AWS Whitepaper

AWS Outposts Pertimbangan Desain dan Arsitektur Ketersediaan Tinggi

AWS Outposts Pertimbangan Desain dan Arsitektur Ketersediaan Tinggi: AWS Whitepaper

Table of Contents

Abstrak dan pengantar i
Apakah Anda sudah Well-Architected? 1
Pengantar 1
Memperluas AWS infrastruktur dan layanan ke lokasi lokal 2
Memahami Model Tanggung Jawab Bersama yang diperbarui 5
Berpikir dalam hal mode kegagalan 7
Mode kegagalan 1: Jaringan 7
Mode kegagalan 2: Contoh 7
Mode kegagalan 3: Hitung 8
Mode kegagalan 4: Rak atau pusat data 8
Mode kegagalan 5: Zona AWS Ketersediaan atau Wilayah 9
Membangun aplikasi HA dan solusi infrastruktur dengan AWS Outposts rak 10
Jaringan 11
Lampiran jaringan 12
Konektivitas jangkar 15
Perutean aplikasi/beban kerja 19
Hitung 22
Perencanaan kapasitas 22
Manajemen kapasitas 26
Penempatan instans 27
Penyimpanan 30
Perlindungan data 31
Mode kegagalan yang lebih besar 33
Kesimpulan 37
Kontributor 38
Riwayat dokumen 39
Pemberitahuan 40
AWS Glosarium 41
..... xlii

AWS Outposts Pertimbangan Desain dan Arsitektur Ketersediaan Tinggi

Tanggal publikasi: 12 Agustus 2021 (Riwayat dokumen)

Whitepaper ini membahas pertimbangan arsitektur dan praktik yang direkomendasikan yang dapat diterapkan oleh manajer TI dan arsitek sistem untuk membangun lingkungan aplikasi lokal yang sangat tersedia. AWS Outposts

Apakah Anda sudah Well-Architected?

Kerangka Kerja AWS Well-Architected membantu Anda memahami pro dan kontra dari keputusan yang Anda buat saat membangun sistem di cloud. Enam pilar dari Kerangka Kerja ini memungkinkan Anda mempelajari praktik terbaik arsitektural untuk merancang dan mengoperasikan sistem yang andal, aman, efisien, hemat biaya, dan berkelanjutan. Dengan menggunakan AWS Well-Architected Tool, tersedia tanpa biaya di AWS Management Console, Anda dapat meninjau beban kerja Anda terhadap praktik terbaik ini dengan menjawab serangkaian pertanyaan untuk setiap pilar.

Untuk panduan lebih lanjut dari para ahli dan praktik terbaik untuk arsitektur cloud Anda—referensi penerapan arsitektur, diagram, dan laporan resmi—lihat Pusat Arsitektur AWS.

Pengantar

Paper ini ditujukan untuk manajer TI dan arsitek sistem yang ingin menyebarkan, memigrasi, dan mengoperasikan aplikasi menggunakan platform AWS cloud dan menjalankan aplikasi tersebut di tempat dengan AWS Outposts rak, faktor bentuk rak 42U. AWS Outposts

Ini memperkenalkan pola arsitektur, anti-pola, dan praktik yang direkomendasikan untuk membangun sistem yang sangat tersedia yang mencakup AWS Outposts rak. Anda akan belajar bagaimana mengelola kapasitas AWS Outposts rak Anda dan menggunakan jaringan dan layanan fasilitas pusat data untuk menyiapkan solusi infrastruktur AWS Outposts rak yang sangat tersedia.

AWS Outposts rack adalah layanan yang dikelola sepenuhnya yang menyediakan kumpulan logis komputasi awan, penyimpanan, dan kemampuan jaringan. Dengan rak Outposts, pelanggan dapat menggunakan layanan AWS terkelola yang didukung di lingkungan lokal mereka, termasuk: Amazon

Elastic Compute Cloud (Amazon EC2), Amazon Elastic Block Store (AmazonEBS), Amazon S3 di Outposts, Amazon Elastic Kubernetes Service (Amazon EKS), Amazon Elastic Container Service (Amazon ECS),Amazon Elastic Container Service (Amazon ECS), Amazon Amazon Relational Database Service (Amazon RDS), dan layanan lainnya di Outposts.AWS Layanan di Outposts dikirimkan pada Sistem AWS Nitro yang sama yang digunakan dalam. Wilayah AWS

Dengan memanfaatkan AWS Outposts rack, Anda dapat membangun, mengelola, dan menskalakan aplikasi lokal yang sangat tersedia menggunakan layanan dan alat AWS cloud yang sudah dikenal. AWS Outposts rack sangat ideal untuk beban kerja yang memerlukan akses latensi rendah ke sistem lokal, pemrosesan data lokal, residensi data, dan migrasi aplikasi dengan saling ketergantungan sistem lokal.

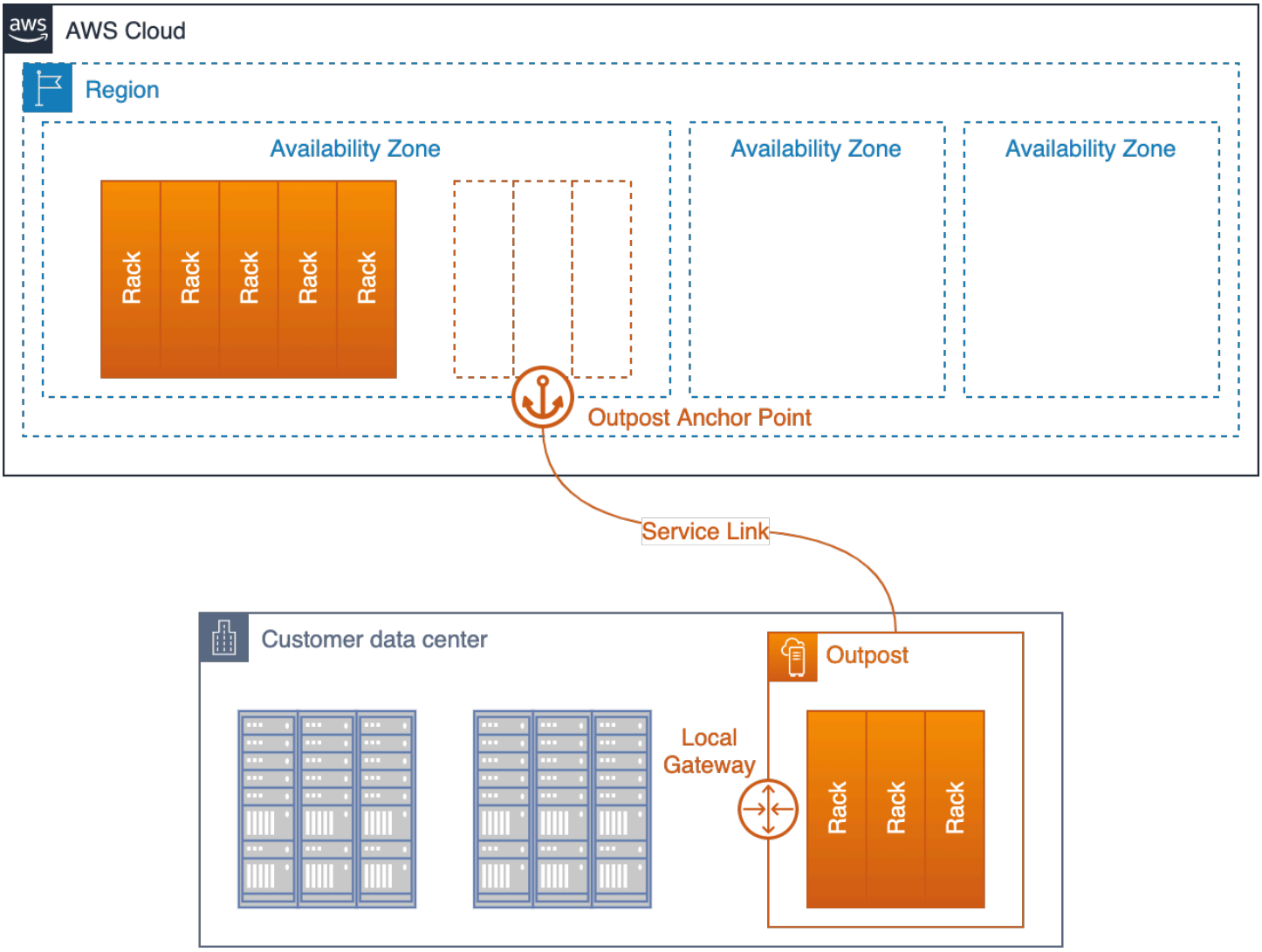
Memperluas AWS infrastruktur dan layanan ke lokasi lokal

AWS Outposts Layanan ini memberikan AWS infrastruktur dan layanan ke lokasi lokal di lebih dari 50 negara dan wilayah, memberikan pelanggan kemampuan untuk menyebarkan AWS infrastruktur, AWS layanan, API, dan alat yang sama ke hampir semua pusat data, ruang lokasi bersama, atau fasilitas lokal untuk pengalaman hybrid yang benar-benar konsisten. Untuk memahami cara mendesain dengan Outposts, Anda harus memahami berbagai tingkatan yang membentuk cloud. AWS

An Wilayah AWSadalah wilayah geografis dunia. Masing-masing Wilayah AWS adalah kumpulan pusat data yang secara logis dikelompokkan ke dalam Availability Zones (AZ). Wilayah AWS menyediakan beberapa (setidaknya dua) Availability Zone yang terpisah secara fisik dan terisolasi yang terhubung dengan latensi rendah, throughput tinggi, dan konektivitas jaringan redundan. Setiap AZ terdiri dari satu atau lebih pusat data fisik.

Pos Luar logis (selanjutnya disebut sebagai Outpost) adalah penyebaran satu atau lebih AWS Outposts rak yang terhubung secara fisik yang dikelola sebagai satu kesatuan. Outpost menyediakan kumpulan kapasitas AWS komputasi dan penyimpanan di salah satu situs Anda sebagai perpanjangan pribadi dari AZ dalam file. Wilayah AWS

Mungkin model konseptual terbaik AWS Outposts adalah memikirkan mencabut satu atau lebih rak dari pusat data di AZ. Wilayah AWS Anda memutar rak dari pusat data AZ ke pusat data Anda. Anda kemudian mencolokkan rak ke titik jangkar di pusat data AZ dengan kabel (sangat) panjang sehingga rak terus berfungsi sebagai bagian dari. Wilayah AWS Anda juga menghubungkannya ke jaringan lokal Anda untuk menyediakan konektivitas latensi rendah antara jaringan lokal Anda dan beban kerja yang berjalan di rak tersebut.



Pos Terdepan ditempatkan di pusat data pelanggan dan terhubung kembali ke jangkar AZ dan Wilayah induknya

Outpost berfungsi sebagai perpanjangan dari AZ di mana ia berlabuh. AWS mengoperasikan, memantau, dan mengelola AWS Outposts infrastruktur sebagai bagian dari Wilayah AWS. Alih-alih kabel fisik yang sangat panjang, Outpost menghubungkan kembali ke Wilayah induknya melalui satu set terowongan VPN terenkripsi yang disebut Service Link.

Tautan Layanan berakhir pada satu set titik jangkar di Availability Zone (AZ) di Wilayah induk Outpost.

Anda memilih di mana konten Anda disimpan. Anda dapat mereplikasi dan mencadangkan konten Anda ke Wilayah AWS atau lokasi lain. Konten Anda tidak akan dipindahkan atau disalin di luar lokasi

yang Anda pilih tanpa persetujuan Anda, kecuali jika diperlukan untuk mematuhi hukum atau perintah yang mengikat dari badan pemerintah. Untuk informasi selengkapnya, lihat FAQ Privasi AWS Data.

Beban kerja yang Anda terapkan di rak tersebut berjalan secara lokal. Dan, sementara kapasitas komputasi dan penyimpanan yang tersedia di rak tersebut terbatas dan tidak dapat mengakomodasi menjalankan layanan skala cloud Wilayah AWS, sumber daya yang digunakan di rak (instans Anda dan penyimpanan lokalnya) menerima manfaat berjalan secara lokal sementara pesawat manajemen terus beroperasi di. Wilayah AWS

Untuk menerapkan beban kerja di Outpost, Anda menambahkan subnet ke lingkungan Virtual Private Cloud (VPC) dan menentukan Outpost sebagai lokasi untuk subnet. Kemudian, Anda memilih subnet yang diinginkan saat menerapkan AWS sumber daya yang didukung melalui alat, AWS Management Console CLI, API, CDK, atau infrastruktur sebagai kode (IAc). Contoh di subnet Outpost berkomunikasi dengan instans lain di Outpost atau di Wilayah melalui jaringan VPC.

Outpost Service Link membawa lalu lintas manajemen Outpost dan lalu lintas VPC pelanggan (lalu lintas VPC antara subnet di Outpost dan subnet di Wilayah).

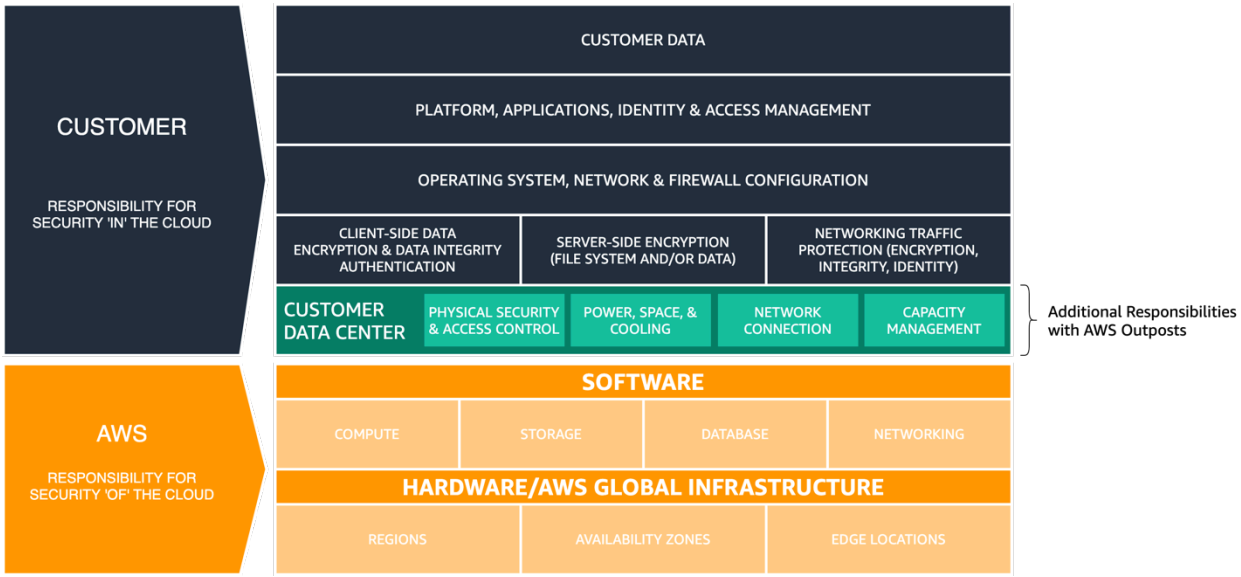
Istilah penting:

- AWS Outposts— adalah layanan yang dikelola sepenuhnya yang menawarkan AWS infrastruktur, AWS layanan, API, dan alat yang sama ke hampir semua pusat data, ruang co-lokasi, atau fasilitas lokal untuk pengalaman hybrid yang benar-benar konsisten.
- Outpost adalah penyebaran satu atau lebih AWS Outposts rak yang terhubung secara fisik yang dikelola sebagai entitas logis tunggal dan kumpulan AWS komputasi, penyimpanan, dan jaringan yang digunakan di situs pelanggan.
- Wilayah Induk — Wilayah AWS yang menyediakan manajemen, layanan pesawat kontrol, dan AWS layanan regional untuk penyebaran Outpost.
- Anchor Availability Zone (anchor AZ) — Availability Zone di Wilayah induk yang menjadi tuan rumah anchor point untuk Outpost. Outpost berfungsi sebagai perpanjangan dari jangkar Availability Zone.
- Anchor Points — titik akhir di anchor AZ yang menerima koneksi dari Outposts yang dikerahkan dari jarak jauh.
- Service Link — satu set terowongan VPN terenkripsi yang menghubungkan Outpost ke Availability Zone jangkar di Wilayah induknya.
- Local Gateway (LGW) — Router virtual interkoneksi logis yang memungkinkan komunikasi antara Outpost Anda dan jaringan lokal Anda.

