aws

Panduan Pengguna

Kisi VPC Amazon

Kisi VPC Amazon: Panduan Pengguna

Table of Contents

Apa itu Amazon VPC Lattice?

Amazon VPC Lattice adalah layanan jaringan aplikasi terkelola penuh yang Anda gunakan untuk menghubungkan, mengamankan, dan memantau layanan dan sumber daya untuk aplikasi Anda. Anda dapat menggunakan VPC Lattice dengan satu virtual private cloud (VPC) atau beberapa VPCs dari satu akun atau lebih.

Aplikasi modern dapat terdiri dari beberapa komponen kecil dan modular yang sering disebut layanan mikro, seperti API HTTP, sumber daya seperti database, dan sumber daya khusus yang terdiri dari DNS dan titik akhir alamat IP. Meskipun modernisasi memiliki kelebihan, modernisasi juga dapat memperkenalkan kompleksitas dan tantangan jaringan ketika Anda menghubungkan layanan mikro dan sumber daya ini. Misalnya, jika pengembang tersebar di tim yang berbeda, mereka mungkin membangun dan menyebarkan layanan mikro dan sumber daya di beberapa akun atau. VPCs

Dalam VPC Lattice, kami merujuk ke layanan mikro sebagai layanan dan mewakili sumber daya hanya sebagai konfigurasi sumber daya. Ini adalah istilah yang Anda lihat dan akan digunakan dalam panduan pengguna VPC Lattice.

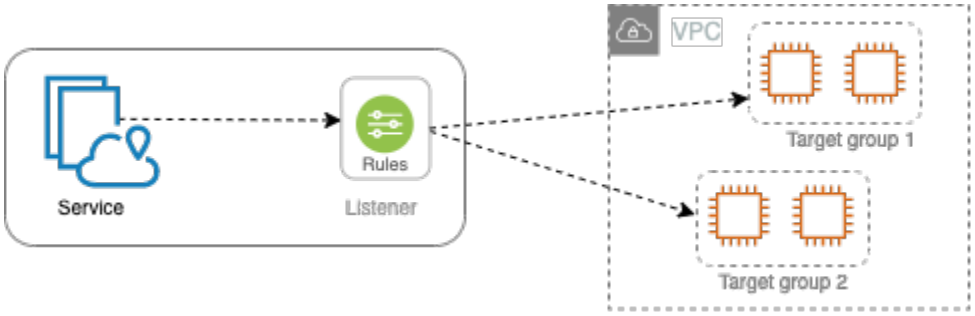
Daftar Isi

Komponen kunci

Untuk menggunakan Amazon VPC Lattice, Anda harus terbiasa dengan komponen utamanya.

Layanan

> Unit perangkat lunak yang dapat digunakan secara independen yang memberikan tugas atau fungsi tertentu. Layanan dapat berjalan pada instans atau ECS/EKS/Fargate kontainer EC2, atau sebagai fungsi Lambda, dalam akun atau virtual private cloud (VPC). Layanan VPC Lattice memiliki komponen berikut: grup target, pendengar, dan aturan.



Grup target

Kumpulan sumber daya, juga dikenal sebagai target, yang menjalankan aplikasi atau layanan Anda. Ini mirip dengan kelompok sasaran yang disediakan oleh Elastic Load Balancing, tetapi mereka tidak dapat dipertukarkan. Jenis target yang didukung meliputi instans EC2, alamat IP, fungsi Lambda, Application Load Balancer, tugas Amazon ECS, dan Kubernetes Pods.

Pendengar

Proses yang memeriksa permintaan koneksi, dan merutekkannya ke target dalam grup target. Anda mengonfigurasi pendengar dengan protokol dan nomor port.

Aturan

Komponen default dari listener yang meneruskan permintaan ke target dalam grup target VPC Lattice. Setiap aturan terdiri dari prioritas, satu atau beberapa tindakan, dan satu atau beberapa syarat. Aturan menentukan cara pendengar merutekan permintaan klien.

Sumber daya

Sumber daya adalah entitas seperti database Amazon Relational Database Service (Amazon RDS), instans Amazon EC2, titik akhir aplikasi, target nama domain, atau alamat IP. Anda dapat berbagi sumber daya di VPC Anda dengan membuat pembagian sumber daya di AWS Resource Access Manager (AWS RAM), membuat gateway sumber daya, dan menentukan konfigurasi sumber daya.

Gateway sumber daya

Gateway sumber daya adalah titik masuknya ke dalam VPC tempat sumber daya berada.

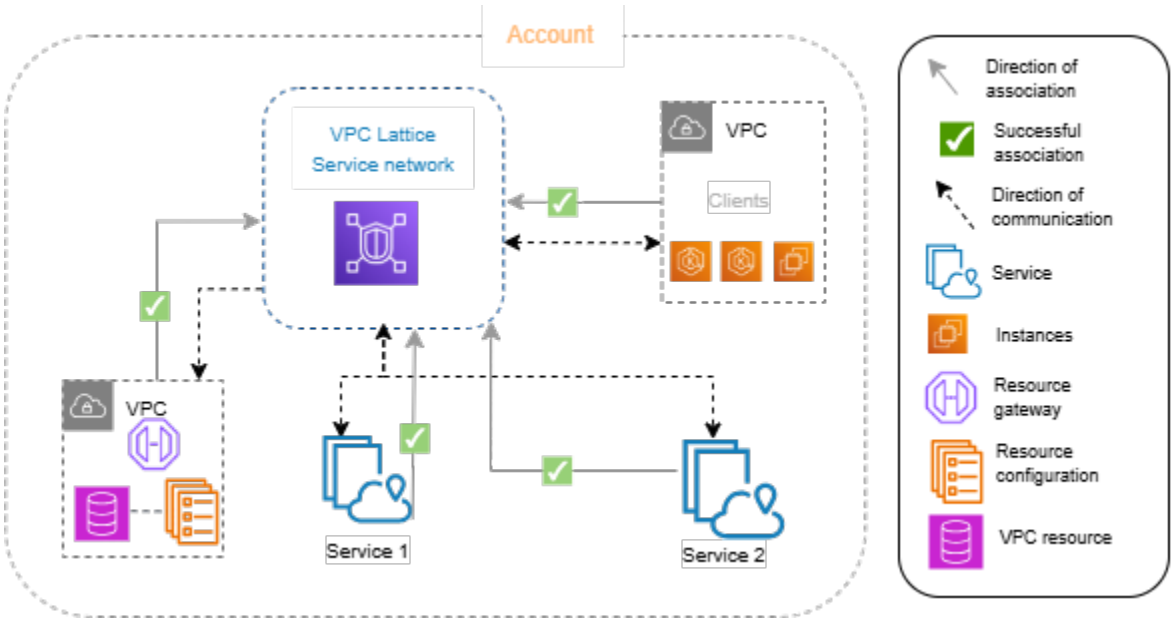
Konfigurasi sumber daya

Konfigurasi sumber daya adalah objek logis yang mewakili sumber daya tunggal atau sekelompok sumber daya. Sumber daya dapat berupa alamat IP, target nama domain, atau database Amazon RDS.

Jaringan layanan

Batas logis untuk kumpulan layanan dan konfigurasi sumber daya. Klien dapat berada di VPC yang terkait dengan jaringan layanan. Klien dan layanan yang terkait dengan jaringan layanan yang sama dapat berkomunikasi satu sama lain jika mereka berwenang untuk melakukannya.

Pada gambar berikut, klien dapat berkomunikasi dengan kedua layanan, karena VPC dan layanan dikaitkan dengan jaringan layanan yang sama.



Direktori layanan

Registri pusat dari semua layanan VPC Lattice yang Anda miliki atau bagikan dengan akun Anda. AWS RAM

Kebijakan autentikasi

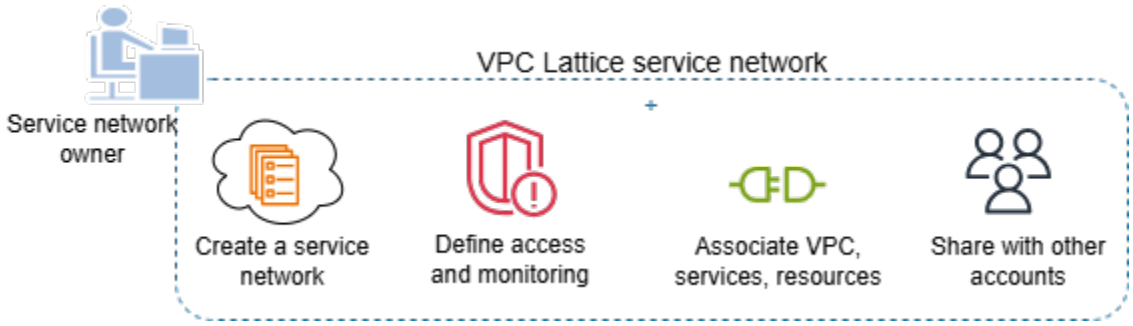
Kebijakan otorisasi berbutir halus yang dapat digunakan untuk menentukan akses ke layanan. Anda dapat melampirkan kebijakan autentikasi terpisah ke layanan individual atau ke jaringan layanan. Misalnya, Anda dapat membuat kebijakan tentang bagaimana layanan pembayaran yang berjalan pada grup penskalaan otomatis instans EC2 harus berinteraksi dengan layanan penagihan yang sedang berjalan. AWS Lambda

Kebijakan Auth-tidak didukung pada konfigurasi sumber daya. Kebijakan autentikasi jaringan layanan tidak berlaku untuk konfigurasi sumber daya di jaringan layanan.

Peran dan tanggung jawab

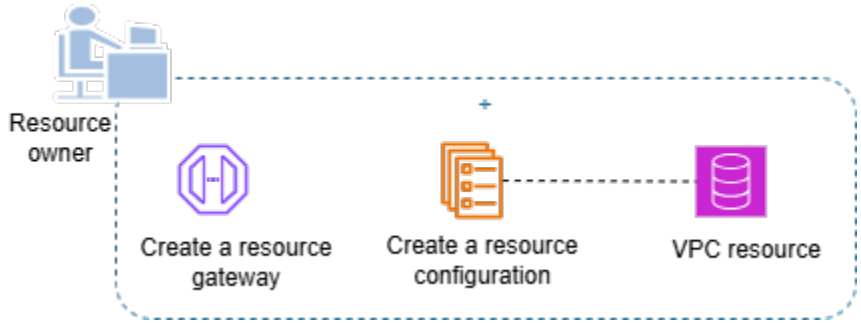
Peran menentukan siapa yang bertanggung jawab atas penyiapan dan aliran informasi dalam Amazon VPC Lattice. Biasanya ada dua peran, pemilik jaringan layanan dan pemilik layanan, dan tanggung jawab mereka dapat tumpang tindih.

Pemilik jaringan layanan — Pemilik jaringan layanan biasanya administrator jaringan atau administrator cloud dalam suatu organisasi. Pemilik jaringan layanan membuat, berbagi, dan menyediakan jaringan layanan. Mereka juga mengelola siapa yang dapat mengakses jaringan layanan atau layanan dalam VPC Lattice. Pemilik jaringan layanan dapat menentukan pengaturan akses berbutir kasar untuk layanan yang terkait dengan jaringan layanan. Kontrol ini digunakan untuk mengelola komunikasi antara klien dan layanan menggunakan kebijakan otentikasi dan otorisasi. Pemilik jaringan layanan juga dapat mengaitkan konfigurasi layanan atau sumber daya dengan satu atau beberapa jaringan layanan, jika konfigurasi layanan atau sumber daya dibagikan dengan akun pemilik jaringan layanan.



Pemilik layanan — Pemilik layanan biasanya adalah pengembang perangkat lunak dalam suatu organisasi. Pemilik layanan membuat layanan dalam VPC Lattice, menentukan aturan perutean, dan juga mengaitkan layanan dengan jaringan layanan. Mereka juga dapat menentukan pengaturan akses berbutir halus, yang dapat membatasi akses hanya ke layanan dan klien yang diautentikasi dan resmi.

Pemilik sumber daya — Pemilik sumber daya biasanya pengembang perangkat lunak dalam suatu organisasi dan berfungsi sebagai admin untuk sumber daya seperti database. Pemilik sumber daya membuat konfigurasi sumber daya untuk sumber daya, mendefinisikan pengaturan akses untuk konfigurasi sumber daya, dan mengaitkan konfigurasi sumber daya dengan jaringan layanan.



Fitur

Berikut ini adalah fitur inti yang disediakan VPC Lattice.

Penemuan Layanan

Semua klien dan layanan yang VPCs terkait dengan jaringan layanan dapat berkomunikasi dengan layanan lain dalam jaringan layanan yang sama. DNS mengarahkan client-to-service dan service-to-service lalu lintas melalui titik akhir VPC Lattice. Ketika klien ingin mengirim permintaan ke layanan, ia menggunakan nama DNS layanan. Resolver Route 53 mengirimkan lalu lintas ke VPC Lattice, yang kemudian mengidentifikasi layanan tujuan.

Konektivitas

Client-to-service dan client-to-resource konektivitas dibangun dalam infrastruktur AWS jaringan. Ketika Anda mengaitkan VPC dengan jaringan layanan, setiap klien dalam VPC dapat terhubung dengan layanan dan sumber daya (melalui konfigurasi sumber daya) di jaringan layanan, jika mereka memiliki akses yang diperlukan. VPC Lattice mendukung teknologi CIDR yang tumpang tindih.

Akses di premis

Anda dapat mengaktifkan konektivitas ke jaringan layanan dari VPC menggunakan titik akhir VPC (didukung oleh). AWS PrivateLink Titik akhir VPC dari jaringan layanan tipe memungkinkan Anda mengaktifkan akses ke layanan dan sumber daya di jaringan layanan dari jaringan lokal melalui Direct Connect dan VPN. Lalu lintas yang melintasi VPC peering atau juga AWS Transit Gateway dapat mengakses sumber daya dan layanan melalui titik akhir VPC.

Observabilitas

VPC Lattice menghasilkan metrik dan log untuk setiap permintaan dan respons yang melintasi jaringan layanan, untuk membantu Anda memantau dan memecahkan masalah aplikasi. Secara default, metrik dipublikasikan ke akun pemilik layanan. Pemilik layanan dan pemilik sumber daya memiliki opsi untuk mengaktifkan logging, dan menerima log untuk semua klien access/requests ke layanan dan sumber daya mereka. Pemilik jaringan layanan juga dapat mengaktifkan logging pada jaringan layanan, untuk mencatat semua access/requests ke layanan dan sumber daya dari klien VPCs yang terhubung ke jaringan layanan.

VPC Lattice bekerja dengan alat berikut untuk membantu Anda memantau dan memecahkan masalah layanan Anda: Amazon CloudWatch grup log, aliran pengiriman Firehose, dan bucket Amazon S3.

Keamanan

VPC Lattice menyediakan kerangka kerja yang dapat Anda gunakan untuk menerapkan strategi pertahanan di beberapa lapisan jaringan. Lapisan pertama adalah kombinasi layanan, konfigurasi sumber daya, asosiasi VPC, dan titik akhir VPC dari jaringan layanan tipe. Tanpa VPC dan asosiasi layanan atau titik akhir VPC dari jaringan layanan tipe, klien tidak dapat mengakses layanan. Demikian pula, tanpa VPC dan konfigurasi sumber daya dan asosiasi layanan atau titik akhir VPC dari jaringan layanan tipe, klien tidak dapat mengakses sumber daya.

Lapisan kedua memungkinkan pengguna untuk melampirkan grup keamanan ke asosiasi antara VPC dan jaringan layanan. Lapisan ketiga dan keempat adalah kebijakan autentikasi yang dapat diterapkan secara individual di tingkat jaringan layanan dan tingkat layanan.

Afinitas Zona Ketersediaan

VPC Lattice mendukung afinitas Availability Zone (AZ) untuk merutekan lalu lintas. Ketika klien mengirim permintaan ke VPC Lattice, VPC Lattice merespons dengan alamat IP untuk layanan atau sumber daya dari AZ yang sama dengan klien. Jika AZ itu tidak tersedia, VPC Lattice merespons dengan alamat IP dari yang lain. AZs Dari VPC Lattice ke target, routing adalah ke target, yang mungkin didistribusikan ke seluruh. AZs Selain itu, tidak ada biaya transfer data antar-AZ di VPC Lattice.

Mengakses Kisi VPC

Anda dapat membuat, mengakses, dan mengelola VPC Lattice menggunakan salah satu antarmuka berikut:

- Konsol Manajemen AWS— Menyediakan antarmuka web yang dapat Anda gunakan untuk mengakses VPC Lattice.
- AWS Command Line Interface (AWS CLI) — Menyediakan perintah untuk serangkaian AWS layanan yang luas, termasuk VPC Lattice. AWS CLI Ini didukung di Windows, macOS, dan Linux. Untuk informasi lebih lanjut tentang CLI, lihat. AWS Command Line Interface Untuk informasi selengkapnya tentang APIs, lihat Referensi API Kisi VPC Amazon.
- VPC Lattice Controller for Kubernetes — Mengelola resource VPC Lattice untuk klaster Kubernetes. Untuk informasi selengkapnya tentang penggunaan VPC Lattice dengan Kubernetes, lihat Panduan Pengguna Gateway API Controller.AWS
- CloudFormation— Membantu Anda memodelkan dan mengatur AWS sumber daya Anda. Untuk informasi selengkapnya, lihat referensi jenis sumber daya Amazon VPC Lattice.

Titik akhir layanan VPC Lattice

Endpoint adalah URL yang berfungsi sebagai titik masuk untuk layanan AWS web. VPC Lattice mendukung jenis endpoint berikut:

- the section called “IPv4 titik akhir”
- Titik akhir dualstack (mendukung keduanya dan) IPv4 IPv6

Saat Anda membuat permintaan, Anda dapat menentukan titik akhir yang akan digunakan. Jika Anda tidak menentukan titik akhir, IPv4 titik akhir digunakan secara default. Untuk menggunakan tipe titik akhir yang berbeda, Anda harus menentukannya dalam permintaan Anda. Untuk contoh cara melakukannya, lihat the section called “Menentukan titik akhir”. Untuk tabel titik akhir yang tersedia, lihat Titik akhir Amazon VPC Lattice.

IPv4 titik akhir

IPv4 endpoint hanya mendukung IPv4 lalu lintas. IPv4 titik akhir tersedia untuk semua Wilayah.

Jika Anda menentukan titik akhir umum, `vpc-lattice.amazonaws.com`, kami menggunakan titik akhir untuk `us-east-1`. Untuk menggunakan Wilayah yang berbeda, tentukan titik akhir yang terkait. Misalnya, jika Anda menentukan `vpc-lattice.us-east-2.amazonaws.com` sebagai titik akhir, kami mengarahkan permintaan Anda ke titik `us-east-2` akhir.

IPv4 nama endpoint menggunakan konvensi penamaan berikut:

- `vpc-lattice.region.amazonaws.com`

Misalnya, nama IPv4 endpoint untuk `eu-west-1` Region adalah`vpc-lattice.eu-west-1.amazonaws.com`.

Titik akhir Dualstack (IPv4 dan IPv6)

Dualstack endpoint mendukung keduanya IPv4 dan lalu lintas. IPv6 Titik akhir dualstack tersedia untuk semua Wilayah. Saat Anda membuat permintaan ke titik akhir dualstack, URL endpoint akan diselesaikan ke alamat IPv6 atau IPv4 alamat, tergantung pada protokol yang digunakan oleh jaringan dan klien Anda.

Nama titik akhir tumpukan ganda menggunakan konvensi penamaan berikut:

- `vpc-lattice.region.api.aws`

Misalnya, nama titik akhir tumpukan ganda untuk Wilayah `eu-west-1` adalah `vpc-lattice.eu-west-1.api.aws`.

Menentukan titik akhir

Contoh berikut menunjukkan cara menentukan titik akhir untuk `us-east-2` Wilayah menggunakan AWS CLI for`vpc-lattice`.

- IPv4

```
aws vpc-lattice get-service --service-identifier svc-0285b53b2eEXAMPLE --region us-east-2 --endpoint-url https://vpc-lattice.us-east-2.amazonaws.com
```

- Tumpukan ganda

```
aws vpc-lattice get-service --service-identifier svc-0285b53b2eEXAMPLE --region us-east-2 --endpoint-url https://vpc-lattice.us-east-2.api.aws
```

Harga

Dengan VPC Lattice Anda membayar untuk waktu penyediaan layanan, jumlah data yang ditransfer melalui setiap layanan, dan jumlah permintaan. Sebagai pemilik sumber daya, Anda membayar

data yang ditransfer ke dan dari setiap sumber daya. Sebagai pemilik jaringan layanan, Anda membayar setiap jam untuk konfigurasi sumber daya yang terkait dengan jaringan layanan Anda. Sebagai konsumen yang memiliki VPC yang terkait dengan jaringan layanan, Anda membayar data yang ditransfer ke dan dari sumber daya di jaringan layanan dari VPC Anda. Untuk informasi selengkapnya, lihat Harga Kisi VPC Amazon.

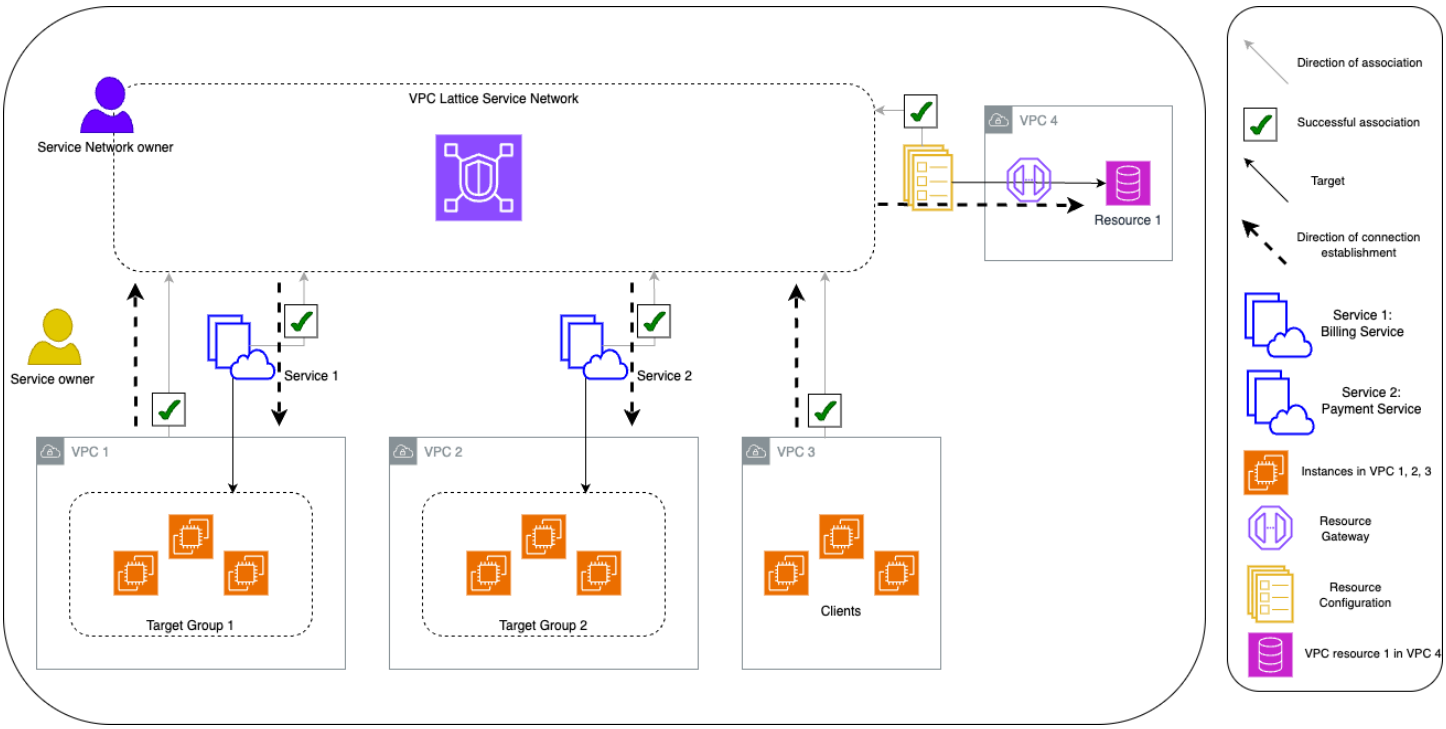
Cara kerja VPC Lattice

VPC Lattice dirancang untuk membantu Anda dengan mudah dan efektif menemukan, mengamankan, menghubungkan, dan memantau semua layanan dan sumber daya di dalamnya. Setiap komponen dalam VPC Lattice berkomunikasi secara searah atau dua arah dalam jaringan layanan berdasarkan hubungannya dengan jaringan layanan dan pengaturan aksesnya. Pengaturan akses terdiri dari kebijakan otentikasi dan otorisasi yang diperlukan untuk komunikasi ini.

Ringkasan berikut menjelaskan komunikasi antar komponen dalam VPC Lattice:

- Ada dua cara VPC dapat dihubungkan ke jaringan layanan - melalui asosiasi VPC dan melalui titik akhir VPC dari jaringan layanan tipe.
- Layanan dan sumber daya yang terkait dengan jaringan layanan dapat menerima permintaan dari klien VPCs yang juga terhubung ke jaringan layanan.
- Klien dapat mengirim permintaan ke layanan dan sumber daya yang terkait dengan jaringan layanan hanya jika berada di VPC yang terhubung ke jaringan layanan yang sama. Lalu lintas klien yang melintasi koneksi peering VPC, gateway transit, Direct Connect, atau VPN dapat menjangkau sumber daya dan layanan hanya jika VPC terhubung ke jaringan layanan melalui titik akhir VPC.
- Target layanan VPCs yang terkait dengan jaringan layanan juga klien dan dapat mengirim permintaan ke layanan dan sumber daya lain yang terkait dengan jaringan layanan.
- Target layanan VPCs yang tidak terkait dengan jaringan layanan bukanlah klien dan tidak dapat mengirim permintaan ke layanan dan sumber daya lain yang terkait dengan jaringan layanan.
- Klien VPCs yang memiliki sumber daya tetapi di mana VPC tidak terkait dengan jaringan layanan bukan klien dan tidak dapat mengirim permintaan ke layanan dan sumber daya lain yang terkait dengan jaringan layanan.

Diagram alir berikut menggunakan contoh skenario untuk menjelaskan aliran informasi dan arah komunikasi antara komponen dalam VPC Lattice. Ada dua layanan yang terkait dengan jaringan layanan. Baik layanan dan semuanya VPCs dibuat dalam akun yang sama dengan jaringan layanan. Kedua layanan dikonfigurasi untuk memungkinkan lalu lintas dari jaringan layanan.



Layanan 1 adalah aplikasi penagihan yang berjalan pada sekelompok instance yang terdaftar dengan grup target 1 di VPC 1. Layanan 2 adalah aplikasi pembayaran yang berjalan pada sekelompok instance yang terdaftar dengan grup target 2 di VPC 2. VPC 3 ada di akun yang sama, dan memiliki klien tetapi tidak ada layanan. Resource 1 adalah database yang memiliki data pelanggan di VPC 4.

Daftar berikut menjelaskan, secara berurutan, alur kerja tipikal tugas untuk VPC Lattice.

1. Buat jaringan layanan

   Pemilik jaringan layanan membuat jaringan layanan.

2. Buat layanan

   Pemilik layanan membuat layanan masing-masing, layanan 1 dan layanan 2. Selama pembuatan, pemilik layanan menambahkan pendengar dan menentukan aturan untuk permintaan perutean ke grup target untuk setiap layanan.

3. Tentukan perutean

   Pemilik layanan membuat grup target untuk setiap layanan (grup target 1 dan grup target 2). Mereka melakukan ini dengan menentukan instance target di mana layanan berjalan. Mereka juga menentukan VPCs di mana target-target ini berada.

Dalam diagram sebelumnya, panah solid mewakili layanan routing lalu lintas ke grup target, dan konfigurasi sumber daya routing ke sumber daya.

VPC Lattice mendukung afinitas Availability Zone (AZ) untuk merutekan lalu lintas. Ketika klien mengirim permintaan ke VPC Lattice, VPC Lattice merespons dengan alamat IP untuk layanan atau sumber daya dari AZ yang sama dengan klien. Jika AZ itu tidak tersedia, VPC Lattice merespons dengan alamat IP dari yang lain. AZs Dari VPC Lattice ke target, routing adalah ke target, yang mungkin didistribusikan ke seluruh. AZs Selain itu, tidak ada biaya transfer data antar-AZ di VPC Lattice.

4. Mengaitkan layanan dengan jaringan layanan

   Pemilik jaringan layanan atau pemilik layanan mengaitkan layanan dengan jaringan layanan. Asosiasi ditampilkan sebagai panah dengan tanda centang yang menunjuk ke jaringan layanan dari layanan. Ketika Anda mengaitkan layanan dengan jaringan layanan, layanan tersebut dapat ditemukan ke layanan lain yang terkait dengan jaringan layanan dan klien yang VPCs terhubung ke jaringan layanan.

   Panah putus-putus antara jaringan layanan dan kelompok sasaran menunjukkan arah pembentukan koneksi. Mengembalikan arus lalu lintas kembali ke klien menggunakan jaringan layanan. Panah yang mewakili lalu lintas yang kembali tidak termasuk dalam diagram ini.

5. Membuat gateway sumber daya

   Pemilik sumber daya membuat gateway sumber daya di VPC 4 agar dapat mengaktifkan konektivitas dari klien ke sumber daya 1.

6. Buat konfigurasi sumber daya

   Pemilik sumber daya membuat konfigurasi sumber daya untuk mewakili sumber daya 1 dan menentukan gateway sumber daya untuk sumber daya 1.

7. Mengaitkan konfigurasi sumber daya dengan jaringan layanan

   Pemilik jaringan layanan atau pemilik sumber daya mengaitkan konfigurasi sumber daya dengan jaringan layanan. Asosiasi ditampilkan sebagai panah dengan tanda centang yang menunjuk ke jaringan layanan dari konfigurasi sumber daya. Ketika Anda mengaitkan konfigurasi sumber daya dengan jaringan layanan, konfigurasi sumber daya tersebut dapat ditemukan ke layanan lain yang terkait dengan jaringan layanan dan klien yang VPCs terhubung ke jaringan layanan.

Panah putus-putus dari jaringan layanan ke sumber daya mewakili sumber daya yang menerima permintaan dari klien. Mengembalikan arus lalu lintas kembali ke klien menggunakan jaringan layanan. Panah yang mewakili lalu lintas yang kembali tidak termasuk dalam diagram ini.

8. Connect VPCs dengan jaringan layanan

   VPCs dapat dihubungkan dengan jaringan layanan dengan dua cara - dengan mengaitkan VPC ke jaringan layanan, atau dengan membuat titik akhir VPC. Di sini, pemilik jaringan layanan mengaitkan VPC 1 dan VPC 3 dengan jaringan layanan. Asosiasi ditampilkan menggunakan panah dengan tanda centang menunjuk ke jaringan layanan. Dengan asosiasi ini, sumber daya apa pun di VPC dapat bertindak sebagai klien, dan dapat membuat permintaan ke layanan dalam jaringan layanan. Panah putus-putus antara VPC 1 dan jaringan layanan menunjukkan arah pembentukan koneksi. Jaringan layanan hanya memulai koneksi ke sumber daya yang ditargetkan oleh kelompok sasaran layanan 1. Setiap sumber daya di VPC 1 dapat bertindak sebagai klien dan memulai koneksi ke layanan jaringan layanan dan sumber daya.

   VPC 2 tidak memiliki tanda panah atau tanda centang yang mewakili asosiasi. Ini berarti bahwa pemilik jaringan layanan atau pemilik layanan belum mengaitkan VPC 2 dengan jaringan layanan. Ini karena layanan 2, dalam contoh ini, hanya perlu menerima permintaan dan mengirim tanggapan menggunakan permintaan yang sama. Dengan kata lain, target untuk layanan 2 bukan klien dan tidak perlu membuat permintaan ke layanan lain di jaringan layanan.

   Demikian pula, VPC 4 tidak memiliki panah atau tanda centang yang mewakili asosiasi. Ini berarti bahwa pemilik jaringan layanan atau pemilik sumber daya belum mengaitkan VPC 4 dengan jaringan layanan. Ini karena sumber daya 1 hanya menerima permintaan dan mengirim tanggapan menggunakan permintaan yang sama. Itu tidak dapat membuat permintaan ke layanan dan sumber daya lain di jaringan layanan.

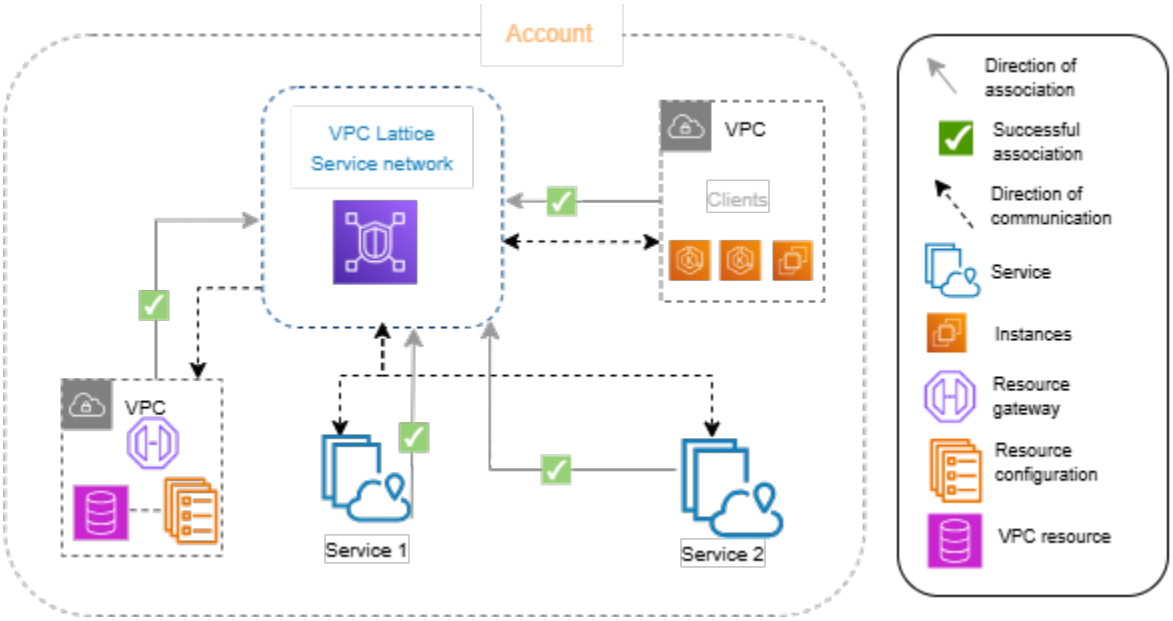
Singkatnya, diagram proses menunjukkan skenario berikut:

- VPCs dengan masuknya hanya koneksi dari VPC Lattice ke sumber dayanya. VPC 2 dan VPC 4 mewakili skenario ini.
- VPC dengan jalan keluar hanya koneksi dari sumber daya mereka ke VPC Lattice. VPC 3 mewakili skenario ini.
- VPC dengan koneksi ingress dari VPC Lattice ke sumber daya mereka dan dengan koneksi keluar dari sumber daya mereka ke VPC Lattice. VPC 1 mewakili skenario ini.

Jaringan layanan di VPC Lattice

Jaringan layanan adalah batas logis untuk kumpulan layanan dan konfigurasi sumber daya. Layanan dan konfigurasi sumber daya yang terkait dengan jaringan dapat diotorisasi untuk penemuan, konektivitas, aksesibilitas, dan observabilitas. Untuk membuat permintaan ke layanan dan konfigurasi sumber daya dalam jaringan, layanan atau klien Anda harus berada dalam VPC yang terhubung ke jaringan layanan baik melalui asosiasi atau melalui titik akhir VPC.

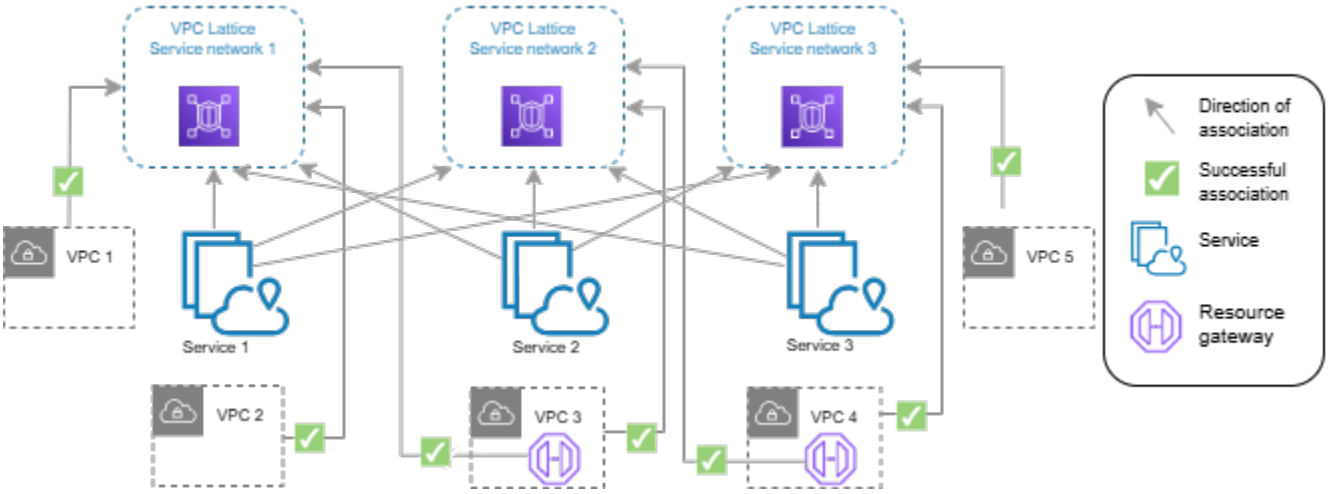
Diagram berikut menunjukkan komponen kunci dari jaringan layanan tipikal dalam Amazon VPC Lattice. Tanda centang pada panah menunjukkan bahwa layanan dan VPC dikaitkan dengan jaringan layanan. Klien di VPC yang terkait dengan jaringan layanan dapat berkomunikasi dengan kedua layanan melalui jaringan layanan.



Anda dapat mengaitkan satu atau lebih layanan dan konfigurasi sumber daya dengan beberapa jaringan layanan. Anda juga dapat menghubungkan beberapa VPCs dengan satu jaringan layanan. Anda dapat menghubungkan VPC ke hanya satu jaringan layanan melalui asosiasi. Untuk menghubungkan VPC ke beberapa jaringan layanan, Anda dapat menggunakan titik akhir VPC dari jenis jaringan layanan. Untuk informasi selengkapnya tentang titik akhir VPC dari jenis jaringan layanan, lihat panduan pengguna.AWS PrivateLink

Dalam diagram berikut, panah mewakili asosiasi antara layanan dan jaringan layanan, serta asosiasi antara jaringan VPCs dan layanan. Anda dapat melihat bahwa beberapa layanan dikaitkan dengan beberapa jaringan layanan, dan beberapa VPCs terkait dengan setiap jaringan layanan. Setiap VPC memiliki persis satu asosiasi ke jaringan layanan. VPC 3 dan VPC 4 namun terhubung ke dua

jaringan layanan. VPC 3 terhubung ke jaringan layanan 1 melalui titik akhir VPC. Demikian pula, VPC 4 terhubung ke jaringan layanan 2 melalui titik akhir VPC.



Untuk informasi selengkapnya, lihat Kuota untuk Amazon VPC Lattice.

Daftar Isi

Buat jaringan layanan VPC Lattice

Gunakan konsol untuk membuat jaringan layanan dan secara opsional mengkonfigurasinya dengan layanan, asosiasi, pengaturan akses, dan log akses.

Untuk membuat jaringan layanan menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih Buat jaringan layanan.

4. Untuk Pengidentifikasi, masukkan nama, deskripsi opsional, dan tag opsional. Nama harus antara 3 dan 63 karakter. Anda dapat menggunakan huruf kecil, angka, dan tanda hubung. Nama harus dimulai dan diakhiri dengan huruf atau angka. Jangan gunakan tanda hubung berturut-turut. Deskripsi dapat memiliki hingga 256 karakter. Untuk menambahkan tag, pilih Tambahkan tag baru dan tentukan kunci tag dan nilai tag.
5. (Opsional) Untuk mengaitkan layanan, pilih layanan dari asosiasi Layanan, Layanan. Daftar ini mencakup layanan yang ada di akun Anda dan layanan apa pun yang dibagikan dengan Anda dari akun yang berbeda. Jika tidak ada layanan dalam daftar, Anda dapat membuat layanan dengan memilih Buat layanan Kisi VPC.

   Atau, untuk mengaitkan layanan setelah Anda membuat jaringan layanan, lihatthe section called “Kelola asosiasi layanan jaringan layanan”.
6. (Opsional) Untuk mengaitkan konfigurasi sumber daya, pilih layanan konfigurasi sumber daya dari asosiasi Konfigurasi Sumber Daya, konfigurasi sumber daya. Daftar ini mencakup konfigurasi sumber daya yang ada di akun Anda dan konfigurasi sumber daya apa pun yang dibagikan dengan Anda dari akun yang berbeda. Jika tidak ada konfigurasi sumber daya dalam daftar, Anda dapat membuat konfigurasi sumber daya dengan memilih Buat konfigurasi sumber daya Amazon VPC Lattice.

   Atau, untuk mengaitkan konfigurasi sumber daya setelah Anda membuat jaringan layanan, lihatthe section called “Kelola asosiasi sumber daya jaringan layanan”.
7. (Opsional) Untuk mengaitkan VPC, pilih Tambahkan asosiasi VPC. Pilih VPC untuk diasosiasikan dari VPC, dan pilih hingga lima grup keamanan dari grup Keamanan. Untuk membuat grup keamanan, pilih Buat grup keamanan baru.

   Atau, Anda dapat melewati langkah ini dan menghubungkan VPC ke jaringan layanan menggunakan titik akhir VPC (didukung oleh). AWS PrivateLink Untuk informasi selengkapnya, lihat Mengakses jaringan layanan di panduan AWS PrivateLink pengguna.
8. Saat membuat jaringan layanan, Anda harus memutuskan apakah Anda berniat berbagi jaringan layanan dengan akun lain atau tidak. Pilihan Anda tidak dapat diubah dan tidak dapat diubah setelah Anda membuat jaringan layanan. Jika Anda memilih untuk mengizinkan berbagi, jaringan layanan dapat dibagikan dengan akun lain melalui AWS Resource Access Manager.

   Untuk berbagi jaringan layanan Anda dengan akun lain, pilih pembagian AWS RAM sumber daya dari Pembagian sumber daya.

Untuk membuat berbagi sumber daya, buka AWS RAM konsol dan pilih Buat berbagi sumber daya.

9. Untuk akses Jaringan, Anda dapat meninggalkan jenis autentikasi default, Tidak Ada, jika Anda ingin klien yang terkait VPCs mengakses layanan di jaringan layanan ini. Untuk menerapkan kebijakan autentikasi untuk mengontrol akses ke layanan Anda, pilih AWS IAM dan lakukan salah satu hal berikut untuk kebijakan Auth:
   - Masukkan kebijakan di kolom input. Misalnya kebijakan yang dapat Anda salin dan tempel, pilih Contoh kebijakan.
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan akses yang diautentikasi dan tidak diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan baik dengan menandatangani permintaan (artinya diautentikasi) atau secara anonim (artinya tidak diautentikasi).
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan hanya akses yang diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan hanya dengan menandatangani permintaan (artinya diautentikasi).
10. (Opsional) Untuk mengaktifkan log akses, pilih sakelar sakelar akses log dan tentukan tujuan untuk log akses Anda sebagai berikut:
    - Pilih Grup CloudWatch log dan pilih grup CloudWatch Log. Untuk membuat grup log, pilih Buat grup log masuk CloudWatch.
    - Pilih bucket S3 dan masukkan path bucket S3, termasuk awalan apa pun. Untuk mencari bucket S3 Anda, pilih Browse S3.
    - Pilih aliran pengiriman Kinesis Data Firehose dan pilih aliran pengiriman. Untuk membuat aliran pengiriman, pilih Buat aliran pengiriman di Kinesis.
11. (Opsional) Untuk berbagi jaringan layanan Anda dengan akun lain, pilih pembagian AWS RAM sumber daya dari pembagian Sumber Daya. Untuk membuat pembagian sumber daya, pilih Buat berbagi sumber daya di konsol RAM.
12. Tinjau konfigurasi Anda di bagian Ringkasan, lalu pilih Buat jaringan layanan.

Untuk membuat jaringan layanan menggunakan AWS CLI

Gunakan perintah create-service-network. Perintah ini hanya menciptakan jaringan layanan dasar. Untuk membuat jaringan layanan yang berfungsi penuh, Anda juga harus menggunakan perintah yang membuat asosiasi layanan, asosiasi VPC, dan pengaturan akses.

Mengelola pengaitan untuk jaringan layanan VPC Lattice

Ketika Anda mengaitkan layanan atau konfigurasi sumber daya dengan jaringan layanan, ini memungkinkan klien yang VPCs terhubung ke jaringan layanan, untuk membuat permintaan ke konfigurasi layanan dan sumber daya. Ketika Anda menghubungkan VPC dengan jaringan layanan, itu memungkinkan semua target dalam VPC itu menjadi klien dan berkomunikasi dengan layanan lain dan konfigurasi sumber daya dalam jaringan layanan.

Properti berkemampuan DNS pribadi dari asosiasi sumber daya jaringan layanan mengesampingkan properti berkemampuan DNS pribadi dari titik akhir jaringan layanan dan asosiasi VPC jaringan layanan.

Jika pemilik jaringan layanan membuat asosiasi sumber daya jaringan layanan dan tidak mengaktifkan DNS pribadi, VPC Lattice tidak akan menyediakan zona yang dihosting pribadi untuk konfigurasi sumber daya tersebut di VPCs mana pun jaringan layanan terhubung, meskipun DNS pribadi diaktifkan pada titik akhir jaringan layanan atau asosiasi VPC jaringan layanan.

Daftar Isi

Kelola asosiasi layanan jaringan layanan

Anda dapat mengaitkan layanan yang berada di akun Anda atau layanan yang dibagikan dengan Anda dari akun yang berbeda. Ini adalah langkah opsional saat membuat jaringan layanan. Namun, jaringan layanan tidak berfungsi penuh sampai Anda mengaitkan layanan. Pemilik layanan dapat mengaitkan layanan mereka ke jaringan layanan jika akun mereka memiliki akses yang diperlukan. Untuk informasi selengkapnya, lihat Contoh kebijakan berbasis identitas untuk VPC Lattice.

Ketika Anda menghapus asosiasi layanan, layanan tidak dapat lagi terhubung ke layanan lain di jaringan layanan.

Untuk mengelola asosiasi layanan menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.

3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab Asosiasi layanan.
5. Untuk membuat asosiasi, lakukan hal berikut:

    a. Pilih Buat asosiasi.
    b. Pilih layanan dari Layanan. Untuk membuat layanan, pilih Buat layanan Amazon VPC Lattice.
    c. (Opsional) Untuk menambahkan tag, perluas tag asosiasi layanan, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
    d. Pilih Simpan perubahan.
6. Untuk menghapus asosiasi, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Hapus asosiasi layanan. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk membuat asosiasi layanan menggunakan AWS CLI

Gunakan perintah create-service-network-service-association.

Untuk menghapus asosiasi layanan menggunakan AWS CLI

Gunakan perintah delete-service-network-service-association.

Kelola asosiasi sumber daya jaringan layanan

Konfigurasi sumber daya adalah objek logis yang mewakili sumber daya tunggal atau sekelompok sumber daya. Anda dapat mengaitkan konfigurasi sumber daya yang ada di akun atau konfigurasi sumber daya yang dibagikan dengan Anda dari akun yang berbeda. Ini adalah langkah opsional saat membuat jaringan layanan. Pemilik konfigurasi sumber daya dapat mengaitkan konfigurasi sumber daya mereka ke jaringan layanan jika akun mereka memiliki akses yang diperlukan. Untuk informasi selengkapnya, lihat Contoh kebijakan berbasis identitas untuk VPC Lattice.

Mengelola asosiasi antara jaringan layanan dan konfigurasi sumber daya

Anda dapat membuat atau menghapus asosiasi antara jaringan layanan dan konfigurasi sumber daya.

