Penyeimbang Beban Klasik

Penyeimbang Beban Elastis

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

Penyeimbang Beban Elastis: Penyeimbang Beban Klasik

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

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

Table of Contents

Apa itu Classic Load Balancer? 1
Ikhtisar Classic Load Balancer 1
Manfaat 2
Cara memulai 3
Harga 3
Penyeimbang beban yang menghadap ke internet 4
Nama DNS publik untuk penyeimbang beban Anda 4
Buat penyeimbang beban yang menghadap ke internet 5
Sebelum Anda mulai 5
Buat Classic Load Balancer menggunakan AWS Management Console 6
Penyeimbang beban internal 9
Nama DNS publik untuk penyeimbang beban Anda 10
Buat penyeimbang beban internal 11
Prasyarat 11
Buat penyeimbang beban internal menggunakan konsol 11
Buat penyeimbang beban internal menggunakan AWS CLI 14
Konfigurasikan penyeimbang beban Anda 16
Batas waktu koneksi idle 17
Konfigurasikan batas waktu idle menggunakan konsol 18
Konfigurasikan batas waktu idle menggunakan AWS CLI 18
Penyeimbangan beban lintas zona 19
Aktifkan penyeimbangan beban lintas zona 19
Nonaktifkan penyeimbangan beban lintas zona 21
Pengurasan koneksi 23
Aktifkan pengeringan koneksi 24
Nonaktifkan pengeringan koneksi 25
Sesi lengket 26
Kelengketan sesi berbasis durasi 27
Kelengketan sesi yang dikontrol aplikasi 29
Mode mitigasi desync 32
Klasifikasi 33
Modus 34
Ubah mode mitigasi desync 35
Protokol proxy 36

Header protokol proxy 36
Prasyarat untuk mengaktifkan protokol proxy 37
Aktifkan protokol proxy menggunakan AWS CLI 37
Nonaktifkan protokol proxy menggunakan AWS CLI 39
Tanda 40
Batasan tag 41
Tambahkan tanda 41
Hapus tag 42
Subnet dan zona 42
Persyaratan 43
Konfigurasikan subnet menggunakan konsol 44
Konfigurasikan subnet menggunakan CLI 44
Grup keamanan 45
Aturan yang disarankan untuk grup keamanan penyeimbang beban 46
Tetapkan grup keamanan menggunakan konsol 48
Tetapkan grup keamanan menggunakan AWS CLI 48
Jaringan ACLs 49
Nama domain kustom 50
Mengaitkan nama domain kustom Anda dengan nama penyeimbang beban Anda 51
Menggunakan failover DNS Route 53 untuk penyeimbang beban Anda 52
Memutuskan nama domain kustom Anda dari penyeimbang beban Anda 53
Listener 54
Protokol 54
Protokol TCP/SSL 55
Protokol HTTP/HTTPS 55
HTTPS/SSL pendengar 56
Sertifikat server SSL 56
Negosiasi SSL 56
Autentikasi server back-end 57
Konfigurasi pendengar 57
Header X-diteruskan 59
X-Diteruskan-Untuk 60
X-Diteruskan-Proto 61
Port-X-Diteruskan 62
Pendengar HTTPS 63
Sertifikat SSL/TLS 64

Membuat atau mengimpor sertifikat SSL/TLS menggunakan AWS Certificate Manager 65
Impor sertifikat SSL/TLS menggunakan IAM 65
Konfigurasi negosiasi SSL 65
Kebijakan Keamanan 66
Protokol SSL 67
Preferensi Pesanan Server 67
Cipher SSL 68
Cipher suite untuk koneksi back-end 71
Kebijakan keamanan SSL yang telah ditentukan 72
Protokol berdasarkan kebijakan 73
Cipher berdasarkan kebijakan 74
Kebijakan oleh cipher 79
Buat penyeimbang beban HTTPS 86
Prasyarat 86
Buat penyeimbang beban HTTPS menggunakan konsol 87
Buat penyeimbang beban HTTPS menggunakan AWS CLI 91
Konfigurasikan pendengar HTTPS 103
Prasyarat 103
Tambahkan pendengar HTTPS menggunakan konsol 104
Tambahkan pendengar HTTPS menggunakan AWS CLI 105
Ganti sertifikat SSL 107
Ganti sertifikat SSL menggunakan konsol 108
Ganti sertifikat SSL menggunakan AWS CLI 109
Perbarui konfigurasi negosiasi SSL 110
Perbarui konfigurasi negosiasi SSL menggunakan konsol 110
Perbarui konfigurasi negosiasi SSL menggunakan AWS CLI 111
Contoh terdaftar 117
Praktik terbaik untuk instans Anda 117
Rekomendasi untuk VPC Anda 118
Daftarkan instans dengan penyeimbang beban Anda 119
Daftarkan sebuah instance 120
Melihat instans yang terdaftar dengan penyeimbang beban 121
Tentukan penyeimbang beban untuk instance terdaftar 121
Deregister sebuah instance 121
Pemeriksaan kondisi 122
Konfigurasi pemeriksaan kesehatan 123

Perbarui konfigurasi pemeriksaan kesehatan 126
Periksa kesehatan instans Anda 126
Memecahkan masalah pemeriksaan kesehatan 127
Grup keamanan 127
Jaringan ACLs 128
Pantau penyeimbang beban Anda 129
CloudWatch metrik 129
Metrik Classic Load Balancer 130
Dimensi metrik untuk Classic Load Balancer 140
Statistik untuk metrik Classic Load Balancer 140
Lihat CloudWatch metrik untuk penyeimbang beban Anda 141
Log akses 143
Berkas log akses 144
Entri log akses 145
Memproses log akses 150
Aktifkan log akses 150
Nonaktifkan log akses 159
Memecahkan masalah penyeimbang beban Anda 161
Kesalahan API 163
CertificateNotFound: Tidak terdefinisi 163
OutofService: Terjadi kesalahan sementara 163
Kesalahan HTTP 164
HTTP 400: PERMINTAAN BURUK 165
HTTP 405: METHOD_NOT_ALLOWED 165
HTTP 408: Waktu habis permintaan 165
HTTP 502: Gateway buruk 166
503 Layanan Tidak Tersedia 166
HTTP 504: Waktu habis gateway 167
Metrik kode respons 167
HTTPCode_ELB_4XX 168
HTTPCode_ELB_5XX 168
HTTPCode_Backend_2xx 168
HTTPCode_Backend_3xx 168
HTTPCode_Backend_4XX 169
HTTPCode_Backend_5XX 169
Pemeriksaan kondisi 169

Kesalahan halaman target pemeriksaan kesehatan 170
Koneksi ke instance telah habis 171
Otentikasi kunci publik gagal 172
Instance tidak menerima lalu lintas dari penyeimbang beban 172
Port pada instance tidak terbuka 173
Instance dalam grup Auto Scaling gagal dalam pemeriksaan kesehatan ELB 173
Konektivitas klien 174
Klien tidak dapat menyambung ke Load Balancer yang menghadap internet 174
Permintaan yang dikirim ke domain kustom tidak diterima oleh penyeimbang beban 174
Permintaan HTTPS yang dikirim ke penyeimbang beban mengembalikan “NET: :ERR_CERT_COMMON_NAME_INVALID” 175
Pendaftaran instans 175
Terlalu lama untuk mendaftarkan sebuah EC2 instans 175
Tidak dapat mendaftarkan instance yang diluncurkan dari AMI berbayar 176
Kuota 177
Riwayat dokumen 178
..... clxxxvi

Apa itu Classic Load Balancer?

> **Note**
>
> Classic Load Balancer adalah load balancer generasi sebelumnya dari Elastic Load Balancing. Kami menyarankan Anda untuk bermigrasi ke penyeimbang beban generasi saat ini. Untuk informasi selengkapnya, lihat Memigrasi Classic Load Balancer Anda.

Elastic Load Balancing secara otomatis mendistribusikan lalu lintas masuk Anda ke beberapa target, seperti EC2 instans, kontainer, dan alamat IP, dalam satu atau beberapa Availability Zone. Ini memantau kesehatan target terdaftarnya, dan mengarahkan lalu lintas hanya ke target yang sehat. Elastic Load Balancing menskalakan load balancer Anda saat lalu lintas masuk Anda berubah seiring waktu. Hal ini dapat secara otomatis menskalakan sebagian besar beban kerja.

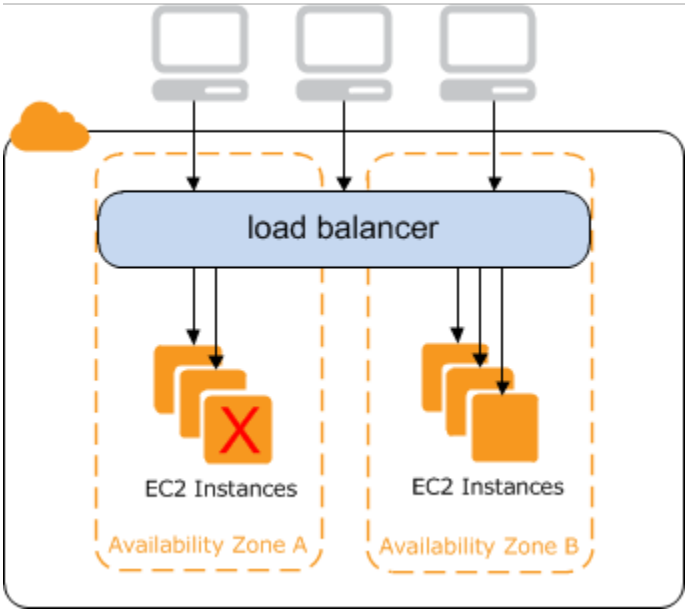
Ikhtisar Classic Load Balancer

Load balancer mendistribusikan lalu lintas aplikasi yang masuk di beberapa EC2 instance di beberapa Availability Zone. Ini meningkatkan toleransi kesalahan aplikasi Anda. Elastic Load Balancing mendeteksi kejadian yang tidak sehat dan mengarahkan lalu lintas hanya ke instans yang sehat.

Penyeimbang beban Anda berfungsi sebagai titik kontak tunggal untuk klien. Hal ini akan meningkatkan ketersediaan aplikasi Anda. Anda dapat menambahkan dan menghapus instans dari penyeimbang beban Anda saat kebutuhan Anda berubah, tanpa mengganggu keseluruhan aliran permintaan ke aplikasi Anda. Elastic Load Balancing menskalakan penyeimbang beban Anda saat lalu lintas ke aplikasi Anda berubah seiring waktu. Elastic Load Balancing dapat menskalakan sebagian besar beban kerja secara otomatis.

Pendengar memeriksa permintaan koneksi dari klien, menggunakan protokol dan port yang Anda konfigurasikan, dan meneruskan permintaan ke satu atau beberapa instance terdaftar menggunakan protokol dan nomor port yang Anda konfigurasikan. Anda menambahkan satu listener atau lebih ke penyeimbang beban Anda.

Anda dapat mengonfigurasi pemeriksaan kesehatan, yang digunakan untuk memantau kesehatan instans terdaftar sehingga penyeimbang beban hanya mengirimkan permintaan ke instans sehat.



Untuk memastikan bahwa instans terdaftar Anda dapat menangani beban permintaan di setiap Availability Zone, penting untuk menyimpan kira-kira jumlah instans yang sama di setiap Availability Zone yang terdaftar dengan penyeimbang beban. Misalnya, jika Anda memiliki sepuluh instance di Availability Zone us-west-2a dan dua instance di us-west-2b, permintaan didistribusikan secara merata di antara dua Availability Zone. Akibatnya, dua contoh di us-barat-2b melayani jumlah lalu lintas yang sama dengan sepuluh contoh di us-barat-2a. Sebagai gantinya, Anda harus memiliki enam instance di setiap Availability Zone.

Secara default, penyeimbang beban mendistribusikan lalu lintas secara merata di seluruh Availability Zone yang Anda aktifkan untuk penyeimbang beban Anda. Untuk mendistribusikan lalu lintas secara merata di semua instans terdaftar di semua Availability Zone yang diaktifkan, aktifkan penyeimbangan beban lintas zona pada penyeimbang beban Anda. Namun, kami tetap menyarankan agar Anda mempertahankan jumlah instance yang kira-kira setara di setiap Availability Zone untuk toleransi kesalahan yang lebih baik.

Untuk informasi selengkapnya, lihat Cara kerja Elastic Load Balancing di Panduan Pengguna Elastic Load Balancing.

Manfaat

Menggunakan Classic Load Balancer sebagai pengganti Application Load Balancer memiliki manfaat sebagai berikut:

- Support untuk pendengar TCP dan SSL

- Support untuk sesi lengket menggunakan cookie yang dihasilkan aplikasi

Untuk informasi selengkapnya tentang fitur yang didukung oleh setiap jenis penyeimbang beban, lihat Perbandingan produk untuk Elastic Load Balancing.

Cara memulai

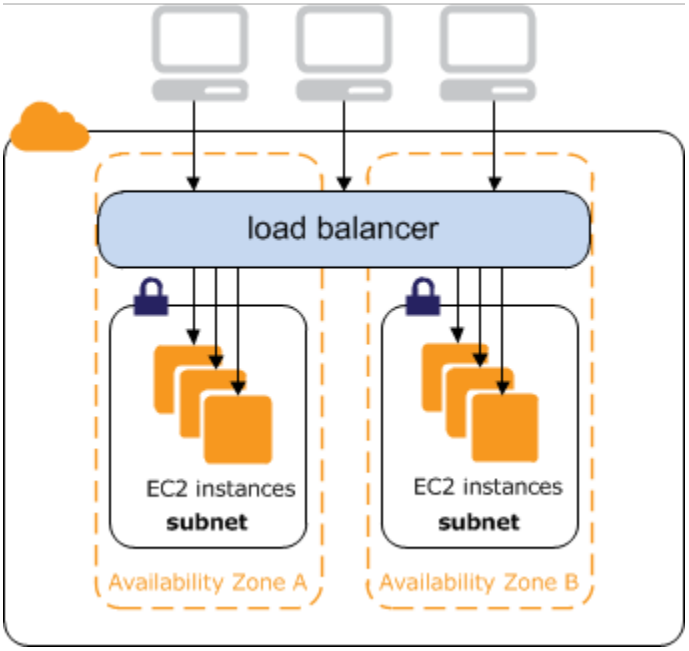
- Untuk mempelajari cara membuat Classic Load Balancer dan mendaftarkan EC2 instance dengannya, lihat. Buat Classic Load Balancer yang menghadap ke internet
- Untuk mempelajari cara membuat penyeimbang beban HTTPS dan mendaftarkan EC2 instance dengannya, lihat. Membuat Classic Load Balancer dengan pendengar HTTPS
- Untuk mempelajari cara menggunakan berbagai fitur yang didukung oleh Classic Load Balancers, lihat. Konfigurasikan Classic Load Balancer Anda

Harga

Dengan penyeimbang beban, Anda hanya membayar apa yang Anda gunakan. Untuk informasi lebih lanjut, lihat Harga Elastic Load Balancing.

Penyeimbang Beban Klasik yang Menghadap Internet

Saat Anda membuat Classic Load Balancer, Anda dapat menjadikannya penyeimbang beban internal atau penyeimbang beban yang menghadap ke internet. Penyeimbang beban yang menghadap ke internet memiliki nama DNS yang dapat diselesaikan secara publik, sehingga dapat merutekan permintaan dari klien melalui internet ke EC2 instance yang terdaftar dengan penyeimbang beban.



Nama DNS Load Balancer internal dapat dibuka secara publik ke alamat IP pribadi dari simpul. Oleh karena itu, Load Balancer internal hanya dapat merutekan permintaan dari klien dengan akses ke VPC untuk Load Balancer. Untuk informasi selengkapnya, lihat Penyeimbang beban internal.

Daftar Isi

- Nama DNS publik untuk penyeimbang beban Anda
- Buat Classic Load Balancer yang menghadap ke internet

Nama DNS publik untuk penyeimbang beban Anda

Ketika penyeimbang beban Anda dibuat, ia menerima nama DNS publik yang dapat digunakan klien untuk mengirim permintaan. Server DNS menyelesaikan nama DNS penyeimbang beban Anda ke alamat IP publik dari node penyeimbang beban untuk penyeimbang beban Anda. Setiap node load balancer terhubung ke instance back-end menggunakan alamat IP pribadi.

Konsol menampilkan nama DNS publik dengan formulir berikut:

```
name-1234567890.region.elb.amazonaws.com
```

Buat Classic Load Balancer yang menghadap ke internet

Saat membuat penyeimbang beban, Anda mengonfigurasi listener, mengonfigurasi pemeriksaan kesehatan, dan mendaftarkan instance back-end. Anda mengonfigurasi listener dengan menentukan protokol dan port untuk koneksi front-end (client to load balancer), serta protokol dan port untuk koneksi back-end (penyeimbang beban ke instance back-end). Anda dapat mengonfigurasi beberapa pendengar untuk penyeimbang beban Anda.

Tutorial ini memberikan pengenalan langsung ke Classic Load Balancers melalui antarmuka berbasis web AWS Management Console. Anda akan membuat penyeimbang beban yang menerima lalu lintas HTTP publik dan mengirimkannya ke EC2 instance Anda.

Untuk membuat penyeimbang beban dengan pendengar HTTPS, lihat. Membuat Classic Load Balancer dengan pendengar HTTPS

Tugas

- Sebelum Anda mulai
- Buat Classic Load Balancer menggunakan AWS Management Console

Sebelum Anda mulai

- Buat cloud pribadi virtual (VPC). Untuk informasi selengkapnya, lihat Rekomendasi untuk VPC Anda.
- Luncurkan EC2 contoh yang Anda rencanakan untuk mendaftar dengan penyeimbang beban Anda. Pastikan bahwa grup keamanan untuk instans ini memungkinkan akses HTTP pada port 80.
- Instal server web, seperti Apache atau Internet Information Services (IIS), pada setiap contoh, masukkan nama DNS-nya ke bidang alamat browser web yang terhubung ke internet, dan verifikasi bahwa browser menampilkan halaman default server.

Buat Classic Load Balancer menggunakan AWS Management Console

Gunakan prosedur berikut untuk membuat Classic Load Balancer Anda. Berikan informasi konfigurasi dasar untuk penyeimbang beban Anda, seperti nama dan skema. Kemudian berikan informasi tentang jaringan Anda, dan pendengar yang mengarahkan lalu lintas ke instans Anda.

Untuk membuat Classic Load Balancer menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi, pilih Wilayah untuk penyeimbang beban Anda. Pastikan untuk memilih Wilayah yang sama yang Anda pilih untuk EC2 instans Anda.
3. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
4. Pilih Create Load Balancer.
5. Perluas bagian Classic Load Balancer, lalu pilih Buat.
6. Konfigurasi dasar
   a. Untuk nama Load balancer, ketikkan nama untuk penyeimbang beban Anda.

      Nama Classic Load Balancer Anda harus unik dalam set Classic Load Balancer untuk Wilayah, dapat memiliki maksimal 32 karakter, hanya dapat berisi karakter alfanumerik dan tanda hubung, dan tidak boleh dimulai atau diakhiri dengan tanda hubung.
   b. Untuk Skema, pilih Menghadap Internet.
7. Pemetaan jaringan
   a. Untuk VPC, pilih VPC yang sama yang Anda pilih untuk instans Anda.
   b. Untuk Pemetaan, pertama-tama pilih Availability Zone, lalu pilih subnet publik dari subnet yang tersedia. Anda hanya dapat memilih satu subnet per Availability Zone. Untuk meningkatkan ketersediaan penyeimbang beban Anda, pilih lebih dari satu Availability Zone dan subnet.
8. Grup keamanan
   - Untuk grup Keamanan, pilih grup keamanan yang ada yang dikonfigurasi untuk mengizinkan lalu lintas HTTP yang diperlukan pada port 80.
9. Pendengar dan perutean
   a. Untuk Listener, pastikan protokolnya `HTTP` dan portnya`80`.
   b. Misalnya, pastikan protokolnya `HTTP` dan portnya`80`.

10. Pemeriksaan Kesehatan

    a. Untuk Protokol Ping, pastikan protokolnya`HTTP`.

    b. Untuk Port Ping, pastikan portnya`80`.

    c. Untuk Ping Path, pastikan jalurnya`/`.

    d. Untuk pengaturan pemeriksaan kesehatan lanjutan, gunakan nilai default.

11. Contoh

    a. Pilih Tambahkan instance, untuk memunculkan layar pemilihan instance.

    b. Di bawah Instans yang tersedia, Anda dapat memilih dari instans saat ini yang tersedia untuk penyeimbang beban, berdasarkan pengaturan jaringan saat ini.

    c. Setelah puas dengan pilihan Anda, pilih Konfirmasi untuk menambahkan instance yang akan didaftarkan ke penyeimbang beban.

12. Atribut

    • Untuk Aktifkan penyeimbangan beban lintas zona, Aktifkan pengurasan koneksi, dan Timeout (interval pengeringan) pertahankan nilai default.

13. Tag penyeimbang beban (opsional)

    a. Bidang kunci diperlukan.

    b. Bidang Nilai adalah opsional.

    c. Untuk menambahkan tag lain, pilih Tambahkan tag baru lalu masukkan nilai Anda ke bidang Kunci, dan opsional bidang Nilai.

    d. Untuk menghapus tag yang ada, pilih Hapus di samping tag yang ingin Anda hapus.

14. Ringkasan dan penciptaan

    a. Jika Anda perlu mengubah pengaturan apa pun, pilih Edit di samping pengaturan yang perlu diubah.

    b. Setelah Anda puas dengan semua pengaturan yang ditampilkan dalam ringkasan, pilih Buat penyeimbang beban untuk memulai pembuatan penyeimbang beban Anda.

    c. Pada halaman pembuatan akhir, pilih Lihat penyeimbang beban untuk melihat penyeimbang beban Anda di konsol Amazon EC2 .

15. Verifikasi

    a. Pilih penyeimbang beban baru Anda.

b. Pada tab Instance target, periksa kolom Status Kesehatan. Setelah setidaknya satu dari EC2 instans Anda dalam layanan, Anda dapat menguji penyeimbang beban Anda.

c. Di bagian Detail, salin nama DNS penyeimbang beban, yang akan terlihat mirip dengan. `my-load-balancer-1234567890.us-east-1.elb.amazonaws.com`

d. Rekatkan nama DNS penyeimbang beban Anda ke bidang alamat browser web yang terhubung internet publik. Jika load balancer Anda berfungsi dengan benar, Anda akan melihat halaman default server Anda.

16. Hapus (opsional)

a. Jika Anda memiliki catatan CNAME untuk domain yang mengarah ke penyeimbang beban, arahkan catatan ke lokasi baru dan tunggu hingga perubahan DNS diterapkan sebelum menghapus penyeimbang beban.

b. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.

c. Pilih penyeimbang beban.

d. Pilih Tindakan, Hapus penyeimbang beban.

e. Saat diminta konfirmasi, ketik `confirm` lalu pilih Hapus.

f. Setelah Anda menghapus penyeimbang beban, EC2 instance yang terdaftar dengan penyeimbang beban terus berjalan. Anda akan ditagih untuk setiap jam sebagian atau penuh yang terus mereka jalankan. Ketika Anda tidak lagi membutuhkan EC2 instance, Anda dapat menghentikan atau menghentikannya untuk mencegah timbulnya biaya tambahan.

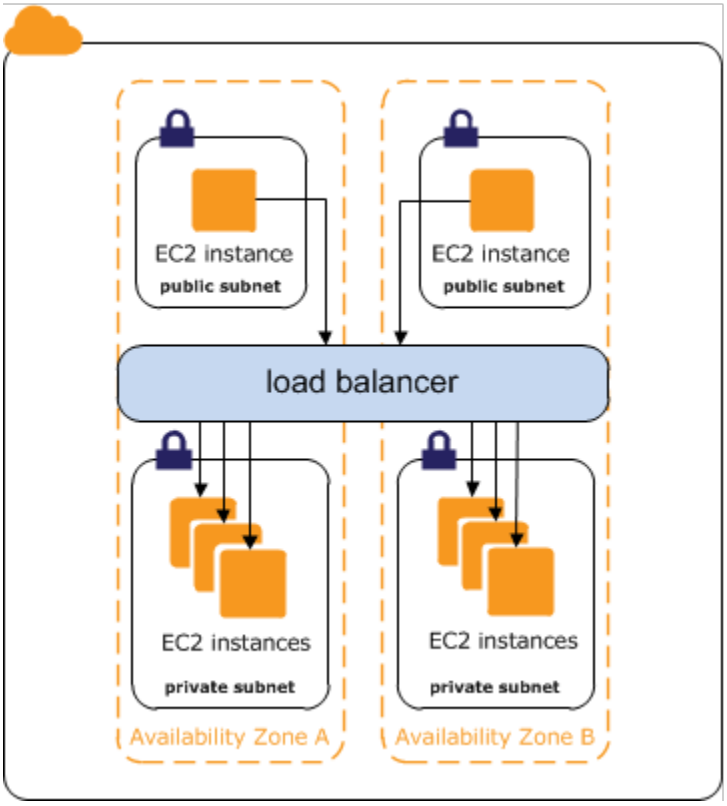
Penyeimbang Beban Klasik Internal

Bila Anda membuat Load Balancer, Anda harus memilih apakah akan menjadikannya internal atau menghadap-internet.

Simpul Load Balancer menghadap-internet memiliki alamat IP publik. Nama DNS dari Load Balancer yang menghadap internet dapat dipecahkan secara publik ke alamat IP publik simpul tersebut. Oleh karena itu, Load Balancer yang menghadap internet dapat merutekan permintaan dari klien melalui internet. Untuk informasi selengkapnya, lihat Penyeimbang Beban Klasik yang Menghadap Internet.

Simpul penyeimbang beban internal hanya memiliki alamat IP privat. Nama DNS Load Balancer internal dapat dibuka secara publik ke alamat IP pribadi dari simpul. Oleh karena itu, Load Balancer internal hanya dapat merutekan permintaan dari klien dengan akses ke VPC untuk Load Balancer.

Jika aplikasi Anda memiliki beberapa tingkatan, misalnya server web yang harus terhubung ke internet dan server database yang hanya terhubung ke server web, Anda dapat merancang arsitektur yang menggunakan penyeimbang beban internal dan internet. Buat Load Balancer yang menghadap internet dan daftarkan server web dengannya. Buat penyeimbang beban internal dan daftarkan server database dengannya. Server web menerima permintaan dari penyeimbang beban yang menghadap ke internet dan mengirim permintaan untuk server database ke penyeimbang beban internal. Server database menerima permintaan dari penyeimbang beban internal.



Daftar Isi

- Nama DNS publik untuk penyeimbang beban Anda
- Buat Classic Load Balancer internal

Nama DNS publik untuk penyeimbang beban Anda

Ketika penyeimbang beban internal dibuat, ia menerima nama DNS publik dengan formulir berikut:

```
internal-name-123456789.region.elb.amazonaws.com
```

Server DNS menyelesaikan nama DNS penyeimbang beban Anda ke alamat IP pribadi node penyeimbang beban untuk penyeimbang beban internal Anda. Setiap node penyeimbang beban terhubung ke alamat IP pribadi dari instance back-end menggunakan antarmuka jaringan elastis. Jika penyeimbangan beban lintas zona diaktifkan, setiap node terhubung ke setiap instance back-end, terlepas dari Availability Zone. Jika tidak, setiap node terhubung hanya ke instance yang ada di Availability Zone-nya.

Buat Classic Load Balancer internal

Anda dapat membuat penyeimbang beban internal untuk mendistribusikan lalu lintas ke EC2 instans Anda dari klien dengan akses ke VPC untuk penyeimbang beban.

Daftar Isi

- Prasyarat
- Buat penyeimbang beban internal menggunakan konsol
- Buat penyeimbang beban internal menggunakan AWS CLI

Prasyarat

- Jika Anda belum membuat VPC untuk penyeimbang beban Anda, Anda harus membuatnya sebelum memulai. Untuk informasi selengkapnya, lihat Rekomendasi untuk VPC Anda.
- Luncurkan EC2 instance yang Anda rencanakan untuk mendaftar dengan penyeimbang beban internal Anda. Pastikan Anda meluncurkannya di subnet pribadi di VPC yang ditujukan untuk penyeimbang beban.

Buat penyeimbang beban internal menggunakan konsol

Gunakan prosedur berikut untuk membuat Classic Load Balancer internal Anda. Berikan informasi konfigurasi dasar untuk penyeimbang beban Anda, seperti nama dan skema. Kemudian berikan informasi tentang jaringan Anda, dan pendengar yang mengarahkan lalu lintas ke instans Anda..

Untuk membuat Classic Load Balancer internal menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi, pilih Wilayah untuk penyeimbang beban Anda. Pastikan untuk memilih Wilayah yang sama yang Anda pilih untuk EC2 instans Anda.
3. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
4. Pilih Create Load Balancer.
5. Perluas bagian Classic Load Balancer, lalu pilih Buat.
6. Konfigurasi dasar

    a. Untuk nama Load balancer, ketikkan nama untuk penyeimbang beban Anda.

Nama Classic Load Balancer Anda harus unik dalam set Classic Load Balancer untuk Wilayah, dapat memiliki maksimal 32 karakter, hanya dapat berisi karakter alfanumerik dan tanda hubung, dan tidak boleh dimulai atau diakhiri dengan tanda hubung.

b. Untuk Skema, pilih Internal.

7. Pemetaan jaringan

   a. Untuk VPC, pilih VPC yang sama yang Anda pilih untuk instans Anda.

   b. Untuk Pemetaan, pertama-tama pilih Availability Zone, lalu pilih subnet dari subnet yang tersedia. Anda hanya dapat memilih satu subnet per Availability Zone. Untuk meningkatkan ketersediaan penyeimbang beban Anda, pilih lebih dari satu Availability Zone dan subnet.

8. Untuk grup Keamanan, pilih grup keamanan yang ada yang dikonfigurasi untuk mengizinkan lalu lintas HTTP yang diperlukan pada port 80. Atau Anda dapat membuat grup keamanan baru jika aplikasi Anda menggunakan protokol dan port yang berbeda.

9. Pendengar dan perutean

   a. Untuk Listener, pastikan protokolnya `HTTP` dan portnya`80`.

   b. Misalnya, pastikan protokolnya `HTTP` dan portnya`80`.

10. Pemeriksaan Kesehatan

    a. Untuk Protokol Ping, defaultnya adalah`HTTP`.

    b. Untuk Port Ping, defaultnya adalah`80`.

    c. Untuk Ping Path, defaultnya adalah`/`.

    d. Untuk pengaturan pemeriksaan kesehatan lanjutan, gunakan nilai default atau masukkan nilai khusus untuk aplikasi Anda.

11. Contoh

    a. Pilih Tambahkan instance, untuk memunculkan layar pemilihan instance.

    b. Di bawah Instans yang tersedia, Anda dapat memilih dari instans saat ini yang tersedia untuk penyeimbang beban, berdasarkan pengaturan jaringan yang dipilih sebelumnya.

    c. Setelah puas dengan pilihan Anda, pilih Konfirmasi untuk menambahkan instance yang akan didaftarkan ke penyeimbang beban.

12. Atribut

    • Untuk Aktifkan penyeimbangan beban lintas zona, Aktifkan pengurasan koneksi, dan Timeout (interval pengeringan) pertahankan nilai default.

13. Tag penyeimbang beban (opsional)

    a. Bidang kunci diperlukan.

    b. Bidang Nilai adalah opsional.

    c. Untuk menambahkan tag lain, pilih Tambahkan tag baru lalu masukkan nilai Anda ke bidang Kunci, dan opsional bidang Nilai.

    d. Untuk menghapus tag yang ada, pilih Hapus di samping tag yang ingin Anda hapus.

14. Ringkasan dan penciptaan

    a. Jika Anda perlu mengubah pengaturan apa pun, pilih Edit di samping pengaturan yang perlu diubah.

    b. Setelah Anda puas dengan semua pengaturan yang ditampilkan dalam ringkasan, pilih Buat penyeimbang beban untuk memulai pembuatan penyeimbang beban Anda.

    c. Pada halaman pembuatan akhir, pilih Lihat penyeimbang beban untuk melihat penyeimbang beban Anda di konsol Amazon EC2 .

15. Verifikasi

    a. Pilih penyeimbang beban baru Anda.

    b. Pada tab Instance Target, periksa kolom Status Kesehatan. Setelah setidaknya satu dari EC2 instans Anda dalam layanan, Anda dapat menguji penyeimbang beban Anda.

    c. Di bagian Detail, salin nama DNS penyeimbang beban, yang akan terlihat mirip dengan. `my-load-balancer-1234567890.us-east-1.elb.amazonaws.com`

    d. Rekatkan nama DNS penyeimbang beban Anda ke bidang alamat browser web yang terhubung internet publik. Jika load balancer Anda berfungsi dengan benar, Anda akan melihat halaman default server Anda.

16. Hapus (opsional)

    a. Jika Anda memiliki catatan CNAME untuk domain yang mengarah ke penyeimbang beban, arahkan catatan ke lokasi baru dan tunggu hingga perubahan DNS diterapkan sebelum menghapus penyeimbang beban.

    b. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.

    c. Pilih penyeimbang beban.

    d. Pilih Tindakan, Hapus penyeimbang beban.

    e. Saat diminta konfirmasi, ketik `confirm` lalu pilih Hapus.

f. Setelah Anda menghapus penyeimbang beban, EC2 instance yang terdaftar dengan penyeimbang beban terus berjalan. Anda akan ditagih untuk setiap jam sebagian atau penuh yang terus mereka jalankan. Ketika Anda tidak lagi membutuhkan EC2 instance, Anda dapat menghentikan atau menghentikannya untuk mencegah timbulnya biaya tambahan.

Buat penyeimbang beban internal menggunakan AWS CLI

Secara default, Elastic Load Balancing menciptakan penyeimbang beban yang menghadap ke internet. Gunakan prosedur berikut untuk membuat penyeimbang beban internal dan daftarkan EC2 instans Anda dengan penyeimbang beban internal yang baru dibuat.

Untuk membuat penyeimbang beban internal

1. Gunakan create-load-balancerperintah dengan `--scheme` opsi diatur `keinternal`, sebagai berikut:

```
aws elb create-load-balancer --load-balancer-name my-internal-loadbalancer --
listeners Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=80
 --subnets subnet-4e05f721 --scheme internal --security-groups sg-b9ffedd5
```

   Berikut ini adalah contoh respons. Perhatikan bahwa nama menunjukkan bahwa ini adalah penyeimbang beban internal.

```
{
    "DNSName": "internal-my-internal-loadbalancer-786501203.us-
west-2.elb.amazonaws.com"
}
```

2. Gunakan perintah register-instances-with-load-balancer berikut untuk menambahkan instance:

```
aws elb register-instances-with-load-balancer --load-balancer-name my-internal-
loadbalancer --instances i-4f8cf126 i-0bb7ca62
```

   Berikut adalah respons contohnya:

```
{
    "Instances": [
        {
```

```
            "InstanceId": "i-4f8cf126"
        },
        {
            "InstanceId": "i-0bb7ca62"
        }
    ]
}
```

3. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi penyeimbang beban internal:

```
aws elb describe-load-balancers --load-balancer-name my-internal-loadbalancer
```

Respons mencakup `Scheme` bidang `DNSName` dan, yang menunjukkan bahwa ini adalah penyeimbang beban internal.

```
{
    "LoadBalancerDescriptions": [
        {
            ...
            "DNSName": "internal-my-internal-loadbalancer-1234567890.us-
west-2.elb.amazonaws.com",
            "SecurityGroups": [
                "sg-b9ffedd5"
            ],
            "Policies": {
                "LBCookieStickinessPolicies": [],
                "AppCookieStickinessPolicies": [],
                "OtherPolicies": []
            },
            "LoadBalancerName": "my-internal-loadbalancer",
            "CreatedTime": "2014-05-22T20:32:19.920Z",
            "AvailabilityZones": [
                "us-west-2a"
            ],
            "Scheme": "internal",
            ...
        }
    ]
}
```

Konfigurasikan Classic Load Balancer Anda

Setelah Anda membuat Classic Load Balancer, Anda dapat mengubah konfigurasinya. Misalnya, Anda dapat memperbarui atribut penyeimbang beban, subnet, dan grup keamanan.

Atribut penyeimbang beban

Pengeringan koneksi

Jika diaktifkan, penyeimbang beban memungkinkan permintaan yang ada selesai sebelum penyeimbang beban mengalihkan lalu lintas dari instans yang tidak terdaftar atau tidak sehat.

Penyeimbangan beban lintas zona

Jika diaktifkan, penyeimbang beban merutekan lalu lintas permintaan secara merata di semua instans terlepas dari Availability Zone.

Mode migitasi desync

Menentukan bagaimana penyeimbang beban menangani permintaan yang mungkin menimbulkan risiko keamanan pada aplikasi Anda. Nilai yang mungkin adalah `monitor`, `defensive`, dan `strictest`. Nilai default-nya `defensive`.

Batas waktu idle

Jika diaktifkan, penyeimbang beban memungkinkan koneksi untuk tetap idle (tidak ada data yang dikirim melalui koneksi) selama durasi yang ditentukan. Bawaannya adalah 60 detik.

Sesi lengket

Classic Load Balancers mendukung kekakuan sesi berbasis durasi dan berbasis aplikasi.

Detail penyeimbang beban

Grup keamanan

Grup keamanan untuk penyeimbang beban Anda harus mengizinkan lalu lintas pada pendengar dan port pemeriksaan kesehatan.

Subnet

Anda dapat memperluas kemampuan penyeimbang beban Anda ke subnet tambahan.

Protokol proxy

Jika diaktifkan, kami menambahkan header dengan informasi koneksi yang dikirim ke instance.

Tanda

Anda dapat menambahkan tag untuk mengkategorikan load balancres Anda.

Konfigurasikan batas waktu koneksi idle untuk Classic Load Balancer

Untuk setiap permintaan yang dibuat klien melalui Classic Load Balancer, penyeimbang beban mempertahankan dua koneksi. Koneksi front-end adalah antara klien dan penyeimbang beban. Koneksi back-end adalah antara penyeimbang beban dan instance terdaftar. EC2 Penyeimbang beban memiliki periode batas waktu idle yang dikonfigurasi yang berlaku untuk koneksinya. Jika tidak ada data yang dikirim atau diterima pada saat periode batas waktu idle berlalu, penyeimbang beban menutup koneksi. Untuk memastikan bahwa operasi yang panjang seperti unggahan file memiliki waktu untuk diselesaikan, kirim setidaknya 1 byte data sebelum setiap periode batas waktu idle berlalu dan tingkatkan panjang periode batas waktu idle sesuai kebutuhan.

