Panduan Pengguna

AWS Akses Terverifikasi

AWS Akses Terverifikasi: Panduan Pengguna

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 tutorial Akses Terverifikasi 6
Buatlah sebuah instans 7
Konfigurasikan penyedia kepercayaan 8
Lampirkan penyedia kepercayaan Anda ke instans 8
Membuat grup 9
Bagikan grup Anda melalui AWS RAM 9
Tambahkan aplikasi Anda dengan membuat titik akhir 10
Konfigurasikan DNS pengaturan untuk titik akhir 11
Uji konektivitas ke aplikasi 12
Mengonfigurasi kebijakan akses tingkat grup 12
Uji ulang konektivitas ke aplikasi 12
Bersihkan 12
Instans Akses Terverifikasi 14
Membuat dan mengelola instance Akses Terverifikasi 14
Buat instance Akses Terverifikasi 14
Lampirkan penyedia kepercayaan ke instans Akses Terverifikasi 15
Lepaskan penyedia kepercayaan dari instans Akses Terverifikasi 15
Menghapus instans Akses Terverifikasi 16
Integrasikan Akses Terverifikasi dengan AWS WAF 16
IAMizin yang diperlukan untuk mengintegrasikan Akses Terverifikasi dengan AWS WAF 17
Kaitkan AWS WAF web ACL 17
Periksa status AWS WAF integrasi 18
Putuskan hubungan web AWS WAF ACL 19
FIPSkepatuhan 19
Lingkungan yang ada 20
Lingkungan baru 20
Penyedia kepercayaan 22

Identitas pengguna 22
IAMPusat Identitas 22
OIDCpenyedia kepercayaan 24
Berbasis perangkat 27
Penyedia kepercayaan perangkat yang didukung 28
Buat penyedia kepercayaan berbasis perangkat 28
Memodifikasi penyedia kepercayaan berbasis perangkat 29
Menghapus penyedia kepercayaan berbasis perangkat 30
Grup Akses Terverifikasi 31
Membuat grup Akses Terverifikasi 31
Ubah kebijakan grup Akses Terverifikasi 32
Menghapus grup Akses Terverifikasi 32
Titik akhir Akses Terverifikasi 33
Jenis titik akhir Akses Terverifikasi 33
Cara kerja Akses Terverifikasi dengan berbagi VPCs dan subnet 33
Buat titik akhir penyeimbang beban 34
Buat titik akhir antarmuka jaringan 35
Izinkan lalu lintas dari titik akhir Anda 37
Ubah titik akhir Akses Terverifikasi 37
Ubah kebijakan titik akhir Akses Terverifikasi 38
Menghapus titik akhir Akses Terverifikasi 38
Data kepercayaan dikirim ke Akses Terverifikasi dari penyedia kepercayaan 40
Konteks default untuk data kepercayaan Akses Terverifikasi 40
AWS IAM Identity Center konteks untuk data kepercayaan Akses Terverifikasi 42
Konteks penyedia kepercayaan pihak ketiga untuk data kepercayaan Akses Terverifikasi 44
Ekstensi browser 44
Jamf 45
CrowdStrike 47
JumpCloud 49
Klaim pengguna lewat 51
JWTuntuk klaim OIDC pengguna 51
JWTuntuk klaim pengguna Pusat IAM Identitas 52
Kunci publik 53
Mengambil dan mendekode JWT 53
Kebijakan Akses Terverifikasi 55
Struktur pernyataan kebijakan Akses Terverifikasi 55

Evaluasi kebijakan Akses Terverifikasi 57
Operator bawaan untuk kebijakan Akses Terverifikasi 57
Komentar kebijakan Akses Terverifikasi 60
Logika kebijakan Akses Terverifikasi hubung singkat 60
Kebijakan contoh Akses Terverifikasi 61
Asisten kebijakan 63
Langkah 1: Tentukan sumber daya Anda 64
Langkah 2: Uji dan edit kebijakan 64
Langkah 3: Tinjau dan terapkan perubahan 65
Keamanan 66
Perlindungan data 66
Enkripsi bergerak 68
Privasi lalu lintas antar jaringan 68
Enkripsi data saat istirahat 68
Pengelolaan identitas dan akses 83
Audiens 84
Mengautentikasi dengan identitas 84
Mengelola akses menggunakan kebijakan 88
Bagaimana Akses Terverifikasi bekerja dengan IAM 91
Contoh kebijakan berbasis identitas 97
Pemecahan Masalah 101
Gunakan peran tertaut layanan 103
AWS kebijakan terkelola 105
Validasi kepatuhan 107
Ketangguhan 108
Beberapa subnet untuk ketersediaan tinggi 108
Pemantauan 110
Log Akses Terverifikasi 110
Versi logging 111
Izin pencatatan 111
Mengaktifkan atau menonaktifkan log 112
Mengaktifkan atau menonaktifkan konteks kepercayaan 114
OCSFversi 0.1 contoh log 115
OCSFversi 1.0.0-rc.2 contoh log 127
CloudTrail log 132
Acara manajemen 134

Contoh acara 134
Kuota 136
Riwayat dokumen 138
..... cxxxix

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 a VPN untuk mengakses aplikasi Anda. Ini membantu mengurangi jumlah kasus dukungan yang timbul dari masalah VPN terkait.
- 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 VPC konsol 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 khususAPIs. AWS SDKsMengurus banyak detail koneksi, seperti menghitung tanda tangan, dan menangani percobaan ulang permintaan dan kesalahan. Untuk informasi lebih lanjut, lihat AWS SDKs.
- Kueri API — Menyediakan API tindakan tingkat rendah yang Anda panggil menggunakan HTTPS permintaan. 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 EC2APIReferensi Amazon.

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:

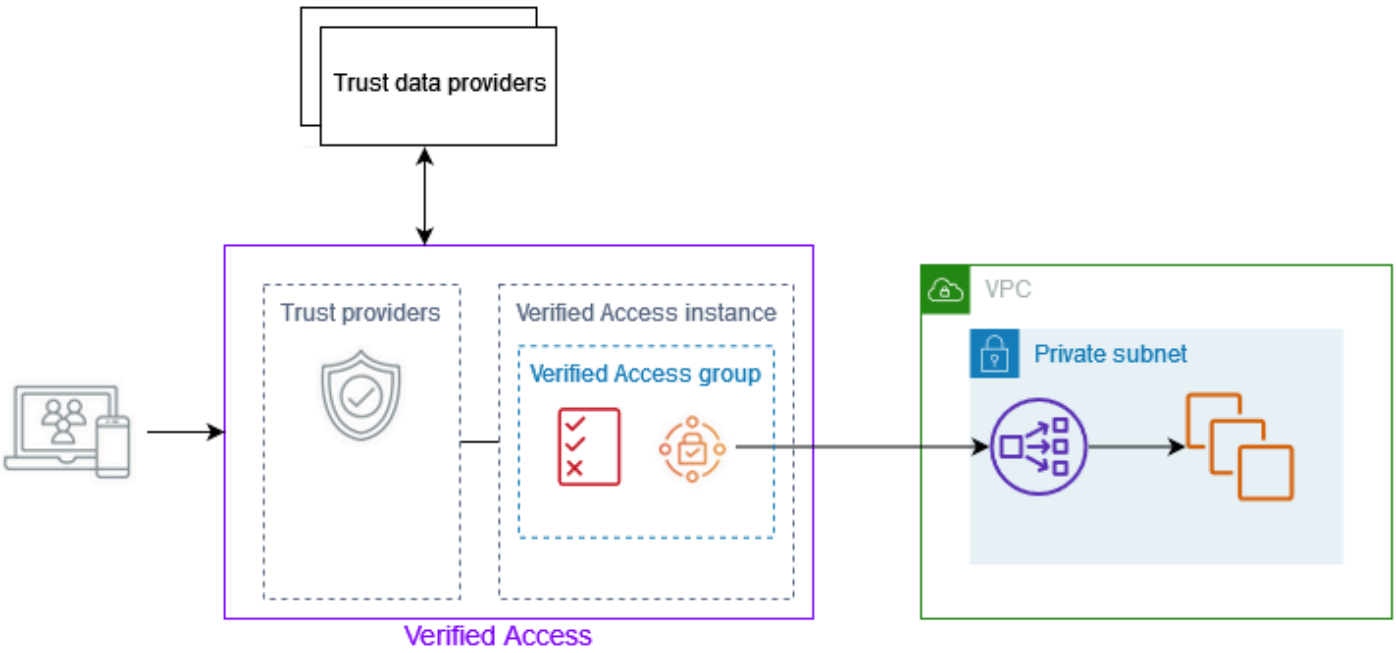
- Data kepercayaan yang dikirim oleh penyedia kepercayaan pilihan Anda (dari AWS atau pihak ketiga).
- Kebijakan akses yang Anda buat di Akses Terverifikasi.

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 gambaran 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. Anda dapat membuat titik akhir penyeimbang beban atau titik akhir antarmuka jaringan.
- 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 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 akan dapat mengakses aplikasi itu melalui internet, tanpa menggunakanVPN.

> **Note**
>
> Tutorial ini tidak menunjukkan integrasi dengan penyedia kepercayaan berbasis perangkat Anda. Sebaliknya, kami hanya bekerja dengan penyedia kepercayaan berbasis identitas.