Untuk mengelola asosiasi konfigurasi sumber daya menggunakan konsol

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

2. Di panel navigasi, di bawah PrivateLink dan Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab Asosiasi konfigurasi sumber daya.
5. Untuk membuat asosiasi, lakukan hal berikut:

    a. Pilih Buat asosiasi.
    b. Untuk konfigurasi Sumber Daya, pilih konfigurasi sumber daya.
    c. Untuk nama DNS, pilih DNS pribadi yang diaktifkan untuk mengizinkan Kisi VPC menyediakan zona host pribadi untuk asosiasi konfigurasi sumber daya Anda berdasarkan nama domain konfigurasi sumber daya.
    d. (Opsional) Untuk menambahkan tag, perluas tag asosiasi layanan, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
    e. Pilih Simpan perubahan.
6. Untuk menghapus asosiasi, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Hapus. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk membuat asosiasi konfigurasi sumber daya menggunakan AWS CLI

Gunakan perintah create-service-network-resource-association.

Untuk menghapus asosiasi konfigurasi sumber daya menggunakan AWS CLI

Gunakan perintah delete-service-network-resource-association.

Kelola asosiasi VPC jaringan layanan

Klien dapat mengirim permintaan ke layanan dan sumber daya yang ditentukan dalam konfigurasi sumber daya yang terkait dengan jaringan layanan jika klien VPCs terkait dengan jaringan layanan. Lalu lintas klien yang melintasi koneksi peering VPC atau gateway transit hanya diperbolehkan melalui jaringan layanan menggunakan titik akhir VPC dari jaringan layanan tipe.

Mengaitkan VPC adalah langkah opsional saat Anda membuat jaringan layanan. Pemilik jaringan dapat mengasosiasikan VPCs ke jaringan layanan jika akun mereka memiliki akses yang diperlukan. Untuk informasi selengkapnya, lihat Contoh kebijakan berbasis identitas untuk VPC Lattice.

Saat membuat asosiasi VPC ke konfigurasi sumber daya, Anda dapat menentukan preferensi DNS pribadi. Preferensi ini memungkinkan VPC Lattice untuk menyediakan zona host pribadi atas nama

konsumen sumber daya. Untuk informasi selengkapnya, lihat the section called “Nama domain khusus untuk penyedia sumber daya”.

Ketika Anda menghapus asosiasi VPC, klien di tidak VPCs dapat lagi terhubung ke layanan di jaringan layanan.

Untuk mengelola asosiasi VPC menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab asosiasi VPC.
5. Untuk membuat asosiasi VPC, lakukan hal berikut:
   a. Pilih Buat asosiasi VPC.
   b. Pilih Tambahkan asosiasi VPC.
   c. Pilih VPC dari VPC dan pilih hingga lima grup keamanan dari grup Keamanan. Untuk membuat grup keamanan, pilih Buat grup keamanan baru.
   d. (Opsional) Untuk mengizinkan VPC Lattice menyediakan zona host pribadi berdasarkan nama domain konfigurasi sumber daya, untuk nama DNS, pilih Aktifkan nama DNS dan lakukan hal berikut:
      i. Untuk preferensi DNS Pribadi, pilih preferensi.

         Jika Anda memilih Semua domain, VPC Lattice menyediakan zona host pribadi untuk nama domain kustom apa pun untuk konfigurasi sumber daya.
      ii. (Opsional) Jika Anda memilih Domain terverifikasi dan yang ditentukan atau Domain tertentu, masukkan daftar domain yang dipisahkan koma yang Anda inginkan untuk menyediakan VPC Lattice untuk zona yang dihosting. VPC Lattice hanya menyediakan zona yang dihosting jika cocok dengan daftar domain pribadi Anda. Anda dapat menggunakan pencocokan wildcard.
   e. (Opsional) Untuk menambahkan tag, perluas tag asosiasi VPC, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
   f. Pilih Simpan perubahan.
6. Untuk mengedit grup keamanan untuk asosiasi, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Edit grup keamanan. Tambahkan dan hapus grup keamanan sesuai kebutuhan.

7. Untuk menghapus asosiasi, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Hapus asosiasi VPC. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk membuat asosiasi VPC menggunakan AWS CLI

Gunakan perintah create-service-network-vpc-association.

Untuk memperbarui grup keamanan untuk asosiasi VPC menggunakan AWS CLI

Gunakan perintah update-service-network-vpc-association.

Untuk menghapus asosiasi VPC menggunakan AWS CLI

Gunakan perintah delete-service-network-vpc-association.

Kelola asosiasi titik akhir VPC jaringan layanan

Klien dapat mengirim permintaan ke layanan dan sumber daya yang ditentukan dalam konfigurasi sumber daya melalui titik akhir VPC (didukung AWS PrivateLink oleh) di VPC mereka. Titik akhir VPC dari jaringan layanan tipe menghubungkan VPC ke jaringan layanan. Lalu lintas klien yang berasal dari luar VPC melalui koneksi peering VPC, Transit Gateway, Direct Connect, atau VPN dapat menggunakan titik akhir VPC untuk menjangkau layanan dan konfigurasi sumber daya. Dengan titik akhir VPC, Anda dapat menghubungkan VPC ke beberapa jaringan layanan. Saat Anda membuat titik akhir VPC di VPC, alamat IP dari VPC (dan bukan alamat IP dari daftar awalan terkelola) digunakan untuk membangun konektivitas ke jaringan layanan.

Saat membuat asosiasi VPC ke konfigurasi sumber daya, Anda dapat menentukan preferensi DNS pribadi. Preferensi ini memungkinkan VPC Lattice untuk menyediakan zona host pribadi atas nama konsumen sumber daya. Untuk informasi selengkapnya, lihat the section called “Nama domain khusus untuk penyedia sumber daya”.

Untuk mengelola asosiasi titik akhir VPC menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab Asosiasi titik akhir untuk melihat titik akhir VPC yang terhubung ke jaringan layanan Anda.

5. Pilih ID Endpoint dari titik akhir VPC untuk membuka halaman detailnya. Kemudian ubah atau hapus asosiasi titik akhir VPC.

Untuk membuat asosiasi titik akhir VPC baru menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Endpoints.
3. Pilih Buat titik akhir.
4. Untuk Jenis, pilih Jaringan layanan.
5. Pilih jaringan layanan yang ingin Anda sambungkan ke VPC Anda.
6. Pilih VPC, subnet, dan grup keamanan.
7. (Opsional) Untuk mengaktifkan DNS pribadi, pilih Aktifkan DNS pribadi.
8. (Opsional) Untuk menambahkan tag, perluas tag asosiasi VPC, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
9. Pilih Buat titik akhir.

Untuk mempelajari lebih lanjut tentang titik akhir VPC tentang cara menyambung ke jaringan layanan, lihat Mengakses jaringan layanan di panduan pengguna.AWS PrivateLink

Mengedit setelan akses untuk jaringan layanan VPC Lattice

Pengaturan akses memungkinkan Anda untuk mengkonfigurasi dan mengelola akses klien ke jaringan layanan. Pengaturan akses mencakup jenis autentikasi dan kebijakan autentikasi. Kebijakan autentikasi membantu Anda mengautentikasi dan mengotorisasi lalu lintas yang mengalir ke layanan dalam VPC Lattice. Pengaturan akses jaringan layanan tidak berlaku untuk konfigurasi sumber daya yang terkait dengan jaringan layanan.

Anda dapat menerapkan kebijakan autentikasi di tingkat jaringan layanan, tingkat layanan, atau keduanya. Biasanya, kebijakan autentikasi diterapkan oleh pemilik jaringan atau administrator cloud. Mereka dapat menerapkan otorisasi kasar, misalnya, memungkinkan panggilan yang diautentikasi dari dalam organisasi, atau mengizinkan permintaan GET anonim yang cocok dengan kondisi tertentu. Pada tingkat layanan, pemilik layanan dapat menerapkan kontrol berbutir halus, yang bisa lebih membatasi. Untuk informasi selengkapnya, lihat Kontrol akses ke layanan VPC Lattice menggunakan kebijakan autentikasi.

Untuk menambah atau memperbarui kebijakan akses menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab Access untuk memeriksa pengaturan akses saat ini.
5. Untuk memperbarui pengaturan akses, pilih Edit pengaturan akses.
6. Jika Anda ingin klien yang VPCs terkait mengakses layanan di jaringan layanan ini, pilih None for Auth type.
7. Untuk menerapkan kebijakan sumber daya ke jaringan layanan, pilih AWS IAM untuk jenis Auth dan lakukan kebijakan Auth berikut ini:
   - Masukkan kebijakan di kolom input. Misalnya kebijakan yang dapat Anda salin dan tempel, pilih Contoh kebijakan.
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan akses yang diautentikasi dan tidak diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan baik dengan menandatangani permintaan (artinya diautentikasi) atau secara anonim (artinya tidak diautentikasi).
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan hanya akses yang diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan hanya dengan menandatangani permintaan (artinya diautentikasi).
8. Pilih Simpan perubahan.

Untuk menambah atau memperbarui kebijakan akses menggunakan AWS CLI

Gunakan perintah put-auth-policy.

Mengedit detail pemantauan untuk jaringan layanan VPC Lattice

VPC Lattice menghasilkan metrik dan log untuk setiap permintaan dan respons, membuatnya lebih efisien untuk memantau dan memecahkan masalah aplikasi.

Anda dapat mengaktifkan log akses dan menentukan sumber daya tujuan untuk log Anda. VPC Lattice dapat mengirim log ke sumber daya berikut: Grup CloudWatch log, aliran pengiriman Firehose, dan bucket S3.

Untuk mengaktifkan log akses atau memperbarui tujuan log menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab Pemantauan. Periksa log Access untuk melihat apakah log akses diaktifkan.
5. Untuk mengaktifkan atau menonaktifkan log akses, pilih Edit log akses, lalu nyalakan atau nonaktifkan sakelar Access log.
6. Ketika Anda mengaktifkan log akses, Anda harus memilih jenis tujuan pengiriman, dan kemudian membuat atau memilih tujuan untuk log akses. Anda juga dapat mengubah tujuan pengiriman kapan saja. Contoh:
   - Pilih Grup CloudWatch log dan pilih grup CloudWatch Log. Untuk membuat grup log, pilih Buat grup log masuk CloudWatch.
   - Pilih bucket S3 dan masukkan path bucket S3, termasuk awalan apa pun. Untuk mencari bucket S3 Anda, pilih Browse S3.
   - Pilih aliran pengiriman Kinesis Data Firehose dan pilih aliran pengiriman. Untuk membuat aliran pengiriman, pilih Buat aliran pengiriman di Kinesis.
7. Pilih Simpan perubahan.

Untuk mengaktifkan log akses menggunakan AWS CLI

Gunakan perintah create-access-log-subscription.

Untuk memperbarui tujuan log menggunakan AWS CLI

Gunakan perintah update-access-log-subscription.

Untuk menonaktifkan log akses menggunakan AWS CLI

Gunakan perintah delete-access-log-subscription.

Mengelola tag untuk jaringan layanan VPC Lattice

Tag membantu Anda untuk mengkategorikan jaringan layanan Anda dengan cara yang berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan.

Anda dapat menambahkan beberapa tag ke setiap jaringan layanan. Kunci tag harus unik untuk setiap jaringan layanan. Jika Anda menambahkan tag dengan kunci yang sudah dikaitkan dengan

jaringan layanan, itu memperbarui nilai tag tersebut. Anda dapat menggunakan karakter seperti huruf, spasi, angka (dalam UTF-8), dan karakter khusus berikut: + - =. _:/@. Jangan gunakan spasi terkemuka atau paling belakang. Kunci dan nilai tanda peka huruf besar dan kecil.

Untuk menambah atau menghapus tag menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pilih tab Tanda.
5. Untuk menambahkan tag, pilih Tambahkan tag dan masukkan kunci tag dan nilai tag. Untuk menambahkan tag lain, pilih Tambahkan tag baru. Setelah Anda selesai menambahkan tanda, pilih Simpan perubahan.
6. Untuk menghapus tag, pilih kotak centang untuk tag dan pilih Hapus. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk menambah atau menghapus tag menggunakan AWS CLI

Gunakan perintah tag-resource dan untag-resource.

Hapus jaringan layanan VPC Lattice

Sebelum Anda dapat menghapus jaringan layanan, Anda harus terlebih dahulu menghapus semua asosiasi yang mungkin dimiliki jaringan layanan dengan layanan, konfigurasi sumber daya, VPC, atau titik akhir VPC apa pun. Saat Anda menghapus jaringan layanan, kami juga menghapus semua sumber daya yang terkait dengan jaringan layanan, seperti kebijakan sumber daya, kebijakan autentikasi, dan langganan log akses.

Untuk menghapus jaringan layanan menggunakan konsol

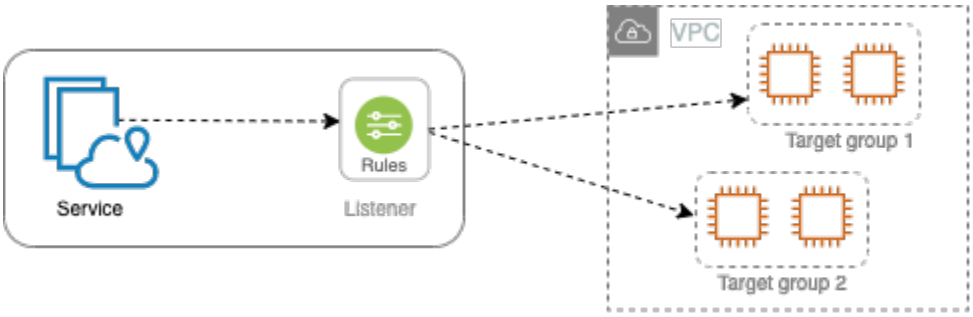
1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih kotak centang untuk jaringan layanan, lalu pilih Tindakan, Hapus jaringan layanan.
4. Saat diminta konfirmasi, masukkan**`confirm`**, lalu pilih Hapus.

Untuk menghapus jaringan layanan menggunakan AWS CLI

Gunakan perintah delete-service-network.

Layanan di VPC Lattice

Layanan dalam VPC Lattice adalah unit perangkat lunak yang dapat digunakan secara independen yang memberikan tugas atau fungsi tertentu. Layanan dapat berjalan pada instance, container, atau sebagai fungsi tanpa server dalam akun atau virtual private cloud (VPC). Layanan memiliki listener yang menggunakan aturan, yang disebut aturan listener, yang dapat Anda konfigurasikan untuk membantu merutekan lalu lintas ke target Anda. Jenis target yang didukung meliputi EC2 instance, alamat IP, fungsi Lambda, Application Load Balancer, tugas Amazon ECS, dan Kubernetes Pods. Untuk informasi selengkapnya, lihat Grup sasaran di VPC Lattice. Anda dapat mengaitkan layanan dengan beberapa jaringan layanan. Diagram berikut menunjukkan komponen kunci dari layanan tipikal dalam VPC Lattice.



Anda dapat membuat layanan dengan memberinya nama dan deskripsi. Namun, untuk mengontrol dan memantau lalu lintas ke layanan Anda, penting bagi Anda untuk menyertakan pengaturan akses dan detail pemantauan. Untuk mengirim lalu lintas dari layanan ke target, Anda harus menyiapkan pendengar dan mengonfigurasi aturan. Untuk memungkinkan lalu lintas mengalir dari jaringan layanan ke layanan Anda, Anda harus mengaitkan layanan Anda dengan jaringan layanan.

Ada batas waktu idle dan batas waktu koneksi keseluruhan untuk koneksi ke target. Batas waktu koneksi idle adalah 1 menit, setelah itu kami menutup koneksi. Durasi maksimum adalah 10 menit, setelah itu kami tidak mengizinkan aliran baru melalui koneksi dan kami memulai proses penutupan aliran yang ada.

Tugas

- Langkah 1: Buat layanan VPC Lattice
- Langkah 2: Tentukan perutean
- Langkah 3: Buat asosiasi jaringan
- Langkah 4: Tinjau dan buat
- Mengelola asosiasi untuk layanan VPC Lattice

- Mengedit setelan akses untuk layanan VPC Lattice
- Mengedit detail pemantauan untuk layanan VPC Lattice
- Mengelola tag untuk layanan VPC Lattice
- Konfigurasikan nama domain khusus untuk layanan VPC Lattice Anda
- Bawa Sertifikat Anda Sendiri (BYOC) untuk Kisi VPC
- Hapus layanan VPC Lattice

Langkah 1: Buat layanan VPC Lattice

Buat layanan VPC Lattice dasar dengan pengaturan akses dan detail pemantauan. Namun, layanan ini tidak berfungsi penuh sampai Anda menentukan konfigurasi routing dan mengaitkannya dengan jaringan layanan.

Untuk membuat layanan dasar menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih Buat layanan.
4. Untuk Identifier, lakukan hal berikut:

   a. Masukkan nama untuk layanan ini. Nama harus antara 3-40 karakter dan menggunakan huruf kecil, angka, dan tanda hubung. Itu harus dimulai dan diakhiri dengan huruf atau angka. Jangan gunakan tanda hubung ganda.

   b. (Opsional) Masukkan deskripsi untuk jaringan layanan. Anda dapat mengatur atau mengubah deskripsi selama atau setelah pembuatan. Deskripsi dapat memiliki hingga 256 karakter.
5. Untuk menentukan nama domain kustom untuk layanan Anda, pilih Tentukan konfigurasi domain kustom dan masukkan nama domain kustom.

   Untuk pendengar HTTPS, Anda dapat memilih sertifikat yang akan digunakan VPC Lattice untuk melakukan penghentian TLS. Jika Anda tidak memilih sertifikat sekarang, Anda dapat memilihnya saat membuat pendengar HTTPS untuk layanan tersebut.

   Untuk pendengar TCP, Anda harus menentukan nama domain khusus untuk layanan Anda. Jika Anda menentukan sertifikat, itu tidak digunakan. Sebagai gantinya, Anda melakukan penghentian TLS dalam aplikasi Anda.

6. Untuk akses Layanan, pilih Tidak Ada jika Anda ingin klien yang VPCs terkait dengan jaringan layanan mengakses layanan Anda. Untuk menerapkan kebijakan autentikasi untuk mengontrol akses ke layanan, pilih AWS IAM. Untuk menerapkan kebijakan sumber daya ke layanan, lakukan salah satu hal berikut untuk kebijakan Auth:
   - Masukkan kebijakan di kolom input. Misalnya kebijakan yang dapat Anda salin dan tempel, pilih Contoh kebijakan.
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan akses yang diautentikasi dan tidak diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan baik dengan menandatangani permintaan (artinya diautentikasi) atau secara anonim (artinya tidak diautentikasi).
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan hanya akses yang diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan hanya dengan menandatangani permintaan (artinya diautentikasi).
7. (Opsional) Untuk mengaktifkan log akses, aktifkan sakelar sakelar sakelar akses log dan tentukan tujuan untuk log akses Anda sebagai berikut:
   - Pilih Grup CloudWatch log dan pilih grup CloudWatch Log. Untuk membuat grup log, pilih Buat grup log masuk CloudWatch.
   - Pilih bucket S3 dan masukkan path bucket S3, termasuk awalan apa pun. Untuk mencari bucket S3 Anda, pilih Browse S3.
   - Pilih aliran pengiriman Kinesis Data Firehose dan pilih aliran pengiriman. Untuk membuat aliran pengiriman, pilih Buat aliran pengiriman di Kinesis.
8. (Opsional) Untuk membagikan layanan Anda dengan akun lain, pilih pembagian AWS RAM sumber daya dari Pembagian sumber daya. Untuk membuat pembagian sumber daya, pilih Buat berbagi sumber daya di konsol RAM.
9. Untuk meninjau konfigurasi dan membuat layanan, pilih Lewati untuk meninjau dan membuat. Jika tidak, pilih Berikutnya untuk menentukan konfigurasi routing untuk layanan Anda.

Langkah 2: Tentukan perutean

Tentukan konfigurasi perutean Anda menggunakan pendengar sehingga layanan Anda dapat mengirim lalu lintas ke target yang Anda tentukan.

Prasyarat

Sebelum Anda dapat menambahkan listener, Anda harus membuat grup target VPC Lattice. Untuk informasi selengkapnya, lihat the section called “Buat grup target”.

Untuk menentukan perutean untuk layanan Anda menggunakan konsol

1. Pilih Tambahkan pendengar.
2. Untuk nama Listener, Anda dapat memberikan nama pendengar kustom atau menggunakan protokol dan port listener Anda sebagai nama listener. Nama kustom yang Anda tentukan dapat memiliki hingga 63 karakter, dan itu harus unik untuk setiap layanan di akun Anda. Karakter yang valid adalah a-z, 0-9, dan tanda hubung (-). Anda tidak dapat menggunakan tanda hubung sebagai karakter pertama atau terakhir, atau segera setelah tanda hubung lainnya. Anda tidak dapat mengubah nama pendengar setelah Anda membuatnya.
3. Pilih protokol dan kemudian masukkan nomor port.
4. Untuk tindakan Default, pilih grup target VPC Lattice untuk menerima lalu lintas dan pilih bobot yang akan ditetapkan ke grup target ini. Anda dapat menambahkan grup target lain secara opsional untuk tindakan default. Pilih Tambah tindakan dan kemudian pilih grup target lain dan tentukan bobotnya.
5. (Opsional) Untuk menambahkan aturan lain, pilih Tambahkan aturan lalu masukkan nama, prioritas, kondisi, dan tindakan untuk aturan tersebut.

   Anda dapat memberikan setiap aturan nomor prioritas antara 1 dan 100. Listener tidak bisa memiliki beberapa aturan dengan prioritas yang sama. Peraturan dievaluasi dalam urutan prioritas, dari nilai terendah ke nilai tertinggi. Peraturan default dievaluasi terakhir.

   Untuk Kondisi, masukkan pola jalur untuk kondisi pencocokan jalur. Ukuran maksimum setiap string adalah 200 karakter. Perbandingannya tidak peka huruf besar/kecil.
6. (Opsional) Untuk menambahkan tag, perluas tag Listener, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
7. Untuk meninjau konfigurasi dan membuat layanan, pilih Lewati untuk meninjau dan membuat. Jika tidak, pilih Berikutnya untuk mengaitkan layanan Anda ke jaringan layanan.

Langkah 3: Buat asosiasi jaringan

Kaitkan layanan Anda dengan jaringan layanan sehingga klien dapat berkomunikasi dengannya.

Untuk mengaitkan layanan ke jaringan layanan menggunakan konsol

1. Untuk jaringan layanan VPC Lattice, pilih jaringan layanan. Untuk membuat jaringan layanan, pilih Buat jaringan kisi VPC. Anda dapat mengaitkan layanan Anda dengan beberapa jaringan layanan.
2. (Opsional) Untuk menambahkan tag, perluas tag asosiasi jaringan layanan, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
3. Pilih Berikutnya.

Langkah 4: Tinjau dan buat

Untuk meninjau konfigurasi dan membuat layanan menggunakan konsol

1. Tinjau konfigurasi untuk layanan Anda.
2. Pilih Edit jika Anda perlu memodifikasi bagian mana pun dari konfigurasi layanan.
3. Setelah selesai meninjau atau mengedit konfigurasi, pilih layanan Create VPC Lattice.
4. Jika Anda menentukan nama domain khusus untuk layanan, Anda harus mengonfigurasi perutean DNS setelah layanan dibuat. Untuk informasi selengkapnya, lihat the section called “Konfigurasikan nama domain khusus”.

Mengelola asosiasi untuk layanan VPC Lattice

Ketika Anda mengaitkan layanan dengan jaringan layanan, ini memungkinkan klien (sumber daya dalam VPC yang terkait dengan jaringan layanan), untuk membuat permintaan ke layanan ini. Anda dapat mengaitkan layanan yang ada di akun Anda atau layanan yang dibagikan dengan Anda dari akun yang berbeda. Langkah ini opsional saat membuat layanan. Namun, setelah pembuatan, layanan tidak dapat berkomunikasi dengan layanan lain sampai Anda mengaitkannya dengan jaringan layanan. Pemilik layanan dapat mengaitkan layanan mereka ke jaringan layanan jika akun mereka memiliki akses yang diperlukan. Untuk informasi selengkapnya, lihat Cara kerja VPC Lattice.

Untuk mengelola asosiasi jaringan layanan menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.

4. Pilih tab Asosiasi jaringan layanan.
5. Untuk membuat asosiasi, lakukan hal berikut:
   a. Pilih Buat asosiasi.
   b. Pilih jaringan layanan dari jaringan layanan VPC Lattice. Untuk membuat jaringan layanan, pilih Buat jaringan kisi VPC.
   c. (Opsional) Untuk menambahkan tag, perluas tag asosiasi layanan, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
   d. Pilih Simpan perubahan.
6. Untuk menghapus asosiasi, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Hapus asosiasi jaringan. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk membuat asosiasi jaringan layanan menggunakan AWS CLI

Gunakan perintah create-service-network-service-association.

Untuk menghapus asosiasi jaringan layanan menggunakan AWS CLI

Gunakan perintah delete-service-network-service-association.

Mengedit setelan akses untuk layanan VPC Lattice

Pengaturan akses memungkinkan Anda mengonfigurasi dan mengelola akses klien ke layanan. Pengaturan akses mencakup jenis autentikasi dan kebijakan autentikasi. Kebijakan autentikasi membantu Anda mengautentikasi dan mengotorisasi lalu lintas yang mengalir ke layanan dalam VPC Lattice.

Anda dapat menerapkan kebijakan autentikasi di tingkat jaringan layanan, tingkat layanan, atau keduanya. Pada tingkat layanan, pemilik layanan dapat menerapkan kontrol berbutir halus, yang bisa lebih membatasi. Biasanya, kebijakan autentikasi diterapkan oleh pemilik jaringan atau administrator cloud. Mereka dapat menerapkan otorisasi berbutir kursus, misalnya, mengizinkan panggilan yang diautentikasi dari dalam organisasi, atau mengizinkan permintaan GET anonim yang cocok dengan kondisi tertentu. Untuk informasi selengkapnya, lihat Kontrol akses ke layanan VPC Lattice menggunakan kebijakan autentikasi.

Untuk menambah atau memperbarui kebijakan akses menggunakan konsol

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

2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pilih tab Access untuk memeriksa pengaturan akses saat ini.
5. Untuk memperbarui pengaturan akses, pilih Edit pengaturan akses.
6. Jika Anda ingin klien VPCs di jaringan layanan terkait mengakses layanan Anda, pilih None for Auth type.
7. Untuk menerapkan kebijakan sumber daya untuk mengontrol akses ke layanan, pilih AWS IAM untuk jenis Auth dan lakukan satu hal berikut untuk kebijakan Auth:
   - Masukkan kebijakan di kolom input. Misalnya kebijakan yang dapat Anda salin dan tempel, pilih Contoh kebijakan.
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan akses yang diautentikasi dan tidak diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan baik dengan menandatangani permintaan (artinya diautentikasi) atau secara anonim (artinya tidak diautentikasi).
   - Pilih Terapkan templat kebijakan dan pilih templat Izinkan hanya akses yang diautentikasi. Template ini memungkinkan klien dari akun lain untuk mengakses layanan hanya dengan menandatangani permintaan (artinya diautentikasi).
8. Pilih Simpan perubahan.

Untuk menambah atau memperbarui kebijakan akses menggunakan AWS CLI

Gunakan perintah put-auth-policy.

Mengedit detail pemantauan untuk layanan VPC Lattice

VPC Lattice menghasilkan metrik dan log untuk setiap permintaan dan respons, membuatnya lebih efisien untuk memantau dan memecahkan masalah aplikasi.

Anda dapat mengaktifkan log akses dan menentukan sumber daya tujuan untuk log Anda. VPC Lattice dapat mengirim log ke sumber daya berikut: Grup CloudWatch log, aliran pengiriman Firehose, dan bucket S3.

Untuk mengaktifkan log akses atau memperbarui tujuan log menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.

3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pilih tab Monitoring dan kemudian pilih Log. Periksa log Access untuk melihat apakah log akses diaktifkan.
5. Untuk mengaktifkan atau menonaktifkan log akses, pilih Edit log akses, lalu nyalakan atau nonaktifkan sakelar Access log.
6. Ketika Anda mengaktifkan log akses, Anda harus memilih jenis tujuan pengiriman, dan kemudian membuat atau memilih tujuan untuk log akses. Anda juga dapat mengubah tujuan pengiriman kapan saja. Contoh:
   - Pilih Grup CloudWatch log dan pilih grup CloudWatch Log. Untuk membuat grup log, pilih Buat grup log masuk CloudWatch.
   - Pilih bucket S3 dan masukkan path bucket S3, termasuk awalan apa pun. Untuk mencari bucket S3 Anda, pilih Browse S3.
   - Pilih aliran pengiriman Kinesis Data Firehose dan pilih aliran pengiriman. Untuk membuat aliran pengiriman, pilih Buat aliran pengiriman di Kinesis.
7. Pilih Simpan perubahan.

Untuk mengaktifkan log akses menggunakan AWS CLI

Gunakan perintah create-access-log-subscription.

Untuk memperbarui tujuan log menggunakan AWS CLI

Gunakan perintah update-access-log-subscription.

Untuk menonaktifkan log akses menggunakan AWS CLI

Gunakan perintah delete-access-log-subscription.

Mengelola tag untuk layanan VPC Lattice

Tag membantu Anda untuk mengkategorikan layanan Anda dengan cara yang berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan.

Anda dapat menambahkan beberapa tag ke setiap layanan. Kunci tag harus unik untuk setiap layanan. Jika Anda menambahkan tag dengan kunci yang sudah dikaitkan dengan layanan, itu memperbarui nilai tag tersebut. Anda dapat menggunakan karakter seperti huruf, spasi, angka (dalam UTF-8), dan karakter khusus berikut: + - =. _:/@. Jangan gunakan spasi terkemuka atau paling belakang. Kunci dan nilai tanda peka huruf besar dan kecil.

Untuk menambah atau menghapus tag menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pilih tab Tanda.
5. Untuk menambahkan tag, pilih Tambahkan tag dan masukkan kunci tag dan nilai tag. Untuk menambahkan tag lain, pilih Tambahkan tag baru. Setelah Anda selesai menambahkan tanda, pilih Simpan perubahan.
6. Untuk menghapus tag, pilih kotak centang untuk tag dan pilih Hapus. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk menambah atau menghapus tag menggunakan AWS CLI

Gunakan perintah tag-resource dan untag-resource.

Konfigurasikan nama domain khusus untuk layanan VPC Lattice Anda

Saat Anda membuat layanan baru, VPC Lattice menghasilkan Nama Domain Berkualitas Penuh (FQDN) yang unik untuk layanan dengan sintaks berikut.

```
service_name-service_id.partition_id.vpc-lattice-svcs.region.on.aws
```

Namun, nama domain yang disediakan VPC Lattice tidak mudah diingat oleh pengguna Anda. Nama domain khusus lebih sederhana dan lebih intuitif URLs yang dapat Anda berikan kepada pengguna Anda. Jika Anda lebih suka menggunakan nama domain khusus untuk layanan Anda, seperti `www.parking.example.com` alih-alih nama DNS yang dihasilkan VPC Lattice, Anda dapat mengonfigurasinya saat membuat layanan VPC Lattice. Saat klien membuat permintaan menggunakan nama domain kustom Anda, server DNS menyelesaikannya ke nama domain yang dihasilkan VPC Lattice.

Prasyarat

- Anda harus memiliki nama domain terdaftar untuk layanan Anda. Jika Anda belum memiliki nama domain terdaftar, Anda dapat mendaftarkannya melalui Amazon Route 53 atau registrar komersial lainnya.

- Untuk menerima permintaan HTTPS, Anda harus memberikan sertifikat Anda sendiri di AWS Certificate Manager. VPC Lattice tidak mendukung sertifikat default sebagai fallback. Oleh karena itu, jika Anda tidak memberikan SSL/TLS sertifikat yang sesuai dengan nama domain kustom Anda, semua koneksi HTTPS ke nama domain kustom Anda akan gagal. Untuk informasi selengkapnya, lihat Bawa Sertifikat Anda Sendiri (BYOC) untuk Kisi VPC.

Keterbatasan dan pertimbangan

- Anda tidak dapat memiliki lebih dari satu nama domain khusus untuk suatu layanan.
- Anda tidak dapat mengubah nama domain kustom setelah Anda membuat layanan.
- Nama domain khusus harus unik untuk jaringan layanan. Ini berarti bahwa layanan tidak dapat dibuat dengan nama domain kustom yang sudah ada (untuk layanan lain) di jaringan layanan yang sama.

Prosedur berikut menunjukkan cara mengonfigurasi nama domain khusus untuk layanan Anda.

Konsol Manajemen AWS

Untuk mengonfigurasi nama domain khusus untuk layanan Anda

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih Buat Layanan. Anda dinavigasi ke Langkah 1: Buat layanan.
4. Di bagian Konfigurasi domain kustom, pilih Tentukan konfigurasi domain khusus.
5. Masukkan nama domain kustom Anda.
6. Untuk melayani permintaan HTTPS, pilih SSL/TLS sertifikat yang cocok dengan nama domain kustom Anda di Custom SSL/TLS certificate. Jika Anda belum memiliki sertifikat, atau tidak ingin menambahkannya sekarang, Anda dapat menambahkan sertifikat saat membuat pendengar HTTPS. Namun, tanpa sertifikat, nama domain kustom Anda tidak akan dapat melayani permintaan HTTPS. Untuk informasi selengkapnya, lihat Menambahkan pendengar HTTPS.
7. Setelah selesai menambahkan semua informasi lain untuk membuat layanan, pilih Buat.

AWS CLI

Untuk mengonfigurasi nama domain khusus untuk layanan Anda

Gunakan perintah create-service.

```
aws vpc-lattice create-service --name service_name --custom-domain-
name your_custom_domain_name --type https --certificate-arn arn:aws:acm:us-
east-1:123456789012:certificate/12345678-1234-1234-1234-123456789012
```

Dalam perintah di atas, untuk`--name`, masukkan nama untuk layanan Anda. Untuk`--custom-domain-name`, masukkan nama domain layanan Anda seperti,`parking.example.com`. Untuk `--certificate-arn` masukkan ARN sertifikat Anda di ACM. Sertifikat ARN tersedia di akun Anda di. AWS Certificate Manager

Kaitkan nama domain khusus dengan layanan Anda

Pertama, jika Anda belum melakukannya, daftarkan nama domain khusus Anda. Internet Corporation for Assigned Names and Numbers (ICANN) mengelola nama domain di internet. Anda mendaftarkan nama domain menggunakan pencatat nama domain, organisasi terakreditasi ICANN yang mengelola registri nama domain. Situs web untuk registrar Anda akan memberikan petunjuk terperinci dan informasi harga untuk mendaftarkan nama domain Anda. Untuk informasi selengkapnya, lihat sumber daya berikut:

- Untuk menggunakan Amazon Route 53 untuk mendaftarkan nama domain, lihat Mendaftarkan nama domain menggunakan Route 53 di Panduan Pengembang Amazon Route 53.
- Untuk daftar pendaftar terakreditasi, lihat Direktori Panitera Terakreditasi.

Selanjutnya, gunakan layanan DNS Anda, seperti registrar domain Anda, untuk membuat catatan untuk merutekan kueri ke layanan Anda. Untuk informasi lebih lanjut, lihat dokumentasi untuk server DNS Anda. Atau, Anda dapat menggunakan Route 53 sebagai layanan DNS Anda.

Jika Anda menggunakan Route 53, Anda dapat menggunakan catatan alias atau catatan CNAME untuk merutekan kueri ke layanan Anda. Kami menyarankan Anda menggunakan catatan alias karena Anda dapat membuat catatan alias di simpul atas namespace DNS, juga dikenal sebagai puncak zona.

Jika Anda menggunakan Route 53, Anda harus terlebih dahulu membuat zona yang dihosting, yang berisi informasi tentang cara merutekan lalu lintas di internet untuk domain Anda. Setelah Anda membuat zona yang dihosting pribadi atau publik, buat catatan sedemikian rupa sehingga nama domain kustom Anda, misalnya`parking.example.com`, dipetakan ke nama domain yang dibuat

secara otomatis VPC Lattice, misalnya,. `my-service-02031c045478f6ddf1.7d67968.vpc-lattice-svcs.us-west-2.on.aws` Tanpa pemetaan ini, nama domain kustom Anda tidak akan berfungsi di VPC Lattice.

Prosedur berikut menunjukkan cara membuat zona yang dihosting pribadi atau publik menggunakan Route 53

Konsol Manajemen AWS

Untuk membuat catatan alias untuk merutekan kueri ke layanan Anda menggunakan Route 53, lihat Merutekan lalu lintas ke titik akhir domain layanan Amazon VPC Lattice.

Gunakan nama domain yang dihasilkan VPC Lattice untuk layanan Anda, misalnya **`my-service-02031c045478f6ddf1.7d67968.vpc-lattice-svcs.us-west-2.on.aws`** untuk Nilai. Anda dapat menemukan nama domain yang dibuat secara otomatis ini di konsol VPC Lattice di halaman layanan Anda.

AWS CLI

Untuk membuat catatan alias di zona yang dihosting

1. Dapatkan nama domain yang dihasilkan VPC Lattice untuk layanan Anda (misalnya,). `my-service-02031c045478f6ddf1.7d67968.vpc-lattice-svcs.us-west-2.on.aws`
2. Untuk mengatur alias, gunakan perintah berikut.

```
aws route53 change-resource-record-sets --hosted-zone-id your-hosted-zone-ID --
change-batch file://~/Desktop/change-set.json
```

Untuk `change-set.json` file tersebut, buat file JSON dengan konten dalam contoh JSON berikut, dan simpan di mesin lokal Anda. Ganti perintah *`file://~/Desktop/change-set.json`* di atas dengan jalur file JSON yang disimpan di mesin lokal Anda. Perhatikan bahwa “Ketik” di JSON berikut dapat berupa tipe catatan A atau AAAA.

```
{
    "Comment": "my-custom-domain-name.com alias",
    "Changes": [
        {
            "Action": "CREATE",
            "ResourceRecordSet": {
                "Name": "my-custom-domain-name.com",
                "Type": "alias-record-type",
```

```
                "AliasTarget": {
                    "HostedZoneId": "your-hosted-zone-ID",
                    "DNSName": "lattice-generated-domain-name",
                    "EvaluateTargetHealth": true
                }
            }
        }
    ]
}
```

Bawa Sertifikat Anda Sendiri (BYOC) untuk Kisi VPC

Untuk melayani permintaan HTTPS, Anda harus memiliki SSL/TLS sertifikat siap pakai AWS Certificate Manager (ACM) sendiri sebelum menyiapkan nama domain kustom. Sertifikat ini harus memiliki Nama Alternatif Subjek (SAN) atau Nama Umum (CN) yang cocok dengan nama domain khusus untuk layanan Anda. Jika SAN hadir, kami memeriksa kecocokan hanya di daftar SAN. Jika SAN tidak ada, kami memeriksa kecocokan di CN.

VPC Lattice melayani permintaan HTTPS menggunakan Server Name Indication (SNI). DNS merutekan permintaan HTTPS ke layanan VPC Lattice Anda berdasarkan nama domain kustom dan sertifikat yang cocok dengan nama domain ini. Untuk meminta SSL/TLS sertifikat nama domain di ACM atau mengimpornya ke ACM, lihat Menerbitkan dan Mengelola Sertifikat dan Mengimpor sertifikat di Panduan Pengguna.AWS Certificate Manager Jika Anda tidak dapat meminta atau mengimpor sertifikat Anda sendiri di ACM, gunakan nama domain dan sertifikat yang dihasilkan oleh VPC Lattice.

VPC Lattice hanya menerima satu sertifikat khusus per layanan. Namun, Anda dapat menggunakan sertifikat khusus untuk beberapa domain kustom. Ini berarti Anda dapat menggunakan sertifikat yang sama untuk semua layanan VPC Lattice yang Anda buat dengan nama domain kustom.

Untuk melihat sertifikat Anda menggunakan konsol ACM, buka Sertifikat, dan pilih ID sertifikat Anda. Anda akan melihat layanan VPC Lattice yang terkait dengan sertifikat tersebut di bawah sumber daya terkait.

Pertimbangan dan batasan

- VPC Lattice memungkinkan pencocokan wildcard yang sedalam satu level di Subject Alternate Name (SAN) atau Common Name (CN) dari sertifikat terkait. Misalnya, jika Anda membuat layanan dengan nama domain khusus `parking.example.com` dan mengaitkan sertifikat Anda sendiri

dengan SAN`*.example.com`. Saat permintaan masuk`parking.example.com`, VPC Lattice mencocokkan SAN dengan nama domain apa pun dengan domain apex. `example.com` Namun, jika Anda memiliki domain khusus `parking.different.example.com` dan sertifikat Anda memiliki SAN`*.example.com`, permintaan gagal.

- VPC Lattice mendukung satu tingkat kecocokan domain wildcard. Ini berarti bahwa wildcard hanya dapat digunakan sebagai subdomain tingkat pertama, dan hanya mengamankan satu tingkat subdomain. Misalnya, jika SAN sertifikat Anda`*.example.com`, maka `parking.*.example.com` tidak didukung.
- VPC Lattice mendukung satu wildcard per nama domain. Ini berarti `*.*.example.com` itu tidak valid. Untuk informasi selengkapnya, lihat Meminta sertifikat publik di Panduan AWS Certificate Manager Pengguna.
- VPC Lattice hanya mendukung sertifikat dengan kunci RSA 2048-bit.
- SSL/TLS Sertifikat di ACM harus berada di Wilayah yang sama dengan layanan VPC Lattice yang Anda kaitkan dengannya.

Mengamankan kunci pribadi sertifikat Anda

Saat Anda meminta SSL/TLS sertifikat menggunakan ACM, ACM menghasilkan public/private key pair. Saat Anda mengimpor sertifikat, Anda menghasilkan key pair. Kunci publik menjadi bagian dari sertifikat. Untuk menyimpan kunci pribadi dengan aman, ACM membuat kunci lain menggunakan AWS KMS, yang disebut kunci KMS, dengan alias aws/acm. AWS KMS menggunakan kunci ini untuk mengenkripsi kunci pribadi sertifikat Anda. Untuk informasi selengkapnya, lihat Perlindungan data AWS Certificate Manager di Panduan AWS Certificate Manager Pengguna.

VPC Lattice menggunakan AWS TLS Connection Manager, layanan yang hanya dapat diakses Layanan AWS, untuk mengamankan dan menggunakan kunci pribadi sertifikat Anda. Saat Anda menggunakan sertifikat ACM untuk membuat layanan VPC Lattice, VPC Lattice mengaitkan sertifikat Anda dengan TLS Connection Manager. AWS Kami melakukan ini dengan membuat hibah AWS KMS terhadap kunci AWS terkelola Anda. Hibah ini memungkinkan TLS Connection Manager digunakan AWS KMS untuk mendekripsi kunci pribadi sertifikat Anda. TLS Connection Manager menggunakan sertifikat dan kunci pribadi yang didekripsi (plaintext) untuk membuat koneksi aman (sesi SSL/TLS) dengan klien layanan VPC Lattice. Ketika sertifikat dipisahkan dari layanan VPC Lattice, hibah dihentikan. Untuk informasi selengkapnya, lihat Hibah di Panduan AWS Key Management Service Pengembang.

Untuk informasi selengkapnya, lihat Enkripsi saat diam.

Hapus layanan VPC Lattice

Untuk menghapus layanan VPC Lattice, Anda harus terlebih dahulu menghapus semua asosiasi yang mungkin dimiliki layanan dengan jaringan layanan apa pun. Jika Anda menghapus layanan, semua sumber daya yang terkait dengan layanan, seperti kebijakan sumber daya, kebijakan autentikasi, pendengar, aturan pendengar, dan langganan log akses, juga akan dihapus.

Untuk menghapus layanan menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pada halaman Layanan, pilih layanan yang ingin Anda hapus, lalu pilih Tindakan, Hapus layanan.
4. Saat diminta konfirmasi, pilih Hapus.

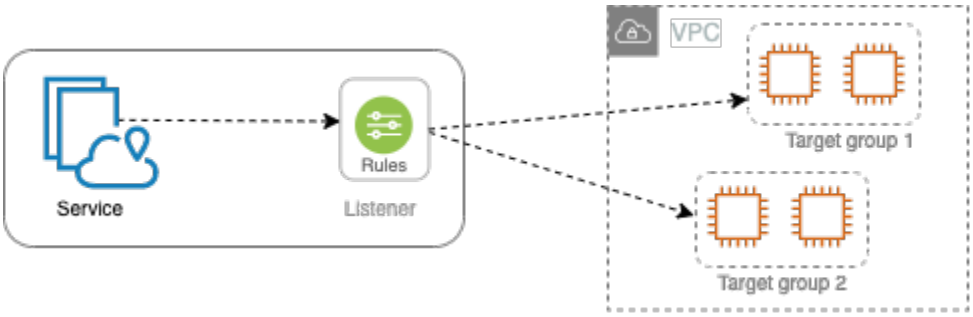
Untuk menghapus layanan menggunakan AWS CLI

Gunakan perintah delete-service.

Grup sasaran di VPC Lattice

Grup target VPC Lattice adalah kumpulan target, atau sumber daya komputasi, yang menjalankan aplikasi atau layanan Anda. Jenis target yang didukung meliputi instans EC2, alamat IP, fungsi Lambda, Application Load Balancer, tugas Amazon ECS, dan Kubernetes Pods. Anda juga dapat melampirkan layanan yang ada ke grup target Anda. Untuk informasi selengkapnya tentang penggunaan Kubernetes dengan VPC Lattice, lihat Panduan Pengguna Gateway API Controller.AWS

Setiapkelompok target terbiasa merutekan permintaan untuk satu atau lebih target terdaftar. Bila Anda membuat aturan listener, Anda menentukan grup target dan kondisi. Ketika kondisi aturan terpenuhi, lalu lintas diteruskan ke kelompok target yang sesuai. Anda dapat membuat kelompok-kelompok target yang berbeda untuk berbagai jenis permintaan. Misalnya, buat satu grup target untuk permintaan umum dan grup target lainnya untuk permintaan yang menyertakan kondisi aturan tertentu, seperti nilai jalur atau header.



Anda menentukan pengaturan pemeriksaan kesehatan untuk layanan Anda berdasarkan per kelompok target. Setiap kelompok target menggunakan pengaturan pemeriksaan kondisi yang sudah ada, kecuali jika Anda menimpa mereka saat Anda membuat kelompok target atau mengubahnya nanti. Setelah Anda menentukan grup target dalam aturan untuk pendengar, layanan akan terus memantau kesehatan semua target yang terdaftar dengan grup target. Rute layanan meminta ke target terdaftar yang sehat.

Untuk menentukan grup target dalam aturan untuk pendengar layanan, grup target harus berada di akun yang sama dengan layanan.

Kelompok target VPC Lattice mirip dengan kelompok target yang disediakan oleh Elastic Load Balancing, tetapi mereka tidak dapat dipertukarkan.

Daftar Isi

- Daftarkan target dengan grup target VPC Lattice
- Pemeriksaan kesehatan untuk grup target VPC Lattice Anda
- Konfigurasi perutean
- Algoritma perutean
- Tipe target
- Jenis alamat IP
- Target HTTP di VPC Lattice
- Lambda berfungsi sebagai target di VPC Lattice
- Aplikasi Load Balancer sebagai target di VPC Lattice
- Versi protokol
- Tag untuk grup target VPC Lattice
- Menghapus grup target VPC Lattice

Buat grup target VPC Lattice

Anda mendaftarkan target Anda dengan grup target. Secara default, layanan VPC Lattice mengirimkan permintaan ke target terdaftar menggunakan port dan protokol yang Anda tentukan untuk grup target. Anda dapat mengganti port ini ketika Anda mendaftar setiap target dengan kelompok target.

Untuk merutekan lalu lintas ke target dalam kelompok target, tentukan kelompok target dalam suatu tindakan saat Anda membuat pendengar atau membuat aturan untuk pendengar Anda. Untuk informasi selengkapnya, lihat Aturan pendengar untuk layanan VPC Lattice. Anda dapat menentukan grup target yang sama di beberapa pendengar, tetapi pendengar ini harus milik layanan yang sama. Untuk menggunakan grup target dengan layanan, Anda harus memverifikasi bahwa grup target tidak digunakan oleh pendengar untuk layanan lain.

Anda dapat menambah atau menghapus target dari grup target Anda kapan saja. Untuk informasi selengkapnya, lihat Daftarkan target dengan grup target VPC Lattice. Anda juga dapat mengubah pengaturan pemeriksaan kesehatan untuk grup target Anda. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan untuk grup target VPC Lattice Anda.

Buat grup target

Anda dapat membuat grup target dan secara opsional mendaftarkan target sebagai berikut.