Jika Anda menggunakan pendengar HTTP dan HTTPS, sebaiknya aktifkan opsi HTTP keep-alive untuk instance Anda. Anda dapat mengaktifkan keep-alive di pengaturan server web untuk instance Anda. Keep-alive, saat diaktifkan, memungkinkan penyeimbang beban untuk menggunakan kembali koneksi back-end hingga batas waktu keep-alive berakhir. Untuk memastikan bahwa penyeimbang beban bertanggung jawab untuk menutup koneksi ke instans Anda, pastikan bahwa nilai yang Anda tetapkan untuk waktu keep-alive HTTP lebih besar daripada pengaturan batas waktu idle yang dikonfigurasi untuk penyeimbang beban Anda.

Perhatikan bahwa probe keep-alive TCP tidak mencegah penyeimbang beban menghentikan koneksi karena mereka tidak mengirim data dalam muatan.

Daftar Isi

- Konfigurasikan batas waktu idle menggunakan konsol
- Konfigurasikan batas waktu idle menggunakan AWS CLI

Konfigurasikan batas waktu idle menggunakan konsol

Secara default, Elastic Load Balancing menetapkan batas waktu idle untuk penyeimbang beban Anda menjadi 60 detik. Gunakan prosedur berikut untuk menetapkan nilai yang berbeda untuk batas waktu idle.

Untuk mengonfigurasi pengaturan batas waktu idle untuk penyeimbang beban Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian konfigurasi Lalu lintas, ketikkan nilai untuk batas waktu Idle. Rentang untuk batas waktu idle adalah dari 1 hingga 4.000 detik.
6. Pilih Simpan perubahan.

Konfigurasikan batas waktu idle menggunakan AWS CLI

Gunakan modify-load-balancer-attributesperintah berikut untuk mengatur batas waktu idle untuk penyeimbang beban Anda:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --load-balancer-attributes "{\"ConnectionSettings\":{\"IdleTimeout\":30}}"
```

Berikut adalah respons contohnya:

```
{
    "LoadBalancerAttributes": {
        "ConnectionSettings": {
            "IdleTimeout": 30
        }
    },
    "LoadBalancerName": "my-loadbalancer"
}
```

Konfigurasikan load balancing lintas zona untuk Classic Load Balancer

Dengan penyeimbangan beban lintas zona, setiap simpul penyeimbang beban untuk Classic Load Balancer Anda mendistribusikan permintaan secara merata ke seluruh instans terdaftar di semua Availability Zone yang diaktifkan. Jika penyeimbangan beban lintas zona dinonaktifkan, setiap simpul penyeimbang beban mendistribusikan permintaan secara merata di seluruh instans terdaftar di Availability Zone saja. Untuk informasi lebih lanjut, lihat Penyeimbang beban lintas zona di Panduan Pengguna Elastic Load Balancing.

Penyeimbangan beban lintas zona mengurangi kebutuhan untuk mempertahankan jumlah instans yang setara di setiap Availability Zone yang diaktifkan, dan meningkatkan kemampuan aplikasi Anda untuk menangani hilangnya satu atau beberapa instance. Namun, kami tetap menyarankan agar Anda mempertahankan jumlah instance yang kira-kira setara di setiap Availability Zone yang diaktifkan untuk toleransi kesalahan yang lebih tinggi.

Untuk lingkungan di mana klien menyimpan cache pencarian DNS, permintaan masuk mungkin mendukung salah satu Availability Zone. Menggunakan penyeimbangan beban lintas zona, ketidakseimbangan dalam beban permintaan ini tersebar di semua instans yang tersedia di Wilayah, mengurangi dampak klien yang berperilaku buruk.

Saat Anda membuat Classic Load Balancer, default untuk load balancing lintas zona tergantung pada cara Anda membuat Load Balancer. Dengan API atau CLI, load balancing lintas zona dinonaktifkan secara default. Dengan AWS Management Console, opsi untuk mengaktifkan penyeimbangan beban lintas zona dipilih secara default. Setelah membuat Classic Load Balancer, Anda dapat mengaktifkan atau menonaktifkan penyeimbangan beban lintas zona kapan saja.

Daftar Isi

- Aktifkan penyeimbangan beban lintas zona
- Nonaktifkan penyeimbangan beban lintas zona

Aktifkan penyeimbangan beban lintas zona

Anda dapat mengaktifkan penyeimbangan beban lintas zona untuk Classic Load Balancer Anda kapan saja.

Untuk mengaktifkan penyeimbangan beban lintas zona menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian konfigurasi perutean Availability Zone, aktifkan penyeimbangan beban lintas zona.
6. Pilih Simpan perubahan.

Untuk mengaktifkan penyeimbangan beban lintas zona menggunakan AWS CLI

1. Gunakan modify-load-balancer-attributesperintah berikut untuk mengatur `CrossZoneLoadBalancing` atribut penyeimbang beban Anda ke`true`:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --
load-balancer-attributes "{\"CrossZoneLoadBalancing\":{\"Enabled\":true}}"
```

   Berikut adalah respons contohnya:

```
{
   "LoadBalancerAttributes": {
     "CrossZoneLoadBalancing": {
         "Enabled": true
       }
   },
   "LoadBalancerName": "my-loadbalancer"
 }
```

2. (Opsional) Gunakan describe-load-balancer-attributesperintah berikut untuk memverifikasi bahwa penyeimbangan beban lintas zona diaktifkan untuk penyeimbang beban Anda:

```
aws elb describe-load-balancer-attributes --load-balancer-name my-loadbalancer
```

   Berikut adalah respons contohnya:

```
{
    "LoadBalancerAttributes": {
```

```
        "ConnectionDraining": {
            "Enabled": false,
            "Timeout": 300
        },
        "CrossZoneLoadBalancing": {
            "Enabled": true
        },
        "ConnectionSettings": {
            "IdleTimeout": 60
        },
        "AccessLog": {
            "Enabled": false
        }
    }
}
```

Nonaktifkan penyeimbangan beban lintas zona

Anda dapat menonaktifkan opsi penyeimbangan beban lintas zona untuk penyeimbang beban Anda kapan saja.

Untuk menonaktifkan penyeimbangan beban lintas zona menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian konfigurasi perutean Availability Zone, nonaktifkan penyeimbangan beban lintas zona.
6. Pilih Simpan perubahan.

Untuk menonaktifkan penyeimbangan beban lintas zona, setel `CrossZoneLoadBalancing` atribut penyeimbang beban Anda ke. `false`

Untuk menonaktifkan penyeimbangan beban lintas zona menggunakan AWS CLI

1. Gunakan perintah modify-load-balancer-attributes berikut:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --
load-balancer-attributes "{\"CrossZoneLoadBalancing\":{\"Enabled\":false}}"
```

Berikut adalah respons contohnya:

```
{
   "LoadBalancerAttributes": {
     "CrossZoneLoadBalancing": {
         "Enabled": false
       }
   },
   "LoadBalancerName": "my-loadbalancer"
 }
```

2. (Opsional) Gunakan describe-load-balancer-attributesperintah berikut untuk memverifikasi bahwa penyeimbangan beban lintas zona dinonaktifkan untuk penyeimbang beban Anda:

```
aws elb describe-load-balancer-attributes --load-balancer-name my-loadbalancer
```

Berikut adalah respons contohnya:

```
{
    "LoadBalancerAttributes": {
        "ConnectionDraining": {
            "Enabled": false,
            "Timeout": 300
        },
        "CrossZoneLoadBalancing": {
            "Enabled": false
        },
        "ConnectionSettings": {
            "IdleTimeout": 60
        },
        "AccessLog": {
            "Enabled": false
        }
    }
}
```

Konfigurasikan pengurasan koneksi untuk Classic Load Balancer Anda

Untuk memastikan bahwa Classic Load Balancer berhenti mengirim permintaan ke instans yang tidak terdaftar atau tidak sehat, sambil menjaga koneksi yang ada tetap terbuka, gunakan pengurasan koneksi. Hal ini memungkinkan penyeimbang beban untuk menyelesaikan permintaan dalam penerbangan yang dibuat untuk instans yang tidak terdaftar atau tidak sehat.

Saat mengaktifkan pengurasan koneksi, Anda dapat menentukan waktu maksimum penyeimbang beban untuk menjaga koneksi tetap hidup sebelum melaporkan instance sebagai tidak terdaftar. Nilai batas waktu maksimum dapat diatur antara 1 dan 3.600 detik (defaultnya adalah 300 detik). Ketika batas waktu maksimum tercapai, penyeimbang beban secara paksa menutup koneksi ke instance de-registrasi.

Jika instans de-registrasi tidak memiliki permintaan dalam penerbangan dan tidak ada koneksi aktif, Elastic Load Balancing segera menyelesaikan proses deregistrasi.

Sementara permintaan dalam penerbangan sedang dilayani, penyeimbang beban melaporkan status instance de-registrasi sebagai. `InService: Instance deregistration currently in progress` Ketika instans pembatalan pendaftaran selesai melayani semua permintaan dalam penerbangan, atau ketika batas waktu tunggu maksimum tercapai, penyeimbang beban melaporkan status instans sebagai. `0utOfService: Instance is not currently registered with the LoadBalancer`

Jika sebuah instance menjadi tidak sehat, penyeimbang beban melaporkan status instance sebagai`0utOfService`. Jika ada permintaan dalam penerbangan yang dibuat untuk contoh yang tidak sehat, mereka selesai. Batas batas waktu maksimum tidak berlaku untuk koneksi ke instance yang tidak sehat.

Jika instans Anda merupakan bagian dari grup Auto Scaling dan pengurasan koneksi diaktifkan untuk penyeimbang beban Anda, Auto Scaling menunggu permintaan dalam penerbangan selesai, atau batas waktu maksimum berakhir, sebelum menghentikan instans karena peristiwa penskalaan atau penggantian pemeriksaan kesehatan.

Anda dapat menonaktifkan pengurasan koneksi jika Anda ingin penyeimbang beban Anda segera menutup koneksi ke instans yang membatalkan pendaftaran atau menjadi tidak sehat. Ketika pengurasan koneksi dinonaktifkan, setiap permintaan dalam penerbangan yang dibuat untuk instans yang membatalkan pendaftaran atau tidak sehat tidak diselesaikan.

Daftar Isi

- Aktifkan pengeringan koneksi
- Nonaktifkan pengeringan koneksi

Aktifkan pengeringan koneksi

Anda dapat mengaktifkan pengurasan koneksi untuk penyeimbang beban Anda kapan saja.

Untuk mengaktifkan pengeringan koneksi menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian Konfigurasi lalu lintas, pilih Aktifkan pengurasan koneksi.
6. (Opsional) Untuk Timeout (interval pengeringan), masukkan nilai antara 1 dan 3.600 detik. Jika tidak, default 300 detik digunakan.
7. Pilih Simpan perubahan.

Untuk mengaktifkan pengeringan koneksi menggunakan AWS CLI

Gunakan perintah modify-load-balancer-attributes berikut:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --load-balancer-attributes "{\"ConnectionDraining\":{\"Enabled\":true,\"Timeout\":300}}"
```

Berikut adalah respons contohnya:

```
{
    "LoadBalancerAttributes": {
        "ConnectionDraining": {
            "Enabled": true,
            "Timeout": 300
        }
    },
    "LoadBalancerName": "my-loadbalancer"
```

```
}
```

Nonaktifkan pengeringan koneksi

Anda dapat menonaktifkan pengurasan koneksi untuk penyeimbang beban Anda kapan saja.

Untuk menonaktifkan pengeringan koneksi menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian konfigurasi Lalu lintas, batalkan pilihan Aktifkan pengurasan koneksi.
6. Pilih Simpan perubahan.

Untuk menonaktifkan pengeringan koneksi menggunakan AWS CLI

Gunakan perintah modify-load-balancer-attributes berikut:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --load-balancer-attributes "{\"ConnectionDraining\":{\"Enabled\":false}}"
```

Berikut adalah respons contohnya:

```
{
    "LoadBalancerAttributes": {
        "ConnectionDraining": {
            "Enabled": false,
            "Timeout": 300
        }
    },
    "LoadBalancerName": "my-loadbalancer"
}
```

Konfigurasikan sesi lengket untuk Classic Load Balancer Anda

Secara default, Classic Load Balancer merutekan setiap permintaan secara independen ke instance terdaftar dengan beban terkecil. Namun, Anda dapat menggunakan fitur sesi lengket (juga dikenal sebagai afinitas sesi), yang memungkinkan penyeimbang beban untuk mengikat sesi pengguna ke instance tertentu. Ini memastikan bahwa semua permintaan dari pengguna selama sesi dikirim ke instance yang sama.

Kunci untuk mengelola sesi lengket adalah menentukan berapa lama penyeimbang beban Anda harus secara konsisten merutekan permintaan pengguna ke instance yang sama. Jika aplikasi Anda memiliki cookie sesi sendiri, maka Anda dapat mengonfigurasi Elastic Load Balancing sehingga cookie sesi mengikuti durasi yang ditentukan oleh cookie sesi aplikasi. Jika aplikasi Anda tidak memiliki cookie sesi sendiri, maka Anda dapat mengonfigurasi Elastic Load Balancing untuk membuat cookie sesi dengan menentukan durasi lengket Anda sendiri.

Elastic Load Balancing membuat cookie, bernama AWSELB, yang digunakan untuk memetakan sesi ke instance.

Persyaratan

- Penyeimbang beban HTTP/HTTPS.
- Setidaknya satu contoh sehat di setiap Availability Zone.

Kompatibilitas

- RFC untuk properti jalur cookie memungkinkan garis bawah. Namun, Elastic Load Balancing URI mengkodekan karakter garis bawah `%5F` karena beberapa browser, seperti Internet Explorer 7, mengharapkan garis bawah untuk dikodekan URI sebagai. `%5F` Karena potensi untuk memengaruhi browser yang saat ini berfungsi, Elastic Load Balancing terus menyandikan karakter garis bawah URI. Misalnya, jika cookie memiliki propertipath=/my_path, Elastic Load Balancing mengubah properti ini dalam permintaan yang diteruskan ke. `path=/my%5Fpath`
- Anda tidak dapat mengatur `secure` bendera atau `HttpOnly` bendera pada cookie lengket sesi berbasis durasi Anda. Namun, cookie ini tidak mengandung data sensitif. Perhatikan bahwa jika Anda menyetel `secure` bendera atau `HttpOnly` bendera pada cookie lengket sesi yang dikendalikan aplikasi, itu juga diatur pada cookie. AWSELB
- Jika Anda memiliki titik koma di `Set-Cookie` bidang cookie aplikasi, penyeimbang beban mengabaikan cookie.

Daftar Isi

- Kelengketan sesi berbasis durasi
- Kelengketan sesi yang dikontrol aplikasi

Kelengketan sesi berbasis durasi

Penyeimbang beban menggunakan cookie khusus, AWSELB, untuk melacak instance untuk setiap permintaan ke setiap pendengar. Ketika penyeimbang beban menerima permintaan, pertama-tama penyeimbang beban memeriksa apakah cookie ini ada di permintaan. Jika demikian, permintaan dikirim ke instance yang ditentukan dalam cookie. Jika tidak ada cookie, load balancer memilih instance berdasarkan algoritma load balancing yang ada. Cookie dimasukkan ke dalam respons untuk mengikat permintaan berikutnya dari pengguna yang sama ke instance itu. Konfigurasi kebijakan stickiness mendefinisikan kedaluwarsa cookie, yang menetapkan durasi validitas untuk setiap cookie. Penyeimbang beban tidak menyegarkan waktu kedaluwarsa cookie dan tidak memeriksa apakah cookie kedaluwarsa sebelum menggunakannya. Setelah cookie kedaluwarsa, sesi tidak lagi lengket. Klien harus menghapus cookie dari toko cookie setelah kedaluwarsa.

Dengan permintaan CORS (cross-origin resource sharing), beberapa peramban memerlukan `SameSite=None; Secure` untuk mengaktifkan kelekatan. Dalam hal ini, Elastic Load Balancing membuat cookie lengket kedua AWSELBCORS, yang mencakup informasi yang sama dengan cookie lengket asli ditambah atribut ini. `SameSite` Klien menerima kedua cookie.

Jika sebuah instance gagal atau menjadi tidak sehat, penyeimbang beban menghentikan permintaan routing ke instance itu, dan memilih instance sehat baru berdasarkan algoritma load balancing yang ada. Permintaan dialihkan ke instance baru seolah-olah tidak ada cookie dan sesi tidak lagi lengket.

Jika klien beralih ke pendengar dengan port backend yang berbeda, kekakuan akan hilang.

Untuk mengaktifkan sesi lengket berbasis durasi untuk penyeimbang beban menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Listeners, pilih Kelola pendengar.
5. Pada halaman Kelola pendengar, cari listener yang akan diperbarui dan pilih Edit di bawah Cookie stickiness.
6. Pada pop-up pengaturan lengket Edit cookie, pilih Dihasilkan oleh penyeimbang beban.

7. (Opsional) Untuk periode kedaluwarsa, ketikkan periode kedaluwarsa cookie, dalam hitungan detik. Jika Anda tidak menentukan periode kedaluwarsa, sesi lengket berlangsung selama sesi browser.
8. Pilih Simpan perubahan untuk menutup jendela pop-up.
9. Pilih Simpan perubahan untuk kembali ke halaman detail penyeimbang beban.

Untuk mengaktifkan sesi lengket berbasis durasi untuk penyeimbang beban menggunakan AWS CLI

1. Gunakan perintah create-lb-cookie-stickiness-policy berikut untuk membuat kebijakan lengket cookie yang dihasilkan load balancer dengan periode kedaluwarsa cookie 60 detik:

```
aws elb create-lb-cookie-stickiness-policy --load-balancer-name my-loadbalancer --
policy-name my-duration-cookie-policy --cookie-expiration-period 60
```

2. Gunakan perintah set-load-balancer-policies-of-listener berikut untuk mengaktifkan kekakuan sesi untuk penyeimbang beban yang ditentukan:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-loadbalancer
 --load-balancer-port 443 --policy-names my-duration-cookie-policy
```

> **Note**
>
> `set-load-balancer-policies-of-listener`Perintah menggantikan kumpulan kebijakan saat ini yang terkait dengan port penyeimbang beban yang ditentukan. Setiap kali Anda menggunakan perintah ini, tentukan `--policy-names` opsi untuk mencantumkan semua kebijakan yang akan diaktifkan.

3. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan diaktifkan:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Respons mencakup informasi berikut, yang menunjukkan bahwa kebijakan diaktifkan untuk listener pada port yang ditentukan:

```
{
    "LoadBalancerDescriptions": [
        {
```

```
            ...
            "ListenerDescriptions": [
                {
                    "Listener": {
                        "InstancePort": 443,
                        "SSLCertificateId": "arn:aws:iam::123456789012:server-
certificate/my-server-certificate",
                        "LoadBalancerPort": 443,
                        "Protocol": "HTTPS",
                        "InstanceProtocol": "HTTPS"
                    },
                    "PolicyNames": [
                        "my-duration-cookie-policy",
                        "ELBSecurityPolicy-TLS-1-2-2017-01"
                    ]
                },
                ...
            ],
            ...
            "Policies": {
                "LBCookieStickinessPolicies": [
                 {
                        "PolicyName": "my-duration-cookie-policy",
                        "CookieExpirationPeriod": 60
                    }

                ],
                "AppCookieStickinessPolicies": [],
                "OtherPolicies": [
                    "ELBSecurityPolicy-TLS-1-2-2017-01"
                ]
            },
            ...
        }
    ]
}
```

Kelengketan sesi yang dikontrol aplikasi

Penyeimbang beban menggunakan cookie khusus untuk mengaitkan sesi dengan instance yang menangani permintaan awal, tetapi mengikuti masa pakai cookie aplikasi yang ditentukan dalam konfigurasi kebijakan. Load balancer hanya menyisipkan cookie lengket baru jika respons aplikasi

menyertakan cookie aplikasi baru. Cookie lengket penyeimbang beban tidak diperbarui dengan setiap permintaan. Jika cookie aplikasi dihapus atau kedaluwarsa secara eksplisit, sesi berhenti menjadi lengket sampai cookie aplikasi baru dikeluarkan.

Atribut berikut yang ditetapkan oleh instans back-end dikirim ke klien dalam cookie:`path`,,,,,,,,,`port`,,`domain`,`secure`,,`httponly`,`discard`,,`max-age`,`expires`, `versioncomment`, `commenturl` dan. `samesite`

Jika sebuah instance gagal atau menjadi tidak sehat, penyeimbang beban menghentikan permintaan routing ke instance itu, dan memilih instance sehat baru berdasarkan algoritma load balancing yang ada. Penyeimbang beban memperlakukan sesi sebagai sekarang “macet” ke instance sehat baru, dan melanjutkan permintaan perutean ke instance itu bahkan jika instance yang gagal kembali.

Untuk mengaktifkan kelengketan sesi yang dikontrol aplikasi menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Listeners, pilih Kelola pendengar.
5. Pada halaman Kelola pendengar, cari listener yang akan diperbarui dan pilih Edit di bawah Cookie stickiness.
6. Pilih Dihasilkan oleh aplikasi.
7. Untuk Nama Cookie, ketikkan nama cookie aplikasi Anda.
8. Pilih Simpan perubahan.

Untuk mengaktifkan kelengketan sesi yang dikontrol aplikasi menggunakan AWS CLI

1. Gunakan perintah create-app-cookie-stickiness-policy berikut untuk membuat kebijakan lengket cookie yang dihasilkan aplikasi:

```
aws elb create-app-cookie-stickiness-policy --load-balancer-name my-loadbalancer --
policy-name my-app-cookie-policy --cookie-name my-app-cookie
```

2. Gunakan perintah set-load-balancer-policies-of-listener berikut untuk mengaktifkan kekakuan sesi untuk penyeimbang beban:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-loadbalancer
 --load-balancer-port 443 --policy-names my-app-cookie-policy
```

> **Note**
>
> `set-load-balancer-policies-of-listener`Perintah menggantikan kumpulan kebijakan saat ini yang terkait dengan port penyeimbang beban yang ditentukan. Setiap kali Anda menggunakan perintah ini, tentukan `--policy-names` opsi untuk mencantumkan semua kebijakan yang akan diaktifkan.

3. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan lengket diaktifkan:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

4. Respons mencakup informasi berikut, yang menunjukkan bahwa kebijakan diaktifkan untuk listener pada port yang ditentukan:

```
{
    "LoadBalancerDescriptions": [
        {
            ...
            "ListenerDescriptions": [
                {
                    "Listener": {
                        "InstancePort": 443,
                        "SSLCertificateId": "arn:aws:iam::123456789012:server-
certificate/my-server-certificate",
                        "LoadBalancerPort": 443,
                        "Protocol": "HTTPS",
                        "InstanceProtocol": "HTTPS"
                    },
                    "PolicyNames": [
                        "my-app-cookie-policy",
                        "ELBSecurityPolicy-TLS-1-2-2017-01"
                    ]
                },
                {
                    "Listener": {
                        "InstancePort": 80,
```

```
                        "LoadBalancerPort": 80,
                        "Protocol": "TCP",
                        "InstanceProtocol": "TCP"
                    },
                    "PolicyNames": []
                }
            ],
            ...
            "Policies": {
                "LBCookieStickinessPolicies": [],
                "AppCookieStickinessPolicies": [
                {
                        "PolicyName": "my-app-cookie-policy",
                        "CookieName": "my-app-cookie"
                    }

                ],
                "OtherPolicies": [
                    "ELBSecurityPolicy-TLS-1-2-2017-01"
                ]
            },
            ...
        }
    ]
}
```

Konfigurasikan mode mitigasi desync untuk Classic Load Balancer

Mode mitigasi desync melindungi aplikasi Anda dari masalah karena HTTP Desync. Penyeimbang beban mengklasifikasikan setiap permintaan berdasarkan tingkat ancamannya, memungkinkan permintaan yang aman, lalu mengurangi risiko seperti yang ditentukan oleh mode mitigasi yang Anda tentukan. Mode mitigasi desync adalah monitor, defensive, dan strictest. Default-nya adalah mode defensive yang memberikan mitigasi tahan lama terhadap HTTP desync sambil mempertahankan ketersediaan aplikasi Anda. Anda dapat beralih ke mode strictest untuk memastikan bahwa aplikasi Anda hanya menerima permintaan yang sesuai dengan RFC 7230.

Pustaka http_desync_guardian menganalisis permintaan HTTP untuk mencegah serangan HTTP Desync. Untuk informasi selengkapnya, lihat HTTP Desync Guardian di github.

Daftar Isi

- Klasifikasi
- Modus
- Ubah mode mitigasi desync

> **Tip**
>
> Konfigurasi ini hanya berlaku untuk Classic Load Balancers. Untuk informasi yang berlaku untuk Application Load Balancers, lihat Mode mitigasi desync untuk Application Load Balancers.

Klasifikasi

Klasifikasinya adalah sebagai berikut.

- Patuh — Permintaan sesuai dengan RFC 7230 dan tidak menimbulkan ancaman keamanan yang diketahui.
- Dapat diterima — Permintaan tidak sesuai dengan RFC 7230, tetapi tidak menimbulkan ancaman keamanan yang diketahui.
- Ambigu — Permintaan tidak sesuai dengan RFC 7230, tetapi menimbulkan risiko karena berbagai server web dan proxy dapat menanganinya secara berbeda.
- Parah — Permintaan menimbulkan risiko keamanan yang tinggi. Penyeimbang beban memblokir permintaan, memberikan 400 respons ke klien, dan menutup koneksi klien.

Daftar berikut menjelaskan masalah untuk setiap klasifikasi.

Dapat diterima

- Header berisi karakter non-ASCII atau kontrol.
- Versi permintaan berisi nilai yang buruk.
- Ada header Content-Length dengan nilai 0 untuk permintaan GET atau HEAD.
- URI permintaan berisi spasi yang bukan URL yang dikodekan.

Ambigu

- URI Permintaan berisi karakter kontrol.

- Permintaan berisi header Transfer-Encoding dan header Content-Length.
- Ada beberapa header Content-Length dengan nilai yang sama.
- Header kosong atau ada garis dengan hanya spasi.
- Ada header yang dapat dinormalisasi ke Transfer-Encoding atau Content-Length menggunakan teknik normalisasi teks yang umum.
- Ada header Content-Length untuk permintaan GET atau HEAD.
- Ada header Transfer-Encoding untuk permintaan GET atau HEAD.

Parah

- URI permintaan berisi karakter null atau carriage return.
- Header Content-Length berisi nilai yang tidak dapat diuraikan atau bukan angka yang valid.
- Header berisi karakter null atau carriage return.
- Header Transfer-Encoding berisi nilai yang buruk.
- Metode permintaannya salah format.
- Versi permintaannya salah format.
- Ada beberapa header Content-Length dengan nilai yang berbeda.
- Ada beberapa Transfer-Encoding: chunked header.

Jika permintaan tidak sesuai dengan RFC 7230, penyeimbang beban akan menambah metrik `DesyncMitigationMode_NonCompliant_Request_Count`. Untuk informasi selengkapnya, lihat Metrik Classic Load Balancer.

Modus

Tabel berikut menjelaskan bagaimana Classic Load Balancers memperlakukan permintaan berdasarkan mode dan klasifikasi.

Klasifikasi	Mode monitor	Mode defensive	Mode strictest
Patuh	Diizinkan	Diizinkan	Diizinkan
Dapat diterima	Diizinkan	Diizinkan	Diblokir
Ambigu	Diizinkan	Diizinkan[1]	Diblokir

Klasifikasi	Mode monitor	Mode defensive	Mode strictest
Parah	Diizinkan	Diblokir	Diblokir

[1] Merutekan permintaan, tetapi menutup koneksi klien dan target.

Ubah mode mitigasi desync

Untuk memperbarui mode mitigasi desync menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bawah konfigurasi Lalu lintas, pilih Defensive - recommended, Strictest, atau Monitor.
6. Pilih Simpan perubahan.

Untuk memperbarui mode mitigasi desync menggunakan AWS CLI

Gunakan modify-load-balancer-attributesperintah dengan `elb.http.desyncmitigationmode` atribut yang disetel ke`monitor`,`defensive`, atau`strictest`.

```
aws elb modify-load-balancer-attributes --load-balancer-name my-load-balancer --load-balancer-attributes file://attribute.json
```

Berikut ini adalah isi dari `attribute.json`.

```
{
    "AdditionalAttributes": [
        {
            "Key": "elb.http.desyncmitigationmode",
            "Value": "strictest"
        }
    ]
}
```

Konfigurasikan protokol proxy untuk Classic Load Balancer Anda

Protokol proxy adalah protokol internet yang digunakan untuk membawa informasi koneksi dari sumber yang meminta koneksi ke tujuan yang koneksi diminta. Elastic Load Balancing menggunakan protokol proxy versi 1, yang menggunakan format header yang dapat dibaca manusia.

Secara default, saat Anda menggunakan Transmission Control Protocol (TCP) untuk koneksi front-end dan back-end, Classic Load Balancer meneruskan permintaan ke instance tanpa mengubah header permintaan. Jika Anda mengaktifkan protokol proxy, header yang dapat dibaca manusia ditambahkan ke header permintaan dengan informasi koneksi seperti alamat IP sumber, alamat IP tujuan, dan nomor port. Header kemudian dikirim ke instance sebagai bagian dari permintaan.

> **Note**
>
> AWS Management Console Tidak mendukung mengaktifkan protokol proxy.

Daftar Isi

- Header protokol proxy
- Prasyarat untuk mengaktifkan protokol proxy
- Aktifkan protokol proxy menggunakan AWS CLI
- Nonaktifkan protokol proxy menggunakan AWS CLI

Header protokol proxy

Header protokol proxy membantu Anda mengidentifikasi alamat IP klien ketika Anda memiliki penyeimbang beban yang menggunakan TCP untuk koneksi back-end. Karena penyeimbang beban mencegat lalu lintas antara klien dan instans Anda, log akses dari instans Anda berisi alamat IP penyeimbang beban, bukan klien asal. Anda dapat mengurai baris pertama permintaan untuk mengambil alamat IP klien Anda dan nomor port.

Alamat proxy di header untuk IPv6 adalah IPv6 alamat publik penyeimbang beban Anda. IPv6 Alamat ini cocok dengan alamat IP yang diselesaikan dari nama DNS penyeimbang beban Anda, yang dimulai dengan salah satu atau`ipv6`. `dualstack` Jika klien terhubung IPv4, alamat proxy di header adalah IPv4 alamat pribadi penyeimbang beban, yang tidak dapat diselesaikan melalui pencarian DNS.

Baris protokol proxy adalah satu baris yang diakhiri dengan carriage return dan line feed (`"\r\n"`), dan memiliki bentuk berikut:

```
PROXY_STRING + single space + INET_PROTOCOL + single space + CLIENT_IP + single space +
 PROXY_IP + single space + CLIENT_PORT + single space + PROXY_PORT + "\r\n"
```

Contoh: IPv4

Berikut ini adalah contoh dari baris protokol proxy untuk IPv4.

```
PROXY TCP4 198.51.100.22 203.0.113.7 35646 80\r\n
```

Prasyarat untuk mengaktifkan protokol proxy

Sebelum memulai, lakukan hal berikut:

- Konfirmasikan bahwa penyeimbang beban Anda tidak berada di belakang server proxy dengan protokol proxy diaktifkan. Jika protokol proxy diaktifkan pada server proxy dan penyeimbang beban, penyeimbang beban menambahkan header lain ke permintaan, yang sudah memiliki header dari server proxy. Bergantung pada bagaimana instans Anda dikonfigurasi, duplikasi ini dapat mengakibatkan kesalahan.
- Konfirmasikan bahwa instans Anda dapat memproses informasi protokol proxy.
- Konfirmasikan bahwa pengaturan pendengar Anda mendukung protokol proxy. Untuk informasi selengkapnya, lihat Konfigurasi pendengar untuk Classic Load Balancers.

Aktifkan protokol proxy menggunakan AWS CLI

Untuk mengaktifkan protokol proxy, Anda harus membuat kebijakan tipe `ProxyProtocolPolicyType` dan kemudian mengaktifkan kebijakan pada port instance.

Gunakan prosedur berikut untuk membuat kebijakan baru untuk jenis penyeimbang beban Anda`ProxyProtocolPolicyType`, setel kebijakan yang baru dibuat ke instance di port`80`, dan verifikasi apakah kebijakan tersebut diaktifkan.

Untuk mengaktifkan protokol proxy untuk penyeimbang beban Anda

1. (Opsional) Gunakan perintah describe-load-balancer-policy-types berikut untuk membuat daftar kebijakan yang didukung oleh Elastic Load Balancing:

```
aws elb describe-load-balancer-policy-types
```

Tanggapan tersebut mencakup nama dan deskripsi jenis kebijakan yang didukung. Berikut ini menunjukkan output untuk `ProxyProtocolPolicyType` jenis:

```
{
    "PolicyTypeDescriptions": [
        ...
        {
            "PolicyAttributeTypeDescriptions": [
                {
                    "Cardinality": "ONE",
                    "AttributeName": "ProxyProtocol",
                    "AttributeType": "Boolean"
                }
            ],
            "PolicyTypeName": "ProxyProtocolPolicyType",
            "Description": "Policy that controls whether to include the IP address
 and port of the originating
request for TCP messages. This policy operates on TCP/SSL listeners only"
        },
        ...
    ]
}
```

2. Gunakan create-load-balancer-policyperintah berikut untuk membuat kebijakan yang mengaktifkan protokol proxy:

```
aws elb create-load-balancer-policy --load-balancer-name my-loadbalancer --policy-
name my-ProxyProtocol-policy --policy-type-name ProxyProtocolPolicyType --policy-
attributes AttributeName=ProxyProtocol,AttributeValue=true
```

3. Gunakan for-backend-server perintah berikut set-load-balancer-policies- untuk mengaktifkan kebijakan yang baru dibuat pada port yang ditentukan. Perhatikan bahwa perintah ini menggantikan kumpulan kebijakan yang diaktifkan saat ini. Oleh karena itu, `--policy-names` opsi harus menentukan kebijakan yang Anda tambahkan ke daftar (misalnya,`my-ProxyProtocol-policy`) dan kebijakan apa pun yang saat ini diaktifkan (misalnya,`my-existing-policy`).

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-loadbalancer --instance-port 80 --policy-names my-ProxyProtocol-policy my-existing-policy
```

4. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa protokol proxy diaktifkan:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Tanggapan tersebut mencakup informasi berikut, yang menunjukkan bahwa `my-ProxyProtocol-policy` kebijakan tersebut terkait dengan port`80`.

```
{
    "LoadBalancerDescriptions": [
        {
            ...
            "BackendServerDescriptions": [
                {
                    "InstancePort": 80,
                    "PolicyNames": [
                        "my-ProxyProtocol-policy"
                    ]
                }
            ],
            ...
        }
    ]
}
```

Nonaktifkan protokol proxy menggunakan AWS CLI

Anda dapat menonaktifkan kebijakan yang terkait dengan instans Anda dan kemudian mengaktifkannya di lain waktu.

Untuk menonaktifkan kebijakan protokol proxy

1. Gunakan for-backend-server perintah berikut set-load-balancer-policies- untuk menonaktifkan kebijakan protokol proxy dengan menghilangkannya dari `--policy-names` opsi, tetapi termasuk kebijakan lain yang harus tetap diaktifkan (misalnya,`my-existing-policy`).

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-loadbalancer --instance-port 80 --policy-names my-existing-policy
```

Jika tidak ada kebijakan lain untuk mengaktifkan, tentukan string kosong dengan `--policy-names` opsi sebagai berikut:

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-loadbalancer --instance-port 80 --policy-names "[]"
```

2. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan dinonaktifkan:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Tanggapan tersebut mencakup informasi berikut, yang menunjukkan bahwa tidak ada port yang terkait dengan kebijakan.

```
{
    "LoadBalancerDescriptions": [
        {
            ...
            "BackendServerDescriptions": [],
            ...
        }
    ]
}
```

Tandai Classic Load Balancer Anda

Tag membantu Anda mengategorikan penyeimbang beban dengan cara yang berbeda, misalnya berdasarkan tujuan, pemilik, atau lingkungan.

Anda dapat menambahkan beberapa tag ke setiap Classic Load Balancer. Kunci tag harus unik untuk setiap penyeimbang beban. Jika Anda menambahkan tag dengan kunci yang sudah terkait dengan penyeimbang beban, kunci akan memperbarui nilai tag tersebut.

Setelah selesai dengan tag, Anda dapat menghapusnya dari penyeimbang beban Anda.

Daftar Isi

- Batasan tag
- Tambahkan tanda
- Hapus tag

Batasan tag

Batasan dasar berikut berlaku untuk tanda:

- 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.

Tambahkan tanda

Anda dapat menambahkan tag ke penyeimbang beban Anda kapan saja.

Untuk menambahkan tag menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Di bagian tab Tanda, pilih Kelola tanda.
5. Pada halaman Kelola tag, untuk setiap tag, pilih Tambahkan tag baru lalu tentukan kunci dan nilai.
6. Setelah Anda selesai menambahkan tag, pilih Simpan perubahan.

Untuk menambahkan tag menggunakan AWS CLI

Gunakan perintah add-tag berikut untuk menambahkan tag yang ditentukan:

```
aws elb add-tags --load-balancer-name my-loadbalancer --tag "Key=project,Value=lima"
```

Hapus tag

Anda dapat menghapus tag dari penyeimbang beban Anda setiap kali Anda selesai dengan mereka.

Untuk menghapus tag menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Di bagian tab Tanda, pilih Kelola tanda.
5. Pada halaman Kelola tag, pilih Hapus di samping setiap tag yang ingin Anda hapus.
6. Setelah Anda selesai menghapus tag, pilih Simpan perubahan.

Untuk menghapus tag menggunakan AWS CLI

Gunakan perintah hapus-tag berikut untuk menghapus tag dengan kunci yang ditentukan:

```
aws elb remove-tags --load-balancer-name my-loadbalancer --tag project
```

Konfigurasikan subnet untuk Classic Load Balancer Anda

Saat Anda menambahkan subnet ke penyeimbang beban Anda, Elastic Load Balancing membuat node penyeimbang beban di Availability Zone. Node penyeimbang beban menerima lalu lintas dari klien dan meneruskan permintaan ke instans terdaftar yang sehat di satu atau beberapa Availability Zone. Kami menyarankan Anda menambahkan satu subnet per Availability Zone untuk setidaknya dua Availability Zone. Ini meningkatkan ketersediaan penyeimbang beban Anda. Perhatikan bahwa Anda dapat memodifikasi subnet untuk penyeimbang beban Anda kapan saja.