Tugas

- Prasyarat tutorial Akses Terverifikasi
- Langkah 1: Buat instance Akses Terverifikasi
- Langkah 2: Konfigurasikan penyedia kepercayaan Akses Terverifikasi
- Langkah 3: Lampirkan penyedia kepercayaan Anda ke instans Akses Terverifikasi
- Langkah 4: Buat grup Akses Terverifikasi
- Langkah 5: Bagikan grup Akses Terverifikasi Anda melalui AWS Resource Access Manager
- Langkah 6: Tambahkan aplikasi Anda dengan membuat titik akhir Akses Terverifikasi
- Langkah 7: Konfigurasikan DNS pengaturan untuk titik akhir Akses Terverifikasi
- Langkah 8: Uji konektivitas ke aplikasi yang Anda tambahkan ke Akses Terverifikasi
- Langkah 9: Konfigurasikan kebijakan akses tingkat grup Akses Terverifikasi
- Langkah 10: Uji ulang konektivitas ke aplikasi yang Anda tambahkan ke Akses Terverifikasi
- Bersihkan sumber daya Akses Terverifikasi yang Anda buat

Prasyarat tutorial Akses Terverifikasi

Berikut ini adalah prasyarat untuk menyelesaikan tutorial ini:

- Ketersediaan dua Akun AWS. Satu akun meng-host aplikasi target Anda, dan sumber daya Akses Terverifikasi dibuat di akun lain.
- AWS IAM Identity Center diaktifkan di Wilayah AWS bahwa Anda bekerja di. Anda kemudian dapat menggunakan Pusat IAM Identitas sebagai penyedia kepercayaan dengan Akses Terverifikasi. Untuk informasi selengkapnya, lihat Mengaktifkan Pusat IAM Identitas di AWS IAM Identity Center Panduan Pengguna.
- Domain yang dihosting publik dan izin yang diperlukan untuk memperbarui DNS catatan untuk domain.
- Aplikasi yang berjalan di belakang penyeimbang beban internal di Akun AWS. Contoh nama domain aplikasi yang akan kita gunakan adalah`www.myapp.example.com`.
- TLSSertifikat yang ditandatangani sendiri atau publik. Gunakan RSA sertifikat dengan panjang kunci 1.024 atau 2.048.
- IAMKebijakan yang memiliki semua izin yang diperlukan untuk membuat Akses Terverifikasi AWS contoh dicatat di siniKebijakan untuk membuat instance Akses Terverifikasi.

Langkah 1: Buat instance Akses Terverifikasi

Gunakan prosedur berikut untuk membuat instance Akses Terverifikasi.

Untuk membuat instance Akses Terverifikasi

1. Buka VPC konsol Amazon di https://console.aws.amazon.com/vpc/.
2. Di panel VPC navigasi Amazon, pilih instans Akses Terverifikasi, lalu Buat instance Akses Terverifikasi.
3. (Opsional) Untuk Nama dan Deskripsi, masukkan nama dan deskripsi untuk instance Akses Terverifikasi.
4. Untuk penyedia Trust, pertahankan opsi default.
5. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
6. Pilih Buat instance Akses Terverifikasi.

Langkah 2: Konfigurasikan penyedia kepercayaan Akses Terverifikasi

Anda dapat mengatur AWS IAM Identity Center sebagai penyedia kepercayaan Anda.

Untuk membuat penyedia kepercayaan Pusat IAM Identitas

1. Di panel VPC navigasi Amazon, pilih penyedia kepercayaan Akses Terverifikasi, lalu Buat penyedia kepercayaan Akses Terverifikasi.
2. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk penyedia kepercayaan Akses Terverifikasi.
3. Masukkan pengenal kustom untuk digunakan nanti saat bekerja dengan aturan kebijakan untuk nama referensi Kebijakan. Misalnya, Anda bisa masuk**`idc`**.
4. Di bawah Jenis penyedia Trust, pilih Penyedia kepercayaan pengguna.
5. Di bawah Jenis penyedia kepercayaan pengguna, pilih Pusat IAM Identitas.
6. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
7. Pilih Buat penyedia kepercayaan Akses Terverifikasi.

Langkah 3: Lampirkan penyedia kepercayaan Anda ke instans Akses Terverifikasi

Setelah mengonfigurasi penyedia kepercayaan, Anda dapat melampirkannya ke instance Akses Terverifikasi yang Anda buat sebelumnya. Gunakan prosedur berikut untuk melampirkan penyedia kepercayaan ke instans Akses Terverifikasi Anda.

Untuk melampirkan penyedia kepercayaan ke instans Anda

1. Di panel VPC navigasi Amazon, pilih instans Akses Terverifikasi.
2. Pilih instans Anda.
3. Pilih Tindakan, Lampirkan penyedia kepercayaan Akses Terverifikasi.
4. Untuk penyedia kepercayaan Akses Terverifikasi, pilih penyedia kepercayaan Anda.
5. Pilih Lampirkan penyedia kepercayaan Akses Terverifikasi.

Langkah 4: Buat grup Akses Terverifikasi

Pada langkah ini, Anda membuat grup yang akan Anda gunakan sebagai titik akhir di Langkah 5.

Untuk membuat grup Akses Terverifikasi

1. Di panel VPC navigasi Amazon, pilih grup Akses Terverifikasi, lalu Buat grup Akses Terverifikasi.
2. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk grup.
3. Untuk instans Akses Terverifikasi, pilih instans Akses Terverifikasi Anda.
4. Untuk definisi Kebijakan, kosongkan ini. Anda akan membuat kebijakan nanti dalam tutorial ini.
5. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
6. Pilih Buat grup Akses Terverifikasi.

Langkah 5: Bagikan grup Akses Terverifikasi Anda melalui AWS Resource Access Manager

Pada langkah ini, Anda membagikan grup yang baru saja Anda buat dengan Akun AWS di mana aplikasi target Anda berjalan. Untuk membagikan grup Akses Terverifikasi, Anda harus menambahkannya ke pembagian sumber daya. Jika Anda tidak memiliki pembagian sumber daya, Anda harus membuatnya terlebih dahulu.

Jika Anda adalah bagian dari sebuah organisasi di AWS Organizations, dan berbagi dalam organisasi Anda diaktifkan, konsumen di organisasi Anda secara otomatis diberikan akses ke grup Akses Terverifikasi bersama. Jika tidak, konsumen menerima undangan untuk bergabung dengan pembagian sumber daya dan diberikan akses ke grup Akses Terverifikasi bersama setelah menerima undangan.

Ikuti langkah-langkah di Buat berbagi sumber daya di AWS RAM Panduan Pengguna. Untuk Pilih jenis sumber daya, pilih grup Akses Terverifikasi, lalu pilih kotak centang untuk grup Akses Terverifikasi Anda.

Untuk informasi selengkapnya, lihat Memulai di AWS RAM Panduan Pengguna.

Langkah 6: Tambahkan aplikasi Anda dengan membuat 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.

Untuk membuat titik akhir Akses Terverifikasi

1. Di panel VPC navigasi Amazon, pilih titik akhir Akses Terverifikasi, lalu Buat titik akhir Akses Terverifikasi.
2. (Opsional) Untuk tag Nama dan Deskripsi, masukkan nama dan deskripsi untuk titik akhir.
3. Untuk grup Akses Terverifikasi, pilih grup Akses Terverifikasi Anda.
4. Untuk detail Aplikasi, lakukan hal berikut:
    a. Untuk domain Aplikasi, masukkan DNS nama untuk aplikasi Anda.
    b. Di bawah Sertifikat domain ARN, pilih Nama Sumber Daya Amazon (ARN) dari TLS sertifikat publik Anda.
5. Untuk detail Endpoint, lakukan hal berikut:
    a. Untuk jenis Lampiran, pilih VPC.
    b. Untuk grup Keamanan, pilih grup keamanan untuk dikaitkan dengan titik akhir.
    c. Untuk awalan domain Endpoint, masukkan pengenal kustom. Ini akan ditambahkan ke DNS nama yang dihasilkan Akses Terverifikasi. Untuk contoh ini, kita bisa menggunakan**`my-ava-app`**.
    d. Untuk tipe Endpoint, pilih Load balancer.
    e. Untuk Protokol, pilih HTTPSatau HTTP. Ini tergantung pada konfigurasi penyeimbang beban Anda.
    f. Untuk Port, masukkan nomor port. Ini tergantung pada konfigurasi penyeimbang beban Anda.
    g. Untuk Load balancer ARN, pilih load balancer Anda.
    h. Untuk Subnet, pilih subnet yang terkait dengan penyeimbang beban Anda.
6. Untuk definisi Kebijakan, jangan masukkan kebijakan saat ini. Kami akan membahas ini nanti di tutorial.

7. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
8. Pilih Buat titik akhir Akses Terverifikasi.

Langkah 7: Konfigurasikan DNS pengaturan 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 DNS pemetaan, buat Canonical Name Record (CNAME) dengan penyedia Anda. DNS Setelah Anda membuat CNAME catatan, semua permintaan dari pengguna ke aplikasi Anda akan dikirim ke Akses Terverifikasi.

Untuk mendapatkan nama domain dari endpoint Anda

1. Di panel VPC navigasi Amazon, pilih titik akhir Akses Terverifikasi.
2. Pilih titik akhir yang Anda buat sebelumnya.
3. Pilih tab Detail untuk titik akhir.
4. Di bawah domain Endpoint, salin domain endpoint.

Untuk tutorial ini, nama domain endpoint akan menjadi`my-ava-app.edge-1a2b3c4d5e6f7g.vai-1a2b3c4d5e6f7g.prod.verified-access.us-west-2.amazonaws.com`.

