Laporan Resmi AWS

Komunikasi Waktu Nyata di AWS

Komunikasi Waktu Nyata di AWS: Laporan Resmi AWS

Table of Contents

Komunikasi Waktu Nyata di AWS

Praktik Terbaik untuk Merancang Beban Kerja Komunikasi Waktu Nyata (RTC) yang Sangat Tersedia dan Dapat Diskalakan di AWS.

Tanggal publikasi: 13 Februari 2020 (Revisi Dokumen)

Abstrak

Saat ini, banyak organisasi mencari cara untuk mengurangi biaya dan mencapai skalabilitas untuk beban kerja suara, olahpesan, dan multimedia waktu nyata. Laporan ini menguraikan praktik terbaik untuk mengelola beban kerja komunikasi waktu nyata di AWS dan mencakup arsitektur referensi untuk memenuhi persyaratan ini. Laporan ini berfungsi sebagai panduan bagi individu yang familier dengan komunikasi waktu nyata tentang cara mencapai ketersediaan dan skalabilitas tinggi untuk beban kerja ini.

Pendahuluan

Aplikasi telekomunikasi menggunakan suara, video, dan olahpesan sebagai saluran merupakan persyaratan utama bagi banyak organisasi dan pengguna akhir mereka. Beban kerja komunikasi waktu nyata (RTC) ini memiliki persyaratan latensi dan ketersediaan khusus yang dapat dipenuhi dengan mengikuti praktik terbaik desain yang relevan. Di masa lalu, beban kerja RTC telah di-deploy di pusat data on-premise tradisional dengan sumber daya khusus.

Namun, karena serangkaian fitur yang matang dan berkembang, beban kerja RTC dapat di-deploy di Amazon Web Services (AWS) meskipun persyaratan tingkat layanannya ketat serta juga mendapat manfaat dari skalabilitas, elastisitas, dan ketersediaan tinggi. Saat ini, beberapa pelanggan menggunakan AWS, partner-nya, dan solusi sumber terbuka untuk menjalankan beban kerja RTC dengan biaya berkurang, ketangkasan yang lebih cepat, kemampuan untuk pergi global dalam hitungan menit, dan fitur kaya dari layanan AWS.

Pelanggan memanfaatkan fitur AWS seperti jaringan yang disempurnakan dengan Elastic Network Adapter (ENA) dan generasi terbaru instans Amazon Elastic Compute Cloud (EC2) untuk mendapatkan manfaat dari kit pengembangan bidang data (DPDK), virtualisasi I/O root tunggal (SR- IOV), halaman besar, dukungan NVM Express (NVMe), akses memori non-seragam (NUMA) serta instans bare metal untuk memenuhi persyaratan beban kerja RTC. Instans ini menawarkan bandwidth jaringan hingga 100 Gbps dan paket yang sepadan per detik, memberikan peningkatan performa untuk aplikasi intensif jaringan. Untuk penskalaan, Elastic Load Balancing menawarkan Application Load Balancer, yang menawarkan dukungan WebSocket dan Network Load Balancer yang dapat menangani jutaan permintaan per detik. Untuk akselerasi jaringan, AWS Global Accelerator menyediakan alamat IP statis yang bertindak sebagai titik masuk tetap ke titik akhir aplikasi Anda di AWS. Ini memiliki dukungan untuk alamat IP statis untuk penyeimbang beban. Untuk mengurangi latensi, biaya, dan peningkatan throughput bandwidth, AWS Direct Connect menetapkan koneksi jaringan khusus dari on-premise ke AWS. Trunking SIP terkelola yang sangat tersedia disediakan oleh Amazon Chime Voice Connector. Amazon Kinesis Video Streams dengan WebRTC dengan mudah streaming media dua arah waktu nyata dengan ketersediaan tinggi.

Laporan ini mencakup arsitektur referensi yang menunjukkan cara mengatur beban kerja RTC di AWS dan praktik terbaik untuk mengoptimalkan solusi guna memenuhi persyaratan pengguna akhir sambil mengoptimalkan cloud. Evolved packet core (EPC) berada di luar cakupan untuk laporan resmi ini, tetapi praktik terbaik yang terperinci dapat diterapkan pada fungsi jaringan virtual (VNF).

Komponen Dasar Arsitektur RTC

Di industri telekomunikasi, komunikasi waktu nyata (RTC) umumnya mengacu pada sesi media langsung antara dua titik akhir dengan latensi minimum. Sesi ini dapat dikaitkan dengan:

- Sesi suara antara dua pihak (misalnya, sistem telepon, ponsel, VoIP)
- Olahpesan instan (misalnya, obrolan, IRC)
- Sesi video langsung (misalnya, konferensi video, telepresence)

Masing-masing solusi sebelumnya memiliki beberapa komponen yang sama (misalnya, komponen yang menyediakan autentikasi, otorisasi dan kontrol akses, transcoding, buffering dan relay, dan sebagainya) dan beberapa komponen unik untuk jenis media yang dikirimkan (misalnya, layanan siaran, server olahpesan dan antrean, dan sebagainya). Bagian ini berfokus pada mendefinisikan sistem RTC berbasis suara dan video dan semua komponen terkait yang diilustrasikan pada Gambar 1.

Gambar 1: Komponen arsitektur penting untuk RTC

Topik

SoftSwitch/PBX

Softswitch atau PBX adalah otak dari sistem telepon suara dan menyediakan kecerdasan untuk membangun, memelihara, dan perutean panggilan suara dalam atau di luar perusahaan dengan menggunakan komponen yang berbeda. Semua pelanggan perusahaan diharuskan mendaftar dengan softswitch untuk menerima atau melakukan panggilan. Fungsi penting dari softswitch adalah untuk melacak setiap pelanggan dan bagaimana menjangkau mereka dengan menggunakan komponen lain dalam jaringan suara.

Session Border Controller (SBC)

Session border controller (SBC) berada di tepi jaringan suara dan melacak semua lalu lintas masuk dan keluar (baik kontrol dan bidang data). Salah satu tanggung jawab utama SBC adalah melindungi sistem suara dari penggunaan berbahaya. SBC dapat digunakan untuk interkoneksi dengan batang protokol inisiasi sesi (SIP) untuk konektivitas eksternal. Beberapa SBC juga menyediakan kemampuan transcoding untuk mengonversi CODECS dari satu format ke format lainnya. Akhirnya, sebagian besar SBC juga menyediakan kemampuan NAT Traversal yang membantu dalam memastikan panggilan dibuat, bahkan di seluruh jaringan firewall.

Konektivitas PSTN

Solusi Voice over IP (VoIP) menggunakan PSTN Gateway dan SIP Trunk untuk terhubung dengan jaringan PSTN warisan.

Gateway Gerbang PSTN

Jaringan telepon switch publik (PSTN) Gateway mengonversi sinyal (antara SIP dan SS7) dan media (antara RTP dan multiplexing pembagian waktu [TDM] menggunakan transcoding CODEC). Gateway PSTN selalu berada di edge dekat dengan jaringan PSTN.

SIP Trunk

Dalam SIP Trunk, perusahaan tidak mengakhiri panggilannya ke jaringan TDM (berbasis SS7), melainkan alur antara perusahaan dan telekomunikasi tetap melalui IP. Sebagian besar SIP Trunk ditetapkan dengan menggunakan SBC. Perusahaan harus menyetujui aturan keamanan yang telah ditetapkan dari telekomunikasi, seperti mengizinkan berbagai alamat IP, port, dan sebagainya.