Memahami Model Tanggung Jawab Bersama yang diperbarui

Saat Anda menyebarkan AWS Outposts infrastruktur ke pusat data atau fasilitas lokasi bersama, Anda mengambil tanggung jawab tambahan dalam model Tanggung Jawab AWS Bersama. Misalnya, di Wilayah, AWS menyediakan sumber daya yang beragam, jaringan inti yang berlebihan, dan konektivitas Wide Area Network (WAN) yang tangguh untuk memastikan layanan tersedia jika terjadi satu atau lebih kegagalan komponen.

Dengan Outposts, Anda bertanggung jawab untuk menyediakan daya tangguh dan konektivitas jaringan ke rak Outpost untuk memenuhi persyaratan ketersediaan Anda untuk beban kerja yang berjalan di Outposts.



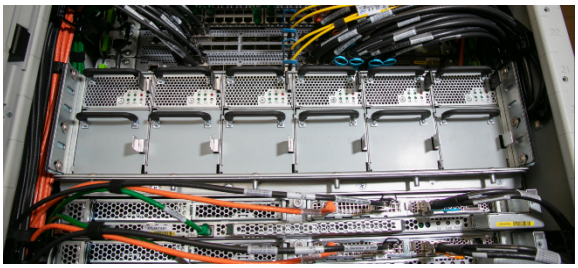
AWS Model Tanggung Jawab Bersama diperbarui untuk AWS Outposts

Dengan AWS Outposts, Anda bertanggung jawab atas keamanan fisik dan kontrol akses lingkungan pusat data. Anda harus menyediakan daya, ruang, dan pendinginan yang cukup untuk menjaga agar Outpost tetap beroperasi dan koneksi jaringan untuk menghubungkan Pos Luar kembali ke Wilayah.

Karena kapasitas Outpost terbatas dan ditentukan oleh ukuran dan jumlah AWS pemasangan rak di situs Anda, Anda harus memutuskan berapa banyak EC2, EBS, dan S3 pada kapasitas Outposts yang Anda butuhkan untuk menjalankan beban kerja awal Anda, mengakomodasi pertumbuhan masa depan, dan untuk menyediakan kapasitas ekstra untuk mengurangi kegagalan server dan peristiwa pemeliharaan.

AWS bertanggung jawab atas ketersediaan infrastruktur Outposts termasuk catu daya, server, dan peralatan jaringan di dalam rak. AWS Outposts AWS juga mengelola hypervisor virtualisasi, sistem penyimpanan, dan AWS layanan yang berjalan di Outposts.

Rak daya pusat di setiap rak Outposts mengkonversi dari daya AC ke DC dan memasok daya ke server di rak melalui arsitektur bus bar. Dengan arsitektur bus bar, setengah catu daya di rak bisa gagal dan semua server akan terus berjalan tanpa gangguan.



Gambar 3 - AWS Outposts Catu daya AC-ke-DC dan distribusi daya batang bus

Sakelar jaringan dan pemasangan kabel di dalam dan di antara rak Outposts juga sepenuhnya berlebihan. Panel patch serat menyediakan konektivitas antara rak Outpost dan jaringan lokal dan berfungsi sebagai titik demarkasi antara lingkungan pusat data yang dikelola pelanggan dan lingkungan yang dikelola. AWS Outposts

Sama seperti di Wilayah, AWS bertanggung jawab atas layanan cloud yang ditawarkan di Outposts dan mengambil tanggung jawab tambahan saat Anda memilih dan menerapkan layanan terkelola tingkat tinggi seperti Amazon RDS di Outposts. Anda harus meninjau Model Tanggung Jawab AWS Bersama dan halaman Pertanyaan yang Sering Diajukan (FAQ) untuk layanan individual saat Anda mempertimbangkan dan memilih layanan untuk diterapkan di Outposts. Sumber daya ini memberikan rincian tambahan tentang pembagian tanggung jawab antara Anda dan AWS.

Berpikir dalam hal mode kegagalan

Saat merancang aplikasi atau sistem yang sangat tersedia, Anda harus mempertimbangkan komponen apa yang mungkin gagal, dampak kegagalan komponen apa yang akan terjadi pada sistem, dan mekanisme apa yang dapat Anda terapkan untuk mengurangi atau menghilangkan dampak kegagalan komponen. Apakah aplikasi Anda berjalan di satu server, dalam satu rak, atau dalam satu pusat data? Apa yang akan terjadi ketika server, rak, atau pusat data mengalami kegagalan sementara atau permanen? Apa yang terjadi ketika ada kegagalan dalam sub-sistem kritis seperti jaringan atau dalam aplikasi itu sendiri? Ini adalah mode kegagalan.

Anda harus mempertimbangkan mode kegagalan di bagian ini saat merencanakan Outposts dan penerapan aplikasi Anda. Bagian berikut akan meninjau cara mengurangi mode kegagalan ini untuk memberikan peningkatan tingkat ketersediaan tinggi untuk lingkungan aplikasi Anda.

Mode kegagalan 1: Jaringan

Penyebaran Outpost bergantung pada koneksi yang tangguh ke Wilayah induknya untuk pengelolaan dan pemantauan. Gangguan jaringan dapat disebabkan oleh berbagai kegagalan seperti kesalahan operator, kegagalan peralatan, dan pemadaman penyedia layanan. Pos Luar, yang mungkin terdiri dari satu atau lebih rak yang terhubung bersama di situs, dianggap terputus ketika tidak dapat berkomunikasi dengan Wilayah melalui Tautan Layanan.

Jalur jaringan yang berlebihan dapat membantu mengurangi risiko peristiwa pemutusan hubungan. Anda harus memetakan dependensi aplikasi dan lalu lintas jaringan untuk memahami dampak peristiwa pemutusan hubungan terhadap operasi beban kerja. Rencanakan redundansi jaringan yang memadai untuk memenuhi persyaratan ketersediaan aplikasi Anda.

Selama peristiwa pemutusan sambungan, instance yang berjalan di Outpost terus berjalan dan dapat diakses dari jaringan lokal melalui Outpost Local Gateway (LGW). Beban kerja dan layanan lokal mungkin terganggu atau gagal jika bergantung pada layanan di Wilayah. Permintaan mutasi (seperti memulai atau menghentikan instance di Pos Luar), operasi bidang kontrol, dan telemetri layanan (misalnya, CloudWatch metrik) akan gagal saat Pos Luar terputus dari Wilayah.

Mode kegagalan 2: Contoh

Instans EC2 dapat menjadi terganggu atau gagal jika server yang mereka jalankan memiliki masalah atau jika instans mengalami kegagalan sistem operasi atau aplikasi. Bagaimana aplikasi menangani

jenis kegagalan ini tergantung pada arsitektur aplikasi. Aplikasi monolitik biasanya menggunakan fitur aplikasi atau sistem untuk pemulihan sementara arsitektur berorientasi layanan modular atau layanan mikro biasanya menggantikan komponen yang gagal untuk mempertahankan ketersediaan layanan.

Anda dapat mengganti instans yang gagal dengan instans baru menggunakan mekanisme otomatis seperti grup EC2 Auto Scaling. Pemulihan otomatis instans dapat memulai ulang instance yang gagal karena kegagalan server asalkan ada kapasitas cadangan yang cukup tersedia di server yang tersisa.

Mode kegagalan 3: Hitung

Server dapat gagal atau menjadi terganggu dan mungkin perlu dikeluarkan dari operasi (sementara atau permanen) karena berbagai alasan, seperti kegagalan komponen dan operasi pemeliharaan terjadwal. Bagaimana layanan di rak Outposts menangani kegagalan dan kerusakan server bervariasi dan dapat bergantung pada bagaimana pelanggan mengonfigurasi opsi ketersediaan tinggi.

Anda harus memesan kapasitas komputasi yang cukup untuk mendukung model `N+M` ketersediaan, di mana `N` kapasitas yang `M` diperlukan dan kapasitas cadangan disediakan untuk mengakomodasi kegagalan server.

Penggantian perangkat keras untuk server yang gagal disediakan sebagai bagian dari layanan AWS Outposts rak yang dikelola sepenuhnya. AWS secara aktif memantau kesehatan semua server dan perangkat jaringan dalam penyebaran Outpost. Jika ada kebutuhan untuk melakukan perawatan fisik, AWS akan menjadwalkan waktu untuk mengunjungi situs Anda untuk mengganti komponen yang gagal. Penyediaan kapasitas cadangan memungkinkan Anda menjaga beban kerja tetap berjalan sementara server yang gagal dikeluarkan dari layanan dan diganti.

Mode kegagalan 4: Rak atau pusat data

Kegagalan rak dapat terjadi karena kehilangan total daya ke rak atau karena kegagalan lingkungan seperti hilangnya pendinginan atau kerusakan fisik pada pusat data akibat banjir atau gempa bumi. Kekurangan dalam arsitektur distribusi daya pusat data atau kesalahan selama pemeliharaan daya pusat data standar dapat mengakibatkan hilangnya daya ke satu atau lebih rak atau bahkan seluruh pusat data.

Skenario ini dapat dikurangi dengan menyebarkan infrastruktur ke beberapa lantai pusat data atau lokasi yang independen satu sama lain dalam kampus atau area metro yang sama.

Mengambil pendekatan ini dengan AWS Outposts rak akan memerlukan pertimbangan yang cermat tentang bagaimana aplikasi dirancang dan didistribusikan untuk berjalan di beberapa Outposts logis terpisah untuk menjaga ketersediaan aplikasi.

Mode kegagalan 5: Zona AWS Ketersediaan atau Wilayah

Setiap Pos Luar ditambatkan ke Availability Zone (AZ) tertentu dalam file. Wilayah AWS Kegagalan dalam jangkar AZ atau Wilayah induk dapat menyebabkan hilangnya manajemen Outpost dan mutabilitas dan dapat mengganggu komunikasi jaringan antara Outpost dan Region.

Mirip dengan kegagalan jaringan, kegagalan AZ atau Region dapat menyebabkan Outpost menjadi terputus dari Wilayah. Instans yang berjalan di Outpost terus berjalan dan dapat diakses dari jaringan lokal melalui Outpost Local Gateway (LGW) dan mungkin terganggu atau gagal jika mengandalkan layanan di Wilayah, seperti yang dijelaskan sebelumnya.

Untuk mengurangi dampak kegagalan AWS AZ dan Wilayah, Anda dapat menyebarkan beberapa Outpost yang masing-masing ditambatkan ke AZ atau Region yang berbeda. Anda kemudian dapat merancang beban kerja Anda untuk beroperasi dalam model penyebaran Multi-outpost terdistribusi menggunakan banyak mekanisme dan pola arsitektur serupa yang Anda gunakan untuk merancang dan menerapkan hari ini. AWS

Membangun aplikasi HA dan solusi infrastruktur dengan AWS Outposts rak

Dengan AWS Outposts rack, Anda dapat membangun, mengelola, dan menskalakan aplikasi lokal yang sangat tersedia menggunakan layanan dan alat AWS cloud yang sudah dikenal. Penting untuk memahami arsitektur dan pendekatan HA cloud umumnya berbeda dari arsitektur HA lokal tradisional yang mungkin Anda jalankan di pusat data Anda hari ini.

Dengan penerapan aplikasi HA lokal tradisional, aplikasi digunakan di mesin virtual (VM). Sistem dan infrastruktur TI yang kompleks dikerahkan dan dipelihara untuk menjaga mesin virtual tetap berjalan dan sehat. VM sering memiliki identitas spesifik dan setiap VM mungkin memainkan peran penting dalam arsitektur aplikasi total.

Peran arsitektur digabungkan erat dengan identitas VM. Arsitek sistem memanfaatkan fitur infrastruktur TI untuk menyediakan lingkungan runtime VM yang sangat tersedia yang menyediakan setiap VM akses andal ke kapasitas komputasi, volume penyimpanan, dan layanan jaringan. Jika VM gagal, proses pemulihan otomatis atau manual dijalankan untuk mengembalikan VM yang gagal ke keadaan sehat, seringkali pada infrastruktur lain atau di pusat data lain sepenuhnya.

Arsitektur Cloud HA mengambil pendekatan yang berbeda. AWS layanan cloud menyediakan kemampuan komputasi, penyimpanan, dan jaringan yang andal. Komponen aplikasi digunakan untuk instans EC2, kontainer, fungsi tanpa server, atau layanan terkelola lainnya.

Instance adalah instantiasi komponen aplikasi — mungkin salah satu dari banyak yang melakukan peran itu. Komponen aplikasi secara longgar digabungkan satu sama lain dan peran yang mereka mainkan dalam arsitektur aplikasi total. Identitas individu dari suatu contoh umumnya tidak penting. Contoh tambahan dapat dibuat atau dihancurkan untuk meningkatkan atau menurunkan dalam menanggapi permintaan. Contoh gagal atau contoh yang tidak sehat hanya diganti dengan contoh sehat baru.

AWS Outposts rack adalah layanan terkelola penuh yang memperluas AWS komputasi, penyimpanan, jaringan, database, dan layanan cloud lainnya ke lokasi lokal untuk pengalaman hybrid yang benar-benar konsisten. Anda tidak boleh menganggap layanan rak Outposts sebagai pengganti drop-in untuk sistem infrastruktur TI dengan mekanisme HA lokal tradisional. Mencoba menggunakan AWS layanan dan Outposts untuk mendukung arsitektur HA lokal tradisional adalah anti-pola.

Beban kerja yang berjalan di AWS Outposts rak menggunakan mekanisme HA cloud seperti Amazon EC2 Auto Scaling (untuk menskalakan secara horizontal untuk memenuhi tuntutan beban kerja),

pemeriksaan kesehatan EC2 (untuk mendeteksi dan menghapus instans yang tidak sehat), dan Application Load Balancer (untuk mengarahkan lalu lintas beban kerja yang masuk ke instans yang diskalakan atau diganti). Saat memigrasikan aplikasi ke cloud, baik ke sebuah Wilayah AWS atau AWS Outposts rak, Anda harus memperbarui arsitektur aplikasi HA Anda untuk mulai memanfaatkan layanan cloud terkelola dan mekanisme HA cloud.

