Praktik Terbaik untuk Menerapkan Amazon 2.0 AppStream

Praktik Terbaik untuk Menerapkan Amazon 2.0 AppStream :

Table of Contents

Abstrak i
Abstrak 1
Pengantar 1
Konsep kunci 2
Desain VPC 3
Pedoman desain 3
Availability Zone 3
Ukuran subnet 4
Perutean subnet 6
Konektivitas Intra-Wilayah 6
Lalu lintas internet keluar 6
On-premise 7
VPC endpoint 7
Titik akhir Amazon S3 VPC 7
Titik akhir VPC antarmuka API Amazon AppStream 2.0 8
Antarmuka streaming Amazon AppStream 2.0 titik akhir VPC 9
Pembuatan dan manajemen gambar 10
Membangun gambar AppStream 2.0 10
Sistem operasi 10
Aplikasi 12
Blok aplikasi 13
Kustomisasi profil pengguna 14
Keamanan 14
Performa 15
AppStream 2.0 pemilihan versi agen 15
Antarmuka Baris Perintah Asisten Gambar (CLI) 16
Mengelola pengalaman streaming pengguna 16
Kustomisasi menggunakan skrip sesi 16
Menggunakan Kebijakan Grup Direktori Aktif 17
Pembaruan gambar 17
Kustomisasi armada 19
Jenis armada 19
Ukuran armada 25
Kapasitas minimum dan penskalaan terjadwal 25
Kapasitas maksimum dan kuota layanan 26

Memilih Tampilan Desktop atau Tampilan Aplikasi 27
Tampilan Desktop 27
Aplikasi Hanya melihat 27
AWS Identity and Access Managementkonfigurasi peran 28
Menggunakan kredensil statis 28
Melindungi Bucket AppStream 2.0 S3 Anda 29
Strategi penskalaan otomatis armada 30
Memahami contoh AppStream 2.0 30
Kebijakan penskalaan 30
Penskalaan langkah 30
Pelacakan target 30
Penskalaan berbasis terjadwal 31
Kebijakan penskalaan dalam produksi 31
Praktik terbaik untuk desain kebijakan penskalaan 33
Menggabungkan kebijakan penskalaan 33
Hindari penskalaan churn 33
Memahami tingkat penyediaan maksimum 34
Memanfaatkan beberapa Availability Zone 34
Pantau metrik Kesalahan Kapasitas Tidak Cukup 35
Metode koneksi 36
Fitur ringkasan dan dukungan perangkat 36
Akses browser web 37
AppStream 2.0 klien untuk Windows 37
AppStream 2.0 mode koneksi klien 38
Penyebaran dan manajemen klien 38
Domain kustom 40
Autentikasi 41
Menentukan metode yang dioptimalkan 41
Mengkonfigurasi penyedia identitas Anda 43
SAML 2.0 43
Kolam pengguna 44
URL streaming 44
Hak aplikasi 45
Integrasi dengan Microsoft Active Directory 46
Opsi layanan 46
Skenario penyebaran 46
Skenario 1: Layanan Domain Direktori Aktif (ADDS) diterapkan di lokasi 47

Skenario 2: Perluas Layanan Domain Aktif (ADDS) ke AWS VPC pelanggan 48
Skenario 3: Direktori Aktif Microsoft yang AWS Dikelola 49
Topologi Situs Active Directory Service 50
Unit Organisasi Direktori Aktif 51
Pembersihan objek komputer Active Directory 52
Keamanan 53
Mengamankan data persisten 53
Status pengguna dan data 53
Keamanan titik akhir dan antivirus 55
Menghapus pengidentifikasi unik 55
Optimalisasi kinerja 55
Pengecualian pemindaian 56
Folder 57
Kebersihan konsol keamanan titik akhir 58
Pengecualian jaringan 58
Mengamankan sesi AppStream 59
Membatasi kontrol aplikasi dan sistem operasi 59
Firewall dan routing 59
Pencegahan kehilangan data 60
Klien ke AppStream 2.0 Kontrol Transfer Data Instance 60
Mengontrol lalu lintas keluar dari Instans 2.0 AppStream 61
Menggunakan AWS layanan 61
AWS Identity and Access Management 61
VPCtitik akhir 62
Pemulihan bencana 64
Perutean identitas 64
Metode 1: Mengubah status relai aplikasi Anda 64
Metode 2: Mengkonfigurasi dua aplikasi AppStream 2.0 dalam IDP Anda 65
Ketekunan penyimpanan 66
Pemantauan 67
Menggunakan dasbor 67
Mengantisipasi pertumbuhan 67
Memantau penggunaan pengguna 68
Aplikasi yang bertahan dan log peristiwa Windows 68
Jaringan audit dan aktivitas administrasi 68
Optimalisasi biaya 69
Merancang penerapan AppStream 2.0 yang hemat biaya 69

Mengoptimalkan biaya dengan pilihan jenis instans 70
Mengoptimalkan biaya dengan pilihan tipe armada 70
Kebijakan penskalaan 72
Biaya pengguna 72
Penggunaan Image Builder 73
Kesimpulan 74
Kontributor 75
Bacaan lebih lanjut 76
Revisi dokumen 77
Pemberitahuan 78
..... lxxix

Praktik Terbaik untuk Menerapkan Amazon 2.0 AppStream

Tanggal publikasi: 19 Januari 2022 (Revisi dokumen)

Abstrak

Whitepaper ini menguraikan serangkaian praktik terbaik untuk penyebaran Amazon 2.0. AppStream Paper ini mencakup desain Amazon Virtual Private Cloud (VPC), pembuatan dan manajemen gambar, kustomisasi armada, dan strategi penskalaan otomatis armada. Ini termasuk metode koneksi pengguna, otentikasi, dan integrasi dengan Microsoft Active Directory. Paper ini juga mencakup rekomendasi untuk merancang keamanan AppStream 2.0, pemantauan, dan optimalisasi biaya.

Whitepaper ini ditulis untuk memungkinkan akses cepat ke informasi yang relevan. Ini ditujukan untuk insinyur jaringan, spesialis pengiriman aplikasi, insinyur direktori, atau insinyur keamanan.

Pengantar

Amazon AppStream 2.0 adalah layanan streaming aplikasi yang dikelola sepenuhnya yang menyediakan pengguna dengan akses instan ke aplikasi desktop mereka dari mana saja. AppStream 2.0 mengelola AWS sumber daya yang diperlukan untuk meng-host dan menjalankan aplikasi Anda. Ini menskalakan secara otomatis, dan menyediakan akses ke pengguna Anda sesuai permintaan. AppStream 2.0 memberi pengguna akhir akses ke aplikasi yang mereka butuhkan pada perangkat pilihan mereka, dengan pengalaman pengguna yang responsif, tidak dapat dibedakan dari aplikasi yang diinstal secara asli.

Bagian berikut memberikan detail tentang Amazon AppStream 2.0, menjelaskan cara kerja layanan, menjelaskan apa yang Anda perlukan untuk meluncurkan layanan, dan memberi tahu Anda opsi dan fitur apa yang tersedia untuk Anda gunakan. Saat menerapkan AppStream 2.0 untuk pengguna akhir, penting untuk menerapkan praktik terbaik untuk memberikan pengalaman pengguna yang luar biasa. Selain itu, perusahaan dari semua ukuran mendapat manfaat dari optimalisasi biaya yang mengurangi biaya operasional bulanan.