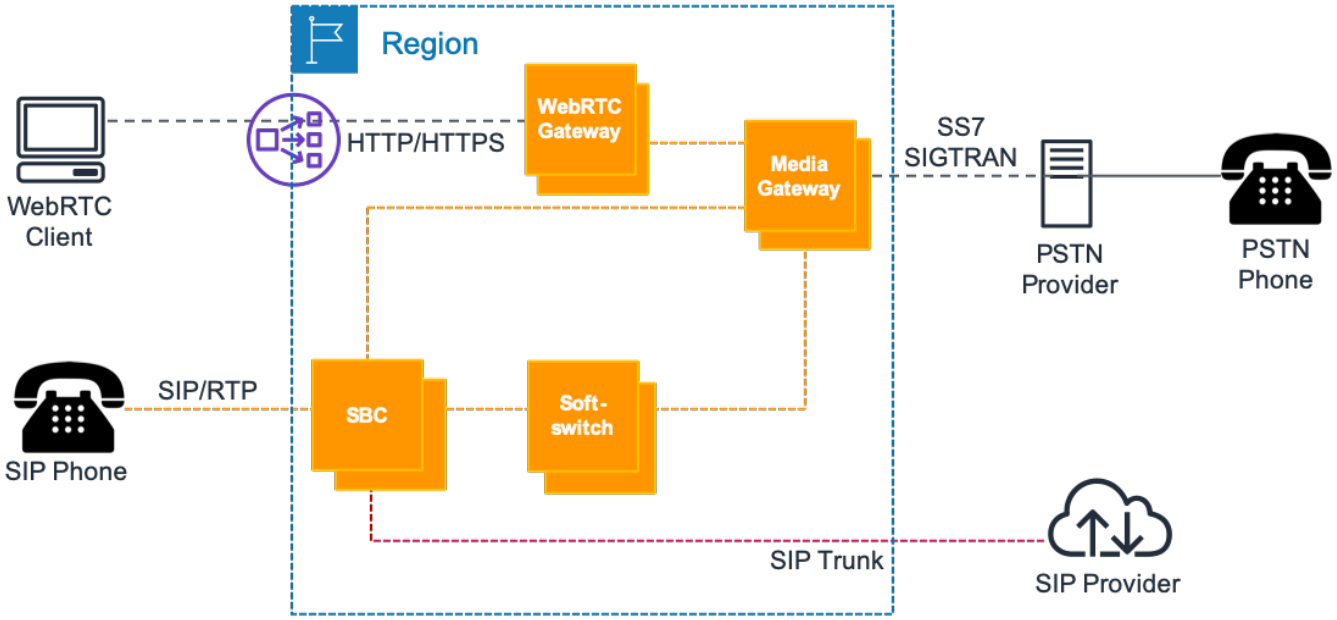
Media Gateway (Transcoder)

Solusi suara khas memungkinkan berbagai jenis CODEC. Beberapa Codec umum adalah G.711 μ-law untuk Amerika Utara, G.711 A-law untuk luar Amerika Utara, G.729, dan G.722. Ketika dua perangkat yang menggunakan dua codec berbeda berkomunikasi satu sama lain, server media menerjemahkan aliran CODEC antara perangkat. Dengan kata lain, media gateway memproses media dan memastikan bahwa perangkat akhir dapat berkomunikasi satu sama lain.

WebRTC dan WebRTC gateway

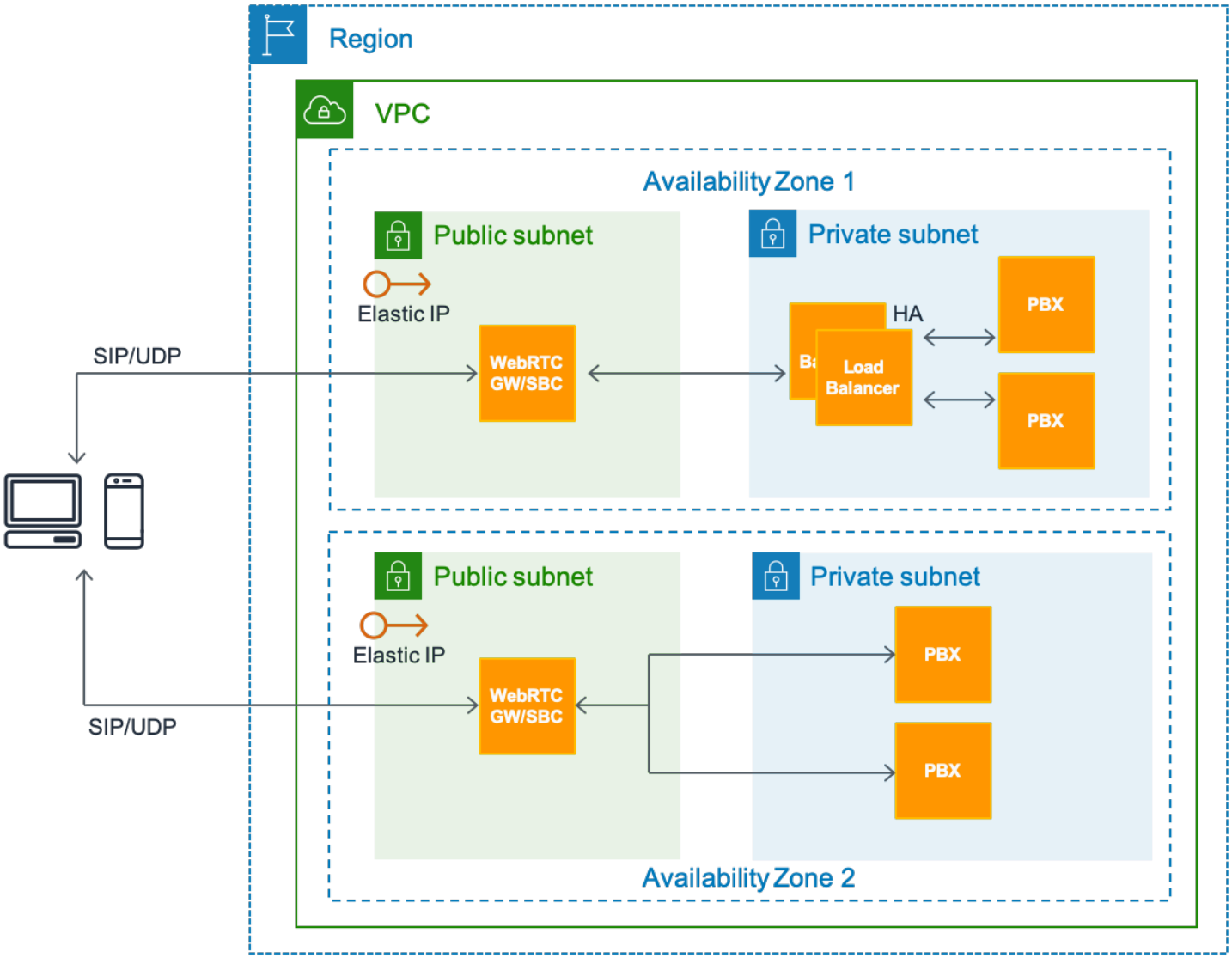
Komunikasi web waktu nyata (WebRTC) memungkinkan Anda untuk membuat panggilan dari peramban web atau meminta sumber daya dari server backend dengan menggunakan API. Teknologi ini dirancang dengan teknologi cloud dalam pikiran dan karena itu menyediakan berbagai API yang dapat digunakan untuk membuat panggilan. Karena tidak semua solusi suara (termasuk SIP) mendukung API ini, gateway WebRTC diperlukan untuk menerjemahkan panggilan API ke dalam pesan SIP dan sebaliknya.

Gambar 2 menunjukkan pola desain untuk arsitektur WebRTC yang sangat tersedia. Lalu lintas masuk dari klien WebRTC diimbangi dengan penyeimbang beban aplikasi Amazon dengan WebRTC yang berjalan pada instans EC2 yang merupakan bagian dari Auto Scaling Group.



Gambar 2: Topologi dasar sistem RTC untuk suara

Pola desain lain untuk lalu lintas SIP dan RTP adalah menggunakan pasang SBC di Amazon EC2 dalam mode pasif aktif di Zona Ketersediaan (Gambar 3). Di sini, alamat IP Elastis dapat dipindahkan secara dinamis antara instans setelah kegagalan tempat DNS tidak dapat digunakan.



Gambar 3: Arsitektur RTC menggunakan Amazon EC2 dalam VPC

Ketersediaan dan Skalabilitas Tinggi di AWS

Sebagian besar penyedia komunikasi waktu nyata selaras dengan tingkat layanan yang menyediakan ketersediaan dari 99,9% hingga 99,999%. Bergantung pada tingkat ketersediaan tinggi (HA) yang Anda inginkan, Anda harus mengambil tindakan yang semakin canggih sepanjang siklus hidup penuh aplikasi. Kami sarankan mengikuti panduan ini untuk mencapai tingkat ketersediaan tinggi yang kuat:

- Rancang sistem agar tidak memiliki satu titik kegagalan. Gunakan pemantauan otomatis, deteksi kegagalan, dan mekanisme failover untuk komponen stateless dan stateful

- Titik kegagalan tunggal (SPOF) umumnya dihilangkan dengan konfigurasi redundansi N+1 atau 2N, di mana N+1 dicapai melalui penyeimbangan beban antara simpul aktif—aktif, dan 2N dicapai dengan sepasang simpul dalam konfigurasi aktif—siaga .
- AWS memiliki beberapa metode untuk mencapai HA melalui kedua pendekatan, seperti melalui klaster yang dapat diskalakan, dengan beban diseimbangkan, atau mengasumsikan pasangan aktif —siaga .