Bagian berikut memperkenalkan pola arsitektur, anti-pola, dan praktik yang direkomendasikan untuk menerapkan AWS Outposts rak di lingkungan lokal Anda untuk menjalankan beban kerja dengan persyaratan ketersediaan tinggi. Bagian ini memperkenalkan pola dan praktik; namun, mereka tidak memberikan detail konfigurasi dan implementasi. Anda harus membaca dan menjadi akrab dengan FAQ AWS Outposts rak dan Panduan Pengguna serta FAQ dan dokumentasi layanan untuk layanan yang berjalan di rak Outposts saat Anda mempersiapkan lingkungan Anda untuk rak Outposts dan aplikasi Anda untuk migrasi ke layanan. AWS

Topik

- Jaringan
- Hitung
- Penyimpanan
- Mode kegagalan yang lebih besar

Jaringan

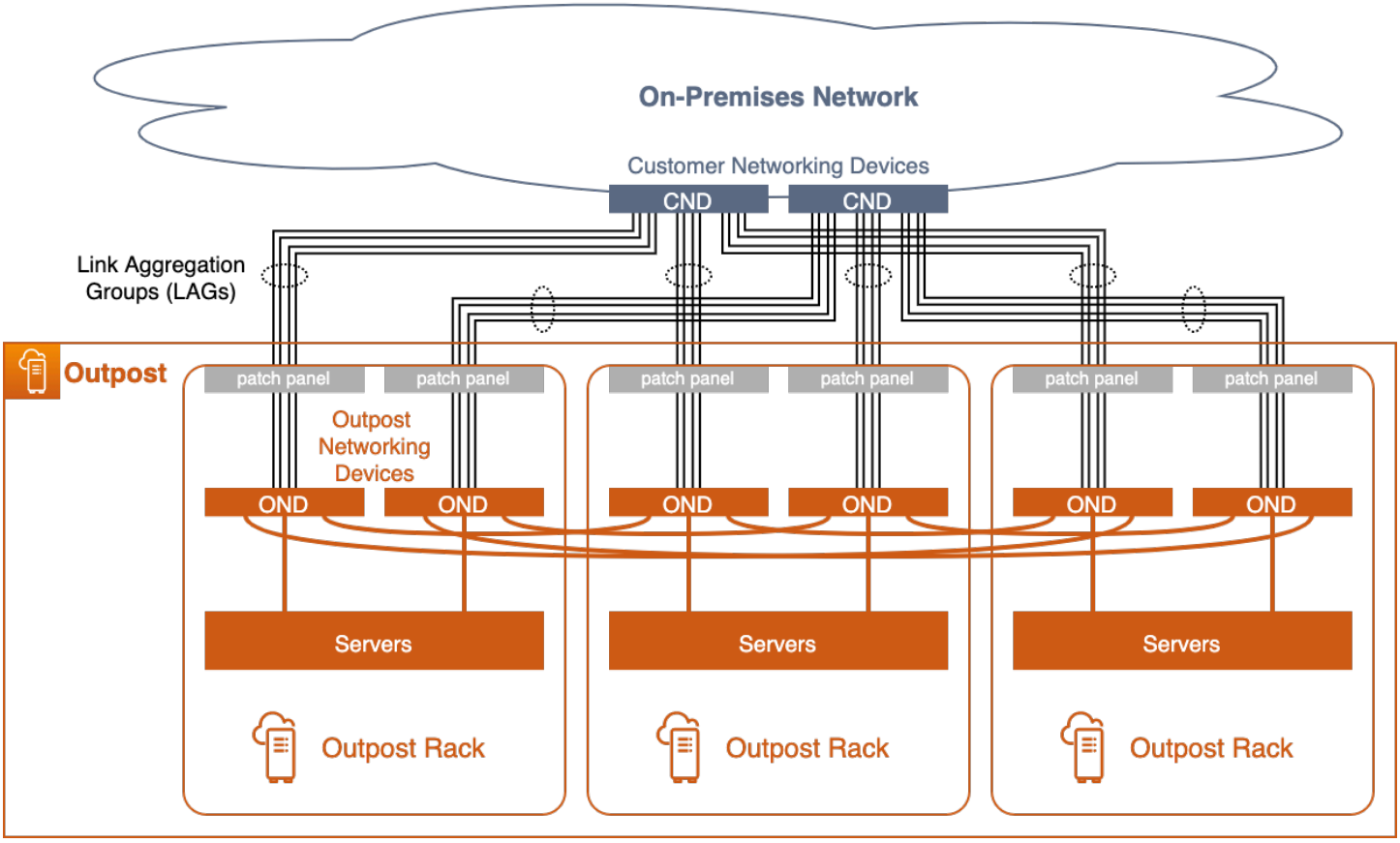
Penyebaran Outpost bergantung pada koneksi yang tangguh ke jangkar AZ agar manajemen, pemantauan, dan operasi layanan berfungsi dengan baik. Anda harus menyediakan jaringan lokal Anda untuk menyediakan koneksi jaringan redundan untuk setiap rak Outpost dan konektivitas yang andal kembali ke titik jangkar di cloud. AWS Pertimbangkan juga jalur jaringan antara beban kerja aplikasi yang berjalan di Outpost dan sistem lokal dan cloud lainnya yang berkomunikasi dengannya — bagaimana Anda akan merutekan lalu lintas ini di jaringan Anda?

Topik

- Lampiran jaringan
- Konektivitas jangkar
- Perutean aplikasi/beban kerja

Lampiran jaringan

Setiap AWS Outposts rak dikonfigurasi dengan top-of-rack sakelar redundan yang disebut Outpost Networking Devices (ONDs). Server komputasi dan penyimpanan di setiap rak terhubung ke kedua OND. Anda harus menghubungkan setiap OND ke sakelar terpisah yang disebut Customer Networking Device (CND) di pusat data Anda untuk menyediakan beragam jalur fisik dan logis untuk setiap rak Outpost. OND terhubung ke CND Anda dengan satu atau lebih koneksi fisik menggunakan kabel serat optik dan transceiver optik. Koneksi fisik dikonfigurasi dalam tautan grup agregasi tautan logis (LAG).



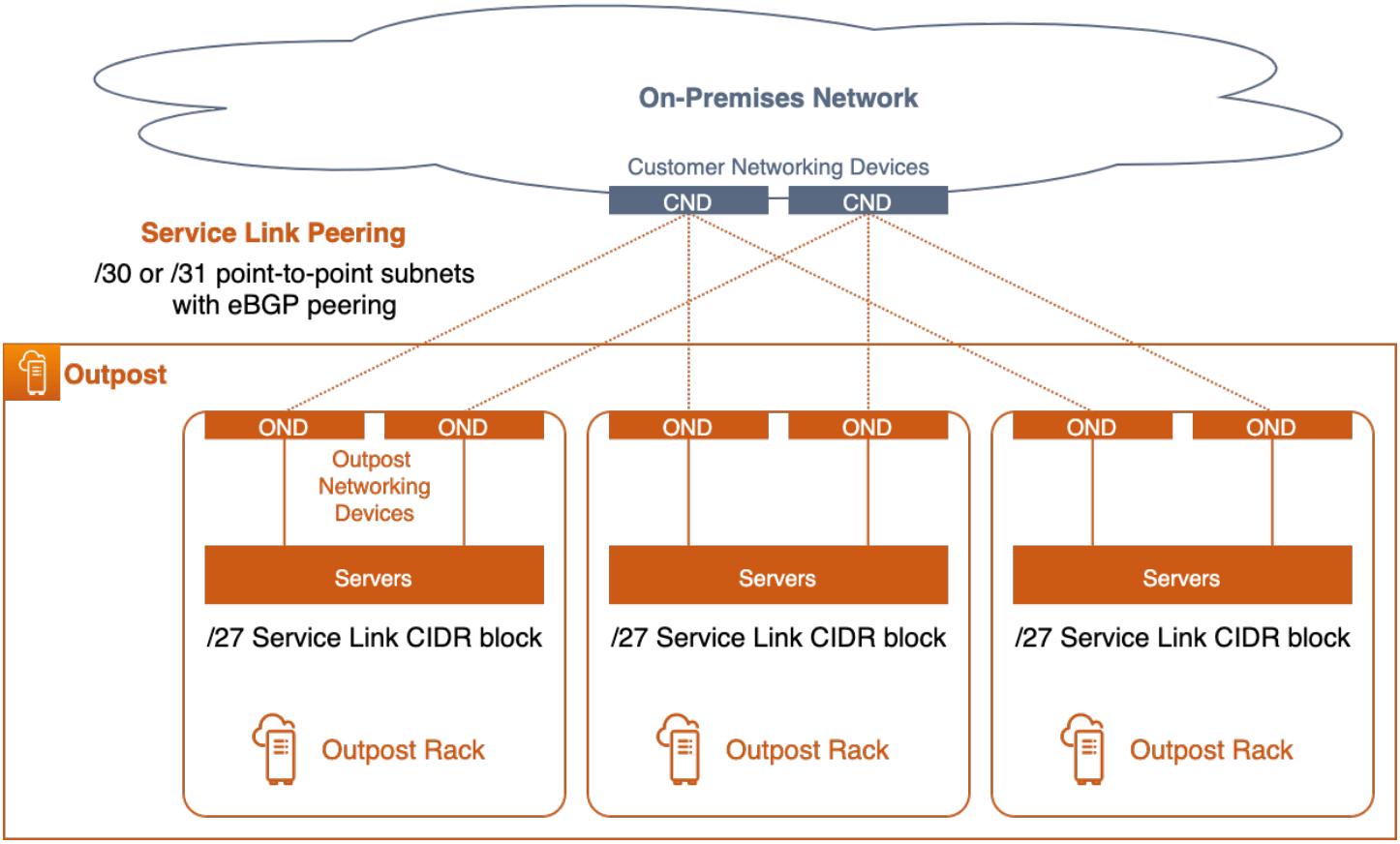
Pos Luar Multi-rak dengan lampiran jaringan redundan

Tautan OND ke CND selalu dikonfigurasi dalam LAG - bahkan jika koneksi fisiknya adalah kabel serat optik tunggal. Mengkonfigurasi tautan sebagai grup LAG memungkinkan Anda meningkatkan bandwidth tautan dengan menambahkan koneksi fisik tambahan ke grup logis. Tautan LAG dikonfigurasi sebagai batang Ethernet IEEE 802.1q untuk mengaktifkan jaringan terpisah antara Outpost dan jaringan lokal.

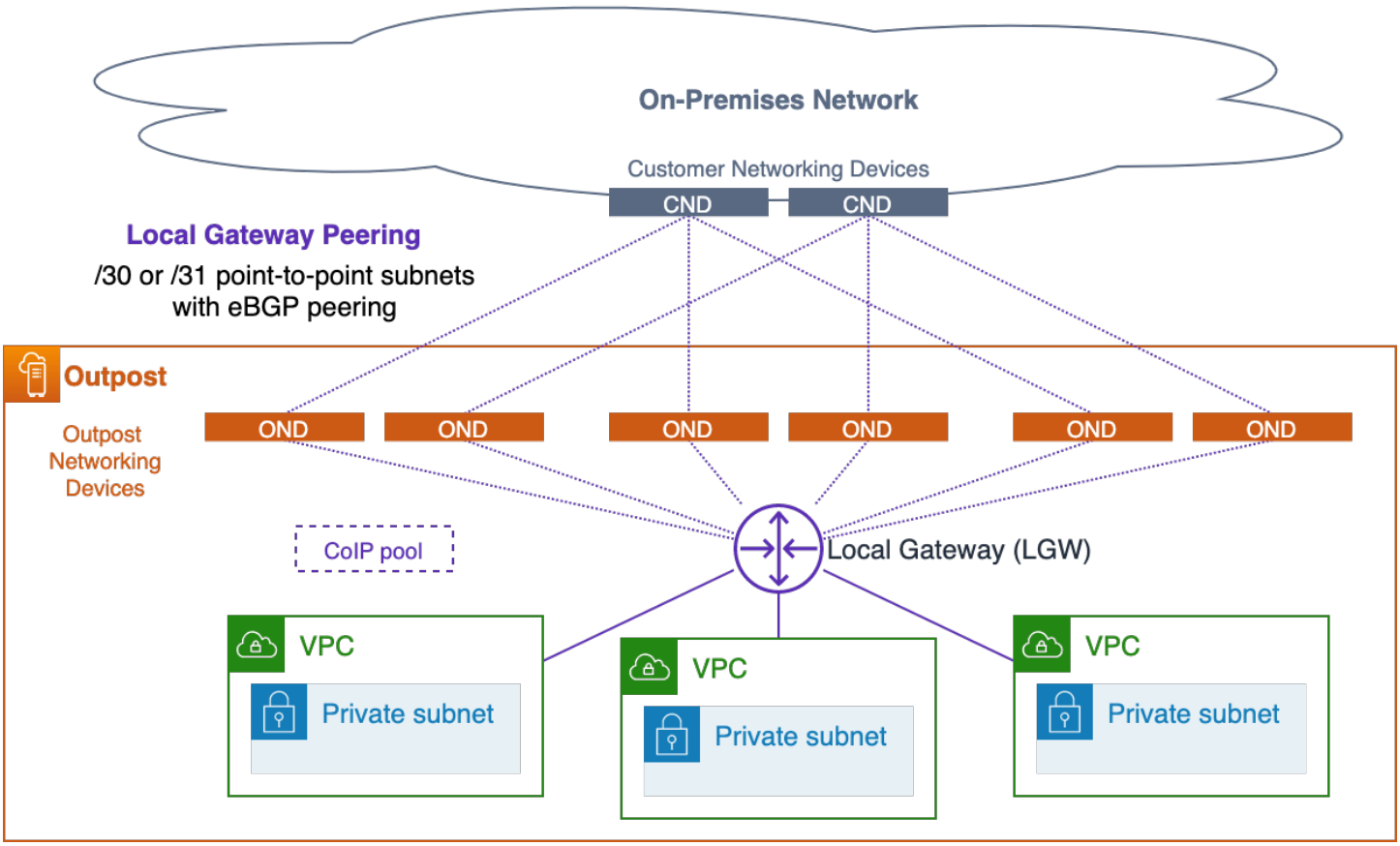
Setiap Outpost memiliki setidaknya dua jaringan terpisah secara logis yang perlu berkomunikasi dengan atau di seluruh jaringan pelanggan:

- Jaringan Service Link - mengalokasikan alamat IP Service Link ke server Outpost dan memfasilitasi komunikasi dengan jaringan lokal untuk memungkinkan server terhubung kembali ke titik jangkar Outpost di Wilayah.
- Jaringan Local Gateway — memungkinkan komunikasi antara subnet VPC di Outpost dan jaringan lokal melalui Outpost Local Gateway (LGW).

Jaringan terpisah ini dilampirkan ke jaringan lokal dengan satu set koneksi point-to-point IP melalui tautan LAG. Setiap tautan OND ke CND LAG dikonfigurasi dengan ID VLAN, point-to-point (/30 atau/31) subnet IP, dan peering eBGP untuk setiap jaringan terpisah (Service Link dan LGW). Anda harus mempertimbangkan tautan LAG, dengan point-to-point VLAN dan subnetnya, sebagai koneksi lapisan-2 tersegmentasi dan dirutekan layer-3. Koneksi IP yang dirutekan menyediakan jalur logis redundan yang memfasilitasi komunikasi antara jaringan terpisah di Outpost dan jaringan lokal.



Layanan Link Peering



Pengintip Gerbang Lokal

Anda harus menghentikan tautan LAG lapisan-2 (dan VLAN-nya) pada sakelar CND yang terpasang langsung dan mengonfigurasi antarmuka IP dan mengintip BGP pada sakelar CND. Anda tidak boleh menjembatani VLAN LAG antara sakelar pusat data Anda. Untuk informasi selengkapnya, lihat Konektivitas lapisan jaringan di Panduan AWS Outposts Pengguna.

Di dalam Outpost multi-rak yang logis, OND saling berhubungan secara berlebihan untuk menyediakan konektivitas jaringan yang sangat tersedia antara rak dan beban kerja yang berjalan di server. AWS bertanggung jawab atas ketersediaan jaringan di dalam Outpost.

Praktik yang direkomendasikan untuk lampiran jaringan yang sangat tersedia

- Hubungkan setiap Outpost Networking Device (OND) di rak Outpost ke Customer Networking Device (CND) terpisah di pusat data.
- Hentikan tautan lapisan-2, VLAN, subnet IP lapisan-3, dan pengintaian BGP pada sakelar Perangkat Jaringan Pelanggan (CND) yang terpasang langsung. Jangan menjembatani OND ke CND VLAN antara CND atau di seluruh jaringan lokal.