Konsep kunci

Untuk mendapatkan hasil maksimal dari AppStream 2.0, kenali konsep-konsep berikut:

- Gambar - Gambar adalah template instance yang telah dikonfigurasi sebelumnya. Gambar berisi aplikasi yang dapat Anda streaming ke pengguna Anda, dan pengaturan Windows dan aplikasi default untuk memungkinkan pengguna Anda memulai aplikasi mereka dengan cepat. AWSmenyediakan gambar dasar yang dapat Anda gunakan untuk membuat gambar yang menyertakan aplikasi Anda sendiri. Setelah Anda membuat gambar, Anda tidak dapat mengubahnya. Untuk menambahkan aplikasi lain, memperbarui aplikasi yang ada, atau mengubah pengaturan gambar, Anda harus membuat gambar baru. Anda dapat menyalin gambar Anda ke orang lain Wilayah AWSatau membagikannya dengan Akun AWS s lain di Wilayah yang sama.
- Pembuat gambar - Pembuat gambar adalah mesin virtual yang Anda gunakan untuk membuat gambar. Anda dapat meluncurkan dan terhubung ke pembuat gambar menggunakan konsol AppStream 2.0. Setelah Anda terhubung ke pembuat gambar, Anda dapat menginstal, menambahkan, dan menguji aplikasi Anda, dan kemudian menggunakan pembuat gambar untuk membuat gambar. Anda dapat meluncurkan pembuat gambar baru dengan menggunakan gambar pribadi yang Anda miliki.
- Armada — Armada terdiri dari instance armada (juga dikenal sebagai instance streaming) yang menjalankan gambar yang Anda tentukan. Anda dapat mengatur jumlah instans streaming yang diinginkan untuk armada Anda, dan mengonfigurasi kebijakan untuk menskalakan armada Anda secara otomatis berdasarkan permintaan. Perhatikan bahwa setiap pengguna memerlukan satu contoh.
- Stack — Stack terdiri dari armada terkait, kebijakan akses pengguna, dan konfigurasi penyimpanan. Anda menyiapkan tumpukan untuk memulai streaming aplikasi ke pengguna.
- Instans streaming — Instans streaming (juga dikenal sebagai instance armada) adalah instans Amazon Elastic Compute Cloud (Amazon EC2) yang tersedia untuk satu pengguna untuk streaming aplikasi. Setelah sesi pengguna selesai, instance dihentikan oleh Amazon EC2.

Desain VPC

Pedoman desain

Terapkan AppStream 2.0 ke dalam VPC khusus. Saat merancang VPC AppStream 2.0, ukuran untuk pertumbuhan yang diperkirakan. Cadangan kapasitas alamat IP untuk kasus penggunaan baru, dan Availability Zone (AZ) tambahan yang dapat ditambahkan di lain waktu. Titik desain dasar AppStream 2.0 adalah bahwa hanya satu pengguna yang dapat menggunakan instance AppStream 2.0. Saat mengalokasikan ruang IP, pikirkan satu pengguna sebagai satu alamat IP per AppStream 2.0 instance. Dengan AppStream 2.0, pengguna dapat menggunakan beberapa instance AppStream 2.0. Oleh karena itu, perencanaan ruang IP juga harus memperhitungkan kasus penggunaan yang memerlukan instance AppStream 2.0 tambahan.

Meskipun ukuran maksimum VPC Classless Inter-Domain Routing (CIDR) adalah /16, merekomendasikan untuk tidak mengalokasikan alamat IP pribadi secara berlebihan. AWS Dimungkinkan untuk memperluas ukuran VPC melalui CIDR tambahan, tetapi ada batasan untuk ini, oleh karena itu, alokasikan apa yang dibutuhkan sejak awal.

Jika penyebaran AppStream 2.0 digabungkan ke domain Active Directory, opsi DHCP yang ditetapkan untuk VPC harus memiliki domain DNS yang dikonfigurasi. Server nama domain harus menentukan alamat IP DNS yang otoritatif untuk domain Active Directory, atau DNS harus meneruskan permintaan DNS ke instans DNS otoritatif untuk domain Active Directory. Selain itu, VPC harus memiliki `enableDnsHostnames` dan `EnableDnsSupport` dikonfigurasi.

Availability Zone

Availability Zone (AZ) adalah satu atau lebih pusat data diskrit dengan daya redundan, jaringan, dan konektivitas dalam file. Wilayah AWS Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Amazon AppStream 2.0 hanya membutuhkan satu subnet untuk armada untuk diluncurkan. Praktik terbaik adalah mengonfigurasi minimal dua Availability Zone, satu subnet per Availability Zone unik. Untuk mengoptimalkan penskalaan otomatis armada, gunakan lebih dari dua Availability Zone. Penskalaan secara horizontal memiliki manfaat tambahan menambahkan ruang IP dalam subnet untuk pertumbuhan, yang tercakup dalam bagian ukuran Subnet berikut dari dokumen ini. AWS

Management Console hanya menyediakan dua subnet yang akan ditentukan selama pembuatan armada. Gunakan AWS Command Line Interface(AWSCLI) atau AWS CloudFormation untuk memungkinkan lebih dari dua ID subnet.

Ukuran subnet

Dedikasikan subnet ke AppStream 2.0 armada untuk memungkinkan fleksibilitas dalam kebijakan perutean, dan Daftar Kontrol Akses Jaringan. Tumpukan kemungkinan akan memiliki persyaratan sumber daya yang terpisah. Misalnya, AppStream 2.0 Stacks dapat memiliki persyaratan isolasi yang memberi jalan untuk memisahkan kumpulan aturan. Ketika beberapa armada Amazon AppStream 2.0 menggunakan subnet yang sama, pastikan jumlah Kapasitas Maksimum semua armada tidak melebihi jumlah total alamat IP yang tersedia.

Jika kapasitas maksimum untuk semua armada di subnet yang sama dapat, atau telah, melebihi jumlah total alamat IP yang tersedia, migrasi armada ke subnet khusus. Ini mencegah peristiwa penskalaan otomatis menghabiskan ruang IP yang dialokasikan. Jika total kapasitas armada melebihi ruang IP yang dialokasikan dari subnet yang ditetapkan, gunakan API, atau “update fleet” AWS CLI untuk menetapkan lebih banyak subnet. Untuk informasi lebih lanjut, lihat kuota Amazon VPC, dan cara meningkatkannya.

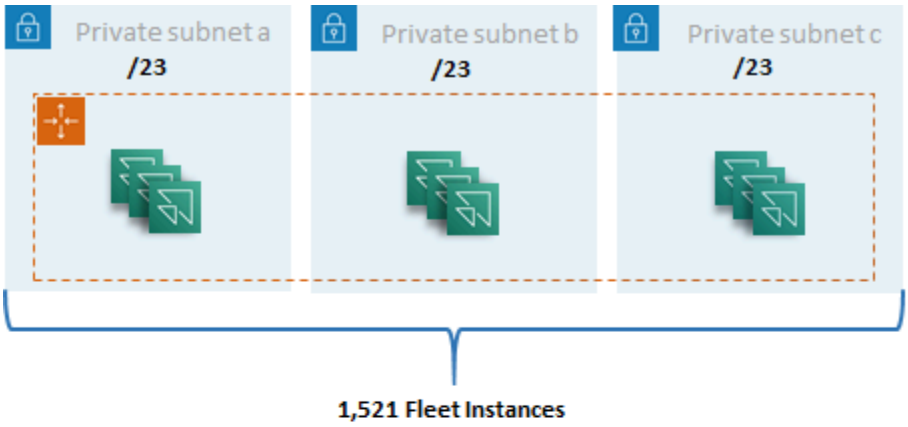
Ini adalah praktik terbaik untuk mengukur jumlah subnet, mengukur subnet yang sesuai sambil menyimpan kapasitas untuk tumbuh di VPC Anda. Selain itu, pastikan bahwa maksimum armada AppStream 2.0 tidak melebihi total ruang IP yang dialokasikan oleh subnet. Untuk setiap subnetAWS, lima alamat IP dicadangkan saat menghitung jumlah total ruang IP. Menggunakan lebih dari dua subnet dan penskalaan secara horizontal menawarkan beberapa manfaat, seperti:

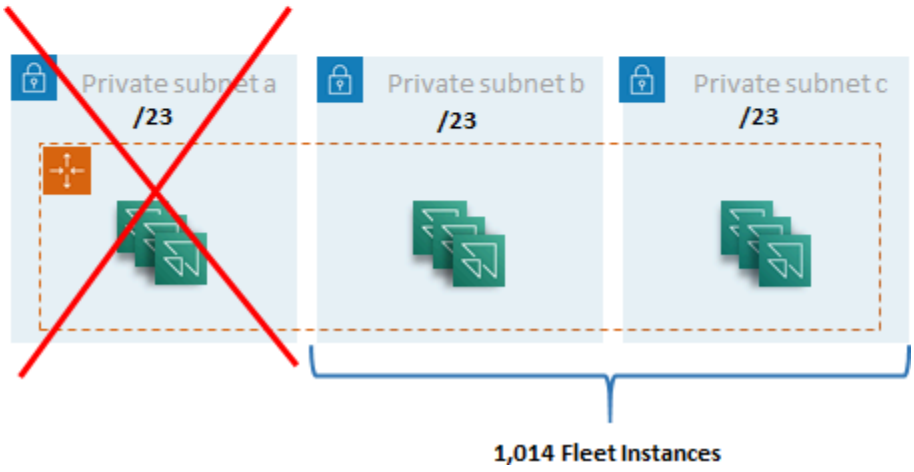
- Ketahanan yang lebih besar dari kegagalan Availability Zone
- Throughput yang lebih besar saat instance armada penskalaan otomatis
- Penggunaan alamat IP pribadi yang lebih efisien, menghindari pembakaran IP

Saat mengukur subnet untuk Amazon AppStream 2.0, pertimbangkan jumlah total subnet, dan konkurensi puncak yang diharapkan selama pemanfaatan puncak. Ini dapat dipantau menggunakan (`InUseCapacity`) ditambah kapasitas cadangan (`AvailableCapacity`) untuk armada. Di Amazon AppStream 2.0, jumlah instance armada yang dikonsumsi dan available-to-be-consumed AppStream 2.0 diberi label`ActualCapacity`. Untuk mengukur total ruang IP dengan benar, perkirakan yang diperlukan`ActualCapacity`, dan bagi dengan jumlah subnet, dikurangi satu subnet untuk ketahanan, yang ditugaskan ke armada.

