    context.http_request.client_ip == "192.0.2.1"
};
```

Contoh kebijakan berikut menolak permintaan dari alamat IP yang ditentukan.

```
forbid(principal,action,resource)
when {
    ip(context.http_request.client_ip).isInRange(ip("192.0.2.1/32"))
};
```

Asisten kebijakan Akses Terverifikasi

Asisten kebijakan Akses Terverifikasi adalah alat di konsol Akses Terverifikasi yang dapat Anda gunakan untuk menguji dan mengembangkan kebijakan Anda. Ini menyajikan kebijakan titik akhir, kebijakan grup, dan konteks kepercayaan pada satu layar, di mana Anda dapat menguji dan mengedit kebijakan.

Format konteks kepercayaan bervariasi di berbagai penyedia kepercayaan, dan terkadang administrator Akses Terverifikasi mungkin tidak mengetahui format persis yang digunakan penyedia kepercayaan tertentu. Itulah mengapa sangat membantu untuk melihat konteks kepercayaan, dan kebijakan kelompok dan titik akhir di satu tempat untuk tujuan pengujian dan pengembangan.

Bagian berikut menjelaskan dasar-dasar penggunaan editor kebijakan.

Tugas

- Langkah 1: Tentukan sumber daya Anda
- Langkah 2: Uji dan edit kebijakan
- Langkah 3: Tinjau dan terapkan perubahan

Langkah 1: Tentukan sumber daya Anda

Pada halaman pertama asisten kebijakan, Anda menentukan titik akhir Akses Terverifikasi yang ingin Anda gunakan. Anda juga akan menentukan pengguna (diidentifikasi oleh alamat email), dan secara opsional, nama pengguna dan/atau pengenal perangkat. Secara default, keputusan otorisasi terbaru diekstraksi dari log Akses Terverifikasi untuk pengguna tertentu. Anda dapat secara opsional memilih mengizinkan atau menolak keputusan terbaru secara khusus.

Terakhir, konteks kepercayaan, keputusan otorisasi, kebijakan titik akhir, dan kebijakan grup semuanya ditampilkan di layar berikutnya.

Untuk membuka asisten kebijakan dan menentukan sumber daya Anda

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instance Akses Terverifikasi, lalu klik ID instans Akses Terverifikasi untuk instance yang ingin Anda gunakan.
3. Pilih Peluncuran asisten kebijakan.
4. Untuk alamat email Pengguna, masukkan alamat email pengguna.
5. Untuk titik akhir Akses Terverifikasi, pilih titik akhir yang ingin Anda edit dan uji kebijakan.
6. (Opsional) Untuk Nama, berikan nama pengguna.
7. (Opsional) Di bawah Pengenal perangkat, berikan pengenal perangkat unik.
8. (Opsional) Untuk hasil Otorisasi, pilih jenis hasil otorisasi terbaru yang ingin Anda gunakan. Secara default, hasil otorisasi terbaru akan digunakan.
9. Pilih Berikutnya.

Langkah 2: Uji dan edit kebijakan

Pada halaman ini Anda akan disajikan dengan informasi berikut untuk bekerja dengan:

- Konteks kepercayaan yang dikirim oleh penyedia kepercayaan Anda untuk pengguna dan (opsional) perangkat yang Anda tentukan pada langkah sebelumnya.
- Kebijakan Cedar untuk titik akhir Akses Terverifikasi yang ditentukan pada langkah sebelumnya.
- Kebijakan Cedar untuk grup Akses Terverifikasi yang menjadi milik titik akhir.

Kebijakan Cedar untuk titik akhir dan grup Akses Terverifikasi dapat diedit di halaman ini, tetapi konteks kepercayaannya statis. Anda sekarang dapat menggunakan halaman ini untuk melihat konteks kepercayaan di samping kebijakan Cedar.

Uji kebijakan terhadap konteks kepercayaan dengan memilih tombol Uji kebijakan, dan hasil otorisasi akan ditampilkan di layar. Anda dapat mengedit kebijakan dan menguji ulang perubahan Anda, mengulangi proses sesuai kebutuhan.

Setelah Anda puas dengan perubahan yang dibuat pada kebijakan, pilih Berikutnya untuk melanjutkan ke layar asisten kebijakan berikutnya.

Langkah 3: Tinjau dan terapkan perubahan

Pada halaman terakhir asisten kebijakan, Anda akan melihat perubahan yang Anda buat pada kebijakan yang disorot agar mudah ditinjau. Anda sekarang dapat meninjaunya untuk terakhir kalinya dan memilih Terapkan perubahan untuk melakukan perubahan.

Anda juga memiliki opsi untuk kembali ke halaman sebelumnya dengan memilih Sebelumnya, atau membatalkan asisten kebijakan sepenuhnya dengan memilih Batal.

Konektivitas Klien untuk Akses Terverifikasi AWS

Akses Terverifikasi AWS menyediakan Connectivity Client sehingga Anda dapat mengaktifkan konektivitas antara perangkat pengguna dan aplikasi non-HTTP. Klien mengenkripsi lalu lintas pengguna dengan aman, menambahkan informasi identitas pengguna dan konteks perangkat, dan merutekan ke Akses Terverifikasi untuk penegakan kebijakan. Jika kebijakan akses mengizinkan akses, pengguna terhubung ke aplikasi. Akses pengguna terus diotorisasi selama Klien Konektivitas terhubung.

Klien berjalan sebagai layanan sistem dan tangguh terhadap crash. Jika koneksi menjadi tidak stabil, klien membangun kembali koneksi.

Klien menggunakan token OAuth akses singkat untuk membangun terowongan aman. Terowongan terputus saat pengguna keluar dari klien.

Token akses dan penyegaran disimpan secara lokal di perangkat pengguna, dalam database terenkripsi SQLite .

Daftar Isi

- Prasyarat
- Unduh Klien Konektivitas
- Ekspor file konfigurasi klien
- Connect ke aplikasi
- Copot pemasangan klien
- Praktik terbaik
- Pemecahan Masalah
- Riwayat versi

Prasyarat

Sebelum menggunakan fungsi , pastikan untuk melengkapi prasyarat berikut:

- Buat instance Akses Terverifikasi dengan penyedia kepercayaan.
- Buat titik akhir TCP untuk aplikasi Anda.
- Putuskan sambungan komputer Anda dari klien VPN apa pun untuk menghindari masalah perutean.

- Aktifkan IPv6 di komputer Anda. Untuk instruksi, lihat dokumentasi untuk sistem operasi yang berjalan di komputer Anda.
- Di komputer Windows, verifikasi bahwa Trusted Platform Module (TPM) didukung dan instal runtime WebView2.

Unduh Klien Konektivitas

Copot pemasangan versi klien sebelumnya. Unduh klien, verifikasi bahwa penginstal ditandatangani, dan jalankan penginstal. Jangan menginstal klien menggunakan penginstal yang tidak ditandatangani.

- Klien Konektivitas untuk Mac dengan Apple Silicon versi 1.0.2
- Klien Konektivitas untuk Mac dengan Intel versi 1.0.2
- Connectivity Client untuk Windows dengan x64 versi 1.0.2

Ekspor file konfigurasi klien

Gunakan prosedur berikut untuk mengekspor informasi konfigurasi yang diperlukan oleh klien dari instans Akses Terverifikasi Anda.

Untuk mengekspor file konfigurasi klien menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi.
4. Pilih Tindakan, Ekspor file konfigurasi klien.

Untuk mengekspor file konfigurasi klien menggunakan AWS CLI

Gunakan perintah export-verified-access-instance-client-configuration. Simpan output ke file.json. Nama file harus dimulai dengan `ClientConfig-` awalan.

Connect ke aplikasi

Gunakan prosedur berikut untuk terhubung ke aplikasi menggunakan klien.

Untuk terhubung ke aplikasi menggunakan klien

1. Menyebarkan file konfigurasi klien ke perangkat pengguna di lokasi berikut:
   - Jendela — `C:\ProgramData\Connectivity Client`
   - macOS — `/Library/Application\ Support/Connectivity\ Client`
2. Pastikan file konfigurasi klien dimiliki oleh root (macOS) atau Admin (Windows).
3. Luncurkan Klien Konektivitas.
4. Setelah Connectivity Client dimuat, pengguna diautentikasi oleh iDP.
5. Setelah otentikasi, pengguna dapat mengakses aplikasi menggunakan nama DNS yang disediakan oleh Akses Terverifikasi, menggunakan klien pilihan mereka.

Copot pemasangan klien

Ketika Anda selesai menggunakan Connectivity Client, Anda dapat menghapus instalannya.

macOS

Versi 1.0.1 dan yang lebih baru

Arahkan ke `/Applications/Connectivity Client` dan jalankan`Connectivity Client Uninstaller.app`.

Versi 1.0.0

Unduh `connectivity_client_cleanup.sh` skrip untuk Mac dengan Apple Silicon atau Mac dengan Intel, atur izin eksekusi pada skrip, dan jalankan skrip sebagai berikut.