- Tambahkan tautan ke Grup Agregasi Tautan (LAG) untuk meningkatkan bandwidth yang tersedia antara Outpost dan pusat data. Jangan mengandalkan bandwidth agregat dari beragam jalur melalui kedua OND.
- Gunakan beragam jalur melalui OND redundan untuk menyediakan konektivitas yang tangguh antara jaringan Outpost dan jaringan lokal.
- Untuk mencapai redundansi yang optimal dan memungkinkan pemeliharaan OND yang tidak mengganggu, kami menyarankan agar pelanggan mengonfigurasi iklan dan kebijakan BGP sebagai berikut:
  - Peralatan jaringan pelanggan harus menerima iklan BGP dari Outpost tanpa mengubah atribut BGP dan untuk memungkinkan BGP multipath/load-balancing untuk mencapai arus lalu lintas masuk yang optimal (dari pelanggan menuju Outpost). As-path prepending digunakan untuk awalan Outpost BGP untuk mengalihkan lalu lintas dari ond/UpLink tertentu jika pemeliharaan diperlukan. Jaringan pelanggan harus memilih rute dari Outpost dengan panjang AS-path 1 daripada rute dengan panjang AS-path 4, yaitu bereaksi terhadap as-Path prepending.
  - Jaringan pelanggan harus mengiklankan awalan BGP yang sama dengan atribut yang sama terhadap semua OND di Outpost. Secara default, beban jaringan Outpost menyeimbangkan lalu lintas keluar (menuju pelanggan) di antara semua uplink. Kebijakan perutean digunakan di sisi Outpost untuk mengalihkan lalu lintas dari OND tertentu jika pemeliharaan diperlukan. Awalan BGP yang sama dari sisi pelanggan di semua OND diperlukan untuk melakukan pergeseran lalu lintas ini, dan melakukan pemeliharaan dengan cara yang tidak mengganggu. Ketika pemeliharaan diperlukan pada jaringan pelanggan, sebaiknya gunakan AS-path prepending untuk sementara mengalihkan lalu lintas dari uplink atau perangkat tertentu.

Konektivitas jangkar

Tautan Layanan Outpost terhubung ke jangkar publik atau pribadi (bukan keduanya) di Availability Zone (AZ) tertentu di Wilayah induk Outpost. Server Outpost memulai koneksi VPN Service Link keluar dari alamat IP Service Link mereka ke titik jangkar di jangkar AZ. Koneksi ini menggunakan UDP dan TCP port 443. AWS bertanggung jawab atas ketersediaan titik jangkar di Wilayah.

Anda harus memastikan alamat IP Outpost Service Link dapat terhubung melalui jaringan Anda ke titik jangkar di jangkar AZ. Alamat IP Service Link tidak perlu berkomunikasi dengan host lain di jaringan lokal Anda.

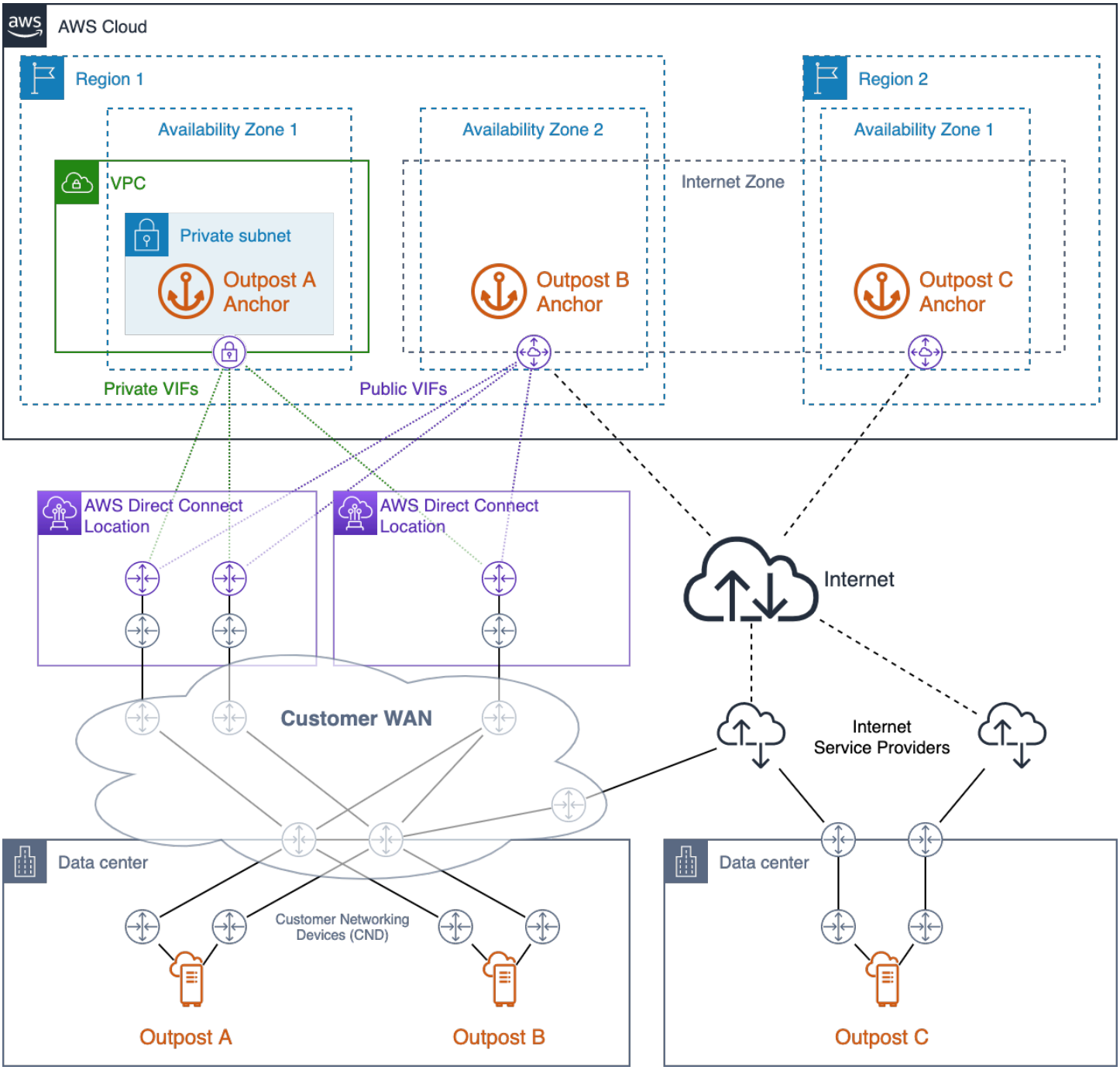
Titik jangkar publik berada di rentang IP publik Wilayah (dalam blok CIDR layanan EC2) dan dapat diakses melalui internet atau AWS Direct Connect(DX) antarmuka virtual publik (VIF). Penggunaan

titik jangkar publik memungkinkan pemilihan jalur yang lebih fleksibel karena lalu lintas Service Link dapat diarahkan ke jalur yang tersedia yang berhasil mencapai titik jangkar di internet publik.

Poin jangkar pribadi memungkinkan Anda menggunakan rentang alamat IP Anda untuk konektivitas jangkar. Poin jangkar pribadi dibuat dalam subnet pribadi dalam VPC khusus menggunakan alamat IP yang ditetapkan pelanggan. VPC dibuat di Akun AWS yang memiliki sumber daya Outpost dan Anda bertanggung jawab untuk memastikan VPC tersedia dan dikonfigurasi dengan benar (jangan hapus!). Titik jangkar pribadi harus diakses menggunakan VIF pribadi Direct Connect.

Anda harus menyediakan jalur jaringan redundan antara Outpost dan anchor point di Region dengan koneksi yang berakhir pada perangkat terpisah di lebih dari satu lokasi. Perutean dinamis harus dikonfigurasi untuk secara otomatis mengalihkan lalu lintas ke jalur alternatif ketika koneksi atau perangkat jaringan gagal. Anda harus menyediakan kapasitas jaringan yang cukup untuk memastikan bahwa kegagalan satu jalur WAN tidak membanjiri jalur yang tersisa.

Diagram berikut menunjukkan tiga Outposts dengan jalur jaringan redundan ke anchor AZ mereka menggunakan AWS Direct Connect serta konektivitas internet publik. Pos Terdepan A dan Pos Terdepan B ditambatkan ke Zona Ketersediaan yang berbeda di Wilayah yang sama. Pos Terdepan A terhubung ke titik jangkar pribadi di AZ 1 dari wilayah 1. Outpost B terhubung ke titik jangkar publik di AZ 2 wilayah 1. Outpost C terhubung ke jangkar publik di AZ 1 wilayah 2.



Konektivitas jangkar yang sangat tersedia dengan AWS Direct Connect dan akses internet publik

Outpost A memiliki tiga jalur jaringan redundan untuk mencapai titik jangkar pribadinya. Dua jalur tersedia melalui sirkuit Direct Connect redundan di satu lokasi Direct Connect. Jalur ketiga tersedia melalui sirkuit Direct Connect di lokasi Direct Connect kedua. Desain ini menjaga lalu lintas Service Link Outpost A di jaringan pribadi dan menyediakan redundansi jalur yang memungkinkan kegagalan salah satu sirkuit Direct Connect atau kegagalan seluruh lokasi Direct Connect.

Outpost B memiliki empat jalur jaringan redundan untuk mencapai titik jangkar publiknya. Tiga jalur tersedia melalui VIF publik yang disediakan di sirkuit Direct Connect dan lokasi yang digunakan oleh Outpost A. Jalur keempat tersedia melalui WAN pelanggan dan internet publik. Lalu lintas Link Layanan Outpost B dapat diarahkan melalui jalur yang tersedia yang dapat berhasil mencapai titik jangkar di internet publik. Menggunakan jalur Direct Connect dapat memberikan latensi yang lebih konsisten dan ketersediaan bandwidth yang lebih tinggi, sedangkan jalur internet publik dapat digunakan untuk Disaster Recovery (DR) atau skenario augmentasi bandwidth.

Outpost C memiliki dua jalur jaringan redundan untuk mencapai titik jangkar publiknya. Outpost C ditempatkan di pusat data yang berbeda dari pusat data Outposts A dan B. Outpost C tidak memiliki sirkuit khusus yang terhubung ke WAN pelanggan. Sebaliknya, pusat data memiliki koneksi internet redundan yang disediakan oleh dua Penyedia Layanan Internet (ISP) yang berbeda. Lalu lintas Service Link Outpost C dapat diarahkan melalui salah satu jaringan ISP untuk mencapai titik jangkar di internet publik. Desain ini memungkinkan fleksibilitas untuk mengarahkan lalu lintas Service Link melalui koneksi internet publik yang tersedia. Namun, end-to-end jalur tergantung pada jaringan pihak ketiga publik di mana ketersediaan bandwidth dan latensi jaringan berfluktuasi.

Jalur jaringan antara Outpost dan titik jangkar Service Link-nya harus memenuhi spesifikasi bandwidth dan latensi berikut:

- 500 Mbps - 1 Gbps bandwidth yang tersedia per rak Outpost (misalnya, 3 rak: 1,5 - 3 Gbps bandwidth yang tersedia)
- Kurang dari 300 milidetik (pulang-pergi) latensi

Praktik yang direkomendasikan untuk konektivitas jangkar yang sangat tersedia:

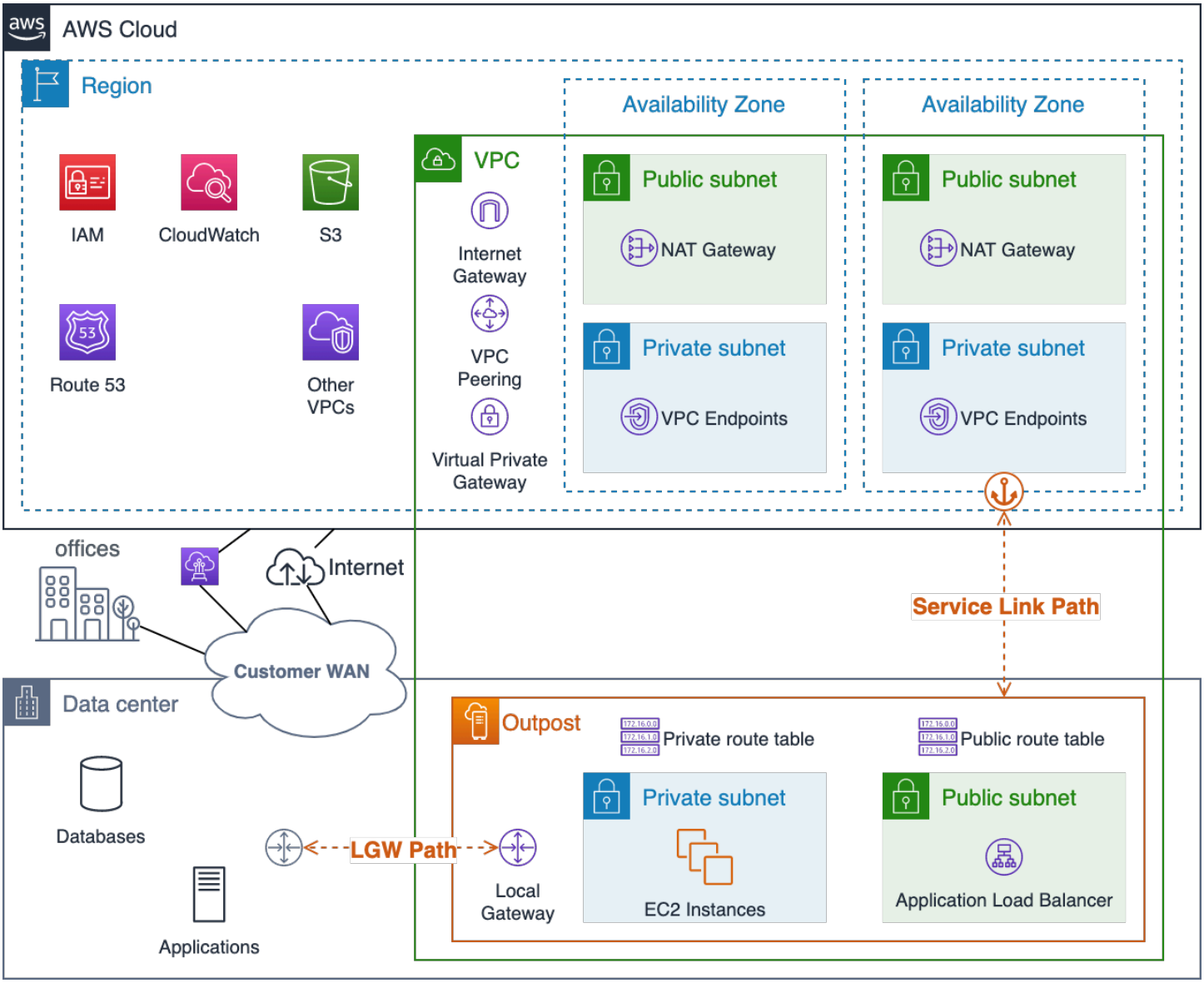
- Menyediakan jalur jaringan yang berlebihan antara setiap Pos Luar dan titik jangkar di Wilayah.
- Gunakan jalur Direct Connect (DX) untuk mengontrol latensi dan ketersediaan bandwidth.
- Pastikan bahwa port TCP dan UDP 443 terbuka (keluar) dari blok CIDR Outpost Service Link ke rentang alamat IP EC2 di Wilayah induk. Pastikan port terbuka di semua jalur jaringan.
- Pastikan setiap jalur memenuhi ketersediaan bandwidth dan persyaratan latensi.
- Gunakan perutean dinamis untuk mengotomatiskan pengalihan lalu lintas di sekitar kegagalan jaringan.
- Uji perutean lalu lintas Service Link melalui setiap jalur jaringan yang direncanakan untuk memastikan jalur berfungsi seperti yang diharapkan.

Perutean aplikasi/beban kerja

Ada dua jalur keluar dari Outpost untuk beban kerja aplikasi:

- Jalur Tautan Layanan
- Jalur Gerbang Lokal (LGW)

Anda mengonfigurasi tabel rute subnet Outpost untuk mengontrol jalur mana yang harus diambil untuk mencapai jaringan tujuan. Rute yang diarahkan ke LGW akan mengarahkan lalu lintas keluar dari Gateway Lokal dan ke jaringan lokal. Rute yang menunjuk ke target di Wilayah seperti Internet Gateways, NAT Gateways, Virtual Private Gateways, dan koneksi peering VPC akan mengarahkan lalu lintas melintasi Tautan Layanan untuk mencapai target ini.