Misalnya, jika jumlah maksimum instans armada yang diantisipasi pada puncaknya adalah 1000, dan persyaratan bisnis harus tangguh dalam satu kegagalan Availability Zone, 3 x/23 subnet memenuhi persyaratan teknis dan bisnis.

- /23 = 512 Host - 5 Cadangan = 507 instance armada per subnet
- 3 subnet - 1 subnet = 2 subnet
- 2 subnet x 507 instance armada per subnet = 1.014 instance armada di puncak





Contoh ukuran subnet

Sementara 2 x /22 subnet juga akan memenuhi ketahanan, pertimbangkan hal berikut:

- Alih-alih 1.536 alamat IP yang dicadangkan, menggunakan dua AZ menghasilkan 2.048 alamat IP yang dicadangkan, membuang-buang alamat IP yang bisa masuk ke fungsi lain.
- Jika satu AZ menjadi tidak dapat diakses, kemampuan untuk skala instance armada dibatasi oleh throughput AZ. Hal ini dapat memperpanjang durasi`PendingCapacity`.

Perutean subnet

Ini adalah praktik terbaik untuk membuat subnet pribadi untuk AppStream 2.0 instance, perutean ke internet publik melalui VPC terpusat untuk lalu lintas keluar. Lalu lintas masuk untuk streaming sesi AppStream 2.0 ditangani melalui layanan Amazon AppStream 2.0 melalui Gateway Streaming: Anda tidak perlu mengonfigurasi subnet publik untuk ini.

Konektivitas Intra-Wilayah

Untuk instance armada AppStream 2.0 yang digabungkan ke Domain Direktori Aktif, konfigurasikan Pengontrol Domain Direktori Aktif di VPC Layanan Bersama di masing-masing. Wilayah AWS Sumber untuk Active Directory dapat berupa Pengontrol Domain berbasis Amazon EC2 atau AWSMicrosoft Managed AD. Perutean antara layanan bersama dan AppStream 2.0 VPC dapat dilakukan melalui koneksi peering VPC atau gateway transit. Meskipun gateway transit memecahkan kompleksitas perutean dalam skala besar, ada sejumlah alasan mengapa peering VPC lebih disukai di sebagian besar pengaturan:

- VPC peering adalah koneksi langsung antara dua VPC (tidak ada hop ekstra).
- Tidak ada biaya per jam, hanya tarif transfer data standar antara Availability Zones.
- Tidak ada batasan bandwidth.
- Support untuk mengakses Grup Keamanan antar VPC.

Hal ini terutama berlaku jika instance AppStream 2.0 terhubung ke infrastruktur aplikasi dan/atau server file dengan kumpulan data besar dalam VPC layanan bersama. Dengan mengoptimalkan jalur ke sumber daya yang umum diakses ini, koneksi peering VPC lebih disukai, bahkan dalam desain di mana semua VPC dan perutean internet lainnya dilakukan melalui gateway transit.

Lalu lintas internet keluar

Sementara routing langsung ke layanan bersama sebagian besar dioptimalkan melalui koneksi peering, lalu lintas keluar untuk AppStream 2.0 dapat dirancang dengan membuat titik keluar internet tunggal dari beberapa VPC menggunakan Transit Gateway. AWS Dalam desain multi-VPC, itu adalah praktik standar untuk memiliki VPC khusus yang mengontrol semua lalu lintas internet keluar. Dengan konfigurasi ini, Transit Gateways memiliki fleksibilitas yang lebih besar, dan kontrol routing atas tabel routing standar yang melekat pada subnet. Desain ini juga mendukung perutean transitif tanpa kerumitan tambahan, dan menghilangkan kebutuhan akan gateway terjemahan alamat jaringan (NAT) redundan, atau instance NAT di setiap VPC.

Setelah semua lalu lintas internet keluar dipusatkan menjadi VPC tunggal, gateway NAT atau instance NAT adalah pilihan desain yang umum. Untuk menentukan mana yang terbaik untuk organisasi Anda, lihat panduan administrasi untuk membandingkan gateway NAT dan instance NAT. AWS Network Firewall dapat memperluas perlindungan di luar kelompok keamanan dan tingkat kontrol akses jaringan dengan melindungi pada tingkat rute dan menawarkan aturan stateless dan stateful dari lapisan 3 hingga 7 dalam model OSI. Untuk informasi selengkapnya, lihat model Deployment untuk AWS Network Firewall. Jika organisasi Anda telah memilih produk pihak ketiga yang melakukan fitur-fitur canggih seperti pemfilteran URL, terapkan layanan ke VPC internet keluar Anda. Ini dapat menggantikan gateway NAT atau instance NAT. Ikuti panduan yang diberikan oleh vendor pihak ketiga.

On-premise

Ketika konektivitas ke sumber daya lokal diperlukan, terutama untuk instance AppStream 2.0 yang digabungkan ke Active Directory, buat koneksi yang sangat tangguh. AWS Direct Connect

VPC endpoint

Titik akhir Amazon S3 VPC

Banyak penerapan Amazon AppStream 2.0 memerlukan persistensi status pengguna melalui folder rumah dan pengaturan aplikasi. Aktifkan komunikasi pribadi ke lokasi Amazon Simple Storage Service (Amazon S3) ini, karena ini menghindari penggunaan internet publik. Anda dapat mencapai ini melalui gateway titik akhir VPC. Gateway titik akhir VPC lebih disukai daripada untuk Amazon AWS PrivateLinkS3 karena:

- Ini dioptimalkan biaya untuk persyaratan akses jaringan AppStream 2.0
- Akses bucket Amazon S3 tidak diperlukan dari sumber daya lokal
- Dokumen kebijakan khusus dapat digunakan untuk membatasi akses hanya dari instance AppStream 2.0

Setelah Anda membuat gateway titik akhir VPC, ini adalah praktik terbaik untuk mengamankan koneksi yang diprivatisasi dengan membuat kebijakan khusus. Kebijakan kustom dimulai dengan Amazon Resource Name (ARN) dari peran Identity and Access Management layanan AppStream 2.0. Tentukan secara eksplisit tindakan S3 yang diperlukan untuk persistensi status pengguna.

> **Note**
>
> Contoh berikut di `Resources` bagian ini menentukan jalur folder rumah negara pertama dan jalur pengaturan aplikasi kedua.