```
sudo ./connectivity_client_cleanup.sh
```

Windows

Untuk menghapus instalasi klien pada Windows, jalankan installer dan pilih Hapus.

Praktik terbaik

Pertimbangkan praktik terbaik berikut:

- Instal versi terbaru klien.

- Jangan menginstal klien menggunakan penginstal yang tidak ditandatangani.
- Pengguna tidak boleh menggunakan konfigurasi kecuali itu adalah konfigurasi tepercaya yang disediakan oleh admin TI. Konfigurasi yang tidak tepercaya dapat dialihkan ke halaman phishing.
- Pengguna harus keluar dari klien sebelum membiarkan workstation mereka menganggur.
- Tambahkan `offline_access` cakupan ke konfigurasi OIDC Anda. Ini memungkinkan permintaan untuk token penyegaran, yang digunakan untuk mendapatkan lebih banyak token akses tanpa mengharuskan pengguna untuk mengautentikasi ulang.

Pemecahan Masalah

Informasi berikut dapat membantu Anda memecahkan masalah dengan klien.

Masalah

- Saat masuk, browser tidak terbuka untuk menyelesaikan otentikasi oleh iDP
- Setelah otentikasi, status klien “tidak terhubung”
- Tidak dapat terhubung menggunakan browser Chrome atau Edge

Saat masuk, browser tidak terbuka untuk menyelesaikan otentikasi oleh iDP

Kemungkinan penyebabnya: File konfigurasi hilang atau cacat.

Solusi: Hubungi administrator sistem Anda dan minta file konfigurasi yang diperbarui.

Setelah otentikasi, status klien “tidak terhubung”

Kemungkinan penyebabnya: Menjalankan perangkat lunak VPN lainnya, seperti, Cisco AWS Client VPN AnyConnect, atau OpenVPN Connect.

Solusi: Putuskan sambungan dari perangkat lunak VPN lainnya. Jika Anda masih tidak dapat terhubung, buat laporan diagnostik dan bagikan dengan administrator sistem Anda.

Kemungkinan penyebabnya: Pada platform Windows, klien menggunakan HTTP pada port 80 untuk komunikasi bidang kontrol. Aturan firewall yang memblokir port TCP 80 mencegah komunikasi bidang kontrol.

Solusi: Periksa aturan Windows Firewall untuk aturan keluar eksplisit yang memblokir TCP pada port 80 dan nonaktifkan.

Tidak dapat terhubung menggunakan browser Chrome atau Edge

Kemungkinan penyebabnya: Saat menghubungkan ke aplikasi web menggunakan browser Chrome atau Edge, browser gagal menyelesaikan nama IPv6 domain.

Solusi: Kontak AWS Dukungan.

Riwayat versi

Tabel berikut berisi riwayat versi klien.

Versi	Perubahan	Unduh	Tanggal
1.0.2	macOS • Perbaikan bug dan peningkatan stabilitas • Penyempurnaan UI Windows • Perbaikan bug dan peningkatan stabilitas • Penyempurnaan UI	• Mac dengan Apple Silicon • Mac dengan Intel • Windows dengan x64	Juni 9, 2025
1.0.1	macOS • Peningkatan stabilitas • Aplikasi uninstaller Windows • Peningkatan stabilitas	• Mac dengan Apple Silicon • Mac dengan Intel • Windows dengan x64	Februari 5, 2025
1.0.0	Pratinjau publik	• Mac dengan Apple Silicon • Mac dengan Intel • Windows dengan x64	Desember 1, 2024

Keamanan dalam Akses Terverifikasi

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 Akses AWS Terverifikasi, lihat AWS Layanan dalam Lingkup oleh AWS Layanan Program Kepatuhan .
- 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 Akses Terverifikasi. Topik berikut menunjukkan cara mengonfigurasi Akses Terverifikasi untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya Akses Terverifikasi Anda.

Daftar Isi

- Perlindungan data dalam Akses Terverifikasi
- Manajemen identitas dan akses untuk Akses Terverifikasi
- Validasi kepatuhan untuk Akses Terverifikasi
- Ketahanan dalam Akses Terverifikasi

Perlindungan data dalam Akses Terverifikasi

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di Akses AWS Terverifikasi. 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 kredensyal 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 sumber daya. AWS Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan logging 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 Akses Terverifikasi 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.

Enkripsi bergerak

Verified Access mengenkripsi semua data dalam perjalanan dari pengguna akhir ke titik akhir Akses Terverifikasi melalui Internet menggunakan Transport Layer Security (TLS) 1.2 atau yang lebih baru.

Privasi lalu lintas antar jaringan

Anda dapat mengonfigurasi Akses Terverifikasi untuk membatasi akses ke sumber daya tertentu di VPC Anda. Untuk otentikasi berbasis pengguna, Anda juga dapat membatasi akses ke bagian jaringan Anda, berdasarkan grup pengguna yang mengakses titik akhir. Untuk informasi selengkapnya, lihat Kebijakan Akses Terverifikasi.

Enkripsi data saat istirahat untuk Akses AWS Terverifikasi

AWS Akses Terverifikasi mengenkripsi data saat istirahat secara default, menggunakan kunci KMS yang AWS dimiliki. Ketika enkripsi data saat istirahat terjadi secara default, ini membantu mengurangi overhead operasional dan kompleksitas yang terlibat dalam melindungi data sensitif. Pada saat yang sama, ini memungkinkan Anda untuk membangun aplikasi aman yang memenuhi kepatuhan enkripsi yang ketat dan persyaratan peraturan. Bagian berikut memberikan rincian tentang bagaimana Akses Terverifikasi menggunakan kunci KMS untuk enkripsi data saat istirahat.

Daftar Isi

- Akses Terverifikasi dan kunci KMS
- Informasi pengenal pribadi
- Bagaimana Akses AWS Terverifikasi menggunakan hibah di AWS KMS
- Menggunakan kunci terkelola pelanggan dengan Akses Terverifikasi
- Menentukan kunci terkelola pelanggan untuk sumber daya Akses Terverifikasi
- AWS Konteks enkripsi Akses Terverifikasi
- Memantau kunci enkripsi Anda untuk Akses AWS Terverifikasi

Akses Terverifikasi dan kunci KMS

AWS kunci yang dimiliki

Akses Terverifikasi menggunakan kunci KMS untuk mengenkripsi informasi identitas pribadi (PII) secara otomatis. Ini terjadi secara default, dan Anda sendiri tidak dapat melihat, mengelola, menggunakan, atau mengaudit penggunaan kunci yang dimiliki AWS. Namun, Anda tidak perlu

mengambil tindakan apa pun atau mengubah program apa pun untuk melindungi kunci yang mengenkripsi data Anda. Untuk informasi selengkapnya, lihat kunci yang AWS dimiliki di Panduan AWS Key Management Service Pengembang.

Meskipun Anda tidak dapat menonaktifkan lapisan enkripsi ini atau memilih jenis enkripsi alternatif, Anda dapat menambahkan lapisan enkripsi kedua di atas kunci enkripsi yang ada AWS dengan memilih kunci yang dikelola pelanggan saat Anda membuat sumber daya Akses Terverifikasi.

Kunci yang dikelola pelanggan

Akses Terverifikasi mendukung penggunaan kunci terkelola pelanggan simetris yang Anda buat dan kelola, untuk menambahkan lapisan enkripsi kedua di atas enkripsi default yang ada. Karena Anda memiliki kontrol penuh atas lapisan enkripsi ini, Anda dapat melakukan tugas-tugas seperti:

- Menetapkan dan memelihara kebijakan utama
- Menetapkan dan memelihara kebijakan dan hibah IAM
- Mengaktifkan dan menonaktifkan kebijakan utama
- Memutar bahan kriptografi kunci
- Menambahkan tanda
- Membuat alias kunci
- Kunci penjadwalan untuk penghapusan

Untuk informasi selengkapnya, lihat Kunci terkelola pelanggan di Panduan AWS Key Management Service Pengembang.

> **Note**
>
> Akses Terverifikasi secara otomatis mengaktifkan enkripsi saat istirahat menggunakan kunci yang AWS dimiliki untuk melindungi data yang dapat diidentifikasi secara pribadi tanpa biaya. Namun, AWS KMS biaya akan berlaku ketika Anda menggunakan kunci yang dikelola pelanggan. Untuk informasi lebih lanjut tentang harga, lihat AWS Key Management Service harga.

Informasi pengenal pribadi

Tabel berikut merangkum informasi yang dapat diidentifikasi secara pribadi (PII) yang digunakan Akses Terverifikasi, dan bagaimana informasi tersebut dienkripsi.

Tipe data	AWS enkripsi kunci yang dimiliki	Enkripsi kunci yang dikelola pelanggan (Opsional)
`Trust provider (user-type)` Penyedia kepercayaan tipe pengguna berisi opsi OIDC seperti AuthorizationEndpoint,,, UserInfoEndpoint ClientId, dan sebagainya ClientSecret, yang dianggap PII.	Diaktifkan	Diaktifkan
`Trust provider (device-type)` Penyedia kepercayaan tipe perangkat berisi a TenantId, yang dianggap PII.	Diaktifkan	Diaktifkan
`Group policy` Disediakan selama pembuatan atau modifikasi grup Akses Terverifikasi. Berisi aturan untuk mengotorisasi permintaan akses. Mungkin berisi PII seperti nama pengguna dan alamat email, dan sebagainya.	Diaktifkan	Diaktifkan
`Endpoint policy` Disediakan selama pembuatan atau modifikasi titik akhir Akses Terverifikasi. Berisi aturan untuk	Diaktifkan	Diaktifkan

Tipe data	AWS enkripsi kunci yang dimiliki	Enkripsi kunci yang dikelola pelanggan (Opsional)
mengotorisasi permintaan akses. Mungkin berisi PII seperti nama pengguna dan alamat email, dan sebagainya.		

Bagaimana Akses AWS Terverifikasi menggunakan hibah di AWS KMS

Akses Terverifikasi memerlukan hibah untuk menggunakan kunci terkelola pelanggan Anda.

Saat Anda membuat sumber daya Akses Terverifikasi yang dienkripsi dengan kunci terkelola pelanggan, Akses Terverifikasi akan membuat hibah atas nama Anda dengan mengirimkan CreateGrantpermintaan ke. AWS KMS Hibah AWS KMS digunakan untuk memberikan Akses Terverifikasi akses ke kunci yang dikelola pelanggan di akun Anda.

Akses Terverifikasi memerlukan hibah untuk menggunakan kunci terkelola pelanggan Anda untuk operasi internal berikut:

- Kirim permintaan Dekripsi ke AWS KMS untuk mendekripsi kunci data terenkripsi sehingga mereka dapat digunakan untuk mendekripsi data Anda.
- Kirim RetireGrantpermintaan AWS KMS untuk menghapus hibah.

Anda dapat mencabut akses ke hibah, atau menghapus akses layanan ke kunci yang dikelola pelanggan kapan saja. Jika Anda melakukannya, Akses Terverifikasi tidak akan dapat mengakses data apa pun yang dienkripsi oleh kunci terkelola pelanggan, yang memengaruhi operasi yang bergantung pada data tersebut.

Menggunakan kunci terkelola pelanggan dengan Akses Terverifikasi

Anda dapat membuat kunci yang dikelola pelanggan simetris dengan menggunakan AWS Management Console, atau. AWS KMS APIs Ikuti langkah-langkah untuk Membuat kunci enkripsi simetris di Panduan AWS Key Management Service Pengembang.

Kebijakan utama

Kebijakan utama mengontrol akses ke kunci yang dikelola pelanggan Anda. Setiap kunci yang dikelola pelanggan harus memiliki persis satu kebijakan utama, yang berisi pernyataan yang

menentukan siapa yang dapat menggunakan kunci dan bagaimana mereka dapat menggunakannya. Saat membuat kunci terkelola pelanggan, Anda dapat menentukan kebijakan kunci. Untuk informasi selengkapnya, lihat Kebijakan utama di Panduan AWS Key Management Service Pengembang.

Untuk menggunakan kunci terkelola pelanggan dengan sumber daya Akses Terverifikasi, operasi API berikut harus diizinkan dalam kebijakan kunci:

- `kms:CreateGrant`— Menambahkan hibah ke kunci yang dikelola pelanggan. Memberikan akses kontrol ke kunci KMS tertentu, yang memungkinkan akses untuk memberikan operasi yang diperlukan Akses Terverifikasi. Untuk informasi selengkapnya, lihat Hibah, di Panduan AWS Key Management Service Pengembang.

  Hal ini memungkinkan Akses Terverifikasi untuk melakukan hal berikut:

  - Panggilan `GenerateDataKeyWithoutPlainText` untuk menghasilkan kunci data terenkripsi dan menyimpannya, karena kunci data tidak segera digunakan untuk mengenkripsi.
  - Panggilan `Decrypt` untuk menggunakan kunci data terenkripsi yang disimpan untuk mengakses data terenkripsi.
  - Siapkan kepala sekolah yang pensiun untuk memungkinkan layanan. `RetireGrant`
- `kms:DescribeKey`— Memberikan detail kunci yang dikelola pelanggan untuk memungkinkan Akses Terverifikasi memvalidasi kunci.
- `kms:GenerateDataKey`— Memungkinkan Akses Terverifikasi untuk menggunakan kunci untuk mengenkripsi data.
- `kms:Decrypt`— Izinkan Akses Terverifikasi untuk mendekripsi kunci data terenkripsi.

Berikut ini adalah contoh kebijakan kunci yang dapat Anda gunakan untuk Akses Terverifikasi.