Visualisasi Outpost Service Link dan jalur jaringan LGW

Anda harus berhati-hati saat merencanakan perutean aplikasi untuk mempertimbangkan operasi normal dan perutean terbatas dan ketersediaan layanan selama kegagalan jaringan. Jalur Tautan Layanan tidak tersedia saat Pos Luar terputus dari Wilayah.

Anda harus menyediakan beragam jalur dan mengonfigurasi perutean dinamis antara Outpost LGW dan aplikasi, sistem, dan pengguna lokal penting Anda. Jalur jaringan redundan memungkinkan jaringan untuk merutekan lalu lintas di sekitar kegagalan dan memastikan bahwa sumber daya lokal akan dapat berkomunikasi dengan beban kerja yang berjalan di Outpost selama kegagalan jaringan sebagian.

Konfigurasi rute VPC pos terdepan bersifat statis. Anda mengonfigurasi tabel perutean subnet melalui, AWS Management Console CLI, API, dan alat Infrastruktur sebagai Kode (IAc) lainnya; Namun, Anda tidak akan dapat memodifikasi tabel perutean subnet selama peristiwa pemutusan sambungan. Anda harus membangun kembali konektivitas antara Outpost dan Region untuk memperbarui tabel rute. Gunakan rute yang sama untuk operasi normal seperti yang Anda rencanakan untuk digunakan selama peristiwa pemutusan sambungan.

Sumber daya di Outpost dapat menjangkau internet melalui Service Link dan Internet Gateway (IGW) di Wilayah atau melalui jalur Local Gateway (LGW). Merutekan lalu lintas internet melalui jalur LGW dan jaringan lokal memungkinkan Anda menggunakan titik masuk/keluar internet lokal yang ada dan dapat memberikan latensi yang lebih rendah, MTU yang lebih tinggi, dan pengurangan biaya keluar AWS data jika dibandingkan dengan menggunakan jalur Tautan Layanan ke IGW di Wilayah.

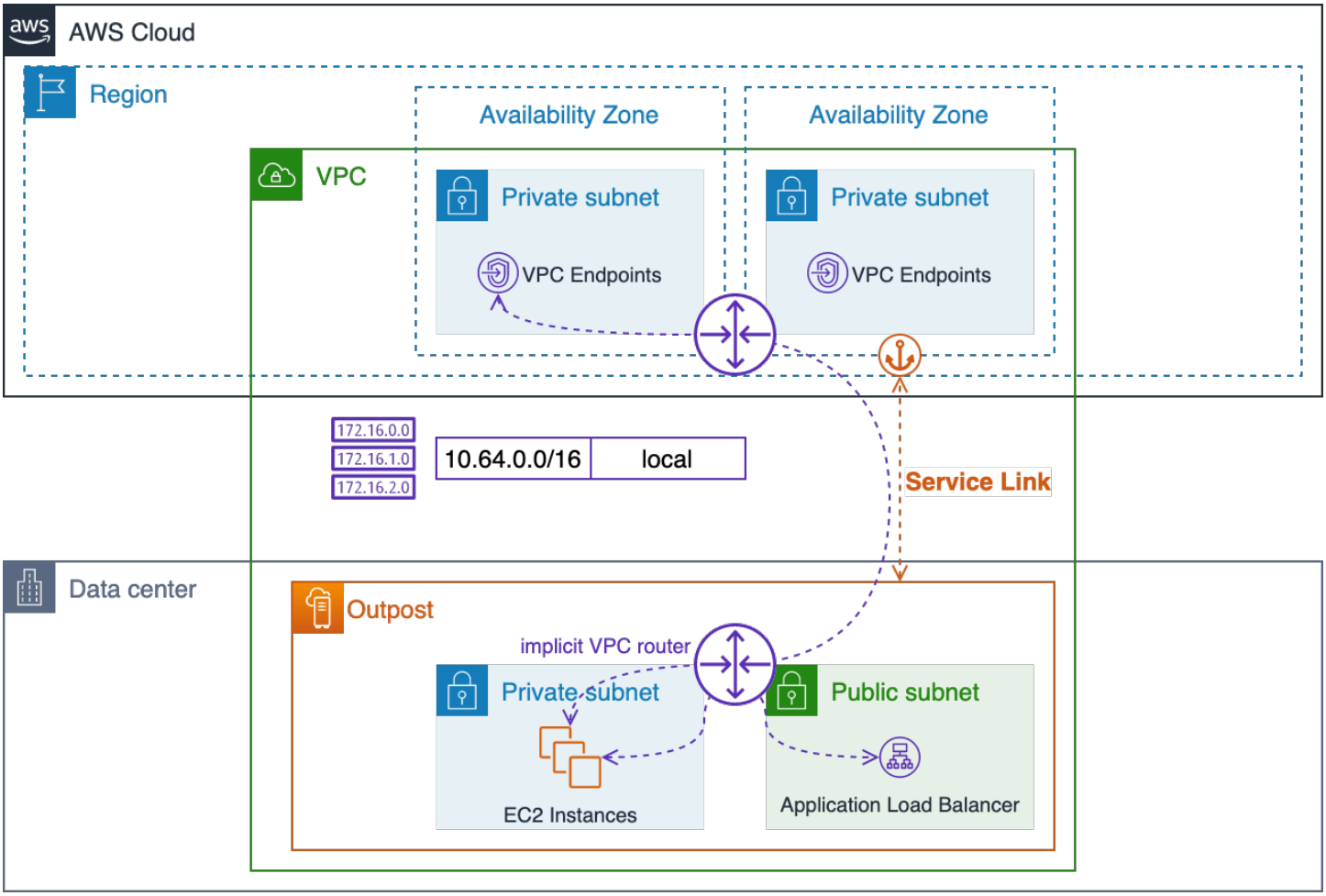
Jika aplikasi Anda harus berjalan di tempat dan harus dapat diakses dari internet publik, Anda harus merutekan lalu lintas aplikasi melalui koneksi internet lokal Anda ke LGW untuk menjangkau sumber daya di Pos Luar.

Meskipun Anda dapat mengonfigurasi subnet di Outpost seperti subnet publik di Wilayah, ini mungkin merupakan praktik yang tidak diinginkan untuk sebagian besar kasus penggunaan. Lalu lintas internet masuk akan masuk melalui Wilayah AWS dan diarahkan melalui Tautan Layanan ke sumber daya yang berjalan di Outpost.

Lalu lintas respons pada gilirannya akan diarahkan melalui Tautan Layanan dan kembali keluar melalui koneksi internet. Wilayah AWS Pola lalu lintas ini dapat menambah latensi dan akan menimbulkan biaya keluar data karena lalu lintas meninggalkan Wilayah dalam perjalanan ke Pos Terdepan dan ketika lalu lintas kembali kembali melalui Wilayah dan keluar ke internet. Jika aplikasi Anda dapat berjalan di Wilayah, Wilayah adalah tempat terbaik untuk menjalankannya.

Lalu lintas antara sumber daya VPC (dalam VPC yang sama) akan selalu mengikuti rute CIDR VPC lokal dan dirutekan antar subnet oleh router VPC implisit.

Misalnya, lalu lintas antara instans EC2 yang berjalan di Outpost dan Titik Akhir VPC di Wilayah akan selalu dirutekan melalui Tautan Layanan.



Perutean VPC lokal melalui router implisit

Praktik yang direkomendasikan untuk perutean aplikasi/beban kerja:

- Gunakan jalur Local Gateway (LGW) alih-alih jalur Service Link jika memungkinkan.
- Rutekan lalu lintas internet melalui jalur LGW.
- Konfigurasikan tabel perutean subnet Outpost dengan serangkaian rute standar - mereka akan digunakan untuk operasi normal dan selama peristiwa pemutusan sambungan.
- Menyediakan jalur jaringan redundan antara Outpost LGW dan sumber daya aplikasi lokal yang penting. Gunakan perutean dinamis untuk mengotomatiskan pengalihan lalu lintas di sekitar kegagalan jaringan lokal.

Hitung

Sementara kapasitas Amazon EC2 tampaknya tak terbatas, kapasitas di Wilayah AWS Outposts terbatas. Anda bertanggung jawab untuk merencanakan dan mengelola kapasitas komputasi penyebaran Outposts Anda.

Topik

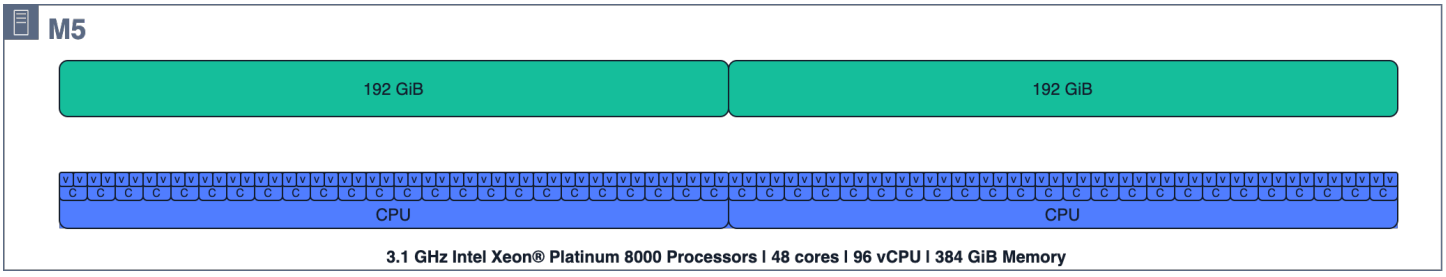
- Perencanaan kapasitas
- Manajemen kapasitas
- Penempatan instans

Perencanaan kapasitas

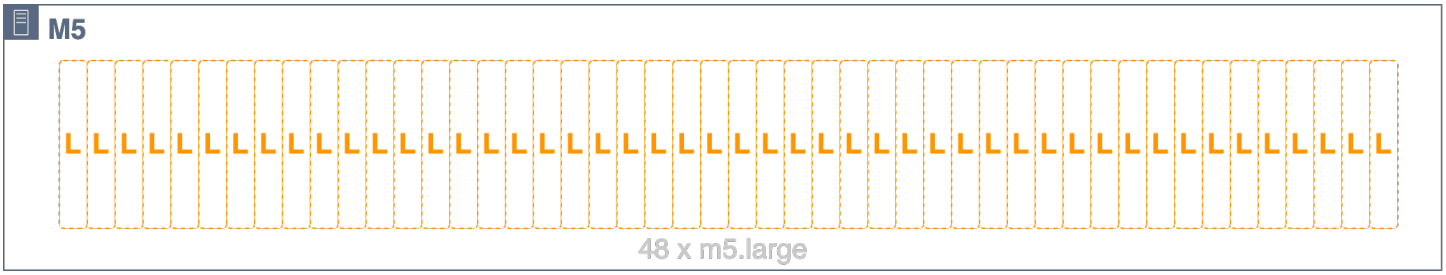
Sementara kapasitas Amazon EC2 tampaknya tak terbatas, kapasitas di Wilayah AWS Outposts terbatas — dibatasi oleh total volume kapasitas komputasi yang dipesan. Anda bertanggung jawab untuk merencanakan dan mengelola kapasitas komputasi penyebaran Outposts Anda. Anda harus memesan kapasitas komputasi yang cukup untuk mendukung model ketersediaan N+M, di mana N adalah jumlah server yang diperlukan dan M adalah jumlah server cadangan yang disediakan untuk mengakomodasi kegagalan server. N+1 dan N+2 adalah tingkat ketersediaan yang paling umum.

Setiap server (`C5`,`M5`,`R5`, dll.) mendukung satu keluarga instance EC2. Sebelum dapat meluncurkan instance di server komputasi EC2, Anda harus menyediakan tata letak slotting yang menentukan ukuran instans EC2 yang ingin disediakan oleh setiap server. AWS mengkonfigurasi setiap server dengan tata letak slotting yang diminta.

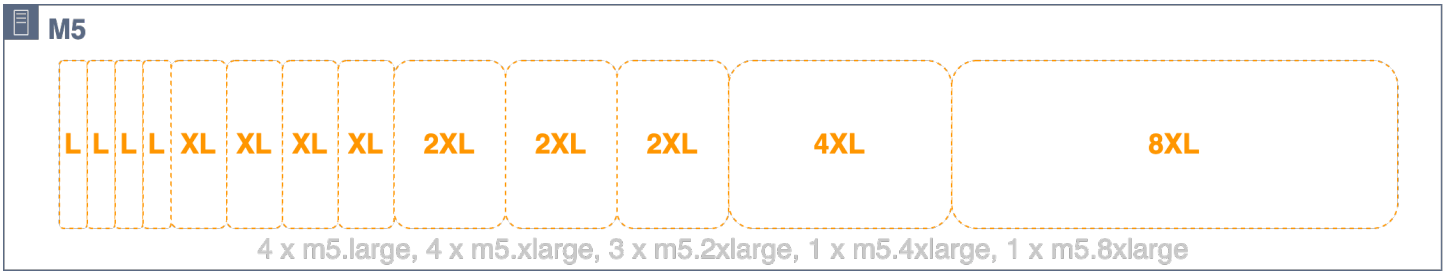
Server dapat ditempatkan secara homogen di mana semua slot memiliki ukuran instance yang sama (misalnya, 48 `m5.large` slot) atau ditempatkan secara heterogen dengan campuran jenis instance (misalnya, 4, 4`m5.large`, 3, 1 `m5.2xlarge`m5.4xlarge, dan 1`m5.8xlarge`) — lihat tiga gambar berikutnya untuk visualisasi konfigurasi slotting ini. `m5.xlarge`



*`m5.24xlarge`*sumber daya komputasi server



*m5.24xlarge*server secara homogen dimasukkan ke dalam 48 slot *m5.large*



*m5.24xlarge*server secara heterogen ditempatkan menjadi 4*m5.large*, 4, 3*m5.2xlarge*, 1*m5.xlarge*, dan 1 *m5.4xlarge* slot *m5.8xlarge*

Kapasitas server penuh tidak harus ditempatkan. Slot dapat ditambahkan ke server yang memiliki kapasitas yang tidak terisi. Anda memodifikasi tata letak slotting dengan membuka tiket dukungan. Enterprise Support mungkin mengharuskan Anda untuk mematikan atau memulai ulang instans tertentu untuk menyelesaikan permintaan reslotting jika tata letak slotting baru tidak dapat diterapkan saat slot tertentu ditempati oleh instance yang sedang berjalan.

Semua server menyumbangkan slot yang disediakan ke kumpulan kapasitas EC2 di Outpost, dan semua slot dari jenis dan ukuran instans tertentu dikelola sebagai kumpulan kapasitas EC2 tunggal. Misalnya, server slotted heterogen sebelumnya dengan`m5.large`,,, `m5.xlarge` `m5.2xlarge``m5.4xlarge`, dan `m5.8xlarge` slot akan menyumbangkan slot ini ke lima kumpulan kapasitas EC2 — satu kumpulan untuk setiap jenis dan ukuran instans.

Penting untuk mempertimbangkan slotting server dan kumpulan kapasitas EC2 saat merencanakan kapasitas cadangan untuk ketersediaan server N+M. AWS mendeteksi ketika server gagal atau terdegradasi dan menjadwalkan kunjungan situs untuk menggantikan server yang gagal. Anda harus merancang kumpulan kapasitas EC2 Anda untuk mentolerir kegagalan setidaknya satu server dari setiap keluarga instance (N+1) di Outpost. Dengan tingkat ketersediaan server minimum ini, ketika server gagal atau perlu dikeluarkan dari layanan, Anda dapat memulai ulang instance yang gagal atau terdegradasi pada slot cadangan server yang tersisa dari keluarga yang sama.