Example

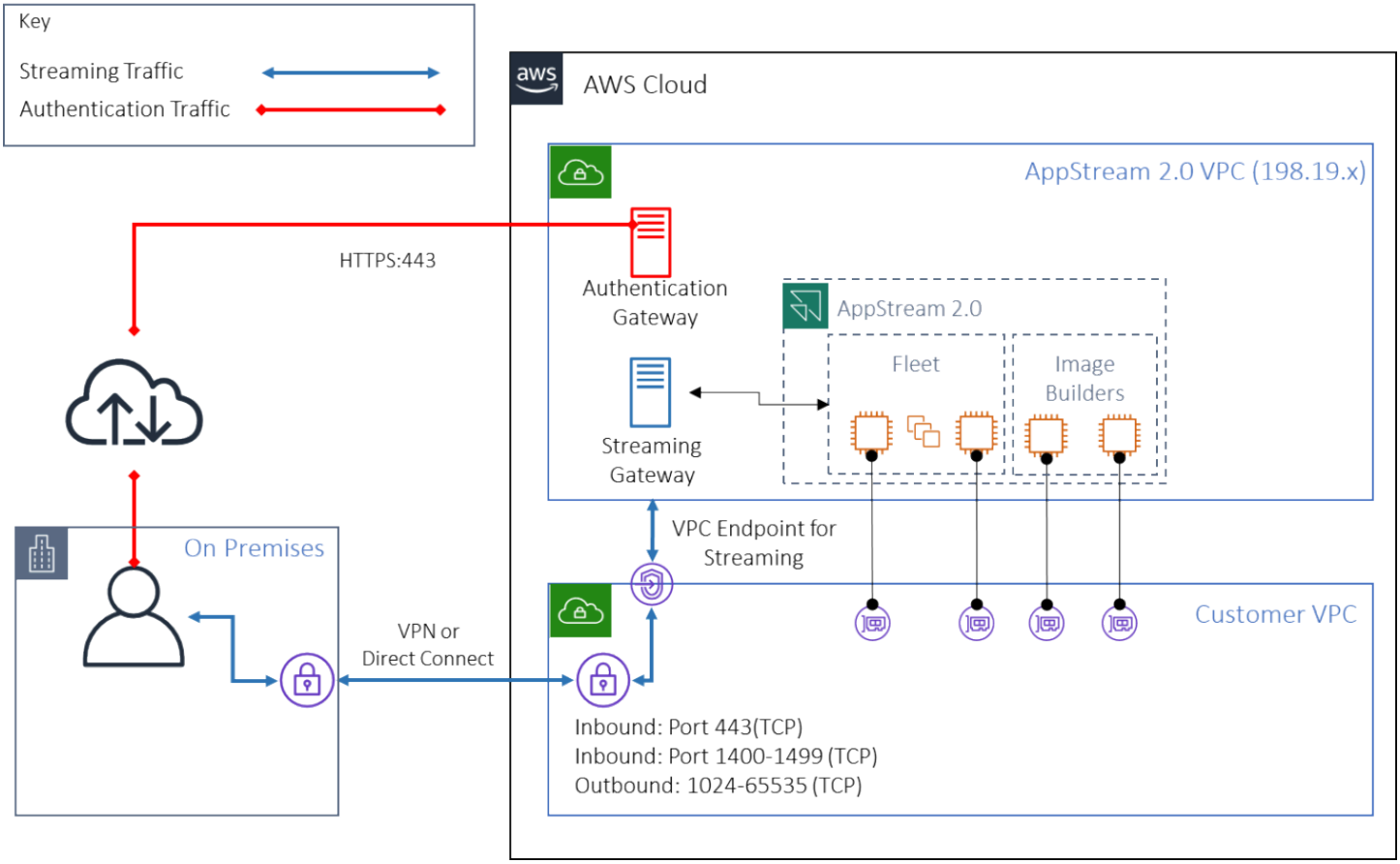
```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow-AppStream-to-access-home-folder-and-
application-settings",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:sts::account-id-without-hyphens:assumed-
role/AmazonAppStreamServiceAccess/AppStream2.0"
      },
      "Action": [
        "s3:ListBucket",
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject",
        "s3:GetObjectVersion",
        "s3:DeleteObjectVersion"
      ],
      "Resource": [
        "arn:aws:s3:::appstream2-36fb080bb8-*",
        "arn:aws:s3:::appstream-app-settings-*"
      ]
    }
  ]
}
```

Titik akhir VPC antarmuka API Amazon AppStream 2.0

Dalam skenario desain di mana perintah API dan CLI ke Amazon AppStream 2.0 berasal dari VPC Anda, privatisasi panggilan terprogram ini melalui titik akhir VPC antarmuka.

Antarmuka streaming Amazon AppStream 2.0 titik akhir VPC

Meskipun dimungkinkan untuk merutekan lalu lintas streaming Amazon AppStream 2.0 melalui titik akhir VPC antarmuka, gunakan konfigurasi ini dengan hati-hati. Perilaku streaming default melalui internet publik adalah metode pengiriman yang paling efisien dan berkinerja untuk lalu lintas streaming Amazon AppStream 2.0.



Antarmuka streaming Amazon AppStream 2.0 titik akhir VPC

Seperti yang ditunjukkan pada gambar sebelumnya, internet publik adalah jalur paling efisien ke Gateway Streaming Amazon AppStream 2.0. Routing melalui VPC dan jaringan yang dikelola pelanggan menambah kompleksitas dan latensi. Ini juga menambahkan biaya transfer dataAWS Direct Connect.

> **Note**
>
> Hanya streaming yang didukung oleh titik akhir VPC, dan otentikasi masih harus dilakukan melalui internet publik. Akses prasyarat seperti SAMP Single Sign-On (SSO) Identity Provider (IDP) tetap menjadi persyaratan yang hanya dapat diakses melalui internet publik.

Pembuatan dan manajemen gambar

Saat meluncurkan armada atau pembuat gambar di AppStream 2.0, Anda harus memilih salah satu gambar dasar AppStream 2.0. Administrator kemudian dapat membangun gambar dasar untuk menambahkan aplikasi dan pengaturan konfigurasi mereka sendiri.

Ada pertimbangan utama saat membuat gambar untuk memastikan aplikasi bekerja dengan benar dan aman. Selain itu, ada pertimbangan desain untuk bagaimana gambar itu akan dipertahankan.

Membangun gambar AppStream 2.0

Saat membangun gambar baru, penting untuk mempertimbangkan hal berikut:

- Sistem operasi
- Aplikasi
- Profil pengguna
- Keamanan
- Performa
- Versi agen
- Asisten Gambar CLI

Membangun gambar AppStream 2.0

Pada November 2021, AppStream 2.0 meluncurkan dukungan untuk Amazon Linux 2. Dengan pengumuman ini, AppStream 2.0 sekarang mendukung empat jenis platform:

- Windows Server 2012 R2
- Windows Server 2016
- Windows Server 2019
- Amazon Linux 2

Ada kemungkinan bahwa Anda mungkin harus memilih platform tertentu berdasarkan apa yang diperlukan oleh aplikasi Anda (misalnya, jika aplikasi Anda memerlukan Windows, Amazon Linux 2 tidak akan menjadi pilihan). Di luar persyaratan aplikasi, rujuk matriks perbandingan berikut untuk

membantu Anda memilih jenis platform mana yang paling sesuai dengan kasus penggunaan dan lingkungan Anda:

Tabel 1 — Jenis platform, kapan menggunakannya, dan harga

Jenis platform	Kapan harus menggunakan	Harga armada*
Windows Server (2012 R2, 2016 atau 2019)	Aplikasi Anda hanya dapat dijalankan di Windows (dan tidak mendukung Amazon Linux 2). Anda ingin domain bergabung dengan instans streaming Anda. Anda ingin menggunakan Kebijakan Grup yang ada pada instans streaming AppStream 2.0 Anda (Linux tidak mematuhi Kebijakan Grup, tetapi Anda dapat menggunakan Skrip Sesi untuk mengotoma tiskan konfigurasi saat sesi dimulai). Anda akan menggunakan Tampilan Desktop dan pengguna Anda lebih memilih pengalama n desktop Windows. Anda lebih suka menggunakan aplikasi Image Assistant, yang menyediakan step-by-step wizard, untuk membuat katalog dan gambar aplikasi Anda. Saat ini, Anda harus membuat image Amazon Linux 2 Anda menggunak an perintah terminal (lihat tutorial ini untuk info lebih lanjut). Anda ingin menggunak	Biaya RDS SAL (Microsof t Remote Desktop Services Subscriber Access License) sebesar $4,19 per bulan untuk setiap pengguna unik** ditambah yang berikut ini: 1. $0,10 per jam untuk armada Always-On, On-Demand 2. $0,15 per jam untuk armada elastis

Jenis platform	Kapan harus menggunakan	Harga armada*
	an Persistensi Pengaturan Aplikasi. Mengaktifkan persistensi pengaturan aplikasi saat ini tidak didukung untuk tumpukan berbasis Linux.	
Amazon Linux 2	Anda ingin memanfaatkan instans streaming berbiaya lebih rendah dan menghindari biaya lisensi RDS SAL. Aplikasi Anda kompatibel dengan Amazon Linux 2	Instans Linux lebih murah dibandingkan dengan instance Window. Dengan Linux, Anda tidak membayar biaya RDS SAL dan biaya per jam berikut: 1. $0,084 per jam untuk armada Always-On, On-Demand 2. $0,112 per jam untuk armada elastis

* Berdasarkan stream.standard.medium di Wilayah Virginia N

** Pelanggan yang memenuhi syarat dapat membawa lisensi mereka sendiri untuk menghilangkan biaya AWS RDS SAL. Lihat halaman harga AppStream 2.0 untuk lebih jelasnya. Pelanggan pendidikan mungkin juga memenuhi syarat untuk mendapatkan penawaran khusus. Sekolah, universitas, dan lembaga publik tertentu mungkin memenuhi syarat untuk mengurangi biaya pengguna Microsoft RDS SAL.