- Siapkan instrumen dan ketersediaan sistem pengujian
- Siapkan prosedur operasi untuk mekanisme manual untuk menanggapi, mengurangi, dan pulih dari kegagalan.

Bagian ini berfokus pada cara mencapai tidak ada satu titik kegagalan menggunakan kemampuan yang tersedia di AWS. Secara khusus, bagian ini menjelaskan subset kemampuan AWS inti dan pola desain yang memungkinkan Anda membangun aplikasi komunikasi real-time yang sangat tersedia di platform.

Topik

- Trunking SIP yang Sangat Tersedia dengan Konektor Suara Amazon Chime

Pola Floating IP untuk HA Antara Server Stateful Active—Standby

Pola desain Floating IP adalah mekanisme yang terkenal untuk mencapai failover otomatis antara sepasang simpul perangkat keras yang aktif dan siaga (server media). Alamat IP virtual sekunder statis ditugaskan ke simpul aktif. Pemantauan terus menerus antara simpul aktif dan siaga mendeteksi kegagalan. Jika simpul aktif gagal, skrip pemantauan memberikan IP virtual ke simpul siaga siap dan simpul siaga mengambil alih fungsi aktif utama. Dengan cara ini, IP virtual mengapung antara simpul aktif dan siaga.

Topik

- Penerapan dalam solusi RTC
- Implementasi AMOS di AWS
- Manfaat
- Keterbatasan dan ekstensibilitas

Penerapan dalam solusi RTC

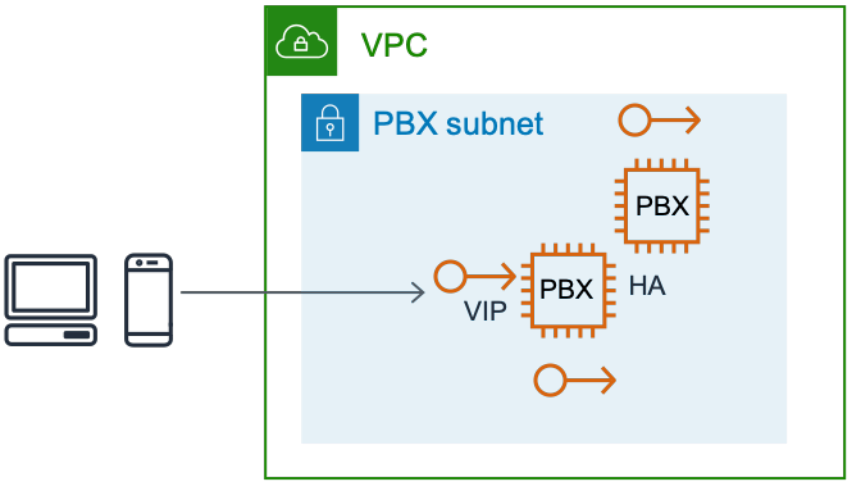
Hal ini tidak selalu mungkin untuk memiliki beberapa contoh aktif dari komponen yang sama dalam layanan, seperti klaster aktif–aktif simpul N. Konfigurasi siaga aktif menyediakan mekanisme terbaik untuk HA. Misalnya, komponen stateful dalam solusi RTC, seperti server media atau server konferensi, atau bahkan server SBC atau database, sangat cocok untuk pengaturan siaga aktif. Server SBC atau media memiliki beberapa sesi berjalan lama atau saluran yang aktif pada waktu tertentu, dan dalam kasus instans aktif SBC gagal, titik akhir dapat menyambung kembali ke simpul siaga tanpa konfigurasi sisi klien karena IP mengambang.

Implementasi AMOS di AWS

Anda dapat menerapkan pola ini di AWS menggunakan kemampuan inti di Amazon Elastic Compute Cloud (Amazon EC2), API Amazon EC2, alamat IP Elastis, dan dukungan di Amazon EC2 untuk alamat IP privat sekunder.

1. Luncurkan dua instans EC2 untuk mengasumsikan peran simpul primer dan sekunder, di mana primer diasumsikan berada dalam keadaan aktif secara default.
2. Tetapkan alamat IP privat sekunder tambahan ke instans EC2 primer.

3. Alamat IP Elastis, yang mirip dengan IP virtual (VIP), dikaitkan dengan alamat privat sekunder. Alamat privat sekunder ini adalah alamat yang digunakan oleh endpoint eksternal untuk mengakses aplikasi.
4. Beberapa konfigurasi OS diperlukan untuk membuat alamat IP sekunder ditambahkan sebagai alias ke antarmuka jaringan utama.
5. Aplikasi harus mengikat alamat IP Elastic ini. Dalam kasus perangkat lunak Asterisk, Anda dapat mengonfigurasi pengikatan melalui pengaturan SIP Asterisk lanjutan.
6. Jalankan skrip pemantauan—kustom, KeepAlive di Linux, Corosync, dan sebagainya—pada setiap simpul untuk memantau keadaan simpul peer. Dalam hal ini, bahwa simpul aktif saat ini gagal, rekan mendeteksi kegagalan ini, dan memanggil API Amazon EC2 untuk menetapkan kembali alamat IP privat sekunder ke dirinya sendiri.
7. Oleh karena itu, aplikasi yang mendengarkan di VIP yang terkait dengan alamat IP privat sekunder tersedia untuk titik akhir melalui simpul siaga.



Gambar 4: Failover antara instans EC2 stateful menggunakan alamat IP Elastic

Manfaat

Pendekatan ini adalah solusi anggaran rendah yang andal yang melindungi dari kegagalan pada instans, infrastruktur, atau tingkat aplikasi EC2.

Keterbatasan dan ekstensibilitas

Pola desain ini biasanya terbatas pada dalam Zona Ketersediaan tunggal. Hal ini dapat diimplementasikan di dua Zona Ketersediaan tetapi dengan variasi. Dalam hal ini, alamat Floating

IP Elastis kembali terkait antara simpul aktif dan siaga di Zona Ketersediaan yang berbeda melalui API alamat IP elastis re-associate yang tersedia. Dalam implementasi failover yang ditunjukkan pada Gambar 4, panggilan yang sedang berlangsung dijatuhkan dan titik akhir harus terhubung kembali. Hal ini dimungkinkan untuk memperluas implementasi ini dengan replikasi data sesi yang mendasari untuk memberikan failover mulus sesi atau kontinuitas media juga.

Penyeimbangan Beban untuk Skalabilitas dan HA dengan WebRTC dan SIP

Penyeimbangan Beban klaster instans aktif berdasarkan aturan yang telah ditetapkan, seperti round robin, afinitas atau latensi, dan sebagainya, adalah pola desain yang dipopulerkan secara luas oleh sifat stateless permintaan HTTP. Bahkan, penyeimbangan beban adalah pilihan yang layak dalam kasus banyak komponen aplikasi RTC.

Penyeimbang beban bertindak sebagai proksi terbalik atau titik masuk untuk permintaan ke aplikasi yang diinginkan, yang dengan sendirinya dikonfigurasi untuk berjalan di beberapa simpul aktif secara bersamaan. Pada titik waktu tertentu, penyeimbang beban mengarahkan permintaan pengguna ke salah satu simpul aktif dalam klaster yang ditentukan. Penyeimbang beban melakukan pemeriksaan kondisi terhadap simpul di klaster target mereka dan tidak mengirim permintaan masuk ke simpul yang gagal pemeriksaan kondisi. Oleh karena itu, tingkat fundamental ketersediaan tinggi dicapai dengan penyeimbangan beban. Juga, karena penyeimbang beban melakukan pemeriksaan kondisi aktif dan pasif terhadap semua simpul klaster dalam interval sub-detik, waktu untuk failover hampir seketika.