Buat CNAME catatan dengan DNS penyedia Anda:

Nama catatan	Tipe	Nilai
www.myapp.example.com	CNAME	my-ava-app.edge-1a 2b3c4d5e6f7g.vai-1 a2b3c4d5e6f7g.prod. verified-access.us-west-2.amazonaws. com

Langkah 8: Uji konektivitas ke aplikasi yang Anda tambahkan ke Akses Terverifikasi

Anda sekarang dapat menguji konektivitas ke aplikasi Anda. Masukkan nama domain aplikasi Anda ke browser web Anda. Perilaku default kebijakan Akses Terverifikasi adalah menolak semua permintaan. Karena kami belum menerapkan kebijakan yang memungkinkan siapa pun mengakses, semua permintaan harus ditolak.

Langkah 9: Konfigurasikan 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 IAM Identitas. Untuk informasi tentang membuat kebijakan, lihatKebijakan Akses Terverifikasi.

Untuk mengubah grup Akses Terverifikasi

1. Di panel VPC navigasi Amazon, pilih grup Akses Terverifikasi.
2. Pilih grup Anda.
3. Pilih Tindakan, Ubah kebijakan grup Akses Terverifikasi.
4. Masukkan kebijakan.
5. Pilih Ubah kebijakan grup Akses Terverifikasi.

Langkah 10: Uji ulang konektivitas ke aplikasi yang Anda tambahkan ke Akses Terverifikasi

Sekarang setelah kebijakan grup Anda diberlakukan, Anda dapat mengakses aplikasi Anda. Masukkan nama domain aplikasi Anda ke browser web Anda. Permintaan harus diizinkan dan Anda harus diarahkan ke aplikasi.

Bersihkan sumber daya Akses Terverifikasi yang Anda buat

Setelah Anda selesai menguji, ikuti langkah di bawah ini untuk menghapus sumber daya yang dibuat.

Untuk menghapus sumber daya Akses Terverifikasi yang dibuat dengan tutorial ini

1. Di panel VPC navigasi Amazon, pilih titik akhir Akses Terverifikasi. Pilih titik akhir yang ingin Anda hapus. Pilih Tindakan, Hapus titik akhir Akses Terverifikasi.
2. Di panel navigasi, pilih grup Akses Terverifikasi. Pilih grup yang ingin Anda hapus. Pilih Tindakan, Hapus grup Akses Terverifikasi. Catatan - Anda mungkin perlu menunggu beberapa menit hingga proses penghapusan titik akhir selesai.
3. Di panel VPC navigasi Amazon, pilih instans Akses Terverifikasi. Pilih contoh yang Anda buat untuk tutorial ini. Pilih Tindakan, Lepaskan penyedia kepercayaan Akses Terverifikasi. Pilih penyedia kepercayaan dari daftar drop-down, pilih Lepaskan penyedia kepercayaan Akses Terverifikasi.
4. Di panel VPC navigasi Amazon, pilih penyedia kepercayaan Akses Terverifikasi. Pilih penyedia kepercayaan yang Anda buat untuk tutorial ini. Pilih Tindakan, Hapus penyedia kepercayaan Akses Terverifikasi.
5. Di panel VPC navigasi Amazon, pilih instans Akses Terverifikasi. Pilih contoh yang Anda buat untuk tutorial ini. Pilih Tindakan, Hapus instans Akses Terverifikasi.

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.

Topik

- Membuat dan mengelola instance Akses Terverifikasi
- Menghapus instans Akses Terverifikasi
- Integrasikan Akses Terverifikasi dengan AWS WAF
- FIPSkepatuhan 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.

Topik

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

Buat instance Akses Terverifikasi

Gunakan prosedur berikut untuk membuat instance Akses Terverifikasi.

Untuk membuat instance Akses Terverifikasi

1. Buka VPC konsol 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. (Opsional) Pilih aktifkan untuk Standar Proses Informasi Federal (FIPS) jika Anda memerlukan Akses Terverifikasi agar FIPS sesuai.
5. (Opsional) Untuk penyedia Trust, pilih penyedia kepercayaan untuk dilampirkan ke instance Akses Terverifikasi.
6. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
7. Pilih Buat instance Akses Terverifikasi.

Lampirkan penyedia kepercayaan ke instans Akses Terverifikasi

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

Untuk melampirkan penyedia kepercayaan ke instans Akses Terverifikasi

1. Buka VPC konsol 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.

Lepaskan penyedia kepercayaan dari instans Akses Terverifikasi

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

Untuk melepaskan penyedia kepercayaan dari instans Akses Terverifikasi

1. Buka VPC konsol 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.

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 instans Akses Terverifikasi

1. Buka VPC konsol 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.

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 (S) yang diteruskan ke sumber daya aplikasi web Anda yang dilindungi. Untuk informasi selengkapnya AWS WAF, lihat AWS WAFdi Panduan AWS WAF Pengembang.

Anda dapat mengintegrasikan AWS WAF dengan Akses Terverifikasi dengan mengaitkan daftar kontrol akses AWS WAF web (ACL) dengan instance Akses Terverifikasi. Web ACL adalah AWS WAF sumber daya yang memberi Anda kontrol halus atas semua permintaan web HTTP (S) yang ditanggapi oleh sumber daya Anda yang dilindungi. 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 ke`active`. Anda dapat melihat status di AWS Management Console atau dengan menjelaskan titik akhir dengan. AWS CLI

> **Note**
>
> Anda juga dapat menggunakan AWS WAF konsol atau API untuk mencapai integrasi ini. Anda akan memerlukan Amazon Resource Name (ARN) dari instans Akses

Terverifikasi Anda. Anda dapat membuat ini ARN menggunakan format berikut:`arn:${Partition}:ec2:${Region}:${Account}:verified-access-instance/${VerifiedAccessInstanceId}`.

Topik

- IAMizin yang diperlukan untuk mengintegrasikan Akses Terverifikasi dengan AWS WAF
- Kaitkan AWS WAF web ACL
- Periksa status AWS WAF integrasi
- Putuskan hubungan web AWS WAF ACL

IAMizin yang diperlukan untuk mengintegrasikan Akses Terverifikasi dengan AWS WAF

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 webACL, AWS Identity and Access Management kepala sekolah Anda harus memiliki izin berikut.

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

Kaitkan AWS WAF web ACL

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

> **Tip**
>
> Anda harus memiliki AWS WAF web yang ada ACL untuk menyelesaikan prosedur di bawah ini. Untuk informasi selengkapnya tentang web, ACLs lihat daftar kontrol akses Web di Panduan AWS WAF Pengembang.

Untuk mengaitkan AWS WAF web ACL ke instans Akses Terverifikasi

1. Buka VPC konsol 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 web yang adaACL, lalu pilih Associate Web ACL.

Anda juga dapat menggunakan AWS Management Console for AWS WAF untuk menyelesaikan tugas ini. Untuk informasi selengkapnya, lihat Mengaitkan atau memisahkan web ACL dengan AWS sumber daya di Panduan Pengembang.AWS WAF

Periksa status AWS WAF integrasi

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

Untuk melihat status AWS WAF integrasi dengan instans Akses Terverifikasi

1. Buka VPC konsol 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 WAF integrasi. Status akan ditampilkan sebagai Terkait atau Tidak terkait, bersama dengan ACL pengenal web, jika dalam status Terkait.

Putuskan hubungan web AWS WAF ACL

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

Untuk memisahkan AWS WAF web ACL dari instance Akses Terverifikasi

1. Buka VPC konsol 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.

Anda juga dapat menggunakan AWS Management Console for AWS WAF untuk menyelesaikan tugas ini. Untuk informasi selengkapnya, lihat Mengaitkan atau memisahkan web ACL dengan AWS sumber daya di Panduan Pengembang.AWS WAF

FIPSkepatuhan 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 Publikasi 140-2. FIPSkepatuhan 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 kepada Anda cara mengonfigurasi lingkungan Akses Terverifikasi baru atau yang sudah ada, agar FIPS sesuai.

Topik

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

Konfigurasikan lingkungan Akses Terverifikasi yang ada untuk FIPS kepatuhan

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

Untuk mengonfigurasi ulang Akses Terverifikasi AWS lingkungan yang ada agar FIPS sesuai, 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 Federal Information Process Standards (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 FIPS kepatuhan

Untuk mengonfigurasi Akses Terverifikasi AWS lingkungan baru yang FIPS sesuai, 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 Anda dengan Akses AWS Terverifikasi.

Prasyarat dan pertimbangan

- Instance Pusat IAM Identitas Anda harus berupa sebuah AWS Organizations instance. Instance Pusat IAM Identitas AWS akun mandiri tidak akan berfungsi.
- Instance Pusat IAM Identitas Anda harus diaktifkan di AWS Wilayah yang sama tempat Anda ingin membuat penyedia kepercayaan Akses Terverifikasi.

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

Buat penyedia kepercayaan Pusat IAM Identitas

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

Untuk membuat penyedia kepercayaan Pusat IAM Identitas (AWS konsol)

1. Buka VPC konsol 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 IAM Identitas.
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 IAM Identitas (AWS CLI)

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

Hapus penyedia kepercayaan Pusat IAM Identitas

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 IAM Identitas (AWS konsol)

1. Buka VPC konsol 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 IAM Identitas (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 OIDC kompatibel sebagai penyedia kepercayaan identitas pengguna dengan Akses Terverifikasi. Namun, karena beragam OIDC penyedia potensial, AWS tidak dapat menguji setiap OIDC integrasi dengan Akses Terverifikasi.