Pilih subnet dari Availability Zone yang sama dengan instans Anda. Jika penyeimbang beban Anda adalah penyeimbang beban yang menghadap ke internet, Anda harus memilih subnet publik agar instans back-end Anda menerima lalu lintas dari penyeimbang beban (bahkan jika instance back-end berada dalam subnet pribadi). Jika penyeimbang beban Anda adalah penyeimbang beban internal,

sebaiknya pilih subnet pribadi. Untuk informasi selengkapnya tentang subnet untuk penyeimbang beban Anda, lihat. Rekomendasi untuk VPC Anda

Untuk menambahkan subnet, daftarkan instance di Availability Zone dengan load balancer, lalu lampirkan subnet dari Availability Zone tersebut ke load balancer. Untuk informasi selengkapnya, lihat Daftarkan instans dengan Classic Load Balancer Anda.

Setelah Anda menambahkan subnet, penyeimbang beban memulai perutean permintaan ke instance terdaftar di Availability Zone yang sesuai. Secara default, penyeimbang beban merutekan permintaan secara merata di seluruh Availability Zones untuk subnetnya. Untuk merutekan permintaan secara merata di seluruh instans terdaftar di Availability Zones untuk subnetnya, aktifkan penyeimbangan beban lintas zona. Untuk informasi selengkapnya, lihat Konfigurasikan load balancing lintas zona untuk Classic Load Balancer.

Anda mungkin ingin menghapus subnet dari penyeimbang beban sementara ketika Availability Zone tidak memiliki instans terdaftar yang sehat, atau ketika Anda ingin memecahkan masalah atau memperbarui instans terdaftar. Setelah Anda menghapus subnet, penyeimbang beban menghentikan permintaan perutean ke instance terdaftar di Availability Zone, tetapi terus merutekan permintaan ke instance terdaftar di Availability Zones untuk subnet yang tersisa. Perhatikan bahwa setelah Anda menghapus subnet, instance di subnet tersebut tetap terdaftar dengan penyeimbang beban, tetapi Anda dapat membatalkan pendaftarannya jika Anda mau. Untuk informasi selengkapnya, lihat Daftarkan instans dengan Classic Load Balancer Anda.

Daftar Isi

- Persyaratan
- Konfigurasikan subnet menggunakan konsol
- Konfigurasikan subnet menggunakan CLI

Persyaratan

Saat memperbarui subnet untuk penyeimbang beban, Anda harus memenuhi persyaratan berikut:

- Load balancer harus memiliki setidaknya satu subnet setiap saat.
- Anda dapat menambahkan paling banyak satu subnet per Availability Zone.
- Anda tidak dapat menambahkan subnet Zona Lokal.

Karena ada yang terpisah APIs untuk menambah dan menghapus subnet dari penyeimbang beban, Anda harus mempertimbangkan urutan operasi dengan hati-hati saat menukar subnet saat ini untuk subnet baru untuk memenuhi persyaratan ini. Selain itu, Anda harus menambahkan subnet sementara dari Availability Zone lain jika Anda perlu menukar semua subnet untuk penyeimbang beban Anda. Misalnya, jika penyeimbang beban Anda memiliki Availability Zone tunggal dan Anda perlu menukar subnetnya dengan subnet lain, Anda harus terlebih dahulu menambahkan subnet dari Availability Zone kedua. Kemudian Anda dapat menghapus subnet dari Availability Zone asli (tanpa pergi di bawah satu subnet), menambahkan subnet baru dari Availability Zone asli (tanpa melebihi satu subnet per Availability Zone), dan kemudian menghapus subnet dari Availability Zone kedua (jika hanya diperlukan untuk melakukan swap).

Konfigurasikan subnet menggunakan konsol

Gunakan prosedur berikut untuk menambah atau menghapus subnet menggunakan konsol.

Untuk mengkonfigurasi subnet menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Pemetaan jaringan, pilih Edit subnet.
5. Pada halaman Edit subnet, di bagian pemetaan Jaringan, tambahkan dan hapus subnet sesuai kebutuhan..
6. Setelah selesai, pilih Simpan perubahan.

Konfigurasikan subnet menggunakan CLI

Gunakan contoh berikut untuk menambah atau menghapus subnet menggunakan. AWS CLI

Untuk menambahkan subnet ke penyeimbang beban Anda menggunakan CLI

Gunakan perintah attach-load-balancer-to-subnets berikut untuk menambahkan dua subnet ke penyeimbang beban Anda:

```
aws elb attach-load-balancer-to-subnets --load-balancer-name my-load-balancer --
subnets subnet-dea770a9 subnet-fb14f6a2
```

Respons mencantumkan semua subnet untuk penyeimbang beban. Misalnya:

```
{
    "Subnets": [
        "subnet-5c11033e",
        "subnet-dea770a9",
        "subnet-fb14f6a2"
    ]
}
```

Untuk menghapus subnet menggunakan AWS CLI

Gunakan perintah detach-load-balancer-from-subnets berikut untuk menghapus subnet yang ditentukan dari penyeimbang beban yang ditentukan:

```
aws elb detach-load-balancer-from-subnets --load-balancer-name my-loadbalancer --
subnets subnet-450f5127
```

Respons mencantumkan subnet yang tersisa untuk penyeimbang beban. Misalnya:

```
{
    "Subnets": [
        "subnet-15aaab61"
    ]
}
```

Konfigurasikan grup keamanan untuk Classic Load Balancer Anda

Ketika Anda menggunakan AWS Management Console untuk membuat penyeimbang beban, Anda dapat memilih grup keamanan yang ada atau membuat yang baru. Jika Anda memilih grup keamanan yang ada, itu harus mengizinkan lalu lintas di kedua arah ke pendengar dan port pemeriksaan kesehatan untuk penyeimbang beban. Jika Anda memilih untuk membuat grup keamanan, konsol secara otomatis menambahkan aturan untuk mengizinkan semua lalu lintas di port ini.

[VPC Nondefault] Jika Anda menggunakan AWS CLI atau API membuat penyeimbang beban di VPC nondefault, tetapi Anda tidak menentukan grup keamanan, penyeimbang beban Anda secara otomatis dikaitkan dengan grup keamanan default untuk VPC.

[VPC default] Jika Anda menggunakan API AWS CLI atau untuk membuat penyeimbang beban di VPC default, Anda tidak dapat memilih grup keamanan yang ada untuk penyeimbang beban

Anda. Sebagai gantinya, Elastic Load Balancing menyediakan grup keamanan dengan aturan untuk mengizinkan semua lalu lintas pada port yang ditentukan untuk penyeimbang beban. Elastic Load Balancing hanya membuat satu grup keamanan seperti itu per AWS akun, dengan nama formulir *id* default_elb_ (misalnya,). `default_elb_fc5fbed3-0405-3b7d-a328-ea290EXAMPLE` Penyeimbang beban berikutnya yang Anda buat di VPC default juga menggunakan grup keamanan ini. Pastikan untuk meninjau aturan grup keamanan untuk memastikan bahwa mereka mengizinkan lalu lintas pada pendengar dan port pemeriksaan kesehatan untuk penyeimbang beban baru. Saat Anda menghapus penyeimbang beban, grup keamanan ini tidak dihapus secara otomatis.

Jika Anda menambahkan listener ke penyeimbang beban yang ada, Anda harus meninjau grup keamanan Anda untuk memastikan mereka mengizinkan lalu lintas pada port listener baru di kedua arah.

Daftar Isi

- Aturan yang disarankan untuk grup keamanan penyeimbang beban
- Tetapkan grup keamanan menggunakan konsol
- Tetapkan grup keamanan menggunakan AWS CLI

Aturan yang disarankan untuk grup keamanan penyeimbang beban

Grup keamanan untuk penyeimbang beban Anda harus memungkinkan mereka berkomunikasi dengan instans Anda. Aturan yang direkomendasikan tergantung pada jenis penyeimbang beban, menghadap ke internet atau internal.

Penyeimbang beban yang menghadap ke internet

Tabel berikut menunjukkan aturan masuk yang direkomendasikan untuk penyeimbang beban yang menghadap ke internet.

Sumber	Protokol	Baris Port	Komentar
0.0.0.0/0	TCP	*listener*	Izinkan semua lalu lintas masuk pada port listener penyeimbang beban

Tabel berikut menunjukkan aturan keluar yang direkomendasikan untuk penyeimbang beban yang menghadap ke internet.

Tujuan	Protokol	Baris Port	Komentar
instance security group	TCP	*instance listener*	Izinkan lalu lintas keluar ke instans pada port listener instans
instance security group	TCP	*health check*	Izinkan lalu lintas keluar ke instans pada port pemeriksaan kondisi

Penyeimbang beban internal

Tabel berikut menunjukkan aturan masuk yang direkomendasikan untuk penyeimbang beban internal.

Sumber	Protokol	Baris Port	Komentar
VPC CIDR	TCP	*listener*	Izinkan lalu lintas masuk dari VPC CIDR pada port listener penyeimbang beban

Tabel berikut menunjukkan aturan keluar yang direkomendasikan untuk penyeimbang beban internal.

Tujuan	Protokol	Baris Port	Komentar
instance security group	TCP	*instance listener*	Izinkan lalu lintas keluar ke instans pada port listener instans
instance security group	TCP	*health check*	Izinkan lalu lintas keluar ke instans pada port pemeriksaan kondisi

Tetapkan grup keamanan menggunakan konsol

Gunakan prosedur berikut untuk mengubah grup keamanan yang terkait dengan penyeimbang beban Anda.

Untuk memperbarui grup keamanan yang ditetapkan ke penyeimbang beban Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Keamanan, pilih Edit.
5. Pada halaman Edit grup keamanan, Di bawah Grup keamanan, tambahkan atau hapus grup keamanan sesuai kebutuhan.

   Anda dapat menambahkan hingga lima grup keamanan.
6. Setelah selesai, pilih Simpan perubahan.

Tetapkan grup keamanan menggunakan AWS CLI

Gunakan perintah apply-security-groups-to-load-balancer berikut untuk mengaitkan grup keamanan dengan penyeimbang beban. Grup keamanan yang ditentukan mengganti grup keamanan yang sebelumnya terkait.

```
aws elb apply-security-groups-to-load-balancer --load-balancer-name my-loadbalancer --
security-groups sg-53fae93f
```

Berikut adalah respons contohnya:

```
{
  "SecurityGroups": [
     "sg-53fae93f"
  ]
}
```

Konfigurasikan jaringan ACLs untuk Classic Load Balancer Anda

Daftar kontrol akses jaringan (ACL) default untuk VPC memungkinkan semua lalu lintas masuk dan keluar. Jika Anda membuat jaringan kustom ACLs, Anda harus menambahkan aturan yang memungkinkan penyeimbang beban dan instance untuk berkomunikasi.

Aturan yang disarankan untuk subnet untuk penyeimbang beban Anda bergantung pada jenis penyeimbang beban, menghadap ke internet atau internal.

Penyeimbang beban yang menghadap ke internet

Berikut ini adalah aturan masuk yang direkomendasikan untuk penyeimbang beban yang menghadap ke internet.

Sumber	Protokol	Baris Port	Komentar
0.0.0.0/0	TCP	*listener*	Izinkan semua lalu lintas masuk pada port listener penyeimbang beban
VPC CIDR	TCP	1024-65535	Izinkan lalu lintas masuk dari CIDR VPC pada port fana

Berikut ini adalah aturan keluar yang direkomendasikan untuk penyeimbang beban yang menghadap ke internet.

Tujuan	Protokol	Baris Port	Komentar
VPC CIDR	TCP	*instance listener*	Izinkan semua lalu lintas keluar pada port pendengar instance
VPC CIDR	TCP	*health check*	Izinkan semua lalu lintas keluar di port pemeriksaan kesehatan
0.0.0.0/0	TCP	1024-65535	Izinkan semua lalu lintas keluar di port fana

Penyeimbang beban internal

Berikut ini adalah aturan masuk yang direkomendasikan untuk penyeimbang beban internal.

Sumber	Protokol	Baris Port	Komentar
VPC CIDR	TCP	*listener*	Izinkan lalu lintas masuk dari VPC CIDR pada port listener penyeimbang beban
VPC CIDR	TCP	1024-65535	Izinkan lalu lintas masuk dari CIDR VPC pada port fana

Berikut ini adalah aturan keluar yang direkomendasikan untuk penyeimbang beban internal.

Tujuan	Protokol	Baris Port	Komentar
VPC CIDR	TCP	*instance listener*	Izinkan lalu lintas keluar ke CIDR VPC pada port pendengar instance
VPC CIDR	TCP	*health check*	Izinkan lalu lintas keluar ke CIDR VPC di port pemeriksaan kesehatan
VPC CIDR	TCP	1024-65535	Izinkan lalu lintas keluar ke CIDR VPC di port fana

Konfigurasikan nama domain khusus untuk Classic Load Balancer Anda

Setiap Classic Load Balancer menerima nama Domain Name System (DNS) default. Nama DNS ini mencakup nama AWS Wilayah tempat penyeimbang beban dibuat. Misalnya, jika Anda membuat penyeimbang beban bernama `my-loadbalancer` di Wilayah AS Barat (Oregon), penyeimbang beban Anda akan menerima nama DNS seperti. `my-loadbalancer-1234567890.us-west-2.elb.amazonaws.com` Untuk mengakses situs web pada instans Anda, Anda menempelkan nama DNS ini ke bidang alamat browser web. Namun, nama DNS ini tidak mudah diingat dan digunakan pelanggan Anda.

Jika Anda lebih suka menggunakan nama DNS yang ramah untuk penyeimbang beban Anda, sepertiwww.example.com, alih-alih nama DNS default, Anda dapat membuat nama domain khusus dan mengaitkannya dengan nama DNS untuk penyeimbang beban Anda. Ketika klien membuat permintaan menggunakan nama domain kustom ini, DNS server menyelesaikan ke nama DNS untuk penyeimbang beban Anda.

Daftar Isi

- Mengaitkan nama domain kustom Anda dengan nama penyeimbang beban Anda
- Menggunakan failover DNS Route 53 untuk penyeimbang beban Anda
- Memutuskan nama domain kustom Anda dari penyeimbang beban Anda

Mengaitkan nama domain kustom Anda dengan nama penyeimbang beban Anda

Pertama, jika Anda belum melakukannya, daftarkan nama domain 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 Daftar Pendaftar Terakreditasi.

Selanjutnya, gunakan layanan DNS Anda, seperti registrar domain Anda, untuk membuat catatan CNAME untuk merutekan kueri ke penyeimbang beban Anda. Untuk informasi lebih lanjut, lihat dokumentasi untuk server DNS Anda.

Atau, Anda dapat menggunakan Route 53 sebagai layanan DNS Anda. Anda membuat zona yang dihosting, yang berisi informasi tentang cara merutekan lalu lintas di internet untuk domain Anda, dan kumpulan catatan sumber daya alias, yang merutekan kueri untuk nama domain Anda ke penyeimbang beban Anda. Route 53 tidak mengenakan biaya untuk kueri DNS untuk kumpulan rekaman alias, dan Anda dapat menggunakan kumpulan rekaman alias untuk merutekan kueri DNS ke penyeimbang beban Anda untuk puncak zona domain Anda (misalnya,). example.com

Untuk informasi tentang mentransfer layanan DNS untuk domain yang ada ke Route 53, lihat Mengonfigurasi Route 53 sebagai layanan DNS Anda di Panduan Pengembang Amazon Route 53.

Terakhir, buat zona yang dihosting dan set catatan alias untuk domain Anda menggunakan Route 53. Untuk informasi selengkapnya, lihat Merutekan lalu lintas ke penyeimbang beban di Panduan Pengembang Amazon Route 53.

Menggunakan failover DNS Route 53 untuk penyeimbang beban Anda

Jika Anda menggunakan Route 53 untuk merutekan kueri DNS ke penyeimbang beban, Anda juga dapat mengonfigurasi failover DNS untuk penyeimbang beban menggunakan Route 53. Dalam konfigurasi failover, Route 53 memeriksa kesehatan EC2 instans terdaftar untuk penyeimbang beban untuk menentukan apakah tersedia. Jika tidak ada EC2 contoh sehat yang terdaftar di penyeimbang beban, atau jika penyeimbang beban itu sendiri tidak sehat, Route 53 mengarahkan lalu lintas ke sumber daya lain yang tersedia, seperti penyeimbang beban yang sehat atau situs web statis di Amazon S3.

Misalnya, misalkan Anda memiliki aplikasi web untuk`www.example.com`, dan Anda ingin instance redundan berjalan di belakang dua penyeimbang beban yang berada di Wilayah yang berbeda. Anda ingin lalu lintas terutama diarahkan ke penyeimbang beban di satu Wilayah, dan Anda ingin menggunakan penyeimbang beban di Wilayah lain sebagai cadangan selama kegagalan. Jika Anda mengonfigurasi failover DNS, Anda dapat menentukan penyeimbang beban primer dan sekunder (cadangan) Anda. Rute 53 mengarahkan lalu lintas ke penyeimbang beban utama jika tersedia, atau ke penyeimbang beban sekunder sebaliknya.

Menggunakan evaluasi kesehatan target

- Ketika mengevaluasi kesehatan target diatur ke `Yes` catatan alias untuk Classic Load Balancer, Route 53 mengevaluasi kesehatan sumber daya yang ditentukan oleh nilai. `alias target` Untuk Classic Load Balancer, Route 53 menggunakan pemeriksaan kesehatan instans yang terkait dengan penyeimbang beban.
- Jika setidaknya salah satu instans terdaftar di Classic Load Balancer sehat, Route 53 menandai catatan alias sebagai sehat. Route 53 kemudian mengembalikan catatan sesuai dengan kebijakan perutean Anda. Jika kebijakan routing failover digunakan, Route 53 mengembalikan catatan utama.
- Jika semua instans terdaftar untuk Classic Load Balancer tidak sehat, Route 53 menandai catatan alias sebagai tidak sehat. Route 53 kemudian mengembalikan catatan sesuai dengan kebijakan perutean Anda. Jika kebijakan routing failover digunakan, maka Route 53 mengembalikan catatan sekunder.

Untuk informasi selengkapnya, lihat Mengonfigurasi failover DNS di Panduan Pengembang Amazon Route 53.

Memutuskan nama domain kustom Anda dari penyeimbang beban Anda

Anda dapat memisahkan nama domain kustom Anda dari instance penyeimbang beban dengan terlebih dahulu menghapus kumpulan catatan sumber daya di zona yang dihosting, lalu menghapus zona yang dihosting. Untuk informasi selengkapnya, lihat Mengedit catatan dan Menghapus zona yang dihosting publik di Panduan Pengembang Amazon Route 53.

Pendengar untuk Classic Load Balancer Anda

Sebelum mulai menggunakan Elastic Load Balancing, Anda harus mengonfigurasi satu atau lebih pendengar untuk Classic Load Balancer Anda. Listener adalah proses memeriksa permintaan koneksi. Ini dikonfigurasi dengan protokol dan port untuk koneksi front-end (client to load balancer), dan protokol dan port untuk koneksi back-end (load balancer to back-end instance).

Elastic Load Balancing mendukung protokol berikut:

- HTTP
- HTTPS (HTTP aman)
- TCP
- SSL (TCP aman)

Protokol HTTPS menggunakan protokol SSL untuk membangun koneksi aman melalui lapisan HTTP. Anda juga dapat menggunakan protokol SSL untuk membuat koneksi aman melalui lapisan TCP.

Jika koneksi front-end menggunakan TCP atau SSL, maka koneksi back-end Anda dapat menggunakan TCP atau SSL. Jika koneksi front-end menggunakan HTTP atau HTTPS, maka koneksi back-end Anda dapat menggunakan HTTP atau HTTPS.

Instans back-end dapat mendengarkan pada port 1-65535.

Load balancer dapat mendengarkan pada port berikut: 1-65535

Daftar Isi

- Protokol
- HTTPS/SSL pendengar
- Konfigurasi pendengar untuk Classic Load Balancers
- Header HTTP dan Classic Load Balancer

Protokol

Komunikasi untuk aplikasi web biasa melewati lapisan perangkat keras dan perangkat lunak. Setiap lapisan menyediakan fungsi komunikasi tertentu. Kontrol atas fungsi komunikasi diteruskan dari satu

lapisan ke lapisan berikutnya, secara berurutan. Open System Interconnection (OSI) mendefinisikan kerangka model untuk menerapkan format standar untuk komunikasi, yang disebut protokol, di lapisan-lapisan ini. Untuk informasi lebih lanjut, lihat model OSI di Wikipedia.

Saat Anda menggunakan Elastic Load Balancing, Anda memerlukan pemahaman dasar tentang layer 4 dan layer 7. Layer 4 adalah layer transport yang menjelaskan koneksi Transmission Control Protocol (TCP) antara klien dan instance back-end Anda, melalui load balancer. Layer 4 adalah level terendah yang dapat dikonfigurasi untuk penyeimbang beban Anda. Layer 7 adalah layer aplikasi yang menjelaskan penggunaan koneksi Hypertext Transfer Protocol (HTTP) dan HTTPS (secure HTTP) dari klien ke load balancer dan dari load balancer ke instance back-end Anda.

Protokol Secure Sockets Layer (SSL) terutama digunakan untuk mengenkripsi data rahasia melalui jaringan yang tidak aman seperti internet. Protokol SSL menetapkan koneksi aman antara klien dan server back-end, dan memastikan bahwa semua data yang dilewatkan antara klien Anda dan server Anda bersifat pribadi dan integral.

Protokol TCP/SSL

Saat Anda menggunakan TCP (layer 4) untuk koneksi front-end dan back-end, penyeimbang beban Anda meneruskan permintaan ke instance back-end tanpa memodifikasi header. Setelah penyeimbang beban Anda menerima permintaan, ia mencoba membuka koneksi TCP ke instance back-end pada port yang ditentukan dalam konfigurasi listener.

Karena penyeimbang beban mencegat lalu lintas antara klien dan instance back-end Anda, log akses untuk instance back-end Anda berisi alamat IP penyeimbang beban alih-alih klien asal. Anda dapat mengaktifkan protokol proxy, yang menambahkan header dengan informasi koneksi klien, seperti alamat IP sumber, alamat IP tujuan, dan nomor port. Header kemudian dikirim ke instance back-end sebagai bagian dari permintaan. Anda dapat mengurai baris pertama dalam permintaan untuk mengambil informasi koneksi. Untuk informasi selengkapnya, lihat Konfigurasikan protokol proxy untuk Classic Load Balancer Anda.

Dengan menggunakan konfigurasi ini, Anda tidak menerima cookie untuk kekakuan sesi atau header X-Forwarded.

Protokol HTTP/HTTPS

Saat Anda menggunakan HTTP (lapisan 7) untuk koneksi front-end dan back-end, penyeimbang beban Anda mem-parsing header dalam permintaan sebelum mengirim permintaan ke instance back-end.

Untuk setiap contoh terdaftar dan sehat di balik HTTP/HTTPS load balancer, Elastic Load Balancing opens and maintains one or more TCP connections. These connections ensure that there is always an established connection ready to receive HTTP/HTTPS permintaan.

Permintaan HTTP dan respons HTTP menggunakan bidang header untuk mengirim informasi tentang pesan HTTP. Elastic Load Balancing mendukung `X-Forwarded-For` header. Karena penyeimbang beban mencegat lalu lintas antara klien dan server, log akses server Anda hanya berisi alamat IP penyeimbang beban. Untuk melihat alamat IP klien, gunakan header permintaan `X-Forwarded-For`. Untuk informasi selengkapnya, lihat X-Diteruskan-Untuk.

Saat Anda menggunakan HTTP/HTTPS, Anda dapat mengaktifkan sesi lengket pada penyeimbang beban Anda. Sesi lengket mengikat sesi pengguna ke instance back-end tertentu. Ini memastikan bahwa semua permintaan yang datang dari pengguna selama sesi dikirim ke instance back-end yang sama. Untuk informasi selengkapnya, lihat Konfigurasikan sesi lengket untuk Classic Load Balancer Anda.

Tidak semua ekstensi HTTP didukung di penyeimbang beban. Anda mungkin perlu menggunakan pendengar TCP jika penyeimbang beban tidak dapat menghentikan permintaan karena metode yang tidak terduga, kode respons, atau implementasi HTTP 1.0/1.1 non-standar lainnya.

HTTPS/SSL pendengar

Anda dapat membuat penyeimbang beban dengan fitur keamanan berikut.

Sertifikat server SSL

Jika Anda menggunakan HTTPS atau SSL untuk koneksi front-end Anda, Anda harus menerapkan sertifikat X.509 (sertifikat server SSL) pada penyeimbang beban Anda. Load balancer mendekripsi permintaan dari klien sebelum mengirimnya ke instance back-end (dikenal sebagai penghentian SSL). Untuk informasi selengkapnya, lihat Sertifikat SSL/TLS untuk Classic Load Balancer.

Jika Anda tidak ingin penyeimbang beban menangani penghentian SSL (dikenal sebagai pembongkaran SSL), Anda dapat menggunakan TCP untuk koneksi front-end dan back-end, dan menerapkan sertifikat pada instans terdaftar yang menangani permintaan.

Negosiasi SSL

Elastic Load Balancing menyediakan konfigurasi negosiasi SSL yang telah ditentukan sebelumnya yang digunakan untuk negosiasi SSL ketika koneksi dibuat antara klien dan penyeimbang

beban Anda. Konfigurasi negosiasi SSL menyediakan kompatibilitas dengan berbagai klien dan menggunakan algoritma kriptografi berkekuatan tinggi yang disebut cipher. Namun, beberapa kasus penggunaan mungkin memerlukan semua data di jaringan untuk dienkripsi dan hanya mengizinkan cipher tertentu. Beberapa standar kepatuhan keamanan (seperti PCI, SOX, dan sebagainya) mungkin memerlukan seperangkat protokol dan sandi khusus dari klien untuk memastikan bahwa standar keamanan terpenuhi. Dalam kasus seperti itu, Anda dapat membuat konfigurasi negosiasi SSL khusus, berdasarkan persyaratan spesifik Anda. Cipher dan protokol Anda akan berlaku dalam waktu 30 detik. Untuk informasi selengkapnya, lihat Konfigurasi negosiasi SSL untuk Classic Load Balancer.

Autentikasi server back-end

Jika Anda menggunakan HTTPS atau SSL untuk koneksi back-end Anda, Anda dapat mengaktifkan otentikasi instans terdaftar Anda. Anda kemudian dapat menggunakan proses otentikasi untuk memastikan bahwa instans hanya menerima komunikasi terenkripsi, dan untuk memastikan bahwa setiap instans terdaftar memiliki kunci publik yang benar.

Untuk informasi selengkapnya, lihat Mengkonfigurasi Autentikasi Server Back-end.

Konfigurasi pendengar untuk Classic Load Balancers

Tabel berikut menjelaskan kemungkinan konfigurasi untuk pendengar HTTP dan HTTPS untuk Classic Load Balancer.

Kasus penggunaan	Protokol front-end	Opsi front-end	Protokol back-end	Opsi back-end	Catatan
Penyeimbang beban HTTP dasar	HTTP	TA	HTTP	TA	• Mendukung header X-Forwarded
Amankan situs web atau aplikasi menggunakan Elastic Load Balancing untuk	HTTPS	Negosiasi SSL	HTTP	TA	• Mendukung header X-Forwarded • Memerlukan sertifikat SSL yang digunakan

Kasus penggunaan	Protokol front-end	Opsi front-end	Protokol back-end	Opsi back-end	Catatan
membongkar dekripsi SSL					pada penyeimbang beban
Amankan situs web atau aplikasi menggunakan end-to-end enkripsi	HTTPS	Negosiasi SSL	HTTPS	Otentikasi back-end	• Mendukung header X-Forwarded • Memerlukan sertifikat SSL yang digunakan pada penyeimbang beban dan instans terdaftar

Tabel berikut menjelaskan kemungkinan konfigurasi untuk pendengar TCP dan SSL untuk Classic Load Balancer.

Kasus penggunaan	Protokol front-end	Opsi front-end	Protokol back-end	Opsi back-end	Catatan
Penyeimbang beban TCP dasar	TCP	TA	TCP	TA	• Mendukung header protokol proxy
Amankan situs web atau aplikasi menggunakan Elastic Load	SSL	Negosiasi SSL	TCP	TA	• Memerlukan sertifikat SSL yang digunakan pada

Kasus penggunaan	Protokol front-end	Opsi front-end	Protokol back-end	Opsi back-end	Catatan
Balancing untuk membongkar dekripsi SSL					penyeimbang beban • Mendukung header protokol proxy
Amankan situs web atau aplikasi menggunakan end-to-end enkripsi dengan Elastic Load Balancing	SSL	Negosiasi SSL	SSL	Otentikasi back-end	• Memerlukan sertifikat SSL yang digunakan pada penyeimbang beban dan instans terdaftar • Tidak menyisipkan header SNI pada koneksi SSL back-end • Tidak mendukung header protokol proxy

Header HTTP dan Classic Load Balancer

Permintaan HTTP dan respons HTTP menggunakan bidang header untuk mengirim informasi tentang pesan HTTP. 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 (dan ditambahkan secara otomatis) yang banyak digunakan oleh aplikasi. Beberapa header HTTP non-standar memiliki awalan `X-Forwarded`. Classic Load Balancers mendukung header berikut`X-Forwarded`.

Untuk informasi lebih lanjut tentang koneksi HTTP, lihat Permintaan perutean di Panduan Pengguna Elastic Load Balancing.

Prasyarat

- Konfirmasikan bahwa setelan pendengar Anda mendukung header X-Forwarded. Untuk informasi selengkapnya, lihat Konfigurasi pendengar untuk Classic Load Balancers.
- Konfigurasikan server web Anda untuk mencatat alamat IP klien.

Header X-Diteruskan

- X-Diteruskan-Untuk
- X-Diteruskan-Proto
- Port-X-Diteruskan

X-Diteruskan-Untuk

Header `X-Forwarded-For` permintaan ditambahkan secara otomatis dan membantu Anda mengidentifikasi alamat IP klien saat Anda menggunakan penyeimbang beban HTTP atau HTTPS. Karena penyeimbang beban mencegat lalu lintas antara klien dan server, log akses server Anda hanya berisi alamat IP penyeimbang beban. Untuk melihat alamat IP klien, gunakan header permintaan `X-Forwarded-For`. Elastic Load Balancing menyimpan alamat IP klien dalam header permintaan `X-Forwarded-For` meneruskan header ke server Anda. Jika header permintaan `X-Forwarded-For` tidak disertakan dalam permintaan, penyeimbang beban membuat satu dengan alamat IP klien sebagai nilai permintaan. Jika tidak, penyeimbang beban menambahkan alamat IP klien ke header yang ada dan meneruskan header ke server Anda. Header permintaan `X-Forwarded-For` mungkin berisi beberapa alamat IP yang dipisahkan koma. Alamat paling kiri adalah IP klien tempat permintaan pertama kali dibuat. Ini diikuti oleh pengidentifikasi proxy berikutnya, dalam sebuah rantai.

Header permintaan `X-Forwarded-For` memiliki bentuk berikut:

```
X-Forwarded-For: client-ip-address
```

Berikut adalah contoh header permintaan `X-Forwarded-For` untuk klien dengan alamat IP `203.0.113.7`.

```
X-Forwarded-For: 203.0.113.7
```

Berikut ini adalah contoh header `X-Forwarded-For` permintaan untuk klien dengan IPv6 alamat`2001:DB8::21f:5bff:febf:ce22:8a2e`.

```
X-Forwarded-For: 2001:DB8::21f:5bff:febf:ce22:8a2e
```

X-Diteruskan-Proto

Header permintaan `X-Forwarded-Proto` membantu Anda mengidentifikasi protokol (HTTP atau HTTPS) yang digunakan klien untuk terhubung ke penyeimbang beban Anda. Log akses server Anda hanya berisi protokol yang digunakan antara server dan penyeimbang beban; mereka tidak berisi informasi tentang protokol yang digunakan antara klien dan penyeimbang beban. Untuk menentukan protokol yang digunakan antara klien dan penyeimbang beban, gunakan header permintaan `X-Forwarded-Proto`. Elastic Load Balancing menyimpan protokol yang digunakan antara klien dan penyeimbang beban di header permintaan `X-Forwarded-Proto` dan meneruskan headerdan meneruskan tajuk ke server Anda ke server Anda.

Aplikasi atau situs web Anda dapat menggunakan protokol yang tersimpan di header permintaan `X-Forwarded-Proto` untuk membuat respons yang mengarahkan ke URL yang sesuai.

Header permintaan `X-Forwarded-Proto` mengambil bentuk berikut:

```
X-Forwarded-Proto: originatingProtocol
```

Contoh berikut berisi header permintaan `X-Forwarded-Proto` untuk permintaan yang berasal dari klien sebagai permintaan HTTPS:

```
X-Forwarded-Proto: https
```

Port-X-Diteruskan

Header permintaan `X-Forwarded-Port` membantu Anda mengidentifikasi port tujuan yang digunakan klien untuk menyambung ke penyeimbang beban.

Pendengar HTTPS untuk Classic Load Balancer Anda

Anda dapat membuat penyeimbang beban yang menggunakan protokol SSL/TLS untuk koneksi terenkripsi (juga dikenal sebagai SSL offload). Fitur ini memungkinkan enkripsi lalu lintas antara penyeimbang beban Anda dan klien yang memulai sesi HTTPS, dan untuk koneksi antara penyeimbang beban dan instans Anda. EC2

Elastic Load Balancing menggunakan konfigurasi negosiasi Secure Sockets Layer (SSL), yang dikenal sebagai kebijakan keamanan, untuk menegosiasikan koneksi antara klien dan penyeimbang beban. Saat Anda menggunakan HTTPS/SSL untuk koneksi front-end Anda, Anda dapat menggunakan kebijakan keamanan yang telah ditentukan sebelumnya atau kebijakan keamanan khusus. Anda harus menerapkan sertifikat SSL pada penyeimbang beban Anda. Load balancer menggunakan sertifikat ini untuk mengakhiri koneksi dan kemudian mendekripsi permintaan dari klien sebelum mengirimnya ke instance. Load balancer menggunakan cipher suite statis untuk koneksi back-end. Anda dapat memilih untuk mengaktifkan otentikasi pada instans Anda secara opsional.

Classic Load Balancer tidak mendukung Server Name Indication (SNI). Anda dapat menggunakan salah satu alternatif berikut sebagai gantinya:

- Menyebarkan satu sertifikat pada penyeimbang beban, dan tambahkan Nama Alternatif Subjek (SAN) untuk setiap situs web tambahan. SANs memungkinkan Anda untuk melindungi beberapa nama host menggunakan satu sertifikat. Periksa dengan penyedia sertifikat Anda untuk informasi lebih lanjut tentang jumlah yang SANs mereka dukung per sertifikat dan cara menambah dan menghapus SANs.
- Gunakan pendengar TCP pada port 443 untuk koneksi front-end dan back-end. Penyeimbang beban meneruskan permintaan apa adanya, sehingga Anda dapat menangani penghentian HTTPS pada EC2 instance.

Classic Load Balancer tidak mendukung otentikasi TLS timbal balik (mTLS). Untuk dukungan mTL, buat pendengar TCP. Load balancer meneruskan permintaan apa adanya, sehingga Anda dapat mengimplementasikan mTL pada EC2 instance.

Daftar Isi

- Sertifikat SSL/TLS untuk Classic Load Balancer
- Konfigurasi negosiasi SSL untuk Classic Load Balancer
- Kebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer

- Membuat Classic Load Balancer dengan pendengar HTTPS
- Konfigurasikan listener HTTPS untuk Classic Load Balancer Anda
- Ganti sertifikat SSL untuk Classic Load Balancer Anda
- Perbarui konfigurasi negosiasi SSL Classic Load Balancer Anda

Sertifikat SSL/TLS untuk Classic Load Balancer

Jika Anda menggunakan HTTPS (SSL atau TLS) untuk pendengar front-end Anda, Anda harus menerapkan sertifikat SSL/TLS pada penyeimbang beban Anda. Load balancer menggunakan sertifikat untuk mengakhiri koneksi dan kemudian mendekripsi permintaan dari klien sebelum mengirimnya ke instance.

Protokol SSL dan TLS menggunakan sertifikat X.509 (sertifikat server SSL/TLS) untuk mengotentikasi klien dan aplikasi back-end. Sertifikat X.509 adalah bentuk identifikasi digital yang dikeluarkan oleh otoritas sertifikat (CA) dan berisi informasi identifikasi, masa berlaku, kunci publik, nomor seri, dan tanda tangan digital penerbit.

Anda dapat membuat sertifikat menggunakan AWS Certificate Manager atau alat yang mendukung protokol SSL dan TLS, seperti OpenSSL. Anda akan menentukan sertifikat ini saat membuat atau memperbarui pendengar HTTPS untuk penyeimbang beban Anda. Ketika Anda membuat sertifikat untuk digunakan dengan penyeimbang beban Anda, Anda harus menentukan nama domain.

Ketika Anda membuat sertifikat untuk digunakan dengan penyeimbang beban Anda, Anda harus menentukan nama domain. Nama domain pada sertifikat harus sesuai dengan catatan nama domain kustom. Jika tidak cocok, lalu lintas tidak akan dienkripsi karena koneksi TLS tidak dapat diverifikasi.

Anda harus menentukan nama domain yang sepenuhnya memenuhi syarat (FQDN) untuk sertifikat Anda, seperti `www.example.com` atau nama domain apex seperti. `example.com` Anda juga dapat menggunakan tanda bintang (*) sebagai kartu liar untuk melindungi beberapa nama situs di domain yang sama. Saat Anda meminta sertifikat kartu liar, tanda bintang (*) harus berada di posisi paling kiri dari nama domain dan hanya dapat melindungi satu tingkat subdomain. Misalnya, `*.example.com` melindungi`corp.example.com`, dan`images.example.com`, tetapi tidak dapat melindungi`test.login.example.com`. Perhatikan juga bahwa `*.example.com` melindungi hanya subdomain dari`example.com`, itu tidak melindungi domain telanjang atau apex (). `example.com` Nama kartu liar akan muncul di bidang Subjek dan di ekstensi Nama Alternatif Subjek sertifikat. Untuk informasi selengkapnya tentang sertifikat publik, lihat Meminta sertifikat publik di Panduan AWS Certificate Manager Pengguna.

Membuat atau mengimpor sertifikat SSL/TLS menggunakan AWS Certificate Manager

Kami menyarankan Anda menggunakan AWS Certificate Manager (ACM) untuk membuat atau mengimpor sertifikat untuk penyeimbang beban Anda. ACM terintegrasi dengan Elastic Load Balancing sehingga Anda dapat men-deploy sertifikat pada penyeimbang beban Anda. Untuk menyebarkan sertifikat pada penyeimbang beban Anda, sertifikat harus berada di Wilayah yang sama dengan penyeimbang beban. Untuk informasi selengkapnya, lihat Meminta sertifikat publik atau Mengimpor sertifikat di Panduan AWS Certificate Manager Pengguna.