Keputusan tentang simpul mana yang akan diarahkan didasarkan pada aturan sistem yang didefinisikan dalam penyeimbang beban, termasuk:

- Round robin
- Sesi atau afinitas IP, yang memastikan bahwa beberapa permintaan dalam sesi atau dari IP yang sama dikirim ke simpul yang sama dalam klaster
- Berbasis latensi
- Berbasis beban

Topik

- Penyeimbangan Beban di AWS untuk WebRTC menggunakan Application Load Balancer dan Auto Scaling
- Implementasi SIP menggunakan Network Load Balancer atau Produk AWS Marketplace

Penerapan dalam Arsitektur RTC

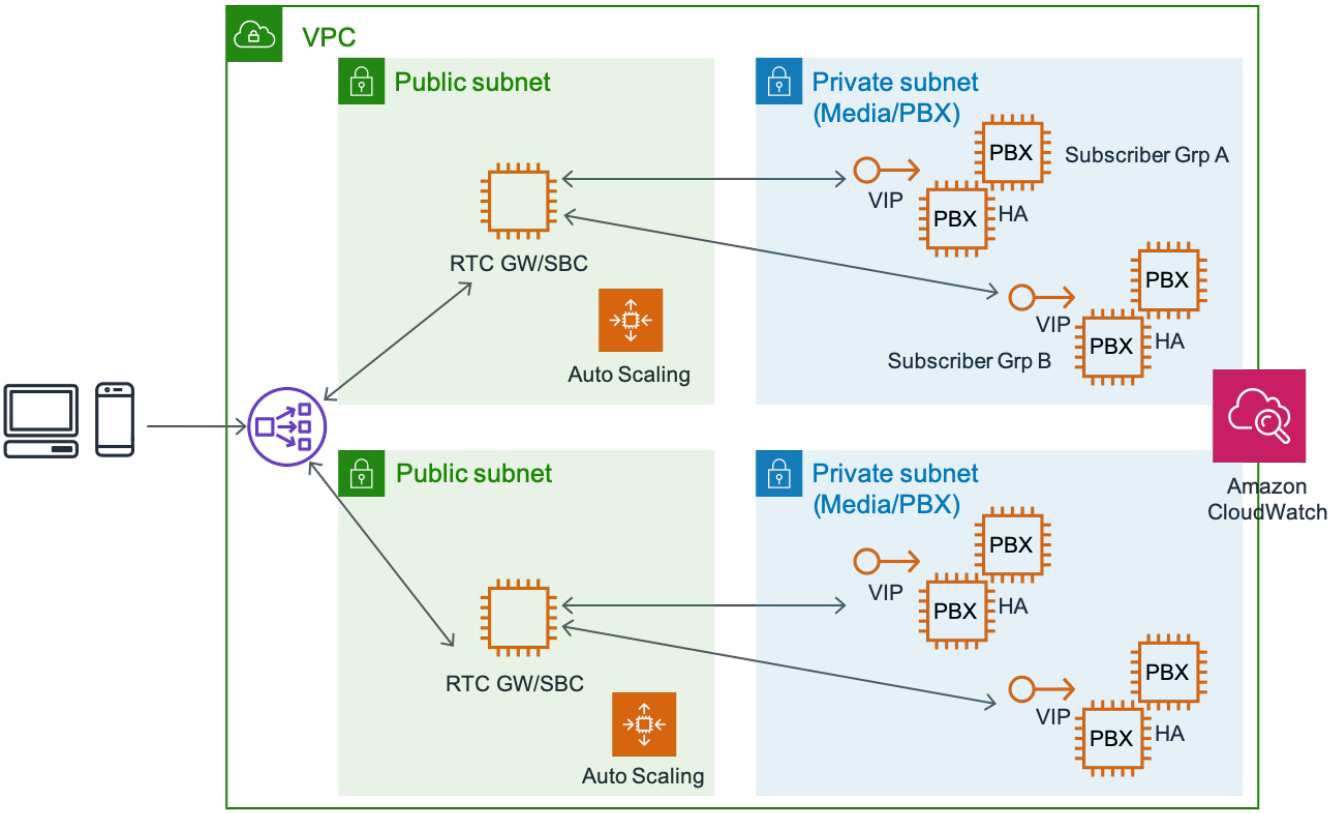
Protokol WebRTC memungkinkan WebRTC Gateway untuk mudah memuat seimbang melalui penyeimbang beban berbasis HTTP, seperti Elastic Load Balancing, Application Load Balancer, atau Network Load Balancer. Dengan sebagian besar implementasi SIP yang mengandalkan transportasi melalui TCP dan UDP, penyeimbangan beban tingkat jaringan atau koneksi dengan dukungan untuk lalu lintas berbasis TCP dan UDP diperlukan.

Penyeimbangan Beban di AWS untuk WebRTC menggunakan Application Load Balancer dan Auto Scaling

Dalam kasus komunikasi berbasis WebRTC, Elastic Load Balancing menyediakan penyeimbang beban yang dikelola sepenuhnya, sangat tersedia dan dapat diskalakan untuk berfungsi sebagai titik masuk untuk permintaan, yang kemudian diarahkan ke klaster target instans EC2 yang terkait dengan Elastic Load Balancing. Selain itu, karena permintaan WebRTC bersifat stateless, Anda dapat menggunakan Amazon EC2 Auto Scaling, untuk menyediakan skalabilitas, elastisitas, dan ketersediaan yang tinggi sepenuhnya otomatis dan dapat dikontrol.

Application Load Balancer menyediakan layanan penyeimbangan beban yang dikelola sepenuhnya yang sangat tersedia menggunakan beberapa Zona Ketersediaan, dan dapat diskalakan. Ini mendukung penyeimbangan beban permintaan WebSocket yang menangani sinyal untuk aplikasi WebRTC dan komunikasi dua arah antara klien dan server menggunakan koneksi TCP berjalan lama. Application Load Balancer juga mendukung perutean berbasis konten dan sesi lekat, merutekan permintaan dari klien yang sama ke target yang sama menggunakan cookie yang dihasilkan penyeimbangan beban. Jika Anda mengaktifkan sesi lekat, target yang sama menerima permintaan dan dapat menggunakan cookie untuk memulihkan konteks sesi.

Gambar 5 menunjukkan topologi target.



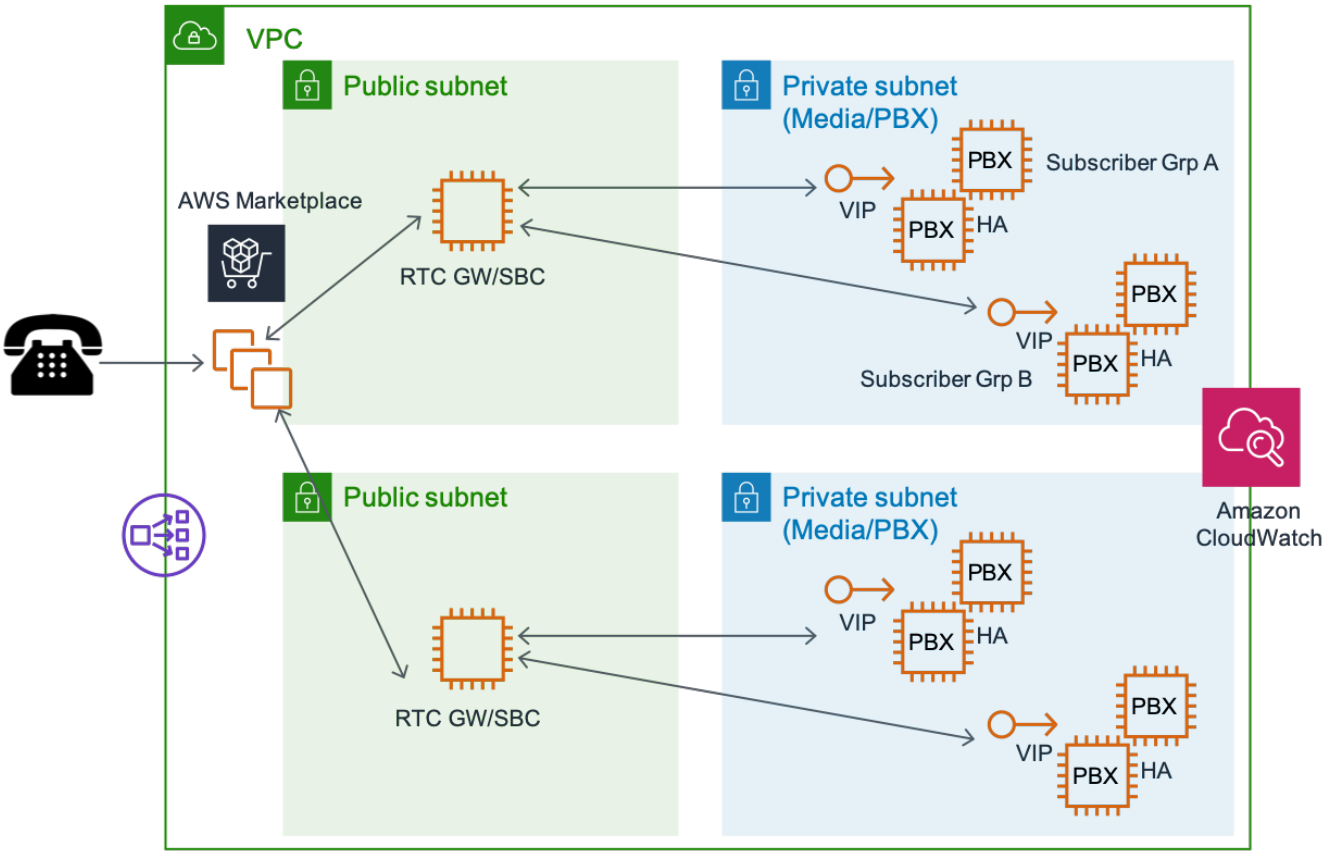
Gambar 5: skalabilitas WebRTC dan arsitektur ketersediaan tinggi