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

> **Note**
>
> Akses Terverifikasi tidak menggunakan data kepercayaan dari yang `ID token` dikirim oleh OIDC penyedia, saat mengevaluasi kebijakan Akses Terverifikasi. Hanya data kepercayaan dari yang `UserInfo Endpoint` dievaluasi terhadap kebijakan.

Daftar Isi

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

Prasyarat untuk menciptakan 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 OIDC kepercayaan

Gunakan prosedur berikut untuk membuat OIDC sebagai penyedia kepercayaan Anda.

Untuk membuat penyedia OIDC kepercayaan (AWS konsol)

1. Buka VPC konsol 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 Penerbit, masukkan pengidentifikasi penerbit. OIDC
8. Untuk titik akhir Otorisasi, masukkan titik akhir URL otorisasi penuh.
9. Untuk titik akhir Token, masukkan titik URL akhir token penuh.
10. Untuk titik akhir Pengguna, masukkan titik URL akhir pengguna penuh.
11. Masukkan pengenal klien OAuth 2.0 untuk ID Klien.
12. Masukkan rahasia klien OAuth 2.0 untuk rahasia Klien.
13. Masukkan daftar cakupan yang dibatasi spasi yang ditentukan dengan penyedia identitas Anda. Minimal, lingkup “openid” diperlukan untuk Lingkup.
14. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
15. Pilih Buat penyedia kepercayaan Akses Terverifikasi.

> **Note**
>
> Anda perlu menambahkan pengalihan URI ke daftar izin OIDC penyedia Anda. Anda akan ingin menggunakan titik akhir Akses Terverifikasi untuk tujuan ini. `ApplicationDomain` 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 OIDC penyedia Anda: https://`ApplicationDomain`/oauth2/idpresponse

Untuk membuat penyedia OIDC kepercayaan (AWS CLI)

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

Memodifikasi penyedia OIDC kepercayaan

Setelah Anda membuat penyedia kepercayaan, Anda dapat memperbarui konfigurasinya.

Untuk memodifikasi penyedia OIDC kepercayaan (AWS konsol)

1. Buka VPC konsol 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 OIDC kepercayaan (AWS CLI)

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

Hapus penyedia OIDC kepercayaan

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 OIDC kepercayaan (AWS konsol)

1. Buka VPC konsol 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 OIDC kepercayaan (AWS CLI)

- 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 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 VPC konsol 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 kunci penandatanganan Publik URL, masukkan kunci unik yang URL 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 pengalihan URI ke daftar izin OIDC penyedia 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 OIDC penyedia Anda: https://`DeviceValidationDomain`/oauth2/idpresponse

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 VPC konsol 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 kunci penandatanganan Publik URL, ubah kunci unik yang URL 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 VPC konsol 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

Sesi Akses Terverifikasi AWS grup adalah kumpulan titik akhir Akses Terverifikasi dan kebijakan Akses Terverifikasi tingkat grup. Setiap titik akhir dalam grup membagikan kebijakan Akses Terverifikasi. Anda dapat menggunakan grup untuk mengumpulkan titik akhir yang memiliki persyaratan keamanan umum. Ini dapat membantu menyederhanakan administrasi kebijakan dengan menggunakan satu kebijakan untuk kebutuhan keamanan beberapa aplikasi.

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.

Tugas

- Membuat grup Akses Terverifikasi
- Ubah kebijakan grup Akses Terverifikasi
- Menghapus grup Akses Terverifikasi

Membuat grup Akses Terverifikasi

Gunakan prosedur berikut untuk membuat grup Akses Terverifikasi.

Untuk membuat grup Akses Terverifikasi

1. Buka VPC konsol 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.

Ubah kebijakan grup Akses Terverifikasi

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

Untuk mengubah kebijakan grup Akses Terverifikasi

1. Buka VPC konsol 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.

Menghapus grup Akses Terverifikasi

Setelah selesai dengan grup Akses Terverifikasi, Anda dapat menghapusnya.

Untuk menghapus grup Akses Terverifikasi

1. Buka VPC konsol 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.

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
- 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.
- Antarmuka jaringan — Permintaan aplikasi dikirim ke antarmuka jaringan menggunakan protokol dan port yang ditentukan.

Cara kerja Akses Terverifikasi dengan berbagi VPCs dan subnet

Berikut ini adalah perilaku terkait VPC subnet bersama:

- Titik akhir Akses Terverifikasi didukung oleh berbagi VPC subnet. 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. VPCPemilik tidak akan diizinkan untuk memodifikasi titik akhir.

- Titik akhir Akses Terverifikasi tidak dapat dibuat di AWS Zona Lokal dan karenanya berbagi melalui Local Zones tidak dimungkinkan.

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

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.
- Hanya HTTPS protokol HTTP dan yang didukung. HTTPSKoneksi berumur panjang, seperti WebSocket koneksi, tidak didukung.
- Load balancer harus berupa Application Load Balancer atau Network Load Balancer, dan harus merupakan penyeimbang beban internal.
- Load balancer dan subnet harus dimiliki oleh virtual private cloud () VPC yang sama.
- HTTPSpenyeimbang beban dapat menggunakan sertifikat yang ditandatangani sendiri atau sertifikat publik. TLS Gunakan RSA sertifikat dengan panjang kunci 1.024 atau 2.048.
- Anda harus memberikan nama domain untuk aplikasi Anda. Ini adalah DNS nama publik yang akan digunakan pengguna Anda untuk mengakses aplikasi Anda. Anda juga perlu memberikan SSL sertifikat 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

1. Buka VPC konsol 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 Aplikasi, lakukan hal berikut:

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

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

7. Untuk detail Endpoint, lakukan hal berikut:

a. Untuk jenis Lampiran, pilih VPC.

b. Untuk grup Keamanan, pilih grup keamanan untuk titik akhir. Lalu lintas dari titik akhir Akses Terverifikasi yang memasuki penyeimbang beban Anda akan dikaitkan dengan grup keamanan ini.

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

d. Untuk tipe Endpoint, pilih Load balancer.

e. Untuk Protokol, pilih HTTPS atau HTTP.

f. Di bawah Port, masukkan nomor port.

g. Untuk Load balancer ARN, pilih load balancer.

h. Untuk Subnet, pilih subnet untuk penyeimbang beban Anda.

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.

Membuat titik akhir antarmuka jaringan untuk Akses Terverifikasi

Gunakan prosedur berikut untuk membuat titik akhir antarmuka jaringan.

Persyaratan

- Hanya IPv4 lalu lintas yang didukung.
- Hanya HTTPS protokol HTTP dan 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.
- Anda harus memberikan nama domain untuk aplikasi Anda. Ini adalah DNS nama publik yang akan digunakan pengguna Anda untuk mengakses aplikasi Anda. Anda juga perlu memberikan

SSL sertifikat publik dengan CN yang cocok dengan nama domain ini. Anda dapat membuat atau mengimpor sertifikat menggunakan AWS Certificate Manager.

Untuk membuat endpoint antarmuka jaringan

1. Buka VPC konsol 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 Aplikasi, lakukan hal berikut:
   a. Untuk domain Aplikasi, masukkan DNS nama untuk aplikasi Anda.
   b. Di bawah Sertifikat domain ARN, pilih TLS sertifikat publik.
7. Untuk detail Endpoint, lakukan hal berikut:
   a. Untuk jenis Lampiran, pilih VPC.
   b. Untuk grup Keamanan, pilih grup keamanan untuk titik akhir. Lalu lintas dari titik akhir Akses Terverifikasi yang memasuki antarmuka jaringan Anda akan dikaitkan dengan grup keamanan ini.
   c. Untuk awalan domain Endpoint, masukkan pengenal kustom untuk menambahkan DNS nama yang dihasilkan Akses Terverifikasi untuk titik akhir.
   d. Untuk tipe Endpoint, pilih Network interface.
   e. Untuk Protokol, pilih HTTPS atau HTTP.
   f. Di bawah Port, masukkan nomor port.
   g. Untuk antarmuka Jaringan, pilih antarmuka jaringan.
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.

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 sumber. 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

1. Buka VPC konsol 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. Di 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.

Ubah titik akhir Akses Terverifikasi

Gunakan prosedur berikut untuk mengubah titik akhir Akses Terverifikasi.

Untuk mengubah titik akhir Akses Terverifikasi

1. Buka VPC konsol 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.

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

1. Buka VPC konsol 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.

Menghapus titik akhir Akses Terverifikasi

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

Untuk menghapus titik akhir Akses Terverifikasi

1. Buka VPC konsol 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.

Data kepercayaan dikirim ke Akses Terverifikasi dari penyedia kepercayaan

Data kepercayaan adalah data yang dikirim ke 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 HTTP permintaan saat ini secara default di semua evaluasi Cedar terlepas dari penyedia kepercayaan Anda yang dikonfigurasi. Ketika kebijakan dievaluasi, Akses Terverifikasi menyertakan data tentang HTTP permintaan saat ini dalam konteks Cedar di bawah. `context.http_request key` Anda dapat menulis kebijakan yang mengevaluasi terhadap data jika Anda memilih. JSONSkema berikut menunjukkan data mana yang termasuk dalam evaluasi.

```
{
     "title": "HTTP Request data included by Verified Access",
     "type": "object",
     "properties": {
         "user_agent": {
             "type": "string",
             "description": "The value of the User-Agent request header"
         },
         "x_forwarded_for": {
             "type": "string",
             "description": "The value of the X-Forwarded-For request header"
         },
         "http_method": {
             "type": "string",
             "description": "The HTTP Method provided (e.g. GET or POST)"
         },
         "hostname": {
             "type": "string",
             "description": "The value of the Host request header"
         },
         "port": {
            "type": "integer",
            "description": "The value of the verified access endpoint port"
         },
         "client_ip": {
             "type": "string",
             "description": "User ip connecting to the verified access endpoint"
         }
     }
 }
```

Berikut ini adalah contoh kebijakan yang mengevaluasi terhadap data HTTP permintaan.