```
"Statement" : [
    {
      "Sid" : "Allow access to principals authorized to use Verified Access",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "*"
      },
      "Action" : [
        "kms:DescribeKey",
        "kms:CreateGrant",
        "kms:GenerateDataKey",
        "kms:Decrypt"
```

```
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "kms:ViaService" : "verified-access.region.amazonaws.com",
          "kms:CallerAccount" : "111122223333"
        }
    },
    {
      "Sid": "Allow access for key administrators",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
       },
      "Action" : [
        "kms:*"
       ],
      "Resource": "arn:aws:kms:region:111122223333:key/key_ID"
    },
    {
      "Sid" : "Allow read-only access to key metadata to the account",
      "Effect" : "Allow",
      "Principal" : {
        "AWS" : "arn:aws:iam::111122223333:root"
      },
      "Action" : [
        "kms:Describe*",
        "kms:Get*",
        "kms:List*",
        "kms:RevokeGrant"
      ],
      "Resource" : "*"
    }
  ]
```

Untuk informasi selengkapnya, lihat Membuat kebijakan kunci dan akses kunci pemecahan masalah di Panduan AWS Key Management Service Pengembang.

Menentukan kunci terkelola pelanggan untuk sumber daya Akses Terverifikasi

Anda dapat menentukan kunci yang dikelola pelanggan untuk menyediakan enkripsi lapisan kedua untuk sumber daya berikut:

- Grup Akses Terverifikasi
- Titik akhir Akses Terverifikasi
- Penyedia kepercayaan Akses Terverifikasi

Bila Anda membuat salah satu sumber daya ini menggunakan AWS Management Console, Anda dapat menentukan kunci yang dikelola pelanggan di bagian Enkripsi tambahan -- opsional. Selama proses, pilih kotak centang Sesuaikan pengaturan enkripsi (lanjutan), lalu masukkan ID AWS KMS kunci yang ingin Anda gunakan. Ini juga dapat dilakukan ketika memodifikasi sumber daya yang ada, atau dengan menggunakan file. AWS CLI

> **Note**
>
> Jika kunci terkelola pelanggan yang digunakan untuk menambahkan enkripsi tambahan ke salah satu sumber daya di atas hilang, nilai konfigurasi untuk sumber daya tidak lagi dapat diakses. Namun sumber daya dapat dimodifikasi, dengan menggunakan AWS Management Console or AWS CLI, untuk menerapkan kunci yang dikelola pelanggan baru dan mengatur ulang nilai konfigurasi.

AWS Konteks enkripsi Akses Terverifikasi

Konteks enkripsi adalah kumpulan opsional pasangan kunci-nilai yang berisi informasi kontekstual tambahan tentang data. AWS KMS menggunakan konteks enkripsi sebagai data otentikasi tambahan untuk mendukung enkripsi yang diautentikasi. Bila Anda menyertakan konteks enkripsi dalam permintaan untuk mengenkripsi data, AWS KMS mengikat konteks enkripsi ke data terenkripsi. Untuk mendekripsi data, Anda menyertakan konteks enkripsi yang sama dalam permintaan.

AWS Konteks enkripsi Akses Terverifikasi

Akses Terverifikasi menggunakan konteks enkripsi yang sama di semua operasi AWS KMS kriptografi, di mana kuncinya `aws:verified-access:arn` dan nilainya adalah sumber daya Amazon Resource Name (ARN). Di bawah ini adalah konteks enkripsi untuk sumber daya Akses Terverifikasi.

Penyedia kepercayaan Akses Terverifikasi

```
"encryptionContext": {
    "aws:verified-access:arn":
    "arn:aws:ec2:region:111122223333:VerifiedAccessTrustProviderId"
```

```
}
```

Grup Akses Terverifikasi

```
"encryptionContext": {
    "aws:verified-access:arn":
    "arn:aws:ec2:region:111122223333:VerifiedAccessGroupId"
}
```

Titik akhir Akses Terverifikasi

```
"encryptionContext": {
    "aws:verified-access:arn":
    "arn:aws:ec2:region:111122223333:VerifiedAccessEndpointId"
}
```

Memantau kunci enkripsi Anda untuk Akses AWS Terverifikasi

Saat Anda menggunakan kunci KMS yang dikelola pelanggan dengan sumber daya Akses AWS Terverifikasi, Anda dapat menggunakannya AWS CloudTrailuntuk melacak permintaan yang dikirimkan Akses Terverifikasi. AWS KMS

Contoh berikut adalah AWS CloudTrail peristiwa untuk`CreateGrant`,,,`RetireGrant`, dan `Decrypt` `DescribeKey``GenerateDataKey`, yang memantau operasi KMS yang dipanggil oleh Akses Terverifikasi untuk mengakses data yang dienkripsi oleh kunci KMS yang dikelola pelanggan Anda:

CreateGrant

Saat Anda menggunakan kunci yang dikelola pelanggan untuk mengenkripsi sumber daya Anda, Akses Terverifikasi mengirimkan `CreateGrant` permintaan atas nama Anda untuk mengakses kunci di AWS akun Anda. Hibah yang dibuat oleh Akses Terverifikasi khusus untuk sumber daya yang terkait dengan kunci yang dikelola pelanggan.

Contoh peristiwa berikut mencatat `CreateGrant` operasi:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
```

```
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T16:27:12Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T16:41:42Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "operations": [
            "Decrypt",
            "RetireGrant",
            "GenerateDataKey"
        ],
        "keyId": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae",
        "constraints": {
            "encryptionContextSubset": {
                "aws:verified-access:arn": "arn:aws:ec2:ca-
central-1:111122223333:verified-access-trust-provider/vatp-0e54f581e2e5c97a2"
            }
        },
        "granteePrincipal": "verified-access.ca-central-1.amazonaws.com",
        "retiringPrincipal": "verified-access.ca-central-1.amazonaws.com"
    },
    "responseElements": {
        "grantId":
 "e5a050fff9893ba1c43f83fddf61e5f9988f579beaadd6d4ad6d1df07df6048f",
```

```
        "keyId": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
    },
    "requestID": "0faa837e-5c69-4189-9736-3957278e6444",
    "eventID": "1b6dd8b8-cbee-4a83-9b9d-d95fa5f6fd08",
    "readOnly": false,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

RetireGrant

Akses Terverifikasi menggunakan `RetireGrant` operasi untuk menghapus hibah saat Anda menghapus sumber daya.

Contoh peristiwa berikut mencatat `RetireGrant` operasi:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
```

```
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T16:42:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T16:47:53Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "RetireGrant",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": null,
    "responseElements": {
        "keyId": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
    },
    "additionalEventData": {
        "grantId":
 "b35e66f9bacb266cec214fcaa353c9cf750785e28773e61ba6f434d8c5c7632f"
    },
    "requestID": "7d4a31c2-d426-434b-8f86-336532a70462",
    "eventID": "17edc343-f25b-43d4-bbff-150d8fff4cf8",
    "readOnly": false,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-central-1:111122223333:key/5ed79e7f-88c9-420c-
ae1a-61ee87104dae"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


Decrypt

Akses Terverifikasi memanggil `Decrypt` operasi untuk menggunakan kunci data terenkripsi yang disimpan untuk mengakses data terenkripsi.

Contoh peristiwa berikut mencatat `Decrypt` operasi:


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T17:19:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T17:47:05Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e",
        "encryptionContext": {
            "aws:verified-access:arn": "arn:aws:ec2:ca-
central-1:111122223333:verified-access-trust-provider/vatp-00f20a4e455e9340f",
            "aws-crypto-public-key": "AkK+vi1W/
acBKv7OR8p2DeUrA8EgpTffSrjBqNucODuBYhyZ3hlMuYYJz9x7CwQWZw=="
        }
    },
    "responseElements": null,
```

```
    "requestID": "2e920fd3-f2f6-41b2-a5e7-2c2cb6f853a9",
    "eventID": "3329e0a3-bcfb-44cf-9813-8106d6eee31d",
    "readOnly": true,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

DescribeKey

Akses Terverifikasi menggunakan `DescribeKey` operasi untuk memverifikasi apakah kunci terkelola pelanggan yang terkait dengan sumber daya Anda ada di akun dan Wilayah.

Contoh peristiwa berikut mencatat `DescribeKey` operasi:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T17:19:33Z",
                "mfaAuthenticated": "false"
```

```
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T17:46:48Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "keyId": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
    },
    "responseElements": null,
    "requestID": "5b127082-6691-48fa-bfb0-4d40e1503636",
    "eventID": "ffcfc2bb-f94b-4c00-b6fb-feac77daff2a",
    "readOnly": true,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

GenerateDataKey

Contoh peristiwa berikut mencatat `GenerateDataKey` operasi:

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AKIAI44QH8DHBEXAMPLE",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
```

```
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AKIAI44QH8DHBEXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Admin",
                "accountId": "111122223333",
                "userName": "Admin"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-09-11T17:19:33Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "verified-access.amazonaws.com"
    },
    "eventTime": "2023-09-11T17:46:49Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "ca-central-1",
    "sourceIPAddress": "verified-access.amazonaws.com",
    "userAgent": "verified-access.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:verified-access:arn": "arn:aws:ec2:ca-
central-1:111122223333:verified-access-trust-provider/vatp-00f20a4e455e9340f",
            "aws-crypto-public-key": "A/ATGxaYatPUlOtM+l/mfDndkzHUmX5Hav+29IlIm
+JRBKFuXf24ulztmOIsqFQliw=="
        },
        "numberOfBytes": 32,
        "keyId": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
    },
    "responseElements": null,
    "requestID": "06535808-7cce-4ae1-ab40-e3afbf158a43",
    "eventID": "1ce79601-5a5e-412c-90b3-978925036526",
    "readOnly": true,
    "resources": [
        {
            "accountId": "AWS Internal",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:ca-
central-1:111122223333:key/380d006e-706a-464b-99c5-68768297114e"
        }
```

```
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Manajemen identitas dan akses untuk Akses Terverifikasi

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 diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya Akses Terverifikasi. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- Audiens
- Mengautentikasi dengan identitas
- Mengelola akses menggunakan kebijakan
- Cara Kerja Akses Terverifikasi dengan IAM
- Contoh kebijakan berbasis identitas untuk Akses Terverifikasi
- Memecahkan masalah Identitas dan akses Akses Terverifikasi
- Menggunakan peran terkait layanan untuk Akses Terverifikasi
- AWS kebijakan terkelola untuk Akses Terverifikasi

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di Akses Terverifikasi.

Pengguna layanan — Jika Anda menggunakan layanan Akses Terverifikasi untuk melakukan pekerjaan Anda, administrator Anda memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak fitur Akses Terverifikasi untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di Akses Terverifikasi, lihatMemecahkan masalah Identitas dan akses Akses Terverifikasi.

Administrator layanan — Jika Anda bertanggung jawab atas sumber daya Akses Terverifikasi di perusahaan Anda, Anda mungkin memiliki akses penuh ke Akses Terverifikasi. Tugas Anda adalah menentukan fitur dan sumber daya Akses Terverifikasi mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM dengan Akses Terverifikasi, lihatCara Kerja Akses Terverifikasi dengan IAM.

Administrator IAM — Jika Anda administrator IAM, Anda mungkin ingin mempelajari detail tentang cara menulis kebijakan untuk mengelola akses ke Akses Terverifikasi. Untuk melihat contoh kebijakan berbasis identitas Akses Terverifikasi yang dapat Anda gunakan di IAM, lihat. Contoh kebijakan berbasis identitas untuk Akses Terverifikasi

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengasumsikan peran IAM.

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensi yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (IAM Identity Center), autentikasi masuk tunggal perusahaan Anda, dan kredensi Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas terfederasi, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS.

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensil Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Guna mengetahui informasi selengkapnya tentang penggunaan metode yang disarankan untuk menandatangani permintaan sendiri, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-

faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, lihat Autentikasi multi-faktor dalam Panduan Pengguna AWS IAM Identity Center dan Autentikasi multi-faktor AWS di IAM dalam Panduan Pengguna IAM.

Akun AWS pengguna root

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat Tugas yang memerlukan kredensial pengguna root dalam Panduan Pengguna IAM.

Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensi sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, direktori Pusat Identitas AWS Directory Service, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensil yang disediakan melalui sumber identitas. Ketika identitas federasi mengakses Akun AWS, mereka mengambil peran, dan peran memberikan kredensi sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat Identitas IAM, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua aplikasi Akun AWS dan aplikasi Anda. Untuk informasi tentang Pusat Identitas IAM, lihat Apakah itu Pusat Identitas IAM? dalam Panduan Pengguna AWS IAM Identity Center .

Pengguna dan grup IAM

Pengguna IAM adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, kami merekomendasikan untuk mengandalkan kredensial sementara, bukan membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan tertentu yang memerlukan kredensial jangka panjang dengan pengguna IAM, kami merekomendasikan Anda merotasi kunci

akses. Untuk informasi selengkapnya, lihat Merotasi kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang dalam Panduan Pengguna IAM.

Grup IAM adalah identitas yang menentukan sekumpulan pengguna IAM. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat meminta kelompok untuk menyebutkan IAMAdmins dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari selengkapnya, lihat Kasus penggunaan untuk pengguna IAM dalam Panduan Pengguna IAM.

Peran IAM

Peran IAM adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Untuk mengambil peran IAM sementara AWS Management Console, Anda dapat beralih dari pengguna ke peran IAM (konsol). Anda dapat mengambil peran dengan memanggil operasi AWS CLI atau AWS API atau dengan menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Peran IAM dengan kredensial sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat Buat peran untuk penyedia identitas pihak ketiga dalam Panduan Pengguna IAM. Jika menggunakan Pusat Identitas IAM, Anda harus mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM akan mengorelasikan set izin ke peran dalam IAM. Untuk informasi tentang set izin, lihat Set izin dalam Panduan Pengguna AWS IAM Identity Center .
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM guna mendapatkan berbagai izin secara sementara untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (prinsipal tepercaya) di akun lain untuk mengakses sumber daya di akun Anda. Peran adalah cara utama

untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk akses lintas akun, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Misalnya, saat Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
  - Sesi akses teruskan (FAS) — Saat Anda menggunakan pengguna IAM atau peran untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses maju.
  - Peran layanan – Peran layanan adalah peran IAM yang dijalankan oleh 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.
  - Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke peran layanan. 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.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan peran IAM untuk mengelola kredensi sementara untuk aplikasi yang berjalan pada EC2 instance dan membuat AWS CLI atau AWS permintaan API. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance. Untuk menetapkan AWS peran ke EC2 instance dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instance berisi peran dan memungkinkan program yang berjalan pada EC2 instance untuk mendapatkan kredensi sementara. Untuk informasi selengkapnya, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon di Panduan Pengguna IAM.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan isi dokumen kebijakan JSON, lihat Gambaran umum kebijakan JSON dalam Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasinya. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut bisa mendapatkan informasi peran dari AWS Management Console, API AWS CLI, atau AWS API.

Kebijakan berbasis identitas

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.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam. Akun AWS Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan yang dikelola atau kebijakan inline, lihat Pilih antara kebijakan yang dikelola 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. 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 prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau. Layanan AWS

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.

Daftar kontrol akses (ACLs)

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.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung. ACLs Untuk mempelajari selengkapnya ACLs, lihat Ringkasan daftar kontrol akses (ACL) di Panduan Pengembang Layanan Penyimpanan Sederhana Amazon.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan tempat Anda mengatur izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas ke entitas IAM (pengguna IAM atau peran IAM). Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya tentang batasan izin, lihat Batasan izin untuk entitas IAM dalam Panduan Pengguna IAM.
- Kebijakan kontrol layanan (SCPs) — SCPs adalah kebijakan JSON yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di. AWS Organizations AWS Organizations

adalah layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur dalam suatu organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCPs) ke salah satu atau semua akun Anda. SCP membatasi izin untuk entitas di akun anggota, termasuk masing-masing. Pengguna root akun AWS Untuk informasi selengkapnya tentang Organizations dan SCPs, lihat Kebijakan kontrol layanan di Panduan AWS Organizations Pengguna.

- Kebijakan kontrol sumber daya (RCPs) — RCPs adalah kebijakan JSON yang dapat Anda gunakan untuk menetapkan izin maksimum yang tersedia untuk sumber daya di akun Anda tanpa memperbarui kebijakan IAM yang dilampirkan ke setiap sumber daya yang Anda miliki. RCP membatasi izin untuk sumber daya di akun anggota dan dapat memengaruhi izin efektif untuk identitas, termasuk Pengguna root akun AWS, terlepas dari apakah itu milik organisasi Anda. Untuk informasi selengkapnya tentang Organizations dan RCPs, termasuk daftar dukungan Layanan AWS tersebut RCPs, lihat Kebijakan kontrol sumber daya (RCPs) di Panduan AWS Organizations Pengguna.
- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. 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 Akses Terverifikasi dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses ke Akses Terverifikasi, pelajari fitur IAM apa yang tersedia untuk digunakan dengan Akses Terverifikasi.

Fitur IAM	Dukungan Akses Terverifikasi
Kebijakan berbasis identitas	Ya

Fitur IAM	Dukungan Akses Terverifikasi
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
Kunci kondisi kebijakan	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 Akses Terverifikasi dan AWS layanan lainnya dengan sebagian besar fitur IAM, lihat AWS layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk Akses Terverifikasi

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. Anda tidak dapat menentukan secara spesifik prinsipal dalam sebuah kebijakan

berbasis identitas karena prinsipal berlaku bagi pengguna atau peran yang melekat kepadanya. 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 Akses Terverifikasi

Untuk melihat contoh kebijakan berbasis identitas Akses Terverifikasi, lihat. Contoh kebijakan berbasis identitas untuk Akses Terverifikasi

Kebijakan berbasis sumber daya dalam Akses Terverifikasi

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 prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan prinsipal 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 prinsipal dalam kebijakan berbasis sumber daya. Menambahkan prinsipal akun silang ke kebijakan berbasis sumber daya hanya setengah dari membangun hubungan kepercayaan. Ketika prinsipal dan sumber daya berbeda Akun AWS, administrator IAM di akun tepercaya juga harus memberikan izin entitas utama (pengguna atau peran) untuk mengakses sumber daya. Mereka memberikan izin dengan melampirkan kebijakan berbasis identitas kepada entitas. Namun, jika kebijakan berbasis sumber daya memberikan akses ke principal dalam akun yang sama, tidak diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun di IAM dalam Panduan Pengguna IAM.

Tindakan kebijakan untuk Akses Terverifikasi

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Action` dari kebijakan JSON menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam sebuah kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan operasi AWS API terkait. Ada beberapa pengecualian, misalnya tindakan hanya izin yang tidak memiliki operasi API yang cocok. Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Sertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Untuk melihat daftar tindakan Akses Terverifikasi, lihat Tindakan yang Ditentukan oleh Amazon EC2 di Referensi Otorisasi Layanan.

Tindakan kebijakan di Akses Terverifikasi menggunakan awalan berikut sebelum tindakan:

```
ec2
```

Untuk menetapkan secara spesifik beberapa tindakan dalam satu pernyataan, pisahkan tindakan tersebut dengan koma.

```
"Action": [
    "ec2:action1",
    "ec2:action2"
]
```

Untuk melihat contoh kebijakan berbasis identitas Akses Terverifikasi, lihat. Contoh kebijakan berbasis identitas untuk Akses Terverifikasi

Sumber daya kebijakan untuk Akses Terverifikasi

Mendukung sumber daya kebijakan: Ya

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen kebijakan JSON `Resource` menentukan objek yang menjadi target penerapan tindakan. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Praktik terbaiknya, tentukan sumber daya menggunakan Amazon Resource Name (ARN). Anda dapat melakukan ini untuk

tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

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

Untuk melihat daftar jenis sumber daya Akses Terverifikasi dan jenisnya ARNs, lihat Sumber Daya yang Ditentukan oleh Amazon EC2 di Referensi Otorisasi Layanan. Untuk mempelajari tindakan mana yang dapat Anda tentukan ARN dari setiap sumber daya, lihat Tindakan yang Ditentukan oleh Amazon. EC2

Untuk melihat contoh kebijakan berbasis identitas Akses Terverifikasi, lihat. Contoh kebijakan berbasis identitas untuk Akses Terverifikasi

Kunci kondisi kebijakan untuk Akses Terverifikasi

Mendukung kunci kondisi kebijakan khusus layanan: Yes

Administrator dapat menggunakan kebijakan AWS JSON untuk menentukan siapa yang memiliki akses ke apa. Artinya, prinsipal manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan kondisi apa.

Elemen `Condition` (atau blok `Condition`) akan memungkinkan Anda menentukan kondisi yang menjadi dasar suatu pernyataan berlaku. Elemen `Condition` bersifat opsional. 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.

Jika Anda menentukan beberapa elemen `Condition` dalam sebuah pernyataan, atau beberapa kunci dalam elemen `Condition` tunggal, maka AWS akan mengevaluasinya menggunakan operasi `AND` logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan `OR` operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Sebagai contoh, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tanda yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, lihat Elemen kebijakan IAM: variabel dan tanda dalam Panduan Pengguna IAM.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci kondisi Akses Terverifikasi, lihat Kunci Kondisi untuk Amazon EC2 di Referensi Otorisasi Layanan. Untuk mempelajari tindakan dan sumber daya yang dapat Anda gunakan kunci kondisi, lihat Tindakan yang Ditentukan oleh Amazon EC2.

Untuk melihat contoh kebijakan berbasis identitas Akses Terverifikasi, lihat. Contoh kebijakan berbasis identitas untuk Akses Terverifikasi

ACLs di Akses Terverifikasi

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 Akses Terverifikasi

Mendukung ABAC (tag dalam kebijakan): Sebagian

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut. Dalam AWS, atribut ini disebut tag. Anda dapat melampirkan tag ke entitas IAM (pengguna atau peran) dan ke banyak AWS sumber daya. Penandaan ke entitas dan sumber daya adalah langkah pertama dari ABAC. Kemudian rancanglah kebijakan ABAC untuk mengizinkan operasi ketika tanda milik prinsipal cocok dengan tanda yang ada di sumber daya yang ingin diakses.

ABAC sangat berguna di lingkungan yang berkembang dengan cepat dan berguna di situasi saat manajemen kebijakan menjadi rumit.

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 kredensi sementara dengan Akses Terverifikasi

Mendukung kredensial sementara: Ya

Beberapa Layanan AWS tidak berfungsi saat Anda masuk menggunakan kredensi sementara. Untuk informasi tambahan, termasuk yang Layanan AWS bekerja dengan kredensi sementara, lihat Layanan AWS yang bekerja dengan IAM di Panduan Pengguna IAM.

Anda menggunakan kredensi sementara jika Anda masuk AWS Management Console menggunakan metode apa pun kecuali nama pengguna dan kata sandi. Misalnya, ketika Anda mengakses AWS menggunakan tautan masuk tunggal (SSO) perusahaan Anda, proses tersebut secara otomatis membuat kredensil sementara. Anda juga akan secara otomatis membuat kredensial sementara ketika Anda masuk ke konsol sebagai seorang pengguna lalu beralih peran. Untuk informasi selengkapnya tentang peralihan peran, lihat Beralih dari pengguna ke peran IAM (konsol) dalam Panduan Pengguna IAM.

Anda dapat membuat kredenal sementara secara manual menggunakan API AWS CLI atau AWS . Anda kemudian dapat menggunakan kredensi sementara tersebut untuk mengakses. AWS 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.

Izin utama lintas layanan untuk Akses Terverifikasi

Mendukung sesi akses maju (FAS): Ya

Saat Anda menggunakan pengguna atau peran IAM untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. Permintaan FAS hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, lihat Sesi akses maju.

Peran layanan untuk Akses Terverifikasi

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.

Peran terkait layanan untuk Akses Terverifikasi

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 Akses Terverifikasi, lihat. Menggunakan peran terkait layanan untuk Akses Terverifikasi

Contoh kebijakan berbasis identitas untuk Akses Terverifikasi

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya Akses Terverifikasi. Mereka juga tidak dapat melakukan tugas dengan menggunakan AWS Management Console, AWS Command Line Interface (AWS CLI), atau AWS API. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian dapat menambahkan kebijakan IAM ke peran, dan pengguna dapat mengambil peran.

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 Akses Terverifikasi, termasuk format ARNs untuk setiap jenis sumber daya, lihat Tindakan, Sumber Daya, dan Kunci Kondisi untuk Amazon EC2 di Referensi Otorisasi Layanan.

Topik

- Praktik terbaik kebijakan
- Kebijakan untuk membuat instance Akses Terverifikasi
- Mengizinkan pengguna melihat izin mereka sendiri

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya Akses Terverifikasi 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 AWS 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.

Kebijakan untuk membuat instance Akses Terverifikasi

Untuk membuat instance Akses Terverifikasi, prinsipal IAM perlu menambahkan pernyataan tambahan ini ke kebijakan IAM mereka.