Implementasi SIP menggunakan Network Load Balancer atau Produk AWS Marketplace

Dalam kasus komunikasi berbasis SIP, koneksi dibuat melalui TCP atau UDP, dengan sebagian besar aplikasi RTC menggunakan UDP. Jika SIP/TCP adalah protokol sinyal pilihan, maka layak untuk menggunakan Network Load Balancer untuk sepenuhnya dikelola, sangat tersedia, dapat diskalakan, dan penyeimbangan beban performa.

Network Load Balancer beroperasi pada level koneksi (Lapisan 4), yang merute koneksi ke target - instans Amazon EC2, kontainer, dan alamat IP berdasarkan data protokol IP. Karena sangat ideal untuk penyeimbangan muatan lalu lintas TCP, Network Load Balancer mampu menangani jutaan permintaan setiap detik sementara tetap mempertahankan latensi yang sangat rendah. Layanan ini terintegrasi dengan layanan AWS populer lainnya, seperti AWS Auto Scaling, Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) dan AWS CloudFormation.

Jika koneksi SIP dimulai, pilihan lain adalah menggunakan perangkat lunak AWS Marketplace komersial off-the-shelf (COTS). AWS Marketplace menawarkan banyak produk yang dapat menangani UDP dan jenis lain dari lapisan 4 koneksi penyeimbangan beban. COTS ini biasanya mencakup dukungan untuk ketersediaan tinggi dan umumnya terintegrasi dengan fitur, seperti AWS Auto Scaling, untuk lebih meningkatkan ketersediaan dan skalabilitas. Gambar 6 menunjukkan topologi target:



Gambar 6: Skalabilitas RTC berbasis SIP dengan AWS Marketplace produk

Lintas Wilayah DNS berbasis Load Balancing dan Failover

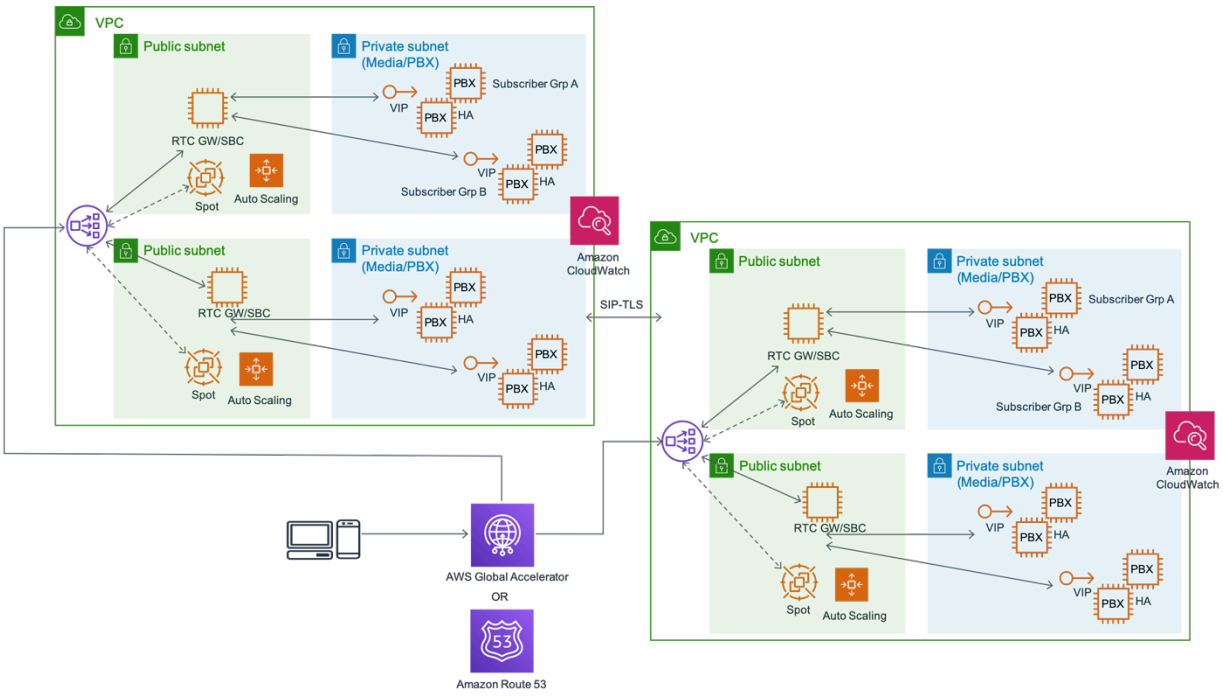
Amazon Route 53 menyediakan layanan DNS global yang dapat digunakan sebagai titik akhir publik atau privat bagi klien RTC untuk mendaftar dan terhubung dengan aplikasi media. Dengan Amazon Route 53, pemeriksaan kondisi DNS dapat dikonfigurasi untuk mengarahkan lalu lintas ke titik akhir yang sehat atau untuk memantau kondisi aplikasi Anda secara independen. Alur Lalu Lintas Amazon Route 53 mempermudah Anda mengelola lalu lintas secara global melalui beragam jenis perutean, termasuk Perutean Berbasis Latensi, Geo DNS, Geoproximity, dan Weighted Round Robin – semuanya dapat dikombinasikan dengan Failover DNS untuk mengaktifkan berbagai arsitektur latensi rendah, dan toleransi terhadap kesalahan. Menggunakan visual sederhana Arus Lalu Lintas

Amazon Route 53, Anda dapat mengelola dengan mudah bagaimana pengguna akhir Anda dirutekan ke titik akhir aplikasi – entah dalam satu wilayah AWS atau didistribusikan di seluruh dunia.

Dalam kasus deployment global, kebijakan perutean berbasis latensi di Route 53 sangat berguna untuk mengarahkan pelanggan ke titik kehadiran terdekat untuk server media untuk meningkatkan kualitas layanan yang terkait dengan pertukaran media real-time.

Perhatikan bahwa untuk menegakkan failover ke alamat DNS baru, cache klien harus memerah. Juga, perubahan DNS mungkin tertunda karena disebarkan di seluruh server DNS global. Anda dapat mengelola interval penyegaran untuk pencarian DNS dengan atribut Time to Live. Atribut ini dapat dikonfigurasi pada saat menyiapkan kebijakan DNS.

Untuk menjangkau pengguna global dengan cepat atau untuk memenuhi persyaratan menggunakan IP publik tunggal, juga AWS Global Accelerator dapat digunakan untuk failover lintas wilayah. AWS Global Accelerator adalah layanan jaringan yang meningkatkan ketersediaan dan performa untuk aplikasi dengan jangkauan lokal dan global. AWS Global Accelerator menyediakan alamat IP statis yang bertindak sebagai titik masuk tetap ke titik akhir aplikasi Anda, seperti Application Load Balancers, Network Load Balancers, atau instans Amazon EC2 di Wilayah AWS tunggal atau beberapa. Ini menggunakan jaringan global AWS untuk mengoptimalkan jalur dari pengguna ke aplikasi Anda, meningkatkan performa, seperti latensi lalu lintas TCP dan UDP Anda. AWS Global Accelerator terus memantau kondisi titik akhir aplikasi Anda, dan secara otomatis mengalihkan lalu lintas ke titik akhir sehat terdekat jika titik akhir saat ini berubah tidak sehat. Untuk persyaratan keamanan tambahan, Accelerated Site-to-Site VPN menggunakan AWS Global Accelerator untuk meningkatkan performa koneksi VPN dengan merutekan lalu lintas secara cerdas melalui AWS Global Network dan lokasi edge AWS.