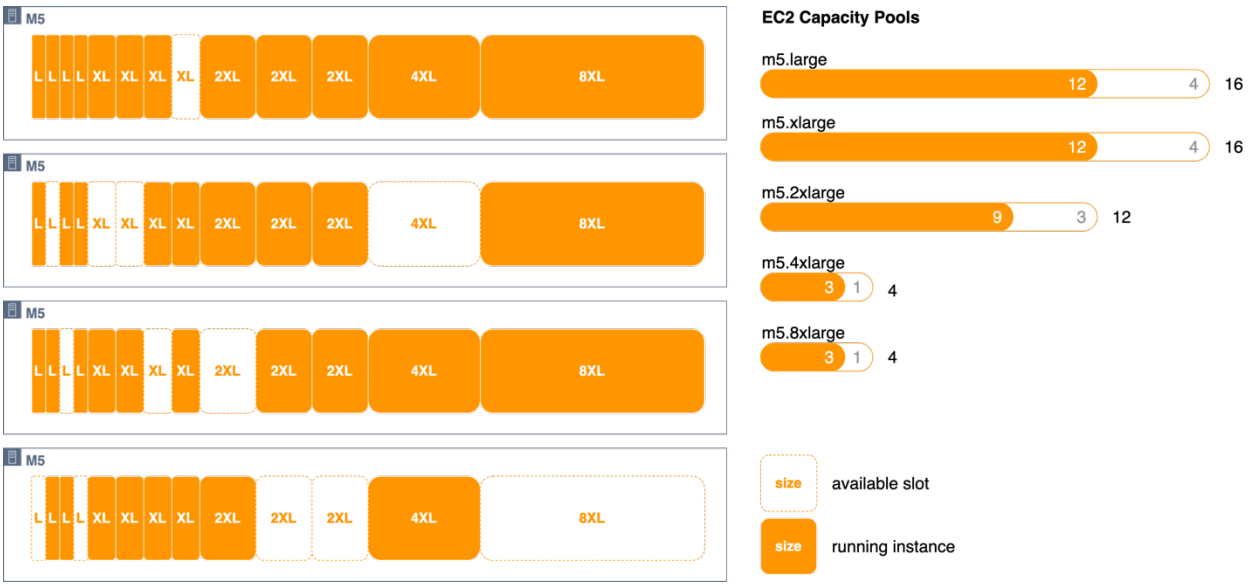
Perencanaan untuk ketersediaan N+M sederhana ketika Anda memiliki server slotting homogen atau kelompok server slotted heterogen dengan tata letak slotting yang identik. Anda cukup menghitung jumlah server (N) kebutuhan Anda untuk menjalankan semua beban kerja Anda dan kemudian menambahkan (M) server tambahan untuk memenuhi persyaratan Anda untuk ketersediaan server selama kegagalan dan peristiwa pemeliharaan.

Konfigurasi slotting berikut tidak dapat digunakan karena batas NUMA:

- 3 `m5.8xlarge`
- 1 `m5.16xlarge` dan 1 `m5.8xlarge`

Konsultasikan dengan Akun AWS tim Anda untuk memvalidasi konfigurasi slot AWS Outposts rak yang Anda rencanakan.

Pada gambar berikut, empat `m5.24xlarge` server ditempatkan secara heterogen dengan tata letak slotting yang identik. Keempat server membuat lima kolam kapasitas EC2. Setiap pool berjalan pada pemanfaatan maksimum (75%) untuk menjaga ketersediaan N+1 untuk instance yang berjalan di keempat server ini. Jika ada server yang gagal, ada cukup ruang untuk memulai ulang instance yang gagal di server yang tersisa.



Visualisasi slot server EC2, instance berjalan, dan kumpulan slot

Untuk tata letak slotting yang lebih kompleks, di mana server tidak ditempatkan secara identik, Anda perlu menghitung ketersediaan N+M untuk setiap kumpulan kapasitas EC2. Anda dapat menggunakan rumus berikut untuk menghitung berapa banyak server (yang berkontribusi slot ke

kumpulan kapasitas EC2 tertentu) yang dapat gagal dan masih mengizinkan server yang tersisa untuk membawa instance yang sedang berjalan:

$$M = \left\lceil \frac{poolSlots_{available}}{serverSlots_{max}} \right\rceil$$

Di mana:

- PoolSlots $_{available}$ adalah jumlah slot yang tersedia di kumpulan kapasitas EC2 yang diberikan (jumlah total slot di kolam dikurangi jumlah instance yang berjalan)
- ServerSlots $_{max}$ adalah jumlah maksimum slot yang disumbangkan oleh server manapun ke kumpulan kapasitas EC2 yang diberikan
- M adalah jumlah server yang dapat gagal dan masih memungkinkan server yang tersisa untuk membawa instance yang sedang berjalan

Contoh: Sebuah Outpost memiliki tiga server yang menyumbangkan slot ke kolam `m5.2xlarge` kapasitas. Yang pertama menyumbang 4 slot, yang kedua menyumbang 3 slot, dan server ketiga menyumbang 2 slot. Kolam `m5.2xlarge` instance di Outpost memiliki kapasitas total 9 slot (4 + 3 + 2). Outpost memiliki 4 `m5.2xlarge` instance yang berjalan. Berapa banyak server yang mungkin gagal dan masih mengizinkan server yang tersisa untuk membawa instance yang sedang berjalan?

$$poolSlots_{available} = total\ capacity - running\ instances = 9 - 4 = 5$$

$$serverSlots_{max} = \max([4, 3, 2]) = 4$$

$$M = \left\lceil \frac{poolSlots_{available}}{serverSlots_{max}} \right\rceil = \left[\frac{5}{4}\right] = [1.25] = 1$$

Jawaban: Anda dapat kehilangan salah satu server dan masih membawa instance yang berjalan di server yang tersisa.

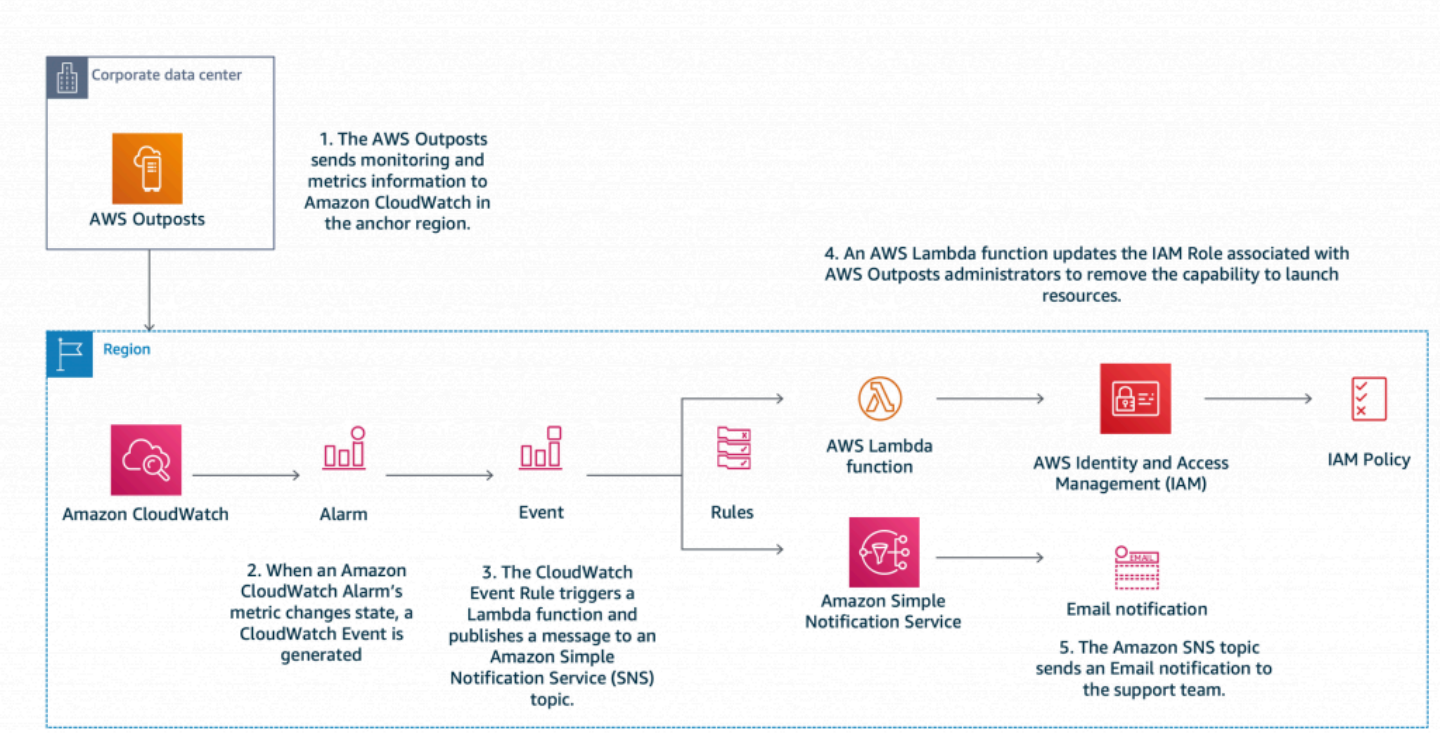
Praktik yang direkomendasikan untuk perencanaan kapasitas komputasi:

- Ukur kapasitas komputasi Anda untuk memberikan redundansi N+M untuk setiap kumpulan kapasitas EC2 di Outpost.
  - Menyebarkan server N+M untuk server slotted heterogen yang homogen atau identik.
  - Hitung ketersediaan N+M untuk setiap kumpulan kapasitas EC2 dan pastikan bahwa setiap kolam memenuhi persyaratan ketersediaan Anda.

Manajemen kapasitas

Anda dapat memantau pemanfaatan kumpulan instans Outpost EC2 di dan AWS Management Console melalui metrik Amazon. CloudWatch Hubungi Enterprise Support untuk mengambil atau mengubah tata letak slotting untuk Outposts Anda.

Anda menggunakan pemulihan otomatis instans yang sama dan mekanisme EC2 Auto Scaling untuk memulihkan atau mengganti instance yang terkena dampak kegagalan server dan peristiwa pemeliharaan. Anda harus memantau dan mengelola kapasitas Outpost Anda untuk memastikan kapasitas cadangan yang cukup selalu tersedia untuk mengakomodasi kegagalan server. Mengelola AWS Outposts kapasitas Anda menggunakan Amazon CloudWatch dan posting AWS Lambda blog menyediakan tutorial langsung yang menunjukkan cara menggabungkan AWS CloudWatch dan mengelola kapasitas Outpost Anda AWS Lambda untuk mempertahankan ketersediaan instans.



Mengelola AWS Outposts kapasitas dengan Amazon CloudWatch dan AWS Lambda

Praktik yang direkomendasikan untuk manajemen kapasitas komputasi:

- Konfigurasikan instans EC2 Anda di grup Auto Scaling atau gunakan pemulihan otomatis instans untuk memulai ulang instans yang gagal.
- Otomatiskan pemantauan kapasitas untuk penerapan Outpost Anda dan konfigurasikan notifikasi dan (opsional) respons otomatis untuk alarm kapasitas.

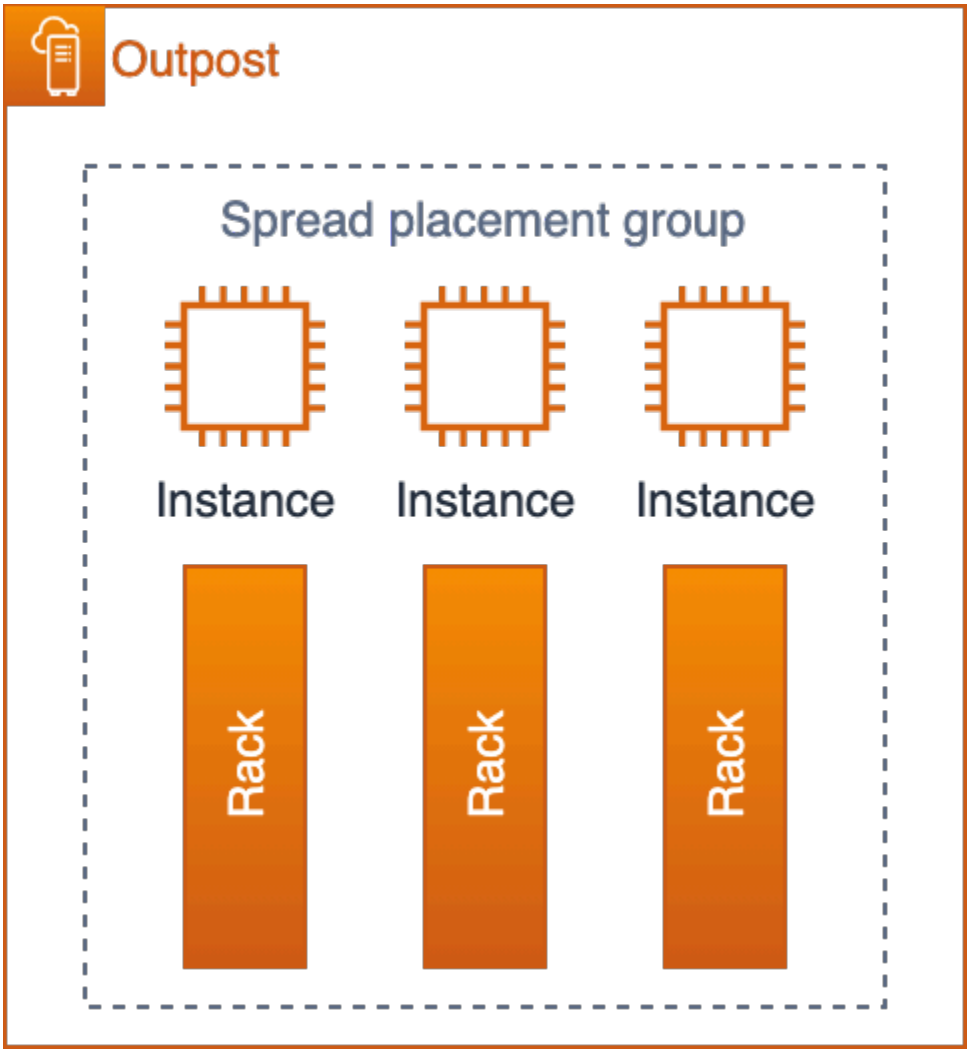
Penempatan instans

Outposts memiliki jumlah server komputasi yang terbatas. Jika aplikasi Anda menyebarkan beberapa instance terkait di Outposts; tanpa konfigurasi tambahan, instance dapat diterapkan di server yang sama atau di server di rak yang sama. Saat ini, ada tiga mekanisme yang dapat Anda gunakan untuk mendistribusikan instans guna mengurangi risiko menjalankan instans terkait pada infrastruktur yang sama:

Penerapan Multi-Outpost — mirip dengan strategi Multi-AZ di Wilayah, Anda dapat menyebarkan Outposts untuk memisahkan pusat data dan menyebarkan sumber daya aplikasi ke Outposts tertentu. Ini memungkinkan Anda untuk menjalankan instance di Outpost yang diinginkan (satu set rak logis). Strategi Multi-Outpost dapat digunakan untuk melindungi terhadap mode kegagalan rak dan pusat data dan, jika Outposts ditambatkan ke AZ atau Wilayah yang terpisah, juga dapat memberikan perlindungan terhadap mode kegagalan AZ atau Wilayah. Untuk informasi selengkapnya tentang arsitektur Multi-Outpost, lihat Mode Kegagalan yang Lebih Besar.