```
{
    "Effect": "Allow",
    "Action": "verified-access:AllowVerifiedAccess",
    "Resource": "*"
}
```

> **Note**
>
> `verified-access:AllowVerifiedAccess`adalah API virtual khusus aksi.
> Itu tidak mendukung otorisasi berbasis kunci sumber daya, tag, atau kondisi.
> Gunakan otorisasi berbasis kunci sumber daya, tag, atau kondisi pada tindakan API.
> `ec2:CreateVerifiedAccessInstance`

Contoh kebijakan untuk membuat instance Akses Terverifikasi. Dalam contoh ini, `123456789012` adalah nomor AWS rekening dan `us-east-1` merupakan AWS wilayah.

JSON

```
{
      "Version": "2012-10-17",
      "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:CreateVerifiedAccessInstance",
            "Resource": "arn:aws:ec2:us-east-1:123456789012:verified-access-
instance/*"
```

```
        },
        {
            "Effect": "Allow",
            "Action": "verified-access:AllowVerifiedAccess",
            "Resource": "*"
        }
    ]
}
```

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": "*"
        }
    ]
}
```

Memecahkan masalah Identitas dan akses Akses Terverifikasi

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan Akses Terverifikasi dan IAM.

Masalah

- Saya tidak berwenang untuk melakukan tindakan di Akses Terverifikasi
- Saya tidak berwenang untuk melakukan iam: PassRole
- Saya ingin mengizinkan orang di luar saya Akun AWS untuk mengakses sumber daya Akses Terverifikasi saya

Saya tidak berwenang untuk melakukan tindakan di Akses Terverifikasi

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 `ec2:`*`GetWidget`* rekaan.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 ec2: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 `ec2:`*`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 bahwa Anda tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran ke Akses Terverifikasi.

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 ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di Akses Terverifikasi. 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 Akses Terverifikasi 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 mengetahui apakah Akses Terverifikasi mendukung fitur ini, lihatCara Kerja Akses Terverifikasi 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.

Menggunakan peran terkait layanan untuk Akses Terverifikasi

Akses Terverifikasi AWS menggunakan peran terkait layanan IAM, yang merupakan jenis peran IAM yang ditautkan langsung ke layanan. AWS Peran terkait layanan untuk Akses Terverifikasi ditentukan oleh Akses Terverifikasi dan mencakup semua izin yang diperlukan layanan untuk memanggil orang lain Layanan AWS atas nama Anda.

Peran terkait layanan membuat pengaturan Akses Terverifikasi lebih mudah karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. Akses Terverifikasi mendefinisikan izin peran terkait layanan, dan kecuali ditentukan lain, hanya Akses Terverifikasi yang dapat mengambil perannya. Izin yang ditetapkan mencakup kebijakan kepercayaan dan kebijakan izin, dan kebijakan izin ini tidak dapat dilampirkan ke entitas IAM lainnya.

Izin peran terkait layanan untuk Akses Terverifikasi

Akses Terverifikasi menggunakan peran terkait layanan bernama AWSServiceRoleForVPCVerifiedAccess untuk menyediakan sumber daya di akun Anda yang diperlukan untuk menggunakan layanan.

Peran terkait layanan AWSServiceRoleForVPCVerifiedAccess mempercayai layanan berikut untuk mengambil peran:

- `verified-access.amazonaws.com`

Kebijakan izin peran, bernama AWSVPCVerifiedAccessServiceRolePolicy, memungkinkan Akses Terverifikasi untuk menyelesaikan tindakan berikut pada sumber daya yang ditentukan:

- Tindakan `ec2:CreateNetworkInterface` pada semua subnet dan grup keamanan, serta semua antarmuka jaringan dengan tag `VerifiedAccessManaged=true`
- Tindakan `ec2:CreateTags` pada semua antarmuka jaringan pada waktu pembuatan

- Tindakan `ec2:DeleteNetworkInterface` pada semua antarmuka jaringan dengan tag `VerifiedAccessManaged=true`
- Tindakan `ec2:ModifyNetworkInterfaceAttribute` pada semua grup keamanan dan semua antarmuka jaringan dengan tag `VerifiedAccessManaged=true`

Anda juga dapat melihat izin untuk kebijakan ini di Panduan Referensi Kebijakan AWS Terkelola; lihat AWSVPCVerifiedAccessServiceRolePolicy.

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (seperti pengguna, grup, atau peran) untuk membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat Izin peran tertaut layanan dalam Panduan Pengguna IAM.

Membuat peran terkait layanan untuk Akses Terverifikasi

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda memanggil CreateVerifiedAccessEndpoint, API AWS Management Console, atau AWS API AWS CLI, Akses Terverifikasi akan membuat peran terkait layanan untuk Anda.

Jika Anda menghapus peran terkait layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda menelepon CreateVerifiedAccessEndpointsekali lagi, Akses Terverifikasi akan membuat peran terkait layanan untuk Anda lagi.

Mengedit peran terkait layanan untuk Akses Terverifikasi

Akses Terverifikasi tidak memungkinkan Anda untuk mengedit peran AWSServiceRoleForVPCVerifiedAccess terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin merujuk peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit deskripsi peran terkait layanan di Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk Akses Terverifikasi

Anda tidak perlu menghapus peran AWSServiceRoleForVPCVerifiedAccess secara manual. Saat Anda memanggil DeleteVerifiedAccessEndpoint, API AWS Management Console AWS CLI, atau AWS API, Akses Terverifikasi membersihkan sumber daya dan menghapus peran terkait layanan untuk Anda.

Untuk menghapus peran tertaut layanan secara manual menggunakan IAM

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran AWSServiceRoleForVPCVerifiedAccess service-linked. Untuk informasi selengkapnya, lihat Menghapus peran terkait layanan di Panduan Pengguna IAM.

Wilayah yang Didukung untuk peran terkait layanan Akses Terverifikasi

Akses Terverifikasi mendukung penggunaan peran terkait layanan di semua Wilayah AWS tempat layanan tersedia. Untuk informasi selengkapnya, lihat AWS Wilayah dan titik akhir.

AWS kebijakan terkelola untuk Akses Terverifikasi

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: AWSVPCVerified AccessServiceRolePolicy

Kebijakan ini dilampirkan pada peran terkait layanan yang memungkinkan Akses Terverifikasi untuk melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Gunakan peran tertaut layanan. Untuk melihat izin kebijakan ini, Anda dapat melihat AWSVPCVerifiedAccessServiceRolePolicydi AWS Management Console, atau Anda dapat melihat AWSVPCVerifiedAccessServiceRolePolicykebijakan di Panduan Referensi Kebijakan AWS Terkelola.

Pembaruan Akses Terverifikasi ke kebijakan AWS terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola untuk Akses Terverifikasi sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS di halaman riwayat Dokumen Akses Terverifikasi.

Perubahan	Deskripsi	Tanggal
AWSVPCVerifiedAccessServiceRolePolicy- Kebijakan diperbarui	Akses Terverifikasi memperbarui kebijakan terkelolanya untuk menyertakan deskripsi semua tindakan di bawah bidang “sid”.	17 November 2023
AWSVPCVerifiedAccessServiceRolePolicy- Kebijakan diperbarui	Akses Terverifikasi memperbarui kebijakan terkelolanya untuk menambahkan sumber daya grup keamanan ke `ec2:CreateNetworkInterface` izin.	31 Mei 2023
AWSVPCVerifiedAccessServiceRolePolicy- Kebijakan baru	Akses Terverifikasi menambahkan kebijakan baru untuk memungkinkannya menyediakan sumber daya di akun Anda yang diperlukan untuk menggunakan layanan.	29 November 2022
Akses Terverifikasi mulai melacak perubahan	Akses Terverifikasi mulai melacak perubahan untuk kebijakan yang AWS dikelola.	29 November 2022

Validasi kepatuhan untuk Akses Terverifikasi

Akses Terverifikasi AWS dapat dikonfigurasi untuk mendukung kepatuhan Federal Information Processing Standards (FIPS). Untuk info dan detail selengkapnya tentang pengaturan kepatuhan FIPS untuk Akses Terverifikasi, buka. Kepatuhan FIPS untuk Akses Terverifikasi

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. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- Kepatuhan dan Tata Kelola Keamanan – Panduan implementasi solusi ini membahas pertimbangan arsitektur serta memberikan langkah-langkah untuk menerapkan fitur keamanan dan kepatuhan.
- Referensi Layanan yang Memenuhi Syarat HIPAA — Daftar layanan yang memenuhi syarat HIPAA. Tidak semua memenuhi Layanan AWS syarat HIPAA.
- AWS Sumber Daya AWS — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- AWS Panduan Kepatuhan Pelanggan - Memahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan ini merangkum praktik terbaik untuk mengamankan Layanan AWS dan memetakan panduan untuk kontrol keamanan di berbagai kerangka kerja (termasuk Institut Standar dan Teknologi Nasional (NIST), Dewan Standar Keamanan Industri Kartu Pembayaran (PCI), dan Organisasi Internasional untuk Standardisasi (ISO)).
- Mengevaluasi Sumber Daya dengan Aturan dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- AWS Security Hub— Ini Layanan AWS memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS. Security Hub menggunakan kontrol keamanan untuk sumber daya AWS Anda serta untuk memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat Referensi kontrol Security Hub.
- Amazon GuardDuty — Ini Layanan AWS mendeteksi potensi ancaman terhadap beban kerja Akun AWS, kontainer, dan data Anda dengan memantau lingkungan Anda untuk aktivitas mencurigakan dan berbahaya. GuardDuty dapat membantu Anda mengatasi berbagai persyaratan kepatuhan, seperti PCI DSS, dengan memenuhi persyaratan deteksi intrusi yang diamanatkan oleh kerangka kerja kepatuhan tertentu.
- AWS Audit ManagerIni Layanan AWS membantu Anda terus mengaudit AWS penggunaan Anda untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap peraturan dan standar industri.

Ketahanan dalam Akses Terverifikasi

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 dengan latensi rendah, throughput tinggi, dan jaringan yang 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.

Untuk informasi selengkapnya tentang Wilayah AWS dan Availability Zone, lihat Infrastruktur AWS Global.