Untuk membuat grup target menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. PilihBuat grup target.
4. Untuk Pilih jenis target, lakukan salah satu hal berikut:
   - Pilih Instans untuk mendaftarkan target berdasarkan ID instans.
   - Pilih alamat IP untuk mendaftarkan target berdasarkan alamat IP.
   - Pilih fungsi Lambda untuk mendaftarkan fungsi Lambda sebagai target.
   - Pilih Application Load Balancer untuk mendaftarkan Application Load Balancer sebagai target.
5. Untuk Name, masukkan nama untuk grup target. Nama ini harus unik untuk akun Anda di setiap AWS Wilayah, dapat memiliki maksimal 32 karakter, harus hanya berisi karakter alfanumerik atau tanda hubung, dan tidak boleh dimulai atau diakhiri dengan tanda hubung.
6. Untuk Protokol dan Port, Anda dapat memodifikasi nilai default sesuai kebutuhan. Protokol defaultnya adalah HTTPS dan port defaultnya adalah 443.

   Jika jenis target adalah fungsi Lambda, Anda tidak dapat menentukan protokol atau port.
7. Untuk jenis alamat IP, pilih IPv4untuk mendaftarkan target dengan IPv4 alamat atau memilih IPv6untuk mendaftarkan target dengan IPv6 alamat. Anda tidak dapat mengubah pengaturan ini setelah grup target dibuat.

   Opsi ini hanya tersedia jika jenis targetnya adalah alamat IP.
8. UntukVPC, pilih Virtual Private Cloud (VPC).

   Opsi ini tidak tersedia jika jenis target adalah fungsi Lambda.
9. Untuk versi Protokol, ubah nilai default sesuai kebutuhan. Nilai default-nya HTTP1.

   Opsi ini tidak tersedia jika jenis target adalah fungsi Lambda.
10. Untuk pemeriksaan Kesehatan, ubah pengaturan default sesuai kebutuhan. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan untuk grup target VPC Lattice Anda.

    Pemeriksaan kesehatan tidak tersedia jika jenis targetnya adalah fungsi Lambda.
11. Untuk versi struktur acara Lambda, pilih versi. Untuk informasi selengkapnya, lihat the section called “Menerima acara dari layanan VPC Lattice”.

Opsi ini hanya tersedia jika jenis target adalah fungsi Lambda

12. (Opsional) Untuk menambahkan tag, memperluas Tag, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
13. Pilih Berikutnya.
14. Untuk Daftar target, Anda dapat melewati langkah ini atau menambahkan target sebagai berikut:
    - Jika jenis targetnya adalah Instans, pilih instance, masukkan port, lalu pilih Sertakan sebagai tertunda di bawah ini.
    - Jika jenis targetnya adalah alamat IP, lakukan hal berikut:
        a. Untuk Pilih jaringan, simpan VPC yang Anda pilih untuk grup target atau pilih Alamat IP pribadi lainnya.
        b. Untuk Tentukan IPs dan tentukan port, masukkan alamat IP dan masukkan port. Port default adalah port grup target.
        c. Pilih Sertakan sebagai tertunda di bawah ini.
    - Jika jenis target adalah fungsi Lambda, pilih fungsi Lambda. Untuk membuat fungsi Lambda, pilih Buat fungsi Lambda baru.
    - Jika jenis target adalah Application Load Balancer, pilih Application Load Balancer. Untuk membuat Application Load Balancer, pilih buat Application Load Balancer.
15. PilihBuat grup target.

    Mungkin perlu beberapa menit bagi VPC Lattice untuk mendaftarkan target. Untuk informasi lebih lanjut lihat, Mengapa butuh waktu lama untuk perubahan DNS saya menyebar di Route 53 dan resolver publik?

Untuk membuat grup target menggunakan AWS CLI

Gunakan create-target-groupperintah untuk membuat grup target dan perintah register-target untuk menambahkan target.

Subnet bersama

Peserta dapat membuat grup target VPC Lattice dalam VPC bersama. Aturan berikut berlaku untuk subnet bersama:

- Semua bagian dari layanan VPC Lattice, seperti pendengar, grup target, dan target, harus dibuat oleh akun yang sama. Mereka dapat dibuat dalam subnet yang dimiliki oleh atau dibagikan dengan pemilik layanan VPC Lattice.
- Target yang terdaftar dengan grup target harus dibuat oleh akun yang sama dengan kelompok sasaran.
- Hanya pemilik VPC yang dapat mengaitkan VPC dengan jaringan layanan. Sumber daya peserta dalam VPC bersama yang terkait dengan jaringan layanan dapat mengirim permintaan ke layanan yang terkait dengan jaringan layanan. Namun, administrator dapat mencegah hal ini dengan menggunakan grup keamanan, jaringan ACLs, atau kebijakan autentikasi.

Untuk informasi selengkapnya tentang sumber daya yang dapat dibagikan untuk VPC Lattice, lihat. Bagikan entitas VPC Lattice

Daftarkan target dengan grup target VPC Lattice

Layanan Anda berfungsi sebagai titik kontak tunggal untuk klien dan mendistribusikan lalu lintas masuk ke seluruh target terdaftar yang sehat. Anda dapat mendaftarkan setiap target dengan satu atau lebih kelompok target.

Jika permintaan pada aplikasi Anda meningkat, Anda dapat mendaftarkan target tambahan dengan satu atau lebih kelompok sasaran untuk menangani permintaan. Layanan mulai merutekan permintaan ke target yang baru terdaftar segera setelah proses pendaftaran selesai dan target melewati pemeriksaan kesehatan awal.

Jika permintaan pada aplikasi Anda menurun, atau Anda perlu untuk melayani target Anda, Anda dapat membatalkan pendaftaran (deregistrasi) target dari kelompok target Anda. Proses deregisterasi target menghapus itu dari kelompok target Anda, tetapi tidak mempengaruhi target sebaliknya. Layanan berhenti merutekan permintaan ke target segera setelah dideregistrasi. Target memasuki keadaan`DRAINING` hingga permintaan dalam penerbangan telah selesai. Anda dapat mendaftarkan target dengan kelompok target lagi ketika target Anda siap untuk untuk melanjutkan menerima permintaan.

Jenis target grup target Anda menentukan bagaimana Anda mendaftarkan target dengan kelompok target tersebut. Untuk informasi selengkapnya, lihat Tipe target.

Gunakan prosedur konsol berikut untuk mendaftarkan atau membatalkan pendaftaran target. Atau, gunakan perintah register-target dan deregister-target dari. AWS CLI

Daftar Isi

Register atau target deregister berdasarkan ID instance

Instance target harus berada di virtual private cloud (VPC) yang Anda tentukan untuk grup target. Contoh juga harus dalam keadaan`running` saat Anda mendaftarkannya.

Saat mendaftarkan target berdasarkan ID instans, Anda dapat menggunakan layanan Anda dengan grup Auto Scaling. Setelah Anda melampirkan grup target ke grup Auto Scaling dan grup skala keluar, instance yang diluncurkan grup Auto Scaling secara otomatis terdaftar dengan grup target. Jika Anda memisahkan grup target dari grup Auto Scaling, maka instans tersebut secara otomatis dihapus dari grup target. Untuk informasi selengkapnya, lihat Merutekan lalu lintas ke grup Auto Scaling Anda dengan grup target VPC Lattice di Panduan Pengguna Amazon EC2 Auto Scaling.

Untuk mendaftarkan atau membatalkan pendaftaran target berdasarkan ID instans menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan detailnya.
4. Pilih tabTarget.
5. Untuk mendaftarkan contoh, pilihTarget daftar. Pilih instance, masukkan port instance, lalu pilih Sertakan sebagai tertunda di bawah ini. Setelah selesai menambahkan instance, pilih Daftarkan target.
6. Untuk membatalkan pendaftaran instance, pilih instance, lalu pilih Deregister.

Mendaftar atau membatalkan pendaftaran target berdasarkan alamat IP

Alamat IP target harus dari subnet VPC yang Anda tentukan untuk grup target. Anda tidak dapat mendaftarkan alamat IP layanan lain di VPC yang sama. Anda tidak dapat mendaftarkan titik akhir VPC atau alamat IP yang dapat dirutekan secara publik.

Untuk mendaftarkan atau membatalkan pendaftaran target berdasarkan alamat IP menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan detailnya.
4. Pilih tabTarget.
5. Untuk mendaftarkan alamat IP, pilihTarget daftar. Untuk setiap alamat IP, pilih rangkaian, masukkan alamat IP dan port, dan pilihSertakan sebagai tertunda di bawah ini. Setelah selesai menentukan alamat, pilih Daftarkan target.
6. Untuk membatalkan pendaftaran alamat IP, pilih alamat IP, lalu pilihDeregister.

Mendaftar atau membatalkan pendaftaran fungsi Lambda

Anda dapat mendaftarkan satu fungsi Lambda dengan grup target. Jika Anda tidak perlu lagi mengirim lalu lintas ke fungsi Lambda Anda, Anda dapat membatalkan pendaftarannya. Setelah Anda membatalkan pendaftaran fungsi Lambda, permintaan dalam penerbangan gagal dengan galat HTTP 5XX. Lebih baik membuat grup target baru daripada mengganti fungsi Lambda untuk grup target.

Untuk mendaftarkan atau membatalkan pendaftaran fungsi Lambda menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan detailnya.
4. Pilih tabTarget.
5. Jika tidak ada fungsi Lambda yang terdaftar, pilih Daftarkan target. Pilih fungsi Lambda dan pilih Daftarkan target.
6. Untuk membatalkan pendaftaran fungsi Lambda, pilihDeregister. Saat diminta konfirmasi, masukkan **`confirm`** lalu pilih Deregister.

Mendaftarkan atau membatalkan pendaftaran Application Load Balancer

Anda dapat mendaftarkan Application Load Balancer tunggal dengan masing-masing grup target. Jika Anda tidak perlu lagi mengirim lalu lintas ke penyeimbang beban Anda, Anda dapat

membatalkan pendaftarannya. Setelah Anda membatalkan pendaftaran penyeimbang beban, permintaan dalam penerbangan gagal dengan kesalahan HTTP 5XX. Lebih baik membuat grup target baru daripada mengganti Application Load Balancer untuk grup target.

Untuk mendaftarkan atau membatalkan pendaftaran Application Load Balancer menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan detailnya.
4. Pilih tabTarget.
5. Jika tidak ada Application Load Balancer yang terdaftar, pilih Register target. Pilih Application Load Balancer dan pilih Register target.
6. Untuk membatalkan pendaftaran Application Load Balancer, pilih Deregister. Saat diminta konfirmasi, masukkan **`confirm`** lalu pilih Deregister.

Pemeriksaan kesehatan untuk grup target VPC Lattice Anda

Layanan Anda secara berkala mengirimkan permintaan ke target yang terdaftar untuk menguji statusnya. Uji ini disebut pemeriksaan kondisi.

Setiap rute layanan VPC Lattice hanya meminta target yang sehat. Setiap layanan memeriksa kesehatan setiap target, menggunakan pengaturan pemeriksaan kesehatan untuk kelompok sasaran yang dengannya target terdaftar. Setelah target Anda terdaftar, target itu harus lulus satu pemeriksaan kondisi agar dapat dianggap sehat. Setelah setiap pemeriksaan kesehatan selesai, layanan menutup koneksi yang dibuat untuk pemeriksaan kesehatan.

Keterbatasan dan pertimbangan

- Ketika versi protokol grup target HTTP1, pemeriksaan kesehatan diaktifkan secara default.
- Ketika versi protokol grup target HTTP2, pemeriksaan kesehatan tidak diaktifkan secara default. Namun, Anda dapat mengaktifkan pemeriksaan kesehatan, dan secara manual mengatur versi protokol ke HTTP1 atau HTTP2.
- Pemeriksaan Kesehatan tidak mendukung versi protokol grup target gRPC. Namun, jika Anda mengaktifkan pemeriksaan kesehatan, Anda harus menentukan versi protokol pemeriksaan kesehatan sebagai HTTP1 atau HTTP2.
- Pemeriksaan Kesehatan tidak mendukung kelompok sasaran Lambda.

- Pemeriksaan Kesehatan tidak mendukung kelompok sasaran Application Load Balancer. Namun, Anda dapat mengaktifkan pemeriksaan kesehatan untuk target Application Load Balancer Anda menggunakan Elastic Load Balancing. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan grup target di Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

Pengaturan pemeriksaan kondisi

Anda mengonfigurasi pemeriksaan kondisi untuk target dalam grup target seperti yang dijelaskan dalam tabel berikut. Nama pengaturan yang digunakan dalam tabel adalah nama yang digunakan dalam API. Layanan mengirimkan permintaan pemeriksaan kesehatan ke setiap target yang terdaftar setiap HealthCheckIntervalSecondsdetik, menggunakan port, protokol, dan jalur ping yang ditentukan. Setiap permintaan pemeriksaan kondisi bersifat independen dan hasilnya berlaku selama seluruh interval. Waktu yang dibutuhkan untuk target untuk merespons tidak memengaruhi interval untuk permintaan pemeriksaan kondisi berikutnya. Jika pemeriksaan kesehatan melebihi kegagalan UnhealthyThresholdCountberturut-turut, layanan menghilangkan target dari layanan. Ketika pemeriksaan kesehatan melebihi keberhasilan HealthyThresholdCountberturut-turut, layanan menempatkan target kembali dalam layanan.

Pengaturan	Deskripsi
HealthCheckProtocol	Protokol yang digunakan layanan saat melakukan pemeriksaan kesehatan pada target. Protokol yang mungkin adalah HTTP dan HTTPS. Defaultnya adalah protokol HTTP.
HealthCheckPort	Port yang digunakan layanan saat melakukan pemeriksaan kesehatan pada target. Defaultny a adalah menggunakan port di mana setiap target menerima lalu lintas dari layanan.
HealthCheckPath	Tujuan pemeriksaan kondisi pada target. Jika versi protokol adalah HTTP1 atau HTTP2, tentukan URI (/path yang valid? kueri). Default-nya adalah /.
HealthCheckTimeoutSeconds	Jumlah waktu, dalam detik, selama tidak ada respons dari target berarti pemeriksa

Pengaturan	Deskripsi
	an kondisi gagal. Kisarannya adalah 1—120 detik. Defaultnya adalah 5 detik jika tipe targetnya adalah `INSTANCE` atau`IP`. Tentukan 0 untuk mengatur ulang pengaturan ini ke nilai defaultnya.
HealthCheckIntervalSeconds	Perkiraan jumlah waktu, dalam hitungan detik, antara pemeriksaan kondisi dari target individu. Rentangnya adalah 5-300 detik. Defaultny a adalah 30 detik jika tipe targetnya adalah `INSTANCE` atau`IP`. Tentukan 0 untuk mengatur ulang pengaturan ini ke nilai defaultnya.
HealthyThresholdCount	Jumlah pemeriksaan kesehatan yang berhasil berturut-turut yang diperlukan sebelum target yang tidak sehat dianggap sehat. Rentangny a adalah 2–10. Defaultnya adalah 5. Tentukan 0 untuk mengatur ulang pengaturan ini ke nilai defaultnya.
UnhealthyThresholdCount	Jumlah pemeriksaan kondisi yang gagal berturut-turut diperlukan sebelum mengangga p target tidak sehat. Rentangnya adalah 2–10. Defaultnya adalah 2. Tentukan 0 untuk mengatur ulang pengaturan ini ke nilai defaultnya.

Pengaturan	Deskripsi
Pencocokan	Kode yang digunakan saat memeriksa respons yang berhasil dari target. Ini disebut Kode berhasil pada konsol. Jika versi protokol adalah HTTP1 atau HTTP2, nilai yang mungkin adalah dari 200 hingga 499. Anda dapat menentukan beberapa nilai (misalnya, “200,202") atau rentang nilai (misalnya, “200-299"). Nilai default adalah 200. Versi protokol pemeriksaan kesehatan untuk gRPC saat ini tidak didukung. Namun, jika versi protokol grup target Anda adalah gRPC, Anda dapat menentukan HTTP1 atau versi HTTP2 protokol dalam konfigurasi pemeriksaan kesehatan Anda.

Periksa kondisi target Anda

Anda dapat memeriksa status kondisi target yang terdaftar dengan kelompok target Anda.

Untuk memeriksa kesehatan target Anda menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan laman detailnya.
4. Pada tab Target, kolom Status Kesehatan menunjukkan status setiap target. Jika status adalah nilai selain`Healthy`, kolom Detail status Kesehatan berisi informasi lebih lanjut.

Untuk memeriksa kesehatan target Anda menggunakan AWS CLI

Gunakan perintah daftar-target. Keluaran dari perintah ini berisi status kondisi target. Jika status adalah nilai selain `Healthy`, output juga termasuk kode alasan.

Untuk menerima pemberitahuan email tentang target yang tidak sehat

Gunakan CloudWatch alarm untuk memulai fungsi Lambda untuk mengirim detail tentang target yang tidak sehat.

Ubah pengaturan pemeriksaan kesehatan

Anda dapat mengubah pengaturan pemeriksaan kondisi untuk grup target kapan saja.

Untuk mengubah pengaturan pemeriksaan kesehatan menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan laman detailnya.
4. Pada tab Pemeriksaan Kesehatan, di bagian Pengaturan pemeriksaan Kesehatan, pilih Edit.
5. Ubah pengaturan pemeriksaan kesehatan sesuai kebutuhan.
6. Pilih Simpan perubahan.

Untuk mengubah pengaturan pemeriksaan kesehatan menggunakan AWS CLI

Gunakan perintah update-target-group.

Konfigurasi perutean

Secara default, layanan merutekan permintaan ke targetnya menggunakan protokol dan nomor port yang Anda tentukan saat Anda membuat grup target. Atau, Anda dapat mengganti port yang digunakan untuk merutekan lalu lintas ke target saat Anda mendaftarkannya dengan grup target.

Kelompok target mendukung protokol dan port berikut ini:

- Protokol: HTTP, HTTPS, TCP
- Port: 1-65535

Jika grup target dikonfigurasi dengan protokol HTTPS atau menggunakan pemeriksaan kesehatan HTTPS, koneksi TLS ke target menggunakan kebijakan keamanan dari pendengar. VPC Lattice membuat koneksi TLS dengan target menggunakan sertifikat yang Anda instal pada target. VPC Lattice tidak memvalidasi sertifikat ini. Oleh karena itu, Anda dapat menggunakan sertifikat ditandatangani sendiri atau sertifikat yang telah kedaluwarsa. Lalu lintas antara VPC Lattice dan

target diautentikasi pada tingkat paket, sehingga tidak berisiko man-in-the-middle serangan atau spoofing bahkan jika sertifikat pada target tidak valid.

Grup target TCP hanya didukung dengan pendengar TLS.

Algoritma perutean

Secara default, algoritma routing round robin digunakan untuk merutekan permintaan ke target yang sehat.

Ketika layanan VPC Lattice menerima permintaan, ia menggunakan proses berikut:

1. Mengevaluasi aturan pendengar dalam rangka prioritas untuk menentukan aturan yang diterapkan.
2. Memilih target dari kelompok target untuk tindakan aturan, menggunakan algoritma round robin default. Routing dilakukan secara mandiri untuk setiap grup target, bahkan ketika target telah terdaftar dengan beberapa kelompok target.

Jika kelompok sasaran hanya berisi target yang tidak sehat, permintaan diarahkan ke semua target, terlepas dari status kesehatan mereka. Ini berarti bahwa jika semua target gagal pemeriksaan kesehatan pada saat yang sama, layanan VPC Lattice gagal dibuka. Efek dari fail-open adalah untuk memungkinkan lalu lintas ke semua target, terlepas dari status kesehatan mereka, berdasarkan algoritma round robin.

VPC Lattice mendukung afinitas Availability Zone (AZ) untuk merutekan lalu lintas. Ketika klien mengirim permintaan ke VPC Lattice, VPC Lattice merespons dengan alamat IP untuk layanan atau sumber daya dari AZ yang sama dengan klien. Jika AZ itu tidak tersedia, VPC Lattice merespons dengan alamat IP dari yang lain. AZs Dari VPC Lattice ke target, routing adalah ke target, yang mungkin didistribusikan ke seluruh. AZs Selain itu, tidak ada biaya transfer data antar-AZ di VPC Lattice.

Tipe target

Bila Anda membuat grup target, Anda menentukan jenis targetnya, yang menentukan jenis target yang Anda tentukan saat mendaftarkan target dengan grup target ini. Setelah Anda membuat grup target, Anda tidak dapat mengubah jenis targetnya.

Status yang mungkin muncul adalah sebagai berikut:

`INSTANCE`

Target ditentukan oleh contoh ID.

`IP`

Targetnya adalah alamat IP.

`LAMBDA`

Targetnya adalah fungsi Lambda.

`ALB`

Targetnya adalah Application Load Balancer.

Pertimbangan-pertimbangan

- Ketika jenis targetnya`IP`, Anda harus menentukan alamat IP dari subnet VPC untuk grup target. Jika Anda perlu mendaftarkan alamat IP dari luar VPC ini, buat kelompok target tipe `ALB` dan daftarkan alamat IP dengan Application Load Balancer.
- Jika jenis targetnya`IP`, Anda tidak dapat mendaftarkan titik akhir VPC atau alamat IP yang dapat dirutekan secara publik.
- Ketika jenis targetnya`LAMBDA`, Anda dapat mendaftarkan satu fungsi Lambda. Ketika layanan menerima permintaan untuk fungsi Lambda, ia memanggil fungsi Lambda. Jika Anda ingin mendaftarkan beberapa fungsi lambda ke layanan, Anda perlu menggunakan beberapa grup target.
- Ketika jenis targetnya adalah`ALB`, Anda dapat mendaftarkan Application Load Balancer internal tunggal sebagai target hingga dua layanan VPC Lattice. Untuk melakukan ini, daftarkan Application Load Balancer dengan dua kelompok target terpisah, yang digunakan oleh dua layanan VPC Lattice yang berbeda. Selain itu, Application Load Balancer yang ditargetkan harus memiliki setidaknya satu pendengar yang portnya cocok dengan port grup target.
- Anda dapat secara otomatis mendaftarkan tugas ECS Anda dengan grup target VPC Lattice saat peluncuran. Kelompok sasaran harus memiliki jenis target`IP`. Untuk informasi selengkapnya, lihat Menggunakan Kisi VPC dengan layanan Amazon ECS Anda di Panduan Pengembang Layanan Kontainer Elastis Amazon.

  Atau, daftarkan Application Load Balancer untuk layanan Amazon ECS Anda dengan jenis grup target VPC Lattice. `ALB` Untuk informasi selengkapnya, lihat Menggunakan load balancing untuk

mendistribusikan lalu lintas layanan Amazon ECS di Panduan Pengembang Layanan Amazon Elastic Container.

- Untuk mendaftarkan pod EKS sebagai target, gunakan AWS Gateway API Controller, yang mendapatkan alamat IP dari layanan Kubernetes.
- Jika protokol grup target adalah TCP, satu-satunya jenis target yang didukung adalah`INSTANCE`,`IP`, atau`ALB`.

Jenis alamat IP

Saat Anda membuat grup target dengan tipe target`IP`, Anda dapat menentukan jenis alamat IP untuk grup target. Ini menentukan jenis alamat apa yang digunakan penyeimbang beban untuk mengirim permintaan dan pemeriksaan kesehatan ke target. Nilai yang mungkin adalah `IPv4` dan `IPv6`. Nilai default-nya `IPV4`.

Pertimbangan-pertimbangan

- Jika Anda membuat grup target dengan jenis alamat IP`IPv6`, VPC yang Anda tentukan untuk grup target harus memiliki rentang IPv6 alamat.
- Alamat IP yang Anda daftarkan dengan grup target harus sesuai dengan jenis alamat IP dari grup target. Misalnya, Anda tidak dapat mendaftarkan IPv6 alamat dengan grup target jika jenis alamat IP-nya`IPv4`.
- Alamat IP yang Anda daftarkan dengan grup target harus berada dalam kisaran alamat IP VPC yang Anda tentukan untuk grup target.

Target HTTP di VPC Lattice

Permintaan HTTP dan respons HTTP menggunakan bidang header untuk mengirim informasi tentang pesan HTTP. Header HTTP ditambahkan secara otomatis. Bidang header adalah pasangan nama-nilai yang dipisahkan titik dua yang dipisahkan oleh carriage return (CR) dan line feed (LF). Satu set standar bidang header HTTP didefinisikan dalam RFC 2616, Header Pesan. Ada juga header HTTP non-standar yang tersedia secara otomatis ditambahkan dan digunakan secara luas oleh aplikasi. Misalnya, ada header HTTP non-standar dengan awalan. `x-forwarded`

x-forwardedheader

Amazon VPC Lattice menambahkan header berikut: `x-forwarded`

`x-forwarded-for`

Alamat IP sumber.

`x-forwarded-port`

Port tujuan.

`x-forwarded-proto`

Protokol koneksi (`http|https`).

Header identitas pemanggil

Amazon VPC Lattice menambahkan header identitas pemanggil berikut:

`x-amzn-lattice-identity`

Informasi identitas. Bidang berikut hadir jika AWS otentikasi berhasil.

- `Principal`— Prinsipal yang diautentikasi.
- `PrincipalOrgID`— ID organisasi untuk prinsipal yang diautentikasi.
- `PrincipalOrgPath`— Jalur organisasi untuk prinsipal yang diautentikasi.
- `SessionName`— Nama sesi yang diautentikasi.

Bidang berikut hadir jika kredensyal Peran Di Mana Saja digunakan dan otentikasi berhasil.

- `X509Issuer/OU`— Penerbit (OU).
- `X509SAN/DNS`— Nama alternatif subjek (DNS).
- `X509SAN/NameCN`— Nama alternatif penerbit (nama/CN).
- `X509SAN/URI`— Nama alternatif subjek (URI).
- `X509Subject/CN`Nama Subjek (CN)

`x-amzn-lattice-identity-tags`

ID utama dan tag utama apa pun. Formatnya adalah sebagai berikut.

```
principal=principal;principalorgid=orgid;principalorgpath=orgpath;principal-
tag1=value1; ...;principal-tag99=value99
```

VPC Lattice lolos dari titik koma (;) dalam nilai dengan garis miring terbalik (\).

`x-amzn-lattice-network`

VPC. Formatnya adalah sebagai berikut.

```
SourceVpcArn=arn:aws:ec2:region:account:vpc/id
```

`x-amzn-lattice-target`

Target. Formatnya adalah sebagai berikut.

```
ServiceArn=arn;ServiceNetworkArn=arn;TargetGroupArn=arn
```

Untuk informasi tentang sumber daya ARNs untuk Kisi VPC, lihat Jenis sumber daya yang ditentukan oleh Amazon VPC Lattice.

Header identitas penelepon tidak dapat dipalsukan. VPC Lattice menghapus header ini dari permintaan yang masuk. Header identitas ini mengekspresikan peta yang mendukung nilai kosong menggunakan format berikut. Saat mengurai, Anda tidak harus bergantung pada urutan spesifik dari header ini, Anda harus berharap bahwa new KEYs dapat ditambahkan kapan saja dan Anda harus siap untuk menangani nilai kosong. KEYs

Formatnya adalah sebagai berikut.

```
key-0=value-0;key-1=value-1;....;key-n=value-n;
```

Lambda berfungsi sebagai target di VPC Lattice

Anda dapat mendaftarkan fungsi Lambda sebagai target dengan grup target VPC Lattice, dan mengonfigurasi aturan listener untuk meneruskan permintaan ke grup target untuk fungsi Lambda Anda. Ketika layanan meneruskan permintaan ke grup target dengan fungsi Lambda sebagai target, layanan akan memanggil fungsi Lambda Anda dan meneruskan konten permintaan ke fungsi Lambda, dalam format JSON.

Batasan

- Fungsi Lambda dan kelompok target harus dalam akun dan di wilayah yang sama.
- Ukuran maksimum badan permintaan yang dapat Anda kirim ke fungsi Lambda adalah 6 MB.

- Ukuran maksimum respons JSON yang dapat dikirim oleh fungsi Lambda adalah 6 MB.
- Protokol harus HTTP atau HTTPS.

Siapkan fungsi Lambda

Rekomendasi berikut berlaku jika Anda menggunakan fungsi Lambda Anda dengan layanan VPC Lattice.

Izin untuk mengaktifkan fungsi Lambda

Saat Anda membuat grup target dan mendaftarkan fungsi Lambda menggunakan Konsol Manajemen AWS atau, AWS CLI VPC Lattice menambahkan izin yang diperlukan ke kebijakan fungsi Lambda Anda atas nama Anda.

Anda juga dapat menambahkan izin sendiri menggunakan panggilan API berikut:

```
aws lambda add-permission \
  --function-name lambda-function-arn-with-alias-name \
  --statement-id vpc-lattice \
  --principal vpc-lattice.amazonaws.com \
  --action lambda:InvokeFunction \
  --source-arn target-group-arn
```

Versioning fungsi Lambda

Anda dapat mendaftarkan satu fungsi Lambda per kelompok target. Untuk memastikan bahwa Anda dapat mengubah fungsi Lambda Anda dan bahwa layanan VPC Lattice selalu memanggil versi fungsi Lambda saat ini, buat alias fungsi dan sertakan alias dalam fungsi ARN saat Anda mendaftarkan fungsi Lambda dengan layanan VPC Lattice. Untuk informasi selengkapnya, lihat Versi fungsi Lambda dan Membuat alias untuk fungsi Lambda di Panduan Pengembang.AWS Lambda

Buat grup target untuk fungsi Lambda

Buat grup target, yang digunakan dalam routing permintaan. Jika konten permintaan cocok dengan aturan listener dengan tindakan untuk meneruskannya ke grup target ini, layanan VPC Lattice akan memanggil fungsi Lambda terdaftar.

Untuk membuat grup target dan mendaftarkan fungsi Lambda menggunakan konsol

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

2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. PilihBuat grup target.
4. UntukPilih jenis targetPilihFungsi Lambda.
5. Untuk Name, masukkan nama untuk grup target.
6. Untuk versi struktur acara Lambda, pilih versi. Untuk informasi selengkapnya, lihat the section called “Menerima acara dari layanan VPC Lattice”.
7. (Opsional) Untuk menambahkan tag, memperluas Tag, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
8. Pilih Berikutnya.
9. UntukFungsi LambdaLakukan salah satu langkah berikut:
   - Pilih fungsi Lambda yang ada.
   - Buat fungsi Lambda baru dan pilih.
   - Daftarkan fungsi Lambda nanti.
10. PilihBuat grup target.

Untuk membuat grup target dan mendaftarkan fungsi Lambda menggunakan AWS CLI

Gunakan perintah create-target-groupdan daftar-target.

Menerima acara dari layanan VPC Lattice

Layanan VPC Lattice mendukung pemanggilan Lambda untuk permintaan melalui HTTP dan HTTPS. Layanan mengirimkan acara dalam format JSON, dan menambahkan `X-Forwarded-For` header ke setiap permintaan.

Enkode Base64

Layanan Base64 mengkodekan badan jika `content-encoding` header ada dan jenis konten bukan salah satu dari yang berikut:

- `text/*`
- `application/json`
- `application/xml`
- `application/javascript`

Jika`content-encoding`header tidak hadir, encoding Base64 tergantung pada jenis konten. Untuk jenis konten di atas, layanan mengirimkan badan apa adanya, tanpa pengkodean Base64.

Format struktur acara

Saat membuat atau memperbarui jenis grup target`LAMBDA`, Anda dapat menentukan versi struktur acara yang diterima fungsi Lambda Anda. Versi yang mungkin adalah `V1` dan`V2`.

Example Contoh acara: V2

```
{
    "version": "2.0",
    "path": "/?query1=value1&query2=value2",
    "method": "GET|POST|HEAD|...",
    "headers": {
        "header-key": ["header-value", ...],
        ...
    },
    "queryStringParameters": {
        "key": ["value", ...]
    },
    "body": "request-body",
    "isBase64Encoded": true|false,
    "requestContext": {
        "serviceNetworkArn": "arn:aws:vpc-
lattice:region:123456789012:servicenetwork/sn-0bf3f2882e9cc805a",
        "serviceArn": "arn:aws:vpc-
lattice:region:123456789012:service/svc-0a40eebed65f8d69c",
        "targetGroupArn": "arn:aws:vpc-
lattice:region:123456789012:targetgroup/tg-6d0ecf831eec9f09",
        "identity": {
            "sourceVpcArn":
 "arn:aws:ec2:region:123456789012:vpc/vpc-0b8276c84697e7339",
            "type": "AWS_IAM",
            "principal": "arn:aws:iam::123456789012:assumed-role/my-role/my-session",
            "principalOrgID": "o-50dc6c495c0c9188",
            "sessionName": "i-0c7de02a688bde9f7",
            "x509IssuerOu": "string",
            "x509SanDns": "string",
            "x509SanNameCn": "string",
            "x509SanUri": "string",
            "x509SubjectCn": "string"
        },
        "region": "region",
```

```
        "timeEpoch": "1690497599177430"
    }
}
```

`body`

Tubuh permintaan. Hadir hanya jika protokolnya HTTP, HTTPS, atau gRPC.

`headers`

Header HTTP dari permintaan. Hadir hanya jika protokolnya HTTP, HTTPS, atau gRPC.

`identity`

Informasi identitas. Berikut ini adalah bidang yang mungkin.

- `principal`— Prinsipal yang diautentikasi. Hadir hanya jika AWS otentikasi berhasil.
- `principalOrgID`— ID organisasi untuk prinsipal yang diautentikasi. Hadir hanya jika AWS otentikasi berhasil.
- `sessionName`— Nama sesi yang diautentikasi. Hadir hanya jika AWS otentikasi berhasil.
- `sourceVpcArn`— ARN dari VPC tempat permintaan berasal. Hadir hanya jika sumber VPC dapat diidentifikasi.
- `type`— Nilainya adalah `AWS_IAM` jika kebijakan autentikasi digunakan dan AWS otentikasi berhasil.

Jika kredensyal Roles Anywhere digunakan dan otentikasi berhasil, berikut ini adalah bidang yang memungkinkan.

- `x509IssuerOu`— Penerbit (OU).
- `x509SanDns`— Nama alternatif subjek (DNS).
- `x509SanNameCn`— Nama alternatif penerbit (nama/CN).
- `x509SanUri`— Nama alternatif subjek (URI).
- `x509SubjectCn`Nama Subjek (CN)

`isBase64Encoded`

Menunjukkan apakah tubuh itu dikodekan base64. Hadir hanya jika protokolnya HTTP, HTTPS, atau gRPC dan badan permintaan belum berupa string.

`method`

Metode HTTP permintaan. Hadir hanya jika protokolnya HTTP, HTTPS, atau gRPC.

`path`

Jalur permintaan dari klien yang menyertakan parameter string kueri. Hadir hanya jika protokolnya HTTP, HTTPS, atau gRPC.

`queryStringParameters`

Parameter string kueri HTTP. Hadir hanya jika protokolnya HTTP, HTTPS, atau gRPC.

`serviceArn`

ARN dari layanan yang menerima permintaan.

`serviceNetworkArn`

ARN dari jaringan layanan yang memberikan permintaan.

`targetGroupArn`

ARN dari kelompok sasaran yang menerima permintaan.

`timeEpoch`

Waktu, dalam mikrodetik.

Example Contoh acara: V1

```
{
    "raw_path": "/path/to/resource?query1=value1&query2=value2",
    "method": "GET|POST|HEAD|...",
    "headers": {"header-key": "header-value", ... },
    "query_string_parameters": {"key": "value", ...},
    "body": "request-body",
    "is_base64_encoded": true|false
}
```

Menanggapi layanan VPC Lattice

Respon dari fungsi Lambda Anda harus mencakup status encoding Base64, kode status, dan header. Anda bisa menghilangkan bagian tubuhnya.

Untuk memasukkan konten biner dalam tubuh respon, Anda harus mengkodekan Base64 konten dan mengatur`isBase64Encoded`ke`true`. Layanan menerjemahkan konten untuk mengambil konten biner dan mengirimkannya ke klien di badan respons HTTP.

Layanan VPC Lattice tidak menghormati hop-by-hop header, seperti atau. `Connection` `Transfer-Encoding` Anda dapat menghilangkan `Content-Length` header karena layanan menghitungnya sebelum mengirim tanggapan ke klien.

Berikut ini adalah contoh respon dari fungsi Lambda:

```
{
    "isBase64Encoded": false,
    "statusCode": 200,
    "headers": {
        "Set-cookie": "cookies",
        "Content-Type": "application/json"
    },
    "body": "Hello from Lambda (optional)"
}
```

Header nilai ganda

VPC Lattice mendukung permintaan dari klien atau tanggapan dari fungsi Lambda yang berisi header dengan beberapa nilai atau berisi header yang sama beberapa kali. VPC Lattice meneruskan semua nilai ke target.

Dalam contoh berikut, ada dua header bernama header1 dengan nilai yang berbeda.

```
header1 = value1
header1 = value2
```

Dengan struktur peristiwa V2, VPC Lattice mengirimkan nilai dalam daftar. Contoh:

```
"header1": ["value1", "value2"]
```

Dengan struktur peristiwa V1, VPC Lattice menggabungkan nilai-nilai menjadi satu string. Contoh:

```
"header1": "value1, value2"
```

Parameter string kueri multi-nilai

VPC Lattice mendukung parameter kueri dengan beberapa nilai untuk kunci yang sama.

Dalam contoh berikut, ada dua parameter bernama QS1 dengan nilai yang berbeda.

```
http://www.example.com?&QS1=value1&QS1=value2
```

Dengan struktur peristiwa V2, VPC Lattice mengirimkan nilai dalam daftar. Contoh:

```
"QS1": ["value1", "value2"]
```

Dengan struktur peristiwa V1, VPC Lattice menggunakan nilai terakhir yang diteruskan. Contoh:

```
"QS1": "value2"
```

Deregristrasi fungsi Lambda

Jika Anda tidak perlu lagi mengirim lalu lintas ke fungsi Lambda Anda, Anda dapat membatalkan pendaftarannya. Setelah Anda membatalkan pendaftaran fungsi Lambda, permintaan dalam penerbangan gagal dengan galat HTTP 5XX.

Untuk mengganti fungsi Lambda, kami sarankan Anda membuat grup target baru, mendaftarkan fungsi baru dengan kelompok target baru, dan memperbarui aturan pendengar untuk menggunakan kelompok target baru bukan yang sudah ada.

Untuk membatalkan pendaftaran fungsi Lambda menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk menampilkan laman detailnya.
4. Pada tab Target, pilihDeregister.
5. Saat diminta konfirmasi, masukkan **`confirm`** lalu pilih Deregister.

Untuk membatalkan pendaftaran fungsi Lambda menggunakan AWS CLI

Gunakan perintah Target deregister.

Aplikasi Load Balancer sebagai target di VPC Lattice

Anda dapat membuat grup target VPC Lattice, mendaftarkan Application Load Balancer internal tunggal sebagai target, dan mengonfigurasi layanan VPC Lattice Anda untuk meneruskan lalu lintas ke grup target ini. Dalam skenario ini, Application Load Balancer mengambil alih keputusan routing

segera setelah lalu lintas mencapainya. Konfigurasi ini memungkinkan Anda untuk menggunakan fitur routing berbasis permintaan lapisan 7 dari Application Load Balancer dalam kombinasi dengan fitur yang didukung VPC Lattice, seperti autentikasi dan otorisasi IAM, dan konektivitas di seluruh dan akun. VPCs

Batasan

- Anda dapat mendaftarkan Application Load Balancer internal tunggal sebagai target dalam jenis grup target VPC Lattice. `ALB`
- Anda dapat mendaftarkan Application Load Balancer sebagai target hingga dua grup target VPC Lattice, yang digunakan oleh dua layanan VPC Lattice yang berbeda.
- VPC Lattice tidak menyediakan pemeriksaan kesehatan untuk kelompok target `ALB` tipe. Namun, Anda dapat mengonfigurasi pemeriksaan kesehatan secara independen di level load balancer untuk target di Elastic Load Balancing. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan kelompok target di Panduan Pengguna untuk Penyeimbang Beban Aplikasi

Prasyarat

Buat Application Load Balancer untuk mendaftar sebagai target dengan grup target VPC Lattice Anda. Penyeimbang beban harus memenuhi kriteria berikut:

- Skema penyeimbang beban adalah Internal.
- Application Load Balancer harus berada dalam akun yang sama dengan grup target VPC Lattice, dan harus dalam status Aktif.
- Application Load Balancer harus berada dalam VPC yang sama dengan grup target VPC Lattice.
- Anda dapat menggunakan pendengar HTTPS di Application Load Balancer untuk mengakhiri TLS, tetapi hanya jika layanan VPC Lattice menggunakan sertifikat yang sama dengan penyeimbang beban. SSL/TLS
- Untuk mempertahankan IP klien dari layanan VPC Lattice di header `X-Forwarded-For` permintaan, Anda harus mengatur atribut untuk Application Load Balancer ke. `routing.http.xff_header_processing.mode` `Preserve` Jika nilainya`Preserve`, penyeimbang beban mempertahankan `X-Forwarded-For` header dalam permintaan HTTP, dan mengirimkannya ke target tanpa perubahan apa pun.

Untuk informasi selengkapnya, lihat Membuat Application Load Balancer di Panduan Pengguna untuk Application Load Balancers.

Langkah 1: Buat grup target tipe ALB

Gunakan prosedur berikut untuk membuat grup target. Perhatikan bahwa VPC Lattice tidak mendukung pemeriksaan kesehatan untuk ALB kelompok sasaran. Namun, Anda dapat mengonfigurasi pemeriksaan kesehatan untuk grup target untuk Application Load Balancer Anda. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan grup target di Panduan Pengguna untuk Penyeimbang Beban Aplikasi.

Untuk membuat grup target

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. PilihBuat grup target.
4. Pada halaman Tentukan detail grup target, di bawah konfigurasi Dasar, pilih Application Load Balancer sebagai tipe target.
5. Untuk Name, masukkan nama untuk grup target.
6. Untuk Protokol, pilih**HTTP**,**HTTPS**, atau**TCP**. Protokol grup target harus sesuai dengan protokol pendengar untuk Application Load Balancer internal Anda.
7. Untuk Port, tentukan port untuk grup target Anda. Port ini harus cocok dengan port listener untuk Application Load Balancer internal Anda. Anda dapat menambahkan port listener pada Application Load Balancer internal agar sesuai dengan port grup target yang Anda tentukan di sini.
8. Untuk VPC, pilih virtual private cloud (VPC) yang sama dengan yang Anda pilih saat membuat Application Load Balancer internal. Ini harus menjadi VPC yang berisi sumber daya VPC Lattice Anda.
9. Untuk versi Protokol, pilih versi protokol yang didukung Application Load Balancer Anda.
10. (Opsional) Tambahkan tag yang diperlukan.
11. Pilih Berikutnya.

Langkah 2: Daftarkan Application Load Balancer sebagai target

Anda dapat mendaftarkan penyeimbang beban sebagai target sekarang atau nanti.

Mendaftarkan Application Load Balancer sebagai target

1. Pilih Daftar sekarang.

2. Untuk Application Load Balancer, pilih Application Load Balancer internal Anda.
3. Untuk Port, pertahankan default atau tentukan port yang berbeda sesuai kebutuhan. Port ini harus cocok dengan port listener yang ada di Application Load Balancer Anda. Jika Anda melanjutkan tanpa port yang cocok, lalu lintas tidak akan mencapai Application Load Balancer Anda.
4. PilihBuat grup target.

Versi protokol

Secara default, layanan mengirim permintaan ke target menggunakan HTTP/1.1. Anda dapat menggunakan versi protokol untuk mengirim permintaan ke target menggunakan HTTP/2 atau gRPC.

Tabel berikut merangkum hasil untuk kombinasi protokol permintaan dan versi protokol kelompok target.

Protokol permintaan	Versi protokol	Hasil
HTTP/1.1	HTTP/1.1	Sukses
HTTP/2	HTTP/1.1	Sukses
gRPC	HTTP/1.1	Kesalahan
HTTP/1.1	HTTP/2	Kesalahan
HTTP/2	HTTP/2	Sukses
gRPC	HTTP/2	Sukses jika target mendukung gRPC
HTTP/1.1	gRPC	Kesalahan
HTTP/2	gRPC	Sukses jika permintaan POST
gRPC	gRPC	Sukses

Pertimbangan untuk versi protokol gRPC

- Satu-satunya protokol pendengar yang didukung adalah HTTPS.
- Jenis-jenis target yang didukung hanya `INSTANCE` dan `IP`.
- Layanan mem-parsing permintaan gRPC dan merutekan panggilan gRPC ke grup target yang sesuai berdasarkan paket, layanan, dan metode.
- Anda tidak dapat menggunakan fungsi Lambda sebagai target.

Pertimbangan untuk versi protokol HTTP/2

- Satu-satunya protokol pendengar yang didukung adalah HTTPS. Anda dapat memilih HTTP atau HTTPS untuk protokol grup target.
- Satu-satunya aturan pendengar yang didukung adalah respons maju dan tetap.
- Jenis-jenis target yang didukung hanya `INSTANCE` dan `IP`.
- Layanan ini mendukung streaming dari klien. Layanan ini tidak mendukung streaming ke target.

Tag untuk grup target VPC Lattice

Tag membantu Anda mengategorikan grup target Auto dengan berbagai cara, misalnya, berdasarkan tujuan, pemilik, atau lingkungan.

Anda dapat menambahkan beberapa tag ke setiap grup Auto Scaling. Tombol tag harus unik untuk setiap kelompok target. Jika Anda menambahkan tag dengan kunci yang sudah terkait dengan grup target, maka akan memperbarui nilai tag tersebut.

Setelah selesai dengan tag, Anda dapat menghapusnya.

Pembatasan

- Jumlah maksimum tanda per sumber daya—50
- Panjang kunci maksimum – 127 karakter Unicode
- Panjang nilai maksimum—255 karakter Unicode
- Kunci dan nilai tag peka huruf besar/kecil. Karakter yang diizinkan adalah huruf, spasi, dan angka yang dapat diwakili dalam UTF-8, ditambah karakter khusus berikut: + - = . _:/@. Jangan gunakan spasi terkemuka atau paling belakang.

- Jangan gunakan `aws:` awalan dalam nama atau nilai tag Anda karena itu dicadangkan untuk AWS digunakan. Anda tidak dapat mengedit atau menghapus nama atau nilai tag dengan awalan ini. Tag dengan awalan ini tidak dihitung terhadap tag Anda per batas sumber daya.

Untuk memperbarui tag untuk grup target menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Grup target.
3. Pilih nama grup target untuk membuka halaman detailnya.
4. Pilih tab Tanda.
5. Untuk menambahkan tag, pilih Tambahkan tag dan masukkan kunci tag dan nilai tag. Untuk menambahkan tag lain, pilih Tambahkan tag baru. Setelah Anda selesai menambahkan tanda, pilih Simpan perubahan.
6. Untuk menghapus tag, pilih kotak centang untuk tag dan pilih Hapus. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk memperbarui tag untuk grup target menggunakan AWS CLI

Gunakan perintah tag-resource dan untag-resource.

Menghapus grup target VPC Lattice

Anda dapat menghapus kelompok target jika tidak direferensikan oleh tindakan lanjut dari aturan pendengar. Menghapus kelompok target tidak mempengaruhi target terdaftar dengan kelompok target. Jika Anda tidak lagi membutuhkan instance EC2 terdaftar, Anda dapat menghentikan atau menghapusnya.

Untuk menghapus grup target menggunakan konsol

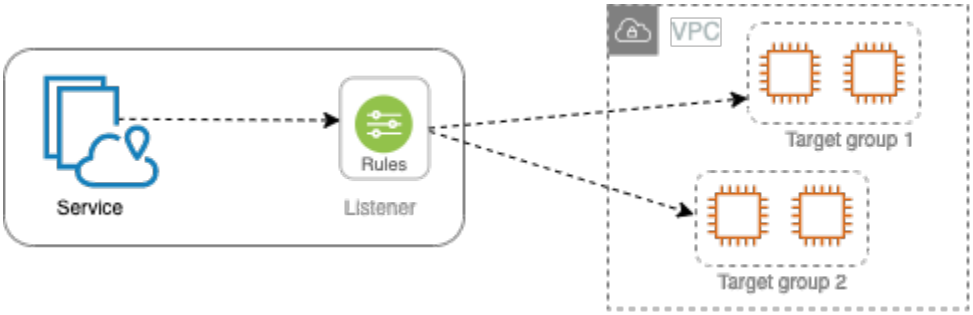
1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pada panel navigasi, pilih Grup sasaran.
3. Pilih kotak centang untuk grup target dan kemudian pilih Tindakan, Hapus.
4. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk menghapus grup target menggunakan AWS CLI

Gunakan perintah delete-target-group.