Aplikasi

Sebelum menginstal aplikasi, penting untuk meninjau persyaratan aplikasi seperti dependensi aplikasi dan persyaratan perangkat keras. Setelah berhasil menginstal aplikasi pada instance pembuat gambar, pastikan untuk mengganti pengguna dan menguji aplikasi di bawah konteks pengguna pengujian.

Saat merencanakan penerapan aplikasi Anda, perhatikan titik akhir dan kuota layanan. Selain itu, bersihkan file installer dan helper untuk mengoptimalkan total ruang C Drive sebelum membuat

gambar. Sebagai pengingat, instance AppStream 2.0 memiliki satu volume ukuran tetap 200 GB. Mengoptimalkan ruang disk setelah instalasi adalah praktik terbaik untuk memastikan bahwa volume ukuran tetap tidak pernah terlampaui.

Jika Anda ingin memodifikasi katalog aplikasi yang dapat diakses pengguna Anda secara real-time, kerangka kerja aplikasi dinamis menyediakan operasi API. Aplikasi yang dikelola oleh penyedia aplikasi dinamis dapat berada di dalam gambar, atau mereka dapat di luar instance, seperti dari berbagi file Windows atau teknologi virtualisasi aplikasi. Fitur ini memerlukan armada AppStream 2.0 yang bergabung ke domain Microsoft Active Directory. Untuk informasi selengkapnya, lihat Menggunakan Direktori Aktif dengan AppStream 2.0.

Blok aplikasi

Blok aplikasi mewakili skrip pengaturan dan file aplikasi yang diperlukan untuk meluncurkan aplikasi yang akan digunakan pengguna Anda. Hard disk virtual (VHD) dapat berupa objek apa saja dari Amazon S3. Disarankan bahwa objek ini kurang dari 1,5GB, karena harus diunduh sepenuhnya sebelum pengguna dapat mengakses aplikasi.

Mengoptimalkan blok aplikasi

Untuk armada berbasis Windows, disarankan Anda membuat file VHDX untuk memuat aplikasi Anda. Untuk armada berbasis Linux, disarankan Anda membuat gambar (IMG). Disk virtual ini harus dibuat sekecil mungkin, untuk meng-host file aplikasi. Disk virtual dapat di-zip untuk lebih mengurangi ukurannya. Dalam skrip pengaturan, Anda harus membuka zip disk sebelum memasang. Contoh skrip PowerShell pengaturan Windows memiliki fungsionalitas unzip yang disertakan. Ada trade off antara memperluas arsip (zip) dan kecepatan unduh. Beberapa pengujian mungkin diperlukan untuk menemukan saldo yang menawarkan waktu peluncuran aplikasi tercepat.

Memperbarui aplikasi

Aplikasi dapat memiliki perubahan kecil dan besar. Untuk pembaruan kecil, gunakan aktifkan pembuatan versi di bucket Amazon S3 yang meng-host file pemblokiran aplikasi Anda. Pengaturan ini memungkinkan administrator untuk memutar kembali ke versi sebelumnya dari aplikasi tertentu dengan mengubah versi objek VHD aplikasi yang dimaksud tanpa mengubah konfigurasi blok aplikasi. Dengan pembaruan besar, buat blok Aplikasi baru untuk VHD yang diperbarui. Ini akan memungkinkan administrator untuk memisahkan perubahan aplikasi utama di tingkat blok aplikasi yang bertentangan dengan tingkat versi, yang menyediakan pendekatan yang lebih terorganisir untuk manajemen aplikasi administratif.

Kustomisasi profil pengguna

Amazon AppStream 2.0 dengan desain aplikasi non-persisten dan solusi desktop. Ketika sesi pengguna dihentikan, perubahan sistem dan pengguna juga dihentikan. Aktifkan ketekunan pengaturan aplikasi hanya jika diperlukan. Ini dapat menambahkan overhead ke proses logon, dan pertimbangan biaya untuk penyimpanan S3 yang diperlukan.

Dalam situasi di mana ketekunan pengaturan aplikasi diperlukan, AWS rekomendasikan untuk mengamankan koneksi tersebut melalui kebijakan khusus dan titik akhir gateway VPC S3. Evaluasi ukuran pengaturan aplikasi secara keseluruhan, dan minimalkan pengaturan yang disimpan dalam persistensi pengaturan aplikasi untuk mengoptimalkan biaya dan kinerja.

Kustomisasi profil pengguna dapat dikonfigurasi pada instance Image Builder AppStream 2.0. Ini termasuk menambahkan dan memodifikasi kunci registri, menambahkan file, dan konfigurasi khusus pengguna lainnya. Dari AppStream 2.0 Image Assistant, ada opsi untuk membuat profil pengguna. Ini menyalin profil pengguna template ke profil pengguna default. Setelah gambar disebarkan ke armada, pengguna akhir yang melakukan streaming sesi dari armada akan membuat profil pengguna mereka dari profil pengguna default. Penting untuk mempertimbangkan meminimalkan ukuran profil pengguna, terutama ketika Persistensi Pengaturan Aplikasi diaktifkan. Secara default, ukuran VHDx maksimum untuk profil pengguna adalah 1 GB. Setiap kali sesi streaming dimulai, file VHDx profil pengguna diunduh dari bucket S3. Ini meningkatkan waktu persiapan sesi streaming dan menimbulkan risiko melebihi batas, yang akan menyebabkan kegagalan pemasangan profil pengguna menggunakan file VHDx.

Untuk kasus penggunaan yang memerlukan profil pengguna lebih besar dari 1 GB, AWS merekomendasikan penggunaan metode alternatif untuk menyimpan profil. Misalnya, menggunakan profil Roaming, atau Kontainer Profil FSLogix pada penyimpanan bersama seperti Amazon FSx for Windows File Server. Untuk informasi selengkapnya, lihat Gunakan Amazon FSx for Windows File Server dan FSLogix untuk Mengoptimalkan Persistensi Pengaturan Aplikasi di Amazon 2.0. AppStream

Keamanan

Ada berbagai pengukuran keamanan yang perlu dipertimbangkan oleh pengembang. AppStream administrator bertanggung jawab untuk menginstal dan memelihara pembaruan untuk sistem operasi Windows, aplikasi Anda, dan dependensinya. Untuk panduan tambahan tentang menjaga gambar dasar tetap up to date, lihat Keep Your AppStream 2.0 Image Update untuk panduan tambahan tentang menjaga gambar dasar tetap up to date.

Secara default, AppStream 2.0 memungkinkan pengguna atau aplikasi untuk memulai program apa pun pada instance, di luar apa yang ditentukan dalam katalog aplikasi gambar. Ini berguna ketika aplikasi Anda bergantung pada aplikasi lain sebagai bagian dari alur kerja, tetapi Anda tidak ingin pengguna dapat memulai aplikasi dependen itu secara langsung. Misalnya, aplikasi Anda memulai browser untuk memberikan instruksi bantuan dari situs web vendor aplikasi, tetapi Anda tidak ingin pengguna memulai browser secara langsung. Dalam beberapa situasi, Anda mungkin ingin mengontrol aplikasi mana yang dapat diluncurkan pada instance streaming. Microsoft AppLocker adalah perangkat lunak kontrol aplikasi yang menggunakan kebijakan kontrol eksplisit untuk mengaktifkan, atau menonaktifkan, aplikasi mana yang dapat dijalankan pengguna.

Perangkat lunak antivirus dapat mempengaruhi sesi streaming dan instance pembuat gambar. AWSmerekomendasikan agar Anda tidak mengaktifkan pembaruan otomatis untuk perangkat lunak antivirus. Untuk informasi lebih lanjut tentang Windows Defender, lihat Perangkat Lunak Antivirus.