Gambar 7: Desain ketersediaan tinggi antar wilayah menggunakan AWS Global Accelerator atau Amazon Route 53

Daya Tahan Data dan HA dengan Penyimpanan Tetap

Sebagian besar aplikasi RTC bergantung pada penyimpanan tetap untuk menyimpan dan mengakses data untuk autentikasi, otorisasi, akuntansi (data sesi, catatan detail panggilan, dll.), Pemantauan operasional, dan penebangan. Di pusat data tradisional, memastikan ketersediaan dan daya tahan tinggi untuk komponen penyimpanan tetap (basis data, sistem file, dan sebagainya) biasanya memerlukan pengangkatan berat melalui pengaturan SAN, desain RAID, dan proses untuk pemrosesan cadangan, pemulihan, dan failover. AWS Cloud sangat menyederhanakan dan meningkatkan praktik pusat data tradisional seputar ketahanan dan ketersediaan data.

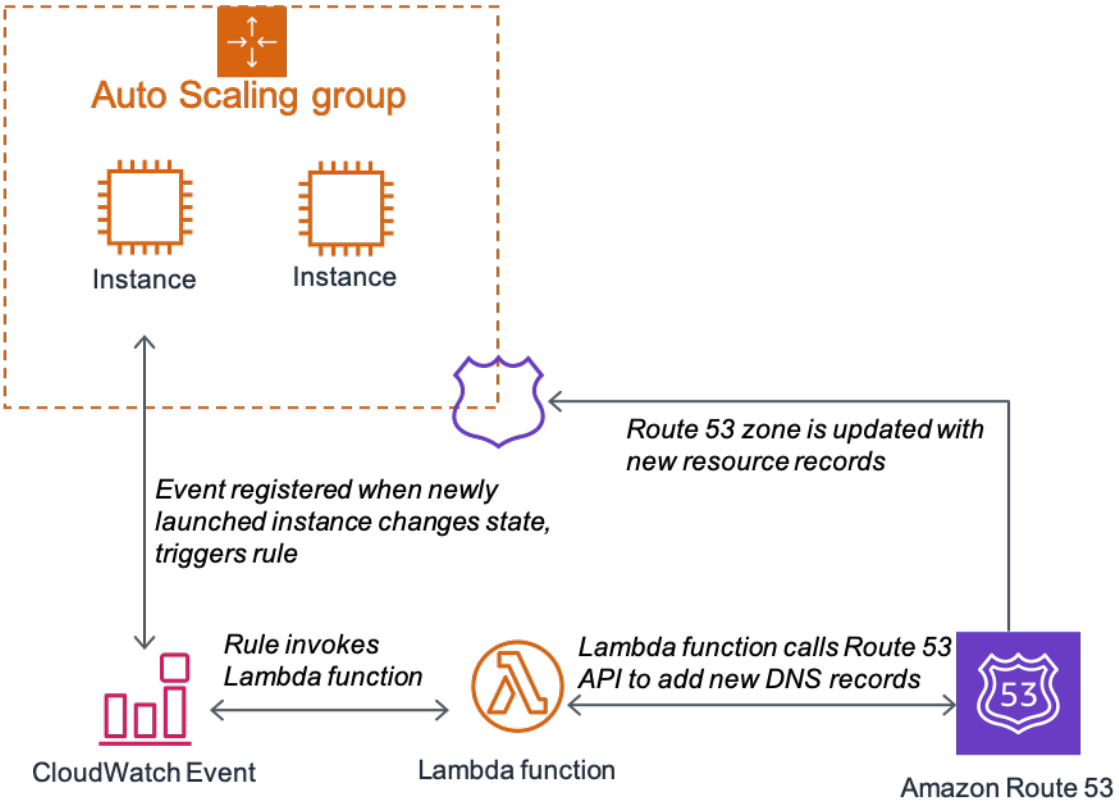
Untuk penyimpanan objek dan penyimpanan file, layanan AWS seperti Amazon Simple Storage Service (Amazon S3) dan Amazon Elastic File System (Amazon EFS) menyediakan ketersediaan dan skalabilitas tinggi yang dikelola. Amazon S3 memiliki daya tahan data 11 sembilan.

Untuk penyimpanan data transaksional, pelanggan memiliki opsi untuk memanfaatkan Amazon Relational Database Service (Amazon RDS) yang dikelola sepenuhnya yang mendukung Amazon Aurora, PostgreSQL, MySQL, MariaDB, Oracle, dan Microsoft SQL Server dengan penyebaran ketersediaan tinggi. Untuk fungsi registrar, profil pelanggan, atau penyimpanan catatan akuntansi

(misalnya CDR), Amazon RDS menyediakan opsi toleran terhadap kesalahan, sangat tersedia, dan dapat diskalakan.

Dynamic Scaling dengan AWS Lambda, Amazon Route 53, dan AWS Auto Scaling

AWS memungkinkan perantaian fitur dan kemampuan untuk menggabungkan fungsi nirserver kustom sebagai layanan berdasarkan peristiwa infrastruktur. Salah satu pola desain yang memiliki banyak kegunaan serbaguna dalam aplikasi RTC adalah kombinasi pengait siklus hidup skala otomatis dengan Amazon CloudWatch Events, Amazon Route 53, dan fungsi AWS Lambda. Fungsi AWS Lambda dapat menanamkan tindakan atau logika. Gambar 8 menunjukkan bagaimana fitur-fitur ini dirantai bersama-sama dapat meningkatkan keandalan sistem dan skalabilitas dengan otomatisasi.



Gambar 8: Penskalaan otomatis dengan pembaruan dinamis ke Amazon Route 53

WebRTC Sangat Tersedia dengan Kinesis Video Streams

Amazon Kinesis Video Streams menawarkan streaming media real-time melalui WebRTC, yang memungkinkan pengguna untuk menangkap, memproses, dan menyimpan aliran media untuk pemutaran, analitik, dan machine learning. Aliran ini sangat tersedia, dapat diskalakan, dan sesuai dengan standar WebRTC. Amazon Kinesis Video Streams menyertakan titik akhir sinyal WebRTC untuk penemuan rekan cepat dan pendirian koneksi yang aman. Ini termasuk dikelola Session Traversal Utilities untuk NAT (STUN) dan Traversal Menggunakan Relay sekitar NAT (TURN) titik akhir untuk pertukaran real-time media antara rekan-rekan. Ini juga mencakup SDK sumber terbuka gratis yang secara langsung terintegrasi dengan firmware kamera untuk memungkinkan komunikasi aman dengan titik akhir Kinesis Video Streams, memungkinkan penemuan rekan dan streaming media. Akhirnya, ia menyediakan perpustakaan klien untuk Android, iOS, dan JavaScript yang memungkinkan WebRTC yang kompatibel dengan pemutar seluler dan web untuk menemukan dan terhubung secara aman dengan perangkat kamera untuk media streaming dan komunikasi dua arah.

Trunking SIP yang Sangat Tersedia dengan Konektor Suara Amazon Chime

Amazon Chime Voice Connector memberikan layanan trunking SIP bayar sesuai penggunaan yang memungkinkan perusahaan untuk melakukan dan/atau menerima panggilan telepon yang aman dan murah dengan sistem telepon mereka. Amazon Chime Voice Connector adalah alternatif berbiaya rendah untuk penyedia layanan SIP batang atau Integrated Services Digital Network (ISDN) Primary Rate Interfaces (PrI). Pelanggan memiliki opsi untuk mengaktifkan panggilan masuk, panggilan keluar, atau keduanya. Layanan ini memanfaatkan jaringan AWS untuk memberikan pengalaman panggilan yang sangat tersedia di beberapa Wilayah AWS. Anda dapat melakukan streaming audio dari panggilan telepon trunking SIP, atau meneruskan umpan perekaman media berbasis SIP (SIPREC) ke Amazon Kinesis Video Streams untuk mendapatkan wawasan dari panggilan bisnis secara real time. Anda dapat dengan cepat membuat aplikasi untuk analitik audio melalui integrasi dengan Amazon Transcribe dan pustaka machine learning umum lainnya.

Praktik Terbaik dari Lapangan

Bagian ini bertujuan untuk meringkas praktik terbaik yang telah dilaksanakan oleh beberapa pelanggan AWS terbesar dan paling sukses yang menjalankan beban kerja Protokol Inisiasi Sesi (SIP) waktu nyata yang besar. Pelanggan AWS yang ingin menjalankan infrastruktur SIP mereka sendiri di cloud publik akan menemukan praktik terbaik ini berharga karena mereka dapat membantu meningkatkan keandalan dan ketahanan sistem jika terjadi berbagai jenis kegagalan. Meskipun beberapa praktik terbaik ini bersifat spesifik SIP, sebagian besar berlaku untuk aplikasi komunikasi real-time yang berjalan di AWS.