Pendengar untuk layanan VPC Lattice Anda

Sebelum Anda mulai menggunakan layanan VPC Lattice Anda, Anda harus menambahkan pendengar. Listener adalah proses yang memeriksa permintaan koneksi, menggunakan protokol dan port yang Anda konfigurasikan. Aturan yang Anda tetapkan untuk pendengar menentukan cara layanan merutekan permintaan ke target terdaftarnya.



Daftar Isi

Konfigurasi listener

Listener mendukung protokol dan port berikut ini:

- Protokol: HTTP, HTTPS, TLS
- Port: 1-65535

Jika protokol pendengar adalah HTTPS, VPC Lattice akan menyediakan dan mengelola sertifikat TLS yang terkait dengan VPC Lattice yang dihasilkan FQDN. VPC Lattice mendukung TLS pada HTTP/1.1 dan HTTP/2. Saat Anda mengonfigurasi layanan dengan pendengar HTTPS, VPC Lattice akan secara otomatis menentukan protokol HTTP menggunakan Application-Layer Protocol

Negotiation (ALPN). Jika ALPN tidak ada, VPC Lattice default ke HTTP/1.1. Untuk informasi selengkapnya, lihat Pendengar HTTPS.

VPC Lattice dapat mendengarkan HTTP, HTTPS, HTTP/1.1, dan HTTP/2 dan berkomunikasi dengan target di salah satu protokol dan versi ini. Kami tidak mengharuskan pendengar dan protokol grup target cocok. VPC Lattice mengelola seluruh proses upgrade dan downgrade antara protokol dan versi. Untuk informasi selengkapnya, lihat Versi protokol.

Anda dapat membuat pendengar TLS untuk memastikan bahwa aplikasi Anda mendekripsi lalu lintas terenkripsi, bukan VPC Lattice. Untuk informasi selengkapnya, lihat Pendengar TLS.

VPC Lattice tidak mendukung secara native. WebSockets Namun, Anda masih dapat terhubung ke layanan berbasis WebSocket dengan menggunakan Pendengar TLS atau perutean melalui sumber daya VPC Lattice.

Pendengar HTTP untuk layanan VPC Lattice

Listener adalah proses memeriksa permintaan koneksi. Anda dapat menentukan listener saat membuat layanan VPC Lattice Anda. Anda dapat menambahkan pendengar ke layanan Anda kapan saja.

Informasi di halaman ini membantu Anda membuat pendengar HTTP untuk layanan Anda. Untuk informasi tentang membuat pendengar yang menggunakan protokol lain, lihat dan. Pendengar HTTPS Pendengar TLS

Prasyarat

- Untuk menambahkan tindakan penerusan ke aturan pendengar default, Anda harus menentukan grup target VPC Lattice yang tersedia. Untuk informasi selengkapnya, lihat Buat grup target VPC Lattice.
- Anda dapat menentukan grup target yang sama di beberapa pendengar, tetapi pendengar ini harus milik layanan yang sama. Untuk menggunakan grup target dengan layanan VPC Lattice, Anda harus memverifikasi bahwa grup tersebut tidak digunakan oleh pendengar untuk layanan VPC Lattice lainnya.

Menambahkan listener HTTP

Anda dapat menambahkan pendengar dan aturan ke layanan Anda kapan saja. Anda mengonfigurasi listener dengan protokol dan port untuk koneksi dari klien ke layanan, dan grup target VPC Lattice untuk aturan pendengar default. Untuk informasi selengkapnya, lihat Konfigurasi listener.

Untuk menambahkan listener HTTP menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Add listener.
5. Untuk nama Listener, Anda dapat memberikan nama pendengar kustom, atau menggunakan protokol dan port listener Anda sebagai nama listener. Nama kustom yang Anda tentukan dapat memiliki hingga 63 karakter, dan itu harus unik untuk setiap layanan di akun Anda. Karakter yang valid adalah a-z, 0-9, dan tanda hubung (-). Anda tidak dapat menggunakan tanda hubung sebagai karakter pertama atau terakhir, atau segera setelah tanda hubung lainnya. Anda tidak dapat mengubah nama setelah Anda membuatnya.
6. Untuk Protokol: port, pilih HTTP dan masukkan nomor port.
7. Untuk tindakan Default, pilih grup target VPC Lattice untuk menerima lalu lintas dan pilih bobot yang akan ditetapkan ke grup target ini. Bobot yang Anda tetapkan ke grup sasaran menetapkan prioritasnya untuk menerima lalu lintas. Misalnya, jika dua kelompok sasaran memiliki bobot yang sama, masing-masing kelompok sasaran menerima setengah dari lalu lintas. Jika Anda telah menentukan hanya satu kelompok target, maka 100 persen dari lalu lintas dikirim ke satu kelompok target.

   Anda dapat menambahkan grup target lain secara opsional untuk tindakan default. Pilih Tambah tindakan dan kemudian pilih grup target dan tentukan bobotnya.
8. (Opsional) Untuk menambahkan aturan lain, pilih Tambahkan aturan lalu masukkan nama, prioritas, kondisi, dan tindakan untuk aturan tersebut.

   Anda dapat memberikan setiap aturan nomor prioritas antara 1 dan 100. Listener tidak bisa memiliki beberapa aturan dengan prioritas yang sama. Peraturan dievaluasi dalam urutan prioritas, dari nilai terendah ke nilai tertinggi. Peraturan default dievaluasi terakhir. Untuk informasi selengkapnya, lihat Aturan pendengar.

9. (Opsional) Untuk menambahkan tag, perluas tag Listener, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
10. Tinjau konfigurasi Anda, lalu pilih Tambah.

Untuk menambahkan pendengar HTTP menggunakan AWS CLI

Gunakan perintah create-listener untuk membuat listener dengan aturan default, dan perintah create-rule untuk membuat aturan listener tambahan.

Pendengar HTTPS untuk layanan VPC Lattice

Listener adalah proses memeriksa permintaan koneksi. Anda menentukan pendengar ketika Anda membuat layanan Anda. Anda dapat menambahkan pendengar ke layanan Anda di VPC Lattice kapan saja.

Anda dapat membuat pendengar HTTPS, yang menggunakan TLS versi 1.2 atau TLS versi 1.3 untuk menghentikan koneksi HTTPS dengan VPC Lattice secara langsung. VPC Lattice akan menyediakan dan mengelola sertifikat TLS yang terkait dengan VPC Lattice generated Fully Qualified Domain Name (FQDN). VPC Lattice mendukung TLS pada HTTP/1.1 dan HTTP/2. Saat Anda mengonfigurasi layanan dengan pendengar HTTPS, VPC Lattice akan secara otomatis menentukan protokol HTTP melalui Application-Layer Protocol Negotiation (ALPN). Jika ALPN tidak ada, VPC Lattice default ke HTTP/1.1.

VPC Lattice menggunakan arsitektur multi-tenancy, yang berarti dapat meng-host beberapa layanan pada titik akhir yang sama. VPC Lattice menggunakan TLS dengan Server Name Indication (SNI) untuk setiap permintaan klien. Hello Klien Terenkripsi (ECH) dan Indikasi Nama Server Terenkripsi (ESNI) tidak didukung.

VPC Lattice dapat mendengarkan HTTP, HTTPS, HTTP/1.1, dan HTTP/2 dan berkomunikasi dengan target di salah satu protokol dan versi ini. Konfigurasi pendengar dan grup target ini tidak perlu dicocokkan. VPC Lattice mengelola seluruh proses upgrade dan downgrade antara protokol dan versi. Untuk informasi selengkapnya, lihat Versi protokol.

Untuk memastikan bahwa aplikasi Anda mendekripsi lalu lintas, buat pendengar TLS sebagai gantinya. Dengan passthrough TLS, VPC Lattice tidak mengakhiri TLS. Untuk informasi selengkapnya, lihat Pendengar TLS.

Daftar Isi

Kebijakan keamanan

VPC Lattice menggunakan kebijakan keamanan yang merupakan kombinasi protokol TLSv1 .2 dan daftar cipher. SSL/TLS Protokol membuat koneksi aman antara klien dan server dan membantu memastikan bahwa semua data yang dilewatkan antara klien dan layanan Anda di VPC Lattice bersifat pribadi. Sandi adalah algoritme enkripsi yang menggunakan kunci enkripsi untuk membuat pesan kode. Protokol menggunakan beberapa cipher untuk mengenkripsi data. Selama proses negosiasi koneksi, klien dan VPC Lattice menyajikan daftar sandi dan protokol yang masing-masing mereka dukung, dalam urutan preferensi. Secara default, sandi pertama pada daftar server yang cocok salah satu sandi klien dipilih untuk sambungan aman.

VPC Lattice menggunakan SSL/TLS cipher TLS 1.2 berikut dalam urutan preferensi ini:

- `ECDHE-RSA-AES128-GCM-SHA256`
- `ECDHE-RSA-AES128-SHA`
- `ECDHE-RSA-AES256-GCM-SHA384`
- `ECDHE-RSA-AES256-SHA`
- `AES128-GCM-SHA256`
- `AES128-SHA`
- `AES256-GCM-SHA384`
- `AES256-SHA`

VPC Lattice juga menggunakan SSL/TLS cipher TLS 1.3 berikut dalam urutan preferensi ini:

- `TLS_AES_128_GCM_SHA256`
- `TLS_AES_256_GCM_SHA384`
- `TLS_CHACHA20_POLY1305_SHA256`

Kebijakan ALPN

Application-Layer Protocol Negotiation (ALPN) adalah ekstensi TLS yang dikirim pada pesan halo jabat tangan TLS awal. ALPN memungkinkan lapisan aplikasi untuk menegosiasikan protokol mana yang harus digunakan melalui koneksi aman, seperti HTTP/1 dan HTTP/2.

Ketika klien memulai koneksi ALPN, layanan VPC Lattice membandingkan daftar preferensi ALPN klien dengan kebijakan ALPN-nya. Jika klien mendukung protokol dari kebijakan ALPN, layanan VPC Lattice menetapkan koneksi berdasarkan daftar preferensi kebijakan ALPN. Jika tidak, layanan tidak menggunakan ALPN.

VPC Lattice mendukung kebijakan ALPN berikut:

`HTTP2Preferred`

Lebih suka HTTP/2 daripada HTTP/1.1. Daftar preferensi ALPN adalah h2, http/1.1.

Menambahkan pendengar HTTPS

Anda mengonfigurasi listener dengan protokol dan port untuk koneksi dari klien ke layanan, dan grup target untuk aturan pendengar default. Untuk informasi selengkapnya, lihat Konfigurasi listener.

Prasyarat

- Untuk menambahkan tindakan penerusan ke aturan pendengar default, Anda harus menentukan grup target VPC Lattice yang tersedia. Untuk informasi selengkapnya, lihat Buat grup target VPC Lattice.
- Anda dapat menentukan grup target yang sama di beberapa pendengar, tetapi pendengar ini harus termasuk dalam layanan VPC Lattice yang sama. Untuk menggunakan grup target dengan layanan VPC Lattice, Anda harus memverifikasi bahwa grup tersebut tidak digunakan oleh pendengar untuk layanan VPC Lattice lainnya.
- Anda dapat menggunakan sertifikat yang disediakan oleh VPC Lattice atau mengimpor sertifikat Anda sendiri ke. AWS Certificate Manager Untuk informasi selengkapnya, lihat the section called “BYOC”.

Untuk menambahkan listener HTTPS menggunakan konsol

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

2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Add listener.
5. Untuk nama Listener, Anda dapat memberikan nama pendengar kustom atau menggunakan protokol dan port listener Anda sebagai nama listener. Nama kustom yang Anda tentukan dapat memiliki hingga 63 karakter, dan itu harus unik untuk setiap layanan di akun Anda. Karakter yang valid adalah a-z, 0-9, dan tanda hubung (-). Anda tidak dapat menggunakan tanda hubung sebagai karakter pertama atau terakhir, atau segera setelah tanda hubung lainnya. Anda tidak dapat mengubah nama pendengar setelah Anda membuatnya.
6. Untuk Protokol: port, pilih HTTPS dan masukkan nomor port.
7. Untuk tindakan Default, pilih grup target VPC Lattice untuk menerima lalu lintas dan pilih bobot yang akan ditetapkan ke grup target ini. Bobot yang Anda tetapkan ke grup sasaran menetapkan prioritasnya untuk menerima lalu lintas. Misalnya, jika dua kelompok sasaran memiliki bobot yang sama, masing-masing kelompok sasaran menerima setengah dari lalu lintas. Jika Anda telah menentukan hanya satu kelompok target, maka 100 persen dari lalu lintas dikirim ke satu kelompok target.

   Anda dapat menambahkan grup target lain secara opsional untuk tindakan default. Pilih Tambah tindakan dan kemudian pilih grup target dan tentukan bobotnya.
8. (Opsional) Untuk menambahkan aturan lain, pilih Tambahkan aturan lalu masukkan nama, prioritas, kondisi, dan tindakan untuk aturan tersebut.

   Anda dapat memberikan setiap aturan nomor prioritas antara 1 dan 100. Listener tidak bisa memiliki beberapa aturan dengan prioritas yang sama. Peraturan dievaluasi dalam urutan prioritas, dari nilai terendah ke nilai tertinggi. Peraturan default dievaluasi terakhir. Untuk informasi selengkapnya, lihat Aturan pendengar.
9. (Opsional) Untuk menambahkan tag, perluas tag Listener, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
10. Untuk pengaturan sertifikat pendengar HTTPS, jika Anda tidak menentukan nama domain kustom saat membuat layanan, VPC Lattice secara otomatis menghasilkan sertifikat TLS untuk mengamankan lalu lintas yang mengalir melalui pendengar.

    Jika Anda membuat layanan dengan nama domain kustom, tetapi tidak menentukan sertifikat yang cocok, Anda dapat melakukannya sekarang dengan memilih sertifikat dari SSL/TLS sertifikat Kustom. Jika tidak, sertifikat yang Anda tentukan saat Anda membuat layanan sudah dipilih.

11. Tinjau konfigurasi Anda, lalu pilih Tambah.

Untuk menambahkan pendengar HTTPS menggunakan AWS CLI

Gunakan perintah create-listener untuk membuat listener dengan aturan default, dan perintah create-rule untuk membuat aturan listener tambahan.

Pendengar TLS untuk layanan VPC Lattice

Listener adalah proses memeriksa permintaan koneksi. Anda dapat menentukan listener saat membuat layanan VPC Lattice Anda. Anda dapat menambahkan pendengar ke layanan Anda kapan saja.

Anda dapat membuat pendengar TLS sehingga VPC Lattice meneruskan lalu lintas terenkripsi ke aplikasi Anda tanpa mendekripsi.

Jika Anda lebih suka VPC Lattice mendekripsi lalu lintas terenkripsi dan mengirimkan lalu lintas yang tidak terenkripsi ke aplikasi Anda, buatlah pendengar HTTPS sebagai gantinya. Untuk informasi selengkapnya, lihat Pendengar HTTPS.

Pertimbangan

Pertimbangan berikut berlaku untuk pendengar TLS:

- Layanan VPC Lattice harus memiliki nama domain khusus. Nama domain kustom layanan digunakan sebagai pencocokan Service Name Indication (SNI). Jika Anda menentukan sertifikat saat Anda membuat layanan, itu tidak digunakan.
- Satu-satunya aturan yang diizinkan untuk pendengar TLS adalah aturan default.
- Tindakan default untuk pendengar TLS harus berupa tindakan penerusan ke grup target TCP.
- Secara default, pemeriksaan kesehatan dinonaktifkan untuk grup target TCP. Jika Anda mengaktifkan pemeriksaan kesehatan untuk grup target TCP, Anda harus menentukan protokol dan versi protokol.
- Pendengar TLS merutekan permintaan menggunakan bidang SNI dari pesan client-hello. Anda dapat menggunakan wildcard dan sertifikat SAN pada target Anda jika kondisi pencocokan sama persis dengan client-hello.
- Karena semua lalu lintas tetap dienkripsi dari klien ke target, VPC Lattice tidak dapat membaca header HTTP dan tidak dapat menyisipkan atau menghapus header HTTP. Oleh karena itu, dengan pendengar TLS, ada batasan berikut:

  - Durasi koneksi dibatasi hingga 10 menit
  - Kebijakan autentikasi terbatas pada prinsipal anonim
  - Target Lambda tidak didukung
- Koneksi Websocket dapat menggunakan Pendengar TLS untuk terhubung ke, layanan VPC Lattice. Keterbatasan berikut ada:
  - Durasi koneksi dibatasi hingga 10 menit
  - Kebijakan autentikasi terbatas pada prinsipal anonim
  - Target Lambda tidak didukung
- Hello Klien Terenkripsi (ECH) tidak didukung.
- Indikasi Nama Server Terenkripsi (ESNI) tidak didukung.

Tambahkan pendengar TLS

Anda mengonfigurasi listener dengan protokol dan port untuk koneksi dari klien ke layanan, dan grup target untuk aturan pendengar default. Untuk informasi selengkapnya, lihat Konfigurasi listener.

Untuk menambahkan pendengar TLS menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Add listener.
5. Untuk nama Listener, Anda dapat memberikan nama pendengar kustom atau menggunakan protokol dan port listener Anda sebagai nama listener. Nama kustom yang Anda tentukan dapat memiliki hingga 63 karakter, dan itu harus unik untuk setiap layanan di akun Anda. Karakter yang valid adalah a-z, 0-9, dan tanda hubung (-). Anda tidak dapat menggunakan tanda hubung sebagai karakter pertama atau terakhir, atau segera setelah tanda hubung lainnya. Anda tidak dapat mengubah nama pendengar setelah Anda membuatnya.
6. Untuk Protokol, pilih TLS. Untuk Port, masukkan nomor port.
7. Untuk Forward to target group, pilih grup target VPC Lattice yang menggunakan protokol TCP untuk menerima lalu lintas, dan pilih bobot yang akan ditetapkan ke grup target ini. Anda dapat menambahkan grup target lain secara opsional. Pilih Tambahkan grup target dan kemudian pilih grup target dan masukkan bobotnya.

8. (Opsional) Untuk menambahkan tag, perluas tag Listener, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
9. Tinjau konfigurasi Anda, lalu pilih Tambah.

Untuk menambahkan pendengar TLS menggunakan AWS CLI

Gunakan perintah create-listener untuk membuat listener dengan aturan default. Tentukan protokol TLS_PASSTHOUGH.

Aturan pendengar untuk layanan VPC Lattice

Setiap pendengar memiliki aturan default dan aturan tambahan yang dapat Anda tentukan. Setiap aturan terdiri dari prioritas, satu atau beberapa tindakan, dan satu atau beberapa syarat. Anda dapat menambahkan atau mengedit peraturan kapan saja.

Daftar Isi

Peraturan default

Bila Anda membuat listener, Anda menentukan tindakan untuk peraturan default. Peraturan default tidak dapat memiliki syarat. Jika tidak ada syarat untuk peraturan listener yang terpenuhi, maka tindakan untuk peraturan default akan dilakukan.

Prioritas peraturan

Setiap peraturan memiliki prioritas. Peraturan dievaluasi dalam urutan prioritas, dari nilai terendah ke nilai tertinggi. Peraturan default dievaluasi terakhir. Anda dapat mengubah prioritas aturan non-default kapan saja. Anda tidak dapat mengubah prioritas peraturan default.

Tindakan aturan

Pendengar untuk layanan VPC Lattice mendukung tindakan maju dan tindakan respons tetap.

Tindakan ke depan

Anda dapat menggunakan `forward` tindakan untuk merutekan permintaan ke satu atau beberapa grup target VPC Lattice. Jika Anda menentukan beberapa kelompok target untuk tindakan `forward`, Anda harus menentukan bobot untuk setiap grup target. Bobot setiap grup target adalah nilai dari 0 hingga 999. Permintaan yang sesuai dengan peraturan listener dengan kelompok target tertimbang didistribusikan ke grup target ini berdasarkan bobot mereka. Misalnya, jika Anda menentukan dua grup target, masing-masing dengan bobot 10, setiap grup target menerima setengah dari permintaan. Jika Anda menentukan dua grup target, satu dengan bobot 10 dan lainnya dengan bobot 20, grup target dengan bobot 20 menerima permintaan dua kali lebih banyak dari grup target lainnya.

Tindakan respons tetap

Anda dapat menggunakan `fixed-response` untuk menjatuhkan permintaan klien dan mengembalikan respons HTTP khusus. Anda dapat menggunakan tindakan ini untuk mengembalikan kode respons 404 atau 500.

Example Contoh tindakan respons tetap untuk AWS CLI

Anda dapat menentukan tindakan saat membuat atau memperbarui aturan. Tindakan berikut mengirimkan respons tetap dengan kode status yang ditentukan.

```
"action": {
    "fixedResponse": {
        "statusCode": 404
},
```

Syarat peraturan

Setiap syarat peraturan memiliki jenis dan konfigurasi informasi. Bila syarat untuk suatu peraturan terpenuhi, maka tindakannya dilakukan.

Berikut ini adalah kriteria pencocokan yang didukung untuk aturan:

Pertandingan header

Routing didasarkan pada header HTTP untuk setiap permintaan. Anda dapat menggunakan syarat header HTTP untuk mengonfigurasi aturan yang merutekan permintaan berdasarkan header HTTP untuk permintaan tersebut. Anda dapat menentukan nama-nama bidang header HTTP standar atau kustom. Nama header dan evaluasi kecocokan tidak peka huruf besar/kecil. Anda dapat mengubah pengaturan ini dengan mengaktifkan sensitivitas huruf besar/kecil. Karakter wildcard tidak didukung dalam nama header. Awalan, tepat, dan berisi pencocokan didukung pada pencocokan header.

Metode pencocokan

Routing didasarkan pada metode permintaan HTTP dari setiap permintaan.

Anda dapat menggunakan syarat metode permintaan HTTP untuk mengonfigurasi aturan yang merutekan permintaan berdasarkan metode permintaan HTTP dari permintaan tersebut. Anda dapat menentukan metode HTTP standar atau kustom. Metode pencocokan peka huruf besar/kecil. Nama metode harus sama persis. Karakter wildcard tidak didukung.

Pertandingan jalur

Routing didasarkan pada pencocokan pola jalur dalam permintaan URLs.

Anda dapat menggunakan kondisi jalur untuk menentukan aturan yang merutekan permintaan berdasarkan URL dalam permintaan. Karakter wildcard tidak didukung. Awalan dan pencocokan tepat di jalur didukung.

Tambahkan peraturan

Anda dapat menambahkan aturan pendengar kapan saja.

Untuk menambahkan aturan listener menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Edit listener.
5. Perluas aturan Listener dan pilih Tambahkan aturan.
6. Untuk nama Aturan, masukkan nama untuk aturan.

7. Untuk Prioritas, masukkan prioritas antara 1 dan 100. Peraturan dievaluasi dalam urutan prioritas, dari nilai terendah ke nilai tertinggi. Peraturan default dievaluasi terakhir.
8. Untuk Kondisi, masukkan pola jalur untuk kondisi pencocokan jalur. Ukuran maksimum setiap string adalah 200 karakter. Perbandingannya tidak peka huruf besar/kecil. Karakter wildcard tidak didukung.

   Untuk menambahkan kondisi aturan kecocokan header atau kecocokan metode, gunakan AWS CLI atau AWS SDK.
9. Untuk Tindakan, pilih grup target VPC Lattice.
10. Pilih Simpan perubahan.

Untuk menambahkan aturan menggunakan AWS CLI

Gunakan perintah create-rule.

Perbarui aturan

Anda dapat memperbarui aturan pendengar kapan saja. Anda dapat memodifikasi prioritas, kondisi, kelompok target, dan bobot masing-masing kelompok target. Anda tidak dapat mengubah nama aturan.

Untuk memperbarui aturan listener menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Edit listener.
5. Ubah prioritas aturan, kondisi, dan tindakan sesuai kebutuhan.
6. Tinjau pembaruan Anda dan pilih Simpan perubahan.

Untuk memperbarui aturan menggunakan AWS CLI

Gunakan perintah update-rule.

Menghapus peraturan

Anda dapat menghapus aturan non-default untuk pendengar kapan saja. Anda tidak dapat menghapus peraturan default untuk listener. Saat Anda menghapus pendengar, semua aturannya akan dihapus.

Untuk menghapus aturan listener menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Edit listener.
5. Temukan aturannya dan pilih Hapus.
6. Pilih Simpan perubahan.

Untuk menghapus aturan menggunakan AWS CLI

Gunakan perintah hapus-peraturan.

Menghapus listener untuk layanan VPC Lattice

Anda dapat menghapus listener kapan saja. Saat Anda menghapus pendengar, semua aturannya akan dihapus secara otomatis.

Untuk menghapus listener menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Layanan.
3. Pilih nama layanan untuk membuka halaman detailnya.
4. Pada tab Routing, pilih Hapus pendengar.
5. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk menghapus pendengar menggunakan AWS CLI

Gunakan perintah hapus-listener.

Sumber daya VPC di Amazon VPC Lattice

Anda dapat berbagi sumber daya VPC dengan tim lain di organisasi Anda atau dengan mitra vendor perangkat lunak independen eksternal (ISV). Sumber daya VPC dapat berupa sumber daya AWS-native seperti database Amazon RDS, nama domain, atau alamat IP. Sumber daya dapat berada di VPC atau jaringan lokal Anda dan tidak perlu diseimbangkan beban. Anda gunakan AWS RAM untuk menentukan kepala sekolah yang dapat mengakses sumber daya. Anda membuat gateway sumber daya di mana sumber daya Anda dapat diakses. Anda juga membuat konfigurasi sumber daya yang mewakili sumber daya atau sekelompok sumber daya yang ingin Anda bagikan.

Prinsipal tempat Anda berbagi sumber daya dapat mengakses sumber daya ini secara pribadi menggunakan titik akhir VPC. Mereka dapat menggunakan titik akhir VPC sumber daya untuk mengakses satu sumber daya atau mengumpulkan beberapa sumber daya dalam jaringan layanan VPC Lattice, dan mengakses jaringan layanan menggunakan titik akhir VPC jaringan layanan.

Bagian berikut menjelaskan cara membuat dan mengelola sumber daya VPC di VPC Lattice:

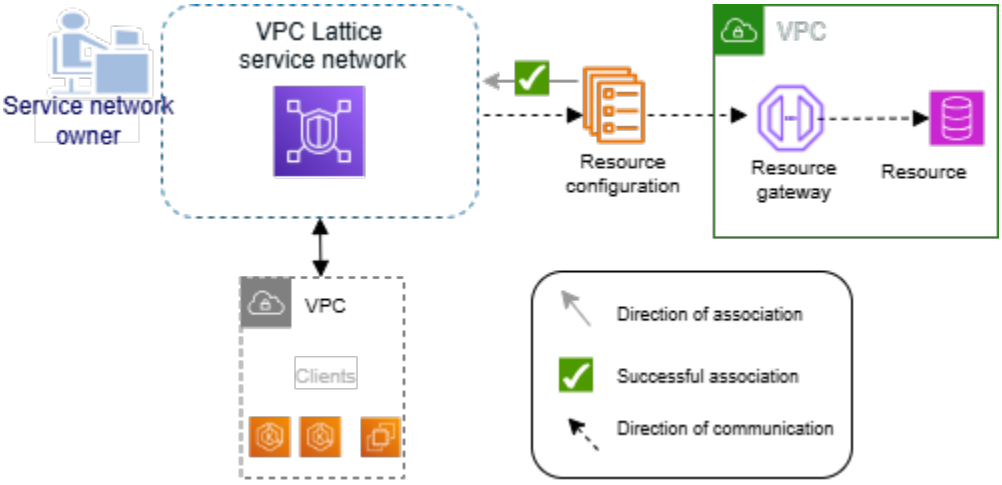
Topik

Gateway sumber daya di VPC Lattice

Gateway sumber daya adalah titik yang menerima lalu lintas ke VPC tempat sumber daya berada. Ini mencakup beberapa Availability Zone.

VPC harus memiliki gateway sumber daya jika Anda berencana membuat sumber daya di dalam VPC dapat diakses dari akun lain atau akun. VPCs Setiap sumber daya yang Anda bagikan dikaitkan dengan gateway sumber daya. Ketika klien di akun lain VPCs atau mengakses sumber daya di VPC Anda, sumber daya melihat lalu lintas yang datang secara lokal dari gateway sumber daya di VPC tersebut. Alamat IP sumber lalu lintas adalah alamat IP gateway sumber daya di Availability Zone. Beberapa konfigurasi sumber daya, masing-masing memiliki banyak sumber daya, dapat dilampirkan ke gateway sumber daya.

Diagram berikut menunjukkan bagaimana klien mengakses sumber daya melalui gateway sumber daya:



Daftar Isi

Pertimbangan-pertimbangan

Pertimbangan berikut berlaku untuk gateway sumber daya:

- Agar sumber daya dapat diakses dari semua Availability Zone, Anda harus membuat gateway sumber daya untuk menjangkau sebanyak mungkin Availability Zone.
- Setidaknya satu Availability Zone dari titik akhir VPC dan gateway sumber daya harus tumpang tindih.
- VPC dapat memiliki maksimal 100 gateway sumber daya. Untuk informasi selengkapnya, lihat Kuota untuk Kisi VPC.
- VPC Lattice mungkin menambahkan yang baru ENIs ke gateway sumber daya Anda.
- Gateway sumber daya dengan subnet VPC bersama:
  - Gateway sumber daya hanya dapat digunakan ke subnet VPC bersama oleh akun yang memiliki VPC.

- Konfigurasi sumber daya untuk gateway sumber daya hanya dapat dibuat oleh akun yang memiliki gateway sumber daya.

Grup keamanan

Anda dapat melampirkan grup keamanan ke gateway sumber daya. Aturan grup keamanan untuk gateway sumber daya mengontrol lalu lintas keluar dari gateway sumber daya ke sumber daya.

Aturan keluar yang disarankan untuk lalu lintas yang mengalir dari gateway sumber daya ke sumber daya database

Agar lalu lintas mengalir dari gateway sumber daya ke sumber daya, Anda harus membuat aturan keluar untuk protokol pendengar dan rentang port sumber daya yang diterima.

Tujuan	Protokol	Rentang port	Komentar
CIDR range for resource	TCP	3306	Mengizinkan lalu lintas dari gateway sumber daya ke database.

Jenis alamat IP

Gateway sumber daya dapat memiliki IPv4, IPv6 atau alamat dual-stack. Jenis alamat IP dari gateway sumber daya harus kompatibel dengan subnet gateway sumber daya dan jenis alamat IP sumber daya, seperti yang dijelaskan di sini:

- IPv4— Tetapkan IPv4 alamat ke antarmuka jaringan gateway sumber daya Anda. Opsi ini didukung hanya jika semua subnet yang dipilih memiliki rentang IPv4 alamat, dan sumber daya juga memiliki IPv4 alamat. Bila Anda menggunakan opsi ini, Anda dapat mengonfigurasi jumlah IPv4 alamat per gateway sumber daya ENI.
- IPv6— Tetapkan IPv6 alamat ke antarmuka jaringan gateway sumber daya Anda. Opsi ini didukung hanya jika semua subnet yang dipilih IPv6 hanya subnet, dan sumber daya juga memiliki IPv6 alamat. Saat Anda menggunakan opsi ini, IPv6 alamat ditetapkan secara otomatis dan tidak perlu dikelola.
- Dualstack — Tetapkan keduanya IPv4 dan IPv6 alamat ke antarmuka jaringan gateway sumber daya Anda. Opsi ini didukung hanya jika semua subnet yang dipilih memiliki rentang IPv6 alamat

IPv4 dan keduanya, dan sumber daya memiliki IPv6 alamat IPv4 atau. Bila Anda menggunakan opsi ini, Anda dapat mengonfigurasi jumlah IPv4 alamat per gateway sumber daya ENI.

Jenis alamat IP dari gateway sumber daya tidak tergantung pada jenis alamat IP klien atau titik akhir VPC yang melaluinya sumber daya diakses.

IPv4 alamat per ENI

Jika gateway sumber daya Anda memiliki IPv4 atau tipe alamat IP dual-stack, Anda dapat mengonfigurasi jumlah IPv4 alamat yang ditetapkan untuk setiap ENI dari gateway sumber daya Anda. Saat Anda membuat gateway sumber daya, Anda memilih dari 1 hingga 62 IPv4 alamat. Setelah Anda mengatur jumlah IPv4 alamat, nilainya tidak dapat diubah.

IPv4 Alamat digunakan untuk terjemahan alamat jaringan dan menentukan jumlah maksimum IPv4 koneksi bersamaan ke sumber daya. Setiap IPv4 alamat dapat mendukung hingga 55.000 koneksi simultan per IP tujuan. Secara default, semua gateway sumber daya diberikan 16 IPv4 alamat per ENI.

Jika gateway sumber daya Anda menggunakan jenis IPv6 alamat, gateway sumber daya secara otomatis menerima /80 CIDR per ENI. Nilai ini tidak dapat diubah. Unit transmisi maksimum (MTU) per koneksi adalah 8500 byte.

Membuat gateway sumber daya di VPC Lattice

Gunakan konsol untuk membuat gateway sumber daya.

Untuk membuat gateway sumber daya menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah PrivateLink dan Lattice, pilih gateway sumber daya.
3. Pilih Buat gateway sumber daya.
4. Untuk nama gateway Sumber Daya, masukkan nama yang unik di AWS akun Anda.
5. Untuk jenis alamat IP, pilih jenis alamat IP untuk gateway sumber daya.

    - Jika Anda memilih IPv4atau Dualstack untuk jenis alamat IP, Anda dapat memasukkan jumlah IPv4 alamat per ENI untuk gateway sumber daya Anda.

      Standarnya adalah 16 IPv4 alamat per ENI. Ini adalah jumlah yang cocok IPs untuk membentuk koneksi dengan sumber daya backend Anda.

6. Untuk VPC, pilih VPC dan subnet untuk membuat gateway sumber daya Anda.
7. Untuk grup Keamanan, pilih hingga lima grup keamanan untuk mengontrol lalu lintas masuk dari VPC ke jaringan layanan.
8. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
9. Pilih Buat gateway sumber daya.

Untuk membuat gateway sumber daya menggunakan AWS CLI

Gunakan perintah create-resource-gateway.

Menghapus gateway sumber daya di VPC Lattice

Gunakan konsol untuk menghapus gateway sumber daya.

Untuk menghapus gateway sumber daya menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah PrivateLink dan Lattice, pilih gateway sumber daya.
3. Pilih kotak centang untuk gateway sumber daya yang ingin Anda hapus dan pilih Tindakan, Hapus. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk menghapus gateway sumber daya menggunakan AWS CLI

Gunakan perintah delete-resource-gateway.

Konfigurasi sumber daya untuk sumber daya VPC

Konfigurasi sumber daya mewakili sumber daya atau sekelompok sumber daya yang ingin Anda buat dapat diakses oleh klien di akun lain VPCs dan akun. Dengan mendefinisikan konfigurasi sumber daya, Anda dapat mengizinkan konektivitas jaringan pribadi, aman, searah ke sumber daya di VPC Anda dari klien di akun lain dan akun. VPCs Konfigurasi sumber daya dikaitkan dengan gateway sumber daya yang melaluinya ia menerima lalu lintas. Agar sumber daya dapat diakses dari VPC lain, ia harus memiliki konfigurasi sumber daya.

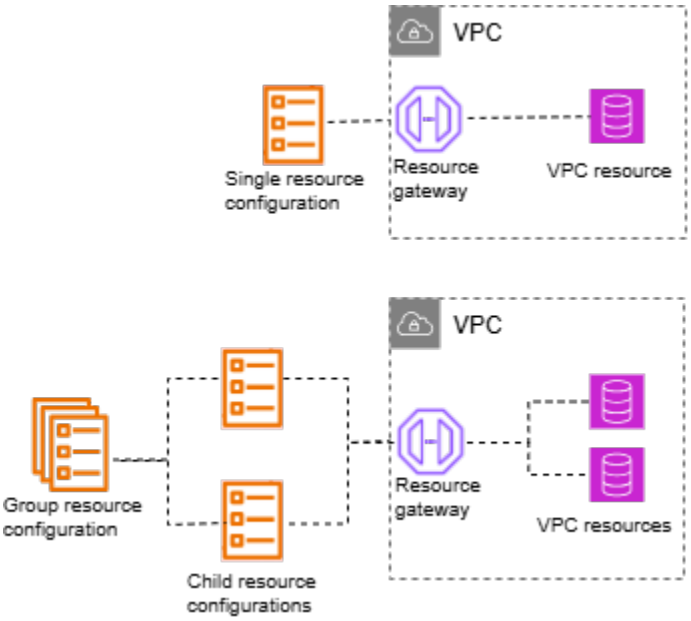
Daftar Isi

Jenis konfigurasi sumber daya

Konfigurasi sumber daya dapat terdiri dari beberapa jenis. Jenis yang berbeda membantu mewakili berbagai jenis sumber daya. Jenisnya adalah:

- Konfigurasi sumber daya tunggal: Merupakan alamat IP atau nama domain. Itu dapat dibagikan secara independen.
- Konfigurasi sumber daya grup: Ini adalah kumpulan konfigurasi sumber daya anak. Ini dapat digunakan untuk mewakili sekelompok DNS dan titik akhir alamat IP.
- Konfigurasi sumber daya anak: Ini adalah anggota dari konfigurasi sumber daya grup. Ini mewakili alamat IP atau nama domain. Itu tidak dapat dibagikan secara independen; itu hanya dapat dibagikan sebagai bagian dari grup. Itu dapat ditambahkan dan dihapus dari grup. Ketika ditambahkan, secara otomatis dapat diakses oleh mereka yang dapat mengakses grup.
- Konfigurasi sumber daya ARN: Merupakan tipe sumber daya yang didukung yang disediakan oleh layanan. AWS Setiap hubungan kelompok-anak secara otomatis diurus.

Gambar berikut menunjukkan konfigurasi sumber daya tunggal, anak, dan grup:



Protokol

Saat Anda membuat konfigurasi sumber daya, Anda dapat menentukan protokol yang akan didukung oleh sumber daya. Saat ini, hanya protokol TCP yang didukung.

Gateway sumber daya

Konfigurasi sumber daya dikaitkan dengan gateway sumber daya. Sebuah gateway sumber daya adalah satu set ENIs yang berfungsi sebagai titik masuknya ke dalam VPC di mana sumber daya berada. Beberapa konfigurasi sumber daya dapat dikaitkan dengan gateway sumber daya yang sama. Ketika klien di akun lain VPCs atau mengakses sumber daya di VPC Anda, sumber daya melihat lalu lintas yang datang secara lokal dari alamat IP gateway sumber daya di VPC itu.

Nama domain khusus untuk penyedia sumber daya

Penyedia sumber daya dapat melampirkan nama domain khusus ke konfigurasi sumber daya, seperti`example.com`, sumber daya yang dapat digunakan konsumen untuk mengakses konfigurasi sumber daya. Nama domain kustom dapat dimiliki dan diverifikasi oleh penyedia sumber daya, atau dapat berupa pihak ketiga atau AWS domain. Penyedia sumber daya dapat menggunakan konfigurasi sumber daya untuk berbagi cluster cache dan cluster Kafka, aplikasi berbasis TLS, atau sumber daya lainnya. AWS

Pertimbangan berikut berlaku untuk penyedia konfigurasi sumber daya:

- Konfigurasi sumber daya hanya dapat memiliki satu domain khusus.
- Nama domain kustom dari konfigurasi sumber daya tidak dapat diubah.
- Nama domain kustom dapat dilihat oleh semua konsumen konfigurasi sumber daya.
- Anda dapat memverifikasi nama domain kustom Anda menggunakan proses verifikasi nama domain di VPC Lattice. Untuk informasi lebih lanjut Untuk informasi lebih lanjut, lihatthe section called “Membuat dan memverifikasi domain”.
- Untuk konfigurasi sumber daya grup tipe dan anak, Anda harus terlebih dahulu menentukan domain grup pada konfigurasi sumber daya grup. Setelah itu, konfigurasi sumber daya anak dapat memiliki domain kustom yang merupakan subdomain dari domain grup. Jika grup tidak memiliki domain grup, Anda dapat menggunakan nama domain khusus apa pun untuk anak, tetapi VPC Lattice tidak akan menyediakan zona yang dihosting untuk nama domain anak di VPC konsumen sumber daya.

Nama domain khusus untuk konsumen sumber daya

Ketika konsumen sumber daya mengaktifkan konektivitas ke konfigurasi sumber daya yang memiliki nama domain khusus, mereka dapat mengizinkan VPC Lattice untuk mengelola zona host pribadi Route 53 di VPC mereka. Konsumen sumber daya memiliki opsi terperinci untuk domain mana yang ingin mereka izinkan VPC Lattice mengelola zona host pribadi.

Konsumen sumber daya dapat mengatur `private-dns-enabled` parameter saat mengaktifkan konektivitas ke konfigurasi sumber daya melalui titik akhir sumber daya, titik akhir jaringan layanan, atau asosiasi VPC jaringan layanan. Seiring dengan `private-dns-enabled` parameter, konsumen dapat menggunakan opsi DNS untuk menentukan domain mana yang mereka inginkan untuk VPC Lattice untuk mengelola zona host pribadi. Konsumen dapat memilih antara preferensi DNS pribadi berikut:

`ALL_DOMAINS`

VPC Lattice menyediakan zona host pribadi untuk semua nama domain kustom.

`VERIFIED_DOMAINS_ONLY`

VPC Lattice menyediakan zona host pribadi hanya jika nama domain kustom telah diverifikasi oleh penyedia.

`VERIFIED_DOMAINS_AND_SPECIFIED_DOMAINS`

VPC Lattice menyediakan zona host pribadi untuk semua nama domain kustom terverifikasi dan nama domain lain yang ditentukan oleh konsumen sumber daya. Konsumen sumber daya menentukan nama domain dalam `private DNS specified domains` parameter.

`SPECIFIED_DOMAINS_ONLY`

VPC Lattice menyediakan zona host pribadi untuk nama domain yang ditentukan oleh konsumen sumber daya. Konsumen sumber daya menentukan nama domain dalam `private DNS specified domains` parameter.

Saat Anda mengaktifkan DNS pribadi, VPC Lattice membuat zona host pribadi di VPC Anda untuk nama domain kustom yang terkait dengan konfigurasi sumber daya. Secara default, preferensi DNS pribadi diatur ke`VERIFIED_DOMAINS_ONLY`. Ini berarti bahwa zona host pribadi dibuat hanya jika nama domain kustom telah diverifikasi oleh penyedia sumber daya. Jika Anda menyetel preferensi DNS pribadi Anda ke `ALL_DOMAINS` atau `SPECIFIED_DOMAINS_ONLY` kemudian VPC Lattice membuat zona host pribadi terlepas dari status verifikasi nama domain kustom. Ketika zona host pribadi dibuat untuk domain tertentu, semua lalu lintas ke domain tersebut dari VPC Anda dirutekan melalui VPC Lattice. Kami menyarankan Anda menggunakan`ALL_DOMAINS`,`VERIFIED_DOMAINS_AND_SPECIFIED_DOMAINS`, atau `SPECIFIED_DOMAINS_ONLY` preferensi hanya ketika Anda ingin lalu lintas ke nama domain kustom ini melalui VPC Lattice.

Kami menyarankan agar konsumen sumber daya menetapkan preferensi DNS pribadi mereka. `VERIFIED_DOMAINS_ONLY` Hal ini memungkinkan konsumen memperketat perimeter keamanan mereka dengan hanya mengizinkan VPC Lattice untuk menyediakan zona host pribadi untuk domain terverifikasi di akun konsumen sumber daya.

Untuk memilih domain di domain yang ditentukan DNS pribadi, konsumen sumber daya dapat memasukkan nama domain yang sepenuhnya memenuhi syarat, seperti `my.example.com` atau menggunakan wildcard seperti. `*.example.com`

Pertimbangan berikut berlaku untuk konsumen konfigurasi sumber daya:

- Parameter berkemampuan DNS pribadi tidak dapat diubah.
- DNS pribadi harus diaktifkan pada asosiasi sumber daya jaringan layanan untuk host pribadi yang akan dibuat dalam VPC. Untuk konfigurasi sumber daya, status DNS pribadi yang diaktifkan dari

asosiasi sumber daya jaringan layanan akan mengesampingkan status diaktifkan DNS pribadi dari titik akhir jaringan layanan atau asosiasi VPC jaringan layanan.

Nama domain khusus untuk pemilik jaringan layanan

Properti berkemampuan DNS pribadi dari asosiasi sumber daya jaringan layanan mengesampingkan properti berkemampuan DNS pribadi dari titik akhir jaringan layanan dan asosiasi VPC jaringan layanan.

Jika pemilik jaringan layanan membuat asosiasi sumber daya jaringan layanan dan tidak mengaktifkan DNS pribadi, VPC Lattice tidak akan menyediakan zona yang dihosting pribadi untuk konfigurasi sumber daya tersebut di VPCs mana pun jaringan layanan terhubung, meskipun DNS pribadi diaktifkan pada titik akhir jaringan layanan atau asosiasi VPC jaringan layanan.

Untuk konfigurasi sumber daya tipe ARN, bendera DNS pribadi adalah benar dan tidak dapat diubah.

Definisi sumber daya

Dalam konfigurasi sumber daya, identifikasi sumber daya dengan salah satu cara berikut:

- Dengan Nama Sumber Daya Amazon (ARN): Jenis sumber daya yang didukung yang disediakan oleh layanan AWS , dapat diidentifikasi oleh ARN mereka. Hanya database Amazon RDS yang didukung. Anda tidak dapat membuat konfigurasi sumber daya untuk klaster yang dapat diakses publik.
- Dengan target nama domain: Anda dapat menggunakan nama domain apa pun yang dapat diselesaikan secara publik. Jika nama domain Anda menunjuk ke IP yang berada di luar VPC Anda, Anda harus memiliki gateway NAT di VPC Anda.
- Dengan alamat IP: Untuk IPv4, tentukan IP pribadi dari rentang berikut: 10.0.0.0/8, 100.64.0.0/10, 172.16.0.0/12, 192.168.0.0/16. Untuk IPv6, tentukan IP dari VPC. Publik IPs tidak didukung.

Rentang pelabuhan

Saat Anda membuat konfigurasi sumber daya, Anda dapat menentukan port yang akan menerima permintaan. Akses klien pada port lain tidak akan diizinkan.

Mengakses sumber daya

Konsumen dapat mengakses konfigurasi sumber daya langsung dari VPC mereka menggunakan titik akhir VPC atau melalui jaringan layanan. Sebagai konsumen, Anda dapat mengaktifkan akses dari VPC Anda ke konfigurasi sumber daya yang ada di akun Anda atau yang telah dibagikan dengan Anda dari akun lain melalui. AWS RAM

- Mengakses konfigurasi sumber daya secara langsung

  Anda dapat membuat titik akhir AWS PrivateLink VPC dari sumber daya tipe (titik akhir sumber daya) di VPC Anda untuk mengakses konfigurasi sumber daya secara pribadi dari VPC Anda. Untuk informasi selengkapnya tentang cara membuat titik akhir sumber daya, lihat Mengakses sumber daya VPC di panduan pengguna.AWS PrivateLink
- Mengakses konfigurasi sumber daya melalui jaringan layanan

  Anda dapat mengaitkan konfigurasi sumber daya ke jaringan layanan, dan menghubungkan VPC Anda ke jaringan layanan. Anda dapat menghubungkan VPC Anda ke jaringan layanan baik melalui asosiasi atau menggunakan titik akhir VPC AWS PrivateLink jaringan layanan.

  Untuk informasi selengkapnya tentang asosiasi jaringan layanan, lihat Mengelola asosiasi untuk jaringan layanan VPC Lattice.

  Untuk informasi selengkapnya tentang titik akhir VPC jaringan layanan, lihat Mengakses jaringan layanan di panduan pengguna.AWS PrivateLink

Saat DNS pribadi diaktifkan untuk VPC, Anda tidak dapat membuat titik akhir sumber daya dan titik akhir jaringan layanan untuk konfigurasi sumber daya yang sama.

Asosiasi dengan jenis jaringan layanan

Ketika Anda berbagi konfigurasi sumber daya dengan akun konsumen, misalnya, Account-B, melalui AWS RAM, Account-B dapat mengakses konfigurasi sumber daya baik secara langsung melalui titik akhir VPC sumber daya, atau melalui jaringan layanan.