Performa

Sebelum membuat gambar baru, penting untuk menguji aplikasi sebagai pengguna uji. Pengujian sebagai pengguna uji memungkinkan Anda memastikan bahwa aplikasi dapat berjalan di bawah konteks pengguna non-administrator. Selain itu, periksa kinerja aplikasi dan pengalaman pengguna menggunakan alat bawaan seperti Task Manager dan Performance Monitor. Ini adalah praktik terbaik untuk memantau pemanfaatan sumber daya seperti CPU, memori, dan memori GPU. Jika ada batasan sumber daya memori CPU, memori, atau GPU, pertimbangkan untuk meningkatkan jenis instance. Untuk meningkatkan kinerja:

- Nonaktifkan jendela pop-up browser
- Nonaktifkan Keamanan IE yang Ditingkatkan

AppStream 2.0 pemilihan versi agen

Saat membuat gambar baru, Anda dapat memilih untuk menggunakan perangkat lunak agen AppStream 2.0 terbaru, atau tidak memperbarui. Setiap versi perangkat lunak agen AppStream 2.0 mencakup perbaikan bug dan peningkatan fitur. Simpan gambar Anda dengan up-to-date perangkat lunak terbanyak. Tinjau mekanisme untuk ini di bagian Pembaruan gambar dokumen ini.

Anda dapat memilih opsi Gunakan agen terbaru. Opsi ini memastikan bahwa saat memulai, agen AppStream 2.0 terbaru selalu diinstal. Namun, perubahan tak terduga dapat memengaruhi pengalaman pengguna, dan pembaruan agen dapat meningkatkan waktu untuk memulai instance. Memperbarui gambar dasar membutuhkan rekreasi gambar. Penting juga bagi Anda

untuk melakukan pengujian sebelum meluncurkan gambar yang diperbarui ke produksi untuk meminimalkan waktu startup.

Antarmuka Baris Perintah Asisten Gambar (CLI)

Untuk pengembang yang ingin mengotomatiskan atau membuat gambar AppStream 2.0 secara terprogram, gunakan Image Assistant CLI. Ini tersedia di pembuat gambar dengan perangkat lunak agen AppStream 2.0 yang dirilis pada atau setelah 26 Juli 2019. Ikhtisar tingkat tinggi berikut menjelaskan proses pembuatan gambar 2.0 secara terprogram: AppStream

1. Gunakan otomatisasi instalasi aplikasi Anda untuk menginstal aplikasi yang diperlukan pada pembuat gambar Anda. Instalasi ini dapat mencakup aplikasi yang akan diluncurkan pengguna Anda, dependensi apa pun, dan aplikasi latar belakang.
2. Tentukan file dan folder untuk dioptimalkan.
3. Jika berlaku, gunakan operasi `add-application` CLI Asisten Gambar untuk menentukan metadata aplikasi dan manifes pengoptimalan untuk gambar 2.0. AppStream
4. Untuk menentukan aplikasi tambahan untuk gambar AppStream 2.0, ulangi langkah 1 hingga 3 untuk setiap aplikasi sesuai kebutuhan.
5. Jika berlaku, gunakan operasi `update-default-profile` CLI Asisten Gambar untuk menimpa profil Windows default dan membuat aplikasi default dan pengaturan Windows untuk pengguna Anda.
6. Gunakan operasi Image Assistant `create-image` CLI untuk membuat gambar.

Untuk informasi selengkapnya, lihat Buat Gambar AppStream 2.0 Anda Secara Terprogram dengan Menggunakan Operasi CLI Asisten Gambar.

Mengelola pengalaman streaming pengguna

Kustomisasi menggunakan skrip sesi

AppStream 2.0 menyediakan skrip sesi on-instance. Anda dapat menggunakan skrip ini untuk menjalankan skrip kustom Anda sendiri ketika peristiwa tertentu terjadi dalam sesi streaming pengguna. Misalnya, Anda dapat menggunakan skrip khusus untuk mempersiapkan lingkungan AppStream 2.0 Anda sebelum sesi streaming pengguna Anda dimulai. Anda juga dapat menggunakan skrip khusus untuk membersihkan instance streaming setelah pengguna menyelesaikan sesi streaming mereka.

Tentukan skrip sesi dalam gambar AppStream 2.0. Untuk informasi selengkapnya tentang mengonfigurasi skrip sesi, tinjau bagian panduan administrasi tentang penggunaan skrip sesi untuk mengelola pengalaman pengguna Anda. Digunakan dengan profil berbagi jaringan atau AWS Identity and Access Management(IAM), Anda dapat menggunakan skrip sesi untuk mengambil skrip tambahan dari lokasi penyimpanan. Dengan skrip tambahan ini, Anda dapat menjalankan pengoptimalan pengalaman pengguna lebih lanjut. Ini dapat meminimalkan jumlah gambar dan armada yang diperlukan untuk mengirimkan lingkungan aplikasi kepada pengguna Anda.

Menggunakan Kebijakan Grup Direktori Aktif

Jika Anda berencana menggunakan armada AppStream 2.0 di domain Active Directory, Anda dapat menggunakan Objek Kebijakan Grup (GPO) untuk mengelola pengalaman pengguna. GPO dapat ditugaskan ke Unit Organisasi (OU) di mana instance AppStream 2.0 dibuat. Untuk menyederhanakan pembuatan gambar, luncurkan Image AppStream 2.0 dasar di OU yang memblokir pewarisan. Ini mencegah kebijakan domain lain yang memengaruhi pengalaman pengguna AppStream 2.0. Menyebarkan setiap armada ke dalam OU khusus, dengan GPO unik yang membangun lingkungan memungkinkan manfaat one-to-many terkonsolidasi dari manajemen gambar AppStream 2.0.

Contoh penggunaan Kebijakan Grup adalah menentukan pengaturan gambar beranda Internet Explorer yang berbeda untuk setiap armada AppStream 2.0.

Pembaruan gambar

Penambalan perangkat lunak sangat penting untuk keamanan dan kinerja sumber daya komputasi. Penambalan yang sering terdaftar sebagai praktik terbaik dalam Pilar Keamanan Kerangka Well-Architected.

Saat gambar Anda dibuat dan digunakan, ada empat kategori perangkat lunak yang memerlukan penambalan pada gambar AppStream 2.0 Anda:

- Aplikasi dan dependensi — Anda bertanggung jawab untuk menambal aplikasi dan dependensi dalam gambar Anda.
- Sistem operasi Microsoft Windows — Anda bertanggung jawab untuk menginstal dan memelihara pembaruan untuk Windows.
- Komponen perangkat lunak — Ini adalah driver, agen, dan perangkat lunak lain yang diperlukan untuk operasi AppStream 2.0 (misalnya, CloudWatch agen Amazon). AppStream 2.0 secara berkala merilis gambar dasar baru yang berisi agen dan driver baru. Anda dapat membangun

kembali gambar Anda menggunakan basis terbaru untuk membawa komponen perangkat lunak pada gambar mereka ke baseline terbaru. Proses untuk membangun kembali gambar pada basis terbaru dapat memakan waktu dan rumit ketika ada banyak aplikasi, atau dengan instalasi aplikasi yang kompleks.

- AppStream Agen 2.0 - Anda dapat memilih Selalu gunakan versi agen terbaru di Image Assistant. Dengan opsi ini, instance streaming yang diluncurkan dari gambar secara otomatis menggunakan versi terbaru agen.

Anda dapat memperbarui gambar AppStream 2.0 Anda dengan melakukan salah satu hal berikut:

- Perbarui Gambar dengan Menggunakan Pembaruan Gambar AppStream 2.0 Terkelola - Metode pembaruan ini menyediakan pembaruan sistem operasi Windows terbaru dan pembaruan driver, dan perangkat lunak agen AppStream 2.0 terbaru. Metode terkelola ini memperbarui layanan dan komponen sistem operasi Microsoft, tetapi tidak memungkinkan Anda untuk memperbarui komponen aplikasi Anda. Ini adalah praktik terbaik untuk menggunakan metode ini ketika penginstalan aplikasi rumit, atau memerlukan konfigurasi manual.
- Perbarui Perangkat Lunak Agen AppStream 2.0 dengan Menggunakan Versi Gambar AppStream 2.0 Terkelola - Metode pembaruan ini menyediakan perangkat lunak agen AppStream 2.0 terbaru. Metode ini memungkinkan Anda untuk memperbarui komponen aplikasi Anda.

Kustomisasi armada

Jenis armada

Saat membuat armada, pelanggan harus memilih jenis armada. Setiap jenis armada memberikan manfaat berbeda untuk pengalaman pengguna, biaya, dan overhead pemeliharaan. Terlepas dari jenis armada yang dipilih, setiap opsi mendukung jenis platform Windows dan Linux, dan Tampilan Desktop atau Tampilan Aplikasi.