Topik

Buat Overlay SIP

AWS memiliki tulang punggung jaringan yang kuat, dapat diskalakan, dan berlebihan yang menyediakan konektivitas antar Wilayah yang berbeda. Ketika peristiwa jaringan, seperti pemotongan serat, menurunkan tautan tulang punggung AWS, lalu lintas failover dengan cepat ke jalur redundan menggunakan protokol perutean tingkat jaringan, seperti BGP. Teknik lalu lintas tingkat jaringan ini adalah kotak hitam untuk pelanggan AWS dan sebagian besar bahkan tidak memperhatikan peristiwa failover ini. Namun, pelanggan yang menjalankan beban kerja real-time, seperti suara, video berkualitas tinggi, dan olahpesan latensi rendah, terkadang memperhatikan peristiwa ini. Jadi, bagaimana pelanggan AWS dapat menerapkan teknik lalu lintas mereka sendiri di atas apa yang disediakan oleh AWS di tingkat jaringan? Solusinya adalah men-deploy infrastruktur SIP di berbagai Wilayah AWS. Sebagai bagian dari fitur kontrol panggilan, SIP juga menyediakan kemampuan untuk rute panggilan melalui proksi SIP tertentu.

Gambar 9: Menggunakan perutean SIP untuk mengganti perutean jaringan

Pada Gambar 9, infrastruktur SIP (diwakili oleh titik-titik hijau) berjalan di keempat Wilayah AS. Garis biru mewakili penggambaran fiksi tulang punggung AWS. Jika tidak ada perutean SIP yang dilaksanakan, panggilan yang berasal dari pantai barat AS dan ditakdirkan untuk pantai timur AS melewati tautan tulang punggung yang langsung menghubungkan wilayah Oregon dan Virginia. Diagram menunjukkan bagaimana pelanggan dapat menimpa perutean tingkat jaringan dan melakukan panggilan yang sama antara Oregon dan Virginia yang dirutekan melalui California menggunakan perutean SIP. Jenis teknik lalu lintas SIP ini dapat diimplementasikan menggunakan proksi SIP dan gateway media berdasarkan metrik jaringan seperti transmisi ulang SIP dan preferensi bisnis khusus pelanggan.

Lakukan Pemantauan Terperinci

Pengguna akhir aplikasi suara dan video waktu nyata mengharapkan tingkat performa yang sama seperti yang mereka capai dengan layanan telefoni tradisional. Jadi, ketika mereka mengalami masalah dengan aplikasi, itu akan merusak reputasi penyedia. Untuk menjadi proaktif daripada reaktif, sangat penting bahwa pemantauan mendetail di-deploy di setiap bagian sistem yang melayani pengguna akhir.

Gambar 10: Menggunakan SIPp untuk Memantau Infrastruktur VoIP

Banyak alat sumber terbuka, seperti iPerf atau SIPp, dan VoIPMonitor, tersedia yang dapat digunakan untuk memantau lalu lintas SIP/RTP. Pada contoh sebelumnya, simpul yang menjalankan SIPp dalam mode klien dan server mengukur metrik SIP seperti Successful Calls dan SIP Retransmits antara keempat Wilayah AWS AS. Metrik ini kemudian dapat diekspor ke Amazon CloudWatch menggunakan skrip khusus. Dengan menggunakan CloudWatch, pelanggan dapat membuat alarm pada metrik khusus ini berdasarkan nilai ambang batas tertentu. Tindakan remediasi otomatis atau manual kemudian dapat diambil berdasarkan keadaan alarm CloudWatch ini.

Bagi pelanggan yang tidak ingin mengalokasikan sumber daya teknik yang diperlukan untuk mengembangkan dan memelihara sistem pemantauan khusus, banyak solusi pemantauan VoIP yang baik tersedia di pasaran, seperti ThousandEyes. Contoh tindakan remediasi adalah mengubah perutean SIP berdasarkan peningkatan pengiriman ulang SIP.

Menggunakan DNS untuk Penyeimbang Beban dan Floating IP untuk Failover

Klien IP telefoni yang mendukung kemampuan DNS SRV dapat secara efisien menggunakan redundansi yang dibangun ke dalam infrastruktur dengan penyeimbang beban klien ke SBCS/PBX yang berbeda.

Gambar 11: Menggunakan catatan DNS SRV untuk menyeimbangkan beban klien SIP

Gambar 11 menunjukkan bagaimana pelanggan dapat menggunakan catatan SRV untuk menyeimbangkan beban lalu lintas SIP. Setiap klien IP telefoni yang mendukung standar SRV akan mencari awalan sip._<transport protocol> dalam catatan DNS tipe SRV. Dalam contoh, bagian jawaban dari DNS berisi kedua PBX yang berjalan di Zona Ketersediaan AWS yang berbeda. Namun, selain URI titik akhir, catatan SRV berisi tiga informasi tambahan:

- Angka pertama adalah Prioritas (1 dalam contoh di atas). Prioritas yang lebih rendah lebih disukai lebih tinggi.
- Angka kedua adalah Bobot (10 dalam contoh di atas).
- Dan nomor ketiga adalah Port yang akan digunakan (5060).

Karena prioritasnya sama (1) untuk kedua server PBX, klien menggunakan bobot untuk memuat keseimbangan antara dua PBX. Dalam hal ini, karena bobotnya sama, lalu lintas SIP harus menyeimbangkan beban secara merata antara dua PBX.

DNS dapat menjadi solusi yang baik untuk menyeimbangkan beban klien, tapi bagaimana dengan menerapkan failover dengan mengubah/memperbarui catatan DNS 'A'? Metode ini tidak dianjurkan karena inkonsistensi ditemukan dalam perilaku caching DNS dalam klien dan simpul menengah. Pendekatan yang lebih baik untuk failover intra-AZ antara sekelompok simpul SIP adalah dengan menggunakan penggantian EC2 IP di mana alamat IP host yang terganggu langsung dipindahkan ke host yang sehat dengan menggunakan API EC2. Dipasangkan dengan solusi pemantauan dan pemeriksaan kondisi yang terperinci, penugasan kembali IP dari simpul yang gagal memastikan bahwa lalu lintas dipindahkan ke host yang sehat pada waktu yang tepat yang meminimalkan gangguan pengguna akhir.

Beberapa zona ketersediaan

Setiap Wilayah AWS dibagi menjadi Zona Ketersediaan yang terpisah. Setiap Zona Ketersediaan memiliki daya, pendinginan, dan konektivitas jaringan sendiri dan dengan demikian membentuk domain kegagalan terisolasi. Dalam konstruksi AWS, selalu didorong agar pelanggan menjalankan beban kerja mereka di lebih dari satu Zona Ketersediaan. Hal ini memastikan bahwa aplikasi pelanggan dapat menahan bahkan kegagalan Zona Ketersediaan lengkap - peristiwa yang sangat langka dalam dirinya sendiri. Rekomendasi ini singkatan dari infrastruktur SIP real-time juga.

Gambar 12: Menangani Kegagalan Zona Ketersediaan

Mari kita asumsikan bahwa peristiwa bencana (seperti badai Kategori 5) menyebabkan pemadaman Zona Ketersediaan lengkap di wilayah as-timur-1. Dengan infrastruktur berjalan seperti yang ditunjukkan dalam diagram, semua klien SIP yang awalnya terdaftar dengan simpul di Zona Ketersediaan gagal harus mendaftar ulang dengan simpul SIP yang berjalan di Zona Ketersediaan #2. (Uji perilaku ini dengan klien SIP Anda/ponsel untuk memastikannya didukung.). Meskipun panggilan SIP aktif pada saat pemadaman Zona Ketersediaan hilang, setiap panggilan baru dirutekan melalui Zona Ketersediaan #2.

Untuk meringkas, catatan DNS SRV harus mengarahkan klien ke beberapa catatan 'A', satu di setiap Zona Ketersediaan. Masing-masing catatan 'A' tersebut harus, pada gilirannya, menunjuk ke beberapa alamat IP SBCS/PBX di Zona Ketersediaan yang menyediakan ketahanan intra dan antar-AZ. Baik intra- dan inter-AZ failover dapat diimplementasikan dengan menggunakan penetapan ulang IP jika IP bersifat publik. Meskipun demikian, IP privat tidak dapat dipindahkan antar Zona Ketersediaan. Jika pelanggan menggunakan alamat IP privat, maka mereka harus bergantung pada pendaftaran ulang klien SIP dengan SBC/PBX cadangan untuk failover Inter-AZ.