Untuk mengizinkan pengguna menerapkan sertifikat pada penyeimbang beban Anda menggunakan AWS Management Console, Anda harus mengizinkan akses ke tindakan ACM API`ListCertificates`. Untuk informasi selengkapnya, lihat Daftar sertifikat di Panduan AWS Certificate Manager Pengguna.

> **Important**
>
> Anda tidak dapat menginstal sertifikat dengan kunci RSA 4096-bit atau kunci EC pada penyeimbang beban Anda melalui integrasi dengan ACM. Anda harus mengunggah sertifikat dengan kunci RSA 4096-bit atau kunci EC ke IAM untuk menggunakannya dengan penyeimbang beban Anda.

Impor sertifikat SSL/TLS menggunakan IAM

Jika Anda tidak menggunakan ACM, Anda dapat menggunakan alat SSL/TLS, seperti OpenSSL, untuk membuat permintaan penandatanganan sertifikat (CSR), mendapatkan CSR yang ditandatangani oleh CA untuk menghasilkan sertifikat, dan mengunggah sertifikat ke IAM. Untuk informasi selengkapnya, lihatBekerja dengan sertifikat serverdiPanduan Pengguna IAM.

Konfigurasi negosiasi SSL untuk Classic Load Balancer

Elastic Load Balancing menggunakan konfigurasi negosiasi Secure Socket Layer (SSL), yang dikenal sebagai kebijakan keamanan, untuk menegosiasikan koneksi SSL antara klien dan penyeimbang beban. Kebijakan keamanan adalah kombinasi protokol SSL, cipher SSL, dan opsi Preferensi Pesanan Server. Untuk informasi selengkapnya tentang mengonfigurasi koneksi SSL untuk penyeimbang beban Anda, lihat. Pendengar untuk Classic Load Balancer Anda

Daftar Isi

- Kebijakan Keamanan
- Protokol SSL
- Preferensi Pesanan Server
- Cipher SSL
- Cipher suite untuk koneksi back-end

Kebijakan Keamanan

Kebijakan keamanan menentukan sandi dan protokol mana yang didukung selama negosiasi SSL antara klien dan penyeimbang beban. Anda dapat mengonfigurasi Classic Load Balancers untuk menggunakan kebijakan keamanan yang telah ditentukan atau kustom.

Perhatikan bahwa sertifikat yang disediakan oleh AWS Certificate Manager (ACM) berisi kunci publik RSA. Oleh karena itu, Anda harus menyertakan cipher suite yang menggunakan RSA dalam kebijakan keamanan Anda jika Anda menggunakan sertifikat yang disediakan oleh ACM; jika tidak, koneksi TLS gagal.

Kebijakan keamanan yang telah ditentukan

Nama-nama kebijakan keamanan standar terbaru mencakup informasi versi berdasarkan tahun dan bulan mereka dirilis. Misalnya, kebijakan keamanan standar default adalah`ELBSecurityPolicy-2016-08`. Setiap kali kebijakan keamanan standar baru dirilis, Anda dapat memperbarui konfigurasi untuk menggunakannya.

Untuk informasi tentang protokol dan cipher yang diaktifkan untuk kebijakan keamanan yang telah ditetapkan, lihat. Kebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer

Kebijakan keamanan khusus

Anda dapat membuat konfigurasi negosiasi khusus dengan sandi dan protokol yang Anda butuhkan. Misalnya, beberapa standar kepatuhan keamanan (seperti PCI dan SOC) mungkin memerlukan seperangkat protokol dan sandi khusus untuk memastikan bahwa standar keamanan terpenuhi. Dalam kasus seperti itu, Anda dapat membuat kebijakan keamanan khusus untuk memenuhi standar tersebut.

Untuk informasi tentang membuat kebijakan keamanan khusus, lihatPerbarui konfigurasi negosiasi SSL Classic Load Balancer Anda.

Protokol SSL

Protokol SSL membuat koneksi aman antara klien dan server, dan memastikan bahwa semua data yang dilewatkan antara klien dan penyeimbang beban Anda bersifat pribadi.

Secure Sockets Layer (SSL) dan Transport Layer Security (TLS) adalah protokol kriptografi yang digunakan untuk mengenkripsi data rahasia melalui jaringan yang tidak aman seperti internet. Protokol TLS adalah versi yang lebih baru dari protokol SSL. Dalam dokumentasi Elastic Load Balancing, kami menyebut protokol SSL dan TLS sebagai protokol SSL.

Protokol yang direkomendasikan

Kami merekomendasikan TLS 1.2, yang digunakan dalam ELBSecurity kebijakan keamanan standar TLS-1-2-2017-01. Anda juga dapat menggunakan TLS 1.2 dalam kebijakan keamanan khusus Anda. Kebijakan keamanan default mendukung TLS 1.2 dan versi TLS sebelumnya, sehingga kurang aman dibandingkan ELBSecurity Policy-TLS-1-2-2017-01.

Protokol usang

Jika sebelumnya Anda mengaktifkan protokol SSL 2.0 dalam kebijakan khusus, sebaiknya Anda memperbarui kebijakan keamanan ke salah satu kebijakan keamanan yang telah ditentukan sebelumnya.

Preferensi Pesanan Server

Elastic Load Balancing mendukung opsi Preferensi Pesanan Server untuk menegosiasikan koneksi antara klien dan penyeimbang beban. Selama proses negosiasi koneksi SSL, klien dan penyeimbang beban menyajikan daftar sandi dan protokol yang masing-masing mereka dukung, sesuai urutan preferensi. Secara default, cipher pertama pada daftar klien yang cocok dengan salah satu cipher load balancer dipilih untuk koneksi SSL. Jika load balancer dikonfigurasi untuk mendukung Server Order Preference, maka load balancer memilih cipher pertama dalam daftarnya yang ada dalam daftar cipher klien. Ini memastikan bahwa penyeimbang beban menentukan cipher mana yang digunakan untuk koneksi SSL. Jika Anda tidak mengaktifkan Preferensi Pesanan Server, urutan cipher yang disajikan oleh klien digunakan untuk menegosiasikan koneksi antara klien dan penyeimbang beban.

Cipher SSL

Sebuah SSL cipher adalah algoritma enkripsi yang menggunakan kunci enkripsi untuk membuat pesan kode. Protokol SSL menggunakan beberapa cipher SSL untuk mengenkripsi data melalui internet.

Perhatikan bahwa sertifikat yang disediakan oleh AWS Certificate Manager (ACM) berisi kunci publik RSA. Oleh karena itu, Anda harus menyertakan cipher suite yang menggunakan RSA dalam kebijakan keamanan Anda jika Anda menggunakan sertifikat yang disediakan oleh ACM; jika tidak, koneksi TLS gagal.

Elastic Load Balancing mendukung cipher berikut untuk digunakan dengan Classic Load Balancers. Subset dari cipher ini digunakan oleh kebijakan SSL yang telah ditentukan. Semua cipher ini tersedia untuk digunakan dalam kebijakan khusus. Kami menyarankan Anda hanya menggunakan cipher yang disertakan dalam kebijakan keamanan default (yang memiliki tanda bintang). Banyak cipher lainnya tidak aman dan harus digunakan dengan risiko Anda sendiri.

Cipher

- ECDS-ECDSA- -GCM- * AES128 SHA256
- ECDHE-RSA- -GCM- * AES128 SHA256
- ECDHE-ECDSA- - * AES128 SHA256
- ECDHE-RSA- - * AES128 SHA256
- ECDHE-ECDSA- -SHA * AES128
- ECDHE-RSA- -SHA * AES128
- DHE-RSA- -SHA AES128
- ECDS-ECDSA- -GCM- * AES256 SHA384
- ECDHE-RSA- -GCM- * AES256 SHA384
- ECDHE-ECDSA- - * AES256 SHA384
- ECDHE-RSA- - * AES256 SHA384
- ECDHE-RSA- -SHA * AES256
- ECDHE-ECDSA- -SHA * AES256
- AES128-GCM- * SHA256
- AES128-SHA256 *
- AES128-SHA *

- AES256-GCM- * SHA384
- AES256-SHA256 *
- AES256-SHA *
- DHE-DSS- -SHA AES128
- CAMELLIA128-SHA
- EDH-RSA-DES- SHA CBC3
- DES- CBC3 -SHA
- ECDHE-RSA- -SHA RC4
- RC4-SHA
- ECDHE-ECDSA- -SHA RC4
- DHE-DSS- -GCM- AES256 SHA384
- DHE-RSA- -GCM- AES256 SHA384
- DHE-RSA- - AES256 SHA256
- DHE-DSS- - AES256 SHA256
- DHE-RSA- -SHA AES256
- DHE-DSS- -SHA AES256
- DHE-RSA- -SHA CAMELLIA256
- DHE-DSS- -SHA CAMELLIA256
- CAMELLIA256-SHA
- EDH-DSS-DES- SHA CBC3
- DHE-DSS- -GCM- AES128 SHA256
- DHE-RSA- -GCM- AES128 SHA256
- DHE-RSA- - AES128 SHA256
- DHE-DSS- - AES128 SHA256
- DHE-RSA- -SHA CAMELLIA128
- DHE-DSS- -SHA CAMELLIA128
- ADH- AES128 -GCM- SHA256
- ADH- AES128 -SHA
- ADH- - AES128 SHA256
- ADH- AES256 -GCM- SHA384

- ADH- AES256 -SHA
- ADH- - AES256 SHA256
- ADH- CAMELLIA128 -SHA
- ADH- CAMELLIA256 -SHA
- ADH-DES- -SHA CBC3
- ADH-DES-CBC-SHA
- ADH- - RC4 MD5
- ADH-BENIH-SHA
- DES-CBC-SHA
- DHE-DSS-BENIH-SHA
- DHE-RSA-BIJI-SHA
- EDH-DSS-DES-CBC-SHA
- EDH-RSA-DES-CBC-SHA
- IDEA-CBC-SHA
- RC4-MD5
- BENIH-SHA
- DES- CBC3 - MD5
- DES-CBC- MD5
- RC2-CBC- MD5
- PSK- AES256 -CBC-SHA
- PSK-3DES-EDE-CBC-SHA
- KRB5-DES- CBC3 -SHA
- KRB5-DES- - CBC3 MD5
- PSK- AES128 -CBC-SHA
- PSK- RC4 -SHA
- KRB5- RC4 -SHA
- KRB5-RC4-MD5
- KRB5-DES-CBC-SHA
- KRB5-DES-CBC- MD5
- EXP-EDH-RSA-DES-CBC-SHA

- EXP-EDH-DSS-DES-CBC-SHA
- EXP-ADH-DES-CBC-SHA
- EXP-DES-CBC-SHA
- EXP- RC2 -CBC- MD5
- EXP- KRB5 - RC2 -CBC-SHA
- EXP- KRB5 -DES-CBC-SHA
- EXP- KRB5 - RC2 -CBC- MD5
- EXP- KRB5 -DES-CBC- MD5
- EXP-ADH- - RC4 MD5
- EXP- - RC4 MD5
- EXP- KRB5 - RC4 -SHA
- EXP- KRB5 - - RC4 MD5

* Ini adalah cipher yang termasuk dalam kebijakan keamanan default, Policy-2016-08. ELBSecurity

Cipher suite untuk koneksi back-end

Classic Load Balancers menggunakan cipher suite statis untuk koneksi back-end. Jika Classic Load Balancer dan instans terdaftar Anda tidak dapat menegosiasikan koneksi, sertakan salah satu cipher berikut.

- AES256-GCM- SHA384
- AES256-SHA256
- AES256-SHA
- CAMELLIA256-SHA
- AES128-GCM- SHA256
- AES128-SHA256
- AES128-SHA
- CAMELLIA128-SHA
- RC4-SHA
- DES- CBC3 -SHA
- DES-CBC-SHA
- DHE-DSS- -GCM- AES256 SHA384

- DHE-RSA- -GCM- AES256 SHA384
- DHE-RSA- - AES256 SHA256
- DHE-DSS- - AES256 SHA256
- DHE-RSA- -SHA AES256
- DHE-DSS- -SHA AES256
- DHE-RSA- -SHA CAMELLIA256
- DHE-DSS- -SHA CAMELLIA256
- DHE-DSS- -GCM- AES128 SHA256
- DHE-RSA- -GCM- AES128 SHA256
- DHE-RSA- - AES128 SHA256
- DHE-DSS- - AES128 SHA256
- DHE-RSA- -SHA AES128
- DHE-DSS- -SHA AES128
- DHE-RSA- -SHA CAMELLIA128
- DHE-DSS- -SHA CAMELLIA128
- EDH-RSA-DES- SHA CBC3
- EDH-DSS-DES- SHA CBC3
- EDH-RSA-DES-CBC-SHA
- EDH-DSS-DES-CBC-SHA

Kebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer

Anda dapat memilih salah satu kebijakan keamanan yang telah ditentukan untuk pendengar HTTPS/SSL Anda. Anda dapat menggunakan salah satu `ELBSecurityPolicy-TLS` kebijakan untuk memenuhi standar kepatuhan dan keamanan yang mengharuskan menonaktifkan versi protokol TLS tertentu. Atau, Anda dapat membuat kebijakan keamanan khusus. Untuk informasi selengkapnya, lihat Perbarui konfigurasi negosiasi SSL.

Cipher berbasis RSA dan DSA khusus untuk algoritma penandatanganan yang digunakan untuk membuat sertifikat SSL. Pastikan untuk membuat sertifikat SSL menggunakan algoritma penandatanganan yang didasarkan pada sandi yang diaktifkan untuk kebijakan keamanan Anda.

Jika Anda memilih kebijakan yang diaktifkan untuk Preferensi Pesanan Server, penyeimbang beban menggunakan cipher dalam urutan yang ditentukan di sini untuk menegosiasikan koneksi antara klien dan penyeimbang beban. Jika tidak, penyeimbang beban menggunakan cipher dalam urutan yang disajikan oleh klien.

Bagian berikut menjelaskan kebijakan keamanan standar terbaru untuk Classic Load Balancer, termasuk protokol SSL dan cipher SSL yang diaktifkan. Anda juga dapat menjelaskan kebijakan yang telah ditentukan menggunakan describe-load-balancer-policiesperintah.

> **Tip**
>
> Informasi ini hanya berlaku untuk Classic Load Balancers. Untuk informasi yang berlaku untuk penyeimbang beban lainnya, lihat Kebijakan keamanan untuk kebijakan Application Load Balancer dan Keamanan untuk Network Load Balancer Anda.

Daftar Isi

- Protokol berdasarkan kebijakan
- Cipher berdasarkan kebijakan
- Kebijakan oleh cipher

Protokol berdasarkan kebijakan

Tabel berikut menjelaskan protokol TLS yang didukung oleh setiap kebijakan keamanan.

Kebijakan Keamanan	TLS 1.2	TLS 1.1	TLS 1.0
ELBSecurityKebijakan-TLS-1-2-2017-01	Ya	Tidak	Tidak
ELBSecurityKebijakan-TLS-1-1-2017-01	Ya	Ya	Tidak
ELBSecurityKebijakan-2016-08	Ya	Ya	Ya

Kebijakan Keamanan	TLS 1.2	TLS 1.1	TLS 1.0
ELBSecurityKebijakan-2015-05	Ya	Ya	Ya
ELBSecurityKebijakan-2015-03	Ya	Ya	Ya
ELBSecurityKebijakan-2015-02	Ya	Ya	Ya

Cipher berdasarkan kebijakan

Tabel berikut menjelaskan sandi yang didukung oleh setiap kebijakan keamanan.

Kebijakan keamanan	Cipher
ELBSecurityKebijakan-TLS-1-2-2017-01	• ECDHE-ECDSA- -GCM- AES128 SHA256 • ECDHE-RSA- -GCM- AES128 SHA256 • ECDHE-ECDSA- - AES128 SHA256 • ECDHE-RSA- - AES128 SHA256 • ECDHE-ECDSA- -GCM- AES256 SHA384 • ECDHE-RSA- -GCM- AES256 SHA384 • ECDHE-ECDSA- - AES256 SHA384 • ECDHE-RSA- - AES256 SHA384 • AES128-GCM- SHA256 • AES128-SHA256 • AES256-GCM- SHA384 • AES256-SHA256
ELBSecurityKebijakan-TLS-1-1-2017-01	• ECDHE-ECDSA- -GCM- AES128 SHA256 • ECDHE-RSA- -GCM- AES128 SHA256 • ECDHE-ECDSA- - AES128 SHA256 • ECDHE-RSA- - AES128 SHA256

Kebijakan keamanan	Cipher
	• ECDHE-ECDSA- -SHA AES128 • ECDHE-RSA- -SHA AES128 • ECDHE-ECDSA- -GCM- AES256 SHA384 • ECDHE-RSA- -GCM- AES256 SHA384 • ECDHE-ECDSA- - AES256 SHA384 • ECDHE-RSA- - AES256 SHA384 • ECDHE-ECDSA- -SHA AES256 • ECDHE-RSA- -SHA AES256 • AES128-GCM- SHA256 • AES128-SHA256 • AES128-SHA • AES256-GCM- SHA384 • AES256-SHA256 • AES256-SHA

Kebijakan keamanan	Cipher
ELBSecurityKebijakan-2016-08	• ECDHE-ECDSA- -GCM- AES128 SHA256 • ECDHE-RSA- -GCM- AES128 SHA256 • ECDHE-ECDSA- - AES128 SHA256 • ECDHE-RSA- - AES128 SHA256 • ECDHE-ECDSA- -SHA AES128 • ECDHE-RSA- -SHA AES128 • ECDHE-ECDSA- -GCM- AES256 SHA384 • ECDHE-RSA- -GCM- AES256 SHA384 • ECDHE-ECDSA- - AES256 SHA384 • ECDHE-RSA- - AES256 SHA384 • ECDHE-ECDSA- -SHA AES256 • ECDHE-RSA- -SHA AES256 • AES128-GCM- SHA256 • AES128-SHA256 • AES128-SHA • AES256-GCM- SHA384 • AES256-SHA256 • AES256-SHA

Kebijakan keamanan	Cipher
ELBSecurityKebijakan-2015-05	• ECDHE-ECDSA- -GCM- AES128 SHA256 • ECDHE-RSA- -GCM- AES128 SHA256 • ECDHE-ECDSA- - AES128 SHA256 • ECDHE-RSA- - AES128 SHA256 • ECDHE-ECDSA- -SHA AES128 • ECDHE-RSA- -SHA AES128 • ECDHE-ECDSA- -GCM- AES256 SHA384 • ECDHE-RSA- -GCM- AES256 SHA384 • ECDHE-ECDSA- - AES256 SHA384 • ECDHE-RSA- - AES256 SHA384 • ECDHE-ECDSA- -SHA AES256 • ECDHE-RSA- -SHA AES256 • AES128-GCM- SHA256 • AES128-SHA256 • AES128-SHA • AES256-GCM- SHA384 • AES256-SHA256 • AES256-SHA • DES- CBC3 -SHA

Kebijakan keamanan	Cipher
ELBSecurityKebijakan-2015-03	• ECDHE-ECDSA- -GCM- AES128 SHA256 • ECDHE-RSA- -GCM- AES128 SHA256 • ECDHE-ECDSA- - AES128 SHA256 • ECDHE-RSA- - AES128 SHA256 • ECDHE-ECDSA- -SHA AES128 • ECDHE-RSA- -SHA AES128 • ECDHE-ECDSA- -GCM- AES256 SHA384 • ECDHE-RSA- -GCM- AES256 SHA384 • ECDHE-ECDSA- - AES256 SHA384 • ECDHE-RSA- - AES256 SHA384 • ECDHE-ECDSA- -SHA AES256 • ECDHE-RSA- -SHA AES256 • AES128-GCM- SHA256 • AES128-SHA256 • AES128-SHA • AES256-GCM- SHA384 • AES256-SHA256 • AES256-SHA • DHE-RSA- -SHA AES128 • DHE-DSS- -SHA AES128 • DES- CBC3 -SHA

Kebijakan keamanan	Cipher
ELBSecurityKebijakan-2015-02	• ECDHE-ECDSA- -GCM- AES128 SHA256 • ECDHE-RSA- -GCM- AES128 SHA256 • ECDHE-ECDSA- - AES128 SHA256 • ECDHE-RSA- - AES128 SHA256 • ECDHE-ECDSA- -SHA AES128 • ECDHE-RSA- -SHA AES128 • ECDHE-ECDSA- -GCM- AES256 SHA384 • ECDHE-RSA- -GCM- AES256 SHA384 • ECDHE-ECDSA- - AES256 SHA384 • ECDHE-RSA- - AES256 SHA384 • ECDHE-ECDSA- -SHA AES256 • ECDHE-RSA- -SHA AES256 • AES128-GCM- SHA256 • AES128-SHA256 • AES128-SHA • AES256-GCM- SHA384 • AES256-SHA256 • AES256-SHA • DHE-RSA- -SHA AES128 • DHE-DSS- -SHA AES128

Kebijakan oleh cipher

Tabel berikut menjelaskan kebijakan keamanan yang mendukung setiap cipher.

Nama sandi	Kebijakan Keamanan	Rangkaian Penyandian
ECDHE-ECDSA-AESOpenSSL - 128-GCM- SHA256 IANA — TLS_ECDHE_ECDSA_DENGAN_AES_128_GCM_ SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c02b
ECDHE-RSA-AESOpenSSL - 128-GCM- SHA256 IANA — TLS_ECDHE_RSA_DENGAN_AES_128_GCM_ SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c02f
ECDHE-ECDSA-AESOpenSSL - 128-SHA256 IANA — TLS_ECDHE_ECDSA_DENGAN_AES_128_CBC_ SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c023
ECDHE-RSA-AESOpenSSL - 128-SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01	c027

Nama sandi	Kebijakan Keamanan	Rangkaian Penyandian
IANA — TLS_ECDHE_RSA_DENGAN_AES_128_CBC_ SHA256	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	
OpenSSL — ECDHE-ECDSA-AES 128-SHA IANA — TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c009
OpenSSL — ECDHE-RSA-AES 128-SHA IANA — TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c013
ECDHE-ECDSA-AESOpenSSL — 256-GCM- SHA384 IANA — TLS_ECDHE_ECDSA_DENGAN_AES_256_GCM_ SHA384	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c02c

Nama sandi	Kebijakan Keamanan	Rangkaian Penyandian
ECDHE-RSA-AESOpenSSL — 256-GCM- SHA384 IANA — TLS_ECDHE_RSA_DENGAN_AES_256_GCM_ SHA384	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c030
ECDHE-ECDSA-AESOpenSSL — 256-SHA384 IANA — TLS_ECDHE_ECDSA_DENGAN_AES_256_CBC_ SHA384	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c024
ECDHE-RSA-AESOpenSSL — 256-SHA384 IANA — TLS_ECDHE_RSA_DENGAN_AES_256_CBC_ SHA384	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c028

Nama sandi	Kebijakan Keamanan	Rangkaian Penyandian
OpenSSL — ECDHE-ECDSA-AES 256-SHA IANA — TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c014
OpenSSL — ECDHE-RSA-AES 256-SHA IANA — TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	c00a
AES128OpenSSL — -GCM- SHA256 IANA — TLS_RSA_WITH_AES_128_GCM_ SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	9c

Nama sandi	Kebijakan Keamanan	Rangkaian Penyandian
AES128OpenSSL — - SHA256 IANA — TLS_RSA_DENGAN_AES_128_CBC_ SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	3c
OpenSSL — AES128 -SHA IANA — TLS_RSA_WITH_AES_128_CBC_SHA	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	2f
AES256OpenSSL — -GCM- SHA384 IANA — TLS_RSA_WITH_AES_256_GCM_ SHA384	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	9d

Nama sandi	Kebijakan Keamanan	Rangkaian Penyandian
AES256OpenSSL — - SHA256 IANA — TLS_RSA_DENGAN_AES_256_CBC_ SHA256	• ELBSecurityKebijakan-TLS-1-2-2017-01 • ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	3d
OpenSSL — AES256 -SHA IANA — TLS_RSA_WITH_AES_256_CBC_SHA	• ELBSecurityKebijakan-TLS-1-1-2017-01 • ELBSecurityKebijakan-2016-08 • ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	35
OpenSSL — DHE-RSA-AES 128-SHA IANA — TLS_DHE_RSA_WITH_AES_128_CBC_SHA	• ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	33
OpenSSL — DHE-DSS-AES 128-SHA IANA — TLS_DHE_DSS_WITH_AES_128_CBC_SHA	• ELBSecurityKebijakan-2015-03 • ELBSecurityKebijakan-2015-02	32
OpenSSL — DES- -SHA CBC3 IANA — TLS_RSA_DENGAN_3DES_EDE_CBC_SHA	• ELBSecurityKebijakan-2015-05 • ELBSecurityKebijakan-2015-03	0a

Membuat Classic Load Balancer dengan pendengar HTTPS

Load balancer mengambil permintaan dari klien dan mendistribusikannya ke seluruh EC2 instance yang terdaftar di load balancer.

Anda dapat membuat penyeimbang beban yang mendengarkan pada port HTTP (80) dan HTTPS (443). Jika Anda menentukan bahwa listener HTTPS mengirim permintaan ke instance di port 80, penyeimbang beban menghentikan permintaan dan komunikasi dari penyeimbang beban ke instance tidak dienkripsi. Jika pendengar HTTPS mengirimkan permintaan ke instance di port 443, komunikasi dari penyeimbang beban ke instans akan dienkripsi.

Jika penyeimbang beban menggunakan koneksi terenkripsi untuk berkomunikasi dengan instans, Anda dapat mengaktifkan otentikasi instance secara opsional. Ini memastikan bahwa penyeimbang beban berkomunikasi dengan instance hanya jika kunci publiknya cocok dengan kunci yang Anda tentukan ke penyeimbang beban untuk tujuan ini.

Untuk informasi tentang menambahkan pendengar HTTPS ke penyeimbang beban yang ada, lihat. Konfigurasikan listener HTTPS untuk Classic Load Balancer Anda

Daftar Isi

- Prasyarat
- Buat penyeimbang beban HTTPS menggunakan konsol
- Buat penyeimbang beban HTTPS menggunakan AWS CLI

Prasyarat

Sebelum memulai, pastikan Anda telah memenuhi prasyarat berikut:

- Selesaikan langkah-langkah dalam Rekomendasi untuk VPC Anda.
- Luncurkan EC2 contoh yang Anda rencanakan untuk mendaftar dengan penyeimbang beban Anda. Grup keamanan untuk contoh ini harus mengizinkan lalu lintas dari penyeimbang beban.
- EC2 Instance harus menanggapi target pemeriksaan kesehatan dengan kode status HTTP 200. Untuk informasi selengkapnya, lihat Health memeriksa instans Classic Load Balancer Anda.
- Jika Anda berencana untuk mengaktifkan opsi keep-alive pada EC2 instans Anda, kami sarankan Anda menyetel setelan keep-alive ke setidaknya pengaturan batas waktu idle penyeimbang beban Anda. Jika Anda ingin memastikan bahwa penyeimbang beban bertanggung jawab untuk menutup koneksi ke instans Anda, pastikan bahwa nilai yang ditetapkan pada instance Anda untuk waktu

keep-alive lebih besar daripada pengaturan batas waktu idle pada penyeimbang beban Anda. Untuk informasi selengkapnya, lihat Konfigurasikan batas waktu koneksi idle untuk Classic Load Balancer.

- Jika Anda membuat pendengar yang aman, Anda harus menerapkan sertifikat server SSL pada penyeimbang beban Anda. Load balancer menggunakan sertifikat untuk mengakhiri dan kemudian mendekripsi permintaan sebelum mengirimnya ke instance. Jika Anda tidak memiliki sertifikat SSL, Anda dapat membuatnya. Untuk informasi selengkapnya, lihat Sertifikat SSL/TLS untuk Classic Load Balancer.

Buat penyeimbang beban HTTPS menggunakan konsol

Dalam contoh ini, Anda mengonfigurasi dua pendengar untuk penyeimbang beban Anda. Listener pertama menerima permintaan HTTP pada port 80 dan mengirimkannya ke instance pada port 80 menggunakan HTTP. Listener kedua menerima permintaan HTTPS pada port 443 dan mengirimkannya ke instance menggunakan HTTP pada port 80 (atau menggunakan HTTPS pada port 443 jika Anda ingin mengonfigurasi otentikasi instance back-end).

Listener adalah proses memeriksa permintaan koneksi. Ini dikonfigurasi dengan protokol dan port untuk koneksi front-end (client to load balancer) dan protokol dan port untuk koneksi back-end (load balancer to instance). Untuk informasi tentang port, protokol, dan konfigurasi listener yang didukung oleh Elastic Load Balancing, lihat. Pendengar untuk Classic Load Balancer Anda

Untuk membuat Classic Load Balancer aman menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi, pilih Wilayah untuk penyeimbang beban Anda. Pastikan untuk memilih Wilayah yang sama yang Anda pilih untuk EC2 instans Anda.
3. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
4. Pilih Create Load Balancer.
5. Perluas bagian Classic Load Balancer, lalu pilih Buat.
6. Konfigurasi dasar

    a. Untuk nama Load balancer, ketikkan nama untuk penyeimbang beban Anda.

    Nama Classic Load Balancer Anda harus unik dalam set Classic Load Balancer untuk Wilayah, dapat memiliki maksimal 32 karakter, hanya dapat berisi karakter alfanumerik dan tanda hubung, dan tidak boleh dimulai atau diakhiri dengan tanda hubung.

b. Untuk Skema, pilih Menghadap Internet.

7. Pemetaan jaringan

   a. Untuk VPC, pilih VPC yang sama yang Anda pilih untuk instans Anda.

   b. Untuk Pemetaan, pertama-tama pilih Availability Zone, lalu pilih subnet publik dari subnet yang tersedia. Anda hanya dapat memilih satu subnet per Availability Zone. Untuk meningkatkan ketersediaan penyeimbang beban Anda, pilih lebih dari satu Availability Zone dan subnet.

8. Grup keamanan

   • Untuk grup Keamanan, pilih grup keamanan yang ada yang dikonfigurasi untuk memungkinkan lalu lintas HTTP yang diperlukan pada port 80 dan lalu lintas HTTPS pada port 443.

     Jika tidak ada, Anda dapat membuat grup keamanan baru dengan aturan yang diperlukan.

9. Pendengar dan perutean

   a. Biarkan pendengar default dengan pengaturan default, dan pilih Tambahkan pendengar.

   b. Untuk Listener pada listener baru, pilih HTTPS sebagai protokol dan port akan diperbarui ke. 443 Secara default, Instance menggunakan HTTP protokol pada port80.

   c. Jika otentikasi back end diperlukan, ubah protokol Instance menjadiHTTPS. Ini juga akan memperbarui port Instance ke 443

10. Pengaturan pendengar yang aman

    Saat Anda menggunakan HTTPS atau SSL untuk pendengar front-end Anda, Anda harus menerapkan sertifikat SSL pada penyeimbang beban Anda. Load balancer menggunakan sertifikat untuk mengakhiri koneksi dan kemudian mendekripsi permintaan dari klien sebelum mengirimnya ke instance. Anda juga harus menentukan kebijakan keamanan. Elastic Load Balancing menyediakan kebijakan keamanan yang memiliki konfigurasi negosiasi SSL yang telah ditentukan sebelumnya, atau Anda dapat membuat kebijakan keamanan khusus Anda sendiri. Jika Anda mengkonfigurasi HTTPS/SSL pada koneksi back-end, Anda dapat mengaktifkan otentikasi instance Anda.

    a. Untuk kebijakan Keamanan, sebaiknya Anda selalu menggunakan kebijakan keamanan terbaru yang telah ditentukan sebelumnya, atau membuat kebijakan khusus. Lihat Memperbarui Konfigurasi Negosiasi SSL.

    b. Untuk sertifikat SSL/TLS default, opsi berikut tersedia:

- Jika Anda membuat atau mengimpor sertifikat menggunakan AWS Certificate Manager, pilih Dari ACM, lalu pilih sertifikat dari Pilih sertifikat.
- Jika Anda mengimpor sertifikat menggunakan IAM, pilih Dari IAM, lalu pilih sertifikat Anda dari Pilih sertifikat.
- Jika Anda memiliki sertifikat untuk diimpor tetapi ACM tidak tersedia di Wilayah Anda, pilih Impor, lalu pilih Ke IAM. Ketik nama sertifikat di bidang Nama sertifikat. Dalam kunci privat Sertifikat, salin dan tempel isi file kunci pribadi (dikodekan PEM). Di badan Sertifikat, salin dan tempel isi file sertifikat kunci publik (dikodekan PEM). Di Rantai Sertifikat, salin dan tempel isi file rantai sertifikat (dikodekan PEM), kecuali Anda menggunakan sertifikat yang ditandatangani sendiri, dan bukan hal yang penting jika browser secara implisit menerima sertifikat.

c. (Opsional) Jika Anda mengonfigurasi listener HTTPS untuk berkomunikasi dengan instance menggunakan koneksi terenkripsi, Anda dapat secara opsional mengatur otentikasi instance dalam sertifikat autentikasi Backend.

> **Note**
>
> Jika Anda tidak melihat bagian sertifikat otentikasi Backend, kembali ke Pendengar dan perutean dan pilih **HTTPS** sebagai protokol untuk Instance.

i. Untuk nama Sertifikat, ketikkan nama sertifikat kunci publik.

ii. Untuk Badan Sertifikat (PEM dikodekan), salin dan tempel isi sertifikat. Load balancer berkomunikasi dengan instance hanya jika kunci publiknya cocok dengan kunci ini.

iii. Untuk menambahkan sertifikat lain, pilih Tambahkan sertifikat backend baru. Batasnya adalah lima.

11. Pemeriksaan Kesehatan

a. Di bagian target Ping, pilih Protokol Ping dan Port Ping. EC2 Instans Anda harus menerima lalu lintas pada port ping yang ditentukan.

b. Untuk Port Ping, pastikan portnya`80`.

c. Untuk Ping Path, ganti nilai default dengan satu garis miring maju, ()`/`. Ini memberi tahu Elastic Load Balancing untuk mengirim permintaan pemeriksaan kesehatan ke halaman beranda default untuk server web Anda, seperti. `index.html`

d. Untuk pengaturan pemeriksaan kesehatan lanjutan, gunakan nilai default.

12. Contoh

    a. Pilih Tambahkan instance, untuk memunculkan layar pemilihan instance.

    b. Di bawah Instans yang tersedia, Anda dapat memilih dari instans saat ini yang tersedia untuk penyeimbang beban, berdasarkan pengaturan jaringan yang dipilih sebelumnya.

    c. Setelah puas dengan pilihan Anda, pilih Konfirmasi untuk menambahkan instance yang akan didaftarkan ke penyeimbang beban.

13. Atribut

    - Untuk Aktifkan penyeimbangan beban lintas zona, Aktifkan pengeringan koneksi, dan Timeout (interval pengeringan) pertahankan nilai default.

14. Tag penyeimbang beban (opsional)

    a. Bidang kunci diperlukan.

    b. Bidang Nilai adalah opsional.

    c. Untuk menambahkan tag lain, pilih Tambahkan tag baru lalu masukkan nilai Anda ke bidang Kunci, dan opsional bidang Nilai.

    d. Untuk menghapus tag yang ada, pilih Hapus di samping tag yang ingin Anda hapus.

15. Ringkasan dan penciptaan

    a. Jika Anda perlu mengubah pengaturan apa pun, pilih Edit di samping pengaturan yang perlu diubah.

    b. Setelah Anda puas dengan semua pengaturan yang ditampilkan dalam ringkasan, pilih Buat penyeimbang beban untuk memulai pembuatan penyeimbang beban Anda.

    c. Pada halaman pembuatan akhir, pilih Lihat penyeimbang beban untuk melihat penyeimbang beban Anda di konsol Amazon EC2 .

16. Verifikasi

    a. Pilih penyeimbang beban baru Anda.

    b. Pada tab Instance target, periksa kolom Status Kesehatan. Setelah setidaknya satu dari EC2 instans Anda dalam layanan, Anda dapat menguji penyeimbang beban Anda.

    c. Di bagian Detail, salin nama DNS penyeimbang beban, yang akan terlihat mirip dengan. `my-load-balancer-1234567890.us-east-1.elb.amazonaws.com`

d. Rekatkan nama DNS penyeimbang beban Anda ke bidang alamat browser web yang terhubung internet publik. Jika load balancer Anda berfungsi dengan benar, Anda akan melihat halaman default server Anda.

17. Hapus (opsional)

a. Jika Anda memiliki catatan CNAME untuk domain yang mengarah ke penyeimbang beban, arahkan catatan ke lokasi baru dan tunggu hingga perubahan DNS diterapkan sebelum menghapus penyeimbang beban.

b. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.

c. Pilih penyeimbang beban.

d. Pilih Tindakan, Hapus penyeimbang beban.

e. Saat diminta konfirmasi, ketik `confirm` lalu pilih Hapus.

f. Setelah Anda menghapus penyeimbang beban, EC2 instance yang terdaftar dengan penyeimbang beban terus berjalan. Anda akan ditagih untuk setiap jam sebagian atau penuh yang terus mereka jalankan. Ketika Anda tidak lagi membutuhkan EC2 instance, Anda dapat menghentikan atau menghentikannya untuk mencegah timbulnya biaya tambahan.

Buat penyeimbang beban HTTPS menggunakan AWS CLI

Gunakan petunjuk berikut untuk membuat penyeimbang beban HTTPS/SSL menggunakan. AWS CLI

Tugas

- Langkah 1: Konfigurasikan pendengar
- Langkah 2: Konfigurasikan kebijakan keamanan SSL
- Langkah 3: Konfigurasikan otentikasi instance back-end (opsional)
- Langkah 4: Konfigurasikan pemeriksaan kesehatan (opsional)
- Langkah 5: Daftarkan EC2 instance
- Langkah 6: Verifikasi instans
- Langkah 7: Hapus penyeimbang beban (opsional)

Langkah 1: Konfigurasikan pendengar

Listener adalah proses memeriksa permintaan koneksi. Ini dikonfigurasi dengan protokol dan port untuk koneksi front-end (client to load balancer) dan protokol dan port untuk koneksi back-end (load balancer to instance). Untuk informasi tentang port, protokol, dan konfigurasi listener yang didukung oleh Elastic Load Balancing, lihat. Pendengar untuk Classic Load Balancer Anda

Dalam contoh ini, Anda mengonfigurasi dua pendengar untuk penyeimbang beban Anda dengan menentukan port dan protokol yang akan digunakan untuk koneksi front-end dan back-end. Listener pertama menerima permintaan HTTP pada port 80 dan mengirimkan permintaan ke instance pada port 80 menggunakan HTTP. Listener kedua menerima permintaan HTTPS pada port 443 dan mengirimkan permintaan ke instance menggunakan HTTP pada port 80.