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

AWS IAM Identity Center konteks untuk data kepercayaan Akses Terverifikasi

Ketika kebijakan dievaluasi, jika Anda mendefinisikan AWS IAM Identity Center Sebagai penyedia kepercayaan, Akses Terverifikasi AWS 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.

> **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.

JSONSkema 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, Pusat IAM Identitas 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 kepada 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 salah satu AWS Ekstensi browser Akses 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 Akses Terverifikasi AWS ekstensi 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 Akses Terverifikasi AWS Native Messaging Host (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 Akses Terverifikasi AWS ekstensi 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. JSONSkema berikut menunjukkan data mana yang termasuk dalam evaluasi.

Untuk informasi selengkapnya tentang penggunaan Jamf dengan Akses Terverifikasi, lihat Mengintegrasikan Akses AWS Terverifikasi dengan Identitas Perangkat Jamf 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. JSONSkema 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 AWSdi situs GitHub 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 environemnt"
    },
    "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. JSONSkema berikut menunjukkan data mana yang termasuk dalam evaluasi.

Untuk informasi lebih lanjut tentang menggunakan JumpCloud dengan AWS Akses Terverifikasi, lihat Mengintegrasikan dan JumpCloud AWS Akses 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_orgnaization_identifier'
};
```

Klaim pengguna lulus dan verifikasi tanda tangan di Akses Terverifikasi

Setelah sebuah Akses Terverifikasi AWS instance mengautentikasi pengguna berhasil, 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, HTTP header berikut ditambahkan:

`x-amzn-ava-user-context`

Header ini berisi klaim pengguna dalam format token JSON web (JWT). JWTFormatnya mencakup header, payload, dan tanda tangan yang dikodekan base64URL. Verified Access menggunakan ES384 (algoritma ECDSA tanda tangan menggunakan SHA -384 algoritma hash) 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: Ditandatangani JWT untuk klaim OIDC pengguna
- Contoh: Ditandatangani JWT untuk klaim pengguna Pusat IAM Identitas
- Kunci publik
- Contoh: Mengambil dan mendekode JWT

Contoh: Ditandatangani JWT untuk klaim OIDC pengguna

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

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 payload:

```
{
   "sub": "xyzsubject",
   "email": "xxx@amazon.com",
   "email_verified": true,
   "groups": [
      "Engineering",
      "finance"
   ]
}
```

Contoh: Ditandatangani JWT untuk klaim pengguna Pusat IAM Identitas

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

> **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 payload:

```
{
```

```
    "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 digunakan. HTTPS Jika Anda mengonfigurasi titik akhir Akses Terverifikasi untuk digunakanHTTP, 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 JWT header berisi instance Akses Terverifikasi yang diharapkan. ARN

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

Endpoint 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 Anda yang dihosting AWS. Mereka ditulis dalam Cedar, AWS bahasa 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.

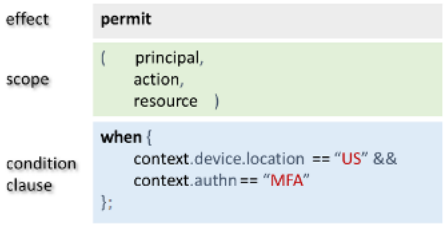
Bagian ini menjelaskan bagaimana kebijakan Akses Terverifikasi terstruktur, isinya, cara mendefinisikannya, dan memberikan beberapa contoh.

Daftar Isi

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

Struktur pernyataan kebijakan Akses Terverifikasi

Bagian ini menjelaskan Akses Terverifikasi AWS pernyataan kebijakan dan bagaimana hal itu dievaluasi. Anda dapat memiliki beberapa pernyataan dalam satu kebijakan Akses Terverifikasi. Diagram berikut menunjukkan struktur kebijakan Akses Terverifikasi.



Kebijakan ini berisi bagian-bagian berikut:

- Efek - Menentukan apakah pernyataan kebijakan adalah `permit` (`Allow`) atau `forbid` (`Deny`).
- Lingkup - Menentukan 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 (seperti yang ditunjukkan pada contoh sebelumnya). Dalam hal ini, kebijakan berlaku untuk semua prinsip, tindakan, dan sumber daya yang mungkin.
- Kondisi klausa - Menentukan 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.

Contoh kebijakan sederhana

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

Pada contoh sebelumnya, perhatikan bahwa Anda dapat menggunakan lebih dari satu klausa kondisi dalam pernyataan kebijakan menggunakan operator. && Bahasa kebijakan Cedar memberi Anda kekuatan ekspresif untuk membuat pernyataan kebijakan yang bersifat adat, berbutir halus, dan ekstensif. Untuk contoh tambahan, lihat Kebijakan contoh Akses Terverifikasi.

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.

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

Operator bawaan untuk kebijakan Akses Terverifikasi

Saat membuat konteks sebuah Akses Terverifikasi AWS kebijakan menggunakan berbagai kondisi, seperti 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.
==	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.

Operator	Jenis dan kelebihan beban	Deskripsi
!=	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.

Komentar kebijakan Akses Terverifikasi

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

Contoh berikut menunjukkan pernyataan komentar dalam kebijakan.

```
// this policy grants access to users in a given domain with 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)
};
```

Logika kebijakan Akses Terverifikasi hubung singkat

Anda mungkin ingin menulis Akses Terverifikasi AWS kebijakan yang mengevaluasi data yang mungkin atau mungkin tidak hadir 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

Contoh 1: Membuat kebijakan untuk Pusat IAM Identitas

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

Contoh kebijakan berikut memungkinkan akses hanya ketika pengguna termasuk dalam `finance` grup (yang memiliki ID grup`c242c5b0-6081-1845-6fa8-6e0d9513c107`) dan memiliki alamat email terverifikasi.

```
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 1b: Menambahkan lebih banyak kondisi ke pernyataan kebijakan untuk IAM Identity Center

Contoh kebijakan berikut mengizinkan akses hanya ketika pengguna termasuk dalam `finance` grup (yang memiliki ID grup`c242c5b0-6081-1845-6fa8-6e0d9513c107`), memiliki alamat email terverifikasi, dan skor risiko perangkat Jamf adalah`L0W`.

```
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"
};
```

Contoh 2: Kebijakan yang sama untuk OIDC penyedia pihak ketiga

Contoh kebijakan berikut memungkinkan akses hanya ketika pengguna berasal dari grup “keuangan”, mereka memiliki alamat email terverifikasi, dan skor risiko perangkat Jamf adalahLOW.

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

Contoh 3: 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: Bekerja dengan karakter khusus

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")
};
```

Contoh 5: Izinkan alamat IP tertentu

Contoh berikut menunjukkan kebijakan yang hanya mengizinkan alamat IP tertentu.