Selain infrastruktur AWS global, Akses Terverifikasi menawarkan fitur berikut untuk membantu mendukung kebutuhan ketersediaan Anda yang tinggi.

Beberapa subnet untuk ketersediaan tinggi

Saat Anda membuat titik akhir Akses Terverifikasi tipe penyeimbang beban, Anda dapat mengaitkan beberapa subnet ke titik akhir. Setiap subnet yang Anda kaitkan dengan endpoint harus dimiliki oleh Availability Zone yang berbeda. Dengan mengaitkan beberapa subnet, Anda dapat memastikan ketersediaan tinggi dengan menggunakan beberapa Availability Zone.

Pemantauan Akses Terverifikasi AWS

Pemantauan merupakan bagian penting dari menjaga keandalan, ketersediaan, dan kinerja Akses Terverifikasi AWS. AWS menyediakan alat pemantauan berikut untuk menonton Akses Terverifikasi, melaporkan ketika ada sesuatu yang salah, dan mengambil tindakan otomatis bila perlu:

- Akses log - Menangkap informasi terperinci tentang permintaan untuk mengakses aplikasi. Untuk informasi selengkapnya, lihat the section called “Log Akses Terverifikasi”.
- AWS CloudTrail— Menangkap panggilan API dan peristiwa terkait yang dibuat oleh atau atas nama Anda Akun AWS dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun mana yang dipanggil AWS, alamat IP sumber dari mana panggilan dilakukan, dan kapan panggilan terjadi. Untuk informasi selengkapnya, lihat the section called “CloudTrail log”.

Log Akses Terverifikasi

Setelah Akses Terverifikasi AWS mengevaluasi setiap permintaan akses, ia mencatat semua upaya akses. Ini memberi Anda visibilitas terpusat ke dalam akses aplikasi, dan membantu Anda dengan cepat menanggapi insiden keamanan dan permintaan audit. Akses Terverifikasi mendukung format pencatatan Open Cybersecurity Schema Framework (OCSF).

Ketika Anda mengaktifkan logging, Anda perlu mengkonfigurasi tujuan untuk log yang akan dikirim. Prinsipal IAM yang digunakan untuk mengonfigurasi tujuan logging harus memiliki izin tertentu agar logging berfungsi dengan baik. Izin IAM yang diperlukan untuk setiap tujuan pencatatan dapat dilihat di bagian ini. Izin pencatatan Akses Terverifikasi Akses Terverifikasi mendukung tujuan berikut untuk menerbitkan log akses:

- Grup CloudWatch log Amazon Logs
- Bucket Amazon S3
- Aliran pengiriman Amazon Data Firehose

Daftar Isi

- Versi logging Akses Terverifikasi
- Izin pencatatan Akses Terverifikasi
- Mengaktifkan atau menonaktifkan log Akses Terverifikasi

- Mengaktifkan atau menonaktifkan konteks kepercayaan Akses Terverifikasi
- Contoh log OCSF versi 0.1 untuk Akses Terverifikasi
- Contoh log OCSF versi 1.0.0-rc.2 untuk Akses Terverifikasi

Versi logging Akses Terverifikasi

Secara default, sistem logging Akses Terverifikasi menggunakan Open Cybersecurity Schema Framework (OCSF) versi 0.1. Untuk contoh log yang menggunakan versi 0.1 lihatContoh log OCSF versi 0.1 untuk Akses Terverifikasi.

Versi logging terbaru kompatibel dengan OCSF versi 1.0.0-rc.2. Untuk informasi lebih lanjut tentang skema, lihat Skema OCSF. Untuk contoh log yang menggunakan versi 1.0.0-rc.2, lihat. Contoh log OCSF versi 1.0.0-rc.2 untuk Akses Terverifikasi

Perhatikan bahwa Anda tidak dapat menggunakan OCSF versi 0.1 jika titik akhir Akses Terverifikasi menggunakan protokol TCP.

Untuk memutakhirkan versi logging menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi yang sesuai.
4. Pada tab konfigurasi pencatatan instans Akses Terverifikasi, pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.
5. Pilih ocsf-1.0.0-rc.2 dari daftar drop-down Perbarui versi log.
6. Pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.

Untuk memutakhirkan versi logging menggunakan AWS CLI

Gunakan perintah modify-verified-access-instance-logging-configuration.

Izin pencatatan Akses Terverifikasi

Prinsipal IAM yang digunakan untuk mengonfigurasi tujuan logging harus memiliki izin tertentu agar logging berfungsi dengan baik. Bagian berikut menunjukkan izin yang diperlukan untuk setiap tujuan pencatatan.

Untuk pengiriman ke CloudWatch Log:

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration`pada instance Akses Terverifikasi
- `logs:CreateLogDelivery`,`logs:DeleteLogDelivery`,`logs:GetLogDelivery`,`logs:ListLogDe` dan `logs:UpdateLogDelivery` pada semua sumber daya
- `logs:DescribeLogGroups`,`logs:DescribeResourcePolicies`, dan `logs:PutResourcePolicy` pada grup log tujuan

Untuk pengiriman ke Amazon S3:

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration`pada instance Akses Terverifikasi
- `logs:CreateLogDelivery`,`logs:DeleteLogDelivery`,`logs:GetLogDelivery`,`logs:ListLogDe` dan `logs:UpdateLogDelivery` pada semua sumber daya
- `s3:GetBucketPolicy`dan `s3:PutBucketPolicy` di ember tujuan

Untuk pengiriman ke Firehose:

- `ec2:ModifyVerifiedAccessInstanceLoggingConfiguration`pada instance Akses Terverifikasi
- `firehose:TagDeliveryStream` di semua sumber daya
- `iam:CreateServiceLinkedRole` di semua sumber daya
- `logs:CreateLogDelivery`,`logs:DeleteLogDelivery`,`logs:GetLogDelivery`,`logs:ListLogDe` dan `logs:UpdateLogDelivery` pada semua sumber daya

Mengaktifkan atau menonaktifkan log Akses Terverifikasi

Anda dapat menggunakan prosedur di bagian ini untuk mengaktifkan atau menonaktifkan logging. Ketika Anda mengaktifkan logging, Anda perlu mengkonfigurasi tujuan untuk log yang akan dikirim. Prinsipal IAM yang digunakan untuk mengonfigurasi tujuan logging harus memiliki izin tertentu agar logging berfungsi dengan baik. Izin IAM yang diperlukan untuk setiap tujuan pencatatan dapat dilihat di bagian ini. Izin pencatatan Akses Terverifikasi

Daftar Isi

- Aktifkan log akses
- Nonaktifkan log akses

Aktifkan log akses

Untuk mengaktifkan log Akses Terverifikasi

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi.
4. Pada tab konfigurasi pencatatan instans Akses Terverifikasi, pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.
5. (Opsional) Untuk menyertakan data kepercayaan yang dikirim dari penyedia kepercayaan di log, lakukan hal berikut:
   a. Pilih ocsf-1.0.0-rc.2 dari daftar drop-down Perbarui versi log.
   b. Pilih Sertakan konteks kepercayaan.
6. Lakukan salah satu hal berikut ini:
   - Aktifkan Kirim ke CloudWatch Log Amazon. Pilih grup log tujuan.
   - Aktifkan Kirim ke Amazon S3. Masukkan nama, pemilik, dan awalan bucket tujuan.
   - Nyalakan Kirim ke Firehose. Pilih aliran pengiriman tujuan.
7. Pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.

Untuk mengaktifkan log Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-instance-logging-configuration.

Nonaktifkan log akses

Anda dapat menonaktifkan log akses untuk instans Akses Terverifikasi kapan saja. Setelah Anda menonaktifkan log akses, data log Anda tetap berada di tujuan log Anda sampai Anda menghapusnya.

Untuk menonaktifkan log Akses Terverifikasi

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/

2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi.
4. Pada tab konfigurasi pencatatan instans Akses Terverifikasi, pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.
5. Matikan pengiriman log.
6. Pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.

Untuk menonaktifkan log Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-instance-logging-configuration.

Mengaktifkan atau menonaktifkan konteks kepercayaan Akses Terverifikasi

Konteks kepercayaan yang dikirim dari penyedia kepercayaan Anda dapat diaktifkan secara opsional untuk dimasukkan dalam log Akses Terverifikasi Anda. Ini dapat berguna saat menentukan kebijakan yang mengizinkan atau menolak akses ke aplikasi Anda. Setelah Anda mengaktifkannya, konteks kepercayaan ditemukan di log di bawah `data` bidang. Jika konteks kepercayaan dinonaktifkan, `data` bidang disetel ke`null`. Untuk mengonfigurasi Akses Terverifikasi untuk menyertakan konteks kepercayaan dalam log, lakukan prosedur berikut.

> **Note**
>
> Menyertakan konteks kepercayaan dalam log Akses Terverifikasi Anda memerlukan peningkatan ke versi `ocsf-1.0.0-rc.2` logging terbaru. Prosedur berikut mengasumsikan bahwa Anda sudah mengaktifkan logging. Jika itu tidak benar, lihat Aktifkan log akses prosedur lengkapnya.

Daftar Isi

- Aktifkan konteks kepercayaan
- Nonaktifkan konteks kepercayaan

Aktifkan konteks kepercayaan

Untuk menyertakan konteks kepercayaan dalam log Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/

2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi yang sesuai.
4. Pada tab konfigurasi pencatatan instans Akses Terverifikasi, pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.
5. Pilih ocsf-1.0.0-rc.2 dari daftar drop-down Perbarui versi log.
6. Aktifkan Sertakan konteks kepercayaan.
7. Pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.

Untuk menyertakan konteks kepercayaan dalam log Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-instance-logging-configuration.

Nonaktifkan konteks kepercayaan

Jika Anda tidak lagi ingin memasukkan konteks kepercayaan dalam log, Anda dapat menghapusnya dengan melakukan prosedur berikut.

Untuk menghapus konteks kepercayaan dari log Akses Terverifikasi menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, pilih instans Akses Terverifikasi.
3. Pilih instance Akses Terverifikasi yang sesuai.
4. Pada tab konfigurasi pencatatan instans Akses Terverifikasi, pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.
5. Matikan Sertakan konteks kepercayaan.
6. Pilih Ubah konfigurasi pencatatan instans Akses Terverifikasi.

Untuk menghapus konteks kepercayaan dari log Akses Terverifikasi menggunakan AWS CLI

Gunakan perintah modify-verified-access-instance-logging-configuration.

Contoh log OCSF versi 0.1 untuk Akses Terverifikasi

Berikut ini adalah contoh log menggunakan OCSF versi 0.1.