Karena pendengar kedua menggunakan HTTPS untuk koneksi front-end, Anda harus menerapkan sertifikat SSL sever pada penyeimbang beban Anda. Load balancer menggunakan sertifikat untuk mengakhiri dan kemudian mendekripsi permintaan sebelum mengirimnya ke instance.

Untuk mengonfigurasi pendengar untuk penyeimbang beban Anda

1. Dapatkan Nama Sumber Daya Amazon (ARN) dari sertifikat SSL. Misalnya:

   ACM

   ```
   arn:aws:acm:region:123456789012:certificate/12345678-1234-1234-1234-123456789012
   ```

   IAM

   ```
   arn:aws:iam::123456789012:server-certificate/my-server-certificate
   ```

2. Gunakan create-load-balancerperintah berikut untuk mengonfigurasi penyeimbang beban dengan dua pendengar:

   ```
   aws elb create-load-balancer --load-balancer-name my-load-balancer --listeners
    "Protocol=http,LoadBalancerPort=80,InstanceProtocol=http,InstancePort=80"
    "Protocol=https,LoadBalancerPort=443,InstanceProtocol=http,InstancePort=80,SSLCertificateI
    --availability-zones us-west-2a
   ```

   Berikut adalah respons contohnya:

   ```
   {
   ```

```
  "DNSName": "my-loadbalancer-012345678.us-west-2.elb.amazonaws.com"
}
```

3. (Opsional) Gunakan describe-load-balancersperintah berikut untuk melihat detail penyeimbang beban Anda:

```
aws elb describe-load-balancers --load-balancer-name my-load-balancer
```

Langkah 2: Konfigurasikan kebijakan keamanan SSL

Anda dapat memilih salah satu kebijakan keamanan yang telah ditentukan sebelumnya, atau Anda dapat membuat kebijakan keamanan khusus Anda sendiri. Jika tidak, Elastic Load Balancing akan mengonfigurasi penyeimbang beban Anda dengan kebijakan keamanan default yang telah ditentukan sebelumnya. `ELBSecurityPolicy-2016-08` Untuk informasi selengkapnya, lihat Konfigurasi negosiasi SSL untuk Classic Load Balancer.

Untuk memverifikasi bahwa penyeimbang beban Anda terkait dengan kebijakan keamanan default

Gunakan perintah describe-load-balancers berikut:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Berikut ini adalah contoh respons. Perhatikan `ELBSecurityPolicy-2016-08` yang terkait dengan penyeimbang beban pada port 443.

```
{
    "LoadBalancerDescriptions": [
        {
            ...
            "ListenerDescriptions": [
                {
                    "Listener": {
                        "InstancePort": 80,
                        "SSLCertificateId": "ARN",
                        "LoadBalancerPort": 443,
                        "Protocol": "HTTPS",
                        "InstanceProtocol": "HTTP"
                    },
                    "PolicyNames": [
                        "ELBSecurityPolicy-2016-08"
```

```
                    ]
                },
                {
                    "Listener": {
                        "InstancePort": 80,
                        "LoadBalancerPort": 80,
                        "Protocol": "HTTP",
                        "InstanceProtocol": "HTTP"
                    },
                    "PolicyNames": []
                }
            ],
            ...
        }
    ]
}
```

Jika mau, Anda dapat mengonfigurasi kebijakan keamanan SSL untuk penyeimbang beban Anda alih-alih menggunakan kebijakan keamanan default.

(Opsional) untuk menggunakan kebijakan keamanan SSL yang telah ditentukan

1. Gunakan describe-load-balancer-policiesperintah berikut untuk mencantumkan nama-nama kebijakan keamanan yang telah ditetapkan:

   ```
   aws elb describe-load-balancer-policies
   ```

   Untuk informasi tentang konfigurasi untuk kebijakan keamanan yang telah ditentukan, lihatKebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer.

2. Gunakan create-load-balancer-policyperintah berikut untuk membuat kebijakan negosiasi SSL menggunakan salah satu kebijakan keamanan yang telah ditentukan sebelumnya yang Anda jelaskan di langkah sebelumnya:

   ```
   aws elb create-load-balancer-policy --load-balancer-name my-loadbalancer
   --policy-name my-SSLNegotiation-policy --policy-type-name SSLNegotiationPolicyType
   --policy-attributes AttributeName=Reference-Security-
   Policy,AttributeValue=predefined-policy
   ```

3. (Opsional) Gunakan describe-load-balancer-policiesperintah berikut untuk memverifikasi bahwa kebijakan dibuat:

```
aws elb describe-load-balancer-policies --load-balancer-name my-loadbalancer --
policy-name my-SSLNegotiation-policy
```

Tanggapan tersebut mencakup deskripsi kebijakan.

4. Gunakan perintah set-load-balancer-policies-of-listener berikut untuk mengaktifkan kebijakan pada port penyeimbang beban 443:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-loadbalancer
 --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

> **Note**
>
> `set-load-balancer-policies-of-listener`Perintah menggantikan kumpulan kebijakan saat ini untuk port penyeimbang beban yang ditentukan dengan kumpulan kebijakan yang ditentukan. `--policy-names`Daftar harus menyertakan semua kebijakan yang akan diaktifkan. Jika Anda menghilangkan kebijakan yang saat ini diaktifkan, kebijakan tersebut akan dinonaktifkan.

5. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan diaktifkan:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Berikut ini adalah contoh respons yang menunjukkan bahwa kebijakan diaktifkan pada port 443.

```
{
    "LoadBalancerDescriptions": [
        {
            ....
            "ListenerDescriptions": [
                {
                    "Listener": {
                        "InstancePort": 80,
                        "SSLCertificateId": "ARN",
                        "LoadBalancerPort": 443,
                        "Protocol": "HTTPS",
                        "InstanceProtocol": "HTTP"
                    },
```

```
                    "PolicyNames": [
                        "my-SSLNegotiation-policy"
                    ]
                },
                {
                    "Listener": {
                        "InstancePort": 80,
                        "LoadBalancerPort": 80,
                        "Protocol": "HTTP",
                        "InstanceProtocol": "HTTP"
                    },
                    "PolicyNames": []
                }
            ],
            ...
        }
    ]
}
```

Saat Anda membuat kebijakan keamanan khusus, Anda harus mengaktifkan setidaknya satu protokol dan satu sandi. Cipher DSA dan RSA khusus untuk algoritma penandatanganan dan digunakan untuk membuat sertifikat SSL. Jika Anda sudah memiliki sertifikat SSL Anda, pastikan untuk mengaktifkan cipher yang digunakan untuk membuat sertifikat Anda. Nama kebijakan kustom Anda tidak boleh dimulai dengan `ELBSecurityPolicy-` atau`ELBSample-`, karena awalan ini dicadangkan untuk nama-nama kebijakan keamanan yang telah ditentukan sebelumnya.

(Opsional) untuk menggunakan kebijakan keamanan SSL khusus

1. Gunakan create-load-balancer-policyperintah untuk membuat kebijakan negosiasi SSL menggunakan kebijakan keamanan khusus. Misalnya:

```
aws elb create-load-balancer-policy --load-balancer-name my-loadbalancer
 --policy-name my-SSLNegotiation-policy --policy-type-name
 SSLNegotiationPolicyType
 --policy-attributes AttributeName=Protocol-TLSv1.2,AttributeValue=true
 AttributeName=Protocol-TLSv1.1,AttributeValue=true
 AttributeName=DHE-RSA-AES256-SHA256,AttributeValue=true
 AttributeName=Server-Defined-Cipher-Order,AttributeValue=true
```

2. (Opsional) Gunakan describe-load-balancer-policiesperintah berikut untuk memverifikasi bahwa kebijakan dibuat:

```
aws elb describe-load-balancer-policies --load-balancer-name my-loadbalancer --
policy-name my-SSLNegotiation-policy
```

Tanggapan tersebut mencakup deskripsi kebijakan.

3. Gunakan perintah set-load-balancer-policies-of-listener berikut untuk mengaktifkan kebijakan pada port penyeimbang beban 443:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-loadbalancer
 --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

> **Note**
>
> `set-load-balancer-policies-of-listener`Perintah menggantikan kumpulan kebijakan saat ini untuk port penyeimbang beban yang ditentukan dengan kumpulan kebijakan yang ditentukan. `--policy-names`Daftar harus menyertakan semua kebijakan yang akan diaktifkan. Jika Anda menghilangkan kebijakan yang saat ini diaktifkan, kebijakan tersebut akan dinonaktifkan.

4. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan diaktifkan:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Berikut ini adalah contoh respons yang menunjukkan bahwa kebijakan diaktifkan pada port 443.

```
{
    "LoadBalancerDescriptions": [
        {
            ....
            "ListenerDescriptions": [
                {
                    "Listener": {
                        "InstancePort": 80,
                        "SSLCertificateId": "ARN",
                        "LoadBalancerPort": 443,
                        "Protocol": "HTTPS",
                        "InstanceProtocol": "HTTP"
                    },
```

```
                    "PolicyNames": [
                        "my-SSLNegotiation-policy"
                    ]
                },
                {
                    "Listener": {
                        "InstancePort": 80,
                        "LoadBalancerPort": 80,
                        "Protocol": "HTTP",
                        "InstanceProtocol": "HTTP"
                    },
                    "PolicyNames": []
                }
            ],
            ...
        }
    ]
}
```

Langkah 3: Konfigurasikan otentikasi instance back-end (opsional)

Jika Anda mengatur HTTPS/SSL pada koneksi back-end, Anda dapat secara opsional mengatur otentikasi instance Anda.

Saat menyiapkan autentikasi instance back-end, Anda membuat kebijakan kunci publik. Selanjutnya, Anda menggunakan kebijakan kunci publik ini untuk membuat kebijakan autentikasi instance back-end. Terakhir, Anda menetapkan kebijakan autentikasi instans back-end dengan port instance untuk protokol HTTPS.

Penyeimbang beban berkomunikasi dengan instance hanya jika kunci publik yang ditampilkan instance ke penyeimbang beban cocok dengan kunci publik dalam kebijakan autentikasi untuk penyeimbang beban Anda.

Untuk mengonfigurasi otentikasi instance back-end

1. Gunakan perintah berikut untuk mengambil kunci publik:

```
openssl x509 -in your X509 certificate PublicKey -pubkey -noout
```

2. Gunakan create-load-balancer-policyperintah berikut untuk membuat kebijakan kunci publik:

```
aws elb create-load-balancer-policy --load-balancer-name my-loadbalancer --policy-
name my-PublicKey-policy \
--policy-type-name PublicKeyPolicyType --policy-attributes
 AttributeName=PublicKey,AttributeValue=MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w
 0BAQUFADCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZ
 WF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIw
 EAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5
 jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBh
 MCVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBb
 WF6b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMx
 HzAdBgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wgZ8wDQYJKoZIhvcNAQE
 BBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ21uUSfwfEvySWtC2XADZ4nB+BLYgVI
 k60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9TrDHudUZg3qX4waLG5M43q7Wgc/MbQ
 ITxOUSQv7c7ugFFDzQGBzZswY6786m86gpEIbb3OhjZnzcvQAaRHhdlQWIMm2nr
 AgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4nUhVVxYUntneD9+h8Mg9q6q+auN
 KyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0FkbFFBjvSfpJIlJ00zbhNYS5f6Guo
 EDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTbNYiytVbZPQUQ5Yaxu2jXnimvw
 3rrszlaEXAMPLE=
```

> **Note**
>
> Untuk menentukan nilai kunci publik`--policy-attributes`, hapus baris pertama dan terakhir dari kunci publik (baris yang berisi "`-----BEGIN PUBLIC KEY-----`" dan baris yang berisi "`-----END PUBLIC KEY-----`“). AWS CLI Tidak menerima karakter spasi putih di`--policy-attributes`.

3. Gunakan create-load-balancer-policyperintah berikut untuk membuat kebijakan otentikasi instance back-end menggunakan. `my-PublicKey-policy`

```
aws elb create-load-balancer-policy --load-balancer-name my-
loadbalancer --policy-name my-authentication-policy --policy-type-
name BackendServerAuthenticationPolicyType --policy-attributes
 AttributeName=PublicKeyPolicyName,AttributeValue=my-PublicKey-policy
```

Anda dapat menggunakan beberapa kebijakan kunci publik secara opsional. Penyeimbang beban mencoba semua kunci, satu per satu. Jika kunci publik yang disajikan oleh sebuah instance cocok dengan salah satu kunci publik ini, instance tersebut diautentikasi.

4. Gunakan for-backend-server perintah berikut set-load-balancer-policies- untuk mengatur `my-authentication-policy` ke port instance untuk HTTPS. Dalam contoh ini, port instance adalah port 443.

```
aws elb set-load-balancer-policies-for-backend-server --load-balancer-name my-loadbalancer --instance-port 443 --policy-names my-authentication-policy
```

5. (Opsional) Gunakan describe-load-balancer-policiesperintah berikut untuk mencantumkan semua kebijakan penyeimbang beban Anda:

```
aws elb describe-load-balancer-policies --load-balancer-name my-loadbalancer
```

6. (Opsional) Gunakan describe-load-balancer-policiesperintah berikut untuk melihat detail kebijakan:

```
aws elb describe-load-balancer-policies --load-balancer-name my-loadbalancer --policy-names my-authentication-policy
```

Langkah 4: Konfigurasikan pemeriksaan kesehatan (opsional)

Elastic Load Balancing secara teratur memeriksa kesehatan setiap EC2 instans terdaftar berdasarkan pemeriksaan kesehatan yang Anda konfigurasi. Jika Elastic Load Balancing menemukan instance yang tidak sehat, Elastic Load Balancing berhenti mengirim lalu lintas ke instans dan mengarahkan lalu lintas ke instans yang sehat. Untuk informasi selengkapnya, lihat Health memeriksa instans Classic Load Balancer Anda.

Saat Anda membuat penyeimbang beban, Elastic Load Balancing menggunakan pengaturan default untuk pemeriksaan kesehatan. Jika mau, Anda dapat mengubah konfigurasi pemeriksaan kesehatan untuk penyeimbang beban Anda alih-alih menggunakan pengaturan default.

Untuk mengonfigurasi pemeriksaan kesehatan untuk instans Anda

Gunakan perintah configure-health-check berikut:

```
aws elb configure-health-check --load-balancer-name my-loadbalancer --health-check
 Target=HTTP:80/ping,Interval=30,UnhealthyThreshold=2,HealthyThreshold=2,Timeout=3
```

Berikut adalah respons contohnya:

```
{
```

```
    "HealthCheck": {
        "HealthyThreshold": 2,
        "Interval": 30,
        "Target": "HTTP:80/ping",
        "Timeout": 3,
        "UnhealthyThreshold": 2
    }
}
```

Langkah 5: Daftarkan EC2 instance

Setelah Anda membuat penyeimbang beban, Anda harus mendaftarkan EC2 instans Anda dengan penyeimbang beban. Anda dapat memilih EC2 instance dari satu Availability Zone atau beberapa Availability Zone dalam Region yang sama dengan load balancer. Untuk informasi selengkapnya, lihat Instans terdaftar untuk Classic Load Balancer Anda.

Gunakan perintah register-instances-with-load-balancer sebagai berikut:

```
aws elb register-instances-with-load-balancer --load-balancer-name my-loadbalancer --
instances i-4f8cf126 i-0bb7ca62
```

Berikut adalah respons contohnya:

```
{
    "Instances": [
        {
            "InstanceId": "i-4f8cf126"
        },
        {
            "InstanceId": "i-0bb7ca62"
        }
    ]
}
```

Langkah 6: Verifikasi instans

Penyeimbang beban Anda dapat digunakan segera setelah salah satu instans terdaftar Anda berada di negara bagian`InService`.

Untuk memeriksa status EC2 instans Anda yang baru terdaftar, gunakan describe-instance-healthperintah berikut:

```
aws elb describe-instance-health  --load-balancer-name my-loadbalancer --
instances i-4f8cf126 i-0bb7ca62
```

Berikut adalah respons contohnya:

```
{
    "InstanceStates": [
        {
            "InstanceId": "i-4f8cf126",
            "ReasonCode": "N/A",
            "State": "InService",
            "Description": "N/A"
        },
        {
            "InstanceId": "i-0bb7ca62",
            "ReasonCode": "Instance",
            "State": "OutOfService",
            "Description": "Instance registration is still in progress"
        }
    ]
}
```

Jika `State` bidang untuk sebuah instance adalah`OutOfService`, itu mungkin karena instance Anda masih mendaftar. Untuk informasi selengkapnya, lihat Memecahkan masalah Classic Load Balancer: Pendaftaran instans.

Setelah status setidaknya satu dari instans Anda`InService`, Anda dapat menguji penyeimbang beban Anda. Untuk menguji penyeimbang beban Anda, salin nama DNS penyeimbang beban dan tempelkan ke bidang alamat browser web yang terhubung ke internet. Jika penyeimbang beban Anda berfungsi, Anda melihat halaman default server HTTP Anda.

Langkah 7: Hapus penyeimbang beban (opsional)

Menghapus penyeimbang beban secara otomatis membatalkan registrasi instance terkait. EC2 Segera setelah penyeimbang beban dihapus, Anda berhenti menimbulkan biaya untuk penyeimbang beban itu. Namun, EC2 instans terus berjalan dan Anda terus dikenakan biaya.

Untuk menghapus penyeimbang beban Anda, gunakan delete-load-balancerperintah berikut:

```
aws elb delete-load-balancer --load-balancer-name my-loadbalancer
```

Untuk menghentikan EC2 instance Anda, gunakan perintah stop-instance. Untuk mengakhiri EC2 instance Anda, gunakan perintah terminate-instance.

Konfigurasikan listener HTTPS untuk Classic Load Balancer Anda

Listener adalah proses memeriksa permintaan koneksi. Ini dikonfigurasi dengan protokol dan port untuk koneksi front-end (client to load balancer) dan protokol dan port untuk koneksi back-end (load balancer to instance). Untuk informasi tentang port, protokol, dan konfigurasi listener yang didukung oleh Elastic Load Balancing, lihat. Pendengar untuk Classic Load Balancer Anda

Jika Anda memiliki penyeimbang beban dengan pendengar yang menerima permintaan HTTP pada port 80, Anda dapat menambahkan pendengar yang menerima permintaan HTTPS pada port 443. Jika Anda menentukan bahwa pendengar HTTPS mengirim permintaan ke instans pada port 80, penyeimbang beban menghentikan permintaan SSL dan komunikasi dari penyeimbang beban ke instans tidak dienkripsi. Jika pendengar HTTPS mengirimkan permintaan ke instance di port 443, komunikasi dari penyeimbang beban ke instans akan dienkripsi.

Jika penyeimbang beban menggunakan koneksi terenkripsi untuk berkomunikasi dengan instans, Anda dapat mengaktifkan otentikasi instance secara opsional. Ini memastikan bahwa penyeimbang beban berkomunikasi dengan instance hanya jika kunci publiknya cocok dengan kunci yang Anda tentukan ke penyeimbang beban untuk tujuan ini.

Untuk informasi tentang membuat pendengar HTTPS baru, lihatMembuat Classic Load Balancer dengan pendengar HTTPS.

Daftar Isi

- Prasyarat
- Tambahkan pendengar HTTPS menggunakan konsol
- Tambahkan pendengar HTTPS menggunakan AWS CLI

Prasyarat

Untuk mengaktifkan dukungan HTTPS untuk pendengar HTTPS, Anda harus menerapkan sertifikat server SSL pada penyeimbang beban Anda. Load balancer menggunakan sertifikat untuk mengakhiri dan kemudian mendekripsi permintaan sebelum mengirimnya ke instance. Jika Anda tidak memiliki sertifikat SSL, Anda dapat membuatnya. Untuk informasi selengkapnya, lihat Sertifikat SSL/TLS untuk Classic Load Balancer.

Tambahkan pendengar HTTPS menggunakan konsol

Anda dapat menambahkan pendengar HTTPS ke penyeimbang beban yang ada.

Untuk menambahkan pendengar HTTPS ke penyeimbang beban Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Listeners, pilih Kelola pendengar.
5. Pada halaman Kelola pendengar, di bagian Pendengar, pilih Tambahkan pendengar.
6. Untuk protokol Listener, pilih HTTPS.

> **Important**
>
> Secara default, protokol Instance adalah HTTP. Jika Anda ingin menyiapkan autentikasi instans back-end, ubah protokol Instance menjadi HTTPS.

7. Untuk kebijakan Keamanan, kami sarankan Anda menggunakan kebijakan keamanan terbaru yang telah ditentukan sebelumnya. Jika Anda perlu menggunakan kebijakan keamanan standar yang berbeda atau membuat kebijakan khusus, lihat Memperbarui Konfigurasi Negosiasi SSL.
8. Untuk sertifikat SSL Default, pilih Edit, lalu lakukan salah satu hal berikut:

   - Jika Anda membuat atau mengimpor sertifikat menggunakan AWS Certificate Manager, pilih Dari ACM, pilih sertifikat dari daftar, lalu pilih Simpan perubahan.

     > **Note**
     >
     > Opsi ini hanya tersedia di Wilayah yang mendukung AWS Certificate Manager.

   - Jika Anda mengimpor sertifikat menggunakan IAM, pilih Dari IAM, pilih sertifikat dari daftar, lalu pilih Simpan perubahan.
   - Jika Anda memiliki sertifikat SSL untuk diimpor ke ACM, pilih Impor dan Ke ACM. Dalam kunci privat Sertifikat, salin dan tempel konten file kunci pribadi yang dikodekan PEM. Di badan Sertifikat, salin dan tempel konten file sertifikat kunci publik yang dikodekan PEM. Dalam rantai Sertifikat - opsional, salin dan tempel konten file rantai sertifikat yang disandikan PEM,

kecuali jika Anda menggunakan sertifikat yang ditandatangani sendiri dan tidak penting bahwa browser secara implisit menerima sertifikat.

- Jika Anda memiliki sertifikat SSL untuk diimpor tetapi ACM tidak didukung di Wilayah ini, pilih Impor dan Ke IAM. Dalam Nama sertifikat ketik nama sertifikat. Dalam kunci privat Sertifikat, salin dan tempel konten file kunci pribadi yang dikodekan PEM. Di badan Sertifikat, salin dan tempel konten file sertifikat kunci publik yang dikodekan PEM. Dalam rantai Sertifikat - opsional, salin dan tempel konten file rantai sertifikat yang disandikan PEM, kecuali jika Anda menggunakan sertifikat yang ditandatangani sendiri dan tidak penting bahwa browser secara implisit menerima sertifikat.
- Pilih Simpan perubahan.

9. Untuk kekakuan Cookie, defaultnya adalah Dinonaktifkan. Untuk mengubah ini pilih Edit. Jika memilih Dihasilkan oleh penyeimbang beban, periode kedaluwarsa harus ditentukan. Jika memilih Dihasilkan oleh aplikasi, nama Cookie harus ditentukan. Setelah membuat pilihan Anda pilih Simpan perubahan.
10. (Opsional) Pilih Tambahkan pendengar untuk menambahkan pendengar tambahan.
11. Pilih Simpan perubahan untuk menambahkan pendengar yang baru saja Anda konfigurasikan.
12. (Opsional) Untuk menyiapkan autentikasi instance back-end untuk penyeimbang beban yang ada, Anda harus menggunakan AWS CLI atau API, karena tugas ini tidak didukung menggunakan konsol. Untuk informasi selengkapnya, lihat Mengkonfigurasi Autentikasi Instans Back-end.

Tambahkan pendengar HTTPS menggunakan AWS CLI

Anda dapat menambahkan pendengar HTTPS ke penyeimbang beban yang ada.

Untuk menambahkan pendengar HTTPS ke penyeimbang beban Anda menggunakan AWS CLI

1. Dapatkan Nama Sumber Daya Amazon (ARN) dari sertifikat SSL. Misalnya:

   ACM

   ```
   arn:aws:acm:region:123456789012:certificate/12345678-1234-1234-1234-123456789012
   ```

   IAM

   ```
   arn:aws:iam::123456789012:server-certificate/my-server-certificate
   ```

2. Gunakan create-load-balancer-listenersperintah berikut untuk menambahkan pendengar ke penyeimbang beban Anda yang menerima permintaan HTTPS pada port 443 dan mengirimkan permintaan ke instance di port 80 menggunakan HTTP:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --
listeners
 Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTP,InstancePort=80,SSLCertificateId
```

Jika Anda ingin menyiapkan autentikasi instans back-end, gunakan perintah berikut untuk menambahkan listener yang menerima permintaan HTTPS pada port 443 dan mengirimkan permintaan ke instance di port 443 menggunakan HTTPS:

```
aws elb create-load-balancer-listeners --load-balancer-name my-load-balancer --
listeners
 Protocol=HTTPS,LoadBalancerPort=443,InstanceProtocol=HTTPS,InstancePort=443,SSLCertificate
```

3. (Opsional) Anda dapat menggunakan describe-load-balancersperintah berikut untuk melihat detail terbaru dari penyeimbang beban Anda:

```
aws elb describe-load-balancers --load-balancer-name my-load-balancer
```

Berikut adalah respons contohnya:

```
{
    "LoadBalancerDescriptions": [
        {
            ...
            "ListenerDescriptions": [
                {
                    "Listener": {
                        "InstancePort": 80,
                        "SSLCertificateId": "ARN",
                        "LoadBalancerPort": 443,
                        "Protocol": "HTTPS",
                        "InstanceProtocol": "HTTP"
                    },
                    "PolicyNames": [
                        "ELBSecurityPolicy-2016-08"
                    ]
                },
                {
```

```
                    "Listener": {
                        "InstancePort": 80,
                        "LoadBalancerPort": 80,
                        "Protocol": "HTTP",
                        "InstanceProtocol": "HTTP"
                    },
                    "PolicyNames": []
                }
            ],
            ...
        }
    ]
}
```

4. (Opsional) Listener HTTPS Anda dibuat menggunakan kebijakan keamanan default. Jika Anda ingin menentukan kebijakan keamanan standar yang berbeda atau kebijakan keamanan khusus, gunakan perintah create-load-balancer-policydan set-load-balancer-policies-of-listener. Untuk informasi selengkapnya, lihat Perbarui konfigurasi negosiasi SSL menggunakan AWS CLI.
5. (Opsional) Untuk mengatur otentikasi instance back-end, gunakan perintah -. set-load-balancer-policies for-backend-server Untuk informasi selengkapnya, lihat Mengkonfigurasi Autentikasi Instans Back-end.

Ganti sertifikat SSL untuk Classic Load Balancer Anda

Jika Anda memiliki pendengar HTTPS, Anda menerapkan sertifikat server SSL pada penyeimbang beban saat membuat listener. Setiap sertifikat memiliki masa berlaku. Anda harus memastikan bahwa Anda memperbarui atau mengganti sertifikat sebelum masa berlakunya berakhir.

Sertifikat yang disediakan oleh AWS Certificate Manager dan digunakan pada penyeimbang beban Anda dapat diperbarui secara otomatis. ACM mencoba untuk memperpanjang sertifikat sebelum masa berlakunya habis. Untuk informasi lebih lanjut, lihat Perpanjangan Terkelola dalam AWS Certificate Manager Panduan Pengguna. Jika Anda mengimpor sertifikat ke ACM, Anda harus memantau tanggal kedaluwarsa sertifikat dan memperpanjang masa berlakunya sebelum kedaluwarsa. Untuk informasi lebih lanjut, lihat Mengimpor sertifikat di AWS Certificate Manager Panduan Pengguna. Setelah sertifikat yang digunakan pada penyeimbang beban diperbarui, permintaan baru menggunakan sertifikat yang diperbarui.

Untuk mengganti sertifikat, Anda harus terlebih dahulu membuat sertifikat baru dengan mengikuti langkah-langkah yang sama yang Anda gunakan saat membuat sertifikat saat ini. Kemudian, Anda

dapat mengganti sertifikat. Setelah sertifikat yang digunakan pada penyeimbang beban diganti, permintaan baru menggunakan sertifikat baru.

Perhatikan bahwa memperbarui atau mengganti sertifikat tidak memengaruhi permintaan yang telah diterima oleh node penyeimbang beban dan sedang menunggu perutean ke target yang sehat.

Daftar Isi

- Ganti sertifikat SSL menggunakan konsol
- Ganti sertifikat SSL menggunakan AWS CLI

Ganti sertifikat SSL menggunakan konsol

Anda dapat mengganti sertifikat yang digunakan pada penyeimbang beban Anda dengan sertifikat yang disediakan oleh ACM atau sertifikat yang diunggah ke IAM.

Untuk mengganti sertifikat SSL untuk penyeimbang beban HTTPS menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Listeners, pilih Kelola pendengar.
5. Pada halaman Kelola pendengar, cari pendengar yang akan diperbarui, pilih Edit di bawah Sertifikat SSL default dan lakukan salah satu hal berikut:

   - Jika Anda membuat atau mengimpor sertifikat menggunakan AWS Certificate Manager, pilih Dari ACM, pilih sertifikat dari daftar, lalu pilih Simpan perubahan.

     > **Note**
     >
     > Opsi ini hanya tersedia di Wilayah yang mendukung AWS Certificate Manager.

   - Jika Anda mengimpor sertifikat menggunakan IAM, pilih Dari IAM, pilih sertifikat dari daftar, lalu pilih Simpan perubahan.
   - Jika Anda memiliki sertifikat SSL untuk diimpor ke ACM, pilih Impor dan Ke ACM. Dalam kunci privat Sertifikat, salin dan tempel konten file kunci pribadi yang dikodekan PEM. Di badan Sertifikat, salin dan tempel konten file sertifikat kunci publik yang dikodekan PEM. Dalam rantai Sertifikat - opsional, salin dan tempel konten file rantai sertifikat yang disandikan PEM,

kecuali jika Anda menggunakan sertifikat yang ditandatangani sendiri dan tidak penting bahwa browser secara implisit menerima sertifikat.

- Jika Anda memiliki sertifikat SSL untuk diimpor tetapi ACM tidak didukung di Wilayah ini, pilih Impor dan Ke IAM. Dalam Nama sertifikat ketik nama sertifikat. Dalam kunci privat Sertifikat, salin dan tempel konten file kunci pribadi yang dikodekan PEM. Di badan Sertifikat, salin dan tempel konten file sertifikat kunci publik yang dikodekan PEM. Dalam rantai Sertifikat - opsional, salin dan tempel konten file rantai sertifikat yang disandikan PEM, kecuali jika Anda menggunakan sertifikat yang ditandatangani sendiri dan tidak penting bahwa browser secara implisit menerima sertifikat.
- Pilih Simpan perubahan.

Ganti sertifikat SSL menggunakan AWS CLI

Anda dapat mengganti sertifikat yang digunakan pada penyeimbang beban Anda dengan sertifikat yang disediakan oleh ACM atau sertifikat yang diunggah ke IAM.

Untuk mengganti sertifikat SSL dengan sertifikat yang disediakan oleh ACM

1. Gunakan perintah request-certificate berikut untuk meminta sertifikat baru:

```
aws acm request-certificate --domain-name www.example.com
```

2. Gunakan perintah set-load-balancer-listener-ssl-certificate berikut untuk mengatur sertifikat:

```
aws elb set-load-balancer-listener-ssl-certificate --load-balancer-
name my-load-balancer --load-balancer-port 443 --ssl-certificate-id
 arn:aws:acm:region:123456789012:certificate/12345678-1234-1234-1234-123456789012
```

Untuk mengganti sertifikat SSL dengan sertifikat yang diunggah ke IAM

1. Jika Anda memiliki sertifikat SSL tetapi belum mengunggahnya, lihat Mengunggah sertifikat server di Panduan Pengguna IAM.
2. Gunakan get-server-certificateperintah berikut untuk mendapatkan ARN sertifikat:

```
aws iam get-server-certificate --server-certificate-name my-new-certificate
```

3. Gunakan perintah set-load-balancer-listener-ssl-certificate berikut untuk mengatur sertifikat:

```
aws elb set-load-balancer-listener-ssl-certificate --load-balancer-
name my-load-balancer --load-balancer-port 443 --ssl-certificate-id
 arn:aws:iam::123456789012:server-certificate/my-new-certificate
```

Perbarui konfigurasi negosiasi SSL Classic Load Balancer Anda

Elastic Load Balancing menyediakan kebijakan keamanan yang memiliki konfigurasi negosiasi SSL yang telah ditentukan sebelumnya untuk digunakan untuk menegosiasikan koneksi SSL antara klien dan penyeimbang beban Anda. Jika Anda menggunakan protokol HTTPS/SSL untuk listener Anda, Anda dapat menggunakan salah satu kebijakan keamanan yang telah ditentukan sebelumnya, atau menggunakan kebijakan keamanan kustom Anda sendiri.

Untuk informasi selengkapnya tentang kebijakan keamanan, lihatKonfigurasi negosiasi SSL untuk Classic Load Balancer. Untuk informasi tentang konfigurasi kebijakan keamanan yang disediakan oleh Elastic Load Balancing, lihat. Kebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer

Jika Anda membuat pendengar HTTPS/SSL tanpa mengaitkan kebijakan keamanan, Elastic Load Balancing mengaitkan kebijakan keamanan standar standar default, dengan penyeimbang beban Anda. `ELBSecurityPolicy-2016-08`

Jika mau, Anda dapat membuat konfigurasi khusus. Kami sangat menyarankan agar Anda menguji kebijakan keamanan sebelum meng-upgrade konfigurasi load balancer Anda.

Contoh berikut menunjukkan cara memperbarui konfigurasi negosiasi SSL untuk pendengar HTTPS/SSL. Perhatikan bahwa perubahan tidak memengaruhi permintaan yang diterima oleh node penyeimbang beban dan sedang menunggu perutean ke instance yang sehat, tetapi konfigurasi yang diperbarui akan digunakan dengan permintaan baru yang diterima.

Daftar Isi

- Perbarui konfigurasi negosiasi SSL menggunakan konsol
- Perbarui konfigurasi negosiasi SSL menggunakan AWS CLI

Perbarui konfigurasi negosiasi SSL menggunakan konsol

Secara default, Elastic Load Balancing mengaitkan kebijakan terbaru yang telah ditentukan sebelumnya dengan penyeimbang beban Anda. Saat kebijakan baru yang telah ditentukan

ditambahkan, sebaiknya Anda memperbarui penyeimbang beban untuk menggunakan kebijakan baru yang telah ditentukan sebelumnya. Atau, Anda dapat memilih kebijakan keamanan yang telah ditentukan sebelumnya atau membuat kebijakan khusus.

Untuk memperbarui konfigurasi negosiasi SSL untuk penyeimbang beban HTTPS/SSL menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Listeners, pilih Kelola pendengar.
5. Pada halaman Kelola listener, cari listener yang akan diperbarui, pilih Edit di bawah Kebijakan keamanan pilih kebijakan keamanan menggunakan salah satu opsi berikut:
   - Simpan kebijakan default, ELBSecurityPolicy-2016-08, lalu pilih Simpan perubahan.
   - Pilih kebijakan yang telah ditentukan sebelumnya selain default, lalu pilih Simpan perubahan.
   - Pilih Custom dan aktifkan setidaknya satu protokol dan satu cipher sebagai berikut:
     a. Untuk Protokol SSL, pilih satu atau beberapa protokol yang akan diaktifkan.
     b. Untuk Opsi SSL, pilih Preferensi Pesanan Server untuk menggunakan urutan yang tercantum dalam negosiasi Kebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer untuk SSL.
     c. Untuk SSL Ciphers, pilih satu atau beberapa cipher untuk mengaktifkan. Jika Anda sudah memiliki sertifikat SSL, Anda harus mengaktifkan cipher yang digunakan untuk membuat sertifikat, karena cipher DSA dan RSA khusus untuk algoritma penandatanganan.
     d. Pilih Simpan perubahan.

Perbarui konfigurasi negosiasi SSL menggunakan AWS CLI

Anda dapat menggunakan kebijakan keamanan standar default`ELBSecurityPolicy-2016-08`, kebijakan keamanan standar yang berbeda, atau kebijakan keamanan khusus.

Untuk menggunakan kebijakan keamanan SSL yang telah ditentukan

1. Gunakan describe-load-balancer-policiesperintah berikut untuk membuat daftar kebijakan keamanan yang telah ditetapkan yang disediakan oleh Elastic Load Balancing. Sintaks yang Anda gunakan tergantung pada sistem operasi dan shell yang Anda gunakan.

   Linux

   ```
   aws elb describe-load-balancer-policies --query 'PolicyDescriptions[?
   PolicyTypeName==`SSLNegotiationPolicyType`].{PolicyName:PolicyName}' --output table
   ```

   Windows

   ```
   aws elb describe-load-balancer-policies --query "PolicyDescriptions[?
   PolicyTypeName==`SSLNegotiationPolicyType`].{PolicyName:PolicyName}" --output table
   ```

   Berikut ini adalah output contoh:

   ```
   ----------------------------------------
   |      DescribeLoadBalancerPolicies      |
   +----------------------------------------+
   |               PolicyName               |
   +----------------------------------------+
   |  ELBSecurityPolicy-2016-08             |
   |  ELBSecurityPolicy-TLS-1-2-2017-01     |
   |  ELBSecurityPolicy-TLS-1-1-2017-01     |
   |  ELBSecurityPolicy-2015-05             |
   |  ELBSecurityPolicy-2015-03             |
   |  ELBSecurityPolicy-2015-02             |
   |  ELBSecurityPolicy-2014-10             |
   |  ELBSecurityPolicy-2014-01             |
   |  ELBSecurityPolicy-2011-08             |
   |  ELBSample-ELBDefaultCipherPolicy      |
   |  ELBSample-OpenSSLDefaultCipherPolicy  |
   +----------------------------------------+
   ```

   Untuk menentukan sandi mana yang diaktifkan untuk kebijakan, gunakan perintah berikut:

   ```
   aws elb describe-load-balancer-policies --policy-names ELBSecurityPolicy-2016-08 --
   output table
   ```

Untuk informasi tentang konfigurasi untuk kebijakan keamanan yang telah ditentukan, lihatKebijakan keamanan SSL yang telah ditentukan sebelumnya untuk Classic Load Balancer.

2. Gunakan create-load-balancer-policyperintah untuk membuat kebijakan negosiasi SSL menggunakan salah satu kebijakan keamanan yang telah ditentukan sebelumnya yang Anda jelaskan di langkah sebelumnya. Misalnya, perintah berikut menggunakan kebijakan keamanan standar standar:

```
aws elb create-load-balancer-policy --load-balancer-name my-loadbalancer
--policy-name my-SSLNegotiation-policy  --policy-type-name SSLNegotiationPolicyType
--policy-attributes AttributeName=Reference-Security-
Policy,AttributeValue=ELBSecurityPolicy-2016-08
```

Jika Anda melebihi batas jumlah kebijakan untuk penyeimbang beban, gunakan delete-load-balancer-policyperintah untuk menghapus kebijakan yang tidak digunakan.

3. (Opsional) Gunakan describe-load-balancer-policiesperintah berikut untuk memverifikasi bahwa kebijakan dibuat:

```
aws elb describe-load-balancer-policies --load-balancer-name my-loadbalancer --
policy-name my-SSLNegotiation-policy
```

Tanggapan tersebut mencakup deskripsi kebijakan.

4. Gunakan perintah set-load-balancer-policies-of-listener berikut untuk mengaktifkan kebijakan pada port penyeimbang beban 443:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-loadbalancer
 --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

> **Note**
>
> `set-load-balancer-policies-of-listener`Perintah menggantikan kumpulan kebijakan saat ini untuk port penyeimbang beban yang ditentukan dengan kumpulan kebijakan yang ditentukan. `--policy-names`Daftar harus menyertakan semua kebijakan yang akan diaktifkan. Jika Anda menghilangkan kebijakan yang saat ini diaktifkan, kebijakan tersebut akan dinonaktifkan.

5. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan baru diaktifkan untuk port penyeimbang beban:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Tanggapan menunjukkan bahwa kebijakan diaktifkan pada port 443.