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

Contoh 5a: Blokir alamat IP tertentu

Contoh berikut menunjukkan kebijakan yang akan memblokir alamat IP tertentu.

```
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 di satu layar, tempat 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 VPC konsol Amazon 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.

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 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.

Konten

- 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

Bagian AWS model tanggung jawab bersama model berlaku untuk perlindungan data di AWS Akses Terverifikasi. Seperti yang dijelaskan dalam model ini, AWS bertanggung jawab untuk

melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk menjaga kontrol atas konten Anda yang di-host di infrastruktur ini. Anda juga bertanggung jawab atas konfigurasi keamanan dan tugas manajemen untuk Layanan AWS yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, lihat Privasi Data FAQ. Untuk informasi tentang perlindungan data di Eropa, lihat AWS Model Tanggung Jawab Bersama dan posting GDPR blog di AWS Blog Keamanan.

Untuk tujuan perlindungan data, kami menyarankan Anda untuk melindungi Akun AWS kredensi 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 otentikasi multi-faktor (MFA) dengan setiap akun.
- GunakanSSL/TLSuntuk berkomunikasi dengan AWS sumber daya. Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang menggunakan CloudTrail jalur untuk menangkap AWS kegiatan, lihat Bekerja dengan CloudTrail jalan setapak di AWS CloudTrail Panduan Pengguna.
- Gunakan AWS solusi enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan FIPS 140-3 modul kriptografi yang divalidasi saat mengakses AWS melalui antarmuka baris perintah atauAPI, gunakan FIPS titik akhir. Untuk informasi selengkapnya tentang FIPS titik akhir yang tersedia, lihat Federal Information Processing Standard (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 ketika 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. Jika Anda memberikan URL ke server eksternal, kami sangat menyarankan agar Anda tidak menyertakan informasi kredensil dalam URL untuk memvalidasi permintaan Anda ke server tersebut.

Enkripsi bergerak

Akses Terverifikasi 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 AndaVPC. 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 AWS Akses Terverifikasi

AWS Akses Terverifikasi mengenkripsi data saat istirahat secara default, menggunakan AWS KMSkunci yang 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 KMS kunci untuk enkripsi data saat istirahat.

Daftar Isi

- Akses dan KMS kunci terverifikasi
- Informasi pengenal pribadi
- Bagaimana AWS Akses 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 AWS Akses Terverifikasi

Akses dan KMS kunci terverifikasi

AWS kunci yang dimiliki

Akses Terverifikasi menggunakan KMS kunci untuk secara otomatis mengenkripsi informasi yang dapat diidentifikasi secara pribadi (). PII Ini terjadi secara default, dan Anda sendiri tidak dapat

melihat, mengelola, menggunakan, atau mengaudit penggunaan kunci yang AWS dimiliki. Namun, Anda tidak perlu mengambil tindakan apa pun atau mengubah program apa pun untuk melindungi kunci yang mengenkripsi data Anda. Untuk informasi selengkapnya, silakan lihat AWS kunci yang dimiliki di AWS Key Management Service Panduan Pengembang.

Meskipun Anda tidak dapat menonaktifkan lapisan enkripsi ini atau memilih jenis enkripsi alternatif, Anda dapat menambahkan lapisan enkripsi kedua di atas yang ada AWS memiliki kunci enkripsi dengan memilih kunci terkelola 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 IAM kebijakan dan hibah
- 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 AWS Key Management Service Panduan Pengembang.

> **Note**
>
> Akses Terverifikasi secara otomatis mengaktifkan enkripsi saat istirahat menggunakan AWS kunci yang 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 selengkapnya 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 OIDC opsi seperti AuthorizationEndpoint,, UserInfoEndpoint, ClientId, ClientSecret, dan sebagainya, yang dipertimbangkan. PII	Diaktifkan	Diaktifkan
`Trust provider (device-type)` Penyedia kepercayaan tipe perangkat berisi a TenantId, yang dipertimbangkan. PII	Diaktifkan	Diaktifkan
`Group policy` Disediakan selama pembuatan atau modifikas i grup Akses Terverifikasi. Berisi aturan untuk mengotori sasi permintaan akses. Mungkin berisi PII seperti nama pengguna dan alamat email, dan sebagainya.	Diaktifkan	Diaktifkan
`Endpoint policy`	Diaktifkan	Diaktifkan

Tipe data	AWS enkripsi kunci yang dimiliki	Enkripsi kunci yang dikelola pelanggan (Opsional)
Disediakan selama pembuatan atau modifikasi titik akhir Akses Terverifikasi. Berisi aturan untuk mengotorisasi permintaan akses. Mungkin berisi PII seperti nama pengguna dan alamat email, dan sebagainya.		

Bagaimana AWS Akses 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 CreateGrantmengirimkan permintaan ke AWS KMS. Hibah di 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 ke 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 terkelola pelanggan simetris dengan menggunakan AWS Management Console, atau AWS KMS APIs. Ikuti langkah-langkah untuk Membuat kunci terkelola pelanggan simetris di AWS Key Management Service Panduan 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 Mengelola akses ke kunci terkelola pelanggan di AWS Key Management Service Panduan Pengembang.

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

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

Untuk informasi selengkapnya tentang pemecahan masalah akses kunci, lihat AWS Key Management Service Panduan 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

Saat 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 AWS KMS ID kunci yang ingin Anda gunakan. Hal ini juga dapat dilakukan ketika memodifikasi sumber daya yang ada, atau dengan menggunakan 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 atau AWS CLI, untuk menerapkan kunci terkelola 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 AWS KMS operasi 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"
}
```

Untuk informasi selengkapnya tentang penggunaan konteks enkripsi untuk hibah atau kebijakan, lihat konteks enkripsi di AWS Key Management Service Panduan Pengembang.

Memantau kunci enkripsi Anda untuk AWS Akses Terverifikasi

Ketika Anda menggunakan KMS kunci yang dikelola pelanggan dengan Anda AWS Sumber daya Akses Terverifikasi, Anda dapat menggunakan AWS CloudTrailuntuk melacak permintaan yang dikirim oleh Akses Terverifikasi AWS KMS.

Contoh berikut adalah AWS CloudTrail peristiwa untuk`CreateGrant`,,`RetireGrant`,`Decrypt`, dan `DescribeKeyGenerateDataKey`, yang memantau KMS operasi yang dipanggil oleh Akses Terverifikasi untuk mengakses data yang dienkripsi oleh kunci terkelola KMS 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. 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. IAMadministrator mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya Akses Terverifikasi. IAMadalah sebuah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- Audiens
- Mengautentikasi dengan identitas
- Mengelola akses menggunakan kebijakan
- Bagaimana Akses Terverifikasi bekerja 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

Bagaimana 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. Anda kemudian harus mengirimkan permintaan ke IAM administrator Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasarIAM. Untuk mempelajari lebih lanjut tentang cara perusahaan Anda dapat menggunakan IAM Akses Terverifikasi, lihatBagaimana Akses Terverifikasi bekerja dengan IAM.

IAMadministrator - Jika Anda seorang IAM administrator, 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, lihat. IAM 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 IAM pengguna, atau dengan mengambil IAM peran.

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

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

Jika Anda mengakses AWS secara terprogram, AWS menyediakan 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. Untuk informasi selengkapnya tentang menggunakan metode yang disarankan untuk menandatangani permintaan sendiri, lihat Menandatangani AWS APIpermintaan di Panduan IAM Pengguna.

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 di AWS IAM Identity Center Panduan Pengguna dan Menggunakan otentikasi multi-faktor () MFA di AWS di 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 Akun AWS pengguna 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 kredensi pengguna root di IAMPanduan Pengguna.

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, AWS Directory Service, direktori Pusat Identitas, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensi yang disediakan melalui sumber identitas. Ketika akses identitas federasi 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 IAM Identitas, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua Akun AWS dan aplikasi. Untuk informasi tentang Pusat IAM Identitas, lihat Apa itu Pusat IAM Identitas? di AWS IAM Identity Center Panduan Pengguna.

Pengguna dan grup IAM

IAMPengguna adalah identitas di dalam Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya mengandalkan kredensi sementara daripada membuat IAM pengguna yang memiliki kredensi jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensil jangka panjang dengan IAM pengguna, kami sarankan Anda memutar kunci akses. Untuk informasi selengkapnya, lihat Memutar kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensi jangka panjang di IAMPanduan Pengguna.

IAMGrup adalah identitas yang menentukan kumpulan IAM pengguna. 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 memiliki grup bernama IAMAdminsdan memberikan izin grup tersebut untuk mengelola sumber dayaIAM.

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 lebih lanjut, lihat Kapan membuat IAM pengguna (bukan peran) di Panduan IAM Pengguna.

IAMperan

IAMPeran adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Ini mirip dengan IAM pengguna, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil IAM peran sementara dalam AWS Management Console dengan beralih peran. Anda dapat mengambil peran dengan memanggil AWS CLI atau AWS APIoperasi atau dengan menggunakan kustomURL. Untuk informasi selengkapnya tentang metode penggunaan peran, lihat Menggunakan IAM peran di Panduan IAM Pengguna.

IAMperan dengan kredensi 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 Membuat peran untuk Penyedia Identitas pihak ketiga di Panduan IAM Pengguna. Jika Anda menggunakan Pusat IAM Identitas, Anda mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah diautentikasi, Pusat IAM Identitas mengkorelasikan izin yang disetel ke peran. IAM Untuk informasi tentang set izin, lihat Set izin di AWS IAM Identity Center Panduan Pengguna.
- Izin IAM pengguna sementara — IAM Pengguna atau peran dapat mengambil IAM peran untuk sementara mengambil izin yang berbeda untuk tugas tertentu.
- Akses lintas akun — Anda dapat menggunakan IAM peran untuk memungkinkan seseorang (prinsipal tepercaya) di akun lain mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan 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 Panduan Pengguna. IAM
- Akses lintas layanan - Beberapa Layanan AWS menggunakan fitur di 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.
  - Teruskan sesi akses (FAS) — Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan di AWS Anda dianggap sebagai kepala sekolah. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FASmenggunakan izin dari prinsipal yang memanggil Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. FASPermintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat Meneruskan sesi akses.
  - Peran layanan — Peran layanan adalah IAMperan yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAMAdministrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalamIAM. Untuk informasi selengkapnya, lihat Membuat peran untuk mendelegasikan izin ke Layanan AWS di Panduan Pengguna IAM.

  - Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke Layanan AWS. Layanan dapat mengambil peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Akun AWS dan dimiliki oleh layanan. IAMAdministrator dapat melihat, tetapi tidak mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan IAM peran untuk mengelola kredensional sementara untuk aplikasi yang berjalan pada EC2 instance dan membuat AWS CLI atau AWS APIpermintaan. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance. Untuk menetapkan AWS peran ke sebuah 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 IAM peran untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon di IAMPanduan Pengguna.

Untuk mempelajari apakah akan menggunakan IAM peran atau IAM pengguna, lihat Kapan membuat IAM peran (bukan pengguna) di Panduan IAM Pengguna.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses di AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek di AWS bahwa, ketika dikaitkan dengan identitas atau sumber daya, mendefinisikan izin mereka. 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 di AWS sebagai JSON dokumen. Untuk informasi selengkapnya tentang struktur dan isi dokumen JSON kebijakan, lihat Ringkasan JSON kebijakan di Panduan IAM Pengguna.