Untuk mengakses konfigurasi sumber daya melalui jaringan layanan, Account-B harus mengaitkan konfigurasi sumber daya dengan jaringan layanan. Jaringan layanan dapat dibagikan antar akun. Jadi, Account-B dapat berbagi jaringan layanan mereka (yang konfigurasi sumber daya dikaitkan dengan) dengan Account-C, membuat sumber daya Anda dapat diakses dari Account-C.

Untuk mencegah berbagi transitif tersebut, Anda dapat menentukan bahwa konfigurasi sumber daya Anda tidak dapat ditambahkan ke jaringan layanan yang dapat dibagikan antar akun. Jika Anda menentukan ini, Account-B tidak akan dapat menambahkan konfigurasi sumber daya Anda ke jaringan layanan yang dibagikan atau dapat dibagikan dengan akun lain di masa mendatang.

Jenis jaringan layanan

Ketika Anda berbagi konfigurasi sumber daya dengan akun lain, misalnya Account-B, melalui AWS RAM, Account-B dapat mengakses sumber daya yang ditentukan dalam konfigurasi sumber daya dalam salah satu dari tiga cara:

- Menggunakan titik akhir VPC dari sumber daya tipe (titik akhir VPC sumber daya).
- Menggunakan titik akhir VPC dari jenis jaringan layanan (titik akhir VPC jaringan layanan).
- Menggunakan asosiasi VPC jaringan layanan.

  Saat Anda menggunakan asosiasi jaringan layanan, setiap sumber daya diberi IP per subnet dari blok 129.224.0.0/17, yang dimiliki dan tidak dapat dirutekan. AWS Ini merupakan tambahan dari daftar awalan terkelola yang digunakan VPC Lattice untuk merutekan lalu lintas ke layanan melalui jaringan VPC Lattice. Keduanya IPs diperbarui ke tabel rute VPC Anda.

Untuk titik akhir VPC jaringan layanan dan asosiasi VPC jaringan layanan, konfigurasi sumber daya harus dikaitkan dengan jaringan layanan di Account-B. Jaringan layanan dapat dibagikan antar akun. Jadi, Account-B dapat berbagi jaringan layanan mereka (yang berisi konfigurasi sumber daya) dengan Account-C, membuat sumber daya Anda dapat diakses dari Account-C. Untuk mencegah berbagi transitif tersebut, Anda dapat melarang konfigurasi sumber daya Anda ditambahkan ke jaringan layanan yang dapat dibagikan antar akun. Jika Anda melarang ini, Account-B tidak akan dapat menambahkan konfigurasi sumber daya Anda ke jaringan layanan yang dibagikan atau dapat dibagikan dengan akun lain.

Berbagi konfigurasi sumber daya melalui AWS RAM

Konfigurasi sumber daya terintegrasi dengan AWS Resource Access Manager. Anda dapat membagikan konfigurasi sumber daya Anda dengan akun lain melalui AWS RAM. Saat Anda berbagi konfigurasi sumber daya dengan AWS akun, klien di akun tersebut dapat mengakses sumber daya secara pribadi. Anda dapat berbagi konfigurasi sumber daya menggunakan pembagian sumber daya di AWS RAM.

Gunakan AWS RAM konsol, untuk melihat pembagian sumber daya yang telah ditambahkan, sumber daya bersama yang dapat Anda akses, dan AWS akun yang telah berbagi sumber daya dengan Anda. Untuk informasi selengkapnya, lihat Sumber daya yang dibagikan dengan Anda di Panduan AWS RAM Pengguna.

Untuk mengakses sumber daya dari VPC lain di akun yang sama dengan konfigurasi sumber daya, Anda tidak perlu membagikan konfigurasi sumber daya. AWS RAM

Memantau

Anda dapat mengaktifkan log pemantauan pada konfigurasi sumber daya Anda. Anda dapat memilih tujuan untuk mengirim log ke.

Membuat dan memverifikasi domain

Verifikasi nama domain adalah entitas yang memungkinkan Anda membuktikan kepemilikan domain tertentu. Sebagai penyedia sumber daya, Anda dapat menggunakan domain dan subdomainnya sebagai nama domain khusus untuk konfigurasi sumber daya Anda. Konsumen sumber daya dapat melihat status verifikasi nama domain kustom Anda saat mereka menjelaskan konfigurasi sumber daya.

Mulai verifikasi domain

Anda memulai verifikasi nama domain menggunakan VPC Lattice, dan kemudian Anda menggunakan zona DNS Anda untuk menyelesaikan proses.

Konsol Manajemen AWS

Untuk memulai verifikasi nama domain

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah PrivateLink dan Lattice, pilih Verifikasi domain
3. Pilih Mulai verifikasi domain.
4. Untuk nama Domain, masukkan nama domain yang Anda miliki.
5. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
6. Pilih Mulai verifikasi nama domain.

Setelah berhasil memulai verifikasi nama domain Anda, VPC Lattice mengembalikan dan file. `Id` `txtMethodConfig` Anda menggunakan `txtMethodConfig` untuk menyelesaikan verifikasi nama domain Anda.

AWS CLI

`start-domain-verification`Perintah berikut memulai verifikasi nama domain:

```
aws vpc-lattice start-domain-verification \
  --domain-name example.com
```

Outputnya terlihat seperti berikut:

```
{
    "id": "dv-aaaa0000000111111",
    "arn": "arn:aws:vpc-lattice:us-west-2:111122223333:domainverification/dv-
aaaa0000000111111",
    "domainName": "example.com",
    "status": "PENDING",
    "txtMethodConfig": {
        "value": "vpc-lattice:1111aaaaaaa",
        "name": "_11111aaaaaaaaa"
    }
}
```

VPC Lattice mengembalikan dan. `Id` `txtMethodConfig` Anda menggunakan `txtMethodConfig` untuk menyelesaikan verifikasi nama domain Anda. Dalam contoh ini, `txtMethodConfig` adalah sebagai berikut:

```
txtMethodConfig": {
        "value": "vpc-lattice:1111aaaaaaa",
        "name": "_11111aaaaaaaaa"
    }
```

Lengkapi verifikasi nama domain

Untuk menyelesaikan verifikasi nama domain, Anda menambahkan catatan TXT di zona DNS Anda. Jika Anda menggunakan Route 53, gunakan zona host nama domain Anda. Saat Anda memverifikasi nama domain, subdomain apa pun juga diverifikasi. Misalnya, jika

Anda memverifikasi`example.com`, Anda dapat mengaitkan konfigurasi sumber daya dengan `alpha.example.com` dan `beta.example.com` tanpa melakukan verifikasi tambahan.

Untuk membuat rekaman TXT menggunakan Konsol Manajemen AWS, lihat Membuat rekaman menggunakan konsol Amazon Route 53.

Untuk membuat catatan TXT menggunakan AWS CLI untuk Route 53

1. Gunakan change-resource-record-setsperintah dengan `TXT-record.json` file contoh berikut:

```
{
  "Changes": [
    {
      "Action": "CREATE",
      "ResourceRecordSet": {
        "Name": "_11111aaaaaaaaa",
        "Type": "TXT",
        "ResourceRecords": [
          {
           "value": "vpc-lattice:1111aaaaaaa"
          }
        ]
      }
    }
  ]
}
```

2. Gunakan AWS CLI perintah berikut untuk menambahkan catatan TXT dari langkah sebelumnya ke zona yang dihosting Route 53:

```
aws route53 change-resource-record-sets \
  --hosted-zone-id ABCD123456 \
  --change-batch file://path/to/your/TXT-record.json
```

   Ganti `hosted-zone-id` dengan ID Zona Dihosting Route 53 dari zona yang dihosting di akun Anda. Nilai parameter change-batch menunjuk ke file JSON (TXT-Record.json) dalam folder (). path/to/your

Untuk memeriksa status verifikasi nama domain Anda, Anda dapat menggunakan konsol VPC Lattice atau perintah. `get-domain-verification`

Setelah Anda memverifikasi nama domain Anda, itu tetap diverifikasi sampai Anda menghapusnya. Jika Anda menghapus catatan TXT dari zona DNS Anda, VPC Lattice menghapus `verification-id` dan Anda perlu memverifikasi ulang nama domain. Jika Anda menghapus catatan TXT di zona DNS, VPC Lattice menetapkan status verifikasi nama domain Anda. `UNVERIFIED` Ini tidak memengaruhi titik akhir sumber daya yang ada, titik akhir jaringan layanan, atau asosiasi VPC jaringan layanan ke konfigurasi sumber daya Anda. Untuk memverifikasi ulang nama domain Anda, mulailah proses verifikasi nama domain.

Membuat konfigurasi sumber daya di VPC Lattice

Buat konfigurasi sumber daya.

Konsol Manajemen AWS

Untuk membuat konfigurasi sumber daya menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah PrivateLink dan Lattice, pilih Konfigurasi sumber daya.
3. Pilih Buat konfigurasi sumber daya.
4. Masukkan nama yang unik di AWS akun Anda. Anda tidak dapat mengubah nama ini setelah konfigurasi sumber daya dibuat.
5. Untuk jenis Konfigurasi, pilih Sumber daya untuk sumber daya tunggal atau turunan atau grup Sumber daya untuk grup sumber daya anak.
6. Pilih gateway sumber daya yang sebelumnya Anda buat atau buat sekarang.
7. (Opsional) Untuk memasukkan nama domain khusus, lakukan salah satu hal berikut:
   - Jika Anda memiliki konfigurasi sumber daya tipe tunggal, Anda dapat memasukkan nama domain khusus. Konsumen sumber daya dapat menggunakan nama domain ini untuk mengakses konfigurasi sumber daya Anda.
   - Jika Anda memiliki konfigurasi sumber daya tipe grup dan anak, Anda harus terlebih dahulu menentukan domain grup pada konfigurasi sumber daya grup. Selanjutnya, konfigurasi sumber daya anak dapat memiliki domain kustom yang merupakan subdomain dari domain grup.
8. (Opsional) Masukkan ID verifikasi.

   Berikan ID verifikasi jika Anda ingin nama domain Anda diverifikasi. Hal ini memungkinkan konsumen sumber daya tahu bahwa Anda memiliki nama domain.

9. Pilih pengenal sumber daya yang Anda inginkan untuk diwakili oleh konfigurasi sumber daya ini.
10. Pilih rentang port di mana Anda ingin berbagi sumber daya.
11. Untuk pengaturan Asosiasi, tentukan apakah konfigurasi sumber daya ini dapat dikaitkan dengan jaringan layanan yang dapat dibagikan.
12. Untuk konfigurasi sumber daya Bagikan, pilih pembagian sumber daya yang mengidentifikasi prinsipal yang dapat mengakses sumber daya ini.
13. (Opsional) Untuk Pemantauan, aktifkan log akses Sumber Daya dan tujuan pengiriman jika Anda ingin memantau permintaan dan tanggapan ke dan dari konfigurasi sumber daya.
14. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
15. Pilih Buat konfigurasi sumber daya.

AWS CLI

create-resource-configurationPerintah berikut membuat konfigurasi sumber daya tunggal dan mengaitkannya dengan nama `example.com` domain kustom.

```
aws vpc-lattice create-resource-configuration \
    --name my-resource-config \
    --type SINGLE \
    --resource-gateway-identifier rgw-0bba03f3d56060135 \
    --resource-configuration-definition 'ipResource={ipAddress=10.0.14.85}' \
    --custom-domain-name example.com \
    --verification-id dv-aaaa0000000111111
```

create-resource-configurationPerintah berikut membuat konfigurasi sumber daya grup dan mengaitkannya dengan nama `example.com` domain kustom.

```
aws vpc-lattice-custom-dns create-resource-configuration \
  --name my-custom-dns-resource-config-group \
  --type GROUP \
  --resource-gateway-identifier rgw-0bba03f3d56060135 \
  --domain-verification-identifier dv-aaaa0000000111111
```

create-resource-configurationPerintah berikut membuat konfigurasi sumber daya anak dan mengaitkannya dengan nama `child.example.com` domain kustom.

```
aws vpc-lattice-custom-dns create-resource-configuration \
  --name my-custom-dns-resource-config-child \
  --type CHILD \
  --resource-configuration-definition 'dnsResource={domainName=my-alb-123456789.us-
west-2.elb.amazonaws.com,ipAddressType=IPV4}' \
  --resource-configuration-group-identifier rcfg-07129f3acded87626 \
  --custom-domain-name child.example.com
```

Mengelola pengaitan konfigurasi sumber daya VPC Lattice

Akun konsumen tempat Anda berbagi konfigurasi sumber daya dan klien di akun Anda dapat mengakses konfigurasi sumber daya baik secara langsung menggunakan titik akhir VPC dari sumber daya tipe atau melalui titik akhir VPC dari jenis jaringan layanan. Akibatnya, konfigurasi sumber daya Anda akan memiliki asosiasi titik akhir dan asosiasi jaringan layanan.

Kelola asosiasi sumber daya jaringan layanan

Membuat atau menghapus asosiasi jaringan layanan.

> **Note**
>
> Jika Anda menerima pesan yang ditolak akses saat membuat asosiasi antara jaringan layanan dan konfigurasi sumber daya, periksa versi AWS RAM kebijakan Anda dan pastikan bahwa itu adalah versi 2. Untuk informasi selengkapnya, lihat panduan AWS RAM pengguna.

Untuk mengelola asosiasi layanan-jaringan menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah PrivateLink dan Lattice, pilih Konfigurasi sumber daya.
3. Pilih nama konfigurasi sumber daya untuk membuka halaman detailnya.
4. Pilih tab Asosiasi jaringan layanan.
5. Pilih Buat asosiasi.
6. Pilih jaringan layanan dari jaringan layanan VPC Lattice. Untuk membuat jaringan layanan, pilih Buat jaringan kisi VPC.
7. (Opsional) Untuk menambahkan tag, perluas tag asosiasi layanan, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.

8. (Opsional) Untuk mengaktifkan nama DNS pribadi untuk asosiasi sumber daya jaringan layanan ini pilih aktifkan nama DNS pribadi. Untuk informasi selengkapnya, lihat the section called “Nama domain khusus untuk pemilik jaringan layanan”.
9. Pilih Simpan perubahan.
10. Untuk menghapus asosiasi, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Hapus. Saat diminta konfirmasi, masukkan **`confirm`**, lalu pilih Hapus.

Untuk membuat asosiasi jaringan layanan menggunakan AWS CLI

Gunakan perintah create-service-network-resource-association.

Untuk menghapus asosiasi jaringan layanan menggunakan AWS CLI

Gunakan perintah delete-service-network-resource-association.

Kelola asosiasi titik akhir VPC sumber daya

Akun konsumen dengan akses ke konfigurasi sumber daya atau klien di akun Anda dapat mengakses konfigurasi sumber daya menggunakan titik akhir VPC sumber daya. Jika konfigurasi sumber daya Anda memiliki nama domain khusus, Anda dapat menggunakan aktifkan DNS pribadi untuk mengizinkan VPC Lattice menyediakan zona yang dihosting pribadi untuk titik akhir sumber daya atau titik akhir jaringan layanan Anda. Dengan ini, klien dapat langsung menggulung nama domain untuk mengakses konfigurasi sumber daya. Untuk informasi selengkapnya, lihat the section called “Nama domain khusus untuk konsumen sumber daya”.

Konsol Manajemen AWS

1. Untuk membuat asosiasi endpoint baru, buka PrivateLink dan Lattice di panel navigasi kiri dan pilih Endpoints.
2. Pilih Buat titik akhir.
3. Pilih konfigurasi sumber daya yang ingin Anda sambungkan ke VPC Anda.
4. Pilih VPC, subnet, dan grup keamanan.
5. (Opsional) Untuk mengaktifkan DNS pribadi dan mengkonfigurasi opsi DNS, pilih Aktifkan nama DNS pribadi.
6. (Opsional) Untuk menandai titik akhir VPC Anda, pilih Tambahkan tag baru, dan masukkan kunci tag dan nilai tag.
7. Pilih Buat titik akhir.

AWS CLI

create-vpc-endpointPerintah berikut membuat titik akhir VPC yang menggunakan DNS pribadi. Preferensi DNS pribadi diatur ke `VERIFIED_AND_SELECTED` dan domain yang dipilih adalah `example.com` dan. `example.org` VPC Lattice hanya menyediakan zona host pribadi untuk domain terverifikasi atau atau. `example.com` `example.org`

```
aws ec2 create-vpc-endpoint \
  --vpc-endpoint-type Resource \
  --vpc-id vpc-111122223333aabbc \
  --subnet-ids subnet-0011aabbcc2233445 \
  --resource-configuration-arn arn:aws:vpc-lattice:us-
west-2:111122223333:resourceconfiguration/rcfg-07129f3acded87625 \
  --private-dns-enabled \
  --private-dns-preferences VERIFIED_DOMAINS_AND_SPECIFIED_DOMAINS \
  --private-domains-set example.com, example.org
```

Untuk membuat asosiasi titik akhir VPC menggunakan AWS CLI

Gunakan perintah create-vpc-endpoint.

Untuk menghapus asosiasi titik akhir VPC menggunakan AWS CLI

Gunakan perintah delete-vpc-endpoint.

Membagikan entitas VPC Lattice Anda

Amazon VPC Lattice terintegrasi dengan AWS Resource Access Manager (AWS RAM) untuk memungkinkan berbagi layanan, konfigurasi sumber daya, dan jaringan layanan. AWS RAM adalah layanan yang memungkinkan Anda untuk berbagi beberapa entitas VPC Lattice dengan yang lain Akun AWS atau melalui. AWS Organizations Dengan AWS RAM, Anda berbagi entitas yang Anda miliki dengan membuat pembagian sumber daya. Pembagian sumber daya menentukan entitas yang akan dibagikan, dan konsumen yang akan dibagikan. Konsumen dapat mencakup:

- Khusus Akun AWS di dalam atau di luar organisasinya di AWS Organizations.
- Unit organisasi di dalam organisasinya di AWS Organizations.
- Seluruh organisasi di AWS Organizations

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

Daftar Isi

Prasyarat untuk berbagi entitas VPC Lattice

- Untuk berbagi entitas, Anda harus memilikinya di entitas Anda Akun AWS. Ini berarti bahwa entitas harus dialokasikan atau disediakan di akun Anda. Anda tidak dapat berbagi entitas yang telah dibagikan dengan Anda.
- Untuk berbagi entitas dengan organisasi Anda atau unit organisasi di AWS Organizations, Anda harus mengaktifkan berbagi dengan AWS Organizations. Untuk informasi selengkapnya, lihat Mengaktifkan berbagi sumber daya AWS Organizations di dalam Panduan AWS RAM Pengguna.

Bagikan entitas VPC Lattice

Untuk berbagi entitas, mulailah dengan membuat pembagian sumber daya menggunakan AWS Resource Access Manager. Pembagian sumber daya menentukan entitas yang akan dibagikan, konsumen dengan siapa mereka dibagikan, dan tindakan apa yang dapat dilakukan oleh prinsipal.

Saat Anda berbagi entitas VPC Lattice yang Anda miliki dengan orang lain Akun AWS, Anda mengaktifkan akun tersebut untuk mengaitkan entitas mereka dengan entitas di akun Anda. Saat Anda membuat asosiasi terhadap entitas bersama, kami membuat Nama Sumber Daya Amazon (ARN) di akun pemilik entitas dan di akun yang membuat asosiasi. Oleh karena itu, baik pemilik entitas maupun akun yang membuat asosiasi dapat menghapus asosiasi.

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

Pertimbangan-pertimbangan

- Anda dapat berbagi tiga jenis entitas VPC Lattice: jaringan layanan, layanan, dan konfigurasi sumber daya.
- Anda dapat membagikan entitas VPC Lattice Anda dengan entitas apa pun. Akun AWS
- Anda tidak dapat membagikan entitas VPC Lattice Anda dengan pengguna dan peran IAM individual.
- VPC Lattice mendukung izin yang dikelola pelanggan untuk layanan, konfigurasi sumber daya, dan jaringan layanan.

Untuk berbagi entitas yang Anda miliki menggunakan konsol VPC Lattice

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Services, Service networks, atau Resource configurations.
3. Pilih nama entitas untuk membuka halaman detailnya, lalu pilih Bagikan layanan, Bagikan jaringan layanan, atau Bagikan konfigurasi sumber daya dari tab Berbagi.
4. Pilih pembagian AWS RAM sumber daya dari pembagian Sumber Daya. Untuk membuat pembagian sumber daya, pilih Buat berbagi sumber daya di konsol RAM.
5. Pilih Bagikan layanan, Bagikan jaringan layanan, atau Bagikan konfigurasi sumber daya.

Untuk berbagi entitas yang Anda miliki menggunakan AWS RAM konsol

Gunakan prosedur yang dijelaskan dalam Buat berbagi sumber daya di Panduan AWS RAM Pengguna.

Untuk berbagi entitas yang Anda miliki menggunakan AWS CLI

Gunakan perintah associate-resource-share.

Berhenti berbagi entitas VPC Lattice

Untuk berhenti berbagi entitas VPC Lattice yang Anda miliki, Anda harus menghapusnya dari pembagian sumber daya. Asosiasi yang ada tetap ada setelah Anda berhenti membagikan entitas Anda. Asosiasi baru untuk entitas bersama sebelumnya tidak diizinkan. Ketika pemilik entitas atau pemilik asosiasi menghapus asosiasi, itu dihapus dari kedua akun. Jika pemilik akun ingin meninggalkan pembagian sumber daya, mereka harus meminta pemilik pembagian sumber daya untuk menghapus akun mereka dari daftar akun yang dibagikan sumber daya ini.

Untuk berhenti berbagi entitas yang Anda miliki menggunakan konsol VPC Lattice

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Services, Service networks, atau Resource configurations.
3. Pilih nama entitas untuk membuka halaman detailnya.
4. Pada tab Berbagi, pilih kotak centang untuk berbagi sumber daya dan kemudian pilih Hapus.

Untuk berhenti berbagi entitas yang Anda miliki menggunakan AWS RAM konsol

Lihat Memperbarui bagian sumber daya di Panduan AWS RAM Pengguna.

Untuk berhenti berbagi entitas yang Anda miliki menggunakan AWS CLI

Gunakan perintah disassociate-resource-share.

Tanggung jawab dan izin

Tanggung jawab dan izin berikut berlaku saat menggunakan entitas VPC Lattice bersama.

Pemilik entitas

- Pemilik jaringan layanan tidak dapat memodifikasi layanan yang dibuat oleh konsumen.
- Pemilik jaringan layanan tidak dapat menghapus layanan yang dibuat oleh konsumen.
- Pemilik jaringan layanan dapat menggambarkan semua asosiasi layanan untuk jaringan layanan.
- Pemilik jaringan layanan dapat memisahkan layanan apa pun yang terkait dengan jaringan layanan, terlepas dari siapa yang membuat asosiasi.
- Pemilik jaringan layanan dapat menggambarkan semua asosiasi VPC untuk jaringan layanan.
- Pemilik jaringan layanan dapat memisahkan VPC apa pun yang terkait dengan konsumen dengan jaringan layanan.
- Pemilik jaringan layanan dapat menjelaskan semua asosiasi konfigurasi sumber daya untuk jaringan layanan.
- Pemilik jaringan layanan dapat memisahkan konfigurasi sumber daya apa pun yang terkait dengan jaringan layanan, terlepas dari siapa yang membuat asosiasi.
- Pemilik jaringan layanan dapat menjelaskan semua asosiasi endpoint untuk jaringan layanan.
- Pemilik jaringan layanan dapat memisahkan titik akhir apa pun yang terkait dengan jaringan layanan, terlepas dari siapa yang membuat asosiasi.
- Pemilik layanan dapat menggambarkan semua asosiasi jaringan layanan dengan layanan.
- Pemilik layanan dapat memisahkan layanan dari jaringan layanan apa pun yang terkait dengannya.
- Pemilik konfigurasi sumber daya dapat menjelaskan semua asosiasi jaringan dengan konfigurasi sumber daya.
- Pemilik konfigurasi sumber daya dapat memisahkan konfigurasi sumber daya dari jaringan layanan apa pun yang terkait dengannya.
- Pemilik titik akhir VPC dapat menggambarkan jaringan layanan yang terkait dengannya.
- Pemilik titik akhir VPC dapat memisahkan titik akhir dari jaringan layanan.
- Hanya akun yang membuat asosiasi yang dapat memperbarui hubungan antara jaringan layanan dan VPC.

Konsumen entitas

- Konsumen tidak dapat menghapus konfigurasi layanan atau sumber daya yang tidak mereka buat.
- Konsumen hanya dapat memisahkan layanan atau konfigurasi sumber daya yang mereka kaitkan dengan jaringan layanan.

- Konsumen dan pemilik jaringan dapat menggambarkan semua asosiasi antara jaringan layanan dan konfigurasi layanan atau sumber daya.
- Konsumen tidak dapat mengambil informasi layanan dari layanan atau informasi konfigurasi sumber daya dari konfigurasi sumber daya yang tidak mereka miliki.
- Konsumen dapat menggambarkan semua asosiasi layanan dan asosiasi konfigurasi sumber daya dengan jaringan layanan bersama.
- Konsumen dapat mengaitkan layanan atau konfigurasi sumber daya dengan jaringan layanan bersama.
- Konsumen dapat melihat semua asosiasi VPC dengan jaringan layanan bersama.
- Konsumen dapat mengaitkan VPC dengan jaringan layanan bersama.
- Konsumen hanya dapat memisahkan VPCs yang mereka kaitkan dengan jaringan layanan.
- Konsumen dapat membuat endpoint VPC jaringan layanan untuk menghubungkan VPC mereka ke jaringan layanan bersama.
- Konsumen hanya dapat menghapus titik akhir VPC jaringan layanan yang mereka buat untuk menghubungkan VPC mereka ke jaringan layanan bersama.
- Konsumen layanan bersama tidak dapat mengaitkan layanan dengan jaringan layanan yang tidak mereka miliki.
- Konsumen jaringan layanan bersama tidak dapat mengaitkan VPC atau layanan yang tidak mereka miliki.
- Konsumen konfigurasi sumber daya bersama tidak dapat mengaitkan konfigurasi sumber daya dengan jaringan layanan yang tidak mereka miliki.
- Konsumen jaringan layanan bersama tidak dapat mengaitkan konfigurasi VPC atau layanan atau sumber daya yang tidak mereka miliki.
- Konsumen dapat menggambarkan layanan, jaringan layanan, atau konfigurasi sumber daya yang dibagikan dengan mereka.
- Konsumen tidak dapat mengaitkan dua entitas jika keduanya dibagikan dengan mereka.

Acara lintas akun

Ketika pemilik entitas dan konsumen melakukan tindakan pada entitas bersama, tindakan tersebut dicatat sebagai peristiwa lintas akun di AWS CloudTrail.

`CreateServiceNetworkResourceAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil CreateServiceNetworkResourceAssociation dengan entitas bersama. Jika pemanggil memiliki konfigurasi sumber daya, acara dikirim ke pemilik jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik konfigurasi sumber daya.

`CreateServiceNetworkServiceAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil CreateServiceNetworkServiceAssociationdengan entitas bersama. Jika penelepon memiliki layanan, acara dikirim ke pemilik jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik layanan.

`CreateServiceNetworkVpcAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil CreateServiceNetworkVpcAssociationdengan jaringan layanan bersama.

`DeleteServiceNetworkResourceAssociationByOwner`

Dikirim ke pemilik asosiasi saat pemilik entitas memanggil DeleteServiceNetworkResourceAssociation dengan entitas bersama. Jika pemanggil memiliki konfigurasi sumber daya, acara dikirim ke pemilik asosiasi jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik asosiasi sumber daya.

`DeleteServiceNetworkResourceAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil DeleteServiceNetworkResourceAssociation dengan entitas bersama. Jika pemanggil memiliki konfigurasi sumber daya, acara dikirim ke pemilik jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik konfigurasi sumber daya.

`DeleteServiceNetworkServiceAssociationByOwner`

Dikirim ke pemilik asosiasi saat pemilik entitas memanggil DeleteServiceNetworkServiceAssociationdengan entitas bersama. Jika penelepon memiliki layanan, acara dikirim ke pemilik asosiasi jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik asosiasi layanan.

`DeleteServiceNetworkServiceAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil DeleteServiceNetworkServiceAssociationdengan entitas bersama. Jika penelepon memiliki layanan, acara dikirim ke pemilik jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik layanan.

`DeleteServiceNetworkVpcAssociationByOwner`

Dikirim ke pemilik asosiasi saat pemilik entitas memanggil DeleteServiceNetworkVpcAssociationdengan jaringan layanan bersama.

`DeleteServiceNetworkVpcAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil DeleteServiceNetworkVpcAssociationdengan jaringan layanan bersama.

`GetServiceBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil GetServicedengan layanan bersama.

`GetServiceNetworkBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil GetServiceNetworkdengan jaringan layanan bersama.

`GetServiceNetworkResourceAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil GetServiceNetworkResourceAssociation dengan entitas bersama. Jika pemanggil memiliki konfigurasi sumber daya, acara dikirim ke pemilik jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik konfigurasi sumber daya.

`GetServiceNetworkServiceAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil GetServiceNetworkServiceAssociationdengan entitas bersama. Jika penelepon memiliki layanan, acara dikirim ke pemilik jaringan layanan. Jika penelepon memiliki jaringan layanan, acara dikirim ke pemilik layanan.

`GetServiceNetworkVpcAssociationBySharee`

Dikirim ke pemilik entitas saat konsumen entitas memanggil GetServiceNetworkVpcAssociationdengan jaringan layanan bersama.

Berikut ini adalah contoh entri untuk `CreateServiceNetworkServiceAssociationBySharee` acara tersebut.


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "Unknown"
    },
    "eventTime": "2023-04-27T17:12:46Z",
    "eventSource": "vpc-lattice.amazonaws.com",
    "eventName": "CreateServiceNetworkServiceAssociationBySharee",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "vpc-lattice.amazonaws.com",
    "userAgent": "ec2.amazonaws.com",
    "requestParameters": null,
    "responseElements": null,
    "additionalEventData": {
        "callerAccountId": "111122223333"
    },
    "requestID": "ddabb0a7-70c6-4f70-a6c9-00cbe8a6a18b",
    "eventID": "bd03cdca-7edd-4d50-b9c9-eaa89f4a47cd",
    "readOnly": false,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::VpcLattice::ServiceNetworkServiceAssociation",
            "ARN": "arn:aws:vpc-
lattice:region:123456789012:servicenetworkserviceassociation/snsa-0d5ea7bc72EXAMPLE"
        }
    ],
    "eventType": "AwsServiceEvent",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Kisi VPC untuk Oracle Database@AWS

VPC Lattice mendukung integrasi layanan AWS terkelola untuk Oracle Database@AWS(ODB) dan memberi Anda konektivitas yang disederhanakan antara jaringan ODB, dan di lokasi. AWS VPCs Untuk mendukung konektivitas ini, VPC Lattice menyediakan entitas berikut atas nama Anda:

Jaringan layanan default

Jaringan layanan default menggunakan konvensi penamaan `default-odb-network-randomHash`

Titik akhir jaringan layanan default

Tidak ada nama untuk AWS sumber daya ini.

Gateway sumber daya

Gateway sumber daya menggunakan konvensi penamaan `default-odb-network-randomHash`

VPC Lattice mendukung integrasi layanan AWS terkelola, yang disebut sebagai integrasi terkelola ke jaringan ODB Anda. Secara default, Oracle Cloud Infrastructure (OCI) Managed Backup ke Amazon S3 diaktifkan. Anda dapat memilih untuk mengaktifkan akses yang dikelola sendiri ke Amazon S3 dan Nol-ETL.

Setelah Anda membuat jaringan ODB Anda, Anda dapat melihat sumber daya yang disediakan menggunakan atau. Konsol Manajemen AWS AWS CLI Contoh perintah berikut mencantumkan integrasi terkelola default jaringan ODB dan sumber daya lain yang mungkin Anda miliki untuk jaringan layanan ini:

```
aws vpc-lattice list-service-network-resource-associations \
        --service-network-identifier default-odb-network-randomHash
```

Pertimbangan-pertimbangan

Pertimbangan berikut berlaku untuk VPC Lattice untuk: Oracle Database@AWS

- Anda tidak dapat menghapus jaringan layanan default, titik akhir jaringan layanan, gateway sumber daya, atau integrasi terkelola ODB apa pun yang disediakan oleh VPC Lattice. Untuk menghapus entitas ini, hapus jaringan ODB Anda atau nonaktifkan integrasi terkelola.

- Klien hanya dapat mengakses integrasi terkelola dalam jaringan ODB. Klien di luar jaringan ODB, seperti di jaringan Anda VPCs, tidak dapat menggunakan integrasi terkelola ini untuk mengakses S3 atau Zero-ETL.
- Anda tidak dapat terhubung ke salah satu integrasi terkelola di luar jaringan ODB yang disediakan oleh VPC Lattice.
- Semua lalu lintas ke Amazon S3 melewati titik akhir jaringan layanan default dan biaya pemrosesan standar untuk mengakses sumber daya berlaku. Semua lalu lintas nol-ETL melewati gateway sumber daya dan biaya pemrosesan data standar untuk sumber daya yang Anda bagikan berlaku. Untuk informasi selengkapnya, lihat harga VPC Lattice.
- Tidak ada biaya per jam untuk integrasi Oracle Database@AWS terkelola.
- Anda dapat mengelola sumber daya yang disediakan oleh VPC Lattice sama seperti jaringan layanan lainnya. Anda dapat berbagi jaringan layanan default dengan organisasi lain Akun AWS , dan menambahkan titik akhir baru, asosiasi VPC, layanan VPC Lattice, dan sumber daya ke jaringan default.
- Izin berikut diperlukan untuk VPC Lattice untuk menyediakan sumber daya: Oracle Database@AWS

  JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
     {
         "Sid": "AllowODBEC2andLatticeActions",
         "Action": [
             "ec2:DescribeVpcs",
             "ec2:CreateTags",
             "ec2:DescribeAvailabilityZones",
             "ec2:CreateOdbNetworkPeering",
             "ec2:DeleteOdbNetworkPeering",
             "ec2:ModifyOdbNetworkPeering",
             "ec2:DescribeVpcEndpointAssociations",
             "ec2:CreateVpcEndpoint",
             "ec2:DeleteVpcEndpoints",
             "ec2:DescribeVpcEndpoints",
             "vpc-lattice:CreateServiceNetwork",
             "vpc-lattice:DeleteServiceNetwork",
             "vpc-lattice:GetServiceNetwork",
             "vpc-lattice:CreateServiceNetworkResourceAssociation",
             "vpc-lattice:DeleteServiceNetworkResourceAssociation",
```

```
            "vpc-lattice:GetServiceNetworkResourceAssociation",
            "vpc-lattice:CreateResourceGateway",
            "vpc-lattice:DeleteResourceGateway",
            "vpc-lattice:GetResourceGateway",
            "vpc-lattice:CreateServiceNetworkVpcEndpointAssociation"
        ],
        "Effect": "Allow",
        "Resource": "*"
    },
    {
  "Sid": "AllowSLRActionsForLattice",
  "Effect": "Allow",
  "Action": [
   "iam:CreateServiceLinkedRole"
  ],
  "Resource": "*",
   "Condition": {
    "StringEquals": {
     "iam:AWSServiceName": [
      "vpc-lattice.amazonaws.com"
     ]
    }
   }
 }
  ]
}
```

Untuk menggunakan VPC Lattice Oracle Database@AWS, kami menyarankan agar Anda terbiasa dengan jaringan layanan, asosiasi jaringan layanan, dan gateway sumber daya di VPC Lattice.

Topik

- the section called “Cadangan Terkelola Oracle Cloud Infrastructure (OCI) ke Amazon S3”
- the section called “Akses Amazon S3”
- the section called “Nol-ETL untuk Amazon Redshift”
- the section called “Akses dan bagikan entitas VPC Lattice”

Cadangan Terkelola Oracle Cloud Infrastructure (OCI) ke Amazon S3

Saat Anda membuat Oracle Database@AWS database, VPC Lattice membuat konfigurasi sumber daya yang disebut. `odb-managed-s3-backup-access` Konfigurasi sumber daya ini mewakili cadangan database Anda yang dikelola OCI ke Amazon S3 dan hanya memungkinkan konektivitas ke bucket Amazon S3 yang dimiliki oleh OCI. Lalu lintas antara ODB Network dan S3 tidak pernah meninggalkan jaringan Amazon.

Akses Amazon S3

Selain Cadangan Terkelola OCI ke Amazon S3, Anda dapat membuat integrasi terkelola yang memungkinkan akses ke Amazon S3 dari jaringan ODB. Saat Anda memodifikasi Oracle Database@AWS jaringan untuk mengaktifkan integrasi terkelola Amazon S3 Access, VPC Lattice menyediakan konfigurasi sumber daya yang disebut `odb-s3-access` dalam jaringan layanan default. Anda dapat menggunakan integrasi ini untuk mengakses Amazon S3 untuk kebutuhan Anda sendiri termasuk pencadangan atau pemulihan yang dikelola sendiri. Anda dapat menetapkan kontrol perimeter dengan memberikan kebijakan autentikasi.

Pertimbangan-pertimbangan

Berikut ini adalah pertimbangan untuk integrasi terkelola Amazon S3 Access:

- Anda hanya dapat membuat satu integrasi terkelola Amazon S3 Access untuk jaringan ODB.
- Integrasi terkelola ini memungkinkan akses ke Amazon S3 dari jaringan ODB saja, dan bukan dari asosiasi VPC lain atau titik akhir jaringan layanan di jaringan layanan default.
- Anda tidak dapat mengakses bucket S3 di Wilayah yang berbeda AWS .

Aktifkan integrasi terkelola Amazon S3 Access

Gunakan perintah berikut untuk mengaktifkan integrasi terkelola Amazon S3 Access:

```
aws odb update-odb-network \
  --odb-network-id odb-network-id \
  --s3-access ENABLED
```

Akses aman dengan kebijakan autentikasi

Anda dapat mengamankan akses ke bucket S3 dengan menentukan kebijakan autentikasi menggunakan ODB API. Contoh kebijakan berikut memberikan akses ke bucket S3 tertentu yang dimiliki oleh organisasi tertentu.

JSON

```
{
  "Version":"2012-10-17",
  "Id": "Policy1515115909152",
  "Statement": [
    {
      "Sid": "GrantAccessToMyOrgS3",
      "Principal": "*",
      "Action": "s3:*",
      "Effect": "Deny",
      "Resource": [
        "arn:aws:s3:::awsexamplebucket1",
        "arn:aws:s3:::awsexamplebucket1/*"
      ],
      "Condition": {
        "StringNotEquals": {
          "aws:ResourceOrgID": "o-abcd1234"
        }
      }
    }
  ]
}
```

> **Note**
>
> Kunci`aws:SourceVpc`,`aws:SourceVpce`, dan `aws:VpcSourceIp` kondisi tidak didukung untuk kebijakan bucket S3 saat menggunakan integrasi terkelola ODB.

Nol-ETL untuk Amazon Redshift

Anda dapat menggunakan jaringan layanan yang disediakan oleh VPC Lattice untuk mengaktifkan nol-ETL. Integrasi terkelola ini menghubungkan database jaringan ODB Anda ke Amazon Redshift untuk membantu menganalisis data di berbagai basis data. Anda dapat memulai penyiapan nol-ETL menggunakan AWS Glue integrasi APIs dan menggunakan ODB APIs untuk mengaktifkan integrasi terkelola dan mengatur jalur jaringan. Untuk informasi selengkapnya, lihat Integrasi nol-ETL dengan Amazon Redshift.

Pertimbangan-pertimbangan

Berikut ini adalah pertimbangan untuk integrasi nol-ETL terkelola:

- Jika Anda mengaktifkan integrasi nol-ETL terkelola, Anda hanya dapat menggunakan nol-ETL untuk mengakses instance di jaringan ODB Anda. Layanan dan sumber daya lain yang terkait dengan jaringan layanan Anda diisolasi dari nol-ETL.

Akses dan bagikan entitas VPC Lattice

Anda juga dapat menghubungkan jaringan ODB Anda ke layanan, sumber daya, dan klien lain dalam VPCs menggunakan VPC Lattice. Opsi konektivitas ini didukung melalui jaringan layanan default, gateway sumber daya, dan titik akhir jaringan layanan yang disediakan oleh VPC Lattice.

Akses layanan dan sumber daya VPC Lattice

Untuk mengakses entitas lain, layanan asosiasi atau sumber daya yang Anda miliki, atau dibagikan dengan Anda, ke jaringan layanan default. Klien dalam jaringan ODB dapat mengakses layanan atau sumber daya melalui endpoint jaringan layanan default.

Pertimbangan-pertimbangan

Berikut ini adalah pertimbangan untuk menghubungkan ke entitas VPC Lattice lainnya:

- Anda dapat menambahkan titik akhir jaringan layanan baru, asosiasi VPC, sumber daya VPC Lattice, dan layanan ke jaringan layanan, tetapi Anda tidak dapat memodifikasi sumber daya yang disediakan oleh VPC Lattice atas nama jaringan ODB. Ini harus dikelola melalui Oracle Database@AWS APIs.

Bagikan jaringan ODB Anda melalui VPC Lattice

Anda dapat membagikan sumber daya jaringan ODB Anda dengan klien di akun lain VPCs, akun, atau di tempat. Untuk memulai, buat konfigurasi sumber daya untuk sumber daya yang ingin Anda bagikan. Konfigurasi sumber daya harus menggunakan gateway sumber daya default untuk jaringan ODB Anda. Anda kemudian dapat mengaitkan sumber daya dengan jaringan layanan default Anda.

Klien di jaringan lain VPCs atau Akun AWS yang telah Anda bagikan dengan jaringan layanan Anda dapat mengakses sumber daya ini melalui titik akhir jaringan layanan mereka sendiri atau asosiasi VPC. Untuk informasi selengkapnya, lihat the section called “Kelola asosiasi”.

Pertimbangan

Berikut ini adalah pertimbangan untuk berbagi jaringan ODB Anda:

- Kami merekomendasikan hanya berbagi instance jaringan ODB sebagai sumber daya berbasis IP.
- VPC Lattice tidak mendukung DNS pendengar Nama Akses Klien Tunggal (SCAN) OCI.

Keamanan di Amazon VPC Lattice

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.

Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Model tanggung jawab bersama menjelaskan hal ini sebagai keamanan dari cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program AWS Kepatuhan Program AWS Kepatuhan . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon VPC Lattice, lihat AWS Layanan dalam Lingkup menurut Program Kepatuhan dalam Lingkup oleh Program Kepatuhan.
- Keamanan di cloud — Anda bertanggung jawab untuk menjaga kontrol atas konten Anda yang di-host di infrastruktur ini. 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 VPC Lattice. Topik berikut menunjukkan cara mengonfigurasi Kisi VPC untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain, yang membantu Anda memantau dan mengamankan layanan VPC Lattice, jaringan layanan, dan konfigurasi sumber daya Anda.

Daftar Isi

Mengelola akses ke layanan VPC Lattice

VPC Lattice aman secara default karena Anda harus eksplisit tentang layanan dan konfigurasi sumber daya untuk menyediakan akses ke dan yang dengannya. VPCs Anda dapat mengakses layanan melalui asosiasi VPC atau titik akhir VPC dari jaringan layanan tipe. Untuk skenario multi-akun, Anda dapat menggunakan AWS Resource Access Manageruntuk berbagi layanan, konfigurasi sumber daya, dan jaringan layanan di seluruh batas akun.

VPC Lattice menyediakan kerangka kerja yang memungkinkan Anda menerapkan defense-in-depth strategi di beberapa lapisan jaringan.

- Lapisan pertama — Asosiasi titik akhir layanan, sumber daya, VPC, dan VPC dengan jaringan layanan. VPC dapat terhubung ke jaringan layanan baik melalui asosiasi atau melalui titik akhir VPC. Jika VPC tidak terhubung ke jaringan layanan, klien di VPC tidak dapat mengakses konfigurasi layanan dan sumber daya yang terkait dengan jaringan layanan.
- Lapisan kedua — Perlindungan keamanan tingkat jaringan opsional untuk jaringan layanan, seperti grup keamanan dan jaringan. ACLs Dengan menggunakan ini, Anda dapat mengizinkan akses ke grup klien tertentu dalam VPC, bukan semua klien di VPC.
- Lapisan ketiga - Kebijakan autentikasi VPC Lattice opsional. Anda dapat menerapkan kebijakan autentikasi ke jaringan layanan dan layanan individual. Biasanya, kebijakan autentikasi pada jaringan layanan dioperasikan oleh administrator jaringan atau cloud, dan mereka menerapkan otorisasi kasar. Misalnya, hanya mengizinkan permintaan yang diautentikasi dari organisasi tertentu di AWS Organizations. Untuk kebijakan autentikasi di tingkat layanan, biasanya pemilik layanan menetapkan kontrol berbutir halus, yang mungkin lebih ketat daripada otorisasi kasar yang diterapkan di tingkat jaringan layanan.

> **Note**
>
> Kebijakan autentikasi pada jaringan layanan tidak berlaku untuk konfigurasi sumber daya di jaringan layanan.

Metode kontrol akses

- Kebijakan autentikasi
- Grup keamanan
- Jaringan ACLs

Kontrol akses ke layanan VPC Lattice menggunakan kebijakan autentikasi

Kebijakan autentikasi VPC Lattice adalah dokumen kebijakan IAM yang Anda lampirkan ke jaringan layanan atau layanan untuk mengontrol apakah prinsipal tertentu memiliki akses ke grup layanan atau layanan tertentu. Anda dapat melampirkan satu kebijakan autentikasi ke setiap jaringan layanan atau layanan yang ingin Anda kontrol aksesnya.

> **Note**
>
> Kebijakan autentikasi pada jaringan layanan tidak berlaku untuk konfigurasi sumber daya di jaringan layanan.

Kebijakan autentikasi berbeda dari kebijakan berbasis identitas IAM. Kebijakan berbasis identitas IAM dilampirkan ke pengguna, grup, atau peran IAM dan menentukan tindakan apa yang dapat dilakukan identitas tersebut pada sumber daya mana. Kebijakan autentikasi dilampirkan ke layanan dan jaringan layanan. Agar otorisasi berhasil, kebijakan autentikasi dan kebijakan berbasis identitas harus memiliki pernyataan izin eksplisit. Untuk informasi selengkapnya, lihat Cara kerja otorisasi.

Anda dapat menggunakan AWS CLI dan konsol untuk melihat, menambah, memperbarui, atau menghapus kebijakan autentikasi pada layanan dan jaringan layanan. Saat Anda menambahkan, memperbarui, atau menghapus kebijakan autentikasi, mungkin perlu beberapa menit untuk siap. Saat menggunakan AWS CLI, pastikan Anda berada di Wilayah yang benar. Anda dapat mengubah Wilayah default untuk profil Anda, atau menggunakan `--region` parameter dengan perintah.

Daftar Isi

Untuk memulai kebijakan autentikasi, ikuti prosedur untuk membuat kebijakan autentikasi yang berlaku untuk jaringan layanan. Untuk izin yang lebih ketat yang tidak ingin diterapkan ke layanan lain, Anda dapat secara opsional menetapkan kebijakan autentikasi pada layanan individual.

Mengelola akses ke jaringan layanan dengan kebijakan autentikasi

AWS CLI Tugas berikut menunjukkan cara mengelola akses ke jaringan layanan menggunakan kebijakan autentikasi. Untuk petunjuk yang menggunakan konsol, lihatJaringan layanan di VPC Lattice.

Tugas

- Menambahkan kebijakan autentikasi ke jaringan layanan
- Mengubah jenis autentikasi jaringan layanan
- Menghapus kebijakan autentikasi dari jaringan layanan

Menambahkan kebijakan autentikasi ke jaringan layanan

Ikuti langkah-langkah di bagian ini untuk menggunakan AWS CLI to:

- Aktifkan kontrol akses pada jaringan layanan menggunakan IAM.
- Tambahkan kebijakan autentikasi ke jaringan layanan. Jika Anda tidak menambahkan kebijakan autentikasi, semua lalu lintas akan mendapatkan kesalahan akses ditolak.

Untuk mengaktifkan kontrol akses dan menambahkan kebijakan autentikasi ke jaringan layanan baru

1. Untuk mengaktifkan kontrol akses pada jaringan layanan sehingga dapat menggunakan kebijakan autentikasi, gunakan create-service-network perintah dengan `--auth-type` opsi dan nilai`AWS_IAM`.

   ```
   aws vpc-lattice create-service-network --name Name --auth-type AWS_IAM [--
   tags TagSpecification]
   ```

   Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

   ```
   {
      "arn": "arn",
      "authType": "AWS_IAM",
      "id": "sn-0123456789abcdef0",
   ```

```
   "name": "Name"
}
```

2. Gunakan put-auth-policy perintah, tentukan ID jaringan layanan tempat Anda ingin menambahkan kebijakan autentikasi dan kebijakan autentikasi yang ingin Anda tambahkan.

   Misalnya, gunakan perintah berikut untuk membuat kebijakan autentikasi untuk jaringan layanan dengan ID*sn-0123456789abcdef0*.

   ```
   aws vpc-lattice put-auth-policy --resource-identifier sn-0123456789abcdef0 --
   policy file://policy.json
   ```

   Gunakan JSON untuk membuat definisi kebijakan. Untuk informasi selengkapnya, lihat Elemen umum dalam kebijakan autentikasi.

   Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut ini.

   ```
   {
      "policy": "policy",
      "state": "Active"
   }
   ```

Untuk mengaktifkan kontrol akses dan menambahkan kebijakan autentikasi ke jaringan layanan yang ada

1. Untuk mengaktifkan kontrol akses pada jaringan layanan sehingga dapat menggunakan kebijakan autentikasi, gunakan update-service-network perintah dengan `--auth-type` opsi dan nilai`AWS_IAM`.

   ```
   aws vpc-lattice update-service-network --service-network-
   identifier sn-0123456789abcdef0 --auth-type AWS_IAM
   ```

   Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

   ```
   {
      "arn": "arn",
      "authType": "AWS_IAM",
      "id": "sn-0123456789abcdef0",
      "name": "Name"
   ```

```
}
```

2. Gunakan put-auth-policy perintah, tentukan ID jaringan layanan tempat Anda ingin menambahkan kebijakan autentikasi dan kebijakan autentikasi yang ingin Anda tambahkan.