```
...
  {
      "Listener": {
          "InstancePort": 443,
          "SSLCertificateId": "ARN",
          "LoadBalancerPort": 443,
          "Protocol": "HTTPS",
          "InstanceProtocol": "HTTPS"
      },
      "PolicyNames": [
          "my-SSLNegotiation-policy"
      ]
  }
...
```

Saat Anda membuat kebijakan keamanan khusus, Anda harus mengaktifkan setidaknya satu protokol dan satu sandi. Cipher DSA dan RSA khusus untuk algoritma penandatanganan dan digunakan untuk membuat sertifikat SSL. Jika Anda sudah memiliki sertifikat SSL, pastikan untuk mengaktifkan cipher yang digunakan untuk membuat sertifikat. Nama kebijakan kustom Anda tidak boleh dimulai dengan `ELBSecurityPolicy-` atau`ELBSample-`, karena awalan ini dicadangkan untuk nama-nama kebijakan keamanan yang telah ditentukan sebelumnya.

Untuk menggunakan kebijakan keamanan SSL khusus

1. Gunakan create-load-balancer-policyperintah untuk membuat kebijakan negosiasi SSL menggunakan kebijakan keamanan khusus. Misalnya:

```
aws elb create-load-balancer-policy --load-balancer-name my-loadbalancer
 --policy-name my-SSLNegotiation-policy --policy-type-name
 SSLNegotiationPolicyType
 --policy-attributes AttributeName=Protocol-TLSv1.2,AttributeValue=true
 AttributeName=Protocol-TLSv1.1,AttributeValue=true
 AttributeName=DHE-RSA-AES256-SHA256,AttributeValue=true
```

```
AttributeName=Server-Defined-Cipher-Order,AttributeValue=true
```

Jika Anda melebihi batas jumlah kebijakan untuk penyeimbang beban, gunakan delete-load-balancer-policyperintah untuk menghapus kebijakan yang tidak digunakan.

2. (Opsional) Gunakan describe-load-balancer-policiesperintah berikut untuk memverifikasi bahwa kebijakan dibuat:

```
aws elb describe-load-balancer-policies --load-balancer-name my-loadbalancer --
policy-name my-SSLNegotiation-policy
```

Tanggapan tersebut mencakup deskripsi kebijakan.

3. Gunakan perintah set-load-balancer-policies-of-listener berikut untuk mengaktifkan kebijakan pada port penyeimbang beban 443:

```
aws elb set-load-balancer-policies-of-listener --load-balancer-name my-loadbalancer
 --load-balancer-port 443 --policy-names my-SSLNegotiation-policy
```

> **Note**
>
> `set-load-balancer-policies-of-listener`Perintah menggantikan kumpulan kebijakan saat ini untuk port penyeimbang beban yang ditentukan dengan kumpulan kebijakan yang ditentukan. `--policy-names`Daftar harus menyertakan semua kebijakan yang akan diaktifkan. Jika Anda menghilangkan kebijakan yang saat ini diaktifkan, kebijakan tersebut akan dinonaktifkan.

4. (Opsional) Gunakan describe-load-balancersperintah berikut untuk memverifikasi bahwa kebijakan baru diaktifkan untuk port penyeimbang beban:

```
aws elb describe-load-balancers --load-balancer-name my-loadbalancer
```

Tanggapan menunjukkan bahwa kebijakan diaktifkan pada port 443.

```
...
  {
      "Listener": {
          "InstancePort": 443,
          "SSLCertificateId": "ARN",
          "LoadBalancerPort": 443,
```

```
            "Protocol": "HTTPS",
            "InstanceProtocol": "HTTPS"
        },
        "PolicyNames": [
            "my-SSLNegotiation-policy"
        ]
    }
...
```

Instans terdaftar untuk Classic Load Balancer Anda

Setelah membuat Classic Load Balancer, Anda harus mendaftarkan EC2 instans Anda dengan load balancer. Anda dapat memilih EC2 instance dari satu Availability Zone atau beberapa Availability Zone dalam Region yang sama dengan load balancer. Elastic Load Balancing secara rutin melakukan pemeriksaan kesehatan pada EC2 instans terdaftar, dan secara otomatis mendistribusikan permintaan masuk ke nama DNS penyeimbang beban Anda di seluruh instans yang terdaftar dan sehat. EC2

Daftar Isi

- Praktik terbaik untuk instans Anda
- Rekomendasi untuk VPC Anda
- Daftarkan instans dengan Classic Load Balancer Anda
- Health memeriksa instans Classic Load Balancer Anda
- Grup keamanan untuk instans Classic Load Balancer Anda
- Jaringan ACLs untuk instans Classic Load Balancer Anda

Praktik terbaik untuk instans Anda

- Anda harus memastikan bahwa penyeimbang beban dapat berkomunikasi dengan instans Anda di port pendengar dan port pemeriksaan kesehatan. Untuk informasi selengkapnya, lihat Konfigurasikan grup keamanan untuk Classic Load Balancer Anda. Grup keamanan untuk instans Anda harus mengizinkan lalu lintas di kedua arah pada kedua port untuk setiap subnet untuk penyeimbang beban Anda.
- Instal server web, seperti Apache atau Internet Information Services (IIS), pada semua contoh yang Anda rencanakan untuk mendaftar dengan penyeimbang beban Anda.
- Untuk pendengar HTTP dan HTTPS, sebaiknya aktifkan opsi keep-alive dalam instans Anda, yang memungkinkan penyeimbang beban menggunakan kembali koneksi ke EC2 instans Anda untuk beberapa permintaan klien. Ini mengurangi beban pada server web Anda dan meningkatkan throughput penyeimbang beban. Batas waktu keep-alive harus setidaknya 60 detik untuk memastikan bahwa penyeimbang beban bertanggung jawab untuk menutup koneksi ke instans Anda.
- Elastic Load Balancing mendukung Penemuan Path Maximum Transmission Unit (MTU). Untuk memastikan bahwa Path MTU Discovery dapat berfungsi dengan benar, Anda harus memastikan

bahwa grup keamanan untuk instans Anda mengizinkan fragmentasi ICMP yang diperlukan (tipe 3, kode 4) pesan. Untuk informasi selengkapnya, lihat Path MTU Discovery di Panduan EC2 Pengguna Amazon.

Rekomendasi untuk VPC Anda

Cloud privat virtual (VPC)

Kecuali Anda membuat Akun AWS sebelum 2014, Anda memiliki VPC default di setiap Wilayah. Anda dapat menggunakan VPC default untuk penyeimbang beban Anda, jika Anda memilikinya, atau Anda dapat membuat VPC baru. Untuk informasi selengkapnya, silakan lihat ACL Jaringan di Panduan Pengguna Amazon VPC.

Subnet untuk penyeimbang beban Anda

Untuk memastikan bahwa penyeimbang beban Anda dapat menskalakan dengan benar, verifikasi bahwa setiap subnet untuk penyeimbang beban Anda memiliki blok CIDR dengan setidaknya `/27` bitmask (misalnya,`10.0.0.0/27`) dan memiliki setidaknya 8 alamat IP gratis. Load balancer Anda menggunakan alamat IP ini untuk membuat koneksi dengan instans, dan untuk memperkecil skala bila diperlukan. Jika alamat IP tidak mencukupi, penyeimbang beban mungkin tidak dapat diskalakan, menyebabkan 503 kesalahan karena kapasitas yang tidak mencukupi.

Buat subnet di setiap Availability Zone tempat Anda ingin meluncurkan instance. Bergantung pada aplikasi Anda, Anda dapat meluncurkan instance Anda di subnet publik, subnet pribadi, atau kombinasi subnet publik dan pribadi. Subnet publik memiliki rute ke gateway internet. Perhatikan bahwa default VPCs memiliki satu subnet publik per Availability Zone secara default.

Saat Anda membuat penyeimbang beban, Anda harus menambahkan satu atau lebih subnet publik ke penyeimbang beban. Jika instans Anda berada dalam subnet pribadi, buat subnet publik di Availability Zone yang sama dengan subnet dengan instans Anda; Anda akan menambahkan subnet publik ini ke penyeimbang beban.

Jaringan ACLs

Jaringan ACLs untuk VPC Anda harus mengizinkan lalu lintas di kedua arah pada port pendengar dan port pemeriksaan kesehatan. Untuk informasi selengkapnya, lihat Jaringan ACLs untuk instans Classic Load Balancer Anda.

Daftarkan instans dengan Classic Load Balancer Anda

Mendaftarkan EC2 instance menambahkannya ke penyeimbang beban Anda. Penyeimbang beban terus memantau kesehatan instans terdaftar di Availability Zone yang diaktifkan, dan mengarahkan permintaan ke instans yang sehat. Jika permintaan pada instans Anda meningkat, Anda dapat mendaftarkan instans tambahan dengan penyeimbang beban untuk menangani permintaan.

Membatalkan pendaftaran EC2 instance menghapusnya dari penyeimbang beban Anda. Penyeimbang beban berhenti merutekan permintaan ke sebuah instance segera setelah dideregistrasi. Jika permintaan menurun, atau Anda perlu memperbaiki instans Anda, Anda dapat membatalkan pendaftaran instans dari penyeimbang beban. Instance yang dideregistrasi tetap berjalan, tetapi tidak lagi menerima lalu lintas dari penyeimbang beban, dan Anda dapat mendaftarkannya dengan penyeimbang beban lagi saat Anda siap.

Saat Anda membatalkan pendaftaran instans, Elastic Load Balancing menunggu hingga permintaan dalam penerbangan selesai jika pengurasan koneksi diaktifkan. Untuk informasi selengkapnya, lihat Konfigurasikan pengurasan koneksi untuk Classic Load Balancer Anda.

Jika penyeimbang beban Anda dilampirkan ke grup Auto Scaling, instance dalam grup secara otomatis terdaftar dengan penyeimbang beban. Jika Anda melepaskan penyeimbang beban dari grup Auto Scaling, instance dalam grup akan dideregistrasi.

Elastic Load Balancing mendaftarkan EC2 instans Anda dengan load balancer menggunakan alamat IP-nya.

[EC2-VPC] Saat Anda mendaftarkan instance dengan elastic network interface (ENI) yang terpasang, load balancer merutekan permintaan ke alamat IP utama antarmuka utama (eth0) instance.

Daftar Isi

- Daftarkan sebuah instance
- Melihat instans yang terdaftar dengan penyeimbang beban
- Tentukan penyeimbang beban untuk instance terdaftar
- Deregister sebuah instance

Daftarkan sebuah instance

Ketika Anda siap, daftarkan instans Anda dengan penyeimbang beban Anda. Jika instance berada di Availability Zone yang diaktifkan untuk penyeimbang beban, instance siap menerima lalu lintas dari penyeimbang beban segera setelah melewati jumlah pemeriksaan kesehatan yang diperlukan.

Untuk mendaftarkan instans Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Instance target, pilih Kelola instance.
5. Pada halaman Kelola instans, dalam tabel Instance yang tersedia, pilih instance yang akan didaftarkan dengan penyeimbang beban Anda.
6. Pastikan instance yang perlu didaftarkan diisi dalam tabel Tinjau instance yang dipilih.
7. Pilih Simpan perubahan.

Untuk mendaftarkan instans Anda menggunakan AWS CLI

Gunakan perintah register-instances-with-load-balancer berikut:

```
aws elb register-instances-with-load-balancer --load-balancer-name my-loadbalancer --instances i-4e05f721
```

Berikut ini adalah contoh respons yang mencantumkan instance yang terdaftar dengan penyeimbang beban:

```
{
    "Instances": [
        {
            "InstanceId": "i-315b7e51"
        },
        {
            "InstanceId": "i-4e05f721"
        }
    ]
}
```

Melihat instans yang terdaftar dengan penyeimbang beban

Gunakan describe-load-balancersperintah berikut untuk membuat daftar instance yang terdaftar dengan penyeimbang beban yang ditentukan:

```
aws elb describe-load-balancers --load-balancer-names my-load-balancer --output text --
query "LoadBalancerDescriptions[*].Instances[*].InstanceId"
```

Berikut ini adalah output contoh:

```
i-e905622e
i-315b7e51
i-4e05f721
```

Tentukan penyeimbang beban untuk instance terdaftar

Gunakan describe-load-balancersperintah berikut untuk mendapatkan nama penyeimbang beban tempat instance yang ditentukan terdaftar:

```
aws elb describe-load-balancers --output text --query "LoadBalancerDescriptions[?
Instances[?InstanceId=='i-e905622e']].[LoadBalancerName]"
```

Berikut ini adalah output contoh:

```
my-load-balancer
```

Deregister sebuah instance

Anda dapat membatalkan pendaftaran instance dari penyeimbang beban Anda jika Anda tidak lagi membutuhkan kapasitas atau jika Anda perlu melayani instans.

Jika penyeimbang beban Anda dilampirkan ke grup Auto Scaling, melepaskan instance dari grup juga membatalkan pendaftarannya dari penyeimbang beban. Untuk informasi selengkapnya, lihat Melepaskan EC2 instans dari grup Auto Scaling di Panduan Pengguna Amazon Auto EC2 Scaling.

Untuk membatalkan pendaftaran instans Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.

3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Instance target, pilih Kelola instance.
5. Pada halaman Kelola instans, dalam tabel Instance yang tersedia, batalkan pilihan instans yang akan dideregister dari penyeimbang beban Anda.
6. Pastikan instance yang perlu dideregistrasi tidak diisi dalam tabel contoh yang dipilih Tinjauan.
7. Pilih Simpan perubahan.

Untuk membatalkan pendaftaran instans Anda menggunakan AWS CLI

Gunakan perintah deregister-instances-from-load-balancer berikut:

```
aws elb deregister-instances-from-load-balancer --load-balancer-name my-loadbalancer --instances i-4e05f721
```

Berikut ini adalah contoh respons yang mencantumkan instance yang tersisa yang terdaftar dengan penyeimbang beban:

```
{
    "Instances": [
        {
            "InstanceId": "i-315b7e51"
        }
    ]
}
```

Health memeriksa instans Classic Load Balancer Anda

Classic Load Balancer Anda secara berkala mengirimkan permintaan ke instans terdaftarnya untuk menguji statusnya. Uji ini disebut pemeriksaan kondisi. Status kasus yang sehat pada saat pemeriksaan kesehatan adalah`InService`. Status setiap kasus yang tidak sehat pada saat pemeriksaan kesehatan adalah`OutOfService`. Penyeimbang beban melakukan pemeriksaan kesehatan pada semua instance yang terdaftar, apakah instans dalam keadaan sehat atau tidak sehat.

Rute penyeimbang beban hanya meminta ke instans sehat. Ketika penyeimbang beban menentukan bahwa sebuah instance tidak sehat, itu menghentikan permintaan perutean ke instance itu. Penyeimbang beban melanjutkan permintaan perutean ke instance ketika telah dikembalikan ke keadaan sehat.

Load balancer memeriksa kesehatan instans terdaftar menggunakan konfigurasi pemeriksaan kesehatan default yang disediakan oleh Elastic Load Balancing atau konfigurasi pemeriksaan kesehatan yang Anda konfigurasikan.

Jika Anda telah mengaitkan grup Auto Scaling Anda dengan Classic Load Balancer, Anda dapat menggunakan pemeriksaan kesehatan load balancer untuk menentukan kondisi kesehatan instans di grup Auto Scaling Anda. Secara default, grup Auto Scaling secara berkala menentukan status kesehatan setiap instance. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan Menambahkan Elastic Load Balancing ke grup Auto Scaling di Panduan Pengguna Amazon Auto EC2 Scaling.

Daftar Isi

- Konfigurasi pemeriksaan kesehatan
- Perbarui konfigurasi pemeriksaan kesehatan
- Periksa kesehatan instans Anda
- Memecahkan masalah pemeriksaan kesehatan

Konfigurasi pemeriksaan kesehatan

Konfigurasi kesehatan berisi informasi yang digunakan penyeimbang beban untuk menentukan kondisi kesehatan dari instans yang terdaftar. Tabel berikut menjelaskan bidang konfigurasi pemeriksaan kesehatan.

Bidang	Deskripsi
Protokol	Protokol yang digunakan untuk terhubung dengan instance. Nilai yang valid:`TCP`,`HTTP`,`HTTPS`, dan `SSL` Konsol default: `HTTP` Default CLI/API: `TCP`
Port	Port yang digunakan untuk terhubung dengan instance, sebagai `protocol:port` pasangan. Jika penyeimba

Bidang	Deskripsi
	ng beban gagal terhubung dengan instance di port yang ditentukan dalam periode batas waktu respons yang dikonfigurasi, instance dianggap tidak sehat. Protokol:TCP,,HTTP, dan HTTPS SSL Rentang port: 1 hingga 65535 Konsol default: `HTTP:80` Default CLI/API: `TCP:80`
Jalur	Tujuan untuk permintaan HTTP atau HTTPS. Permintaan HTTP atau HTTPS GET dikeluarkan untuk instance pada port dan path. Jika penyeimbang beban menerima respons apa pun selain “200 OK” dalam periode batas waktu respons, instance dianggap tidak sehat. Jika respons menyertakan isi, aplikasi Anda harus menyetel header Content-Length ke nilai yang lebih besar dari atau sama dengan nol, atau menentukan Transfer-Encoding dengan nilai yang disetel ke 'chunked'. Default: `/index.html`
Waktu Respons Habis	Jumlah waktu untuk menunggu ketika menerima respons dari pemeriksaan kesehatan, dalam hitungan detik. Nilai yang valid: 2 hingga 60 Default: 5

Bidang	Deskripsi
HealthCheck Interval	Jumlah waktu antara pemeriksaan kesehatan dari contoh individu, dalam hitungan detik. Nilai yang valid: 5 hingga 300 Bawaan: 30
Batas Kondisi Tidak Baik	Jumlah pemeriksaan kesehatan gagal berturut-turut yang harus dilakukan sebelum menyatakan contoh EC2 tidak sehat. Nilai yang valid: 2 hingga 10 Default: 2
Ambang Sehat	Jumlah pemeriksaan kesehatan yang berhasil berturut-turut yang harus dilakukan sebelum menyatakan contoh EC2 sehat. Nilai yang valid: 2 hingga 10 Default: 10

Penyeimbang beban mengirimkan permintaan pemeriksaan kesehatan ke setiap instance terdaftar setiap `Interval` detik, menggunakan port, protokol, dan jalur yang ditentukan. Setiap permintaan pemeriksaan kesehatan bersifat independen dan berlangsung sepanjang interval. Waktu yang dibutuhkan untuk merespons tidak mempengaruhi interval untuk pemeriksaan kesehatan berikutnya. Jika pemeriksaan kesehatan melebihi kegagalan UnhealthyThresholdCountberturut-turut, penyeimbang beban mengambil contoh keluar dari layanan. Ketika pemeriksaan kesehatan melebihi keberhasilan HealthyThresholdCountberturut-turut, penyeimbang beban menempatkan instance kembali dalam layanan.

Pemeriksaan kesehatan HTTP/HTTPS berhasil jika instance mengembalikan kode respons 200 dalam interval pemeriksaan kesehatan. Pemeriksaan kesehatan TCP berhasil jika koneksi TCP berhasil. Pemeriksaan kesehatan SSL berhasil jika jabat tangan SSL berhasil.

Perbarui konfigurasi pemeriksaan kesehatan

Anda dapat memperbarui konfigurasi pemeriksaan kesehatan untuk penyeimbang beban Anda kapan saja.

Untuk memperbarui konfigurasi pemeriksaan kesehatan untuk penyeimbang beban Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Pada tab Pemeriksaan kondisi, pilih Edit.
5. Pada halaman Edit pengaturan pemeriksaan kesehatan, di bawah Pemeriksaan Kesehatan, perbarui konfigurasi sesuai kebutuhan.
6. Setelah puas dengan pilihan Anda, pilih Simpan perubahan.

Untuk memperbarui konfigurasi pemeriksaan kesehatan untuk penyeimbang beban Anda menggunakan AWS CLI

Gunakan perintah configure-health-check berikut:

```
aws elb configure-health-check --load-balancer-name my-load-balancer --health-check
 Target=HTTP:80/path,Interval=30,UnhealthyThreshold=2,HealthyThreshold=2,Timeout=3
```

Periksa kesehatan instans Anda

Anda dapat memeriksa status kesehatan instans terdaftar Anda.

Untuk memeriksa status kesehatan instans Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.
4. Di bagian Detail, Status menunjukkan berapa banyak instance yang ada dalam layanan.
5. Pada tab Instance target, di dalam tabel instans Target, kolom Status Kesehatan menunjukkan status spesifik dari setiap instans terdaftar.

Untuk memeriksa status kesehatan instans Anda menggunakan AWS CLI

Gunakan perintah describe-instance-health berikut:

```
aws elb describe-instance-health --load-balancer-name my-load-balancer
```

Memecahkan masalah pemeriksaan kesehatan

Instans terdaftar Anda dapat gagal dalam pemeriksaan kesehatan penyeimbang beban karena beberapa alasan. Alasan paling umum untuk gagal pemeriksaan kesehatan adalah di mana EC2 instance menutup koneksi ke penyeimbang beban Anda atau di mana respons dari EC2 instance habis waktu. Untuk informasi tentang penyebab potensial dan langkah-langkah yang dapat Anda ambil untuk menyelesaikan masalah pemeriksaan kesehatan yang gagal, lihatMemecahkan Masalah Classic Load Balancer: Pemeriksaan Kesehatan.

Grup keamanan untuk instans Classic Load Balancer Anda

Grup keamanan bertindak sebagai firewall yang mengontrol lalu lintas yang diizinkan ke dan dari satu atau lebih contoh. Saat meluncurkan EC2 instance, Anda dapat mengaitkan satu atau beberapa grup keamanan dengan instans tersebut. Untuk setiap grup keamanan, Anda menambahkan satu atau beberapa aturan untuk mengizinkan lalu lintas. Anda dapat mengubah aturan untuk grup keamanan kapan saja; aturan baru diterapkan secara otomatis ke semua instance yang terkait dengan grup keamanan. Untuk informasi selengkapnya, lihat Grup EC2 keamanan Amazon di Panduan EC2 Pengguna Amazon.

Grup keamanan untuk instans Anda harus memungkinkan mereka berkomunikasi dengan penyeimbang beban. Tabel berikut menunjukkan aturan masuk yang direkomendasikan.

Sumber	Protokol	Baris Port	Komentar
load balancer security group	TCP	*instance listener*	Izinkan lalu lintas dari penyeimbang beban pada port pendengar instance
load balancer security group	TCP	*health check*	Izinkan lalu lintas dari penyeimbang beban di port pemeriksaan kesehatan

Kami juga merekomendasikan Anda untuk mengizinkan inbound ICMP lalu lintas untuk mendukung jalan MTU penemuan. Untuk informasi selengkapnya, lihat Path MTU Discovery di Panduan EC2 Pengguna Amazon.

Jaringan ACLs untuk instans Classic Load Balancer Anda

Daftar kontrol akses jaringan (ACL) memungkinkan atau menolak lalu lintas masuk atau keluar tertentu di tingkat subnet. Anda dapat menggunakan ACL jaringan default untuk VPC Anda, atau Anda dapat membuat ACL jaringan khusus untuk VPC Anda dengan aturan yang mirip dengan aturan untuk grup keamanan Anda untuk menambahkan lapisan keamanan tambahan ke VPC Anda.

Daftar kontrol akses jaringan default (ACL) untuk VPC memungkinkan semua lalu lintas masuk dan keluar. Jika Anda membuat jaringan kustom ACLs, Anda harus menambahkan aturan yang memungkinkan penyeimbang beban dan instance untuk berkomunikasi.

Aturan yang disarankan untuk subnet untuk instans Anda tergantung pada apakah subnet itu pribadi atau publik. Aturan berikut adalah untuk subnet pribadi. Jika instans Anda berada di subnet publik, ubah sumber dan tujuan dari CIDR VPC menjadi. `0.0.0.0/0`

Berikut ini adalah aturan masuk yang direkomendasikan.

Sumber	Protokol	Baris Port	Komentar
VPC CIDR	TCP	*instance listener*	Izinkan lalu lintas masuk dari CIDR VPC pada port pendengar instance
VPC CIDR	TCP	*health check*	Izinkan lalu lintas masuk dari CIDR VPC di port pemeriksaan kesehatan

Berikut ini adalah aturan keluar yang direkomendasikan.

Tujuan	Protokol	Baris Port	Komentar
VPC CIDR	TCP	1024-65535	Izinkan lalu lintas keluar ke CIDR VPC di port fana

Pantau Classic Load Balancer Anda

Anda dapat menggunakan fitur-fitur berikut untuk memantau penyeimbang beban, menganalisis pola lalu lintas, dan memecahkan masalah dengan penyeimbang beban dan instans back-end Anda.

CloudWatch metrik

Elastic Load Balancing menerbitkan titik data ke Amazon CloudWatch tentang load balancer dan instans back-end Anda. CloudWatch memungkinkan Anda untuk mengambil statistik tentang titik-titik data tersebut sebagai kumpulan data deret waktu yang diurutkan, yang dikenal sebagai metrik. Anda dapat menggunakan metrik ini untuk memverifikasi bahwa sistem Anda bekerja sesuai harapan. Untuk informasi selengkapnya, lihat CloudWatch metrik untuk Classic Load Balancer Anda.

Log akses Elastic Load Balancing

Log akses untuk Elastic Load Balancing menangkap informasi terperinci untuk permintaan yang dibuat ke penyeimbang beban Anda dan menyimpannya sebagai file log di bucket Amazon S3 yang Anda tentukan. Setiap log berisi detail seperti waktu permintaan diterima, alamat IP klien, latensi, jalur permintaan, dan respons server. Anda dapat menggunakan log akses ini untuk menganalisis pola lalu lintas dan untuk memecahkan masalah aplikasi back-end Anda. Untuk informasi selengkapnya, lihat Akses log untuk Classic Load Balancer Anda.

CloudTrail log

AWS CloudTrail memungkinkan Anda untuk melacak panggilan yang dilakukan ke Elastic Load Balancing API oleh atau atas nama akun Anda AWS . CloudTrail menyimpan informasi dalam file log di bucket Amazon S3 yang Anda tentukan. Anda dapat menggunakan file log ini untuk memantau aktivitas penyeimbang beban Anda dengan menentukan permintaan mana yang dibuat, alamat IP sumber dari mana permintaan berasal, siapa yang membuat permintaan, kapan permintaan dibuat, dan sebagainya. Untuk informasi selengkapnya, lihat Log panggilan API untuk Elastic Load Balancing menggunakan. CloudTrail

CloudWatch metrik untuk Classic Load Balancer Anda

Elastic Load Balancing menerbitkan titik data ke Amazon CloudWatch untuk penyeimbang beban dan instans back-end Anda. CloudWatch memungkinkan Anda untuk mengambil statistik tentang titik-titik data tersebut sebagai kumpulan data deret waktu yang diurutkan, yang dikenal sebagai metrik.

Anggap metrik sebagai variabel untuk memantau, dan titik data sebagai nilai variabel tersebut dari waktu ke waktu. Misalnya, Anda dapat memantau jumlah total EC2 instans sehat untuk penyeimbang beban selama periode waktu tertentu. Setiap titik data memiliki stempel waktu terkait dan unit pengukuran opsional.

Anda dapat menggunakan metrik untuk memverifikasi bahwa sistem Anda bekerja sesuai harapan. Misalnya, Anda dapat membuat CloudWatch alarm untuk memantau metrik tertentu dan memulai tindakan (seperti mengirim pemberitahuan ke alamat email) jika metrik berada di luar rentang yang Anda anggap dapat diterima.

Elastic Load Balancing melaporkan metrik CloudWatch hanya ketika permintaan mengalir melalui penyeimbang beban. Jika ada permintaan yang mengalir melalui penyeimbang beban, Elastic Load Balancing mengukur dan mengirimkan metriknya dalam interval 60 detik. Jika tidak ada permintaan yang mengalir melalui penyeimbang beban atau tidak ada data untuk metrik, metrik tidak dilaporkan.

Untuk informasi selengkapnya tentang Amazon CloudWatch, lihat Panduan CloudWatch Pengguna Amazon.

Daftar Isi

- Metrik Classic Load Balancer
- Dimensi metrik untuk Classic Load Balancer
- Statistik untuk metrik Classic Load Balancer
- Lihat CloudWatch metrik untuk penyeimbang beban Anda

Metrik Classic Load Balancer

Namespace `AWS/ELB` mencakup metrik berikut.

Metrik	Deskripsi
`BackendConnectionErrors`	Jumlah koneksi yang tidak berhasil dibuat antara penyeimbang beban dan instans terdaftar. Karena penyeimbang beban mencoba ulang koneksi ketika ada kesalahan, jumlah ini dapat melebihi tingkat permintaan. Perhatikan bahwa hitungan ini juga mencakup kesalahan koneksi yang terkait dengan pemeriksaan kesehatan. Reporting criteria: Ada nilai bukan nol

Metrik	Deskripsi
	Statistics: Statistik yang paling berguna adalah `Sum`. Perhatikan bahwa `Average`, `Minimum`, dan `Maximum` dilaporkan per node penyeimbang beban dan biasanya tidak berguna. Namun, perbedaan antara minimum dan maksimum (atau puncak ke rata-rata atau rata-rata ke palung) mungkin berguna untuk menentukan apakah node penyeimbang beban adalah outlier. Contoh: Misalkan penyeimbang beban Anda memiliki 2 instance di us-west-2a dan 2 instance di us-west-2b, dan upaya untuk terhubung ke 1 instance di us-west-2a menghasilkan kesalahan koneksi back-end. Jumlah untuk us-west-2a mencakup kesalahan koneksi ini, sedangkan jumlah untuk us-west-2b tidak menyertakannya. Oleh karena itu, jumlah untuk penyeimbang beban sama dengan jumlah untuk us-barat-2a.
`DesyncMitigationMode_NonCompliant_Request_Count`	[HTTP listener] Jumlah permintaan yang tidak sesuai dengan RFC 7230. Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah `Sum`.

Metrik	Deskripsi
`HealthyHostCount`	Jumlah instans sehat yang terdaftar di penyeimbang beban Anda. Contoh yang baru terdaftar dianggap sehat setelah melewati pemeriksaan kesehatan pertama. Jika penyeimbangan beban lintas zona diaktifkan, jumlah instans sehat untuk `LoadBalancerName` dimensi dihitung di semua Availability Zone. Jika tidak, dihitung per Availability Zone. Kriteria pelaporan: Ada contoh terdaftar Statistik: Statistik yang paling berguna adalah `Average` dan `Maximum`. Statistik ini ditentukan oleh node penyeimbang beban. Perhatikan bahwa beberapa node penyeimbang beban mungkin menentukan bahwa sebuah instance tidak sehat untuk waktu yang singkat sementara node lain menentukan bahwa itu sehat. Contoh: Misalkan penyeimbang beban Anda memiliki 2 instance di us-west-2a dan 2 instance di us-west-2b, us-west-2a memiliki 1 instance yang tidak sehat, dan us-west-2b tidak memiliki instance yang tidak sehat. Dengan `AvailabilityZone` ukuran tersebut, ada rata-rata 1 contoh sehat dan 1 tidak sehat di us-barat-2a, dan rata-rata 2 kasus sehat dan 0 tidak sehat di us-barat-2b.

Metrik	Deskripsi
`HTTPCode_Backend_2XX`, `HTTPCode_Backend_3XX`, `HTTPCode_Backend_4XX`, `HTTPCode_Backend_5XX`	[HTTP listener] Jumlah kode respons HTTP yang dihasilkan oleh instance terdaftar. Hitungan ini tidak termasuk kode respons apa pun yang dihasilkan oleh penyeimbang beban. Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah `Sum`. Perhatikan bahwa`Minimum`,`Maximum`, dan `Average` semuanya 1. Contoh: Misalkan penyeimbang beban Anda memiliki 2 instance di us-west-2a dan 2 instance di us-west-2b, dan permintaan yang dikirim ke 1 instance di us-west-2a menghasilkan respons HTTP 500. Jumlah untuk us-west-2a mencakup respons kesalahan ini, sedangkan jumlah untuk us-west-2b tidak menyertakannya. Oleh karena itu, jumlah untuk penyeimbang beban sama dengan jumlah untuk us-barat-2a.
`HTTPCode_ELB_4XX`	[HTTP listener] Jumlah kode kesalahan klien HTTP 4XX yang dihasilkan oleh penyeimbang beban. Kesalahan klien dihasilkan ketika permintaan salah bentuk atau tidak lengkap. Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah `Sum`. Perhatikan bahwa`Minimum`,`Maximum`, dan `Average` semuanya 1. Contoh: Misalkan penyeimbang beban Anda mengaktifkan us-west-2a dan us-west-2b, dan permintaan klien menyertakan URL permintaan yang salah format. Akibatnya, kesalahan klien kemungkinan akan meningkat di semua Availability Zone. Jumlah untuk penyeimbang beban adalah jumlah nilai untuk Availability Zones.

Metrik	Deskripsi
HTTPCode_ELB_5XX	[HTTP listener] Jumlah kode kesalahan server HTTP 5XX yang dihasilkan oleh penyeimbang beban. Hitungan ini tidak termasuk kode respons apa pun yang dihasilkan oleh instans terdaftar . Metrik dilaporkan jika tidak ada contoh sehat yang terdaftar ke penyeimbang beban, atau jika tingkat permintaan melebihi kapasitas instans (spillover) atau penyeimbang beban. Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah Sum. Perhatikan bahwaMinimum,Maximum, dan Average semuanya 1. Contoh: Misalkan penyeimbang beban Anda mengaktifkan us-west-2a dan us-west-2b, dan instance di us-west-2a mengalami latensi tinggi dan lambat dalam menanggapi permintaan. Akibatnya , antrian lonjakan untuk node penyeimbang beban di us-west-2a mengisi dan klien menerima kesalahan 503. Jika us-barat-2b terus merespons secara normal, jumlah untuk penyeimbang beban sama dengan jumlah untuk us-barat-2a.

Metrik	Deskripsi
`Latency`	[HTTP listener] Total waktu berlalu, dalam hitungan detik, dari saat penyeimbang beban mengirim permintaan ke instance terdaftar hingga instance mulai mengirim header respons. [Pendengar TCP] Total waktu berlalu, dalam hitungan detik, agar penyeimbang beban berhasil membuat koneksi ke instance terdaftar. Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah `Average`. Gunakan `Maximum` untuk menentukan apakah beberapa permintaan membutuhkan waktu lebih lama dari rata-rata. Perhatikan bahwa `Minimum` biasanya tidak berguna. Contoh: Misalkan penyeimbang beban Anda memiliki 2 instance di us-west-2a dan 2 instance di us-west-2b, dan permintaan yang dikirim ke 1 instance di us-west-2a memiliki latensi yang lebih tinggi. Rata-rata untuk us-barat-2a memiliki nilai yang lebih tinggi daripada rata-rata untuk us-barat-2b.

Metrik	Deskripsi
`RequestCount`	Jumlah permintaan selesai atau koneksi yang dibuat selama interval yang ditentukan (1 atau 5 menit). [HTTP listener] Jumlah permintaan yang diterima dan dirutekan, termasuk respons kesalahan HTTP dari instance terdaftar. [Pendengar TCP] Jumlah koneksi yang dibuat ke instance terdaftar . Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah `Sum`. Perhatikan bahwa`Minimum`,`Maximum`, dan `Average` semua kembali 1. Contoh: Misalkan penyeimbang beban Anda memiliki 2 instance di us-west-2a dan 2 instance di us-west-2b, dan 100 permintaan dikirim ke penyeimbang beban. Ada 60 permintaan yang dikirim ke us-west-2a, dengan setiap instans menerima 30 permintaan, dan 40 permintaan dikirim ke us-west-2b, dengan setiap instans menerima 20 permintaan. Dengan `AvailabilityZone` dimensi, ada jumlah 60 permintaan di us-west-2a dan 40 permintaan di us-west-2b. Dengan `LoadBalancerName` dimensi, ada jumlah 100 permintaan.

Metrik	Deskripsi
`SpilloverCount`	Jumlah total permintaan yang ditolak karena antrian lonjakan penuh. [HTTP listener] Load balancer mengembalikan kode kesalahan HTTP 503. [Pendengar TCP] Penyeimbang beban menutup koneksi. Reporting criteria: Ada nilai bukan nol Statistics: Statistik yang paling berguna adalah `Sum`. Perhatikan bahwa`Average`,`Minimum`, dan `Maximum` dilaporkan per node penyeimbang beban dan biasanya tidak berguna. Contoh: Misalkan penyeimbang beban Anda mengaktifkan us-west-2a dan us-west-2b, dan instance di us-west-2a mengalami latensi tinggi dan lambat dalam menanggapi permintaan. Akibatnya, antrian lonjakan untuk node penyeimbang beban di us-west-2a terisi, menghasilkan limpahan. Jika us-west-2b terus merespons secara normal, jumlah untuk penyeimbang beban akan sama dengan jumlah untuk us-west-2a.

Metrik	Deskripsi
SurgeQueueLength	Jumlah total permintaan (pendengar HTTP) atau koneksi (pendengar TCP) yang menunggu perutean ke instance yang sehat. Ukuran maksimum antrian adalah 1.024. Permintaan atau koneksi tambahan ditolak saat antrian penuh. Untuk informasi selengkapnya, lihat SpilloverCount . Kriteria pelaporan: Ada nilai bukan nol. Statistik: Statistik yang paling berguna adalahMaximum, karena mewakili puncak permintaan antrian. AverageStatistik dapat berguna dalam kombinasi dengan Minimum dan Maximum untuk menentukan kisaran permintaan antrian. Perhatikan bahwa Sum tidak berguna. Contoh: Misalkan penyeimbang beban Anda mengaktifkan us-west-2a dan us-west-2b, dan instance di us-west-2a mengalami latensi tinggi dan lambat dalam menanggapi permintaan. Akibatnya , antrian lonjakan untuk node penyeimbang beban di us-west-2a terisi, dengan klien kemungkinan mengalami peningkatan waktu respons. Jika ini terus berlanjut, penyeimbang beban kemungkin an akan memiliki limpahan (lihat metrik). SpilloverCount Jika us-west-2b terus merespons secara normal, penyeimbang beban akan sama dengan us-west-2a. max max

Metrik	Deskripsi
`UnHealthyHostCount`	Jumlah instans tidak sehat yang terdaftar di penyeimbang beban Anda. Sebuah contoh dianggap tidak sehat setelah melebihi ambang tidak sehat yang dikonfigurasi untuk pemeriksaan kesehatan. Contoh yang tidak sehat dianggap sehat kembali setelah memenuhi ambang batas sehat yang dikonfigurasi untuk pemeriksaan kesehatan. Kriteria pelaporan: Ada contoh terdaftar Statistik: Statistik yang paling berguna adalah `Average` dan `Minimum`. Statistik ini ditentukan oleh node penyeimbang beban. Perhatikan bahwa beberapa node penyeimbang beban mungkin menentukan bahwa sebuah instance tidak sehat untuk waktu yang singkat sementara node lain menentukan bahwa itu sehat. Contoh: Lihat`HealthyHostCount` .

Metrik berikut memungkinkan Anda memperkirakan biaya jika memigrasikan Classic Load Balancer ke Application Load Balancer. Metrik ini dimaksudkan untuk penggunaan informasi saja, bukan untuk digunakan dengan CloudWatch alarm. Perhatikan bahwa jika Classic Load Balancer Anda memiliki beberapa pendengar, metrik ini digabungkan di seluruh pendengar.

Perkiraan ini didasarkan pada penyeimbang beban dengan satu aturan default dan sertifikat berukuran 2K. Jika Anda menggunakan sertifikat berukuran 4K atau lebih besar, sebaiknya Anda memperkirakan biaya sebagai berikut: buat Application Load Balancer berdasarkan Classic Load Balancer menggunakan alat migrasi dan pantau `ConsumedLCUs` metrik untuk Application Load Balancer. Untuk informasi selengkapnya, lihat Memigrasikan Classic Load Balancer Anda di Panduan Pengguna Elastic Load Balancing.

Metrik	Deskripsi
`EstimatedALBActive ConnectionCount`	Perkiraan jumlah koneksi TCP bersamaan yang aktif dari klien ke penyeimbang beban dan dari penyeimbang beban ke target.

Metrik	Deskripsi
`EstimatedALBConsumedLCUs`	Perkiraan jumlah load balancer capacity unit (LCU) yang digunakan oleh Application Load Balancer. Anda membayar untuk jumlah LCUs yang Anda gunakan per jam. Untuk informasi lebih lanjut, lihat Harga Elastic Load Balancing.
`EstimatedALBNewConnectionCount`	Perkiraan jumlah koneksi TCP baru yang dibuat dari klien ke penyeimbang beban dan dari penyeimbang beban ke target.
`EstimatedProcessedBytes`	Perkiraan jumlah byte yang diproses oleh Application Load Balancer.

Dimensi metrik untuk Classic Load Balancer

Untuk memfilter metrik Classic Load Balancer, gunakan dimensi berikut.

Dimensi	Deskripsi
`AvailabilityZone`	Memfilter data metrik berdasarkan Availability Zone yang ditentukan.
`LoadBalancerName`	Memfilter data metrik dengan penyeimbang beban yang ditentukan.

Statistik untuk metrik Classic Load Balancer

CloudWatch menyediakan statistik berdasarkan titik data metrik yang diterbitkan oleh Elastic Load Balancing. Statistik adalah agregasi data metrik selama periode waktu tertentu. Bila Anda meminta statistik, aliran data yang dikembalikan akan diidentifikasi dengan nama metrik dan dimensi. Dimensi adalah pasangan nama/nilai yang merupakan bagian dari identitas metrik. Misalnya, Anda dapat meminta statistik untuk semua EC2 instans sehat di balik penyeimbang beban yang diluncurkan di Availability Zone tertentu.

Statistik `Minimum` dan `Maximum` mencerminkan minimum dan maksimum yang dilaporkan oleh masing-masing simpul penyeimbang beban. Misalnya, ada 2 simpul penyeimbang beban. Satu simpul memiliki `HealthyHostCount` dengan `Minimum` 2, `Maximum` 10, dan `Average` 6, sedangkan simpul lainnya memiliki `HealthyHostCount` dengan `Minimum` 1, `Maximum` 5, dan `Average` 3. Oleh karena itu, penyeimbang beban memiliki `Minimum` 1, `Maximum` 10, dan `Average`sekitar 4.

Statistik `Sum` adalah nilai agregat di semua simpul penyeimbang beban. Karena metrik menyertakan beberapa laporan per periode, `Sum` hanya berlaku untuk metrik yang dikumpulkan di semua node penyeimbang beban, seperti,,, `RequestCount`HTTPCode_ELB_XXX, `HTTPCode_Backend_XXX` dan. `BackendConnectionErrors` `SpilloverCount`

Statistik `SampleCount` adalah jumlah sampel yang diukur. Karena metrik dikumpulkan berdasarkan interval dan peristiwa pengambilan sampel, statistik ini biasanya tidak berguna. Misalnya dengan `HealthyHostCount`, `SampleCount` didasarkan pada jumlah sampel yang dilaporkan setiap simpul penyeimbang beban, bukan jumlah host yang sehat.

Persentil menunjukkan posisi relatif suatu nilai dalam set data. Anda dapat menentukan persentil apa pun, menggunakan hingga dua tempat desimal (misalnya, hal 95.45). Misalnya, persentil ke-95 berarti bahwa 95 persen data berada di bawah nilai ini dan 5 persen di atas. Persentil sering kali digunakan untuk mengisolasi anomali. Misalnya, anggaplah aplikasi melayani sebagian besar permintaan dari cache dalam 1-2 ms, tetapi dalam 100-200 ms jika cache kosong. Maksimumnya mencerminkan kasus paling lambat, sekitar 200 ms. Rata-ratanya tidak menunjukkan distribusi data. Persentil memberikan tampilan performa aplikasi yang lebih bermakna. Dengan menggunakan persentil ke-99 sebagai pemicu Auto Scaling atau CloudWatch alarm, Anda dapat menargetkan bahwa tidak lebih dari 1 persen permintaan membutuhkan waktu lebih dari 2 ms untuk diproses.

Lihat CloudWatch metrik untuk penyeimbang beban Anda

Anda dapat melihat CloudWatch metrik untuk penyeimbang beban menggunakan konsol Amazon. EC2 Metrik ini ditampilkan sebagai grafik pemantauan. Grafik pemantauan menunjukkan titik data jika penyeimbang beban aktif dan menerima permintaan.

Atau, Anda dapat melihat metrik untuk penyeimbang beban menggunakan konsol. CloudWatch

Untuk melihat metrik menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban untuk membuka halaman detailnya.

4. Pilih tab Pemantauan.
5. Untuk mendapatkan tampilan yang lebih besar dari satu metrik, arahkan kursor ke grafiknya lalu pilih `Maximize` ikonnya. Metrik berikut tersedia:

   - Tuan Rumah Sehat — `HealthyHostCount`
   - Tuan Rumah Tidak Sehat — `UnHealthyHostCount`
   - Latensi Rata-Rata — `Latency`
   - Permintaan — `RequestCount`
   - Kesalahan Koneksi Backend - `BackendConnectionErrors`
   - Panjang Antrian Lonjakan - `SurgeQueueLength`
   - Jumlah limpahan — `SpilloverCount`
   - HTTP 2 XXs — `HTTPCode_Backend_2XX`
   - HTTP 3 XXs — `HTTPCode_Backend_3XX`
   - HTTP 4 XXs — `HTTPCode_Backend_4XX`
   - HTTP 5 XXs — `HTTPCode_Backend_5XX`
   - ELB HTTP 4 XXs — `HTTPCode_ELB_4XX`
   - ELB HTTP 5 XXs — `HTTPCode_ELB_5XX`
   - Perkiraan byte yang diproses — `EstimatedProcessedBytes`
   - Diperkirakan ALB dikonsumsi LCUs — `EstimatedALBConsumedLCUs`
   - Perkiraan jumlah koneksi aktif ALB — `EstimatedALBActiveConnectionCount`
   - Perkiraan jumlah koneksi baru ALB — `EstimatedALBNewConnectionCount`

Untuk melihat metrik menggunakan konsol CloudWatch

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Pada panel navigasi, silakan pilih Metrik.
3. Pilih namespace ELB.
4. Lakukan salah satu tindakan berikut:

   - Pilih dimensi metrik untuk melihat metrik menurut penyeimbang beban, berdasarkan Availability Zone, atau di semua penyeimbang beban.
   - Untuk melihat metrik di semua dimensi, ketikkan namanya di bidang pencarian.
   - Untuk melihat metrik penyeimbang beban tunggal, ketikkan namanya di kolom pencarian.

- Untuk melihat metrik untuk Availability Zone tunggal, ketikkan namanya di kolom pencarian.

Akses log untuk Classic Load Balancer Anda

Elastic Load Balancing memberikan log akses yang mengambil informasi mendetail tentang permintaan yang dikirim ke penyeimbang beban Anda. Setiap log berisi informasi, seperti waktu permintaan diterima, alamat IP klien, latensi, jalur permintaan, dan respons server. Anda dapat menggunakan log akses ini untuk menganalisis pola lalu lintas dan untuk memecahkan masalah.

Log akses adalah fitur opsional Elastic Load Balancing yang dinonaktifkan secara default. Setelah mengaktifkan log akses untuk penyeimbang beban, Elastic Load Balancing menangkap log dan menyimpannya di bucket Amazon S3 yang Anda tentukan. Anda dapat menonaktifkan pengelogan akses kapan saja.

Setiap berkas log akses dienkripsi secara otomatis menggunakan SSE-S3 sebelum disimpan dalam bucket S3 Anda dan didekripsi saat Anda mengaksesnya. Anda tidak perlu mengambil tindakan apapun; enkripsi dan dekripsi dilakukan secara transparan. Setiap file log dienkripsi dengan kunci unik, yang dienkripsi dengan kunci KMS yang diputar secara teratur. Untuk informasi selengkapnya, lihat Melindungi data menggunakan enkripsi sisi server dengan kunci enkripsi terkelola Amazon S3 (SSE-S3) di Panduan Pengguna Amazon S3.

Tidak ada biaya tambahan untuk log akses. Anda akan dikenakan biaya penyimpanan untuk Amazon S3, tetapi tidak akan dikenakan biaya untuk bandwidth yang digunakan oleh Elastic Load Balancing untuk mengirim file log ke Amazon S3. Untuk informasi selengkapnya tentang biaya penyimpanan, lihat Harga Amazon S3.

Daftar Isi

- Berkas log akses
- Entri log akses
- Memproses log akses
- Aktifkan log akses untuk Classic Load Balancer Anda
- Nonaktifkan log akses untuk Classic Load Balancer Anda

Berkas log akses

Elastic Load Balancing menerbitkan file log untuk setiap node penyeimbang beban pada interval yang Anda tentukan. Anda dapat menentukan interval penerbitan 5 menit atau 60 menit saat Anda mengaktifkan log akses untuk penyeimbang beban Anda. Secara default, Elastic Load Balancing menerbitkan log pada interval 60 menit. Jika interval diatur selama 5 menit, log diterbitkan pada 1:05, 1:10, 1:15, dan seterusnya. Awal pengiriman log tertunda hingga 5 menit jika interval diatur ke 5 menit, dan hingga 15 menit jika interval diatur ke 60 menit. Anda dapat memodifikasi interval penerbitan kapan saja.

Penyeimbang beban dapat mengirimkan beberapa log untuk periode yang sama. Ini biasanya terjadi jika situs memiliki lalu lintas tinggi, beberapa node penyeimbang beban, dan interval penerbitan log pendek.

Nama file log akses menggunakan format berikut:

```
amzn-s3-demo-loadbalancer-logs[/logging-prefix]/AWSLogs/aws-account-id/
elasticloadbalancing/region/yyyy/mm/dd/aws-account-id_elasticloadbalancing_region_load-
balancer-name_end-time_ip-address_random-string.log
```

amzn-s3- demo-loadbalancer-logs

Nama bucket S3.

prefix

(Opsional) Awalan (hierarki logis) untuk bucket. Awalan yang Anda tentukan tidak boleh menyertakan string`AWSLogs`. Untuk informasi selengkapnya, lihat Mengatur objek menggunakan awalan.

`AWSLogs`

Kami menambahkan bagian dari nama file dimulai dengan `AWSLogs` setelah nama bucket dan awalan opsional yang Anda tentukan.

aws-account-id

ID AWS akun pemilik.

region

Wilayah untuk penyeimbang beban dan bucket S3 Anda.

yyyy/mm/dd

Tanggal pengiriman log.

load-balancer-name

Nama penyeimbang beban.

akhir zaman

Tanggal dan waktu interval pengelogan berakhir. Misalnya, waktu akhir 20140215T2340Z berisi entri untuk permintaan yang dibuat antara 23:35 dan 23:40 jika interval penerbitan adalah 5 menit.

alamat ip

Alamat IP simpul penyeimbang beban yang menangani permintaan. Untuk penyeimbang beban internal, ini adalah alamat IP privat.

string acak

String acak yang dihasilkan sistem.

Berikut ini adalah contoh nama file log dengan awalan "my-app“:

```
s3://amzn-s3-demo-loadbalancer-logs/my-app/AWSLogs/123456789012/elasticloadbalancing/
us-west-2/2018/02/15/123456789012_elasticloadbalancing_us-west-2_my-
loadbalancer_20180215T2340Z_172.160.001.192_20sg8hgm.log
```

Berikut ini adalah contoh nama file log tanpa awalan:

```
s3://amzn-s3-demo-loadbalancer-logs/AWSLogs/123456789012/elasticloadbalancing/
us-west-2/2018/02/15/123456789012_elasticloadbalancing_us-west-2_my-
loadbalancer_20180215T2340Z_172.160.001.192_20sg8hgm.log
```

Anda dapat menyimpan file log dalam bucket selama yang diinginkan, tetapi Anda juga dapat menentukan aturan siklus hidup Amazon S3 untuk mengarsipkan atau menghapus file log secara otomatis. Untuk informasi selengkapnya, lihat Manajemen siklus hidup objek di Panduan Pengguna Amazon S3.

Entri log akses

Elastic Load Balancing mencatat permintaan yang dikirim ke penyeimbang beban, termasuk permintaan yang tidak pernah berhasil masuk ke instance back-end. Misalnya, jika klien mengirim

permintaan yang salah bentuk, atau tidak ada contoh yang sehat untuk merespons, permintaan tersebut masih dicatat.

> **Important**
>
> Permintaan log Elastic Load Balancing berdasarkan upaya terbaik. Sebaiknya gunakan log akses untuk memahami sifat permintaan, bukan sebagai penghitungan lengkap semua permintaan.

Sintaksis

Setiap entri log berisi rincian permintaan tunggal yang dibuat untuk penyeimbang beban. Semua bidang dalam entri log dibatasi oleh spasi. Setiap entri dalam file log memiliki format berikut:

```
timestamp elb client:port backend:port request_processing_time backend_processing_time
 response_processing_time elb_status_code backend_status_code received_bytes sent_bytes
 "request" "user_agent" ssl_cipher ssl_protocol
```

Tabel berikut menjelaskan bidang entri log akses.

Bidang	Deskripsi
Waktu	Waktu saat penyeimbang beban menerima permintaan dari klien, dalam format ISO 8601.
elb	Nama penyeimbang beban
client:port	Alamat IP dan port dari klien yang meminta.
backend:port	Alamat IP dan port dari instance terdaftar yang memproses permintaan ini. Jika penyeimbang beban tidak dapat mengirim permintaan ke instance terdaftar, atau jika instance menutup koneksi sebelum respons dapat dikirim, nilai ini disetel ke. - Nilai ini juga dapat diatur ke - jika instance terdaftar tidak merespons sebelum batas waktu idle.

Bidang	Deskripsi
request_processing_time	[HTTP listener] Total waktu berlalu, dalam hitungan detik, dari saat penyeimbang beban menerima permintaan hingga saat mengirimnya ke instance terdaftar. [Pendengar TCP] Total waktu berlalu, dalam hitungan detik, dari saat penyeimbang beban menerima koneksi TCP/SSL dari klien hingga saat penyeimbang beban mengirimkan byte pertama data ke instance terdaftar. Nilai ini disetel ke `-1` jika penyeimbang beban tidak dapat mengirimkan permintaan ke instance terdaftar. Ini dapat terjadi jika instance terdaftar menutup koneksi sebelum batas waktu idle atau jika klien mengirim permintaan yang salah bentuk. Selain itu, untuk pendengar TCP, ini dapat terjadi jika klien membuat koneksi dengan penyeimbang beban tetapi tidak mengirim data apa pun. Nilai ini juga dapat diatur ke `-1` jika instance terdaftar tidak merespons sebelum batas waktu idle.
backend_processing_time	[HTTP listener] Total waktu berlalu, dalam hitungan detik, dari saat penyeimbang beban mengirim permintaan ke instance terdaftar hingga instance mulai mengirim header respons. [Pendengar TCP] Total waktu berlalu, dalam hitungan detik, agar penyeimbang beban berhasil membuat koneksi ke instance terdaftar. Nilai ini disetel ke `-1` jika penyeimbang beban tidak dapat mengirimkan permintaan ke instance terdaftar. Ini dapat terjadi jika instance terdaftar menutup koneksi sebelum batas waktu idle atau jika klien mengirim permintaan yang salah bentuk. Nilai ini juga dapat diatur ke `-1` jika instance terdaftar tidak merespons sebelum batas waktu idle.

Bidang	Deskripsi
response_processing_time	[HTTP listener] Total waktu berlalu (dalam detik) dari saat penyeimbang beban menerima header respons dari instance terdaftar hingga mulai mengirim respons ke klien. Ini termasuk waktu antrean di penyeimbang beban dan waktu akuisisi koneksi dari penyeimbang beban ke klien. [Pendengar TCP] Total waktu berlalu, dalam hitungan detik, dari saat penyeimbang beban menerima byte pertama dari instance terdaftar hingga mulai mengirim respons ke klien. Nilai ini disetel ke `-1` jika penyeimbang beban tidak dapat mengirimkan permintaan ke instance terdaftar. Ini dapat terjadi jika instance terdaftar menutup koneksi sebelum batas waktu idle atau jika klien mengirim permintaan yang salah bentuk. Nilai ini juga dapat diatur ke `-1` jika instance terdaftar tidak merespons sebelum batas waktu idle.
elb_status_code	[HTTP listener] Kode status respons dari penyeimbang beban.
backend_status_code	[HTTP listener] Kode status respons dari instance terdaftar.
received_bytes	Ukuran permintaan dalam byte, diterima dari klien (peminta). [HTTP listener] Nilainya mencakup badan permintaan tetapi bukan header. [Pendengar TCP] Nilainya mencakup badan permintaan dan header.
sent_bytes	Ukuran respons dalam byte, dikirim ke klien (peminta). [HTTP listener] Nilainya mencakup badan respons tetapi bukan header. [Pendengar TCP] Nilainya mencakup badan permintaan dan header.

Bidang	Deskripsi
request	Baris permintaan dari klien tertutup tanda kutip ganda dan dicatat dalam format berikut: Metode HTTP + Protokol: //Host header: port+Path+versi HTTP. Penyeimbang beban mempertahankan URL yang dikirim oleh klien, sebagaimana adanya, saat mencatat URI permintaan. Ini tidak mengatur jenis konten untuk berkas log akses. Saat Anda memproses bidang ini, pertimbangkan bagaimana klien mengirim URL. [TCP listener] URL adalah tiga tanda hubung, masing-masing dipisahkan oleh spasi, dan diakhiri dengan spasi (“- - - “).
user_agent	[HTTP/HTTPS listener] A User-Agent string that identifies the client that originated the request. The string consists of one or more product identifiers, product[/version]. Jika string lebih panjang dari 8 KB, itu terpotong.
ssl_cipher	[HTTPS/SSL listener] The SSL cipher. This value is recorded only if the incoming SSL/TLSkoneksi dibuat setelah negosiasi yang sukses. Jika tidak, nilainya diatur ke-.
ssl_protocol	[HTTPS/SSL listener] The SSL protocol. This value is recorded only if the incoming SSL/TLSkoneksi dibuat setelah negosiasi yang sukses. Jika tidak, nilainya diatur ke-.

Contoh

Contoh entri HTTP

Berikut ini adalah contoh entri log untuk listener HTTP (port 80 ke port 80):

```
2015-05-13T23:39:43.945958Z my-loadbalancer 192.168.131.39:2817 10.0.0.1:80 0.000073
 0.001048 0.000057 200 200 0 29 "GET http://www.example.com:80/ HTTP/1.1" "curl/7.38.0"
 - -
```

Contoh entri HTTPS

Berikut ini adalah contoh entri log untuk listener HTTPS (port 443 ke port 80):

```
2015-05-13T23:39:43.945958Z my-loadbalancer 192.168.131.39:2817 10.0.0.1:80
 0.000086 0.001048 0.001337 200 200 0 57 "GET https://www.example.com:443/ HTTP/1.1"
 "curl/7.38.0" DHE-RSA-AES128-SHA TLSv1.2
```

Contoh entri TCP

Berikut ini adalah contoh entri log untuk pendengar TCP (port 8080 ke port 80):

```
2015-05-13T23:39:43.945958Z my-loadbalancer 192.168.131.39:2817 10.0.0.1:80 0.001069
 0.000028 0.000041 - - 82 305 "- - - " "-" - -
```

Contoh entri SSL

Berikut ini adalah contoh entri log untuk pendengar SSL (port 8443 ke port 80):

```
2015-05-13T23:39:43.945958Z my-loadbalancer 192.168.131.39:2817 10.0.0.1:80 0.001065
 0.000015 0.000023 - - 57 502 "- - - " "-" ECDHE-ECDSA-AES128-GCM-SHA256 TLSv1.2
```

Memproses log akses

Jika ada banyak permintaan di situs web Anda, penyeimbang beban Anda dapat menghasilkan berkas log dengan gigabyte data. Anda mungkin tidak dapat memproses data dalam jumlah besar menggunakan line-by-line pemrosesan. Oleh karena itu, Anda mungkin harus menggunakan alat analisis yang memberikan solusi pemrosesan paralel. Misalnya, Anda dapat menggunakan alat analisis berikut untuk menganalisis dan memproses log akses:

- Amazon Athena adalah layanan kueri interaktif yang memudahkan untuk menganalisis data di Amazon S3 menggunakan SQL standar. Untuk informasi selengkapnya, lihat Menanyakan log Classic Load Balancer di Panduan Pengguna Amazon Athena.
- Loggly
- Splunk
- Logika Sumo

Aktifkan log akses untuk Classic Load Balancer Anda

Untuk mengaktifkan log akses penyeimbang beban, Anda harus menentukan nama bucket Amazon S3 tempat penyeimbang beban akan menyimpan log. Anda juga harus melampirkan kebijakan bucket ke bucket ini yang memberikan izin Elastic Load Balancing untuk menulis ke bucket.

Tugas

- Langkah 1: Buat ember S3
- Langkah 2: Lampirkan kebijakan ke bucket S3 Anda
- Langkah 3: Konfigurasikan log akses
- Langkah 4: Verifikasi izin bucket
- Pemecahan Masalah

Langkah 1: Buat ember S3

Saat mengaktifkan log akses, Anda harus menentukan bucket S3 untuk file log akses. Bucket harus memenuhi persyaratan berikut.

Persyaratan

- Bucket harus ditempatkan di Wilayah yang sama dengan penyeimbang beban. Bucket dan load balancer dapat dimiliki oleh akun yang berbeda.
- Satu-satunya opsi enkripsi sisi server yang didukung adalah kunci yang dikelola Amazon S3 (SSE-S3). Untuk informasi selengkapnya, lihat kunci enkripsi terkelola Amazon S3 (SSE-S3).

Untuk membuat bucket S3 menggunakan konsol Amazon S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Pilih Buat bucket.
3. Pada halaman Create bucket, lakukan hal berikut:
    a. Untuk Bucket name, masukkan nama untuk bucket Anda. Nama ini harus unik di semua nama bucket yang ada di Amazon S3. Di beberapa Wilayah, mungkin ada pembatasan tambahan pada nama bucket. Untuk informasi selengkapnya, lihat Kuota, batasan, dan batasan bucket di Panduan Pengguna Amazon S3.
    b. Untuk AWS Region, pilih Wilayah tempat Anda membuat penyeimbang beban.
    c. Untuk enkripsi Default, pilih kunci yang dikelola Amazon S3 (SSE-S3).
    d. Pilih Buat bucket.

Langkah 2: Lampirkan kebijakan ke bucket S3 Anda

Bucket S3 Anda harus memiliki kebijakan bucket yang memberikan izin Elastic Load Balancing untuk menulis log akses ke bucket. Kebijakan bucket adalah kumpulan pernyataan JSON yang ditulis dalam bahasa kebijakan akses untuk menentukan izin akses untuk bucket Anda. Setiap pernyataan mencakup informasi tentang satu izin dan berisi serangkaian elemen.

Jika Anda menggunakan bucket yang sudah memiliki kebijakan terlampir, Anda dapat menambahkan pernyataan untuk log akses Elastic Load Balancing ke kebijakan. Jika Anda melakukannya, sebaiknya Anda mengevaluasi kumpulan izin yang dihasilkan untuk memastikan bahwa izin tersebut sesuai untuk pengguna yang memerlukan akses ke bucket untuk log akses.

Kebijakan bucket yang tersedia

Kebijakan bucket yang akan Anda gunakan bergantung Wilayah AWS pada bucket.

⚠ Tingkatkan keamanan dengan menggunakan bucket ARNs S3 yang presisi.

- Gunakan jalur sumber daya lengkap, bukan hanya ARN bucket S3.
- Sertakan bagian ID akun dari bucket S3 ARN.
- Jangan gunakan wildcard (*) di bagian ID akun ARN bucket S3.

Wilayah tersedia per Agustus 2022 atau lebih baru

Kebijakan ini memberikan izin ke layanan pengiriman log yang ditentukan. Gunakan kebijakan ini untuk penyeimbang beban di Wilayah berikut:

- Asia Pasifik (Hyderabad)
- Asia Pasifik (Malaysia)
- Asia Pasifik (Melbourne)
- Asia Pasifik (Taipei)
- Asia Pasifik (Thailand)
- Kanada Barat (Calgary)
- Eropa (Spanyol)
- Eropa (Zürich)
- Israel (Tel Aviv)

- Timur Tengah (UEA)
- Meksiko (Tengah)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "logdelivery.elasticloadbalancing.amazonaws.com"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/prefix/AWSLogs/123456789012/*"
    }
  ]
}
```

Untuk`Resource`, masukkan ARN lokasi untuk log akses, menggunakan format yang ditunjukkan dalam kebijakan contoh. Selalu sertakan ID akun akun dengan penyeimbang beban di jalur sumber daya bucket S3 ARN. Ini memastikan bahwa hanya penyeimbang beban dari akun tertentu yang dapat menulis log akses ke bucket S3.

ARN bucket S3 yang Anda tentukan bergantung pada apakah Anda berencana untuk menyertakan awalan saat Anda mengaktifkan log akses di langkah 3.

Contoh S3 bucket ARN dengan awalan

Nama bucket S3 adalah amzn-s3-demo-logging-bucket dan awalannya adalah. logging-prefix

```
arn:aws:s3:::amzn-s3-demo-logging-bucket/logging-prefix/AWSLogs/123456789012/*
```

Contoh S3 bucket ARN tanpa awalan

Nama bucket S3 adalahamzn-s3-demo-logging-bucket. Tidak ada bagian awalan di ember S3 ARN.

```
arn:aws:s3:::amzn-s3-demo-logging-bucket/AWSLogs/123456789012/*
```

Wilayah tersedia sebelum Agustus 2022

Kebijakan ini memberikan izin ke akun Elastic Load Balancing yang ditentukan. Gunakan kebijakan ini untuk penyeimbang beban di Wilayah yang tercantum di bawah ini.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::elb-account-id:root"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/prefix/AWSLogs/123456789012/*"
    }
  ]
}
```

Untuk`Principal`, ganti *`elb-account-id`* dengan ID akun Elastic Load Balancing untuk Wilayah penyeimbang beban:

- AS Timur (Virginia N.) — 127311923021
- AS Timur (Ohio) — 033677994240
- AS Barat (California N.) — 027434742980
- AS Barat (Oregon) — 797873946194
- Afrika (Cape Town) — 098369216593
- Asia Pasifik (Hong Kong) — 754344448648
- Asia Pasifik (Jakarta) - 589379963580
- Asia Pasifik (Mumbai) — 718504428378
- Asia Pasifik (Osaka) — 383597477331
- Asia Pasifik (Seoul) — 600734575887
- Asia Pasifik (Singapura) — 114774131450
- Asia Pasifik (Sydney) — 783225319266
- Asia Pasifik (Tokyo) — 582318560864
- Kanada (Tengah) — 985666609251
- Eropa (Frankfurt am Main) — 054676820928
- Eropa (Irlandia) — 156460612806
- Eropa (London) — 652711504416
- Eropa (Milan) — 635631232127

- Eropa (Paris) — 009996457667
- Eropa (Stockholm) — 897822967062
- Timur Tengah (Bahrain) — 076674570225
- Amerika Selatan (São Paulo) — 507241528517

Untuk`Resource`, masukkan ARN lokasi untuk log akses, menggunakan format yang ditunjukkan dalam kebijakan contoh. Selalu sertakan ID akun akun dengan penyeimbang beban di jalur sumber daya bucket S3 ARN. Ini memastikan bahwa hanya penyeimbang beban dari akun tertentu yang dapat menulis log akses ke bucket S3.

ARN bucket S3 yang Anda tentukan bergantung pada apakah Anda berencana untuk menyertakan awalan saat Anda mengaktifkan log akses di langkah 3.

Contoh S3 bucket ARN dengan awalan

Nama bucket S3 adalah amzn-s3-demo-logging-bucket dan awalannya adalah. logging-prefix

```
arn:aws:s3:::amzn-s3-demo-logging-bucket/logging-prefix/AWSLogs/123456789012/*
```

Contoh S3 bucket ARN tanpa awalan

Nama bucket S3 adalah`amzn-s3-demo-logging-bucket`.

```
arn:aws:s3:::amzn-s3-demo-logging-bucket/AWSLogs/123456789012/*
```

AWS GovCloud (US) Regions

Kebijakan ini memberikan izin ke ID akun Elastic Load Balancing yang ditentukan. Gunakan kebijakan ini untuk penyeimbang beban di Wilayah. AWS GovCloud (US)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws-us-gov:iam::elb-account-id:root"
      },
      "Action": "s3:PutObject",
```

```
      "Resource": "arn:aws-us-gov:s3:::amzn-s3-demo-bucket/prefix/AWSLogs/123456789012/
*"
    }
  ]
}
```

Untuk`Principal`, ganti *`elb-account-id`* dengan ID akun Elastic Load Balancing untuk Wilayah penyeimbang beban:

- AWS GovCloud (AS-Barat) — 048591011584
- AWS GovCloud (AS-Timur) — 190560391635

Untuk`Resource`, masukkan ARN lokasi untuk log akses. Selalu sertakan ID akun akun dengan penyeimbang beban di jalur sumber daya bucket S3 ARN. Ini memastikan bahwa hanya penyeimbang beban dari akun tertentu yang dapat menulis log akses ke bucket S3.

ARN bucket S3 yang Anda tentukan bergantung pada apakah Anda berencana untuk menyertakan awalan saat Anda mengaktifkan log akses di langkah 3.

Contoh S3 bucket ARN dengan awalan

Nama bucket S3 adalah amzn-s3-demo-logging-bucket dan awalannya adalah. logging-prefix

```
arn:aws-us-gov:s3:::amzn-s3-demo-logging-bucket/logging-prefix/AWSLogs/123456789012/*
```

Contoh S3 bucket ARN tanpa awalan

Nama bucket S3 adalah`amzn-s3-demo-logging-bucket`. Tidak ada bagian awalan di ember S3 ARN.

```
arn:aws-us-gov:s3:::amzn-s3-demo-logging-bucket/AWSLogs/123456789012/*
```

Untuk melampirkan kebijakan bucket untuk log akses ke bucket Anda menggunakan konsol Amazon S3

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Pilih nama bucket untuk membuka halaman detailnya.
3. Pilih Izin lalu pilih Kebijakan Bucket, Edit.

4. Perbarui kebijakan bucket untuk memberikan izin yang diperlukan.
5. Pilih Simpan perubahan.

Langkah 3: Konfigurasikan log akses

Gunakan prosedur berikut untuk mengonfigurasi log akses untuk menangkap informasi permintaan dan mengirimkan file log ke bucket S3 Anda.

Persyaratan

Bucket harus memenuhi persyaratan yang dijelaskan pada langkah 1, dan Anda harus melampirkan kebijakan bucket seperti yang dijelaskan pada langkah 2. Jika Anda menentukan awalan, itu tidak harus menyertakan string "AWSLogs".

Untuk mengonfigurasi log akses untuk penyeimbang beban Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban Anda untuk membuka halaman detailnya.
4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian Monitoring, lakukan hal berikut:
    a. Aktifkan log Access.
    b. Untuk URI S3, masukkan URI S3 untuk file log Anda. URI yang Anda tentukan bergantung pada apakah Anda menggunakan awalan.
        - URI dengan awalan: `s3://amzn-s3-demo-logging-bucket/logging-prefix`
        - URI tanpa awalan: `s3://amzn-s3-demo-logging-bucket`
    c. Pertahankan interval Logging sebagai`60 minutes - default`.
    d. Pilih Simpan perubahan.

Untuk mengonfigurasi log akses untuk penyeimbang beban Anda menggunakan AWS CLI

Pertama, buat file.json yang memungkinkan Elastic Load Balancing menangkap dan mengirimkan log setiap 60 menit ke bucket S3 yang Anda buat untuk log:

```
{
```

```
  "AccessLog": {
    "Enabled": true,
    "S3BucketName": "amzn-s3-demo-logging-bucket",
    "EmitInterval": 60,
    "S3BucketPrefix": "my-app"
  }
}
```

Selanjutnya, tentukan file.json dalam modify-load-balancer-attributesperintah sebagai berikut:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --load-
balancer-attributes file://my-json-file.json
```

Berikut ini adalah contoh respons.