Contoh

- Akses diberikan dengan OIDC

- Akses diberikan dengan OIDC dan JAMF
- Akses diberikan dengan OIDC dan CrowdStrike
- Akses ditolak karena cookie yang hilang
- Akses ditolak oleh kebijakan
- Entri log tidak dikenal

Akses diberikan dengan OIDC

Dalam entri log contoh ini, Akses Terverifikasi memungkinkan akses ke titik akhir dengan penyedia kepercayaan pengguna OIDC.


```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
```

```
    "identity": {
        "authorizations": [
            {
                "decision": "Allow",
                "policy": {
                    "name": "inline"
                }
            }
        ],
        "idp": {
            "name": "user",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "00u6wj48lbxTAEXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-63748362-6408d24241120b942EXAMPLE",
        "logged_time": 1668580281337,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:29:54.344948Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-002fa341aeEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.24.57.68",
        "port": "48234"
    },
    "start_time": "1668580194340",
```

```
    "status_code": "100",
    "status_details": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "20800101",
    "type_name": "AccessLogs: Access Granted",
    "unmapped": null
}
```

Akses diberikan dengan OIDC dan JAMF

Dalam entri log contoh ini, Akses Terverifikasi memungkinkan akses ke titik akhir dengan penyedia kepercayaan perangkat OIDC dan JAMF.

```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0,
        "uid": "41b07859-4222-4f41-f3b9-97dc1EXAMPLE"
    },
    "duration": "0.347",
    "end_time": "1668804944086",
    "time": "1668804944086",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "h2",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
        "version": "HTTP/2.0"
    },
```

```
    "http_response": {
        "code": 304
    },
    "identity": {
        "authorizations": [
            {
                "decision": "Allow",
                "policy": {
                    "name": "inline"
                }
            }
        ],
        "idp": {
            "name": "oidc",
            "uid": "vatp-9778003bc2EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "4f040d0f96becEXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-321318ce-6100d340adf4fb29dEXAMPLE",
        "logged_time": 1668805278555,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-18T20:55:44.086480Z",
    "proxy": {
        "ip": "10.5.192.96",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-3598f66575EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "192.168.20.246",
```

```
        "port": 61769
    },
    "start_time": "1668804943739",
    "status_code": "100",
    "status_details": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "20800101",
    "type_name": "AccessLogs: Access Granted",
    "unmapped": null
}
```

Akses diberikan dengan OIDC dan CrowdStrike

Dalam entri log contoh ini, Akses Terverifikasi memungkinkan akses ke titik akhir dengan OIDC dan penyedia kepercayaan CrowdStrike perangkat.

```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.173.3",
        "os": {
            "name": "Windows 11",
            "type": "Windows",
            "type_id": 100
        },
        "type": "Unknown",
        "type_id": 0,
        "uid": "122978434f65093aee5dfbdc0EXAMPLE",
        "hw_info": {
            "serial_number": "751432a1-d504-fd5e-010d-5ed11EXAMPLE"
        }
    },
    "duration": "0.028",
    "end_time": "1668816620842",
    "time": "1668816620842",
    "http_request": {
        "http_method": "GET",
```

```
        "url": {
            "hostname": "test.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "h2",
            "text": "https://test.app.example.com:443/"
        },
        "user_agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 304
    },
    "identity": {
        "authorizations": [
            {
                "decision": "Allow",
                "policy": {
                    "name": "inline"
                }
            }
        ],
        "idp": {
            "name": "oidc",
            "uid": "vatp-506d9753f6EXAMPLE"
        },
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "23bb45b16a389EXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-c16c5a65-b641e4056cc6cb0eeEXAMPLE",
        "logged_time": 1668816977134,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
```

```
    "ref_time": "2022-11-19T00:10:20.842295Z",
    "proxy": {
        "ip": "192.168.144.62",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-2f80f37e64EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "10.14.173.3",
        "port": 55706
    },
    "start_time": "1668816620814",
    "status_code": "100",
    "status_details": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "20800101",
    "type_name": "AccessLogs: Access Granted",
    "unmapped": null
}
```

Akses ditolak karena cookie yang hilang

Dalam entri log contoh ini, Akses Terverifikasi menolak akses karena cookie otentikasi hilang.

```
{
    "activity": "Access Denied",
    "activity_id": "2",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": null,
    "duration": "0.0",
    "end_time": "1668593568259",
    "time": "1668593568259",
    "http_request": {
        "http_method": "POST",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/dns-query",
```

```
            "port": 443,
            "scheme": "h2",
            "text": "https://hello.app.example.com:443/dns-query"
        },
        "user_agent": "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML",
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 302
    },
    "identity": null,
    "message": "",
    "metadata": {
        "uid": "Root=1-5cf1c832-a565309ce20cc7dafEXAMPLE",
        "logged_time": 1668593776720,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T10:12:48.259762Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-108ed7a672EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "10.7.178.16",
        "port": "46246"
    },
    "start_time": "1668593568258",
    "status_code": "200",
    "status_details": "Authentication Denied",
    "status_id": "2",
    "status": "Failure",
    "type_uid": "20800102",
    "type_name": "AccessLogs: Access Denied",
    "unmapped": null
}
```

Akses ditolak oleh kebijakan

Dalam entri log contoh ini, Akses Terverifikasi menolak permintaan yang diautentikasi karena permintaan tidak diizinkan oleh kebijakan akses.


```
{
    "activity": "Access Denied",
    "activity_id": "2",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.4.133.137",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.023",
    "end_time": "1668573630978",
    "time": "1668573630978",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "h2",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
        "version": "HTTP/2.0"
    },
    "http_response": {
        "code": 401
    },
    "identity": {
        "authorizations": [],
        "idp": {
            "name": "user",
            "uid": "vatp-e048b3e0f8EXAMPLE"
        },
        "user": {
```

```
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "0e1281ad3580aEXAMPLE"
        }
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-531a036a-09e95794c7b96aefbEXAMPLE",
        "logged_time": 1668573773753,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T04:40:30.978732Z",
    "proxy": {
        "ip": "3.223.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-021d5eaed2EXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "10.4.133.137",
        "port": "31746"
    },
    "start_time": "1668573630955",
    "status_code": "300",
    "status_details": "Authorization Denied",
    "status_id": "2",
    "status": "Failure",
    "type_uid": "20800102",
    "type_name": "AccessLogs: Access Denied",
    "unmapped": null
}
```

Entri log tidak dikenal

Dalam entri log contoh ini, Akses Terverifikasi tidak dapat menghasilkan entri log lengkap sehingga memancarkan entri log yang tidak dikenal. Ini memastikan bahwa setiap permintaan muncul di log akses.


```
{
    "activity": "Unknown",
    "activity_id": "0",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": null,
    "duration": "0.004",
    "end_time": "1668580207898",
    "time": "1668580207898",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
    "identity": null,
    "message": "",
    "metadata": {
        "uid": "Root=1-435eb955-6b5a1d529343f5adaEXAMPLE",
        "logged_time": 1668580579147,
        "version": "0.1",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
```

```
    "ref_time": "2022-11-16T06:30:07.898344Z",
    "proxy": {
        "ip": "10.1.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-6c32b53b3cEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.28.57.68",
        "port": "47220"
    },
    "start_time": "1668580207893",
    "status_code": "000",
    "status_details": "Unknown",
    "status_id": "0",
    "status": "Unknown",
    "type_uid": "20800100",
    "type_name": "AccessLogs: Unknown",
    "unmapped": null
}
```


Contoh log OCSF versi 1.0.0-rc.2 untuk Akses Terverifikasi

Berikut ini adalah contoh log menggunakan OCSF versi 1.0.0-rc.2.

Contoh

- Akses yang diberikan dengan konteks kepercayaan disertakan
- Akses yang diberikan dengan konteks kepercayaan dihilangkan
- Tetapkan hak istimewa dengan titik akhir CIDR jaringan

Akses yang diberikan dengan konteks kepercayaan disertakan


```
{
    "activity_name": "Access Grant",
    "activity_id": "1",
    "actor": {
        "authorizations": [{
            "decision": "Allow",
            "policy": {
```

```
                "name": "inline"
            }
        }],
        "idp": {
            "name": "user",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
        "invoked_by": "",
        "process": {},
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "00u6wj48lbxTAEXAMPLE"
        },
        "session": {}
    },
    "category_name": "Audit Activity",
    "category_uid": "3",
    "class_name": "Access Activity",
    "class_uid": "3006",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
```

```
    "message": "",
    "metadata": {
        "uid": "Root=1-63748362-6408d24241120b942EXAMPLE",
        "logged_time": 1668580281337,
        "version": "1.0.0-rc.2",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:29:54.344948Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-002fa341aeEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.24.57.68",
        "port": "48234"
    },
    "start_time": "1668580194340",
    "status_code": "100",
    "status_detail": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "300601",
    "type_name": "Access Activity: Access Grant",
    "data": {
        "context": {
            "oidc": {
                "family_name": "Last",
                "zoneinfo": "America/Los_Angeles",
                "exp": 1670631145,
                "middle_name": "Middle",
                "given_name": "First",
                "email_verified": true,
                "name": "Test User Display",
                "updated_at": 1666305953,
                "preferred_username": "johndoe-user@test.com",
                "profile": "http://www.example.com",
                "locale": "US",
```

```
                "nickname": "Tester",
                "email": "johndoe-user@test.com",
                "additional_user_context": {
                    "aud": "xxx",
                    "exp": 1000000000,
                    "groups": [
                        "group-id-1",
                        "group-id-2"
                    ],
                    "iat": 1000000000,
                    "iss": "https://oidc-tp.com/",
                    "sub": "xyzsubject",
                    "ver": "1.0"
                }
            },
            "http_request": {
                "x_forwarded_for": "1.1.1.1,2.2.2.2",
                "http_method": "GET",
                "user_agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7)
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36",
                "port": "80",
                "hostname": "hostname.net"
            }
        }
    }
}
```

Akses yang diberikan dengan konteks kepercayaan dihilangkan

```
{
    "activity_name": "Access Grant",
    "activity_id": "1",
    "actor": {
        "authorizations": [{
            "decision": "Allow",
            "policy": {
                "name": "inline"
            }
        }],
        "idp": {
            "name": "user",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
```

```
        "invoked_by": "",
        "process": {},
        "user": {
            "email_addr": "johndoe@example.com",
            "name": "Test User Display",
            "uid": "johndoe@example.com",
            "uuid": "00u6wj48lbxTAEXAMPLE"
        },
        "session": {}
    },
    "category_name": "Audit Activity",
    "category_uid": "3",
    "class_name": "Access Activity",
    "class_uid": "3006",
    "device": {
        "ip": "10.2.7.68",
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "http_request": {
        "http_method": "GET",
        "url": {
            "hostname": "hello.app.example.com",
            "path": "/",
            "port": 443,
            "scheme": "https",
            "text": "https://hello.app.example.com:443/"
        },
        "user_agent": "python-requests/2.28.1",
        "version": "HTTP/1.1"
    },
    "http_response": {
        "code": 200
    },
    "message": "",
    "metadata": {
        "uid": "Root=1-63748362-6408d24241120b942EXAMPLE",
        "logged_time": 1668580281337,
        "version": "1.0.0-rc.2",
        "product": {
            "name": "Verified Access",
```

```
            "vendor_name": "AWS"
        }
    },
    "ref_time": "2022-11-16T06:29:54.344948Z",
    "proxy": {
        "ip": "192.168.34.167",
        "port": 443,
        "svc_name": "Verified Access",
        "uid": "vai-002fa341aeEXAMPLE"
    },
    "severity": "Informational",
    "severity_id": "1",
    "src_endpoint": {
        "ip": "172.24.57.68",
        "port": "48234"
    },
    "start_time": "1668580194340",
    "status_code": "100",
    "status_detail": "Access Granted",
    "status_id": "1",
    "status": "Success",
    "type_uid": "300601",
    "type_name": "Access Activity: Access Grant",
    "data": null
}
```