```
aws vpc-lattice put-auth-policy --resource-identifier sn-0123456789abcdef0 --
policy file://policy.json
```

Gunakan JSON untuk membuat definisi kebijakan. Untuk informasi selengkapnya, lihat Elemen umum dalam kebijakan autentikasi.

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
{
   "policy": "policy",
   "state": "Active"
}
```

Mengubah jenis autentikasi jaringan layanan

Untuk menonaktifkan kebijakan autentikasi untuk jaringan layanan

Gunakan update-service-network perintah dengan `--auth-type` opsi dan nilai`NONE`.

```
aws vpc-lattice update-service-network --service-network-
identifier sn-0123456789abcdef0 --auth-type NONE
```

Jika Anda perlu mengaktifkan kebijakan autentikasi lagi nanti, jalankan perintah ini dengan `AWS_IAM` ditentukan untuk `--auth-type` opsi.

Menghapus kebijakan autentikasi dari jaringan layanan

Untuk menghapus kebijakan autentikasi dari jaringan layanan

Gunakan perintah delete-auth-policy.

```
aws vpc-lattice delete-auth-policy --resource-identifier sn-0123456789abcdef0
```

Permintaan gagal jika Anda menghapus kebijakan autentikasi sebelum mengubah jenis autentikasi jaringan layanan menjadi. `NONE`

Mengelola akses ke layanan dengan kebijakan autentikasi

AWS CLI Tugas berikut menunjukkan cara mengelola akses ke layanan menggunakan kebijakan autentikasi. Untuk petunjuk yang menggunakan konsol, lihatLayanan di VPC Lattice.

Tugas

- Menambahkan kebijakan autentikasi ke layanan
- Mengubah jenis autentikasi layanan
- Menghapus kebijakan autentikasi dari layanan

Menambahkan kebijakan autentikasi ke layanan

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk:

- Aktifkan kontrol akses pada layanan menggunakan IAM.
- Tambahkan kebijakan autentikasi ke layanan. Jika Anda tidak menambahkan kebijakan autentikasi, semua lalu lintas akan mendapatkan kesalahan akses ditolak.

Untuk mengaktifkan kontrol akses dan menambahkan kebijakan autentikasi ke layanan baru

1. Untuk mengaktifkan kontrol akses pada layanan sehingga dapat menggunakan kebijakan autentikasi, gunakan create-service perintah dengan `--auth-type` opsi dan nilai`AWS_IAM`.

   ```
   aws vpc-lattice create-service --name Name --auth-type AWS_IAM [--
   tags TagSpecification]
   ```

   Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

   ```
   {
      "arn": "arn",
      "authType": "AWS_IAM",
      "dnsEntry": {
         ...
      },
      "id": "svc-0123456789abcdef0",
      "name": "Name",
      "status": "CREATE_IN_PROGRESS"
   }
   ```

2. Gunakan put-auth-policy perintah, tentukan ID layanan tempat Anda ingin menambahkan kebijakan autentikasi dan kebijakan autentikasi yang ingin Anda tambahkan.

   Misalnya, gunakan perintah berikut untuk membuat kebijakan autentikasi untuk layanan dengan ID*svc-0123456789abcdef0*.

   ```
   aws vpc-lattice put-auth-policy --resource-identifier svc-0123456789abcdef0 --
   policy file://policy.json
   ```

   Gunakan JSON untuk membuat definisi kebijakan. Untuk informasi selengkapnya, lihat Elemen umum dalam kebijakan autentikasi.

   Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut ini.

   ```
   {
      "policy": "policy",
      "state": "Active"
   }
   ```

Untuk mengaktifkan kontrol akses dan menambahkan kebijakan autentikasi ke layanan yang ada

1. Untuk mengaktifkan kontrol akses pada layanan sehingga dapat menggunakan kebijakan autentikasi, gunakan update-service perintah dengan `--auth-type` opsi dan nilai`AWS_IAM`.

   ```
   aws vpc-lattice update-service --service-identifier svc-0123456789abcdef0 --auth-
   type AWS_IAM
   ```

   Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

   ```
   {
      "arn": "arn",
      "authType": "AWS_IAM",
      "id": "svc-0123456789abcdef0",
      "name": "Name"
   }
   ```

2. Gunakan put-auth-policy perintah, tentukan ID layanan tempat Anda ingin menambahkan kebijakan autentikasi dan kebijakan autentikasi yang ingin Anda tambahkan.

```
aws vpc-lattice put-auth-policy --resource-identifier svc-0123456789abcdef0 --
policy file://policy.json
```

Gunakan JSON untuk membuat definisi kebijakan. Untuk informasi selengkapnya, lihat Elemen umum dalam kebijakan autentikasi.

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut ini.

```
{
   "policy": "policy",
   "state": "Active"
}
```

Mengubah jenis autentikasi layanan

Untuk menonaktifkan kebijakan autentikasi untuk layanan

Gunakan update-service perintah dengan `--auth-type` opsi dan nilai`NONE`.

```
aws vpc-lattice update-service --service-identifier svc-0123456789abcdef0 --auth-type
 NONE
```

Jika Anda perlu mengaktifkan kebijakan autentikasi lagi nanti, jalankan perintah ini dengan `AWS_IAM` ditentukan untuk `--auth-type` opsi.

Menghapus kebijakan autentikasi dari layanan

Untuk menghapus kebijakan autentikasi dari layanan

Gunakan perintah delete-auth-policy.

```
aws vpc-lattice delete-auth-policy --resource-identifier svc-0123456789abcdef0
```

Permintaan gagal jika Anda menghapus kebijakan autentikasi sebelum mengubah jenis autentikasi layanan menjadi. `NONE`

Jika Anda mengaktifkan kebijakan autentikasi yang memerlukan permintaan terautentikasi ke layanan, permintaan apa pun ke layanan tersebut harus berisi tanda tangan permintaan yang valid

yang dihitung menggunakan Sigv4 (SigV4). Untuk informasi selengkapnya, lihat SIGv4 permintaan yang diautentikasi untuk Amazon VPC Lattice.

Elemen umum dalam kebijakan autentikasi

Kebijakan autentikasi VPC Lattice ditentukan menggunakan sintaks yang sama dengan kebijakan IAM. Untuk informasi selengkapnya, lihat Kebijakan berbasis identitas dan kebijakan berbasis sumber daya di Panduan Pengguna IAM.

Kebijakan autentikasi berisi elemen-elemen berikut:

- Kepala Sekolah — Orang atau aplikasi yang diizinkan mengakses tindakan dan sumber daya dalam pernyataan. Dalam kebijakan autentikasi, prinsipal adalah entitas IAM yang merupakan penerima izin ini. Prinsipal diautentikasi sebagai entitas IAM untuk membuat permintaan ke sumber daya tertentu, atau kelompok sumber daya seperti dalam kasus layanan dalam jaringan layanan.

  Anda harus menentukan principal dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau layanan. AWS Untuk informasi selengkapnya, lihat elemen kebijakan AWS JSON: Principal dalam Panduan Pengguna IAM.
- Efek — Efek ketika prinsipal yang ditentukan meminta tindakan spesifik. Ini bisa salah satu `Allow` atau`Deny`. Secara default, ketika Anda mengaktifkan kontrol akses pada layanan atau jaringan layanan menggunakan IAM, prinsipal tidak memiliki izin untuk membuat permintaan ke jaringan layanan atau layanan.
- Tindakan — Tindakan API spesifik yang Anda berikan atau tolak izinnya. VPC Lattice mendukung tindakan yang menggunakan awalan. `vpc-lattice-svcs` Untuk informasi selengkapnya, lihat Tindakan yang ditentukan oleh Amazon VPC Lattice Services di Referensi Otorisasi Layanan.
- Sumber Daya — Layanan yang dipengaruhi oleh tindakan.
- Kondisi - Kondisi bersifat opsional. Anda dapat menggunakannya untuk mengontrol kapan kebijakan Anda berlaku. Untuk informasi selengkapnya, lihat Kunci kondisi untuk Layanan Kisi VPC Amazon di Referensi Otorisasi Layanan.

Saat Anda membuat dan mengelola kebijakan autentikasi, Anda mungkin ingin menggunakan IAM Policy Generator.

Persyaratan

Kebijakan di JSON tidak boleh berisi baris baru atau baris kosong.

Format sumber daya untuk kebijakan autentikasi

Anda dapat membatasi akses ke sumber daya tertentu dengan membuat kebijakan autentikasi yang menggunakan skema yang cocok dengan `<serviceARN>/<path>` pola dan kode `Resource` elemen seperti yang ditunjukkan pada contoh berikut.

Protokol	Contoh
HTTP	• `"Resource": "arn:aws:vpc-lattice:us-west-2:1234567890:service/svc-0123456789abcdef0/rates"` • `"Resource": "*/rates"` • `"Resource": "*/*"`
gRPC	• `"Resource": "arn:aws:vpc-lattice:us-west-2:1234567890:service/svc-0123456789abcdef0/api.parking/GetRates"` • `"Resource": "arn:aws:vpc-lattice:us-west-2:1234567890:service/svc-0123456789abcdef0/api.parking/*"` • `"Resource": "arn:aws:vpc-lattice:us-west-2:1234567890:service/svc-0123456789abcdef0/*"`

Gunakan format sumber daya Amazon Resource Name (ARN) berikut untuk: `<serviceARN>`

```
arn:aws:vpc-lattice:region:account-id:service/service-id
```

Contoh:

```
"Resource": "arn:aws:vpc-lattice:us-west-2:123456789012:service/svc-0123456789abcdef0"
```

Kunci kondisi yang dapat digunakan dalam kebijakan autentikasi

Akses dapat dikontrol lebih lanjut oleh kunci kondisi dalam elemen Kondisi kebijakan autentikasi. Kunci kondisi ini hadir untuk evaluasi tergantung pada protokol dan apakah permintaan ditandatangani dengan Signature Version 4 (SigV4) atau anonim. Kunci kondisi peka huruf besar/kecil.

AWS menyediakan kunci kondisi global yang dapat Anda gunakan untuk mengontrol akses, seperti `aws:PrincipalOrgID` dan`aws:SourceIp`. Untuk melihat daftar kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Kisah berikut mencantumkan kunci kondisi VPC Lattice. Untuk informasi selengkapnya, lihat Kunci kondisi untuk Layanan Kisi VPC Amazon di Referensi Otorisasi Layanan.

Kunci syarat	Deskripsi	Contoh	Tersedia untuk penelepon anonim (tidak diautentikasi)?	Tersedia untuk gRPC?
`vpc-lattice-svcs:Port`	Memfilter akses oleh port layanan permintaan dibuat	80	Ya	Ya
`vpc-lattice-svcs:RequestMethod`	Memfilter akses dengan metode permintaan	GET	Ya	Selalu POST
`vpc-lattice-svcs:RequestPath`	Memfilter akses berdasarkan bagian jalur dari URL permintaan	/path	Ya	Ya
`vpc-lattice-svcs:RequestHeader/` *`header-name`* `:` *`value`*	Memfilter akses dengan pasangan nama-nilai header di header permintaan	content-type: application/json	Ya	Ya

Kunci syarat	Deskripsi	Contoh	Tersedia untuk penelepon anonim (tidak diautentikasi)?	Tersedia untuk gRPC?
`vpc-lattice-svcs:RequestQueryString/` *`key-name`*`: `*`value`*	Memfilter akses dengan pasangan nilai kunci string kueri di URL permintaan	quux: [corge, grault]	Ya	Tidak
`vpc-lattice-svcs:ServiceNetworkArn`	Memfilter akses oleh ARN dari jaringan layanan layanan yang menerima permintaan	arn:aws:vpc-lattice:us-west-2:123456789012:servicenetwork/sn-0123456789abcdef0	Ya	Ya
`vpc-lattice-svcs:ServiceArn`	Memfilter akses oleh ARN dari layanan yang menerima permintaan	arn:aws:vpc-lattice:us-west-2:123456789012:service/svc-0123456789abcdef0	Ya	Ya
`vpc-lattice-svcs:SourceVpc`	Memfilter akses oleh VPC permintaan dibuat dari	vpc-1a2b3c4d	Ya	Ya

Kunci syarat	Deskripsi	Contoh	Tersedia untuk penelepon anonim (tidak diautentikasi)?	Tersedia untuk gRPC?
`vpc-lattice-svcs:SourceVpcOwnerAccount`	Memfilter akses oleh akun VPC yang memiliki permintaan dibuat	123456789012	Ya	Ya

Tag sumber daya

Tag adalah label metadata yang Anda tetapkan atau yang ditetapkan ke sumber AWS daya. AWS Setiap tag memiliki dua bagian:

- Sebuah kunci tag (misalnya, `CostCenter`, `Environment`, atau `Project`). Kunci tag peka huruf besar dan kecil.
- Bidang opsional yang dikenal sebagai nilai tag (misalnya, `111122223333` atau `Production`). Mengabaikan nilai tag sama dengan menggunakan sebuah string kosong. Seperti kunci tag, nilai tag peka huruf besar/kecil.

Untuk informasi selengkapnya tentang penandaan, lihat Mengontrol akses ke AWS sumber daya menggunakan tag

Anda dapat menggunakan tag dalam kebijakan autentikasi menggunakan kunci konteks kondisi `aws:ResourceTag/key` AWS global.

Contoh kebijakan berikut memberikan akses ke layanan dengan tag`Environment=Gamma`. Kebijakan ini memungkinkan Anda merujuk ke layanan tanpa layanan hard-coding atau. ARNs IDs

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

```
      "Effect": "Allow",
      "Principal": "*",
      "Action": "vpc-lattice-svcs:Invoke",
      "Resource": "arn:aws:vpc-lattice:us-west-2:123456789012:service/
svc-0124446789abcdef0/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Environment": "Gamma",
        }
      }
    }
  ]
}
```

Tag utama

Anda dapat mengontrol akses ke layanan dan sumber daya Anda berdasarkan tag yang dilampirkan pada identitas pemanggil. VPC Lattice mendukung kontrol akses berdasarkan tag utama apa pun pada tag pengguna, peran, atau sesi menggunakan variabel. `aws:PrincipalTag/context` Untuk informasi selengkapnya, lihat Mengontrol akses untuk prinsipal IAM.

Contoh kebijakan berikut memberikan akses hanya ke identitas dengan tag. `Team=Payments` Kebijakan ini memungkinkan Anda mengontrol akses tanpa akun IDs atau peran hardcoding. ARNs

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPaymentsTeam",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "vpc-lattice-svcs:Invoke",
      "Resource": "arn:aws:vpc-lattice:us-west-2:123456789012:service/
svc-0123456789abcdef0/*",
      "Condition": {
        "StringEquals": {
          "aws:PrincipalTag/Team": "Payments",
        }
      }
    }
  ]
}
```

Prinsipal anonim (tidak diautentikasi)

Prinsipal anonim adalah penelepon yang tidak menandatangani AWS permintaan mereka dengan Signature Version 4 (SigV4), dan berada dalam VPC yang terhubung ke jaringan layanan. Prinsipal anonim dapat membuat permintaan yang tidak diautentikasi ke layanan di jaringan layanan jika kebijakan autentikasi mengizinkannya.

Contoh kebijakan autentikasi

Berikut ini adalah contoh kebijakan autentikasi yang mengharuskan permintaan dibuat oleh prinsipal yang diautentikasi.

Semua contoh menggunakan `us-west-2` Wilayah dan berisi akun fiktif. IDs

Contoh 1: Batasi akses ke layanan oleh organisasi tertentu AWS

Contoh kebijakan autentikasi berikut memberikan izin untuk setiap permintaan yang diautentikasi untuk mengakses layanan apa pun di jaringan layanan tempat kebijakan tersebut berlaku. Namun, permintaan harus berasal dari kepala sekolah yang termasuk dalam AWS organisasi yang ditentukan dalam kondisi.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "vpc-lattice-svcs:Invoke",
         "Resource": "*",
         "Condition": {
            "StringEquals": {
               "aws:PrincipalOrgID": [
                  "o-123456example"
               ]
            }
         }
      }
   ]
}
```

Contoh 2: Batasi akses ke layanan dengan peran IAM tertentu

Contoh kebijakan autentikasi berikut memberikan izin untuk setiap permintaan yang diautentikasi yang menggunakan peran IAM `rates-client` untuk membuat permintaan HTTP GET pada layanan yang ditentukan dalam elemen. `Resource` Sumber daya dalam `Resource` elemen sama dengan layanan yang dilampirkan kebijakan tersebut.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect": "Allow",
         "Principal": {
            "AWS": [
               "arn:aws:iam::123456789012:role/rates-client"
            ]
         },
         "Action": "vpc-lattice-svcs:Invoke",
         "Resource": [
            "arn:aws:vpc-lattice:us-
west-2:123456789012:service/svc-0123456789abcdef0/*"
         ],
         "Condition": {
            "StringEquals": {
               "vpc-lattice-svcs:RequestMethod": "GET"
            }
         }
      }
   ]
}
```

Contoh 3: Batasi akses ke layanan oleh prinsipal yang diautentikasi di VPC tertentu

Contoh kebijakan autentikasi berikut hanya mengizinkan permintaan yang diautentikasi dari prinsipal di VPC yang ID VPC-nya. *vpc-1a2b3c4d*

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Principal": "*",
         "Action": "vpc-lattice-svcs:Invoke",
         "Resource": "*",
         "Condition": {
            "StringNotEquals": {
               "aws:PrincipalType": "Anonymous"
            },
            "StringEquals": {
               "vpc-lattice-svcs:SourceVpc": "vpc-1a2b3c4d"
            }
         }
      }
   ]
}
```

Cara kerja otorisasi

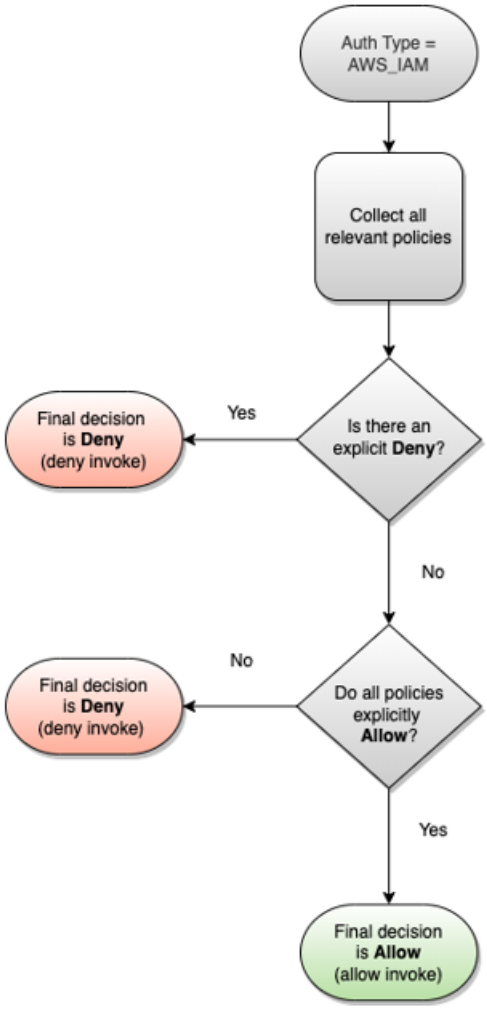
Ketika layanan VPC Lattice menerima permintaan, kode AWS penegakan akan mengevaluasi semua kebijakan izin yang relevan secara bersamaan untuk menentukan apakah akan mengotorisasi atau menolak permintaan tersebut. Ini mengevaluasi semua kebijakan berbasis identitas IAM dan kebijakan autentikasi yang berlaku dalam konteks permintaan selama otorisasi. Secara default, semua permintaan ditolak secara implisit saat jenis autentikasi. `AWS_IAM` Izin eksplisit dari semua kebijakan yang relevan akan mengesampingkan default.

Otorisasi meliputi:

- Mengumpulkan semua kebijakan dan kebijakan autentikasi berbasis identitas IAM yang relevan.
- Mengevaluasi serangkaian kebijakan yang dihasilkan:
  - Memverifikasi bahwa pemohon (seperti pengguna atau peran IAM) memiliki izin untuk melakukan operasi dari akun tempat pemohon berada. Jika tidak ada pernyataan izin eksplisit, AWS tidak mengotorisasi permintaan.

- Memverifikasi bahwa permintaan diizinkan oleh kebijakan autentikasi untuk jaringan layanan. Jika kebijakan autentikasi diaktifkan, tetapi tidak ada pernyataan izin eksplisit, AWS tidak mengotorisasi permintaan. Jika ada pernyataan allow eksplisit, atau tipe autentikasi`NONE`, kode berlanjut.
- Memverifikasi bahwa permintaan diizinkan oleh kebijakan autentikasi untuk layanan. Jika kebijakan autentikasi diaktifkan, tetapi tidak ada pernyataan izin eksplisit, AWS tidak mengotorisasi permintaan. Jika ada pernyataan allow eksplisit, atau tipe autentikasi**`NONE`**, maka kode penegakan mengembalikan keputusan akhir Izinkan.
- Penolakan secara tegas dalam kebijakan apa pun akan mengesampingkan izin apa pun.

Diagram menunjukkan alur kerja otorisasi. Ketika permintaan dibuat, kebijakan yang relevan mengizinkan atau menolak akses permintaan ke layanan tertentu.



Kontrol lalu lintas di VPC Lattice menggunakan grup keamanan

AWS Kelompok keamanan bertindak sebagai firewall virtual, mengendalikan lalu lintas jaringan ke dan dari entitas yang terkait dengannya. Dengan VPC Lattice, Anda dapat membuat grup keamanan dan menetapkannya ke asosiasi VPC yang menghubungkan VPC ke jaringan layanan untuk menegakkan perlindungan keamanan tingkat jaringan tambahan untuk jaringan layanan Anda. Jika Anda menghubungkan VPC ke jaringan layanan menggunakan titik akhir VPC, Anda juga dapat menetapkan grup keamanan ke titik akhir VPC. Demikian pula Anda dapat menetapkan grup keamanan ke gateway sumber daya yang Anda buat untuk mengaktifkan akses ke sumber daya di VPC Anda.

Daftar Isi

Daftar awalan terkelola

VPC Lattice menyediakan daftar awalan terkelola yang menyertakan alamat IP yang digunakan untuk merutekan lalu lintas melalui jaringan VPC Lattice saat Anda menggunakan asosiasi jaringan layanan untuk menghubungkan VPC Anda ke jaringan layanan menggunakan asosiasi VPC. Ini IPs adalah tautan pribadi-lokal IPs atau publik yang tidak dapat dirutekan. IPs

Anda dapat mereferensikan daftar awalan terkelola VPC Lattice dalam aturan grup keamanan Anda. Hal ini memungkinkan lalu lintas mengalir dari klien, melalui jaringan layanan VPC Lattice, dan ke target layanan VPC Lattice.

Misalnya, Anda memiliki instans EC2 yang terdaftar sebagai target di Wilayah Barat AS (Oregon) (`us-west-2`). Anda dapat menambahkan aturan ke grup keamanan instans yang mengizinkan akses HTTPS masuk dari daftar awalan terkelola VPC Lattice, sehingga lalu lintas VPC Lattice di Wilayah ini dapat mencapai instance. Jika Anda menghapus semua aturan masuk lainnya dari grup keamanan, Anda dapat mencegah lalu lintas apa pun selain lalu lintas VPC Lattice mencapai instans.

Nama-nama daftar awalan terkelola untuk VPC Lattice adalah sebagai berikut:

- com.amazonaws. *region*.vpc-kisi
- com.amazonaws. *region*.ipv6.vpc-kisi

Untuk informasi selengkapnya, lihat daftar awalan AWS-terkelola di Panduan Pengguna Amazon VPC.

Klien Windows dan macOS

Alamat dalam daftar awalan VPC Lattice adalah alamat link-local dan alamat publik yang tidak dapat dirutekan. Jika Anda terhubung ke VPC Lattice dari klien ini, Anda harus memperbarui konfigurasi mereka sehingga meneruskan alamat IP dalam daftar awalan terkelola ke alamat IP utama untuk klien. Berikut ini adalah contoh perintah yang memperbarui konfigurasi klien Windows, di mana 169.254.171.0 salah satu alamat dalam daftar awalan terkelola.

```
C:\> route add 169.254.171.0 mask 255.255.255.0 primary-ip-address
```

Berikut ini adalah contoh perintah yang memperbarui konfigurasi klien macOS, di mana 169.254.171.0 salah satu alamat dalam daftar awalan terkelola.

```
sudo route -n add -net 169.254.171.0 primary-ip-address 255.255.255.0
```

Untuk menghindari pembuatan rute statis, kami sarankan Anda menggunakan titik akhir jaringan layanan di VPC untuk membangun konektivitas. Untuk informasi selengkapnya, lihat the section called “Kelola asosiasi titik akhir VPC jaringan layanan”.

Aturan-aturan grup keamanan

Menggunakan VPC Lattice dengan atau tanpa grup keamanan tidak akan memengaruhi konfigurasi grup keamanan VPC Anda yang ada. Namun, Anda dapat menambahkan grup keamanan Anda sendiri kapan saja.

Pertimbangan utama

- Aturan grup keamanan untuk klien mengontrol lalu lintas keluar ke VPC Lattice.
- Aturan grup keamanan untuk target mengontrol lalu lintas masuk dari Kisi VPC ke target, termasuk lalu lintas pemeriksaan kesehatan.
- Aturan grup keamanan untuk hubungan antara jaringan layanan dan kontrol VPC yang klien dapat mengakses jaringan layanan VPC Lattice.
- Aturan grup keamanan untuk gateway sumber daya mengontrol lalu lintas keluar dari gateway sumber daya ke sumber daya.

Aturan keluar yang disarankan untuk lalu lintas yang mengalir dari gateway sumber daya ke sumber daya database

Agar lalu lintas mengalir dari gateway sumber daya ke sumber daya, Anda harus membuat aturan keluar untuk port terbuka dan protokol pendengar yang diterima untuk sumber daya.

Destinasi	Protokol	Rentang port	Komentar
CIDR range for resource	*TCP*	*3306*	Izinkan lalu lintas dari gateway sumber daya ke database

Aturan masuk yang direkomendasikan untuk jaringan layanan dan asosiasi VPC

Agar lalu lintas mengalir dari klien VPCs ke layanan yang terkait dengan jaringan layanan, Anda harus membuat aturan masuk untuk port pendengar dan protokol pendengar untuk layanan.

Sumber	Protokol	Rentang port	Komentar
VPC CIDR	*listener*	*listener*	Izinkan lalu lintas dari klien ke VPC Lattice

Aturan keluar yang disarankan untuk lalu lintas yang mengalir dari instance klien ke VPC Lattice

Secara default, grup keamanan mengizinkan semua lalu lintas ke luar. Namun, jika Anda memiliki aturan keluar khusus, Anda harus mengizinkan lalu lintas keluar ke awalan VPC Lattice untuk port dan protokol pendengar sehingga instance klien dapat terhubung ke semua layanan yang terkait dengan jaringan layanan VPC Lattice. Anda dapat mengizinkan lalu lintas ini dengan mereferensikan ID daftar awalan untuk VPC Lattice.

Destinasi	Protokol	Rentang port	Komentar
ID of the VPC Lattice prefix list	*listener*	*listener*	Izinkan lalu lintas dari klien ke VPC Lattice

Aturan masuk yang direkomendasikan untuk lalu lintas yang mengalir dari VPC Lattice ke instance target

Anda tidak dapat menggunakan grup keamanan klien sebagai sumber untuk grup keamanan target Anda, karena lalu lintas mengalir dari VPC Lattice. Anda dapat mereferensikan ID daftar awalan untuk VPC Lattice.

Sumber	Protokol	Rentang port	Komentar
ID of the VPC Lattice prefix list	*target*	*target*	Izinkan lalu lintas dari VPC Lattice ke target
ID of the VPC Lattice prefix list	*health check*	*health check*	Izinkan lalu lintas pemeriksaan kesehatan dari VPC Lattice ke target

Mengelola grup keamanan untuk asosiasi VPC

Anda dapat menggunakan AWS CLI untuk melihat, menambah, atau memperbarui grup keamanan di VPC ke asosiasi jaringan layanan. Saat menggunakan AWS CLI, ingatlah bahwa perintah Anda berjalan di AWS Region konfigurasi untuk profil Anda. Jika Anda ingin menjalankan perintah di Wilayah yang berbeda, ubah Wilayah default untuk profil Anda, atau gunakan parameter `--region` bersama perintah tersebut.

Sebelum Anda mulai, konfirmasikan bahwa Anda telah membuat grup keamanan di VPC yang sama dengan VPC yang ingin Anda tambahkan ke jaringan layanan. Untuk informasi selengkapnya, lihat Mengontrol lalu lintas ke sumber daya Anda menggunakan grup keamanan di Panduan Pengguna Amazon VPC

Untuk menambahkan grup keamanan saat Anda membuat asosiasi VPC menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pada tab asosiasi VPC, pilih Buat asosiasi VPC lalu pilih Tambahkan asosiasi VPC.

5. Pilih VPC dan hingga lima grup keamanan.
6. Pilih Simpan perubahan.

Untuk menambah atau memperbarui grup keamanan untuk asosiasi VPC yang ada menggunakan konsol

1. Buka konsol VPC Amazon di. https://console.aws.amazon.com/vpc/
2. Di panel navigasi, di bawah VPC Lattice, pilih Jaringan layanan.
3. Pilih nama jaringan layanan untuk membuka halaman detailnya.
4. Pada tab Asosiasi VPC, pilih kotak centang untuk asosiasi, lalu pilih Tindakan, Edit grup keamanan.
5. Tambahkan dan hapus grup keamanan sesuai kebutuhan.
6. Pilih Simpan perubahan.

Untuk menambahkan grup keamanan saat Anda membuat asosiasi VPC menggunakan AWS CLI

Gunakan perintah create-service-network-vpc-association, tentukan ID VPC untuk asosiasi VPC dan ID grup keamanan yang akan ditambahkan.

```
aws vpc-lattice create-service-network-vpc-association \
    --service-network-identifier sn-0123456789abcdef0 \
    --vpc-identifier vpc-1a2b3c4d \
    --security-group-ids sg-7c2270198example
```

Jika berhasil, perintah mengembalikan output yang serupa dengan berikut.

```
{
  "arn": "arn",
  "createdBy": "464296918874",
  "id": "snva-0123456789abcdef0",
  "status": "CREATE_IN_PROGRESS",
  "securityGroupIds": ["sg-7c2270198example"]
}
```

Untuk menambah atau memperbarui grup keamanan untuk asosiasi VPC yang ada menggunakan AWS CLI

Gunakan perintah update-service-network-vpc-association, tentukan ID jaringan layanan dan grup IDs keamanan. Grup keamanan ini mengesampingkan grup keamanan yang sebelumnya terkait. Tentukan setidaknya satu grup keamanan saat memperbarui daftar.

```
aws vpc-lattice update-service-network-vpc-association
    --service-network-vpc-association-identifier sn-903004f88example \
    --security-group-ids sg-7c2270198example sg-903004f88example
```

Warning

Anda tidak dapat menghapus semua grup keamanan. Sebagai gantinya, Anda harus terlebih dahulu menghapus asosiasi VPC, dan kemudian membuat ulang asosiasi VPC tanpa grup keamanan apa pun. Berhati-hatilah saat menghapus asosiasi VPC. Ini mencegah lalu lintas mencapai layanan yang ada di jaringan layanan itu.

Kontrol lalu lintas ke VPC Lattice menggunakan jaringan ACLs

Network Access Control List (ACL) memungkinkan atau menolak lalu lintas masuk atau keluar tertentu di tingkat subnet. ACL jaringan default memungkinkan semua lalu lintas masuk dan keluar. Anda dapat membuat jaringan khusus ACLs untuk subnet Anda untuk memberikan lapisan keamanan tambahan. Untuk informasi selengkapnya, lihat ACLs Jaringan dalam Panduan Pengguna Amazon VPC.

Daftar Isi

Jaringan ACLs untuk subnet klien Anda

Jaringan ACLs untuk subnet klien harus memungkinkan lalu lintas antara klien dan VPC Lattice. Anda bisa mendapatkan rentang alamat IP untuk mengizinkan dari daftar awalan terkelola untuk VPC Lattice.

Berikut ini adalah contoh aturan inbound.

Sumber	Protokol	Rentang port	Komentar
vpc_latti ce_cidr_block	TCP	1025-65535	Izinkan lalu lintas dari VPC Lattice ke klien

Berikut ini adalah contoh aturan outbound.

Destinasi	Protokol	Rentang port	Komentar
vpc_latti ce_cidr_block	*listener*	*listener*	Izinkan lalu lintas dari klien ke VPC Lattice

Jaringan ACLs untuk subnet target Anda

Jaringan ACLs untuk subnet target harus memungkinkan lalu lintas antara target dan Kisi VPC pada port target dan port pemeriksaan kesehatan. Anda bisa mendapatkan rentang alamat IP untuk mengizinkan dari daftar awalan terkelola untuk VPC Lattice.

Berikut ini adalah contoh aturan inbound.

Sumber	Protokol	Rentang port	Komentar
vpc_latti ce_cidr_block	*target*	*target*	Izinkan lalu lintas dari VPC Lattice ke target
vpc_latti ce_cidr_block	*health check*	*health check*	Izinkan lalu lintas pemeriksaan kesehatan dari VPC Lattice ke target

Berikut ini adalah contoh aturan outbound.

Destinasi	Protokol	Rentang port	Komentar
vpc_latti ce_cidr_block	*target*	1024-65535	Izinkan lalu lintas dari target ke VPC Lattice
vpc_latti ce_cidr_block	*health check*	1024-65535	Izinkan lalu lintas pemeriksaan kesehatan dari target ke VPC Lattice

SIGv4 permintaan yang diautentikasi untuk Amazon VPC Lattice

VPC Lattice menggunakan Signature Version 4 (SIGv4) atau Signature Version 4A (SIGv4A) untuk otentikasi klien. Untuk informasi selengkapnya, lihat AWS Signature Version 4 untuk permintaan API dalam Panduan Pengguna IAM.

Pertimbangan-pertimbangan

- VPC Lattice mencoba untuk mengautentikasi permintaan apa pun yang ditandatangani dengan atau A. SIGv4 SIGv4 Permintaan gagal tanpa otentikasi.
- VPC Lattice tidak mendukung penandatanganan payload. Anda harus mengirim `x-amz-content-sha256` header dengan nilai yang disetel ke`"UNSIGNED-PAYLOAD"`.

Contoh

- Python
- Java
- Node.js
- Golang
- Golang - GRPC

Python

Contoh ini mengirimkan permintaan yang ditandatangani melalui koneksi aman ke layanan yang terdaftar di jaringan. Jika Anda lebih suka menggunakan permintaan, paket botocore

menyederhanakan proses otentikasi, tetapi tidak sepenuhnya diperlukan. Untuk informasi selengkapnya, lihat Kredensyal dalam dokumentasi Boto3.

Untuk menginstal `botocore` dan `awscrt` paket, gunakan perintah berikut. Untuk informasi lebih lanjut, lihat AWS CRT Python.

```
pip install botocore awscrt
```

Jika Anda menjalankan aplikasi klien di Lambda, instal modul yang diperlukan menggunakan lapisan Lambda, atau sertakan dalam paket penyebaran Anda.

Dalam contoh berikut, ganti nilai placeholder dengan nilai Anda sendiri.

SIGv4

```
from botocore import crt
import requests
from botocore.awsrequest import AWSRequest
import botocore.session

if __name__ == '__main__':
    session = botocore.session.Session()
    signer = crt.auth.CrtSigV4Auth(session.get_credentials(), 'vpc-lattice-svcs',
 'us-west-2')
    endpoint = 'https://data-svc-022f67d3a42.1234abc.vpc-lattice-svcs.us-
west-2.on.aws'
    data = "some-data-here"
    headers = {'Content-Type': 'application/json', 'x-amz-content-sha256':
 'UNSIGNED-PAYLOAD'}
    request = AWSRequest(method='POST', url=endpoint, data=data, headers=headers)
    request.context["payload_signing_enabled"] = False
    signer.add_auth(request)

    prepped = request.prepare()

    response = requests.post(prepped.url, headers=prepped.headers, data=data)
    print(response.text)
```

SIGv4A

```
from botocore import crt
import requests
```

```
from botocore.awsrequest import AWSRequest
import botocore.session

if __name__ == '__main__':
    session = botocore.session.Session()
    signer = crt.auth.CrtSigV4AsymAuth(session.get_credentials(), 'vpc-lattice-
svcs', '*')
    endpoint = 'https://data-svc-022f67d3a42.1234abc.vpc-lattice-svcs.us-
west-2.on.aws'
    data = "some-data-here"
    headers = {'Content-Type': 'application/json', 'x-amz-content-sha256':
 'UNSIGNED-PAYLOAD'}
    request = AWSRequest(method='POST', url=endpoint, data=data, headers=headers)
    request.context["payload_signing_enabled"] = False
    signer.add_auth(request)

    prepped = request.prepare()

    response = requests.post(prepped.url, headers=prepped.headers, data=data)
    print(response.text)
```

Java

Contoh ini menunjukkan bagaimana Anda dapat melakukan penandatanganan permintaan dengan menggunakan pencegat khusus. Ini menggunakan kelas penyedia kredensyal default dari AWS SDK for Java 2.x, yang mendapatkan kredensyal yang benar untuk Anda. Jika Anda lebih suka menggunakan penyedia kredensi tertentu, Anda dapat memilih salah satu dari. AWS SDK for Java 2.x Hanya AWS SDK for Java memungkinkan muatan yang tidak ditandatangani melalui HTTPS. Namun, Anda dapat memperpanjang penandatangan untuk mendukung muatan yang tidak ditandatangani melalui HTTP.

SIGv4

```
package com.example;

import software.amazon.awssdk.http.auth.aws.signer.AwsV4HttpSigner;
import software.amazon.awssdk.http.auth.spi.signer.SignedRequest;

import software.amazon.awssdk.http.SdkHttpMethod;
import software.amazon.awssdk.http.SdkHttpClient;
import software.amazon.awssdk.identity.spi.AwsCredentialsIdentity;
```

```
import software.amazon.awssdk.http.SdkHttpRequest;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.http.HttpExecuteRequest;
import software.amazon.awssdk.http.HttpExecuteResponse;
import java.io.IOException;
import java.net.URI;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;

public class sigv4 {

    public static void main(String[] args) {
        AwsV4HttpSigner signer = AwsV4HttpSigner.create();

        AwsCredentialsIdentity credentials =
 DefaultCredentialsProvider.create().resolveCredentials();

        if (args.length < 2) {
            System.out.println("Usage: sample <url> <region>");
            System.exit(1);
        }
        // Create the HTTP request to be signed
        var url = args[0];
        SdkHttpRequest httpRequest = SdkHttpRequest.builder()
                .uri(URI.create(url))
                .method(SdkHttpMethod.GET)
                .build();

        SignedRequest signedRequest = signer.sign(r -> r.identity(credentials)
                .request(httpRequest)
                .putProperty(AwsV4HttpSigner.SERVICE_SIGNING_NAME, "vpc-lattice-
svcs")
                .putProperty(AwsV4HttpSigner.PAYLOAD_SIGNING_ENABLED, false)
                .putProperty(AwsV4HttpSigner.REGION_NAME, args[1]));

        System.out.println("[*] Raw request headers:");
        signedRequest.request().headers().forEach((key, values) -> {
            values.forEach(value -> System.out.println("  " + key + ": " + value));
        });

        try (SdkHttpClient httpClient = ApacheHttpClient.create()) {
            HttpExecuteRequest httpExecuteRequest = HttpExecuteRequest.builder()
                    .request(signedRequest.request())
                    .contentStreamProvider(signedRequest.payload().orElse(null))
```

```
                    .build();

            System.out.println("[*] Sending request to: " + url);

            HttpExecuteResponse httpResponse =
 httpClient.prepareRequest(httpExecuteRequest).call();

            System.out.println("[*] Request sent");

            System.out.println("[*] Response status code: " +
 httpResponse.httpResponse().statusCode());
            // Read and print the response body
            httpResponse.responseBody().ifPresent(inputStream -> {
                try {
                    String responseBody = new String(inputStream.readAllBytes());
                    System.out.println("[*] Response body: " + responseBody);
                } catch (IOException e) {
                    System.err.println("[*] Failed to read response body");
                    e.printStackTrace();
                } finally {
                    try {
                        inputStream.close();
                    } catch (IOException e) {
                        System.err.println("[*] Failed to close input stream");
                        e.printStackTrace();
                    }
                }
            });
        } catch (IOException e) {
            System.err.println("[*] HTTP Request Failed.");
            e.printStackTrace();
        }

    }
}
```

SIGv4A

Contoh ini membutuhkan ketergantungan tambahan pada`software.amazon.awssdk:http-auth-aws-crt`.

```
package com.example;
```

```
import software.amazon.awssdk.http.auth.aws.signer.AwsV4aHttpSigner;
import software.amazon.awssdk.http.auth.aws.signer.RegionSet;
import software.amazon.awssdk.http.auth.spi.signer.SignedRequest;

import software.amazon.awssdk.http.SdkHttpMethod;
import software.amazon.awssdk.http.SdkHttpClient;
import software.amazon.awssdk.identity.spi.AwsCredentialsIdentity;
import software.amazon.awssdk.http.SdkHttpRequest;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.http.HttpExecuteRequest;
import software.amazon.awssdk.http.HttpExecuteResponse;
import java.io.IOException;
import java.net.URI;
import java.util.Arrays;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;

public class sigv4a {

    public static void main(String[] args) {
        AwsV4aHttpSigner signer = AwsV4aHttpSigner.create();

        AwsCredentialsIdentity credentials =
 DefaultCredentialsProvider.create().resolveCredentials();

        if (args.length < 2) {
            System.out.println("Usage: sample <url> <regionset>");
            System.exit(1);
        }
        // Create the HTTP request to be signed
        var url = args[0];
        SdkHttpRequest httpRequest = SdkHttpRequest.builder()
                .uri(URI.create(url))
                .method(SdkHttpMethod.GET)
                .build();

        SignedRequest signedRequest = signer.sign(r -> r.identity(credentials)
                .request(httpRequest)
                .putProperty(AwsV4aHttpSigner.SERVICE_SIGNING_NAME, "vpc-lattice-
svcs")
                .putProperty(AwsV4aHttpSigner.PAYLOAD_SIGNING_ENABLED, false)
                .putProperty(AwsV4aHttpSigner.REGION_SET,
 RegionSet.create(String.join(" ",Arrays.copyOfRange(args, 1, args.length)))));
```

```
        System.out.println("[*] Raw request headers:");
        signedRequest.request().headers().forEach((key, values) -> {
            values.forEach(value -> System.out.println("  " + key + ": " + value));
        });

        try (SdkHttpClient httpClient = ApacheHttpClient.create()) {
            HttpExecuteRequest httpExecuteRequest = HttpExecuteRequest.builder()
                    .request(signedRequest.request())
                    .contentStreamProvider(signedRequest.payload().orElse(null))
                    .build();

            System.out.println("[*] Sending request to: " + url);

            HttpExecuteResponse httpResponse =
httpClient.prepareRequest(httpExecuteRequest).call();

            System.out.println("[*] Request sent");

            System.out.println("[*] Response status code: " +
httpResponse.httpResponse().statusCode());
            // Read and print the response body
            httpResponse.responseBody().ifPresent(inputStream -> {
                try {
                    String responseBody = new String(inputStream.readAllBytes());
                    System.out.println("[*] Response body: " + responseBody);
                } catch (IOException e) {
                    System.err.println("[*] Failed to read response body");
                    e.printStackTrace();
                } finally {
                    try {
                        inputStream.close();
                    } catch (IOException e) {
                        System.err.println("[*] Failed to close input stream");
                        e.printStackTrace();
                    }
                }
            });
        } catch (IOException e) {
            System.err.println("[*] HTTP Request Failed.");
            e.printStackTrace();
        }
    }
```

```
}
```

Node.js

Contoh ini menggunakan binding NodeJS aws-crt untuk mengirim permintaan yang ditandatangani menggunakan HTTPS.

Untuk menginstal `aws-crt` paket, gunakan perintah berikut.

```
npm -i aws-crt
```

Jika variabel `AWS_REGION` lingkungan ada, contoh menggunakan Region ditentukan oleh`AWS_REGION`. Wilayah default adalah`us-east-1`.

SIGv4

```
const https = require('https')
const crt = require('aws-crt')
const { HttpRequest } = require('aws-crt/dist/native/http')

function sigV4Sign(method, endpoint, service, algorithm) {
    const host = new URL(endpoint).host
    const request = new HttpRequest(method, endpoint)
    request.headers.add('host', host)
    // crt.io.enable_logging(crt.io.LogLevel.INFO)
    const config = {
        service: service,
        region: process.env.AWS_REGION ? process.env.AWS_REGION : 'us-east-1',
        algorithm: algorithm,
        signature_type: crt.auth.AwsSignatureType.HttpRequestViaHeaders,
        signed_body_header: crt.auth.AwsSignedBodyHeaderType.XAmzContentSha256,
        signed_body_value: crt.auth.AwsSignedBodyValue.UnsignedPayload,
        provider: crt.auth.AwsCredentialsProvider.newDefault()
    }

    return crt.auth.aws_sign_request(request, config)
}

if (process.argv.length === 2) {
  console.error(process.argv[1] + ' <url>')
  process.exit(1)
```

```
}

const algorithm = crt.auth.AwsSigningAlgorithm.SigV4;

sigV4Sign('GET', process.argv[2], 'vpc-lattice-svcs', algorithm).then(
  httpResponse => {
    var headers = {}

    for (const sigv4header of httpResponse.headers) {
      headers[sigv4header[0]] = sigv4header[1]
    }

    const options = {
      hostname: new URL(process.argv[2]).host,
      path: new URL(process.argv[2]).pathname,
      method: 'GET',
      headers: headers
    }

    req = https.request(options, res => {
      console.log('statusCode:', res.statusCode)
      console.log('headers:', res.headers)
      res.on('data', d => {
        process.stdout.write(d)
      })
    })
    req.on('error', err => {
      console.log('Error: ' + err)
    })
    req.end()
  }
)
```

SIGv4A

```
const https = require('https')
const crt = require('aws-crt')
const { HttpRequest } = require('aws-crt/dist/native/http')

function sigV4Sign(method, endpoint, service, algorithm) {
    const host = new URL(endpoint).host
    const request = new HttpRequest(method, endpoint)
    request.headers.add('host', host)
```

```
    // crt.io.enable_logging(crt.io.LogLevel.INFO)
    const config = {
        service: service,
        region: process.env.AWS_REGION ? process.env.AWS_REGION : 'us-east-1',
        algorithm: algorithm,
        signature_type: crt.auth.AwsSignatureType.HttpRequestViaHeaders,
        signed_body_header: crt.auth.AwsSignedBodyHeaderType.XAmzContentSha256,
        signed_body_value: crt.auth.AwsSignedBodyValue.UnsignedPayload,
        provider: crt.auth.AwsCredentialsProvider.newDefault()
    }

    return crt.auth.aws_sign_request(request, config)
}

if (process.argv.length === 2) {
  console.error(process.argv[1] + ' <url>')
  process.exit(1)
}

const algorithm = crt.auth.AwsSigningAlgorithm.SigV4Asymmetric;

sigV4Sign('GET', process.argv[2], 'vpc-lattice-svcs', algorithm).then(
  httpResponse => {
    var headers = {}

    for (const sigv4header of httpResponse.headers) {
      headers[sigv4header[0]] = sigv4header[1]
    }

    const options = {
      hostname: new URL(process.argv[2]).host,
      path: new URL(process.argv[2]).pathname,
      method: 'GET',
      headers: headers
    }

    req = https.request(options, res => {
      console.log('statusCode:', res.statusCode)
      console.log('headers:', res.headers)
      res.on('data', d => {
        process.stdout.write(d)
      })
    })
    req.on('error', err => {
```

```
      console.log('Error: ' + err)
    })
    req.end()
  }
)
```

Golang

Contoh ini menggunakan generator kode Smithy untuk Go dan AWS SDK untuk bahasa pemrograman Go untuk menangani permintaan penandatanganan permintaan. Contoh ini membutuhkan versi Go 1.21 atau lebih tinggi.