Simpan Lalu Lintas dalam Satu Zona Ketersediaan dan gunakan Grup Penempatan EC2

Juga dikenal sebagai Zona Ketersediaan Affinity, praktik terbaik ini juga berlaku untuk kejadian langka kegagalan Zona Ketersediaan yang lengkap. Disarankan agar Anda menghilangkan lalu lintas lintas-AZ sehingga setiap lalu lintas SIP atau RTP yang memasuki satu Zona Ketersediaan harus tetap berada di Zona Ketersediaan tersebut sampai keluar dari Wilayah.

Gambar 13: Afinitas Zona Ketersediaan (paling banyak, 50% panggilan aktif hilang)

Gambar 13 menunjukkan arsitektur yang disederhanakan yang menggunakan Afinitas Zona Ketersediaan. Keuntungan komparatif dari pendekatan ini menjadi jelas jika salah satu akun untuk efek pemadaman Zona Ketersediaan lengkap. Seperti yang digambarkan dalam diagram, jika Zona Ketersediaan #2 hilang, 50% dari panggilan aktif paling banyak terpengaruh (dengan asumsi penyeimbangan beban yang sama antara Zona Ketersediaan). Jika Afinitas Zona Ketersediaan belum dilaksanakan, beberapa panggilan akan mengalir antara Zona Ketersediaan Zone di satu Wilayah dan kegagalan kemungkinan besar akan memengaruhi lebih dari 50% dari panggilan aktif.

Selain itu, untuk meminimalkan latensi lalu lintas, kami juga menyarankan agar Anda mempertimbangkan menggunakan grup penempatan EC2 dalam setiap Zona Ketersediaan. Instans yang diluncurkan dalam grup penempatan EC2 yang sama memiliki bandwidth yang lebih tinggi dan mengurangi latensi karena EC2 memastikan kedekatan jaringan dari instans ini relatif terhadap satu sama lain.

Menggunakan Tipe Instans EC2 Jaringan yang Ditingkatkan

Memilih tipe instans yang tepat di Amazon EC2 memastikan keandalan sistem serta penggunaan infrastruktur yang efisien. Amazon EC2 menyediakan berbagai pilihan tipe instans yang dioptimalkan untuk menyesuaikan dengan kasus penggunaan yang berbeda. Tipe instans terdiri dari berbagai kombinasi kapasitas CPU, memori, penyimpanan, dan jaringan dan memberi Anda fleksibilitas untuk memilih campuran sumber daya yang tepat untuk aplikasi Anda. Tipe instans jaringan yang disempurnakan ini memastikan bahwa beban kerja SIP yang berjalan pada mereka memiliki akses ke bandwidth yang konsisten dan latensi agregat yang relatif lebih rendah. Tambahan terbaru untuk Amazon EC2 adalah ketersediaan Elastic Network Adapter (ENA) yang menyediakan bandwidth hingga 100 Gbps. Katalog terbaru tipe instans EC2 dan fitur terkait dapat ditemukan di halaman tipe instans EC2.

Bagi sebagian besar pelanggan, generasi terbaru dari instans komputasi yang dioptimalkan harus memberikan nilai terbaik untuk biaya. Misalnya, C5N mendukung Adapter Jaringan Elastis baru dengan bandwidth hingga 100 Gbps dengan jutaan paket per detik (PPS). Sebagian besar aplikasi waktu nyata juga akan mendapatkan keuntungan dari menggunakan Intel Data Plane Developer Kit (DPDK) yang dapat sangat meningkatkan pemrosesan paket jaringan.

Namun, selalu merupakan praktik terbaik untuk mengukur berbagai tipe instans EC2 sesuai dengan kebutuhan Anda untuk melihat tipe instans mana yang paling sesuai untuk Anda. Benchmarking juga memungkinkan Anda untuk menemukan parameter konfigurasi lainnya, seperti jumlah maksimum panggilan tipe instans tertentu dapat memproses pada suatu waktu.

Pertimbangan Keamanan

Komponen aplikasi RTC biasanya berjalan langsung di internet yang menghadap instans Amazon EC2. Selain TCP, alur menggunakan protokol seperti UDP dan SIP. Dalam kasus ini, AWS Shield Standard melindungi instans Amazon EC2 dari serangan DDoS lapisan infrastruktur umum (Lapisan 3 dan 4), seperti serangan refleksi UDP, refleksi DNS, refleksi NTP, refleksi SSDP, dan sebagainya. AWS Shield Standard menggunakan berbagai teknik seperti pembentukan lalu lintas berbasis prioritas yang secara otomatis terlibat ketika tanda tangan serangan DDoS yang didefinisikan dengan baik terdeteksi.

AWS juga memberikan perlindungan lanjutan terhadap serangan DDoS yang besar dan canggih untuk aplikasi ini dengan AWS Shield Advanced mengaktifkan alamat IP Elastis. AWS Shield Advanced menyediakan deteksi DDoS yang disempurnakan yang secara otomatis mendeteksi jenis sumber daya AWS dan ukuran instans EC2 dan menerapkan mitigasi standar yang sesuai dengan perlindungan terhadap luapan SYN atau UDP. Dengan AWS Shield Advanced, pelanggan juga dapat membuat profil mitigasi kustom mereka sendiri dengan melibatkan Tim Respons DDoS AWS (DRT) 24x7. AWS Shield Advanced juga memastikan bahwa selama serangan DDoS, semua Daftar Kontrol Akses (ACL) Jaringan VPC Amazon Anda secara otomatis diberlakukan di perbatasan jaringan AWS yang memberi Anda akses ke bandwidth tambahan dan kapasitas scrubbing untuk mengurangi serangan DDoS volumetrik besar.

Kesimpulan

Beban kerja komunikasi waktu nyata (RTC) dapat di-deploy di Amazon Web Services (AWS) untuk mencapai skalabilitas, elastisitas, dan ketersediaan tinggi serta memenuhi persyaratan utama. Saat ini, beberapa pelanggan menggunakan AWS, partner-nya, dan solusi sumber terbuka untuk menjalankan beban kerja RTC dengan mengurangi biaya dan ketangkasan yang lebih cepat serta jejak global yang berkurang.

Arsitektur referensi dan praktik terbaik yang disediakan dalam laporan resmi ini dapat membantu pelanggan berhasil menyiapkan beban kerja RTC di AWS dan mengoptimalkan solusi untuk memenuhi persyaratan pengguna akhir serta mengoptimalkan cloud.

Kontributor

Individu dan organisasi berikut berkontribusi pada dokumen ini:

- Ahmad Khan, Arsitek Solusi Senior, Amazon Web Services
- Tipu Qureshi, Insinyur Utama, AWS Support, Amazon Web Services
- Hasan Khan, Manajer Akun Teknis Senior, Amazon Web Services
- Shoma Chakravarty, Pemimpin Teknis WW, Telekom, Amazon Web Services

Revisi Dokumen

Untuk menerima pemberitahuan tentang pembaruan laporan resmi ini, berlangganan umpan RSS.

pembaruan-riwayat-perubahan	pembaruan-riwayat-deskripsi	pembaruan-riwayat-tanggal
Laporan resmi diperbarui	Diperbarui untuk layanan dan fitur terbaru.	13 Februari 2020
Publikasi awal	Laporan resmi pertama kali dipublikasikan.	1 Oktober 2018

Pemberitahuan

Pelanggan bertanggung jawab untuk membuat penilaian independen mereka sendiri atas informasi dalam dokumen ini. Dokumen ini: (a) hanya disediakan sebagai informasi, (b) berisi penawaran produk dan praktik AWS saat ini, yang dapat berubah tanpa pemberitahuan, dan (c) tidak menjadi komitmen atau jaminan apa pun dari AWS dan afiliasi, pemasok, atau pemberi lisensinya. Produk atau layanan AWS disediakan “sebagaimana adanya” tanpa jaminan, representasi, atau ketentuan apa pun, baik tersurat maupun tersirat. Tanggung jawab dan kewajiban AWS kepada pelanggannya dikendalikan oleh perjanjian AWS, dan dokumen ini bukan bagian dari, juga tidak mengubah, perjanjian apa pun antara AWS dan pelanggannya.