Pelanggan sekarang dapat memilih dari jenis armada berikut:

- Selalu Aktif — Jenis armada ini memberi pengguna akses instan ke aplikasi mereka. Anda akan dikenakan biaya untuk semua instans yang berjalan di armada Anda meskipun tidak ada pengguna yang melakukan streaming aplikasi.
- Sesuai Permintaan — Pilih jenis armada ini untuk mengoptimalkan biaya streaming Anda. Dengan armada sesuai permintaan, pengguna akan mengalami waktu mulai sekitar satu hingga dua menit untuk sesi mereka. Namun, Anda hanya akan dikenakan biaya instans streaming saat pengguna terhubung, dan sedikit biaya per jam untuk setiap instans di armada yang bukan aplikasi streaming.
- Elastis — Armada elastis dapat digunakan untuk aplikasi yang tidak memerlukan instalasi dan dapat dijalankan dari hard disk virtual (VHD). Armada elastis tidak mendukung gambar AppStream 2.0, juga tidak memerlukan kebijakan penskalaan. Anda hanya dikenakan biaya selama sesi streaming.

Tabel 2 - Jenis armada Amazon AppStream 2.0

Jenis armada	Kapan harus menggunakan	Pengalaman pengguna	Model harga	Catatan
Selalu Aktif	Pengguna Anda memerlukan akses instan ke aplikasi saat mereka memulai sesi. Anda tidak akan memiliki	Akses instan ke aplikasi	Anda membayar harga penuh untuk setiap instance yang tersedia di armada Anda (terlepas dari	Mendukung gambar kustom dan kebijakan penskalaan.

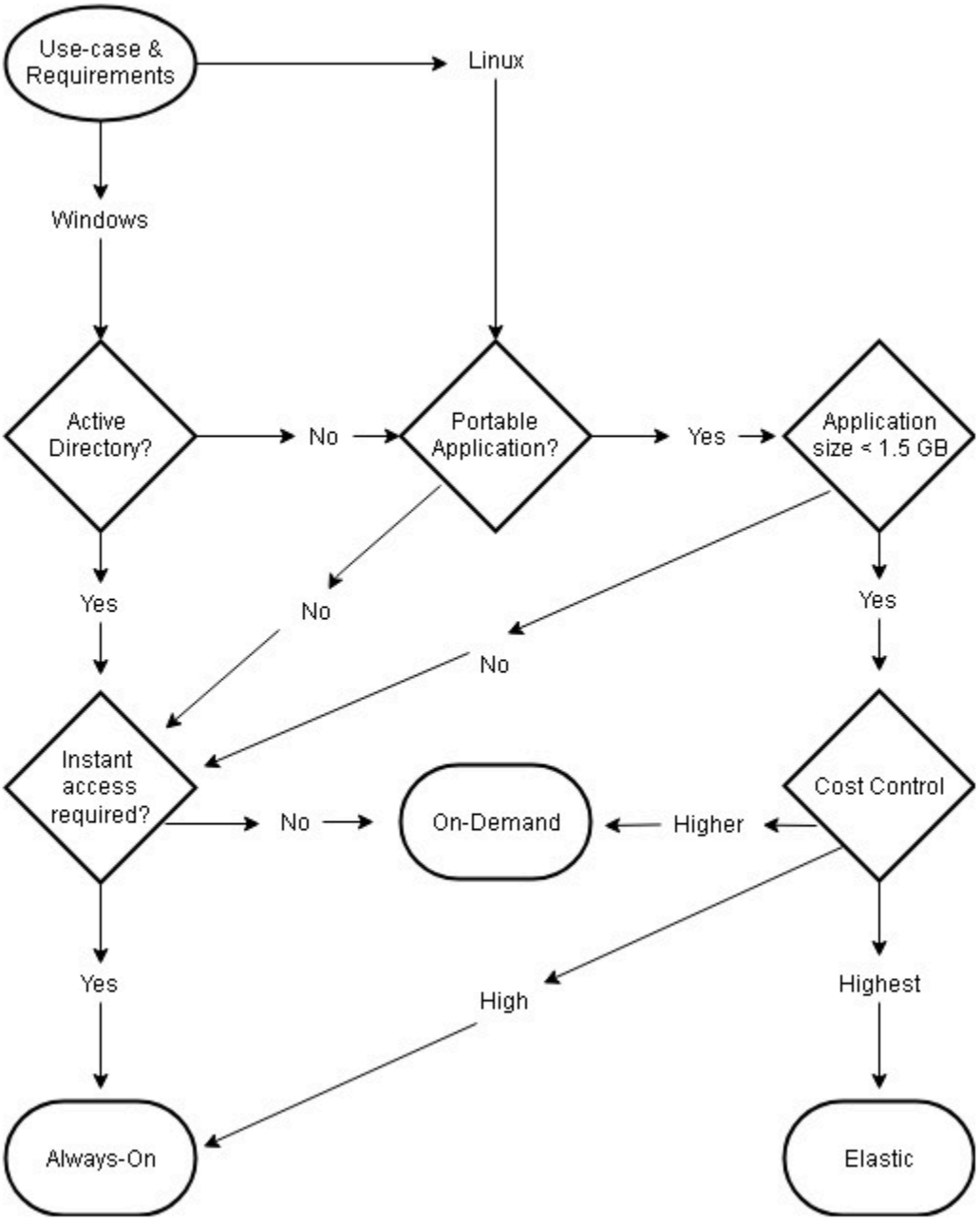
Jenis armada	Kapan harus menggunakan	Pengalaman pengguna	Model harga	Catatan
	kelebihan kapasitas yang signifikan dalam armada Anda, mungkin karena pola penggunaan Anda dapat diprediksi dan Anda dapat mengontrol biaya dengan andal dengan kebijakan penskalaan.		apakah itu digunakan untuk sesi).	

Jenis armada	Kapan harus menggunakan	Pengalaman pengguna	Model harga	Catatan
Sesuai Permintaan	Anda harus mempertah ankan kapasitas berlebih yang signifikan di armada AndaAnda menginginkan lingkungan yang paling dioptimal kan biaya dan tidak ingin membayar harga penuh untuk kapasitas yang tidak digunakan Pengguna Anda dapat menunggu satu hingga dua menit untuk mengakses aplikasi mereka setelah memulai sesi.Anda menggunakan jenis instance yang lebih besar. Biaya per jam dari instans yang sedang berjalan jauh lebih mahal daripada biaya	Pengguna menunggu satu hingga dua menit untuk mengakses aplikasi mereka setelah memulai sesi.	Anda membayar harga penuh hanya untuk streaming instans dengan sesi aktif, dan kemudian biaya per jam kecil untuk instans idle.	Mendukung gambar kustom dan kebijakan penskalaan.

Jenis armada	Kapan harus menggunakan	Pengalaman pengguna	Model harga	Catatan
	instans yang dihentikan.			

Jenis armada	Kapan harus menggunakan	Pengalaman pengguna	Model harga	Catatan
Elastis	Aplikasi Anda dan dependensinya lebih kecil dari ~ 1,5 GB. Setiap kali pengguna memulai sesi dalam armada Elastic, file hard disk virtual (VHD) Anda harus diunduh dari Amazon S3 ke dalam sesi. Akibatnya, file VHD yang lebih besar (yaitu berukuran lebih dari 1,5 GB) akan menghasilkan pengalaman pengguna akhir yang buruk. Aplikasi Anda portabel. Artinya, aplikasi Anda dan semua dependensinya dapat ditempatkan ke VHD dan diluncurkan dari VHD.Anda tidak memerlukan instance	Pengguna menunggu 45 detik hingga 3 menit untuk mengakses aplikasi setelah memulai sesi (waktu tunggu tergantung pada ukuran Virtual Hard Disk).	Anda hanya dikenakan biaya selama sesi streaming. Karena tidak ada konsep instance idle dengan armada Elastic, Anda tidak dikenakan biaya untuk instance yang tidak digunakan.	Tidak mendukung gambar kustom (pelanggan menyediakan VHD dengan aplikasi) atau kebijakan penskalaan. Saat ini mendukung `stream.standard.small` dan `stream.standard.medium` contoh. Jika kasus penggunaan Anda memerlukan jenis instans yang berbeda, silakan hubungi tim AWS akun Anda.