Administrator dapat menggunakan AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

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

IAMkebijakan menentukan izin untuk tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasi. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan

`iam:GetRole`. Pengguna dengan kebijakan tersebut dapat memperoleh informasi peran dari AWS Management Console, AWS CLI, atau AWS API.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat Anda lampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. 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 Membuat IAM kebijakan di 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 di Akun AWS. Kebijakan terkelola meliputi AWS kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan sebaris, lihat Memilih antara kebijakan terkelola dan kebijakan sebaris di IAMPanduan Pengguna.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran 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 AWS kebijakan 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. ACLsmirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan. JSON

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukungACLs. Untuk mempelajari selengkapnyaACLs, lihat Ikhtisar 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.

- Batas izin — Batas izin adalah fitur lanjutan tempat Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas (pengguna atau peran). IAM 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 batas izin, lihat Batas izin untuk IAM entitas di IAMPanduan Pengguna.
- Kebijakan kontrol layanan (SCPs) — SCPs adalah JSON kebijakan yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan mengelola 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. SCPMembatasi izin untuk entitas di akun anggota, termasuk masing-masing Pengguna root akun AWS. Untuk informasi selengkapnya tentang Organizations danSCPs, lihat Kebijakan kontrol layanan di AWS Organizations Panduan 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 secara tegas dalam salah satu kebijakan ini membatalkan izin. Untuk informasi selengkapnya, lihat Kebijakan sesi di Panduan IAM Pengguna.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari caranya AWS menentukan apakah akan mengizinkan

permintaan saat beberapa jenis kebijakan terlibat, lihat Logika evaluasi kebijakan di Panduan IAM Pengguna.

Bagaimana Akses Terverifikasi bekerja dengan IAM

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

IAMfitur	Dukungan Akses Terverifikasi
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Tidak
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
Kunci kondisi kebijakan	Ya
ACLs	Tidak
ABAC(tag dalam kebijakan)	Parsial
Kredensial sementara	Ya
Izin prinsipal	Ya
Peran layanan	Tidak
Peran terkait layanan	Ya

Untuk mendapatkan tampilan tingkat tinggi tentang bagaimana Akses Terverifikasi dan lainnya AWS layanan bekerja dengan sebagian besar IAM fitur, lihat AWS layanan yang bekerja dengan IAM dalam Panduan IAM Pengguna.

Kebijakan berbasis identitas untuk Akses Terverifikasi

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat Anda lampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. 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 Membuat IAM kebijakan di Panduan Pengguna. IAM

Dengan kebijakan IAM berbasis identitas, Anda dapat menentukan tindakan dan sumber daya yang diizinkan atau ditolak serta kondisi di mana tindakan diizinkan atau ditolak. 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 JSON kebijakan, lihat referensi elemen IAM JSON kebijakan di Panduan IAM Pengguna.

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 JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran 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 seluruh akun atau IAM entitas 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 kepala sekolah dan sumber daya berbeda Akun AWS, IAM administrator 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 prinsipal dalam akun yang sama, tidak

diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, lihat Akses sumber daya lintas akun IAM di Panduan IAM Pengguna.

Tindakan kebijakan untuk Akses Terverifikasi

Mendukung tindakan kebijakan: Ya

Administrator dapat menggunakan AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

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

Menyertakan 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 AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen `Resource` JSON kebijakan menentukan objek atau objek yang tindakan tersebut berlaku. Pernyataan harus menyertakan elemen `Resource` atau `NotResource`. Sebagai praktik terbaik, 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 jenisnyaARNs, lihat Sumber Daya yang Ditentukan oleh Amazon EC2 di Referensi Otorisasi Layanan. Untuk mempelajari tindakan 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: Ya

Administrator dapat menggunakan AWS JSONkebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam 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 `Condition` elemen dalam pernyataan, atau beberapa kunci dalam satu `Condition` elemen, AWS mengevaluasi mereka menggunakan `AND` operasi 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. Misalnya, Anda dapat memberikan izin IAM pengguna untuk mengakses sumber daya hanya jika ditandai dengan nama IAM pengguna mereka. Untuk informasi selengkapnya, lihat elemen IAM kebijakan: variabel dan tag di Panduan IAM Pengguna.

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

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

ACLsdi Akses Terverifikasi

MendukungACLs: Tidak

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLsmirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan. JSON

ABACdengan Akses Terverifikasi

Mendukung ABAC (tag dalam kebijakan): Sebagian

Attribute-based access control (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan atribut. Masuk AWS, atribut ini disebut tag. Anda dapat melampirkan tag ke IAM entitas (pengguna atau peran) dan ke banyak AWS sumber daya. Menandai entitas dan sumber daya adalah langkah pertama dari. ABAC Kemudian Anda merancang ABAC kebijakan untuk mengizinkan operasi ketika tag prinsipal cocok dengan tag pada sumber daya yang mereka coba akses.

ABACmembantu dalam lingkungan yang berkembang pesat dan membantu dengan situasi di mana manajemen kebijakan menjadi rumit.

Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tag 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 lebih lanjut tentangABAC, lihat Apa ituABAC? dalam IAMUser Guide. Untuk melihat tutorial dengan langkah-langkah penyiapanABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) di IAMPanduan Pengguna.

Menggunakan kredensi sementara dengan Akses Terverifikasi

Mendukung kredensi 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 dalam Panduan IAM Pengguna.

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

Anda dapat secara manual membuat kredensi sementara menggunakan AWS CLI atau AWS API. 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 Kredensi keamanan sementara di. IAM

Izin utama lintas layanan untuk Akses Terverifikasi

Mendukung sesi akses maju (FAS): Ya

Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan AWS Anda dianggap sebagai kepala sekolah. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FASmenggunakan izin dari prinsipal yang memanggil Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. FASPermintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat Meneruskan sesi akses.

Peran layanan untuk Akses Terverifikasi

Mendukung peran layanan: Tidak

Peran layanan adalah IAMperan yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAMAdministrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalamIAM. Untuk informasi selengkapnya, lihat Membuat peran untuk mendelegasikan izin ke Layanan AWS di 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 dapat mengambil peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Akun AWS dan dimiliki oleh layanan. IAMAdministrator dapat melihat, tetapi tidak 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 pada sumber daya yang mereka butuhkan, IAM administrator dapat membuat IAM kebijakan. Administrator kemudian dapat menambahkan IAM kebijakan ke peran, dan pengguna dapat mengambil peran.

Untuk mempelajari cara membuat kebijakan IAM berbasis identitas menggunakan contoh dokumen kebijakan ini, lihat Membuat JSON IAM kebijakan 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 dapat menimbulkan biaya untuk Anda Akun AWS. Saat Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi berikut:

- Memulai dengan AWS kebijakan terkelola dan beralih ke izin hak istimewa terkecil — Untuk memulai pemberian izin kepada pengguna dan beban kerja Anda, gunakan AWS kebijakan terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan mendefinisikan AWS kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, silakan lihat AWS kebijakan terkelola atau AWS kebijakan terkelola untuk fungsi pekerjaan di Panduan IAM Pengguna.
- Menerapkan izin hak istimewa paling sedikit — Saat Anda menetapkan izin dengan IAM kebijakan, berikan hanya 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 penggunaan IAM untuk menerapkan izin, lihat Kebijakan dan izin IAM di IAM Panduan Pengguna.
- Gunakan ketentuan dalam IAM kebijakan untuk membatasi akses lebih lanjut — Anda dapat menambahkan kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Misalnya, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakanSSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui tindakan tertentu Layanan AWS, seperti AWS

CloudFormation. Untuk informasi selengkapnya, lihat elemen IAM JSON kebijakan: Kondisi dalam Panduan IAM Pengguna.

- Gunakan IAM Access Analyzer untuk memvalidasi IAM kebijakan Anda guna memastikan izin yang aman dan fungsional — IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan mematuhi bahasa IAM kebijakan () JSON dan praktik terbaik. IAM IAMAccess 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 IAM Access Analyzer di IAMPanduan Pengguna.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan IAM pengguna atau pengguna root di Akun AWS, nyalakan MFA untuk keamanan tambahan. Untuk meminta MFA kapan API operasi dipanggil, tambahkan MFA kondisi ke kebijakan Anda. Untuk informasi selengkapnya, lihat Mengonfigurasi API akses MFA yang dilindungi di IAMPanduan Pengguna.

Untuk informasi selengkapnya tentang praktik terbaik diIAM, lihat Praktik terbaik keamanan IAM di Panduan IAM Pengguna.

Kebijakan untuk membuat instance Akses Terverifikasi

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

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

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

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

```
{
      "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 Anda membuat kebijakan yang memungkinkan IAM pengguna melihat kebijakan sebaris dan terkelola yang dilampirkan pada identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau secara terprogram menggunakan AWS CLI atau AWS API.

```
{
    "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 danIAM.