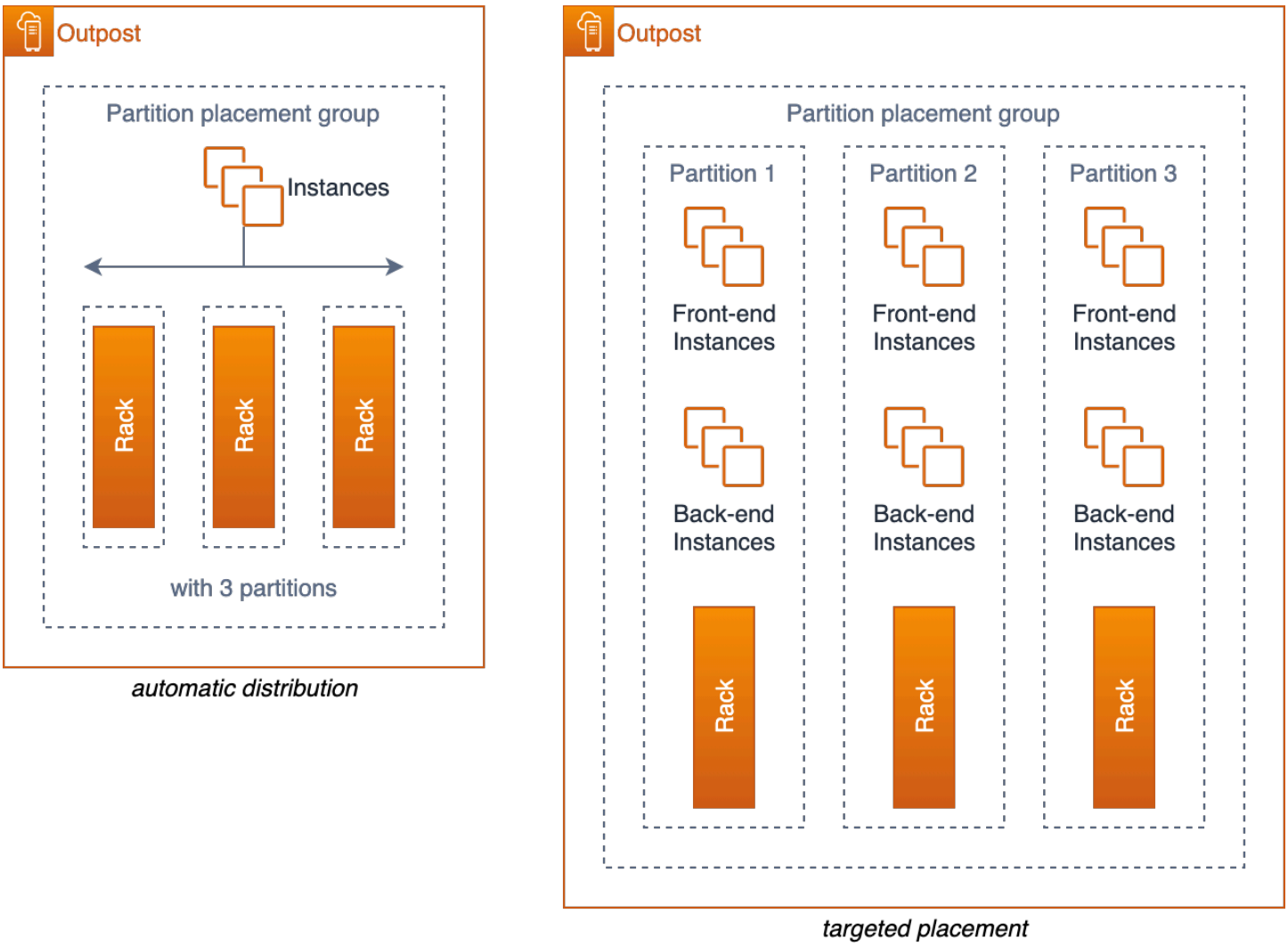
Grup penempatan Amazon EC2 di Outposts (penempatan instans multi-rak Single-outpost) — memungkinkan Anda menggunakan strategi cluster, spread, dan partisi untuk memengaruhi penempatan. Strategi penyebaran dan penempatan partisi memungkinkan Anda mendistribusikan instance di seluruh rak di Outpost multi-rak.

Grup penempatan spread menyediakan cara sederhana mendistribusikan instance tunggal di seluruh rak untuk mengurangi potensi kegagalan yang berkorelasi. Anda hanya dapat menyebarkan ke dalam grup sebanyak yang Anda miliki rak di Pos Luar Anda.



Grup penempatan spread EC2 di Pos Terdepan dengan tiga rak

Anda juga dapat mendistribusikan instance di beberapa rak dengan grup penempatan partisi. Gunakan distribusi otomatis untuk menyebarkan instance di seluruh partisi dalam grup atau menyebarkan instance ke partisi target yang dipilih. Menerapkan instance ke partisi target memungkinkan Anda menyebarkan sumber daya yang dipilih ke rak yang sama sambil mendistribusikan sumber daya lain di seluruh rak. Misalnya, jika Anda memiliki Outpost logis dengan tiga rak, membuat grup penempatan partisi dengan tiga partisi memungkinkan Anda untuk mendistribusikan sumber daya di seluruh rak.



Grup penempatan partisi EC2 di Outpost dengan tiga rak

Creative server slotting — jika Anda memiliki Outpost rak tunggal atau jika layanan yang Anda gunakan di Outposts tidak mendukung grup penempatan, Anda mungkin dapat menggunakan slotting kreatif untuk memastikan instans Anda tidak disebarkan pada server fisik yang sama. Jika instance terkait memiliki ukuran instans EC2 yang sama, Anda mungkin dapat membuat slot server Anda untuk membatasi jumlah slot dengan ukuran yang dikonfigurasi pada setiap server — menyebarkan slot di seluruh server. Server slotting akan membatasi jumlah instance (dari ukuran itu) yang dapat berjalan pada satu server.

Sebagai contoh, pertimbangkan tata letak slotting yang ditunjukkan sebelumnya pada Gambar 13. Jika aplikasi Anda perlu menerapkan tiga `m5.4xlarge` instance di Outpost yang dikonfigurasi dengan tata letak slotting ini, EC2 akan menempatkan setiap instance di server terpisah dan tidak

akan ada kemungkinan bahwa instance ini dapat berjalan di server yang sama — selama konfigurasi slotting tidak berubah untuk membuka slot tambahan di server. `m5.4xlarge`

Praktik yang disarankan untuk penempatan instans komputasi:

- Gunakan grup penempatan Amazon EC2 di Outposts untuk mengontrol penempatan instans di seluruh rak dalam satu Outpost.
- Alih-alih memesan Outpost dengan rak Outpost tunggal sedang atau besar, pertimbangkan untuk membagi kapasitas menjadi dua rak kecil atau sedang untuk memungkinkan Anda memanfaatkan kemampuan grup penempatan EC2 untuk mendistribusikan instance di seluruh rak.

Penyimpanan

Layanan AWS Outposts rak menyediakan tiga jenis penyimpanan:

- Penyimpanan instans pada jenis instans EC2 yang didukung
- Volume Amazon Elastic Block Store (EBS) gp2 untuk penyimpanan blok persisten
- Amazon Simple Storage Service di Outposts (S3 di Outposts) untuk penyimpanan objek lokal

Penyimpanan instans disediakan pada server yang didukung (`C5d`,`M5d`,`R5d`,`G4dn`, dan`I3en`). Sama seperti di Wilayah, data dalam penyimpanan instance hanya bertahan selama masa pakai (berjalan) instance.

Outposts Volume EBS dan S3 on Outposts penyimpanan objek disediakan sebagai bagian dari layanan yang dikelola rak. AWS Outposts Pelanggan bertanggung jawab atas manajemen kapasitas kolam penyimpanan Outpost. Pelanggan menentukan persyaratan penyimpanan mereka untuk penyimpanan EBS dan S3 saat memesan Outpost. AWS mengkonfigurasi Outpost dengan jumlah server penyimpanan yang diperlukan untuk menyediakan kapasitas penyimpanan yang diminta. AWS bertanggung jawab atas ketersediaan layanan penyimpanan EBS dan S3 pada Outposts. Server penyimpanan yang memadai disediakan untuk menyediakan layanan penyimpanan yang sangat tersedia untuk Outpost. Kehilangan server penyimpanan tunggal seharusnya tidak mengganggu layanan atau mengakibatkan kehilangan data.

Anda dapat menggunakan CloudWatch metrik AWS Management Console dan untuk memantau Outpost EBS dan S3 pada pemanfaatan kapasitas Outposts.

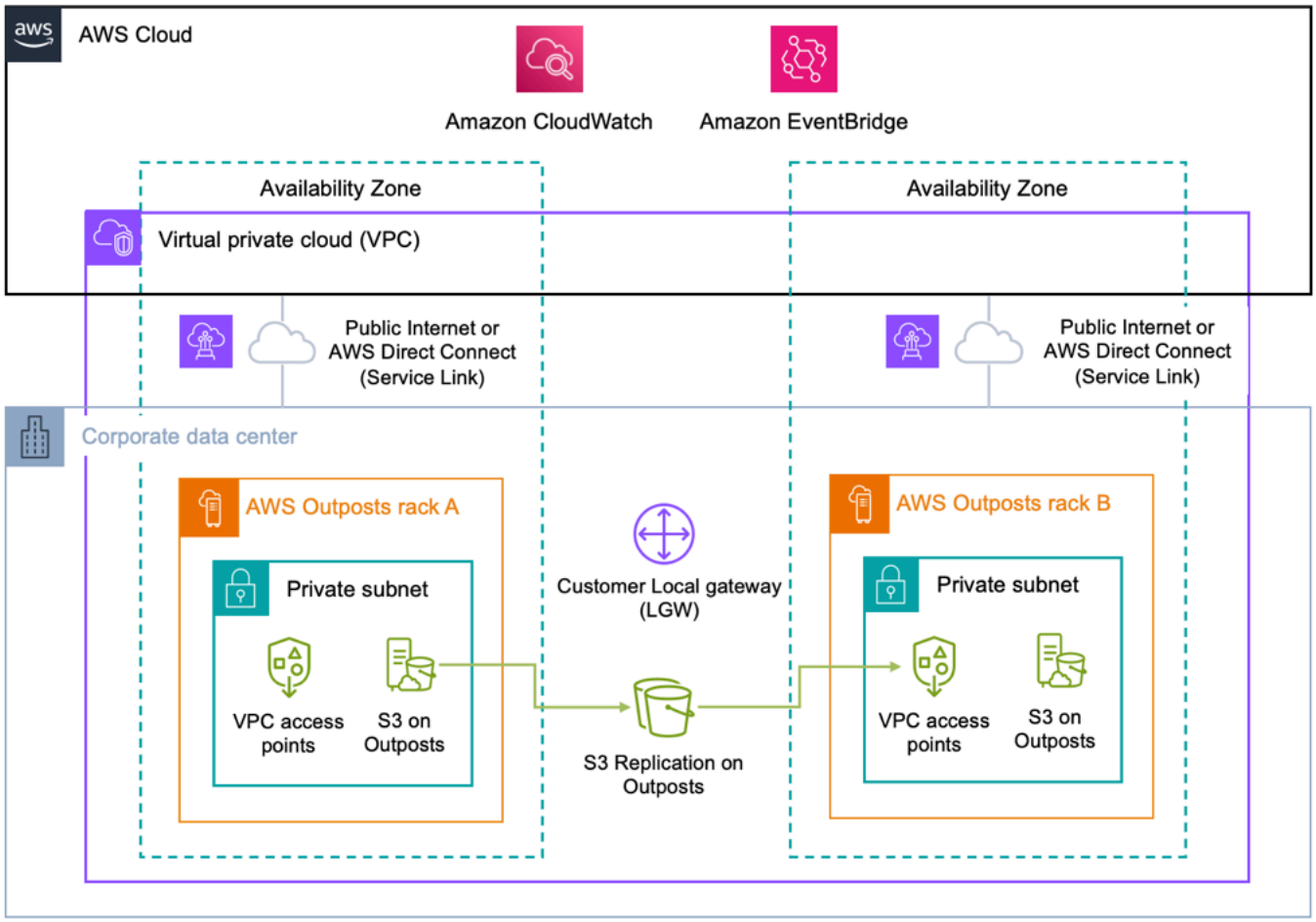
Perlindungan data

Untuk Volume EBS: AWS Outposts rack mendukung snapshot volume EBS untuk menyediakan mekanisme perlindungan data yang sederhana dan aman untuk melindungi data penyimpanan blok Anda. Snapshot adalah backup point-in-time tambahan dari volume EBS Anda. Secara default, snapshot volume Amazon EBS di Outpost Anda disimpan di Amazon S3 di Wilayah. Jika Outposts Anda telah dikonfigurasi dengan S3 pada kapasitas Outposts, Anda dapat menggunakan EBS Local Snapshots on Outposts untuk menyimpan snapshot secara lokal di Outpost Anda menggunakan S3 pada penyimpanan Outposts.

Untuk ember S3 pada Outposts (kasus penggunaan residensi data):

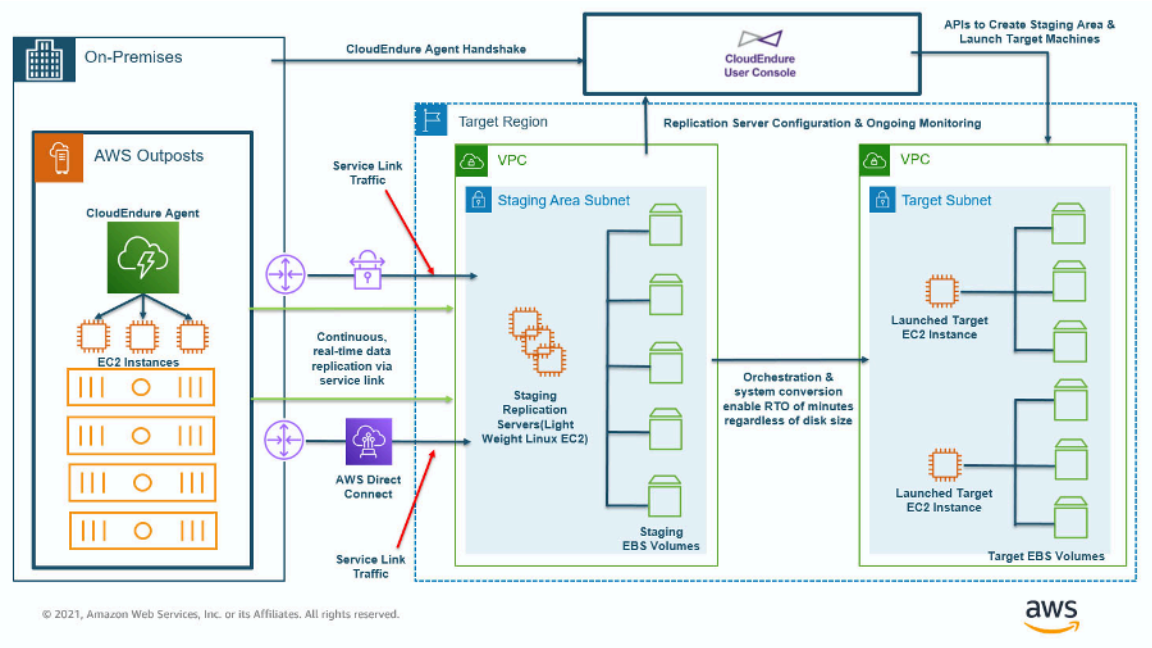
- Anda dapat menggunakan S3 Versioning di Outposts, untuk menyimpan semua perubahan, dan riwayat objek. Saat diaktifkan, S3 Versioning menyimpan beberapa salinan objek yang berbeda dalam bucket yang sama. Anda dapat menggunakan Versioning S3 untuk menyimpan, mengambil, dan memulihkan setiap versi dari setiap objek yang disimpan dalam bucket Outposts Anda. Versioning S3 membantu Anda memulihkan dari tindakan pengguna yang tidak diinginkan dan kegagalan aplikasi.
- Anda dapat menggunakan Replikasi S3 di Outposts, untuk membuat dan mengonfigurasi aturan replikasi agar secara otomatis mereplikasi objek S3 Anda ke Outpost lain, atau ke bucket lain di Outpost yang sama. Selama replikasi, objek S3 pada Outposts dikirim melalui gateway lokal pelanggan (LGW), dan objek tidak melakukan perjalanan kembali ke. Wilayah AWS Replikasi S3 di Outposts menyediakan cara yang mudah dan fleksibel untuk secara otomatis mereplikasi data dalam perimeter data tertentu untuk mengatasi redundansi data dan persyaratan kepatuhan.

  Replikasi S3 di Outposts juga menyediakan metrik dan notifikasi terperinci untuk memantau status replikasi objek Anda. Anda dapat memantau kemajuan replikasi dengan melacak byte yang tertunda, operasi tertunda, dan latensi replikasi antara bucket Outposts sumber dan tujuan menggunakan Amazon. CloudWatch Anda juga dapat mengatur EventBridge aturan Amazon untuk menerima peristiwa kegagalan replikasi untuk mendiagnosis dan memperbaiki masalah konfigurasi dengan cepat.