SIGv4

```
package main

import (
        "context"
        "flag"
        "fmt"
        "io"
        "log"
        "net/http"
        "net/http/httputil"
        "os"
        "strings"

        "github.com/aws/aws-sdk-go-v2/aws"
        "github.com/aws/aws-sdk-go-v2/config"
        "github.com/aws/smithy-go/aws-http-auth/credentials"
        "github.com/aws/smithy-go/aws-http-auth/sigv4"
        v4 "github.com/aws/smithy-go/aws-http-auth/v4"
)

type nopCloser struct {
        io.ReadSeeker
}

func (nopCloser) Close() error {
        return nil
}
```

```
type stringFlag struct {
        set   bool
        value string
}



        flag.PrintDefaults()
        os.Exit(1)
}

func main() {
        flag.Parse()
        if !url.set || !region.set {
                Usage()
        }

        cfg, err := config.LoadDefaultConfig(context.TODO(),
 config.WithClientLogMode(aws.LogSigning))
        if err != nil {
                log.Fatalf("failed to load SDK configuration, %v", err)
        }

        if len(os.Args) < 2 {
                log.Fatalf("Usage: go run main.go  <url>")
        }

        // Retrieve credentials from an SDK source, such as the instance profile
        sdkCreds, err := cfg.Credentials.Retrieve(context.TODO())
        if err != nil {
                log.Fatalf("Unable to retrieve credentials from SDK, %v", err)
        }

        creds := credentials.Credentials{
                AccessKeyID:     sdkCreds.AccessKeyID,
                SecretAccessKey: sdkCreds.SecretAccessKey,
                SessionToken:    sdkCreds.SessionToken,
        }

        // Add a payload body, which will not be part of the signature calculation
        body := nopCloser{strings.NewReader(`Example payload body`)}

        req, _ := http.NewRequest(http.MethodPost, url.value, body)

        // Create a sigv4a signer with specific options
```

```
        signer := sigv4.New(func(o *v4.SignerOptions) {
                o.DisableDoublePathEscape = true
                // This will add the UNSIGNED-PAYLOAD sha256 header
                o.AddPayloadHashHeader = true
                o.DisableImplicitPayloadHashing = true
        })

        // Perform the signing on req, using the credentials we retrieved from the
SDK
        err = signer.SignRequest(&sigv4.SignRequestInput{
                Request:     req,
                Credentials: creds,
                Service:     "vpc-lattice-svcs",
                Region: region.String(),
        })

        if err != nil {
                log.Fatalf("%s", err)
        }

        res, err := httputil.DumpRequest(req, true)

        if err != nil {
                log.Fatalf("%s", err)
        }

        log.Printf("[*] Raw request\n%s\n", string(res))

        log.Printf("[*] Sending request to %s\n", url.value)

        resp, err := http.DefaultClient.Do(req)
        if err != nil {
                log.Fatalf("%s", err)
        }

        log.Printf("[*] Request sent\n")

        log.Printf("[*] Response status code: %d\n", resp.StatusCode)

        respBody, err := io.ReadAll(resp.Body)
        if err != nil {
                log.Fatalf("%s", err)
        }
```

```
        log.Printf("[*] Response body: \n%s\n", respBody)
}
```

SIGv4A

```
package main

import (
        "context"
        "flag"
        "fmt"
        "io"
        "log"
        "net/http"
        "net/http/httputil"
        "os"
        "strings"

        "github.com/aws/aws-sdk-go-v2/aws"
        "github.com/aws/aws-sdk-go-v2/config"
        "github.com/aws/smithy-go/aws-http-auth/credentials"
        "github.com/aws/smithy-go/aws-http-auth/sigv4a"
        v4 "github.com/aws/smithy-go/aws-http-auth/v4"
)

type nopCloser struct {
        io.ReadSeeker
}

func (nopCloser) Close() error {
        return nil
}

type stringFlag struct {

func main() {
        flag.Parse()
        if !url.set || !regionSet.set {
                Usage()
        }

        cfg, err := config.LoadDefaultConfig(context.TODO(),
 config.WithClientLogMode(aws.LogSigning))
```

```
        if err != nil {
                log.Fatalf("failed to load SDK configuration, %v", err)
        }

        if len(os.Args) < 2 {
                log.Fatalf("Usage: go run main.go <url>")
        }

        // Retrieve credentials from an SDK source, such as the instance profile
        sdkCreds, err := cfg.Credentials.Retrieve(context.TODO())
        if err != nil {
                log.Fatalf("Unable to retrieve credentials from SDK, %v", err)
        }

        creds := credentials.Credentials{
                AccessKeyID:     sdkCreds.AccessKeyID,
                SecretAccessKey: sdkCreds.SecretAccessKey,
                SessionToken:    sdkCreds.SessionToken,
        }

        // Add a payload body, which will not be part of the signature calculation
        body := nopCloser{strings.NewReader(`Example payload body`)}

        req, _ := http.NewRequest(http.MethodPost, url.value, body)

        // Create a sigv4a signer with specific options
        signer := sigv4a.New(func(o *v4.SignerOptions) {
                o.DisableDoublePathEscape = true
                // This will add the UNSIGNED-PAYLOAD sha256 header
                o.AddPayloadHashHeader = true
                o.DisableImplicitPayloadHashing = true
        })

        // Create a slice out of the provided regionset
        rs := strings.Split(regionSet.value, ",")

        // Perform the signing on req, using the credentials we retrieved from the
SDK
        err = signer.SignRequest(&sigv4a.SignRequestInput{
                Request:     req,
                Credentials: creds,
                Service:     "vpc-lattice-svcs",
                RegionSet: rs,
        })
```

```
        if err != nil {
                log.Fatalf("%s", err)
        }

        res, err := httputil.DumpRequest(req, true)

        if err != nil {
                log.Fatalf("%s", err)
        }

        log.Printf("[*] Raw request\n%s\n", string(res))

        log.Printf("[*] Sending request to %s\n", url.value)

        resp, err := http.DefaultClient.Do(req)
        if err != nil {
                log.Fatalf("%s", err)
        }

        log.Printf("[*] Request sent\n")

        log.Printf("[*] Response status code: %d\n", resp.StatusCode)

        respBody, err := io.ReadAll(resp.Body)
        if err != nil {
                log.Fatalf("%s", err)
        }

        log.Printf("[*] Response body: \n%s\n", respBody)
}
```

Golang - GRPC

Contoh ini menggunakan AWS SDK untuk bahasa pemrograman Go untuk menangani penandatanganan permintaan untuk permintaan GRPC. Ini dapat digunakan dengan server gema dari repositori kode sampel GRPC.

```
package main

import (
    "context"
```

```
    "crypto/tls"
    "crypto/x509"

    "flag"
    "fmt"
    "log"
    "net/http"
    "net/url"
    "strings"
    "time"

    "google.golang.org/grpc"
    "google.golang.org/grpc/credentials"

    "github.com/aws/aws-sdk-go-v2/aws"
    v4 "github.com/aws/aws-sdk-go-v2/aws/signer/v4"
    "github.com/aws/aws-sdk-go-v2/config"

    ecpb "google.golang.org/grpc/examples/features/proto/echo"
)

const (
    headerContentSha    = "x-amz-content-sha256"
    headerSecurityToken = "x-amz-security-token"
    headerDate          = "x-amz-date"
    headerAuthorization = "authorization"
    unsignedPayload     = "UNSIGNED-PAYLOAD"
)

type SigV4GrpcSigner struct {
    service      string
    region       string
    credProvider aws.CredentialsProvider
    signer       *v4.Signer
}

func NewSigV4GrpcSigner(service string, region string, credProvider
 aws.CredentialsProvider) *SigV4GrpcSigner {
    signer := v4.NewSigner()
    return &SigV4GrpcSigner{
        service:      service,
        region:       region,
        credProvider: credProvider,
        signer:       signer,
```

```
    }
}

func (s *SigV4GrpcSigner) GetRequestMetadata(ctx context.Context, uri ...string)
 (map[string]string, error) {
    ri, _ := credentials.RequestInfoFromContext(ctx)
    creds, err := s.credProvider.Retrieve(ctx)
    if err != nil {
        return nil, fmt.Errorf("failed to load credentials: %w", err)
    }

    // The URI we get here is scheme://authority/service/ - for siging we want to
 include the RPC name
    // But RequestInfoFromContext only has the combined /service/rpc-name - so read the
 URI, and
    // replace the Path with what we get from RequestInfo.
    parsed, err := url.Parse(uri[0])
    if err != nil {
        return nil, err
    }
    parsed.Path = ri.Method

    // Build a request for the signer.
    bodyReader := strings.NewReader("")
    req, err := http.NewRequest("POST", uri[0], bodyReader)
    if err != nil {
        return nil, err
    }
    date := time.Now()
    req.Header.Set(headerContentSha, unsignedPayload)
    req.Header.Set(headerDate, date.String())
    if creds.SessionToken != "" {
        req.Header.Set(headerSecurityToken, creds.SessionToken)
    }
    // The signer wants this as //authority/path
    // So get this by triming off the scheme and the colon before the first slash.
    req.URL.Opaque = strings.TrimPrefix(parsed.String(), parsed.Scheme+":")

    err = s.signer.SignHTTP(context.Background(), creds, req, unsignedPayload,
 s.service, s.region, date)
    if err != nil {
        return nil, fmt.Errorf("failed to sign request: %w", err)
    }
```

```
    // Pull the relevant headers out of the signer, and return them to get
    // included in the request we make.
    reqHeaders := map[string]string{
        headerContentSha:    req.Header.Get(headerContentSha),
        headerDate:          req.Header.Get(headerDate),
        headerAuthorization: req.Header.Get(headerAuthorization),
    }
    if req.Header.Get(headerSecurityToken) != "" {
        reqHeaders[headerSecurityToken] = req.Header.Get(headerSecurityToken)
    }

    return reqHeaders, nil
}

func (c *SigV4GrpcSigner) RequireTransportSecurity() bool {
    return true
}

var addr = flag.String("addr", "some-lattice-service:443", "the address to connect to")
var region = flag.String("region", "us-west-2", "region")

func callUnaryEcho(client ecpb.EchoClient, message string) {
    ctx, cancel := context.WithTimeout(context.Background(), 10*time.Second)
    defer cancel()
    resp, err := client.UnaryEcho(ctx, &ecpb.EchoRequest{Message: message})
    if err != nil {
        log.Fatalf("client.UnaryEcho(_) = _, %v: ", err)
    }
    fmt.Println("UnaryEcho: ", resp.Message)
}

func main() {
    flag.Parse()
    cfg, err := config.LoadDefaultConfig(context.TODO(),
 config.WithClientLogMode(aws.LogSigning))
    if err != nil {
        log.Fatalf("failed to load SDK configuration, %v", err)
    }

    pool, _ := x509.SystemCertPool()
    tlsConfig := &tls.Config{
        RootCAs: pool,
    }
```

```
    authority, _, _ := strings.Cut(*addr, ":") // Remove the port from the addr
    opts := []grpc.DialOption{
        grpc.WithTransportCredentials(credentials.NewTLS(tlsConfig)),

        // Lattice needs both the Authority to be set (without a port), and the SigV4
 signer
        grpc.WithAuthority(authority),
        grpc.WithPerRPCCredentials(NewSigV4GrpcSigner("vpc-lattice-svcs", *region,
 cfg.Credentials)),
    }

    conn, err := grpc.Dial(*addr, opts...)

    if err != nil {
        log.Fatalf("did not connect: %v", err)
    }
    defer conn.Close()
    rgc := ecpb.NewEchoClient(conn)

    callUnaryEcho(rgc, "hello world")
}
```

Perlindungan data di Amazon VPC Lattice

Model tanggung jawab AWS bersama model berlaku untuk perlindungan data di Amazon VPC Lattice. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda harus bertanggung jawab untuk memelihara kendali terhadap konten yang di-hosting pada infrastruktur ini. Konten ini meliputi konfigurasi keamanan dan tugas-tugas pengelolaan untuk berbagai layanan Layanan AWS yang Anda gunakan. Lihat informasi yang lebih lengkap tentang privasi data dalam Pertanyaan Umum Privasi Data. Lihat informasi tentang perlindungan data di Eropa di pos blog Model Tanggung Jawab Bersama dan GDPR AWS di Blog Keamanan AWS .

Enkripsi saat bergerak

VPC Lattice adalah layanan yang dikelola sepenuhnya yang terdiri dari bidang kontrol dan pesawat data. Setiap pesawat melayani tujuan yang berbeda dalam layanan. Bidang kontrol menyediakan administrasi yang APIs digunakan untuk membuat, membaca/mendeskripsikan, memperbarui, menghapus, dan membuat daftar (CRUDL) sumber daya (misalnya, dan). `CreateService` `UpdateService` Komunikasi ke pesawat kontrol VPC Lattice dilindungi dalam perjalanan oleh TLS.

Bidang data adalah VPC Lattice Invoke API, yang menyediakan interkoneksi antar layanan. TLS mengenkripsi komunikasi ke bidang data VPC Lattice saat Anda menggunakan HTTPS atau TLS. Suite cipher dan versi protokol menggunakan default yang disediakan oleh VPC Lattice dan tidak dapat dikonfigurasi. Untuk informasi selengkapnya, lihat Pendengar HTTPS untuk layanan VPC Lattice.

Enkripsi saat diam

Secara default, enkripsi data saat istirahat 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.

Daftar Isi

Enkripsi di sisi server dengan kunci terkelola Amazon S3 (SSE-S3)

Saat Anda menggunakan enkripsi sisi server dengan kunci terkelola Amazon S3 (SSE-S3), setiap objek dienkripsi dengan kunci unik. Sebagai perlindungan tambahan, kami mengenkripsi kunci itu sendiri dengan kunci root yang kami putar secara teratur. Enkripsi sisi server Amazon S3 menggunakan salah satu cipher blok terkuat yang tersedia, 256-bit Advanced Encryption Standard (AES-256) GCM, untuk mengenkripsi data Anda. Untuk objek yang dienkripsi sebelum AES-GCM, AES-CBC masih didukung untuk mendekripsi objek tersebut. Untuk informasi selengkapnya, lihat Menggunakan enkripsi sisi server dengan kunci enkripsi terkelola Amazon S3 (SSE-S3).

Jika Anda mengaktifkan enkripsi sisi server dengan kunci enkripsi terkelola Amazon S3 (SSE-S3) untuk bucket S3 Anda untuk log akses VPC Lattice, kami secara otomatis mengenkripsi setiap file log akses sebelum disimpan di bucket S3 Anda. Untuk informasi selengkapnya, lihat Log yang dikirim ke Amazon S3 di CloudWatch Panduan Pengguna Amazon.

Enkripsi sisi server dengan AWS KMS kunci yang disimpan di (SSE-KMS) AWS KMS

Enkripsi sisi server dengan AWS KMS kunci (SSE-KMS) mirip dengan SSE-S3, tetapi dengan manfaat dan biaya tambahan untuk menggunakan layanan ini. Ada izin terpisah untuk AWS KMS kunci yang memberikan perlindungan tambahan terhadap akses tidak sah objek Anda di Amazon S3. SSE-KMS juga memberi Anda jejak audit yang menunjukkan kapan AWS KMS kunci Anda digunakan

dan oleh siapa. Untuk informasi selengkapnya, lihat Menggunakan enkripsi sisi server dengan AWS Key Management Service (SSE-KMS).

Daftar Isi

Enkripsi dan dekripsi kunci pribadi sertifikat Anda

Sertifikat ACM dan kunci pribadi Anda dienkripsi menggunakan kunci KMS AWS terkelola yang memiliki alias aws/acm. Anda dapat melihat ID kunci dengan alias ini di AWS KMS konsol di bawah kunci AWS terkelola.

VPC Lattice tidak langsung mengakses sumber daya ACM Anda. Ini menggunakan AWS TLS Connection Manager untuk mengamankan dan mengakses kunci pribadi untuk sertifikat Anda. Saat Anda menggunakan sertifikat ACM untuk membuat layanan VPC Lattice, VPC Lattice mengaitkan sertifikat Anda dengan TLS Connection Manager. AWS Ini dilakukan dengan membuat hibah AWS KMS terhadap Kunci AWS Terkelola Anda dengan awalan aws/acm. Hibah adalah instrumen kebijakan yang memungkinkan TLS Connection Manager untuk menggunakan kunci KMS dalam operasi kriptografi. Hibah ini memungkinkan prinsipal penerima hibah (TLS Connection Manager) untuk memanggil operasi hibah yang ditentukan pada kunci KMS untuk mendekripsi kunci pribadi sertifikat Anda. TLS Connection Manager kemudian menggunakan sertifikat dan kunci pribadi yang didekripsi (plaintext) untuk membuat koneksi aman (sesi SSL/TLS) dengan klien layanan VPC Lattice. Ketika sertifikat dipisahkan dari layanan VPC Lattice, hibah dihentikan.

Jika Anda ingin menghapus akses ke kunci KMS, kami sarankan Anda mengganti atau menghapus sertifikat dari layanan menggunakan Konsol Manajemen AWS atau `update-service` perintah di AWS CLI.

Konteks enkripsi untuk VPC Lattice

Konteks enkripsi adalah kumpulan opsional pasangan kunci-nilai yang berisi informasi kontekstual tentang apa kunci pribadi Anda mungkin digunakan untuk. AWS KMS mengikat konteks enkripsi ke data terenkripsi dan menggunakannya sebagai data otentikasi tambahan untuk mendukung enkripsi yang diautentikasi.

Ketika kunci TLS Anda digunakan dengan VPC Lattice dan TLS Connection manager, nama layanan VPC Lattice Anda disertakan dalam konteks enkripsi yang digunakan untuk mengenkripsi kunci Anda

saat istirahat. Anda dapat memverifikasi layanan VPC Lattice yang digunakan untuk sertifikat dan kunci pribadi Anda dengan melihat konteks enkripsi di CloudTrail log Anda seperti yang ditunjukkan di bagian berikutnya, atau dengan melihat tab Sumber Daya Terkait di konsol ACM.

Untuk mendekripsi data, konteks enkripsi yang sama disertakan dalam permintaan. VPC Lattice menggunakan konteks enkripsi yang sama di semua operasi kriptografi AWS KMS, di mana kuncinya adalah `aws:vpc-lattice:arn` dan nilainya adalah Nama Sumber Daya Amazon (ARN) dari layanan VPC Lattice.

Contoh berikut menunjukkan konteks enkripsi dalam output dari operasi seperti`CreateGrant`.

```
"encryptionContextEquals": {
    "aws:acm:arn": "arn:aws:acm:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "aws:vpc-lattice:arn": "arn:aws:vpc-lattice:us-west-2:111122223333:service/
svc-0b23c1234567890ab"
}
```

Memantau kunci enkripsi Anda untuk VPC Lattice

Bila Anda menggunakan kunci AWS terkelola dengan layanan VPC Lattice, Anda dapat menggunakannya AWS CloudTrailuntuk melacak permintaan yang dikirimkan oleh VPC Lattice. AWS KMS

CreateGrant

Ketika Anda menambahkan sertifikat ACM Anda ke layanan VPC Lattice, `CreateGrant` permintaan dikirim atas nama Anda untuk TLS Connection Manager untuk dapat mendekripsi kunci pribadi yang terkait dengan sertifikat ACM Anda

Anda dapat melihat `CreateGrant` operasi sebagai peristiwa di CloudTrail, Riwayat acara, CreateGrant.

Berikut ini adalah contoh catatan peristiwa dalam riwayat CloudTrail acara untuk `CreateGrant` operasi.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
```

```
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "IAMUser",
                "principalId": "EX_PRINCIPAL_ID",
                "arn": "arn:aws:iam::111122223333:user/Alice",
                "accountId": "111122223333",
                "userName": "Alice"
            },
            "webIdFederationData": {},
            "attributes": {
                "creationDate": "2023-02-06T23:30:50Z",
                "mfaAuthenticated": "false"
            }
        },
        "invokedBy": "acm.amazonaws.com"
    },
    "eventTime": "2023-02-07T00:07:18Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "acm.amazonaws.com",
    "userAgent": "acm.amazonaws.com",
    "requestParameters": {
        "granteePrincipal": "tlsconnectionmanager.amazonaws.com",
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "operations": [
            "Decrypt"
        ],
        "constraints": {
            "encryptionContextEquals": {
                "aws:acm:arn": "arn:aws:acm:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                "aws:vpc-lattice:arn": "arn:aws:vpc-lattice:us-
west-2:111122223333:service/svc-0b23c1234567890ab"
            }
        },
        "retiringPrincipal": "acm.us-west-2.amazonaws.com"
    },
    "responseElements": {
        "grantId": "f020fe75197b93991dc8491d6f19dd3cebb24ee62277a05914386724f3d48758",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
```

```
    },
    "requestID": "ba178361-8ab6-4bdd-9aa2-0d1a44b2974a",
    "eventID": "8d449963-1120-4d0c-9479-f76de11ce609",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Dalam `CreateGrant` contoh di atas, prinsipal penerima hibah adalah TLS Connection Manager, dan konteks enkripsi memiliki layanan VPC Lattice ARN.

ListGrants

Anda dapat menggunakan ID kunci KMS dan ID akun Anda untuk memanggil `ListGrants` API. Ini memberi Anda daftar semua hibah untuk kunci KMS yang ditentukan. Untuk informasi selengkapnya, lihat ListGrants.

Gunakan `ListGrants` perintah berikut di AWS CLI untuk melihat rincian semua hibah.

```
aws kms list-grants –key-id your-kms-key-id
```

Berikut ini adalah output contoh.

```
{
    "Grants": [
        {
            "Operations": [
                "Decrypt"
            ],
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "Name": "IssuedThroughACM",
            "RetiringPrincipal": "acm.us-west-2.amazonaws.com",
```

```
            "GranteePrincipal": "tlsconnectionmanager.amazonaws.com",
            "GrantId":
 "f020fe75197b93991dc8491d6f19dd3cebb24ee62277a05914386724f3d48758",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "CreationDate": "2023-02-06T23:30:50Z",
            "Constraints": {
                "encryptionContextEquals": {
                  "aws:acm:arn": "arn:aws:acm:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                  "aws:vpc-lattice:arn": "arn:aws:vpc-lattice:us-
west-2:111122223333:service/svc-0b23c1234567890ab"
                }
            }
        }
    ]
}
```

Dalam `ListGrants` contoh di atas, prinsipal penerima hibah adalah TLS Connection Manager dan konteks enkripsi memiliki layanan VPC Lattice ARN.

Dekripsi

VPC Lattice menggunakan TLS Connection Manager untuk memanggil `Decrypt` operasi untuk mendekripsi kunci pribadi Anda untuk melayani koneksi TLS di layanan VPC Lattice Anda. Anda dapat melihat `Decrypt` operasi sebagai peristiwa dalam riwayat CloudTrailAcara, Dekripsi.

Berikut ini adalah contoh catatan peristiwa dalam riwayat CloudTrail acara untuk `Decrypt` operasi.

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "tlsconnectionmanager.amazonaws.com"
    },
    "eventTime": "2023-02-07T00:07:23Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "tlsconnectionmanager.amazonaws.com",
    "userAgent": "tlsconnectionmanager.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
```

```
            "aws:acm:arn": "arn:aws:acm:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "aws:vpc-lattice:arn": "arn:aws:vpc-lattice:us-west-2:111122223333:service/
svc-0b23c1234567890ab"
        },
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
    },
    "responseElements": null,
    "requestID": "12345126-30d5-4b28-98b9-9153da559963",
    "eventID": "abcde202-ba1a-467c-b4ba-f729d45ae521",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "sharedEventID": "abcde202-ba1a-467c-b4ba-f729d45ae521",
    "eventCategory": "Management"
}
```


Manajemen identitas dan akses untuk Amazon VPC Lattice

Bagian berikut menjelaskan bagaimana Anda dapat menggunakan AWS Identity and Access Management (IAM) untuk membantu mengamankan sumber daya VPC Lattice Anda, dengan mengontrol siapa yang dapat melakukan tindakan VPC Lattice API.

Topik

- Bagaimana Amazon VPC Lattice bekerja dengan IAM
- Izin API Amazon VPC Lattice
- Kebijakan berbasis identitas untuk Amazon VPC Lattice
- Menggunakan peran terkait layanan untuk Amazon VPC Lattice
- AWS kebijakan terkelola untuk Amazon VPC Lattice

Bagaimana Amazon VPC Lattice bekerja dengan IAM

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

Fitur IAM	Dukungan VPC Lattice
Kebijakan berbasis identitas	Ya
Kebijakan berbasis sumber daya	Ya
Tindakan kebijakan	Ya
Sumber daya kebijakan	Ya
Kunci kondisi kebijakan	Ya
ACLs	Tidak
ABAC (tanda dalam kebijakan)	Ya
Kredensial sementara	Ya
Peran layanan	Tidak
Peran terkait layanan	Ya

Untuk tampilan tingkat tinggi tentang cara kerja Kisi VPC dan layanan AWS lainnya dengan sebagian besar fitur IAM, AWS lihat layanan yang bekerja dengan IAM di Panduan Pengguna IAM.

Kebijakan berbasis identitas untuk VPC Lattice

Mendukung kebijakan berbasis identitas: Ya

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, seperti pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat Tentukan izin IAM kustom dengan kebijakan terkelola pelanggan dalam Panduan Pengguna IAM.

Dengan kebijakan berbasis identitas IAM, Anda dapat menentukan secara spesifik apakah tindakan dan sumber daya diizinkan atau ditolak, serta kondisi yang menjadi dasar dikabulkan atau ditolaknya tindakan tersebut. Untuk mempelajari semua elemen yang dapat Anda gunakan dalam kebijakan JSON, lihat Referensi elemen kebijakan JSON IAM dalam Panduan Pengguna IAM.

Kebijakan berbasis sumber daya dalam VPC Lattice

Mendukung kebijakan berbasis sumber daya: Ya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. AWS Dalam AWS layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu dari layanan tersebut. AWS Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh principal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus menentukan principal dalam kebijakan berbasis sumber daya.

VPC Lattice mendukung kebijakan auth, kebijakan berbasis sumber daya yang memungkinkan Anda mengontrol akses ke layanan di jaringan layanan Anda. Untuk informasi selengkapnya, lihat Kontrol akses ke layanan VPC Lattice menggunakan kebijakan autentikasi.

VPC Lattice juga mendukung kebijakan izin berbasis sumber daya untuk integrasi dengan. AWS Resource Access Manager Anda dapat menggunakan kebijakan berbasis sumber daya ini untuk memberikan izin mengelola konektivitas ke AWS akun atau organisasi lain untuk layanan, konfigurasi sumber daya, dan jaringan layanan. Untuk informasi selengkapnya, lihat Membagikan entitas VPC Lattice Anda.

Tindakan kebijakan untuk VPC Lattice

Mendukung tindakan kebijakan: Ya

Dalam pernyataan kebijakan IAM, Anda dapat menentukan tindakan API apa pun dari layanan apa pun yang mendukung IAM. Untuk VPC Lattice, gunakan awalan berikut dengan nama aksi API:. `vpc-lattice:` Misalnya:`vpc-lattice:CreateService`,`vpc-lattice:CreateTargetGroup`, dan`vpc-lattice:PutAuthPolicy`.

Untuk menentukan beberapa tindakan dalam satu pernyataan, pisahkan dengan koma, sebagai berikut:

```
"Action": [ "vpc-lattice:action1", "vpc-lattice:action2" ]
```

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard. Misalnya, Anda dapat menentukan semua tindakan yang namanya dimulai dengan kata`Get`, sebagai berikut:

```
"Action": "vpc-lattice:Get*"
```

Untuk daftar lengkap tindakan VPC Lattice API, lihat Tindakan yang ditentukan oleh Amazon VPC Lattice dalam Referensi Otorisasi Layanan.

Sumber daya kebijakan untuk VPC Lattice

Mendukung sumber daya kebijakan: Ya

Dalam pernyataan kebijakan IAM, `Resource` elemen menentukan objek atau objek yang dicakup oleh pernyataan tersebut. Untuk VPC Lattice, setiap pernyataan kebijakan IAM berlaku untuk sumber daya yang Anda tentukan menggunakan mereka. ARNs

Format Amazon Resource Name (ARN) tertentu bergantung pada sumber daya. Saat Anda memberikan ARN, ganti *`italicized`* teks dengan informasi khusus sumber daya Anda.

- Akses langganan log:

```
"Resource": "arn:aws:vpc-lattice:region:account-id:accesslogsubscription/access-log-subscription-id"
```

- Pendengar:

```
"Resource": "arn:aws:vpc-lattice:region:account-id:service/service-id/
listener/listener-id"
```

- Gateway sumber daya

```
"Resource": "arn:aws:vpc-lattice:region:account-id:resourcegateway/resource-gateway-id"
```

- Konfigurasi sumber daya

```
"Resource": "arn:aws:vpc-lattice:region:account-id:resourceconfiguration/resource-configuration-id"
```

- Aturan:

```
"Resource": "arn:aws:vpc-lattice:region:account-id:service/service-id/
listener/listener-id/rule/rule-id"
```

- Layanan:

```
"Resource": "arn:aws:vpc-lattice:region:account-id:service/service-id"
```

- Jaringan layanan:

```
"Resource": "arn:aws:vpc-lattice:region:account-id:servicenetwork/service-network-id"
```

- Asosiasi layanan jaringan layanan:

```
"Resource": "arn:aws:vpc-lattice:region:account-
id:servicenetworkserviceassociation/service-network-service-association-id"
```

- Asosiasi konfigurasi sumber daya jaringan layanan

```
"Resource": "arn:aws:vpc-lattice:region:account-
id:servicenetworkresourceassociation/service-network-resource-association-id"
```

- Asosiasi VPC jaringan layanan:

```
"Resource": "arn:aws:vpc-lattice:region:account-
id:servicenetworkvpcassociation/service-network-vpc-association-id"
```

- Kelompok sasaran:

```
"Resource": "arn:aws:vpc-lattice:region:account-id:targetgroup/target-group-id"
```

Kunci kondisi kebijakan untuk VPC Lattice

Mendukung kunci kondisi kebijakan khusus layanan: Yes

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

Elemen `Condition` menentukan ketika pernyataan dieksekusi berdasarkan kriteria yang ditetapkan. Anda dapat membuat ekspresi bersyarat yang menggunakan operator kondisi, misalnya sama

dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta. Untuk melihat semua kunci kondisi AWS global, lihat kunci konteks kondisi AWS global di Panduan Pengguna IAM.

Untuk melihat daftar kunci kondisi Kisi VPC, lihat Kunci kondisi untuk Amazon VPC Lattice di Referensi Otorisasi Layanan.

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

Daftar kontrol akses (ACLs) di VPC Lattice

Mendukung ACLs: Tidak

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

Kontrol akses berbasis atribut (ABAC) dengan VPC Lattice

Mendukung ABAC (tanda dalam kebijakan): Ya

Kontrol akses berbasis atribut (ABAC) adalah strategi otorisasi yang menentukan izin berdasarkan atribut tanda. Anda dapat melampirkan tag ke entitas dan AWS sumber daya IAM, lalu merancang kebijakan ABAC untuk mengizinkan operasi saat tag prinsipal cocok dengan tag pada sumber daya.

Untuk mengendalikan akses berdasarkan tanda, berikan informasi tentang tanda di elemen kondisi dari kebijakan menggunakan kunci kondisi `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`.

Jika sebuah layanan mendukung ketiga kunci kondisi untuk setiap jenis sumber daya, nilainya adalah Ya untuk layanan tersebut. Jika suatu layanan mendukung ketiga kunci kondisi untuk hanya beberapa jenis sumber daya, nilainya adalah Parsial.

Untuk informasi selengkapnya tentang ABAC, lihat Tentukan izin dengan otorisasi ABAC dalam Panduan Pengguna IAM. Untuk melihat tutorial yang menguraikan langkah-langkah pengaturan ABAC, lihat Menggunakan kontrol akses berbasis atribut (ABAC) dalam Panduan Pengguna IAM.

Menggunakan kredensi sementara dengan VPC Lattice

Mendukung kredensial sementara: Ya

Kredensyal sementara menyediakan akses jangka pendek ke AWS sumber daya dan secara otomatis dibuat saat Anda menggunakan federasi atau beralih peran. AWS merekomendasikan agar Anda secara dinamis menghasilkan kredensyal sementara alih-alih menggunakan kunci akses jangka panjang. Untuk informasi selengkapnya, lihat Kredensial keamanan sementara di IAM dan Layanan AWS yang berfungsi dengan IAM dalam Panduan Pengguna IAM.

Peran layanan untuk VPC Lattice

Mendukung peran layanan: Tidak

Peran layanan adalah peran IAM yang diambil oleh sebuah layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

> **Warning**
>
> Mengubah izin untuk peran layanan dapat merusak fungsionalitas VPC Lattice. Edit peran layanan hanya jika VPC Lattice memberikan panduan untuk melakukannya.

Peran terkait layanan untuk VPC Lattice

Mendukung peran terkait layanan: Ya

Peran terkait layanan adalah jenis peran layanan yang ditautkan ke. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran terkait layanan.

Untuk informasi tentang membuat atau mengelola peran terkait layanan VPC Lattice, lihat. Menggunakan peran terkait layanan untuk Amazon VPC Lattice

Izin API Amazon VPC Lattice

Anda harus memberikan izin identitas IAM (seperti pengguna atau peran) untuk memanggil tindakan VPC Lattice API yang mereka butuhkan, seperti yang dijelaskan dalam. Tindakan kebijakan untuk VPC Lattice Selain itu, untuk beberapa tindakan VPC Lattice, Anda harus memberikan izin identitas IAM untuk memanggil tindakan tertentu dari yang lain. AWS APIs

Izin yang diperlukan untuk API

Saat memanggil tindakan berikut dari API, Anda harus memberikan izin kepada pengguna IAM untuk memanggil tindakan yang ditentukan.

`CreateResourceConfiguration`

- `vpc-lattice:CreateResourceConfiguration`
- `ec2:DescribeSubnets`
- `rds:DescribeDBInstances`
- `rds:DescribeDBClusters`

`CreateResourceGateway`

- `vpc-lattice:CreateResourceGateway`
- `ec2:AssignPrivateIpAddresses`
- `ec2:AssignIpv6Addresses`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`
- `ec2:DeleteNetworkInterface`
- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`

`DeleteResourceGateway`

- `vpc-lattice:DeleteResourceGateway`
- `ec2:DeleteNetworkInterface`

`UpdateResourceGateway`

- `vpc-lattice:UpdateResourceGateway`
- `ec2:AssignPrivateIpAddresses`
- `ec2:AssignIpv6Addresses`
- `ec2:UnassignPrivateIpAddresses`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`
- `ec2:DeleteNetworkInterface`

- ec2:DescribeNetworkInterfaces
- ec2:DescribeSecurityGroups
- ec2:DescribeSubnets
- ec2:ModifyNetworkInterfaceAttribute

CreateServiceNetworkResourceAssociation

- vpc-lattice:CreateServiceNetworkResourceAssociation
- ec2:AssignIpv6Addresses
- ec2:CreateNetworkInterface
- ec2:CreateNetworkInterfacePermission
- ec2:DescribeNetworkInterfaces

CreateServiceNetworkVpcAssociation

- vpc-lattice:CreateServiceNetworkVpcAssociation
- ec2:DescribeVpcs
- ec2:DescribeSecurityGroups(Hanya diperlukan ketika kelompok keamanan disediakan)

UpdateServiceNetworkVpcAssociation

- vpc-lattice:UpdateServiceNetworkVpcAssociation
- ec2:DescribeSecurityGroups(Hanya diperlukan ketika kelompok keamanan disediakan)

CreateTargetGroup

- vpc-lattice:CreateTargetGroup
- ec2:DescribeVpcs

RegisterTargets

- vpc-lattice:RegisterTargets
- ec2:DescribeInstances(Hanya INSTANCE diperlukan kapan tipe grup target)
- ec2:DescribeVpcs(Hanya diperlukan ketika INSTANCE atau IP tipe grup target)
- ec2:DescribeSubnets(Hanya diperlukan ketika INSTANCE atau IP tipe grup target)
- lambda:GetFunction(Hanya LAMBDA diperlukan kapan tipe grup target)
- lambda:AddPermission(Hanya diperlukan jika grup target belum memiliki izin untuk menjalankan fungsi Lambda yang ditentukan)

DeregisterTargets

- vpc-lattice:DeregisterTargets

`CreateAccessLogSubscription`

- `vpc-lattice:CreateAccessLogSubscription`
- `logs:GetLogDelivery`
- `logs:CreateLogDelivery`

`DeleteAccessLogSubscription`

- `vpc-lattice:DeleteAccessLogSubscription`
- `logs:DeleteLogDelivery`

`UpdateAccessLogSubscription`

- `vpc-lattice:UpdateAccessLogSubscription`
- `logs:UpdateLogDelivery`

Kebijakan berbasis identitas untuk Amazon VPC Lattice

Secara default, pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi sumber daya VPC Lattice. Untuk memberikan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM.

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM dengan menggunakan contoh dokumen kebijakan JSON ini, lihat Membuat kebijakan IAM (konsol) di Panduan Pengguna IAM.

Untuk detail tentang tindakan dan jenis sumber daya yang ditentukan oleh Kisi VPC, termasuk format ARNs untuk setiap jenis sumber daya, lihat Kunci Tindakan, Sumber Daya, dan Kondisi untuk Kisi VPC Amazon di Referensi Otorisasi Layanan.

Daftar Isi

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus sumber daya VPC Lattice di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin dengan hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat Kebijakan yang dikelola AWS atau Kebijakan yang dikelola AWS untuk fungsi tugas dalam Panduan Pengguna IAM.
- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan IAM untuk mengajukan izin, lihat Kebijakan dan izin dalam IAM dalam Panduan Pengguna IAM.
- Gunakan kondisi dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti CloudFormation. Untuk informasi selengkapnya, lihat Elemen kebijakan JSON IAM: Kondisi dalam Panduan Pengguna IAM.
- Gunakan IAM Access Analyzer untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. IAM Access Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat Validasi kebijakan dengan IAM Access Analyzer dalam Panduan Pengguna IAM.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Anda, Akun AWS aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan kondisi MFA pada kebijakan Anda. Untuk informasi selengkapnya, lihat Amankan akses API dengan MFA dalam Panduan Pengguna IAM.

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

Izin tambahan yang diperlukan untuk akses penuh

Untuk menggunakan AWS layanan lain yang terintegrasi dengan VPC Lattice dan seluruh rangkaian fitur VPC Lattice, Anda harus memiliki izin tambahan tertentu. Izin ini tidak termasuk dalam kebijakan `VPCLatticeFullAccess` terkelola karena risiko eskalasi hak istimewa wakil yang membingungkan.

Anda harus melampirkan kebijakan berikut ke peran Anda dan menggunakannya bersama dengan kebijakan `VPCLatticeFullAccess` terkelola.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "firehose:TagDeliveryStream",
                "lambda:AddPermission",
                "s3:PutBucketPolicy"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:PutResourcePolicy"
            ],
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:CalledVia": [
                        "vpc-lattice.amazonaws.com"
                    ]
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:AttachRolePolicy",
                "iam:PutRolePolicy"
```

```
            ],
            "Resource": "arn:aws:iam::*:role/aws-service-role/vpc-
lattice.amazonaws.com/AWSServiceRoleForVpcLattice"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:AttachRolePolicy",
                "iam:PutRolePolicy"
            ],
            "Resource": "arn:aws:iam::*:role/aws-service-role/
delivery.logs.amazonaws.com/AWSServiceRoleForLogDelivery*"
        }
    ]
}
```

Kebijakan ini memberikan izin tambahan berikut:

- `iam:AttachRolePolicy`: Memungkinkan Anda melampirkan kebijakan terkelola yang ditentukan ke peran IAM yang ditentukan.
- `iam:PutRolePolicy`: Memungkinkan Anda menambahkan atau memperbarui dokumen kebijakan sebaris yang disematkan dalam peran IAM yang ditentukan.
- `s3:PutBucketPolicy`: Memungkinkan Anda menerapkan kebijakan bucket ke bucket Amazon S3.
- `firehose:TagDeliveryStream`: Memungkinkan Anda menambahkan atau memperbarui tag untuk aliran pengiriman Firehose.

Contoh kebijakan berbasis identitas untuk VPC Lattice

Topik

- Contoh kebijakan: Mengelola asosiasi VPC ke jaringan layanan
- Contoh kebijakan: Buat asosiasi layanan ke jaringan layanan
- Contoh kebijakan: Tambahkan tag ke sumber daya
- Contoh kebijakan: Membuat peran terkait layanan

Contoh kebijakan: Mengelola asosiasi VPC ke jaringan layanan

Contoh berikut menunjukkan kebijakan yang memberi pengguna kebijakan ini izin untuk membuat, memperbarui, dan menghapus asosiasi VPC ke jaringan layanan, tetapi hanya untuk VPC dan jaringan layanan yang ditentukan dalam kondisi tersebut. Untuk informasi selengkapnya tentang menentukan kunci kondisi, lihatKunci kondisi kebijakan untuk VPC Lattice.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "vpc-lattice:CreateServiceNetworkVpcAssociation",
            "vpc-lattice:UpdateServiceNetworkVpcAssociation",
            "vpc-lattice:DeleteServiceNetworkVpcAssociation"
         ],
         "Resource": [
            "*"
         ],
         "Condition": {
            "StringEquals": {
               "vpc-lattice:ServiceNetworkArn": "arn:aws:vpc-lattice:us-
west-2:123456789012:servicenetwork/sn-903004f88example",
               "vpc-lattice:VpcId": "vpc-1a2b3c4d"
            }
         }
      }
   ]
}
```

Contoh kebijakan: Buat asosiasi layanan ke jaringan layanan

Jika Anda tidak menggunakan tombol kondisi untuk mengontrol akses ke sumber daya VPC Lattice, Anda dapat menentukan sumber daya dalam `Resource` elemen untuk mengontrol akses sebagai gantinya. ARNs

Contoh berikut menunjukkan kebijakan yang membatasi asosiasi layanan ke jaringan layanan yang dapat dibuat oleh pengguna dengan kebijakan ini dengan menentukan jaringan layanan dan layanan

yang dapat digunakan dengan tindakan `CreateServiceNetworkServiceAssociation` API. ARNs Untuk informasi selengkapnya tentang menentukan nilai ARN, lihat. Sumber daya kebijakan untuk VPC Lattice

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "vpc-lattice:CreateServiceNetworkServiceAssociation"
         ],
         "Resource": [
            "arn:aws:vpc-lattice:us-
west-2:123456789012:servicenetworkserviceassociation/*",
            "arn:aws:vpc-lattice:us-west-2:123456789012:service/
svc-04d5cc9b88example",
            "arn:aws:vpc-lattice:us-west-2:123456789012:servicenetwork/
sn-903004f88example"
         ]
      }
   ]
}
```

Contoh kebijakan: Tambahkan tag ke sumber daya

Contoh berikut menunjukkan kebijakan yang memberi pengguna izin kebijakan ini untuk membuat tag pada resource VPC Lattice.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "vpc-lattice:TagResource"
```

```
        ],
        "Resource": "arn:aws:vpc-lattice:us-west-2:123456789012:*/*"
      }
   ]
}
```

Contoh kebijakan: Membuat peran terkait layanan

VPC Lattice memerlukan izin untuk membuat peran terkait layanan saat pertama kali setiap pengguna di Anda membuat sumber daya VPC Lattice. Akun AWS Jika peran terkait layanan belum ada, VPC Lattice membuatnya di akun Anda. Peran terkait layanan memberikan izin ke VPC Lattice sehingga dapat memanggil orang lain atas nama Anda. Layanan AWS Untuk informasi selengkapnya, lihat the section called “Menggunakan Peran Terkait Layanan”.

Agar pembuatan peran otomatis berhasil, pengguna harus memiliki izin untuk tindakan `iam:CreateServiceLinkedRole` nyata.

```
"Action": "iam:CreateServiceLinkedRole"
```

Contoh berikut menunjukkan kebijakan yang memberi pengguna izin kebijakan ini untuk membuat peran terkait layanan untuk VPC Lattice.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": "iam:CreateServiceLinkedRole",
         "Resource": "arn:aws:iam::*:role/aws-service-role/vpc-
lattice.amazonaws.com/AWSServiceRoleForVpcLattice",
         "Condition": {
            "StringLike": {
               "iam:AWSServiceName":"vpc-lattice.amazonaws.com"
            }
         }
      }
   ]
}
```

Untuk informasi selengkapnya, lihat Izin peran terkait layanan dalam Panduan Pengguna IAM.

Menggunakan peran terkait layanan untuk Amazon VPC Lattice

Amazon VPC Lattice menggunakan peran terkait layanan untuk izin yang diperlukan untuk memanggil orang lain atas nama Anda. Layanan AWS Untuk informasi selengkapnya, lihat Peran terkait layanan di Panduan Pengguna IAM.

VPC Lattice menggunakan peran terkait layanan bernama. AWSServiceRoleForVpcLattice

Izin peran terkait layanan untuk VPC Lattice

Peran terkait layanan AWSServiceRoleForVpcLattice memercayai layanan berikut untuk mengambil peran tersebut:

- `vpc-lattice.amazonaws.com`

Kebijakan izin peran bernama AWSVpcLatticeServiceRolePolicy memungkinkan VPC Lattice CloudWatch mempublikasikan metrik di namespace. `AWS/VpcLattice` Untuk informasi selengkapnya, lihat AWSVpcLatticeServiceRolePolicydi Referensi Kebijakan AWS Terkelola.

Anda harus mengonfigurasi izin agar entitas IAM (seperti pengguna, grup, atau peran) dapat membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat the section called “Contoh kebijakan: Membuat peran terkait layanan”.

Membuat peran terkait layanan untuk VPC Lattice

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda membuat resource VPC Lattice di Konsol Manajemen AWS, the, atau API AWS CLI AWS , VPC Lattice membuat peran terkait layanan untuk Anda.

Jika Anda menghapus peran terkait layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda membuat resource VPC Lattice, VPC Lattice membuat peran yang ditautkan layanan untuk Anda lagi.

Mengedit peran terkait layanan untuk VPC Lattice

Anda dapat mengedit deskripsi AWSServiceRoleForVpcLatticemenggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit deskripsi peran terkait layanan di Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk VPC Lattice

Jika Anda tidak perlu lagi menggunakan Amazon VPC Lattice, kami sarankan Anda menghapus. AWSServiceRoleForVpcLattice

Anda dapat menghapus peran terkait layanan ini hanya setelah Anda menghapus semua sumber daya VPC Lattice di situs Anda. Akun AWS

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

Setelah menghapus peran yang ditautkan layanan, VPC Lattice akan membuat peran tersebut lagi saat Anda membuat resource VPC Lattice di VPC Lattice. Akun AWS

Wilayah yang Didukung untuk peran terkait layanan VPC Lattice

VPC Lattice mendukung penggunaan peran terkait layanan di semua Wilayah tempat layanan tersedia.

AWS kebijakan terkelola untuk Amazon VPC Lattice

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, pemutakhiran akan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui kebijakan AWS terkelola saat baru Layanan AWS diluncurkan atau operasi API baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat Kebijakan terkelola AWS dalam Panduan Pengguna IAM.