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 `mateojackson` IAM pengguna mencoba menggunakan konsol untuk melihat detail tentang *`my-example-widget`* sumber daya fiksi tetapi tidak memiliki izin `ec2:`*`GetWidget`* fiksi.

```
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 membutuhkan bantuan, hubungi AWS administrator. 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 IAM pengguna 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 membutuhkan bantuan, hubungi AWS administrator. 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, lihatBagaimana Akses Terverifikasi bekerja dengan IAM.

- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh Akun AWS yang Anda miliki, lihat Menyediakan akses ke IAM pengguna di pengguna lain Akun AWS yang Anda miliki di Panduan IAM Pengguna.
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda ke pihak ketiga Akun AWS, lihat Menyediakan akses ke Akun AWS dimiliki oleh pihak ketiga dalam Panduan IAM Pengguna.
- Untuk mempelajari cara menyediakan akses melalui federasi identitas, lihat Menyediakan akses ke pengguna yang diautentikasi secara eksternal (federasi identitas) di 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 Panduan Pengguna. IAM

Menggunakan peran terkait layanan untuk Akses Terverifikasi

Akses Terverifikasi AWS menggunakan AWS Identity and Access Management (IAM) peran terkait layanan. Peran terkait layanan adalah jenis peran unik yang ditautkan langsung ke Akses Terverifikasi. IAM Peran terkait layanan telah ditentukan sebelumnya 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 lain. IAM

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, lihat AWS layanan yang bekerja dengan IAM dan cari layanan yang memiliki Ya di kolom Peran terkait layanan. Pilih Ya bersama tautan untuk melihat dokumentasi peran tertaut layanan untuk layanan tersebut.

Izin peran terkait layanan untuk Akses Terverifikasi

Akses Terverifikasi menggunakan peran terkait layanan yang diberi nama AWSServiceRoleForVPCVerifiedAccessuntuk menyediakan sumber daya di akun Anda yang diperlukan untuk menggunakan layanan.

Peran AWSServiceRoleForVPCVerifiedAccessterkait layanan 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 AWS Management Console AWSVPCVerifiedAccessServiceRolePolicy, atau Anda dapat melihat AWSVPCVerifiedAccessServiceRolePolicykebijakan di Panduan Referensi Kebijakan AWS Terkelola.

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

Membuat peran terkait layanan untuk Akses Terverifikasi

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

Jika Anda menghapus peran tertaut layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda 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 mengedit peran AWSServiceRoleForVPCVerifiedAccessterkait layanan. Setelah Anda membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengedit deskripsi peran menggunakanIAM. Untuk informasi selengkapnya, lihat Mengedit peran terkait layanan di IAMPanduan Pengguna.

Menghapus peran terkait layanan untuk Akses Terverifikasi

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

Untuk menghapus peran terkait layanan secara manual menggunakan IAM

Gunakan IAM konsol, AWS CLI, atau AWS API untuk menghapus peran AWSServiceRoleForVPCVerifiedAccessterkait layanan. 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 API operasi baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat kebijakan AWS terkelola di Panduan IAM Pengguna.

AWS kebijakan terkelola: AWSVPCVerifiedAccessServiceRolePolicy

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 RSS feed 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 Standar Pemrosesan Informasi Federal (FIPS). Untuk info dan detail selengkapnya tentang pengaturan FIPS kepatuhan untuk Akses Terverifikasi, bukaFIPSkepatuhan 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:

- Panduan Memulai Cepat Keamanan dan Kepatuhan — Panduan penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar AWS yang berfokus pada keamanan dan kepatuhan.
- Arsitektur untuk HIPAA Keamanan dan Kepatuhan di Amazon Web Services — Whitepaper ini menjelaskan bagaimana perusahaan dapat menggunakan AWS untuk membuat HIPAA aplikasi yang memenuhi syarat.

  > **Note**
  >
  > Tidak semua Layanan AWS HIPAA memenuhi syarat. Untuk informasi selengkapnya, lihat Referensi Layanan yang HIPAA Memenuhi Syarat.

- 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 PCIDSS, 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 rinci tentang permintaan untuk mengakses aplikasi. Untuk informasi selengkapnya, lihat the section called “Log Akses Terverifikasi”.
- AWS CloudTrail— Menangkap API panggilan 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. Verified Access mendukung format logging Open Cybersecurity Schema Framework (OCSF).

Ketika Anda mengaktifkan logging, Anda perlu mengkonfigurasi tujuan untuk log yang akan dikirim. IAMPrinsipal yang digunakan untuk mengonfigurasi tujuan logging harus memiliki izin tertentu agar logging berfungsi dengan baik. IAMIzin yang diperlukan untuk setiap tujuan pencatatan dapat dilihat di Izin pencatatan Akses Terverifikasi bagian. 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
- OCSFversi 0.1 contoh log untuk Akses Terverifikasi
- OCSFversi 1.0.0-rc.2 contoh log untuk Akses Terverifikasi

Versi logging Akses Terverifikasi

Secara default, sistem logging Akses Terverifikasi menggunakan Open Cybersecurity Schema Framework (OCSF) versi 0.1. Contoh log menggunakan versi 0.1 dapat dilihat di OCSFversi 0.1 contoh log untuk Akses Terverifikasi bagian.

Versi logging terbaru kompatibel dengan OCSF versi 1.0.0-rc.2. Rincian spesifik tentang skema dapat ditemukan di sini OCSFSkema. Contoh log menggunakan versi 1.0.0-rc.2 dapat dilihat di bagian. OCSFversi 1.0.0-rc.2 contoh log untuk Akses Terverifikasi

Jika Anda ingin memutakhirkan versi logging yang digunakan, gunakan prosedur berikut.

Untuk memutakhirkan versi logging menggunakan konsol

1. Buka VPC konsol 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

IAMPrinsipal 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. IAMPrinsipal yang digunakan untuk mengonfigurasi tujuan logging harus memiliki izin tertentu agar logging berfungsi dengan baik. IAMIzin yang diperlukan untuk setiap tujuan pencatatan dapat dilihat di Izin pencatatan Akses Terverifikasi bagian.

Daftar Isi

- Aktifkan log akses
- Nonaktifkan log akses

Aktifkan log akses

Untuk mengaktifkan log Akses Terverifikasi

1. Buka VPC konsol 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 VPC konsol 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 VPC konsol 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 VPC konsol 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.

OCSFversi 0.1 contoh log untuk Akses Terverifikasi

Berikut ini adalah contoh log menggunakan OCSF versi logging default 0.1.

Contoh

- Akses yang diberikan dengan OIDC

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

Akses yang diberikan dengan OIDC

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


```
{
    "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 yang diberikan dengan OIDC dan JAMF

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

```
{
    "activity": "Access Granted",
    "activity_id": "1",
    "category_name": "Application Activity",
    "category_uid": "8",
    "class_name": "Access Logs",
    "class_uid": "208001",
    "device": {
        "ip": "10.2.7.68",
        "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 yang diberikan dengan OIDC dan CrowdStrike

Dalam entri log contoh ini, Akses Terverifikasi memungkinkan akses ke titik akhir dengan keduanya 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
}
```


OCSFversi 1.0.0-rc.2 contoh log untuk Akses Terverifikasi

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

Daftar Isi

- Akses yang diberikan dengan konteks kepercayaan disertakan
- Akses yang diberikan dengan konteks kepercayaan dihilangkan

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"
            },
            "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
}
```


Log API panggilan Akses Terverifikasi menggunakan AWS CloudTrail

AWS Verified Access terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau Layanan AWS di Akses Terverifikasi. CloudTrail menangkap API panggilan untuk Akses Terverifikasi sebagai acara. Panggilan yang diambil termasuk panggilan dari konsol Akses Terverifikasi dan panggilan kode ke API operasi 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 Pusat IAM Identitas.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna terfederasi.
- Apakah permintaan itu dibuat oleh orang lain Layanan AWS.

CloudTrail Aktif dalam 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 Wilayah AWS. Untuk informasi selengkapnya, lihat Bekerja dengan riwayat

CloudTrail Acara di AWS CloudTrail Panduan Pengguna. Tidak ada CloudTrail biaya untuk melihat riwayat Acara.

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

CloudTrail jalan setapak

Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Semua jalur dibuat menggunakan AWS Management Console adalah Multi-region. Anda dapat membuat jalur Single-region atau Multi-region dengan menggunakan AWS CLI. Membuat jejak Multi-wilayah disarankan karena Anda menangkap aktivitas di semua Wilayah AWS di akun Anda. Jika Anda membuat jejak wilayah Tunggal, Anda hanya dapat melihat peristiwa yang dicatat di jejak Wilayah AWS. Untuk informasi selengkapnya tentang jalur, lihat Membuat jejak untuk Anda Akun AWSdan Menciptakan jejak untuk organisasi di AWS CloudTrail Panduan 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 Menyimpan data acara danau

CloudTrail Lake memungkinkan Anda menjalankan kueri SQL berbasis pada acara Anda. CloudTrail Lake mengonversi peristiwa yang ada dalam JSON format berbasis baris ke format Apache. ORC ORCadalah 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 AWS CloudTrail Panduan 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 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 EC2APIReferensi Amazon.

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 AWS CloudTrail Panduan 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 Trust 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.

HTTPheader

Berikut ini adalah batas ukuran untuk HTTP header.

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

OIDCukuran klaim

Berikut ini adalah batas ukuran OIDC klaim.

Nama	Default	Dapat disesuaikan
OIDCukuran klaim	11 K	Tidak

Riwayat dokumen untuk Panduan Pengguna Akses Terverifikasi

Tabel berikut menjelaskan rilis dokumentasi untuk Akses Terverifikasi.

Perubahan	Deskripsi	Tanggal
AWS kebijakan terkelola diperbarui	Pembaruan dibuat untuk IAM kebijakan AWS terkelola untuk Akses Terverifikasi.	17 November 2023
Enkripsi data saat istirahat	AWS Akses Terverifikasi mengenkripsi data saat istirahat secara default, menggunakan kunci yang AWS dimilikiKMS.	28 September 2023
Support untuk FIPS kepatuhan	Konfigurasikan Akses Terverifikasi untuk FIPS kepatuhan.	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 IAM kebijakan 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.