Tetapkan hak istimewa dengan titik akhir CIDR jaringan

```
{
    "activity_id": "1",
    "activity_name": "Assign Privileges",
    "category_name": "Audit Activity",
    "category_uid": "3",
    "class_name": "Authorization",
    "class_uid": "3003",
    "data": {
        "endpoint_type": "cidr",
        "protocol": "tcp",
        "access_path": "public",
        "idp": {
            "name": "my-oidc-instance",
            "uid": "vatp-09bc4cbce2EXAMPLE"
        },
```

```
        "authorizations": [{
            "decision": "Allow",
            "policy": {
                "name": "inline"
            }
        }],
        "context": {
            "oidc": {
                "family_name": "Last",
                "zoneinfo": "America/Los_Angeles",
                "exp": 1670631145,
                "middle_name": "Middle",
                "given_name": "First",
                "email_verified": true,
                "name": "Test User Display",
                "updated_at": 1666305953,
                "preferred_username": "johndoe-user@test.com",
                "profile": "http://www.example.com",
                "locale": "US",
                "nickname": "Tester",
                "email": "johndoe-user@test.com",
                "additional_user_context": {
                    "aud": "xxx",
                    "exp": 1000000000,
                    "groups": [
                        "group-id-1",
                        "group-id-2"
                    ],
                    "iat": 1000000000,
                    "iss": "https://oidc-tp.com/",
                    "sub": "xyzsubject",
                    "ver": "1.0"
                }
            },
            "tcp_flow": {
                "destination_ip": "10.0.0.1",
                "destination_port": 22,
                "client_ip": "10.2.7.68"
            }
        }
    },
    "device": {
        "ip":  "10.2.7.68",
        "port": 1002,
```

```
        "type": "Unknown",
        "type_id": 0
    },
    "duration": "0.004",
    "end_time": "1668580194344",
    "time": "1668580194344",
    "metadata": {
        "uid": "",
        "logged_time": 1668580281337,
        "version": "1.0.0-rc.2",
        "product": {
            "name": "Verified Access",
            "vendor_name": "AWS"
        }
    },
    "severity": "Informational",
    "severity_id": "1",
    "start_time": "1668580194340",
    "status_code": "200",
    "status_id": "1",
    "status": "Success",
    "type_uid": "300301",
    "type_name": "Authorization: Assign Privileges",
    "count": 1,
    "dst_endpoint": {
        "ip": "107.22.231.155",
        "port": 22
    },
    "privileges": [
        "vae-12345cbce2EXAMPLE"
    ],
    "user": {
        "email_addr": "johndoe-user@test.com",
        "uid": "johndoe-user",
        "uuid": "9bcce02a-fc15-4091-a0b7-874d157c67b8"
    }
}
```

Log panggilan API Akses Terverifikasi menggunakan AWS CloudTrail

AWS Akses Terverifikasi terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau Layanan AWS dalam Akses Terverifikasi. CloudTrail menangkap panggilan API untuk Akses Terverifikasi sebagai peristiwa. Panggilan yang diambil termasuk panggilan dari konsol Akses Terverifikasi dan panggilan kode ke operasi API Akses Terverifikasi. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat untuk Akses Terverifikasi, alamat IP dari mana permintaan dibuat, kapan dibuat, dan detail tambahan.

Setiap entri peristiwa atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut hal ini:

- Baik permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna.
- Apakah permintaan dibuat atas nama pengguna IAM Identity Center.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

CloudTrail aktif di Anda Akun AWS ketika Anda membuat akun dan Anda secara otomatis memiliki akses ke riwayat CloudTrail Acara. Riwayat CloudTrail Acara menyediakan catatan yang dapat dilihat, dapat dicari, dapat diunduh, dan tidak dapat diubah dari 90 hari terakhir dari peristiwa manajemen yang direkam dalam file. Wilayah AWS Untuk informasi selengkapnya, lihat Bekerja dengan riwayat CloudTrail Acara di Panduan AWS CloudTrail Pengguna. Tidak ada CloudTrail biaya untuk melihat riwayat Acara.

Untuk catatan acara yang sedang berlangsung dalam 90 hari Akun AWS terakhir Anda, buat jejak atau penyimpanan data acara CloudTrailDanau.

CloudTrail jalan setapak

> Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Semua jalur yang dibuat menggunakan AWS Management Console Multi-region. Anda dapat membuat jalur Single-region atau Multi-region dengan menggunakan. AWS CLI Membuat jejak Multi-wilayah disarankan karena Anda menangkap aktivitas Wilayah AWS di semua akun Anda. Jika Anda membuat jejak wilayah Tunggal, Anda hanya dapat melihat peristiwa yang dicatat di jejak.

Wilayah AWS Untuk informasi selengkapnya tentang jejak, lihat Membuat jejak untuk Anda Akun AWS dan Membuat jejak untuk organisasi di Panduan AWS CloudTrail Pengguna.

Anda dapat mengirimkan satu salinan acara manajemen yang sedang berlangsung ke bucket Amazon S3 Anda tanpa biaya CloudTrail dengan membuat jejak, namun, ada biaya penyimpanan Amazon S3. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga. Untuk informasi tentang harga Amazon S3, lihat Harga Amazon S3.

CloudTrail Penyimpanan data acara danau

CloudTrail Lake memungkinkan Anda menjalankan kueri berbasis SQL pada acara Anda. CloudTrail Lake mengonversi peristiwa yang ada dalam format JSON berbasis baris ke format Apache ORC. ORC adalah format penyimpanan kolumnar yang dioptimalkan untuk pengambilan data dengan cepat. Peristiwa digabungkan ke dalam penyimpanan data peristiwa, yang merupakan kumpulan peristiwa yang tidak dapat diubah berdasarkan kriteria yang Anda pilih dengan menerapkan pemilih acara tingkat lanjut. Penyeleksi yang Anda terapkan ke penyimpanan data acara mengontrol peristiwa mana yang bertahan dan tersedia untuk Anda kueri. Untuk informasi lebih lanjut tentang CloudTrail Danau, lihat Bekerja dengan AWS CloudTrail Danau di Panduan AWS CloudTrail Pengguna.

CloudTrail Penyimpanan data acara danau dan kueri menimbulkan biaya. Saat Anda membuat penyimpanan data acara, Anda memilih opsi harga yang ingin Anda gunakan untuk penyimpanan data acara. Opsi penetapan harga menentukan biaya untuk menelan dan menyimpan peristiwa, dan periode retensi default dan maksimum untuk penyimpanan data acara. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga.

Acara manajemen Akses Terverifikasi

Acara manajemen memberikan informasi tentang operasi manajemen yang dilakukan pada sumber daya di Anda Akun AWS. Ini juga dikenal sebagai operasi pesawat kontrol. Secara default, CloudTrail mencatat peristiwa manajemen.

Akses terverifikasi mencatat operasi rencana kontrol sebagai peristiwa manajemen. Untuk daftar, lihat Referensi Amazon EC2 API.

Contoh acara Akses Terverifikasi

Contoh berikut menunjukkan CloudTrail peristiwa yang menunjukkan `CreateVerifiedAccessInstance` tindakan.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAIKK4OOINJWEXAMPLE:jdoe",
        "arn": "arn:aws:iam::123456789012:user/jdoe",
        "accountId": "123456789012",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "jdoe"
    },
    "eventTime": "2022-11-18T20:44:04Z",
    "eventSource": "ec2.amazonaws.com",
    "eventName": "CreateVerifiedAccessInstance",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "198.51.100.1",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "CreateVerifiedAccessInstanceRequest": {
            "Description": "",
            "ClientToken": "85893b1e-49f6-4d24-97de-280c664edf1b"
        }
    },
    "responseElements": {
        "CreateVerifiedAccessInstanceResponse": {
            "verifiedAccessInstance": {
                "creationTime": "2022-11-18T20:44:04",
                "description": "",
                "verifiedAccessInstanceId": "vai-0d79d91875542c549",
                "verifiedAccessTrustProviderSet": ""
            },
            "requestId": "2eae195d-6bfd-46d7-b46e-a68cb791de09"
        }
    },
    "requestID": "2eae195d-6bfd-46d7-b46e-a68cb791de09",
    "eventID": "297d6529-1144-40f6-abf8-3a76f18d88f0",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Untuk informasi tentang konten CloudTrail rekaman, lihat konten CloudTrail rekaman di Panduan AWS CloudTrail Pengguna.

Kuota untuk Akses Terverifikasi AWS

Anda Akun AWS memiliki kuota default, sebelumnya disebut sebagai batas, untuk masing-masing. Layanan AWS Kecuali dinyatakan lain, setiap kuota bersifat khusus per Wilayah.

Akun AWS-kuota tingkat

Anda Akun AWS memiliki kuota berikut yang terkait dengan Akses Terverifikasi.

Nama	Default	Dapat disesuaikan	Deskripsi
Instans Akses Terverifikasi	5	Ya	Jumlah maksimum Instans Akses Terverifikasi yang dapat dibuat pelanggan di Wilayah saat ini.
Grup Akses Terverifikasi	10	Ya	Jumlah maksimum Grup Akses Terverifikasi yang dapat dibuat pelanggan di Wilayah saat ini.
Penyedia Kepercayaan Akses Terverifikasi	15	Ya	Jumlah maksimum Penyedia Kepercayaan Akses Terverifikasi yang dapat dibuat pelanggan di Wilayah saat ini.
Titik Akhir Akses Terverifikasi	50	Ya	Jumlah maksimum Titik Akhir Akses Terverifikasi yang dapat dibuat pelanggan di Wilayah saat ini.

Header HTTP

Berikut ini adalah batas ukuran untuk header HTTP.

Nama	Default	Dapat disesuaikan
Baris permintaan	16 K	Tidak
Header tunggal	16 K	Tidak

Nama	Default	Dapat disesuaikan
Seluruh header respons	32 K	Tidak
Seluruh header permintaan	64 K	Tidak

Lalu lintas HTTP

Batas waktu idle koneksi adalah 60 detik. Jika aplikasi membutuhkan waktu lebih dari 60 detik untuk menanggapi permintaan HTTP, klien menerima kesalahan batas waktu gateway HTTP 504. Jika log Akses Terverifikasi diaktifkan, kami mencatat kesalahan HTTP 504 apa pun.

Ukuran klaim OIDC

Berikut ini adalah batas ukuran klaim OIDC.

Nama	Default	Dapat disesuaikan
Ukuran klaim OIDC	11 K	Tidak

Pusat Identitas IAM

Akses Terverifikasi dapat menyediakan akses ke pengguna di Pusat Identitas IAM yang ditetapkan hingga 1.000 grup.

Riwayat dokumen untuk Panduan Pengguna Akses Terverifikasi

Tabel berikut menjelaskan rilis dokumentasi untuk Akses Terverifikasi.

Perubahan	Deskripsi	Tanggal
Support untuk token akses dalam konteks kepercayaan	Update `additional_user_context` untuk menambah klaim pengguna OIDC.	Februari 24, 2025
Support untuk sumber daya melalui protokol non-HTTP	Pelepasan akses ke sumber daya melalui protokol non-HTTP.	Februari 5, 2025
Rilis pratinjau	Pratinjau rilis akses ke sumber daya melalui protokol non-HTTP.	Desember 1, 2024
AWS kebijakan terkelola diperbarui	Pembaruan dibuat untuk kebijakan IAM AWS terkelola untuk Akses Terverifikasi.	17 November 2023
Enkripsi data saat istirahat	AWS Akses Terverifikasi mengenkripsi data saat istirahat secara default, menggunakan kunci KMS yang AWS dimiliki.	28 September 2023
Support untuk kepatuhan FIPS	Konfigurasikan Akses Terverifikasi untuk kepatuhan FIPS.	26 September 2023
Penebangan yang ditingkatkan	Penambahan fitur logging yang menambahkan konteks kepercayaan ke log.	19 Juni 2023

AWS kebijakan terkelola diperbarui	Pembaruan dibuat untuk kebijakan IAM AWS terkelola untuk Akses Terverifikasi.	31 Mei 2023
Rilis GA	Rilis GA dari Panduan Pengguna Akses Terverifikasi. Termasuk AWS WAF integrasi .	27 April 2023
Rilis pratinjau	Pratinjau rilis Panduan Pengguna Akses Terverifikasi	29 November 2022

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.