Untuk bucket S3 on Outposts (kasus penggunaan residensi non-data) Wilayah AWS ke: Anda dapat menggunakan AWS DataSyncuntuk mengotomatiskan transfer data S3 pada Outposts antara Outpost dan Region. DataSync memungkinkan Anda memilih apa yang akan ditransfer, kapan harus mentransfer, dan berapa banyak bandwidth yang akan digunakan. Mencadangkan bucket S3 di Outposts lokal Anda ke bucket S3 di dalamnya Wilayah AWS memungkinkan Anda memanfaatkan 99,99999999999% (11 9's) daya tahan data dan tingkatan penyimpanan tambahan (Standar, Akses Jarang, dan Gletser) untuk pengoptimalan biaya yang tersedia dengan layanan S3 regional.

Replikasi instans: Anda dapat menggunakan CloudEndureuntuk mereplikasi instance individual dari sistem lokal ke Pos Luar, dari Pos Luar ke Wilayah, dari Wilayah ke Pos Luar, atau dari satu Pos Luar ke pos lainnya. Arsitektur untuk DR AWS Outposts dengan posting CloudEndure blog menjelaskan masing-masing skenario ini dan bagaimana merancang solusi dengannya CloudEndure.



Pemulihan bencana (DR) dari Pos Terdepan ke Wilayah

Menggunakan AWS Outposts rak sebagai CloudEndure tujuan (target replikasi) membutuhkan S3 pada penyimpanan Outposts.

Praktik yang direkomendasikan untuk perlindungan data:

- Gunakan snapshot EBS untuk membuat point-in-time cadangan volume penyimpanan blok ke Amazon S3 di Wilayah atau S3 di Outposts.
- Gunakan S3 pada versi objek Outposts untuk mempertahankan beberapa versi dan riwayat objek Anda.
- Gunakan Replikasi S3 di Outposts untuk secara otomatis mereplikasi data objek Anda ke Outpost lain.
- Untuk kasus penggunaan residensi non-data, gunakan AWS DataSync untuk mencadangkan objek yang disimpan di S3 di Outpost ke Amazon S3 di Wilayah.
- Gunakan CloudEndure untuk mereplikasi instance antara sistem lokal, Outposts logis, dan Region.

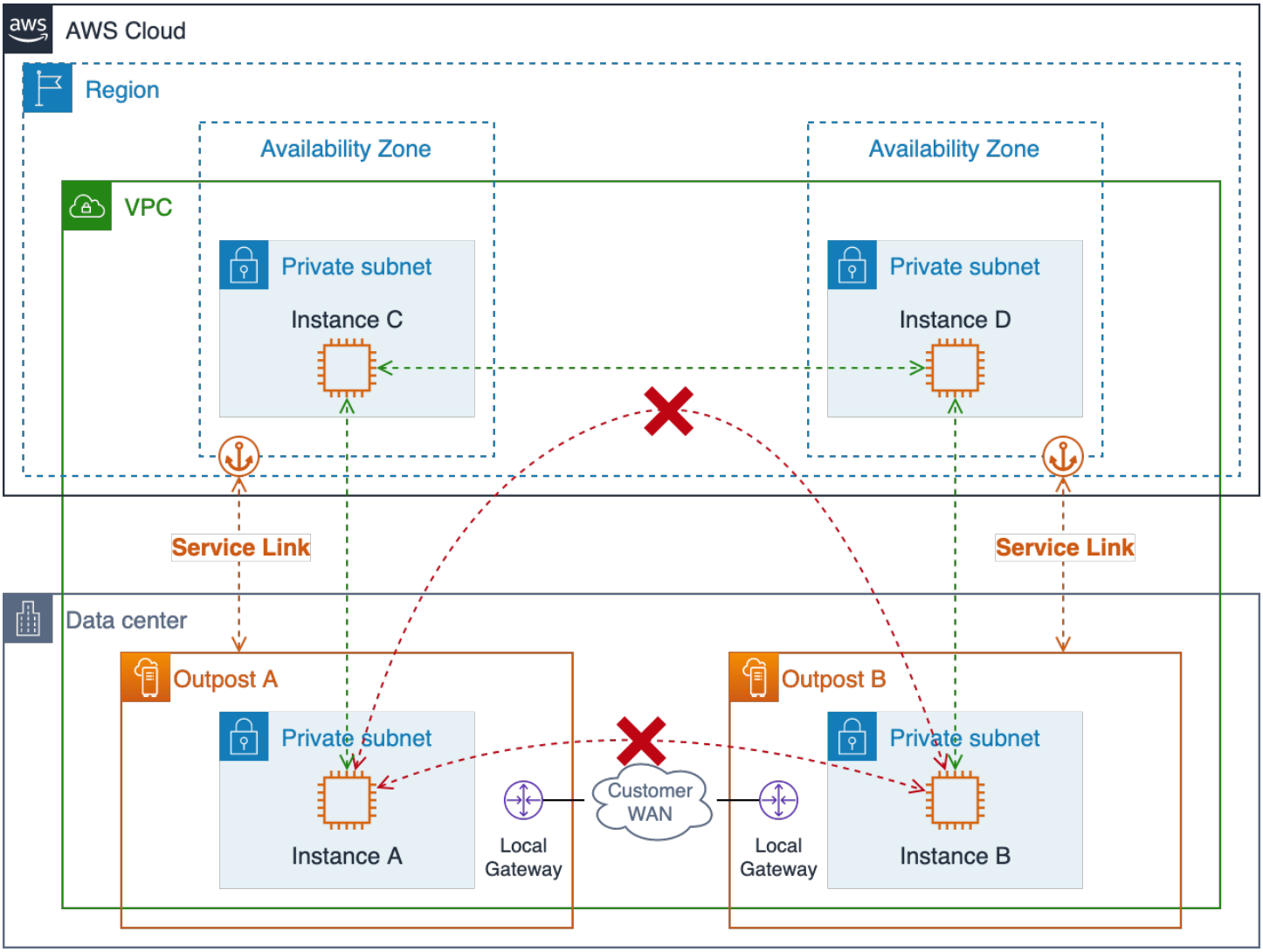
Mode kegagalan yang lebih besar

Untuk merancang arsitektur HA untuk mengurangi mode kegagalan yang lebih besar seperti rak, pusat data, Availability Zone (AZ), atau kegagalan Wilayah, Anda harus menerapkan beberapa

Outposts dengan kapasitas infrastruktur yang memadai di pusat data terpisah dengan daya independen dan konektivitas WAN. Anda menambatkan Outposts ke Availability Zone (AZ) yang berbeda dalam Wilayah AWS satu atau di beberapa Wilayah. Anda juga harus menyediakan site-to-site konektivitas yang tangguh dan memadai antara lokasi untuk mendukung replikasi data sinkron atau asinkron dan pengalihan lalu lintas beban kerja. Bergantung pada arsitektur aplikasi Anda, Anda dapat menggunakan DNS Amazon Route 53 yang tersedia secara global dan layanan Elastic Load Balancing yang tersedia secara regional untuk mengarahkan lalu lintas ke lokasi yang diinginkan dan mengotomatiskan pengalihan lalu lintas ke lokasi yang masih ada jika terjadi kegagalan skala besar.

Ada keterbatasan jaringan yang harus Anda waspadai saat merancang dan menerapkan beban kerja aplikasi di beberapa Outposts. Sumber daya di dua Outposts terpisah tidak dapat berkomunikasi satu sama lain dengan transit lalu lintas melalui Wilayah. Sumber daya pada dua Outpost terpisah yang digunakan dalam VPC yang sama tidak dapat berkomunikasi satu sama lain di seluruh jaringan pelanggan. Sumber daya pada dua Outpost terpisah yang digunakan di VPC yang berbeda dapat berkomunikasi satu sama lain di seluruh jaringan pelanggan.

Dua gambar berikut menggambarkan jalur jaringan yang diblokir dan berhasil.

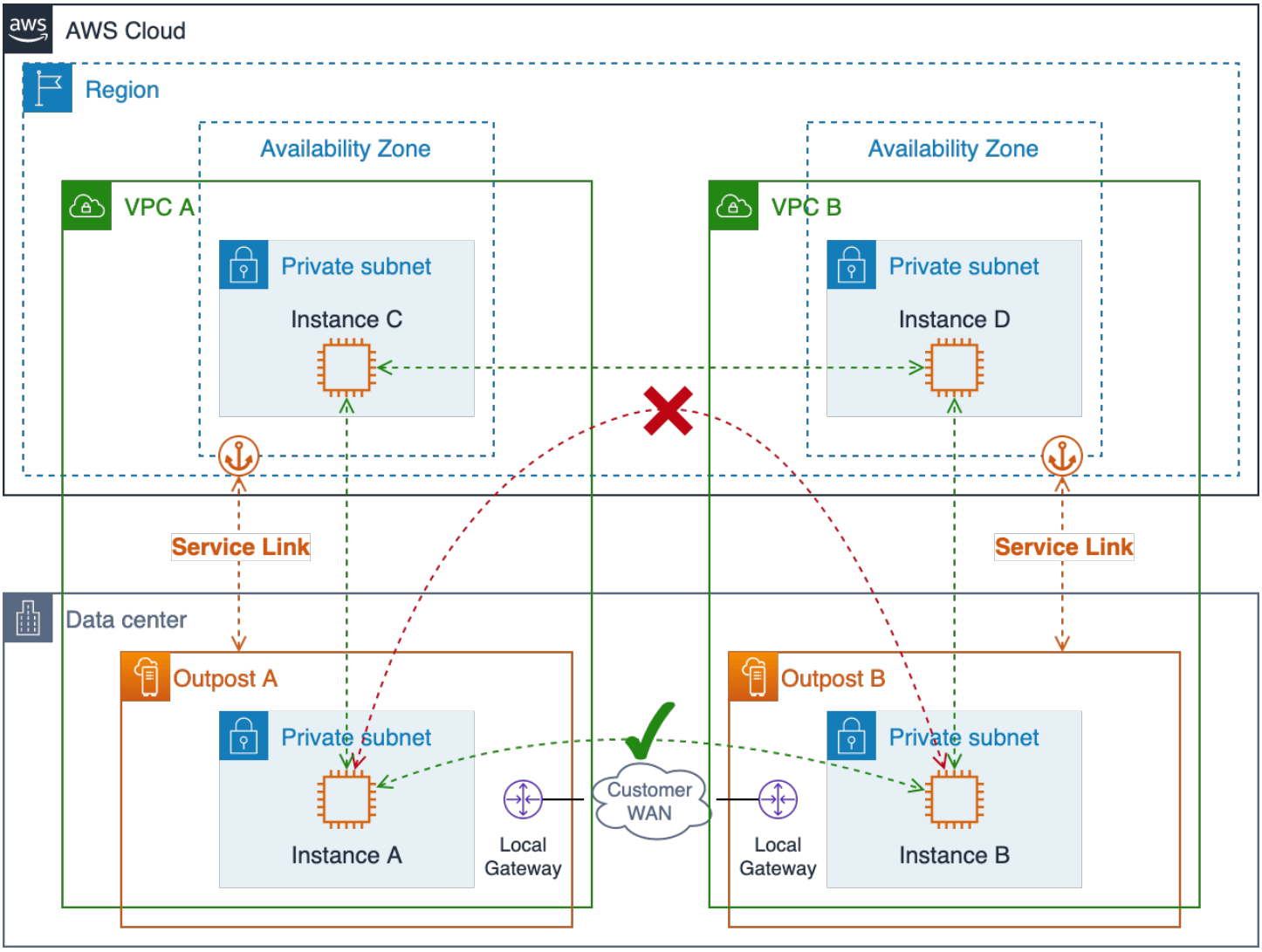


Jalur jaringan multiple-outpost VPC tunggal

Lalu lintas outpost-to-outpost yang transit di Wilayah diblokir karena ini adalah anti-pola. Lalu lintas semacam itu akan menimbulkan biaya keluar di kedua arah dan kemungkinan memiliki latensi yang jauh lebih tinggi daripada sekadar merutekan lalu lintas di WAN Pelanggan.

Sumber daya di beberapa Outpost dalam VPC yang sama tidak dapat berkomunikasi satu sama lain. Lalu lintas antara Outpost di VPC yang sama akan selalu mengikuti rute CIDR VPC lokal melalui Wilayah di mana ia akan diblokir.

Anda harus menggunakan VPC terpisah untuk menyebarkan sumber daya di beberapa Outpost untuk memungkinkan Anda merutekan lalu lintas Outpost-to-Outpost di seluruh jaringan lokal dan WAN lokal Anda.



Jalur jaringan Multiple-outpost Multiple-VPC

Praktik yang disarankan untuk melindungi terhadap mode kegagalan yang lebih besar:

- Terapkan beberapa Outposts yang ditambatkan ke beberapa AZ dan Wilayah.
- Gunakan VPC terpisah untuk setiap Outpost dalam penyebaran Multi-Outpost.

Kesimpulan

Dengan AWS Outposts rack, Anda dapat membuat, mengelola, dan menskalakan aplikasi lokal yang sangat tersedia menggunakan AWS alat dan layanan yang sudah dikenal seperti Amazon EC2, Amazon EBS, Amazon S3 di Outposts, Amazon ECS, Amazon EKS, dan Amazon RDS. Beban kerja dapat berjalan secara lokal, melayani klien, mengakses aplikasi dan sistem di jaringan lokal Anda, dan mengakses rangkaian lengkap layanan di. Wilayah AWS Outposts rack sangat ideal untuk beban kerja yang memerlukan akses latensi rendah ke sistem lokal, pemrosesan data lokal, residensi data, dan migrasi aplikasi dengan saling ketergantungan sistem lokal.

Ketika Anda menyediakan penyebaran Outpost dengan daya, ruang, dan pendinginan yang memadai serta koneksi yang tangguh ke Wilayah AWS, Anda dapat membangun layanan pusat data tunggal yang sangat tersedia. Dan, untuk tingkat ketersediaan dan ketahanan yang lebih tinggi, Anda dapat menyebarkan beberapa Outposts dan mendistribusikan aplikasi Anda melintasi batas logis dan geografis.

Outposts rack menghilangkan beban berat bangunan yang tidak terdiferensiasi dari komputasi, penyimpanan, dan kumpulan jaringan aplikasi lokal dan memungkinkan Anda memperluas jangkauan Infrastruktur AWS Global ke pusat data dan fasilitas lokasi bersama Anda. Sekarang, Anda dapat memfokuskan waktu dan energi Anda untuk memodernisasi aplikasi Anda, merampingkan penerapan aplikasi Anda, dan meningkatkan dampak bisnis dari layanan TI Anda.

Kontributor

Kontributor dokumen ini meliputi:

- Mallory Gershenfeld, S3 di Outposts, Amazon Web Services
- Chris Lunsford, Arsitek Solusi Spesialis Senior, AWS Outposts, Amazon Web Services
- Rohan Mathews, Arsitek Utama,, Amazon Web AWS Outposts Services

Riwayat dokumen

Untuk mengetahui jika ada perubahan pada laporan resmi ini, Anda dapat berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Pembaruan kecil	Menambahkan panduan slotting tambahan dalam perencanaan kapasitas.	Februari 9, 2024
Pembaruan kecil	Diperbarui untuk mencermin kan peluncuran fitur sejak publikasi awal.	Juli 19, 2023
Pembaruan kecil	Praktik yang direkomen dasikan diperbarui untuk lampiran jaringan yang sangat tersedia.	29 Juni 2023
Publikasi awal	Whitepaper pertama kali diterbitkan.	Agustus 12, 2021

> **Note**
>
> Untuk berlangganan pembaruan RSS, Anda harus mengaktifkan plug-in RSS untuk browser yang Anda gunakan.

Pemberitahuan

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

AWS Glosarium

Untuk AWS terminologi terbaru, lihat AWS glosarium di Referensi.Glosarium AWS

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