```
{
    "LoadBalancerAttributes": {
        "AccessLog": {
            "Enabled": true,
            "EmitInterval": 60,
            "S3BucketName": "amzn-s3-demo-logging-bucket",
            "S3BucketPrefix": "my-app"
        }
    },
    "LoadBalancerName": "my-loadbalancer"
}
```

Untuk mengelola bucket S3 untuk log akses Anda

Pastikan untuk menonaktifkan log akses sebelum menghapus bucket yang dikonfigurasi untuk log akses. Jika tidak, jika ada bucket baru dengan nama yang sama dan kebijakan bucket yang diperlukan yang dibuat dalam bucket Akun AWS yang tidak Anda miliki, Elastic Load Balancing dapat menulis log akses untuk penyeimbang beban Anda ke bucket baru ini.

Langkah 4: Verifikasi izin bucket

Setelah log akses diaktifkan untuk penyeimbang beban Anda, Elastic Load Balancing memvalidasi bucket S3 dan membuat file pengujian untuk memastikan bahwa kebijakan bucket menentukan izin yang diperlukan. Anda dapat menggunakan konsol S3 untuk memverifikasi bahwa file pengujian telah dibuat. File uji bukan berkas log akses yang sebenarnya; file tersebut tidak berisi contoh catatan.

Untuk memverifikasi bahwa Elastic Load Balancing membuat file uji di bucket S3 Anda

1. Buka konsol Amazon S3 di. https://console.aws.amazon.com/s3/
2. Pilih nama bucket S3 yang Anda tentukan untuk log akses.
3. Arahkan ke file pengujian,`ELBAccessLogTestFile`. Lokasi tergantung pada apakah Anda menggunakan awalan.
   - Lokasi dengan awalan:*amzn-s3-demo-loadbalancer-logs*//*logging-prefix*/AWSLogs/*123456789012*ELBAccessLogTestFile
   - Lokasi tanpa awalan:*amzn-s3-demo-loadbalancer-logs*//AWSLogs/*123456789012*ELBAccessLogTestFile

Pemecahan Masalah

Akses Ditolak untuk ember:***bucket-name***. Silakan periksa izin S3bucket

Jika Anda menerima kesalahan ini, berikut ini adalah kemungkinan penyebabnya:

- Kebijakan bucket tidak memberikan izin Elastic Load Balancing untuk menulis log akses ke bucket. Verifikasi bahwa Anda menggunakan kebijakan bucket yang benar untuk Wilayah tersebut. Verifikasi bahwa ARN sumber daya menggunakan nama bucket yang sama dengan yang Anda tentukan saat mengaktifkan log akses. Verifikasi bahwa ARN sumber daya tidak menyertakan awalan jika Anda tidak menentukan awalan saat Anda mengaktifkan log akses.
- Bucket menggunakan opsi enkripsi sisi server yang tidak didukung. Bucket harus menggunakan kunci yang dikelola Amazon S3 (SSE-S3).

Nonaktifkan log akses untuk Classic Load Balancer Anda

Anda dapat menonaktifkan log akses untuk penyeimbang beban Anda kapan saja. Setelah Anda menonaktifkan log akses, log akses Anda tetap berada di Amazon S3 hingga Anda menghapusnya. Untuk informasi selengkapnya, lihat Bekerja dengan bucket S3 di Panduan Pengguna Amazon S3.

Untuk menonaktifkan log akses untuk penyeimbang beban Anda menggunakan konsol

1. Buka EC2 konsol Amazon di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di bawah PENYEIMBANGAN BEBAN, pilih Penyeimbang beban.
3. Pilih nama penyeimbang beban Anda untuk membuka halaman detailnya.

4. Pada tab Atribut, pilih Edit.
5. Pada halaman Edit atribut penyeimbang beban, di bagian Monitoring, nonaktifkan log Access.

Untuk menonaktifkan log akses menggunakan AWS CLI

Gunakan modify-load-balancer-attributesperintah berikut untuk menonaktifkan log akses:

```
aws elb modify-load-balancer-attributes --load-balancer-name my-loadbalancer --load-balancer-attributes "{\"AccessLog\":{\"Enabled\":false}}"
```

Berikut adalah respons contohnya:

```
{
    "LoadBalancerName": "my-loadbalancer",
    "LoadBalancerAttributes": {
        "AccessLog": {
            "S3BucketName": "amzn-s3-demo-loadbalancer-logs",
            "EmitInterval": 60,
            "Enabled": false,
            "S3BucketPrefix": "my-app"
        }
    }
}
```

Memecahkan masalah Classic Load Balancer

Tabel berikut mencantumkan sumber pemecahan masalah yang menurut Anda berguna saat bekerja dengan Classic Load Balancer.

Kesalahan API

Kesalahan
CertificateNotFound: Tidak terdefinisi
OutofService: Terjadi kesalahan sementara

Kesalahan HTTP

Kesalahan
HTTP 400: PERMINTAAN BURUK
HTTP 405: METHOD_NOT_ALLOWED
HTTP 408: Waktu habis permintaan
HTTP 502: Gateway buruk
503 Layanan Tidak Tersedia
HTTP 504: Waktu habis gateway

Metrik kode respons

Metrik kode respons
HTTPCode_ELB_4XX
HTTPCode_ELB_5XX
HTTPCode_Backend_2xx

Metrik kode respons
HTTPCode_Backend_3xx
HTTPCode_Backend_4XX
HTTPCode_Backend_5XX

Masalah pemeriksaan kesehatan

Isu
Kesalahan halaman target pemeriksaan kesehatan
Koneksi ke instance telah habis
Otentikasi kunci publik gagal
Instance tidak menerima lalu lintas dari penyeimbang beban
Port pada instance tidak terbuka
Instance dalam grup Auto Scaling gagal dalam pemeriksaan kesehatan ELB

Masalah konektivitas

Isu
Klien tidak dapat menyambung ke Load Balancer yang menghadap internet
Permintaan yang dikirim ke domain kustom tidak diterima oleh penyeimbang beban
Permintaan HTTPS yang dikirim ke penyeimbang beban mengembalikan “NET: :ERR_CERT _COMMON_NAME_INVALID”

Masalah pendaftaran instans

Isu
Terlalu lama untuk mendaftarkan sebuah EC2 instans
Tidak dapat mendaftarkan instance yang diluncurkan dari AMI berbayar

Memecahkan masalah Classic Load Balancer: Kesalahan API

Berikut ini adalah pesan kesalahan yang ditampilkan oleh Elastic Load Balancing API, penyebab potensial, dan langkah-langkah yang dapat Anda ambil untuk menyelesaikan masalah.

Pesan Kesalahan

- CertificateNotFound: Tidak terdefinisi
- OutofService: Terjadi kesalahan sementara

CertificateNotFound: Tidak terdefinisi

Penyebab 1: Ada keterlambatan dalam menyebarkan sertifikat ke semua Wilayah saat dibuat menggunakan. AWS Management Console Ketika penundaan ini terjadi, pesan kesalahan ditampilkan pada langkah terakhir dalam proses pembuatan penyeimbang beban.

Solusi 1: Tunggu sekitar 15 menit dan kemudian coba lagi. Jika masalah berlanjut, pergi ke AWS Dukungan Pusat untuk bantuan.

Penyebab 2: Jika Anda menggunakan API AWS CLI atau secara langsung, Anda dapat menerima kesalahan ini jika Anda memberikan Nama Sumber Daya Amazon (ARN) untuk sertifikat yang tidak ada.

Solusi 2: Gunakan tindakan AWS Identity and Access Management (IAM) GetServerCertificateuntuk mendapatkan sertifikat ARN dan verifikasi bahwa Anda memberikan nilai yang benar untuk ARN.

OutofService: Terjadi kesalahan sementara

Penyebab: Ada masalah internal sementara dalam layanan Elastic Load Balancing atau jaringan yang mendasarinya. Masalah sementara ini juga dapat terjadi ketika Elastic Load Balancing menanyakan kesehatan penyeimbang beban dan instans terdaftarnya.

Solusi: Coba lagi panggilan API. Jika masalah berlanjut, pergi ke AWS Dukungan Pusat untuk bantuan.

Memecahkan masalah Classic Load Balancer: Kesalahan HTTP

Metode HTTP (juga disebut kata kerja) menentukan tindakan yang akan dilakukan pada sumber daya yang menerima permintaan HTTP. Metode standar untuk permintaan HTTP didefinisikan dalam RFC 2616, Definisi Metode. Metode standar termasuk GET, POST, PUT, HEAD, dan OPTIONS. Beberapa aplikasi web memerlukan (dan terkadang memperkenalkan) metode yang merupakan ekstensi dari metode HTTP/1.1. Contoh umum dari metode ekstensi HTTP termasuk PATCH, REPORT, MKCOL, PROPFIND, MOVE, dan LOCK. Elastic Load Balancing menerima semua metode HTTP standar dan non-standar.

Permintaan dan tanggapan HTTP menggunakan bidang header untuk mengirim informasi tentang pesan HTTP. Bidang header adalah pasangan nama-nilai dipisahkan titik dua yang dipisahkan oleh cariage return (CR) dan line feed (LF). Satu set standar bidang header HTTP didefinisikan dalam RFC 2616, Message Header. Untuk informasi selengkapnya, lihat Header HTTP dan Classic Load Balancer.

Ketika penyeimbang beban menerima permintaan HTTP, ia memeriksa permintaan yang salah bentuk dan panjang metode. Total panjang metode dalam permintaan HTTP ke penyeimbang beban tidak boleh melebihi 127 karakter. Jika permintaan HTTP melewati kedua pemeriksaan, penyeimbang beban mengirimkan permintaan ke EC2 instance. Jika bidang metode dalam permintaan salah bentuk, penyeimbang beban merespons dengan kesalahan. HTTP 400: PERMINTAAN BURUK Jika panjang metode dalam permintaan melebihi 127 karakter, penyeimbang beban merespons dengan kesalahan. HTTP 405: METHOD_NOT_ALLOWED

EC2 Instance memproses permintaan yang valid dengan menerapkan metode dalam permintaan dan mengirim respons kembali ke klien. Instance Anda harus dikonfigurasi untuk menangani metode yang didukung dan tidak didukung.

Berikut ini adalah pesan kesalahan yang dikembalikan oleh penyeimbang beban Anda, penyebab potensial, dan langkah-langkah yang dapat Anda ambil untuk menyelesaikan masalah.

Pesan Kesalahan

- HTTP 400: PERMINTAAN BURUK
- HTTP 405: METHOD_NOT_ALLOWED
- HTTP 408: Waktu habis permintaan

- HTTP 502: Gateway buruk
- 503 Layanan Tidak Tersedia
- HTTP 504: Waktu habis gateway

HTTP 400: PERMINTAAN BURUK

Deskripsi: Menunjukkan bahwa klien mengirim permintaan yang buruk.

Penyebab 1: Klien mengirim permintaan cacat yang tidak memenuhi spesifikasi HTTP. Misalnya, permintaan tidak dapat memiliki spasi di URL.

Penyebab 2: Klien menggunakan metode HTTP CONNECT, yang tidak didukung oleh Elastic Load Balancing.

Solusi: Connect langsung ke instans Anda dan tangkap detail permintaan klien. Tinjau header dan URL untuk permintaan yang salah. Verifikasi bahwa permintaan tersebut memenuhi spesifikasi HTTP. Verifikasi bahwa HTTP CONNECT tidak digunakan.

HTTP 405: METHOD_NOT_ALLOWED

Deskripsi: Menunjukkan bahwa panjang metode tidak valid.

Penyebab: Panjang metode di header permintaan melebihi 127 karakter.

Solusi: Periksa panjang metode.

HTTP 408: Waktu habis permintaan

Deskripsi: Menunjukkan bahwa klien membatalkan permintaan atau gagal mengirim permintaan penuh.

Penyebab 1: Gangguan jaringan atau konstruksi permintaan yang buruk, seperti header yang terbentuk sebagian; ukuran konten yang ditentukan tidak sesuai dengan ukuran konten aktual yang ditransmisikan; dan seterusnya.

Solusi 1: Periksa kode yang membuat permintaan dan coba kirimkan langsung ke instance terdaftar Anda (atau lingkungan pengembangan/pengujian) di mana Anda memiliki kontrol lebih besar untuk memeriksa permintaan yang sebenarnya.

Penyebab 2: Koneksi ke klien ditutup (penyeimbang beban tidak dapat mengirim respons)

Solusi 2: Verifikasi bahwa klien tidak menutup koneksi sebelum respons dikirim dengan menggunakan packet sniffer pada mesin yang membuat permintaan.

HTTP 502: Gateway buruk

Deskripsi: Menunjukkan bahwa penyeimbang beban tidak dapat mengurai respons yang dikirim dari instance terdaftar.

Penyebab: Respons yang salah bentuk dari instance atau berpotensi menjadi masalah dengan penyeimbang beban.

Solusi: Verifikasi bahwa respons yang dikirim dari instance sesuai dengan spesifikasi HTTP. Pergi ke AWS Dukungan Pusat untuk bantuan.

503 Layanan Tidak Tersedia

Deskripsi: Menunjukkan bahwa penyeimbang beban atau instans terdaftar menyebabkan kesalahan.

Penyebab 1: Kapasitas yang tidak mencukupi dalam penyeimbang beban untuk menangani permintaan.

Solusi 1: Ini harus menjadi masalah sementara dan tidak boleh berlangsung lebih dari beberapa menit. Jika terus berlanjut, pergilah ke AWS Dukungan Pusat untuk meminta bantuan.

Penyebab 2: Tidak ada contoh terdaftar.

Solusi 2: Daftarkan setidaknya satu instans di setiap Availability Zone tempat penyeimbang beban Anda dikonfigurasi untuk merespons. Verifikasi ini dengan melihat `HealthyHostCount` metrik di CloudWatch. Jika Anda tidak dapat memastikan bahwa instans terdaftar di setiap Availability Zone, sebaiknya aktifkan penyeimbangan beban lintas zona. Untuk informasi selengkapnya, lihat Konfigurasikan load balancing lintas zona untuk Classic Load Balancer.

Penyebab 3: Tidak ada contoh yang sehat.

Solusi 3: Pastikan Anda memiliki instans yang sehat di setiap Availability Zone tempat penyeimbang beban Anda dikonfigurasi untuk merespons. Verifikasi ini dengan melihat `HealthyHostCount` metrik.

Penyebab 4: Antrian lonjakan penuh.

Solusi 4: Pastikan instans Anda memiliki kapasitas yang cukup untuk menangani tingkat permintaan. Verifikasi ini dengan melihat `SpilloverCount` metrik.

HTTP 504: Waktu habis gateway

Deskripsi: Menunjukkan bahwa penyeimbang beban menutup sambungan karena permintaan tidak selesai dalam periode batas waktu idle.

Penyebab 1: Aplikasi membutuhkan waktu lebih lama untuk merespons daripada batas waktu idle yang dikonfigurasi.

Solusi 1: Pantau `HTTPCode_ELB_5XX` dan `Latency` metrik. Jika ada peningkatan metrik ini, bisa jadi karena aplikasi tidak merespons dalam periode batas waktu idle. Untuk detail tentang permintaan yang waktunya habis, aktifkan log akses pada penyeimbang beban dan tinjau 504 kode respons di log yang dihasilkan oleh Elastic Load Balancing. Jika perlu, Anda dapat meningkatkan kapasitas atau meningkatkan batas waktu idle yang dikonfigurasi sehingga operasi yang panjang (seperti mengunggah file besar) dapat selesai. Untuk informasi selengkapnya, lihat Konfigurasikan batas waktu koneksi idle untuk Classic Load Balancer dan Bagaimana cara mengatasi masalah latensi tinggi Elastic Load Balancing.

Penyebab 2: Instans terdaftar menutup koneksi ke Elastic Load Balancing.

Solusi 2: Aktifkan pengaturan keep-alive pada EC2 instans Anda dan pastikan batas waktu keep-alive lebih besar daripada pengaturan batas waktu idle penyeimbang beban Anda.

Memecahkan masalah Classic Load Balancer: Metrik kode respons

Penyeimbang beban Anda mengirimkan metrik ke Amazon CloudWatch untuk kode respons HTTP yang dikirim ke klien, mengidentifikasi sumber kesalahan sebagai penyeimbang beban atau instans terdaftar. Anda dapat menggunakan metrik yang dikembalikan oleh penyeimbang beban Anda CloudWatch untuk memecahkan masalah. Untuk informasi selengkapnya, lihat CloudWatch metrik untuk Classic Load Balancer Anda.

Berikut ini adalah metrik kode respons yang dikembalikan CloudWatch untuk penyeimbang beban Anda, penyebab potensial, dan langkah-langkah yang dapat Anda ambil untuk menyelesaikan masalah.

Metrik Kode Respons

- HTTPCode_ELB_4XX

- HTTPCode_ELB_5XX
- HTTPCode_Backend_2xx
- HTTPCode_Backend_3xx
- HTTPCode_Backend_4XX
- HTTPCode_Backend_5XX

HTTPCode_ELB_4XX

Penyebab: Permintaan yang cacat atau dibatalkan dari klien.

Solusi

- Lihat HTTP 400: PERMINTAAN BURUK.
- Lihat HTTP 405: METHOD_NOT_ALLOWED.
- Lihat HTTP 408: Waktu habis permintaan.

HTTPCode_ELB_5XX

Penyebab: Baik penyeimbang beban atau instance terdaftar menyebabkan kesalahan atau penyeimbang beban tidak dapat mengurai respons.

Solusi

- Lihat HTTP 502: Gateway buruk.
- Lihat 503 Layanan Tidak Tersedia.
- Lihat HTTP 504: Waktu habis gateway.

HTTPCode_Backend_2xx

Penyebab: Respons yang normal dan berhasil dari instans terdaftar.

Solusi: Tidak ada.

HTTPCode_Backend_3xx

Penyebab: Respons pengalihan yang dikirim dari instans terdaftar.

Solusi: Lihat log akses atau log kesalahan pada instans Anda untuk menentukan penyebabnya. Kirim permintaan langsung ke instance (melewati penyeimbang beban) untuk melihat tanggapan.

HTTPCode_Backend_4XX

Penyebab: Respons kesalahan klien yang dikirim dari instance terdaftar.

Solusi: Lihat akses atau log kesalahan pada instans Anda untuk menentukan penyebabnya. Kirim permintaan langsung ke instance (melewati penyeimbang beban) untuk melihat tanggapan.

> **Note**
>
> Jika klien membatalkan permintaan HTTP yang dimulai dengan `Transfer-Encoding: chunked` header, ada masalah yang diketahui di mana penyeimbang beban meneruskan permintaan ke instance meskipun klien membatalkan permintaan tersebut. Hal ini dapat menyebabkan kesalahan backend.

HTTPCode_Backend_5XX

Penyebab: Respons kesalahan server yang dikirim dari instance terdaftar.

Solusi: Lihat log akses atau log kesalahan pada instance Anda untuk menentukan penyebabnya. Kirim permintaan langsung ke instance (melewati penyeimbang beban) untuk melihat tanggapan.

> **Note**
>
> Jika klien membatalkan permintaan HTTP yang dimulai dengan `Transfer-Encoding: chunked` header, ada masalah yang diketahui di mana penyeimbang beban meneruskan permintaan ke instance meskipun klien membatalkan permintaan tersebut. Hal ini dapat menyebabkan kesalahan backend.

Memecahkan Masalah Classic Load Balancer: Pemeriksaan Kesehatan

Penyeimbang beban Anda memeriksa kesehatan instans yang terdaftar menggunakan konfigurasi pemeriksaan kesehatan default yang disediakan oleh Elastic Load Balancing atau konfigurasi

pemeriksaan kesehatan khusus yang Anda tentukan. Konfigurasi pemeriksaan kesehatan berisi informasi seperti protokol, port ping, jalur ping, batas waktu respons, dan interval pemeriksaan kesehatan. Sebuah instance dianggap sehat jika mengembalikan kode respons 200 dalam interval pemeriksaan kesehatan. Untuk informasi selengkapnya, lihat Health memeriksa instans Classic Load Balancer Anda.

Jika status saat ini dari beberapa atau semua instance Anda `OutOfService` dan bidang deskripsi menampilkan pesan bahwa`Instance has failed at least the Unhealthy Threshold number of health checks consecutively`, instance telah gagal dalam pemeriksaan kesehatan penyeimbang beban. Berikut ini adalah masalah yang harus dicari, penyebab potensial, dan langkah-langkah yang dapat Anda ambil untuk menyelesaikan masalah.

Masalah

- Kesalahan halaman target pemeriksaan kesehatan
- Koneksi ke instance telah habis
- Otentikasi kunci publik gagal
- Instance tidak menerima lalu lintas dari penyeimbang beban
- Port pada instance tidak terbuka
- Instance dalam grup Auto Scaling gagal dalam pemeriksaan kesehatan ELB

Kesalahan halaman target pemeriksaan kesehatan

Masalah: Permintaan HTTP GET yang dikeluarkan untuk instance pada port ping yang ditentukan dan jalur ping (misalnya, http:80/index.html) menerima kode respons non-200.

Penyebab 1: Tidak ada halaman target yang dikonfigurasi pada instance.

Solusi 1: Buat halaman target (misalnya,`index.html`) pada setiap instance terdaftar dan tentukan jalurnya sebagai jalur ping.

Penyebab 2: Nilai header Content-Length dalam respons tidak disetel.

Solusi 2: Jika respons menyertakan badan, maka setel header Content-Length ke nilai yang lebih besar dari atau sama dengan nol, atau atur nilai Transfer-Encoding ke 'chunked'.

Penyebab 3: Aplikasi tidak dikonfigurasi untuk menerima permintaan dari penyeimbang beban atau mengembalikan kode respons 200.

Solusi 3: Periksa aplikasi pada instans Anda untuk menyelidiki penyebabnya.

Koneksi ke instance telah habis

Masalah: Permintaan pemeriksaan kesehatan dari penyeimbang beban Anda ke EC2 instans Anda habis waktu atau gagal sebentar-sebentar.

Pertama, verifikasi masalah dengan menghubungkan langsung dengan instance. Kami menyarankan agar Anda terhubung ke instans Anda dari dalam jaringan menggunakan alamat IP pribadi dari instans.

Gunakan perintah berikut untuk koneksi TCP:

```
telnet private-IP-address-of-the-instance port
```

Gunakan perintah berikut untuk koneksi HTTP atau HTTPS:

```
curl -I private-IP-address-of-the-instance:port/health-check-target-page
```

Jika Anda menggunakan koneksi HTTP/HTTPS dan mendapatkan respons non-200, lihat. Kesalahan halaman target pemeriksaan kesehatan Jika Anda dapat terhubung langsung ke instans, periksa hal-hal berikut:

Penyebab 1: Instance gagal merespons dalam periode batas waktu respons yang dikonfigurasi.

Solusi 1: Sesuaikan pengaturan batas waktu respons dalam konfigurasi pemeriksaan kesehatan penyeimbang beban Anda.

Penyebab 2: Instans berada di bawah beban yang signifikan dan membutuhkan waktu lebih lama dari periode batas waktu respons yang dikonfigurasi untuk merespons.

Solusi 2:

- Periksa grafik pemantauan untuk pemanfaatan CPU yang berlebihan. Untuk selengkapnya, lihat Mendapatkan statistik untuk EC2 instans tertentu di Panduan EC2 Pengguna Amazon.
- Periksa pemanfaatan sumber daya aplikasi lain, seperti memori atau batas, dengan menghubungkan ke EC2 instance Anda.
- Jika perlu, tambahkan lebih banyak instance atau aktifkan Auto Scaling. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon EC2 Auto Scaling.

Penyebab 3: Jika Anda menggunakan koneksi HTTP atau HTTPS dan pemeriksaan kesehatan sedang dilakukan pada halaman target yang ditentukan dalam bidang jalur ping (misalnya,`HTTP:80/index.html`), halaman target mungkin membutuhkan waktu lebih lama untuk merespons daripada batas waktu yang dikonfigurasi.

Solusi 3: Gunakan halaman target pemeriksaan kesehatan yang lebih sederhana atau sesuaikan pengaturan interval pemeriksaan kesehatan.

Otentikasi kunci publik gagal

Masalah: Penyeimbang beban yang dikonfigurasi untuk menggunakan protokol HTTPS atau SSL dengan otentikasi back-end diaktifkan gagal otentikasi kunci publik.

Penyebab: Kunci publik pada sertifikat SSL tidak cocok dengan kunci publik yang dikonfigurasi pada penyeimbang beban. Gunakan `s_client` perintah untuk melihat daftar sertifikat server dalam rantai sertifikat. Untuk informasi selengkapnya, lihat s_client di dokumentasi OpenSSL.

Solusi: Anda mungkin perlu memperbarui sertifikat SSL Anda. Jika sertifikat SSL Anda saat ini, coba instal ulang pada penyeimbang beban Anda. Untuk informasi selengkapnya, lihat Ganti sertifikat SSL untuk Classic Load Balancer Anda.

Instance tidak menerima lalu lintas dari penyeimbang beban

Masalah: Grup keamanan misalnya memblokir lalu lintas dari penyeimbang beban.

Lakukan pengambilan paket pada instance untuk memverifikasi masalah. Gunakan perintah berikut ini.

```
# tcpdump port health-check-port
```

Penyebab 1: Grup keamanan yang terkait dengan instance tidak mengizinkan lalu lintas dari penyeimbang beban.

Solusi 1: Edit grup keamanan instans untuk mengizinkan lalu lintas dari penyeimbang beban. Tambahkan aturan untuk mengizinkan semua lalu lintas dari grup keamanan penyeimbang beban.

Penyebab 2: Grup keamanan untuk penyeimbang beban Anda tidak mengizinkan lalu lintas ke EC2 instans.

Solusi 2: Edit grup keamanan penyeimbang beban Anda untuk memungkinkan lalu lintas ke subnet dan instans. EC2

Untuk informasi tentang mengelola grup keamanan, lihatKonfigurasikan grup keamanan untuk Classic Load Balancer Anda.

Port pada instance tidak terbuka

Masalah: Pemeriksaan kesehatan yang dikirim ke EC2 instance oleh penyeimbang beban diblokir oleh port atau firewall.

Verifikasi masalah dengan menggunakan perintah berikut:

```
netstat –ant
```

Penyebab: Port kesehatan yang ditentukan atau port pendengar (jika dikonfigurasi berbeda) tidak terbuka. Baik port yang ditentukan untuk pemeriksaan kesehatan dan port pendengar harus terbuka dan mendengarkan.

Solusi: Buka port listener dan port yang ditentukan dalam konfigurasi pemeriksaan kesehatan Anda (jika dikonfigurasi berbeda) pada instans Anda untuk menerima lalu lintas penyeimbang beban.

Instance dalam grup Auto Scaling gagal dalam pemeriksaan kesehatan ELB

Masalah: Instance di grup Auto Scaling Anda lulus pemeriksaan kesehatan Auto Scaling default tetapi gagal dalam pemeriksaan kesehatan ELB.

Penyebab: Auto Scaling menggunakan pemeriksaan EC2 status untuk mendeteksi masalah perangkat keras dan perangkat lunak dengan instans, tetapi penyeimbang beban melakukan pemeriksaan kesehatan dengan mengirimkan permintaan ke instance dan menunggu kode respons 200, atau dengan membuat koneksi TCP (untuk pemeriksaan kesehatan berbasis TCP) dengan instance.

Sebuah instance mungkin gagal dalam pemeriksaan kesehatan ELB karena aplikasi yang berjalan pada instance memiliki masalah yang menyebabkan penyeimbang beban mempertimbangkan instance di luar layanan. Instance ini mungkin lulus pemeriksaan kesehatan Auto Scaling; itu tidak akan digantikan oleh kebijakan Auto Scaling karena dianggap sehat berdasarkan EC2 pemeriksaan status.

Solusi: Gunakan pemeriksaan kesehatan ELB untuk grup Auto Scaling Anda. Saat Anda menggunakan pemeriksaan kesehatan ELB, Auto Scaling menentukan status kesehatan instans Anda dengan memeriksa hasil pemeriksaan status instans dan pemeriksaan kesehatan ELB. Untuk

informasi selengkapnya, lihat Pemeriksaan kesehatan Menambahkan Elastic Load Balancing ke grup Auto Scaling di Panduan Pengguna Amazon Auto EC2 Scaling.

Memecahkan Masalah Classic Load Balancer: Konektivitas klien

Klien tidak dapat menyambung ke Load Balancer yang menghadap internet

Jika Load Balancer tidak merespons permintaan, periksa masalah berikut ini:

Load Balancer yang menghadap internet Anda terpasang ke subnet pribadi

Anda harus menentukan subnet publik untuk Load Balancer Anda. Subnet publik memiliki rute ke gateway internet untuk virtual private cloud (VPC) Anda.

Grup keamanan atau jaringan ACL tidak mengizinkan lalu lintas

Grup keamanan untuk penyeimbang beban dan jaringan apa pun ACLs untuk subnet penyeimbang beban harus memungkinkan lalu lintas masuk dari klien dan lalu lintas keluar ke klien di port pendengar. Untuk informasi selengkapnya, lihat Konfigurasikan grup keamanan untuk Classic Load Balancer Anda.

Permintaan yang dikirim ke domain kustom tidak diterima oleh penyeimbang beban

Jika penyeimbang beban tidak menerima permintaan yang dikirim ke domain kustom, periksa masalah berikut:

Nama domain kustom tidak diselesaikan ke alamat IP penyeimbang beban

- Konfirmasikan alamat IP apa yang diselesaikan oleh nama domain khusus untuk menggunakan antarmuka baris perintah.
  - Linux, macOS, atau Unix — Anda dapat menggunakan `dig` perintah di dalam Terminal. Mantan. `dig example.com`
  - Windows — Anda dapat menggunakan `nslookup` perintah dalam Command Prompt. Mantan. `nslookup example.com`
- Konfirmasikan alamat IP apa yang diselesaikan oleh nama DNS penyeimbang beban untuk menggunakan antarmuka baris perintah.
- Bandingkan hasil dari dua output. Alamat IP harus cocok.

Permintaan HTTPS yang dikirim ke penyeimbang beban mengembalikan “NET: :ERR_CERT_COMMON_NAME_INVALID”

Jika permintaan HTTPS diterima `NET::ERR_CERT_COMMON_NAME_INVALID` dari penyeimbang beban, periksa kemungkinan penyebab berikut:

- Nama domain yang digunakan dalam permintaan HTTPS tidak cocok dengan nama alternatif yang ditentukan dalam sertifikat ACM terkait pendengar.
- Nama DNS default load balancers sedang digunakan. Nama DNS default tidak dapat digunakan untuk membuat permintaan HTTPS karena sertifikat publik tidak dapat diminta untuk `*.amazonaws.com` domain.

Memecahkan masalah Classic Load Balancer: Pendaftaran instans

Saat Anda mendaftarkan instans dengan penyeimbang beban Anda, ada sejumlah langkah yang diambil sebelum penyeimbang beban dapat mulai mengirim permintaan ke instans Anda.

Berikut ini adalah masalah yang mungkin dihadapi penyeimbang beban Anda saat mendaftarkan EC2 instans Anda, penyebab potensial, dan langkah-langkah yang dapat Anda ambil untuk menyelesaikan masalah.

Masalah

- Terlalu lama untuk mendaftarkan sebuah EC2 instans
- Tidak dapat mendaftarkan instance yang diluncurkan dari AMI berbayar

Terlalu lama untuk mendaftarkan sebuah EC2 instans

Masalah: EC2 Instans terdaftar membutuhkan waktu lebih lama dari yang diharapkan berada di `InService` negara bagian.

Penyebab: Instance Anda mungkin gagal dalam pemeriksaan kesehatan. Setelah langkah pendaftaran instans awal selesai (dapat memakan waktu hingga sekitar 30 detik), penyeimbang beban mulai mengirimkan permintaan pemeriksaan kesehatan. Contoh Anda tidak `InService` sampai satu pemeriksaan kesehatan berhasil.

Solusi: LihatKoneksi ke instance telah habis.

Tidak dapat mendaftarkan instance yang diluncurkan dari AMI berbayar

Masalah: Elastic Load Balancing tidak mendaftarkan instans yang diluncurkan menggunakan AMI berbayar.

Penyebab: Instance Anda mungkin telah diluncurkan menggunakan AMI berbayar dari Amazon DevPay.

Solusi: Elastic Load Balancing tidak mendukung pendaftaran instans yang diluncurkan menggunakan berbayar dari AMIs Amazon. DevPay Perhatikan bahwa Anda dapat menggunakan berbayar AMIs dari AWS Marketplace. Jika Anda sudah menggunakan AMI berbayar dari AWS Marketplace dan tidak dapat mendaftarkan instans yang diluncurkan dari AMI berbayar itu, kunjungi AWS Dukungan Pusat untuk bantuan.

Kuota untuk Classic Load Balancer Anda

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

Untuk melihat kuota untuk Classic Load Balancers Anda, buka konsol Service Quotas. Di panel navigasi, pilih AWS layanan dan pilih Elastic Load Balancing. Anda juga dapat menggunakan perintah describe-account-limits(AWS CLI) untuk Elastic Load Balancing.

Untuk meminta penambahan kuota, lihat Meminta penambahan kuota di Panduan Pengguna Service Quotas.

AWS Akun Anda memiliki kuota berikut yang terkait dengan Classic Load Balancers.

Nama	Default	Dapat disesuaikan
Classic Load Balancer per Wilayah	20	Ya
Listener per Classic Load Balancer	100	Ya
Instans Terdaftar per Classic Load Balancer	1.000	Ya

Riwayat dokumen untuk Classic Load Balancers

Tabel berikut menjelaskan rilis untuk Classic Load Balancers.

Perubahan	Deskripsi	Tanggal
Mode mitigasi desync	Menambahkan dukungan untuk modus mitigasi desync. Untuk informasi selengkapnya, lihat Mengonfigurasi mode mitigasi desync untuk Classic Load Balancer Anda.	17 Agustus 2020
Penyeimbang Beban Klasik	Dengan diperkenalkannya Application Load Balancers dan Network Load Balancers , load balancer yang dibuat dengan API 2016-06-01 sekarang dikenal sebagai Classic Load Balancers. Untuk informasi selengkap nya tentang perbedaan antara jenis load balancer ini, lihat fitur Elastic Load Balancing.	11 Agustus, 2016
Support untuk AWS Certificate Manager (ACM)	Anda dapat meminta sertifikat SSL/TLS dari ACM dan menyebarkannya ke penyeimbang beban Anda. Untuk informasi selengkapnya, lihat sertifikat SSL/TLS untuk Classic Load Balancer.	21 Januari 2016
Support untuk port tambahan	Load balancer dapat mendengarkan pada port apa pun dalam kisaran 1-65535. Untuk informasi selengkapnya,	15 September 2015

	lihat Pendengar untuk Classic Load Balancer Anda.	
Bidang tambahan untuk entri log akses	Ditambahkan `user_agent` , `ssl_cipher` , dan `ssl_protocol` bidang. Untuk informasi selengkapnya, lihat Mengakses file log.	18 Mei 2015
Support untuk menandai penyeimbang beban Anda	Dimulai dengan rilis ini, Elastic Load Balancing CLI (ELB CLI) telah digantikan oleh AWS Command Line Interface (AWS CLI), alat terpadu untuk mengelola beberapa layanan. AWS Fitur baru yang dirilis setelah ELB CLI versi 1.0.35.0 (tanggal 7/24/14) akan dimasukkan dalam satu-satunya. AWS CLI Jika saat ini Anda menggunakan ELB CLI, kami sarankan Anda mulai menggunakan AWS CLI sebagai gantinya. Untuk informasi selengkapnya, lihat Panduan Pengguna AWS Command Line Interface .	11 Agustus 2014
Batas waktu koneksi idle	Anda dapat mengonfigurasi batas waktu koneksi idle untuk penyeimbang beban Anda.	Juli 24, 2014
Support untuk memberikan pengguna dan grup akses ke load balancer atau tindakan API tertentu	Anda dapat membuat kebijakan untuk memberi pengguna dan grup akses ke penyeimbang beban atau tindakan API tertentu.	12 Mei 2014

Support untuk AWS CloudTrail	Anda dapat menggunakan CloudTrail untuk menangkap panggilan API yang dilakukan oleh atau atas nama Anda Akun AWS menggunakan ELB API, CLI ELB AWS Management Console, atau. AWS CLI	April 4, 2014
Pengeringan koneksi	Menambahkan informasi tentang pengeringan koneksi. Dengan dukungan ini, Anda dapat mengaktifkan penyeimbang beban untuk berhenti mengirim permintaan baru ke instans terdaftar saat instans tidak mendaftar atau saat instance menjadi tidak sehat, sambil menjaga koneksi yang ada tetap terbuka. Untuk informasi selengkapnya, lihat Mengonfigurasi pengurasan koneksi untuk Classic Load Balancer Anda.	20 Maret 2014
Akses log	Anda dapat mengaktifkan penyeimbang beban untuk menangkap informasi terperinci tentang permintaan yang dikirim ke penyeimbang beban dan menyimpannya di bucket Amazon S3. Untuk informasi selengkapnya, lihat Akses log untuk Classic Load Balancer Anda.	6 Maret 2014

Support untuk TLSv1 .1-1.2	Menambahkan informasi tentang dukungan protokol TLSv1 .1-1.2 untuk penyeimbang beban yang dikonfigurasi dengan pendengar HTTPS/SSL. Dengan dukungan ini, Elastic Load Balancing juga memperbarui konfigurasi negosiasi SSL yang telah ditentukan sebelumnya. Untuk informasi tentang konfigurasi negosiasi SSL yang telah ditentukan sebelumnya yang diperbarui, lihat konfigurasi negosiasi SSL untuk Classic Load Balancer. Untuk informasi tentang memperbarui konfigurasi negosiasi SSL Anda saat ini, lihat Memperbarui konfigurasi negosiasi SSL Classic Load Balancer Anda.	Februari 19, 2014
Penyeimbangan beban lintas zona	Menambahkan informasi tentang mengaktifkan penyeimbangan beban lintas zona untuk penyeimbang beban Anda. Untuk informasi selengkapnya, lihat Mengonfigurasi penyeimbangan beban lintas zona untuk Classic Load Balancer Anda.	6 November 2013

CloudWatch Metrik Tambahan	Menambahkan informasi tentang metrik Cloudwatch tambahan yang dilaporkan oleh Elastic Load Balancing. Untuk informasi selengkapnya, lihat CloudWatch metrik untuk Classic Load Balancer Anda.	Oktober 28, 2013
Support untuk protokol proxy	Menambahkan informasi tentang dukungan protokol proxy untuk penyeimbang beban yang dikonfigurasi untuk koneksi TCP/SSL. Untuk informasi selengkapnya, lihat Header protokol proxy.	Juli 30, 2013
Support untuk failover DNS	Menambahkan informasi tentang mengonfigurasi failover DNS Amazon Route 53 untuk penyeimbang beban. Untuk informasi selengkapnya, lihat Menggunakan failover DNS Amazon Route 53 untuk penyeimbang beban Anda.	3 Juni 2013
Dukungan konsol untuk melihat CloudWatch metrik dan membuat alarm	Menambahkan informasi tentang melihat CloudWatc h metrik dan membuat alarm untuk penyeimbang beban tertentu menggunakan konsol. Untuk informasi selengkapnya, lihat CloudWatch metrik untuk Classic Load Balancer Anda.	28 Maret 2013
Support untuk mendaftarkan EC2 instance di VPC default	Menambahkan dukungan untuk EC2 instance yang diluncurkan di VPC default.	11 Maret 2013

Penyeimbang beban internal	Dengan rilis ini, load balancer di virtual private cloud (VPC) dapat dibuat baik internal maupun internet-facing. Penyeimbang beban internal memiliki nama DNS yang dapat diselesaikan secara publik yang dapat diselesai kan ke alamat IP pribadi. Load balancer yang menghadap ke internet memiliki nama DNS yang dapat diselesai kan secara publik yang dapat diselesaikan ke alamat IP publik. Untuk informasi selengkapnya, lihat Membuat Classic Load Balancer internal.	10 Juni 2012
Dukungan konsol untuk mengelola pendengar, pengaturan sandi, dan sertifika t SSL	Untuk selengkapnya, lihat Mengonfigurasi listener HTTPS untuk Classic Load Balancer dan Ganti sertifika t SSL untuk Classic Load Balancer Anda.	18 Mei 2012
Support untuk Elastic Load Balancing di Amazon VPC	Menambahkan dukungan untuk membuat penyeimbang beban di cloud pribadi virtual (VPC).	November 21, 2011
Amazon CloudWatch	Anda dapat memantau penyeimbang beban Anda menggunakan CloudWatch. Untuk informasi selengkapnya, lihat CloudWatch metrik untuk Classic Load Balancer Anda.	17 Oktober 2011

Fitur keamanan tambahan	Anda dapat mengonfig urasi cipher SSL, SSL back-end, dan otentikasi server back-end. Untuk informasi selengkapnya, lihat Membuat Classic Load Balancer dengan pendengar HTTPS.	Agustus 30, 2011
Nama domain puncak zona	Untuk informasi selengkapnya, lihat Mengonfigurasi nama domain khusus untuk Classic Load Balancer Anda.	24 Mei 2011
Support untuk X-Forwarded-Proto dan X-Forwarded-Port header	X-Forwarded-ProtoHeader menunjukkan protokol permintaan asal, dan X-Forwarded-Port header menunjukkan port permintaan asal. Penambahan header ini ke permintaan memungkinkan pelanggan untuk menentukan apakah permintaan masuk ke penyeimbang beban mereka dienkripsi, dan port spesifik pada penyeimbang beban tempat permintaan diterima. Untuk informasi selengkapnya, lihat header HTTP dan Classic Load Balancer.	27 Oktober 2010

Support untuk HTTPS	Dengan rilis ini, Anda dapat memanfaatkan protokol SSL/TLS untuk mengenkripsi lalu lintas dan membongkar pemrosesan SSL dari instance aplikasi ke penyeimbang beban. Fitur ini juga menyediakan manajemen terpusat sertifikat server SSL di load balancer, daripada mengelola sertifikat pada instance aplikasi individual.	14 Oktober 2010
Support untuk AWS Identity and Access Management (IAM)	Dukungan tambahan untuk IAM.	2 September 2010
Sesi lengket	Untuk informasi selengkapnya, lihat Mengonfigurasi sesi lengket untuk Classic Load Balancer Anda.	April 7, 2010
AWS SDK untuk Java	Menambahkan dukungan untuk SDK for Java.	Maret 22, 2010
AWS SDK untuk .NET	Menambahkan dukungan untuk SDK untuk .NET.	11 November 2009
Layanan baru	Rilis beta publik awal Elastic Load Balancing.	18 Mei 2009

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