AWS kebijakan terkelola: VPCLattice FullAccess

Kebijakan ini menyediakan akses penuh ke Amazon VPC Lattice dan akses terbatas ke layanan dependen lainnya. Ini termasuk izin untuk melakukan hal berikut:

- ACM — Ambil sertifikat SSL/TLS ARN untuk nama domain kustom.
- CloudWatch — Lihat log akses dan data pemantauan.
- CloudWatch Log - Mengatur dan mengirim log akses ke CloudWatch Log.
- Amazon EC2 — Konfigurasikan antarmuka jaringan dan ambil informasi tentang instans EC2 dan. VPCs Ini digunakan untuk membuat konfigurasi sumber daya, gateway sumber daya, dan grup target, mengonfigurasi asosiasi entitas VPC Lattice, dan mendaftarkan target.
- Elastic Load Balancing — Ambil informasi tentang Application Load Balancer untuk mendaftarkannya sebagai target.
- Firehose — Mengambil informasi tentang aliran pengiriman yang digunakan untuk menyimpan log akses.
- Lambda — Ambil informasi tentang fungsi Lambda untuk mendaftarkannya sebagai target.
- Amazon RDS — Ambil informasi tentang cluster dan instans RDS.
- Amazon S3 - Ambil informasi tentang bucket S3 yang digunakan untuk menyimpan log akses.

Untuk melihat izin kebijakan ini, lihat VPCLatticeFullAccessdi Referensi Kebijakan AWS Terkelola.

Untuk menggunakan AWS layanan lain yang terintegrasi dengan VPC Lattice dan seluruh rangkaian fitur VPC Lattice, Anda harus memiliki izin tambahan tertentu. Izin ini tidak termasuk dalam kebijakan `VPCLatticeFullAccess` terkelola karena risiko eskalasi hak istimewa wakil yang membingungkan. Untuk informasi selengkapnya, lihat Izin tambahan yang diperlukan untuk akses penuh.

AWS kebijakan terkelola: VPCLattice ReadOnlyAccess

Kebijakan ini menyediakan akses hanya-baca ke Amazon VPC Lattice dan akses terbatas ke layanan dependen lainnya. Ini termasuk izin untuk melakukan hal berikut:

- ACM — Ambil sertifikat SSL/TLS ARN untuk nama domain kustom.
- CloudWatch — Lihat log akses dan data pemantauan.
- CloudWatch Log - Lihat informasi pengiriman log untuk langganan log akses.
- Amazon EC2 — Ambil informasi tentang instans EC2 dan VPCs untuk membuat grup target dan mendaftarkan target.

- Elastic Load Balancing — Mengambil informasi tentang Application Load Balancer.
- Firehose — Ambil informasi tentang aliran pengiriman untuk pengiriman log akses.
- Lambda - Lihat informasi tentang fungsi Lambda.
- Amazon RDS — Ambil informasi tentang cluster dan instans RDS.
- Amazon S3 - Ambil informasi tentang bucket S3 untuk pengiriman log akses.

Untuk melihat izin kebijakan ini, lihat VPCLatticeReadOnlyAccessdi Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: VPCLattice ServicesInvokeAccess

Kebijakan ini menyediakan akses untuk memanggil layanan Amazon VPC Lattice.

Untuk melihat izin kebijakan ini, lihat VPCLatticeServicesInvokeAccessdi Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSVpc LatticeServiceRolePolicy

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama AWSServiceRoleForVpcLatticeuntuk mengizinkan VPC Lattice melakukan tindakan atas nama Anda. Anda tidak dapat melampirkan kebijakan ini ke entitas IAM Anda. Untuk informasi selengkapnya, lihat Menggunakan peran terkait layanan untuk Amazon VPC Lattice.

Untuk melihat izin kebijakan ini, lihat AWSVpcLatticeServiceRolePolicydi Referensi Kebijakan AWS Terkelola.

VPC Lattice memperbarui kebijakan terkelola AWS

Lihat detail tentang pembaruan kebijakan AWS terkelola untuk VPC Lattice sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan umpan RSS untuk Panduan Pengguna VPC Lattice.

Ubah	Deskripsi	Date
VPCLatticeFullAccess	VPC Lattice menambahkan izin hanya-baca untuk mendeskripsikan kluster dan instance Amazon RDS.	Desember 1, 2024

Ubah	Deskripsi	Date
VPCLatticeReadOnlyAccess	VPC Lattice menambahkan izin hanya-baca untuk mendeskripsikan kluster dan instance Amazon RDS.	Desember 1, 2024
AWSVpcLatticeServiceRolePolicy	VPC Lattice menambahkan izin untuk memungkinkan VPC Lattice membuat antarmuka jaringan yang dikelola pemohon.	Desember 1, 2024
VPCLatticeFullAccess	VPC Lattice menambahkan kebijakan baru untuk memberikan izin akses penuh ke Amazon VPC Lattice dan akses terbatas ke layanan dependen lainnya.	31 Maret 2023
VPCLatticeReadOnlyAccess	VPC Lattice menambahkan kebijakan baru untuk memberikan izin akses hanya-baca ke Amazon VPC Lattice dan akses terbatas ke layanan dependen lainnya.	31 Maret 2023
VPCLatticeServicesInvokeAccess	VPC Lattice menambahkan kebijakan baru untuk memberikan akses ke layanan Amazon VPC Lattice.	31 Maret 2023

Ubah	Deskripsi	Date
AWSVpcLatticeServiceRolePolicy	VPC Lattice menambahkan izin ke peran terkait layananny a untuk memungkinkan VPC Lattice mempublikasikan metrik di namespace. CloudWatch `AWS/VpcLattice` `AWSVpcLatticeServiceRolePolicy` Kebijakan ini mencakup izin untuk memanggil tindakan CloudWatch PutMetricDataAPI. Untuk informasi selengkapnya, lihat Menggunakan peran terkait layanan untuk Amazon VPC Lattice.	Desember 5, 2022
VPC Lattice mulai melacak perubahan	VPC Lattice mulai melacak perubahan untuk kebijakan terkelola nya AWS .	Desember 5, 2022

Validasi kepatuhan untuk Amazon VPC Lattice

Auditor pihak ketiga menilai keamanan dan kepatuhan Amazon VPC Lattice sebagai bagian dari beberapa AWS program kepatuhan.

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program AWS Kepatuhan Program AWS .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. Untuk informasi selengkapnya tentang tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS, lihat Dokumentasi AWS Keamanan.

Akses Amazon VPC Lattice menggunakan titik akhir antarmuka ()AWS PrivateLink

Anda dapat membuat koneksi pribadi antara VPC dan Amazon VPC Lattice dengan membuat antarmuka VPC endpoint. Endpoint antarmuka didukung oleh AWS PrivateLink, teknologi yang memungkinkan Anda mengakses VPC Lattice secara pribadi APIs tanpa gateway internet, perangkat NAT, koneksi VPN, atau koneksi. Direct Connect Instans di VPC Anda tidak memerlukan alamat IP publik untuk berkomunikasi dengan VPC Lattice. APIs

Setiap titik akhir antarmuka diwakili oleh satu atau lebih antarmuka jaringan di subnet Anda.

Pertimbangan untuk titik akhir VPC antarmuka

Sebelum Anda menyiapkan antarmuka VPC endpoint untuk VPC Lattice, pastikan Anda meninjau Access melalui Panduan. Layanan AWS AWS PrivateLinkAWS PrivateLink

VPC Lattice mendukung panggilan ke semua tindakan API-nya dari VPC Anda.

Membuat antarmuka VPC endpoint untuk VPC Lattice

Anda dapat membuat titik akhir VPC untuk layanan VPC Lattice menggunakan konsol VPC Amazon atau (). AWS Command Line Interface AWS CLIUntuk informasi selengkapnya, lihat Membuat titik akhir VPC antarmuka di Panduan.AWS PrivateLink

Buat titik akhir VPC untuk VPC Lattice menggunakan nama layanan berikut:

`com.amazonaws.region.vpc-lattice`

Jika Anda mengaktifkan DNS pribadi untuk titik akhir, Anda dapat membuat permintaan API ke VPC Lattice menggunakan nama DNS default untuk Wilayah, misalnya,. `vpc-lattice.us-east-1.amazonaws.com`

Ketahanan di Amazon VPC Lattice

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.

Keamanan infrastruktur di Amazon VPC Lattice

Sebagai layanan terkelola, Amazon VPC Lattice dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat Keamanan AWS Cloud. Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat Perlindungan Infrastruktur dalam Kerangka Kerja yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses VPC Lattice melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Sandi cocok dengan sistem kerahasiaan maju sempurna (perfect forward secrecy, PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Memantau Amazon VPC Lattice

Gunakan fitur di bagian ini untuk memantau jaringan layanan Amazon VPC Lattice, layanan, grup target, dan koneksi VPC.

Konten

CloudWatch metrik untuk Amazon VPC Lattice

Amazon VPC Lattice mengirimkan data yang terkait dengan grup target dan layanan Anda ke Amazon CloudWatch, dan memprosesnya menjadi metrik hampir real-time yang dapat dibaca. Metrik ini disimpan selama 15 bulan, sehingga Anda dapat mengakses informasi historis dan mendapatkan perspektif yang lebih baik tentang kinerja aplikasi atau layanan web Anda. Anda juga dapat mengatur alarm yang mengawasi ambang batas tertentu dan mengirim pemberitahuan atau mengambil tindakan ketika ambang batas tersebut terpenuhi. Untuk informasi selengkapnya, lihat Panduan CloudWatch Pengguna Amazon.

Amazon VPC Lattice menggunakan peran terkait layanan di AWS akun Anda untuk mengirim metrik ke Amazon. CloudWatch Untuk informasi selengkapnya, lihat Menggunakan peran terkait layanan untuk Amazon VPC Lattice.

Daftar Isi

Lihat CloudWatch metrik Amazon

Anda dapat melihat CloudWatch metrik Amazon untuk grup dan layanan target menggunakan CloudWatch konsol atau AWS CLI.

Untuk melihat metrik menggunakan konsol CloudWatch

1. Buka CloudWatch konsol Amazon di https://console.aws.amazon.com/cloudwatch/.
2. Pada panel navigasi, silakan pilih Metrik.
3. Pilih `AWS/VpcLattice` namespace.
4. (Opsional) Untuk melihat metrik di semua dimensi, masukkan namanya di kolom pencarian.
5. (Opsional) Untuk memfilter metrik berdasarkan dimensi, pilih salah satu hal berikut:
   - Untuk hanya menampilkan metrik yang dilaporkan untuk grup target Anda, pilih Grup target. Untuk melihat metrik untuk satu grup target, masukkan namanya di kolom pencarian.
   - Untuk hanya menampilkan metrik yang dilaporkan untuk layanan Anda, pilih Layanan. Untuk melihat metrik untuk satu layanan, masukkan namanya di bidang pencarian.

Untuk melihat metrik menggunakan AWS CLI

Gunakan AWS CLI perintah CloudWatch daftar-metrik berikut untuk membuat daftar metrik yang tersedia:

```
aws cloudwatch list-metrics --namespace AWS/VpcLattice
```

Untuk informasi tentang masing-masing metrik dan dimensinya, lihat Metrik kelompok sasaran danMetrik Layanan.

Metrik kelompok sasaran

VPC Lattice secara otomatis menyimpan metrik yang terkait dengan grup target di namespace Amazon. `AWS/VpcLattice` CloudWatch Untuk informasi selengkapnya tentang kelompok sasaran, lihatGrup sasaran di VPC Lattice.

Dimensi

Untuk memfilter metrik untuk grup target, gunakan dimensi berikut:

- `AvailabilityZone`
- `TargetGroup`

Metrik	Deskripsi	TargetGroup Protokol
`TotalConnectionCount`	Total koneksi. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`
`ActiveConnectionCount`	Koneksi aktif. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`

Metrik	Deskripsi	TargetGroup Protokol
`ConnectionErrorCount`	Kegagalan koneksi total. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`
`HTTP1_ConnectionCount`	Total koneksi HTTP/1.1. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS`

Metrik	Deskripsi	TargetGroup Protokol
`HTTP2_ConnectionCount`	Total koneksi HTTP/2. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS`
`ConnectionTimeoutCount`	Total koneksi menghubungkan batas waktu. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`

Metrik	Deskripsi	TargetGroup Protokol
`TotalReceivedConnectionBytes`	Total byte koneksi yang diterima. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`
`TotalSentConnectionBytes`	Total byte koneksi terkirim. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`

Metrik	Deskripsi	TargetGroup Protokol
`TotalRequestCount`	Total permintaan. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS`
`ActiveRequestCount`	Total permintaan aktif. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS`

Metrik	Deskripsi	TargetGroup Protokol
`RequestTime`	Minta waktu ke byte terakhir dalam milidetik. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah `Average` dan `pNN.NN` (persentil).	`HTTP, HTTPS`

Metrik	Deskripsi	TargetGroup Protokol
`HTTPCode_2XX_Count`, `HTTPCode_3XX_Count`, `HTTPCode_4XX_Count`, `HTTPCode_5XX_Count`	Kode respons HTTP agregat. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS`

Metrik	Deskripsi	TargetGroup Protokol
`TLSConnectionErrorCount`	Total kesalahan koneksi TLS tidak termasuk verifikasi sertifikat yang gagal. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`

Metrik	Deskripsi	TargetGroup Protokol
`TotalTLSConnectionHandshakeCount`	Total jabat tangan koneksi TLS yang berhasil. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.	`HTTP, HTTPS, TCP`

Metrik Layanan

VPC Lattice secara otomatis menyimpan metrik yang terkait dengan layanan di namespace Amazon. `AWS/VpcLattice` CloudWatch Untuk informasi selengkapnya tentang layanan, lihatLayanan di VPC Lattice.

Dimensi

Untuk memfilter metrik untuk grup target, gunakan dimensi berikut:

- `AvailabilityZone`
- `Service`

Metrik	Deskripsi
`RequestTimeoutCount`	Total permintaan yang waktunya habis menunggu respons.

Metrik	Deskripsi
	Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah `Sum`.
`TotalRequestCount`	Total permintaan. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah `Sum`.

Metrik	Deskripsi
`RequestTime`	Minta waktu dalam milidetik. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah `Average` dan `pNN.NN` (persentil).
`HTTPCode_2XX_Count`, `HTTPCode_3XX_Count`, `HTTPCode_4XX_Count`, `HTTPCode_5XX_Count`	Kode respons HTTP agregat. Kriteria pelaporan • Selalu dilaporkan (apakah itu nilai nol atau bukan nol) sejak sumber daya menerima lalu lintas. Frekuensi pelaporan • Sekali semenit. Statistik • Statistik yang paling berguna adalah`Sum`.

Akses log untuk Amazon VPC Lattice

Log akses menangkap informasi terperinci tentang layanan VPC Lattice dan konfigurasi sumber daya Anda. Anda dapat menggunakan log akses ini untuk menganalisis pola lalu lintas dan mengaudit

semua layanan di jaringan. Untuk layanan VPC Lattice, kami menerbitkan `VpcLatticeAccessLogs` dan untuk konfigurasi sumber daya, kami menerbitkan `VpcLatticeResourceAccessLogs` yang perlu dikonfigurasi secara terpisah.

Log akses bersifat opsional dan dinonaktifkan secara default. Setelah Anda mengaktifkan log akses, Anda dapat menonaktifkannya kapan saja.

Harga

Biaya berlaku ketika log akses dipublikasikan. Log yang diterbitkan AWS secara native atas nama Anda disebut vended logs. Untuk informasi selengkapnya tentang harga untuk log penjual, lihat CloudWatch Harga Amazon, pilih Log, dan lihat harga di bawah Log Penjual.

Daftar Isi

Izin IAM diperlukan untuk mengaktifkan log akses

Untuk mengaktifkan log akses dan mengirim log ke tujuan mereka, Anda harus memiliki tindakan berikut dalam kebijakan yang dilampirkan pada pengguna, grup, atau peran IAM yang Anda gunakan.

JSON

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

```
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "vpc-lattice:CreateAccessLogSubscription",
                "vpc-lattice:GetAccessLogSubscription",
                "vpc-lattice:UpdateAccessLogSubscription",
                "vpc-lattice:DeleteAccessLogSubscription",
                "vpc-lattice:ListAccessLogSubscriptions"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Untuk informasi selengkapnya, lihat Menambahkan dan menghapus izin identitas IAM di AWS Identity and Access Management Panduan Pengguna.

Setelah memperbarui kebijakan yang dilampirkan ke pengguna, grup, atau peran IAM yang Anda gunakan, buka. Aktifkan log akses

Akses tujuan log

Anda dapat mengirim log akses ke tujuan berikut.

CloudWatch Log Amazon

- VPC Lattice biasanya mengirimkan log ke CloudWatch Log dalam waktu 2 menit. Namun, perlu diingat bahwa waktu pengiriman log yang sebenarnya adalah upaya terbaik dan mungkin ada latensi tambahan.
- Kebijakan sumber daya dibuat secara otomatis dan ditambahkan ke grup CloudWatch log jika grup log tidak memiliki izin tertentu. Untuk informasi selengkapnya, lihat Log yang dikirim ke CloudWatch Log di Panduan CloudWatch Pengguna Amazon.
- Anda dapat menemukan log akses yang dikirim ke CloudWatch bawah Grup Log di CloudWatch konsol. Untuk informasi selengkapnya, lihat Melihat data log yang dikirim ke CloudWatch Log di Panduan CloudWatch Pengguna Amazon.

Amazon S3

- VPC Lattice biasanya mengirimkan log ke Amazon S3 dalam waktu 6 menit. Namun, perlu diingat bahwa waktu pengiriman log yang sebenarnya adalah upaya terbaik dan mungkin ada latensi tambahan.
- Kebijakan bucket akan dibuat secara otomatis dan ditambahkan ke bucket Amazon S3 Anda jika bucket tidak memiliki izin tertentu. Untuk informasi selengkapnya, lihat Log yang dikirim ke Amazon S3 di CloudWatchPanduan Pengguna Amazon.
- Akses log yang dikirim ke Amazon S3 menggunakan konvensi penamaan berikut:

```
[bucket]/[prefix]/AWSLogs/[accountId]/VpcLattice/AccessLogs/[region]/[YYYY/
MM/DD]/[resource-id]/[accountId]_VpcLatticeAccessLogs_[region]_[resource-
id]_YYYYMMDDTHHmmZ_[hash].json.gz
```

- VpcLatticeResourceAccessLogs yang dikirim ke Amazon S3 gunakan konvensi penamaan berikut:

```
[bucket]/[prefix]/AWSLogs/[accountId]/VpcLattice/ResourceAccessLogs/[region]/[YYYY/
MM/DD]/[resource-id]/[accountId]_VpcLatticeResourceAccessLogs_[region]_[resource-
id]_YYYYMMDDTHHmmZ_[hash].json.gz
```

Amazon Data Firehose

- VPC Lattice biasanya mengirimkan log ke Firehose dalam waktu 2 menit. Namun, perlu diingat bahwa waktu pengiriman log yang sebenarnya adalah upaya terbaik dan mungkin ada latensi tambahan.
- Peran terkait layanan dibuat secara otomatis yang memberikan izin VPC Lattice untuk mengirim log akses ke. Amazon Data Firehose Agar pembuatan peran otomatis berhasil, pengguna harus memiliki izin untuk tindakan `iam:CreateServiceLinkedRole` nyata. Untuk informasi selengkapnya, lihat Log yang dikirimkan Amazon Data Firehose di Panduan CloudWatch Pengguna Amazon.
- Untuk informasi selengkapnya tentang melihat log yang dikirimkan Amazon Data Firehose, lihat Memantau Aliran Data Amazon Kinesis Amazon Data Firehose di Panduan Pengembang.

Aktifkan log akses

Selesaikan prosedur berikut untuk mengonfigurasi log akses untuk menangkap dan mengirimkan log akses ke tujuan yang Anda pilih.

Daftar Isi

Aktifkan log akses menggunakan konsol

Anda dapat mengaktifkan log akses untuk jaringan layanan, layanan, atau konfigurasi sumber daya selama pembuatan. Anda juga dapat mengaktifkan log akses setelah membuat jaringan layanan, layanan, atau konfigurasi sumber daya seperti yang dijelaskan dalam prosedur berikut.

Untuk membuat layanan dasar menggunakan konsol

1. Buka konsol Amazon VPC di. https://console.aws.amazon.com/vpc/
2. Pilih jaringan layanan, layanan, atau konfigurasi sumber daya.
3. Pilih Tindakan, Edit pengaturan log.
4. Aktifkan sakelar sakelar Access logs.
5. Tambahkan tujuan pengiriman untuk log akses Anda sebagai berikut:
   - Pilih Grup CloudWatch log dan pilih grup log. Untuk membuat grup log, pilih Buat grup log masuk CloudWatch.
   - Pilih bucket S3 dan masukkan path bucket S3, termasuk awalan apa pun. Untuk mencari bucket S3 Anda, pilih Browse S3.
   - Pilih aliran pengiriman Kinesis Data Firehose dan pilih aliran pengiriman. Untuk membuat aliran pengiriman, pilih Buat aliran pengiriman di Kinesis.
6. Pilih Simpan perubahan.

Aktifkan log akses menggunakan AWS CLI

Gunakan perintah CLI create-access-log-subscriptionuntuk mengaktifkan log akses untuk jaringan layanan atau layanan.

Permintaan pelacakan

VPC Lattice mendukung pelacakan permintaan dan korelasi di seluruh klien, target, dan log untuk observabilitas dan debugging dengan header. x-amzn-requestid Header ini dapat diatur dan dikirim oleh klien atau dihasilkan oleh VPC Lattice dan dikirim ke target dan juga tersedia di log akses.

Perilaku default

- VPC Lattice secara otomatis menghasilkan header ini untuk setiap permintaan.
- Nilainya adalah pengidentifikasi yang dihasilkan secara acak (gaya UUID secara default).
- Pengidentifikasi yang dihasilkan adalah:
  - Disebarkan ke target hilir.
  - Dikembalikan dalam header respons ke klien.
  - Masuk log akses

Contoh (respons default)

Berikut ini adalah contoh respons yang dikirim ke klien dengan perilaku default VPC Lattice menghasilkan nilai acak untuk header valu eof. x-amzn-requestid

```
{
    "HTTP/1.1 200 OK
    x-amzn-requestid: a9f2c7a1-6b4f-4c79-9e87-ff5a1234a001"
}
```

Klien menetapkan nilai

- Klien secara opsional dapat mengatur header ini pada permintaan yang masuk untuk mengganti nilai yang dihasilkan secara otomatis.
- Pertimbangan-pertimbangan
  - Nilai header tidak perlu mengikuti format UUID.
  - Jika nilai header melebihi 512 byte, VPC Lattice akan memotongnya menjadi 512.

- Ketika berhasil diganti, nilai header yang diberikan akan:
  - Muncul di header respons
  - Disebarkan ke target
  - Muncul di log akses dan metrik

Contoh (timpa persyaratan klien)

Berikut ini adalah contoh reqeust yang dikirim oleh klien dengan nilai header.

```
{
    "GET /my-service/endpoint HTTP/1.1
    Host: my-api.example.com
    x-amzn-requestid: trace-request-foobar"
}
```

Contoh (respons penggantian default)

Berikut ini adalah contoh respons yang dikirim ke klien dengan nilai yang diganti.

```
{
    "HTTP/1.1 200 OK
    x-amzn-requestid: trace-request-foobar"
}
```

Akses isi log

Tabel berikut menjelaskan bidang entri log akses.

Bidang	Deskripsi	Format
`callerPrincipalTags`	PrincipalTags Dalam permintaan.	JSON
`hostHeader`	Header otoritas permintaan.	string
`sslCipher`	Nama OpenSSL untuk set cipher yang digunakan untuk membangun koneksi TLS klien.	string
`serviceNetworkArn`	Jaringan layanan ARN.	arn:aws:vpc-kisi: ::service network/ *`region account id`*
`resolvedUser`	ARN pengguna saat otentikasi diaktifkan dan otentikasi dilakukan.	null | ARN | “Anonim” | “Tidak diketahui”
`authDeniedReason`	Alasan bahwa akses ditolak ketika otentikasi diaktifkan.	null | “Layanan” | “Jaringan” | “Identitas”

Bidang	Deskripsi	Format
`requestMethod`	Header metode permintaan.	string
`targetGroupArn`	Grup host target tempat host target berada.	string
`tlsVersion`	Versi TLS.	TLSv*x*
`userAgent`	Header user-agent.	string
`serverNameIndication`	[Hanya HTTPS] Nilai yang ditetapkan pada soket koneksi ssl untuk Indikasi Nama Server (SNI).	string
`destinationVpcId`	ID VPC tujuan.	vpc- *xxxxxxxx*
`sourceIpPort`	Alamat IP dan:port sumber.	*ip*:*port*
`targetIpPort`	Alamat IP dan port target.	*ip*:*port*
`serviceArn`	Layanan ARN.	arn:aws:vpc-kisi: ::layanan /*regionaccountid*
`sourceVpcId`	ID VPC sumber.	vpc- *xxxxxxxx*
`requestPath`	Jalur permintaan.	LatticePath?:*path*
`startTime`	Waktu mulai permintaan.	*YYYY*- *MM* - *DD* T*HH*:*MM*: *SS* Z
`protocol`	Protokol. Saat ini HTTP/1.1 atau HTTP/2.	string
`responseCode`	Kode respons HTTP. Hanya kode respons untuk header akhir yang dicatat. Untuk informasi selengkapnya, lihat Memecahkan masalah log akses.	integer

Bidang	Deskripsi	Format
`bytesReceived`	Body dan header byte diterima.	integer
`bytesSent`	Body dan header byte dikirim.	integer
`duration`	Total durasi dalam milidetik permintaan dari waktu mulai hingga byte terakhir keluar.	integer
`requestToTargetDuration`	Total durasi dalam milidetik permintaan dari waktu mulai hingga byte terakhir yang dikirim ke target.	integer
`responseFromTargetDuration`	Total durasi dalam milidetik permintaan dari byte pertama yang dibaca dari host target ke byte terakhir yang dikirim ke klien.	integer
`grpcResponseCode`	Kode respons gRPC. Untuk informasi selengkapnya, lihat Kode status dan penggunaannya di gRPC. Bidang ini dicatat hanya jika layanan mendukung gRPC.	integer
`requestId`	Ini pengidentifikasi unik secara otomatis disertakan dalam tanggapan sebagai nilai x-amzn-requestid header. Ini memungkinkan korelasi permintaan di seluruh klien, target, dan log untuk observabilitas dan debugging.	string

Bidang	Deskripsi	Format
`callerPrincipal`	Prinsipal yang diautentikasi.	string
`callerX509SubjectCN`	Nama subjek (CN).	string
`callerX509IssuerOU`	Penerbit (OU).	string
`callerX509SANNameCN`	Alternatif penerbit (Nama/CN).	string
`callerX509SANDNS`	Nama alternatif subjek (DNS).	string
`callerX509SANURI`	Nama alternatif subjek (URI).	string
`sourceVpcArn`	ARN dari VPC tempat permintaan itu berasal.	arn:aws:e c2: ::vpc/*regionaccountid*

Bidang	Deskripsi	Format
`failureReason`	Menunjukkan alasan permintaan gagal. Nilai yang mungkin adalah sebagai berikut: • `TargetConnectionError` - Permintaan gagal terhubung ke target di grup target. • `TargetProtocolError` - Target tidak merespon dengan data yang valid. Ini mungkin menunjukkan target memiliki catatan TLS yang tidak valid atau menggunakan protokol grup target yang tidak valid. • `TargetDataTimeout` - Batas waktu idle tercapai. • `TargetConnectionClosed` - Target menutup koneksi sebelum menyelesaikan respons. • `ClientConnectionClosed` - Klien menutup koneksi sebelum menerima respons lengkap. • `ClientRateLimited` - Klien melebihi batas koneksi dan VPC Lattice membatasi tarif. • `ClientAccessDenied` - VPC Lattice ditolak akses	string

Bidang	Deskripsi	Format
	ke sumber daya. Gunakan `authDeniedReason` untuk informasi lebih lanjut tentang mengapa VPC Lattice menolak akses. • `ClientProtocolError` - Klien mengirim data yang tidak dipahami. Ini mungkin menunjukkan klien menggunakan catatan TLS yang tidak valid atau protokol yang tidak valid. • `ConnectionDurationExceeded` - Koneksi mencapai batas durasi koneksi maksimum. • `InternalError` - Terjadi kesalahan internal saat memproses permintaan.	

Contoh

Berikut ini adalah contoh entri log.

```
{
    "callerPrincipalTags" : "{ "TagA": "ValA", "TagB": "ValB", ... }",
    "hostHeader": "example.com",
    "sslCipher": "-",
    "serviceNetworkArn": "arn:aws:vpc-lattice:us-west-2:123456789012:servicenetwork/
svn-1a2b3c4d",
    "resolvedUser": "Unknown",
    "authDeniedReason": "null",
    "requestMethod": "GET",
    "targetGroupArn": "arn:aws:vpc-lattice:us-west-2:123456789012:targetgroup/
tg-1a2b3c4d",
    "tlsVersion": "-",
```

```
    "userAgent": "-",
    "serverNameIndication": "-",
    "destinationVpcId": "vpc-0abcdef1234567890",
    "sourceIpPort": "178.0.181.150:80",
    "targetIpPort": "131.31.44.176:80",
    "serviceArn": "arn:aws:vpc-lattice:us-west-2:123456789012:service/svc-1a2b3c4d",
    "sourceVpcId": "vpc-0abcdef1234567890",
    "requestPath": "/billing",
    "startTime": "2023-07-28T20:48:45Z",
    "protocol": "HTTP/1.1",
    "responseCode": 200,
    "bytesReceived": 42,
    "bytesSent": 42,
    "duration": 375,
    "requestToTargetDuration": 1,
    "responseFromTargetDuration": 1,
    "grpcResponseCode": 1,
    "requestId": "a9f2c7a1-6b4f-4c79-9e87-ff5a1234a001"
}
```


Isi log akses sumber daya

Tabel berikut menjelaskan bidang entri log akses sumber daya.

Bidang	Deskripsi	Format
`serviceNetworkArn`	Jaringan layanan ARN.	arn: *partition* vpc-kisi: ::servicenetwork/ *region account id*
`serviceNetworkResourceAssociationId`	ID sumber daya jaringan layanan.	*snra-xxx*
`vpcEndpointId`	Endpoint ID yang digunakan untuk mengakses sumber daya.	string
`sourceVpcArn`	Sumber VPC ARN atau VPC dari mana koneksi dimulai.	string

Bidang	Deskripsi	Format
`resourceConfigurationArn`	ARN dari konfigurasi sumber daya yang diakses.	string
`protocol`	Protokol yang digunakan untuk berkomunikasi dengan konfigurasi sumber daya. Saat ini hanya tcp yang didukung.	string
`sourceIpPort`	Alamat IP dan port sumber yang memulai koneksi.	*ip*:*port*
`destinationIpPort`	Alamat IP dan port tempat koneksi dimulai. Ini akan menjadi IP SN-E/SN-A.	*ip*:*port*
`gatewayIpPort`	Alamat IP dan port yang digunakan oleh gateway sumber daya untuk mengakses sumber daya.	*ip*:*port*
`resourceIpPort`	Alamat IP dan port sumber daya.	*ip*:*port*

Contoh

Berikut ini adalah contoh entri log.

```
{
    "eventTimestamp": "2024-12-02T10:10:10.123Z",
    "serviceNetworkArn": "arn:aws:vpc-lattice:us-west-2:1234567890:servicenetwork/
sn-1a2b3c4d",
    "serviceNetworkResourceAssociationId": "snra-1a2b3c4d",
    "vpcEndpointId": "vpce-01a2b3c4d",
    "sourceVpcArn": "arn:aws:ec2:us-west-2:1234567890:vpc/vpc-01a2b3c4d",
    "resourceConfigurationArn": "arn:aws:vpc-lattice:us-
west-2:0987654321:resourceconfiguration/rcfg-01a2b3c4d",
    "protocol": "tcp",
    "sourceIpPort": "172.31.23.56:44076",
```

```
    "destinationIpPort": "172.31.31.226:80",
    "gatewayIpPort": "10.0.28.57:49288",
    "resourceIpPort": "10.0.18.190:80"
}
```

Memecahkan masalah log akses

Bagian ini berisi penjelasan tentang kode kesalahan HTTP yang mungkin Anda lihat di log akses.

Kode kesalahan	Kemungkinan penyebab
HTTP 400: Permintaan Buruk	• Klien mengirim permintaan cacat yang tidak memenuhi spesifikasi HTTP. • Header permintaan melebihi 60K untuk seluruh header permintaan atau lebih dari 100 header. • Klien menutup koneksi sebelum mengirim badan permintaan lengkap.
HTTP 403: Terlarang	Otentikasi telah dikonfigurasi untuk layanan, tetapi permintaan yang masuk tidak diautentikasi atau diotorisasi.
HTTP 404: Layanan Tidak Ada	Anda mencoba untuk terhubung ke layanan yang tidak ada atau tidak terdaftar ke jaringan layanan yang tepat.
HTTP 500: Kesalahan Server Internal	VPC Lattice telah mengalami kesalahan, seperti kegagalan untuk terhubung ke target.
HTTP 502: Gerbang Buruk	VPC Lattice mengalami kesalahan.

CloudTrail log untuk Amazon VPC Lattice

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

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

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

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

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

CloudTrail jalan setapak

Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Semua jalur yang dibuat menggunakan Konsol Manajemen AWS Multi-region. Anda dapat membuat jalur Single-region atau Multi-region dengan menggunakan. AWS CLI Membuat jejak Multi-wilayah disarankan karena Anda menangkap aktivitas Wilayah AWS di semua akun Anda. Jika Anda membuat jejak wilayah Tunggal, Anda hanya dapat melihat peristiwa yang dicatat di jejak. AWS Region Untuk informasi selengkapnya tentang jejak, lihat Membuat jejak untuk Anda Akun AWS dan Membuat jejak untuk organisasi di Panduan AWS CloudTrail Pengguna.

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

CloudTrail Menyimpan data acara danau

CloudTrail Lake memungkinkan Anda menjalankan kueri berbasis SQL pada acara Anda. CloudTrail Lake mengonversi peristiwa yang ada dalam format JSON berbasis baris ke format Apache ORC. ORC adalah format penyimpanan kolumnar yang dioptimalkan untuk

pengambilan data dengan cepat. Peristiwa digabungkan ke dalam penyimpanan data peristiwa, yang merupakan kumpulan peristiwa yang tidak dapat diubah berdasarkan kriteria yang Anda pilih dengan menerapkan pemilih acara tingkat lanjut. Penyeleksi yang Anda terapkan ke penyimpanan data acara mengontrol peristiwa mana yang bertahan dan tersedia untuk Anda kueri. Untuk informasi lebih lanjut tentang CloudTrail Danau, lihat Bekerja dengan AWS CloudTrail Danau di Panduan AWS CloudTrail Pengguna.

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

Untuk memantau tindakan tambahan, gunakan log akses. Untuk informasi selengkapnya, lihat Log akses.

Acara manajemen VPC Lattice di CloudTrail

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

Amazon VPC Lattice mencatat operasi bidang kontrol VPC Lattice sebagai peristiwa manajemen. Untuk daftar operasi bidang kontrol Amazon VPC Lattice yang dicatat oleh VPC Lattice, lihat CloudTrail Referensi API Amazon VPC Lattice.

Contoh acara VPC Lattice

Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang operasi API yang diminta, tanggal dan waktu operasi, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, sehingga peristiwa tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan CloudTrail peristiwa untuk CreateServiceoperasi.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "abcdef01234567890",
```

```
    "arn": "arn:abcdef01234567890",
    "accountId": "abcdef01234567890",
    "accessKeyId": "abcdef01234567890",
    "sessionContext": {
        "sessionIssuer": {
            "type": "Role",
            "principalId": "abcdef01234567890",
            "arn": "arn:abcdef01234567890",
            "accountId": "abcdef01234567890",
            "userName": "abcdef01234567890"
        },
        "webIdFederationData": {},
        "attributes": {
            "creationDate": "2022-08-16T03:34:54Z",
            "mfaAuthenticated": "false"
        }
    }
  },
  "eventTime": "2022-08-16T03:36:12Z",
  "eventSource": "vpc-lattice.amazonaws.com",
  "eventName": "CreateService",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "abcdef01234567890",
  "userAgent": "abcdef01234567890",
  "requestParameters": {
    "name": "rates-service"
  },
  "responseElements": {
    "name": "rates-service",
    "id": "abcdef01234567890",
    "arn": "arn:abcdef01234567890",
    "status": "CREATE_IN_PROGRESS"
  },
  "requestID": "abcdef01234567890",
  "eventID": "abcdef01234567890",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "abcdef01234567890",
  "eventCategory": "Management"
}
```


Contoh berikut menunjukkan CloudTrail peristiwa untuk DeleteServiceoperasi.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "abcdef01234567890",
    "arn": "arn:ABCXYZ123456",
    "accountId": "abcdef01234567890",
    "accessKeyId": "abcdef01234567890",
    "sessionContext": {
        "sessionIssuer": {
            "type": "Role",
            "principalId": "abcdef01234567890",
            "arn": "arn:aws:iam::AIDACKCEVSQ6C2EXAMPLE:role/Admin",
            "accountId": "abcdef01234567890",
            "userName": "Admin"
        },
        "webIdFederationData": {},
        "attributes": {
            "creationDate": "2022-10-27T17:42:36Z",
            "mfaAuthenticated": "false"
        }
    }
  },
  "eventTime": "2022-10-27T17:56:41Z",
  "eventSource": "vpc-lattice.amazonaws.com",
  "eventName": "DeleteService",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "72.21.198.64",
  "userAgent": "abcdef01234567890",
  "requestParameters": {
    "serviceIdentifier": "abcdef01234567890"
  },
  "responseElements": {
    "name": "test",
    "id": "abcdef01234567890",
    "arn": "arn:abcdef01234567890",
    "status": "DELETE_IN_PROGRESS"
  },
  "requestID": "abcdef01234567890",
  "eventID": "abcdef01234567890",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
```

```
  "recipientAccountId": "abcdef01234567890",
  "eventCategory": "Management"
}
```

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

Kuota untuk Amazon VPC Lattice

Anda Akun AWS memiliki kuota default, sebelumnya disebut sebagai batas, untuk masing-masing. Layanan AWS Kecuali dinyatakan lain, setiap kuota bersifat khusus per Wilayah. Anda dapat meminta kenaikan untuk beberapa kuota, dan kuota lainnya tidak dapat ditingkatkan.

Untuk melihat kuota untuk VPC Lattice, buka konsol Service Quotas. Di panel navigasi, pilih Layanan AWSdan pilih VPC Lattice.

Untuk meminta penambahan kuota, lihat Meminta penambahan kuota di Panduan Pengguna Service Quotas.

Anda Akun AWS memiliki kuota berikut yang terkait dengan VPC Lattice.

Nama	Default	Dapat disesuaikan	Deskripsi
Ukuran kebijakan autentikasi	Setiap Wilayah yang didukung: 10 Kilobyte	Tidak	Ukuran maksimum file JSON dalam kebijakan Auth.
Konfigurasi Sumber Daya Anak per Konfigurasi Sumber Daya Grup	Setiap Wilayah yang didukung: 60	Ya	Jumlah maksimum konfigurasi sumber daya anak dalam konfigurasi sumber daya grup. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Verifikasi Domain per Wilayah AWS	Setiap Wilayah yang didukung: 5	Ya	Jumlah maksimum verifikasi domain yang dapat dibuat per akun. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.

Nama	Default	Dapat disesuaikan	Deskripsi
Pendengar per layanan	Setiap Wilayah yang didukung: 2	Ya	Jumlah maksimum pendengar yang dapat Anda buat untuk suatu layanan. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Konfigurasi Sumber Daya per jaringan layanan	Setiap Wilayah yang didukung: 500	Ya	Jumlah maksimum konfigurasi sumber daya yang terkait dengan jaringan layanan. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Konfigurasi sumber daya per Wilayah AWS	Setiap Wilayah yang didukung: 2.000	Ya	Jumlah maksimum konfigurasi sumber daya yang dapat dimiliki AWS akun per AWS Wilayah. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Gateway sumber daya untuk VPC	Setiap Wilayah yang didukung: 500	Ya	Jumlah maksimum gateway sumber daya dalam VPC. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.

Nama	Default	Dapat disesuaikan	Deskripsi
Aturan per pendengar	il-central-1:5 Masing-masing Wilayah yang didukung lainnya: 10	Ya	Jumlah maksimum aturan yang dapat Anda tentukan untuk pendengar layanan Anda. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Kelompok keamanan per asosiasi	Setiap Wilayah yang didukung: 5	Tidak	Jumlah maksimum grup keamanan yang dapat Anda tambahkan ke asosiasi antara VPC dan jaringan layanan.
Asosiasi layanan per jaringan layanan	Setiap Wilayah yang didukung: 500	Ya	Jumlah maksimum layanan yang dapat Anda kaitkan dengan satu jaringan layanan. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Jaringan layanan per wilayah	il-central-1:10 Masing-masing Wilayah yang didukung lainnya: 50	Ya	Jumlah maksimum jaringan layanan per wilayah. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.

Nama	Default	Dapat disesuaikan	Deskripsi
Layanan per wilayah	il-central-1:500 Masing-masing Wilayah yang didukung lainnya: 2.000	Ya	Jumlah maksimum layanan per wilayah. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Kelompok sasaran per wilayah	Setiap Wilayah yang didukung: 500	Ya	Jumlah maksimum kelompok sasaran per wilayah. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Grup target per layanan	il-central-1:5 Masing-masing Wilayah yang didukung lainnya: 10	Ya	Jumlah maksimum kelompok sasaran yang dapat Anda kaitkan dengan layanan. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Target per kelompok sasaran	Setiap Wilayah yang didukung: 1.000	Ya	Jumlah maksimum target yang dapat Anda kaitkan dengan satu kelompok target. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.

Nama	Default	Dapat disesuaikan	Deskripsi
Asosiasi VPC per jaringan layanan	Setiap Wilayah yang didukung: 500	Ya	Jumlah maksimum VPCs yang dapat Anda kaitkan dengan satu jaringan layanan. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.
Titik akhir VPC dari jenis jaringan layanan per jaringan layanan	Setiap Wilayah yang didukung: 200	Ya	Jumlah maksimum titik akhir jaringan layanan yang terkait dengan jaringan layanan. Untuk peningkatan kapasitas dan batas tambahan, hubungi AWS Support.

Availability Zone berikut tidak didukung untuk VPC Lattice:`use1-az3`,,,`usw1-az2`,,`apne1-az3`, `apne2-az2euc1-az2`,`euw1-az4`. `cac1-az3` `ilc1-az2`

Batasan berikut juga berlaku.

Kuota	Nilai	Deskripsi
Bandwidth per layanan per Availability Zone	10 Gbps	Bandwidth default dialokasikan per layanan per Availability Zone. Ini dapat ditingkatkan, hubungi Solutions Architect (SA) atau Technical Account Manager (TAM) Anda untuk bantuan lebih lanjut.
Unit transmisi maksimum (MTU) per koneksi	8500 byte	Ukuran paket data terbesar yang dapat diterima layanan.

Kuota	Nilai	Deskripsi
Permintaan per detik per layanan per Availability Zone	10.000	Untuk layanan HTTP, ini adalah jumlah default permintaan per detik per layanan per Availability Zone. Ini dapat ditingkatkan, hubungi Solutions Architect (SA) atau Technical Account Manager (TAM) Anda untuk bantuan lebih lanjut.
Waktu idle koneksi per koneksi untuk layanan VPC Lattice	1 menit	Waktu default koneksi dapat diam tanpa permintaan aktif (untuk HTTP dan GRPC), atau tanpa transfer data aktif (untuk TLS-PASSTHOUGH) untuk layanan VPC Lattice. Anda dapat menggunakan HTTP dan keepalives tingkat aplikasi untuk memperpanjang batas waktu idle ini hingga durasi masa pakai koneksi maksimum. Ini dapat ditingkatkan, hubungi Solutions Architect (SA) atau Technical Account Manager (TAM) Anda untuk bantuan lebih lanjut.
Masa pakai koneksi maksimum per koneksi untuk layanan VPC Lattice	10 menit	Waktu maksimum koneksi dapat dibuka antara klien dan server untuk layanan VPC Lattice.
Masa pakai koneksi maksimum per koneksi untuk sumber daya VPC Lattice	TA	VPC Lattice tidak memaksakan batas koneksi seumur hidup untuk sumber daya. Klien dan server menentukan durasi koneksi seumur hidup sambil mengetahui batas waktu idle untuk sumber daya VPC Lattice, yaitu 350 detik.

Kuota	Nilai	Deskripsi
Waktu idle koneksi per koneksi untuk sumber daya VPC Lattice	350 detik	Anda dapat menggunakan keepalive s TCP untuk memperpanjang batas waktu idle ini.
Jaringan layanan untuk VPC	1 jaringan layanan	Anda dapat menghubungkan VPC ke hanya satu jaringan layanan melalui asosiasi. Untuk menghubungkan VPC ke beberapa jaringan layanan, Anda dapat menggunakan titik akhir VPC dari jenis jaringan layanan.

Riwayat dokumen untuk Panduan Pengguna Amazon VPC Lattice

Tabel berikut menjelaskan rilis dokumentasi untuk VPC Lattice.

Perubahan	Deskripsi	Tanggal
Menambahkan alamat IP yang dapat dikonfigurasi untuk gateway sumber daya	VPC Lattice sekarang mendukung alamat IP yang dapat dikonfigurasi untuk gateway sumber daya.	Oktober 7, 2025
Ditambahkan VPC Lattice untuk Oracle Database@AWS	VPC Lattice untuk dirilis. Oracle Database@AWS	Juni 26, 2025
Menambahkan dukungan dual-stack untuk endpoint manajemen	VPC Lattice sekarang mendukung IPv6 titik akhir dual-stack (IPv4 dan) untuk semua manajemen VPC Lattice. APIs	April 30, 2025
Bagikan dan akses sumber daya	VPC Lattice sekarang mendukung berbagi dan mengakses sumber daya di seluruh VPC dan batas akun. Ini termasuk pembaruan VPCLatticeReadOnly Accessdan VPCLattic eFullAccesskebijakan.	Desember 1, 2024
Passthrough TLS	VPC Lattice sekarang mendukung passthrough TLS, yang memungkinkan Anda melakukan penghenti an TLS di aplikasi Anda untuk otentikasi. end-to-end	14 Mei 2024

Versi struktur acara Lambda	VPC Lattice sekarang mendukung versi baru dari struktur acara Lambda.	7 September 2023
Support untuk berbagi VPCs	Peserta dapat membuat grup target VPC Lattice dalam VPC bersama.	5 Juli 2023
Rilis Ketersediaan Umum	Rilis Panduan Pengguna Kisi VPC untuk Ketersediaan Umum (GA)	31 Maret 2023
VPC Lattice sekarang melaporkan perubahan pada kebijakan yang dikelola AWS	Perubahan pada kebijakan terkelola dilaporkan dalam "kebijakan AWS terkelola untuk Kisi VPC" di bagian "Keamanan".	29 Maret 2023
Support untuk tipe target Application Load Balancer	VPC Lattice sekarang mendukung pembuatan grup target tipe Application Load Balancer.	29 Maret 2023
Support untuk semua jenis instans	VPC Lattice sekarang mendukung semua jenis instance.	Maret 27, 2023
IPv6 dukungan	VPC Lattice sekarang mendukung keduanya IPv4 dan kelompok target IPv6 IP.	Maret 27, 2023
HTTP2 versi protokol untuk pemeriksaan kesehatan	Pemeriksaan kesehatan sekarang didukung ketika versi protokol grup target HTTP2.	Maret 27, 2023

Tindakan respons tetap untuk aturan pendengar	Pendengar untuk layanan VPC Lattice sekarang mendukung tindakan respons tetap selain tindakan penerusan.	Maret 27, 2023
Support untuk nama domain kustom	Anda sekarang dapat mengonfigurasi nama domain khusus untuk layanan VPC Lattice Anda	14 Februari 2023
Support untuk BYOC (Bring Your Own Certificate)	VPC Lattice mendukung penggunaan SSL/TLS sertifikat Anda sendiri di ACM untuk nama domain kustom.	14 Februari 2023
VPC Lattice sekarang melaporkan daftar terbaru dari jenis instans yang tidak didukung	Tiga instance tambahan telah ditambahkan ke daftar instans yang tidak didukung.	26 Januari 2023
VPC Lattice sekarang melaporkan perubahan pada kebijakan yang dikelola AWS	Mulai 5 Desember 2022, perubahan kebijakan terkelola dilaporkan dalam topik "kebijakan AWS terkelola untuk Kisi VPC" di bagian "Keamanan". Perubahan pertama yang tercantum adalah penambahan izin yang diperlukan untuk CloudWatch pemantauan.	Desember 5, 2022
Rilis awal	Rilis awal Panduan Pengguna VPC Lattice	Desember 5, 2022

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