Jenis armada	Kapan harus menggunakan	Pengalaman pengguna	Model harga	Catatan
	streaming yang bergabung dengan domain (penggabungan domain saat ini tidak tersedia dengan armada Elastic) Anda hanya ingin membayar untuk sesi aktif (yaitu Anda tidak membayar kapasitas yang tidak digunakan di armada Anda) .Pengguna Anda dapat menunggu 45 detik atau lebih untuk mengakses aplikasi mereka setelah memulai sesi.Anda ingin AWS mengelola penskalaan untuk Anda (tidak ada kebijakan penskalaan untuk dikelola).			



Kasus dan persyaratan penggunaan tipe armada

Ukuran armada

Kapasitas minimum dan penskalaan terjadwal

Saat mengukur armada AppStream 2.0 Anda, ada beberapa pertimbangan yang langsung diterjemahkan ke pengalaman pengguna dan biaya. Nilai yang dimasukkan untuk kapasitas Minimum

memastikan bahwa jumlah instance AppStream 2.0 jarang akan kurang dari nilai ini. Setelah sesi AppStream 2.0 berakhir, jika total instance AppStream 2.0 kurang dari nilai kapasitas Minimum, instance armada baru dimulai. Seperti biasa, penting untuk mengingat satu instance AppStream 2.0 memetakan langsung ke satu sesi pengguna, yang secara langsung memengaruhi nilai untuk kapasitas Minimum.

Memasukkan nilai untuk kapasitas Minimum yang berada di luar konkurensi yang diantisipasi menghasilkan peningkatan biaya, meskipun pengalaman pengguna tidak terpengaruh. Nilai yang terlalu rendah menghasilkan biaya rendah, tetapi berdampak pada pengalaman pengguna ketika total permintaan melebihi kapasitas yang tersedia. Administrator akan mengamati kesalahan "Kapasitas Tidak Cukup" dalam situasi seperti ini. Misalnya, menunggu `PendingCapacity` menjadi `AvailableCapacity` adalah penggunaan waktu pengguna yang tidak efisien ketika jumlah koneksi yang diantisipasi pada awal hari adalah nilai yang dapat diprediksi konsisten.

Mulailah dengan kapasitas minimum yang mengakomodasi jam off-peak yang khas, dan kemudian gunakan kebijakan penskalaan terjadwal untuk secara efektif mengatur ulang kapasitas minimum sebelum dimulainya hari kerja. Jangan lupa untuk membuat kebijakan penskalaan terjadwal lainnya untuk mengembalikan kapasitas Minimum ke jam sibuk. Untuk informasi selengkapnya tentang kebijakan penskalaan dan cara menerapkannya, lihat bagian strategi auto-scaling Armada di dokumen ini.

Kapasitas maksimum dan kuota layanan

Menetapkan kapasitas maksimum mungkin tampak sebagai nilai arbitrer, tetapi ketika diperkirakan dan ditetapkan dengan benar, itu mengoptimalkan total konsumsi sumber daya dan biaya. Nilai yang dimasukkan yang lebih tinggi dari kuota layanan untuk armada AppStream 2.0 dalam Anda Akun AWS dapat tampak valid, tetapi, ketika peristiwa penskalaan otomatis mencoba untuk menskalakan sumber daya ke kapasitas maksimum, mereka gagal diluncurkan, karena nilai kapasitas maksimum melebihi kuota layanan yang tersedia. Pastikan permintaan kuota layanan ditempatkan untuk kapasitas maksimum yang diinginkan untuk memastikan fungsi penskalaan otomatis seperti yang diharapkan organisasi Anda.

Pertimbangan penting lainnya ketika menetapkan nilai kapasitas maksimum adalah biaya. Untuk informasi lebih lanjut, lihat bagian Mengoptimalkan biaya dengan pilihan jenis armada dokumen ini.

Memilih Tampilan Desktop atau Tampilan Aplikasi

Penentuan untuk memilih tampilan aplikasi atau tampilan desktop tidak berdampak pada kinerja atau biaya. Hanya satu tampilan yang dapat diakses pada waktu tertentu per AppStream 2,0 armada. Anda dapat mengubah opsi tampilan Stream. Rencanakan perubahan ini selama jam kerja di luar puncak, karena mengubah tampilan aliran memerlukan restart armada.

Tidak ada satu pun praktik terbaik untuk tampilan streaming. Dampak opsi tampilan aliran dirangkum melalui yang berikut:

- Pelaporan terperinci untuk penggunaan aplikasi melalui fitur Laporan Penggunaan untuk administrator
- Pengalaman dan alur kerja keseluruhan untuk pengguna akhir (misalnya, apakah desktop lengkap memenuhi kebutuhan kasus penggunaan atau hanya melihat aplikasi saja sudah cukup?).

Tampilan Desktop

Untuk kasus penggunaan di mana semua alur kerja pengguna dilakukan dalam sesi, Tampilan Desktop menyederhanakan pengalaman pengguna dengan menjaga semua aplikasi tetap fokus dalam satu lingkungan. Desktop View dapat memberikan pengalaman pengguna yang lebih konsisten untuk penerapan lebih dari 3-5 aplikasi yang memerlukan integrasi dengan sistem operasi (OS). Tampilan Desktop efektif saat mempertahankan dua lingkungan yang terpisah dan berbeda. Misalnya, pengguna dapat memiliki akses bersamaan ke lingkungan desktop produksi dan pra-produksi untuk memvalidasi perubahan tata letak, konfigurasi, dan akses aplikasi.

AppStream 2.0 Usage Reports membuat laporan aplikasi harian untuk Tampilan Desktop. Output yang dihasilkan untuk aplikasi hanya 'desktop', pemetaan langsung ke sesi AppStream 2.0. Untuk informasi selengkapnya, lihat bagian Pemantauan penggunaan pengguna pada dokumen ini.

Aplikasi Hanya melihat

Tampilan Applications Only juga efektif ketika tumpukan AppStream 2.0 dimaksudkan untuk mengirimkan beberapa aplikasi yang diperlukan sebentar-sebentar. Di lingkungan kios, pengiriman aplikasi yang terkunci dengan aman dikirimkan melalui Tampilan Aplikasi. Dengan Application View, AppStream 2.0 menggantikan shell Windows default dengan shell khusus. Shell khusus ini hanya menyajikan aplikasi yang berjalan, meminimalkan permukaan serangan OS.

Untuk kasus penggunaan di mana AppStream 2.0 digunakan untuk menambah lingkungan desktop organisasi yang ada, tampilan Applications Only lebih disukai. Menyebarkan AppStream 2.0 Windows Client dalam mode aplikasi asli untuk meminimalkan kebingungan pengguna dengan memungkinkan penggunaan penuh pintasan keyboard.

Laporan Penggunaan Amazon 2.0 membuat laporan aplikasi harian untuk tampilan aplikasi. Untuk pelaporan aplikasi dan penggunaan yang lebih terperinci, pertimbangkan solusi pihak ketiga untuk melaporkan di tingkat sistem operasi. Anda dapat menggunakan Microsoft AppLocker dalam mode pelaporan, atau mempertimbangkan solusi yang tersedia diAWS Marketplace, seperti Stratusphere UX Liquidware.

AWS Identity and Access Managementkonfigurasi peran

Jika beban kerja mengharuskan pengguna akhir AppStream 2.0 untuk mengakses AWS layanan lain dari dalam sesi mereka, itu adalah praktik terbaik untuk mendelegasikan akses melalui penggunaan peran AWS Identity and Access Management(IAM). Peran IAM dapat langsung dilampirkan ke sesi pengguna akhir Anda melalui penugasan di tingkat armada. Untuk praktik terbaik tambahan saat menggunakan peran IAM dengan AppStream 2.0, lihat bagian panduan administrator ini.

Menggunakan kredensil statis

Beberapa beban kerja mungkin memerlukan input statis untuk kunci akses IAM yang bertentangan dengan mewarisinya dari peran terlampir. Ada dua metode untuk menerima kredensil ini. Metode pertama melibatkan penyimpanan kunci akses dalam AWS layanan dan kemudian memberikan pengguna akhir Anda akses IAM eksplisit untuk menarik nilai tertentu dari layanan. Dua contoh mekanisme penyimpanan kunci akses menggunakan AWS Secrets Manageratau AWSSSM Parameter Store. Metode kedua adalah menggunakan penyedia kredensi AppStream 2.0 untuk mengakses kunci akses peran terlampir. Ini dapat dilakukan dengan memanggil penyedia kredensi dan mengurai output untuk kunci akses dan kunci rahasia Anda. Contoh bagaimana melakukan tindakan ini dalam PowerShell berikut.

```
$CMD = 'C:\Program Files\Amazon\Photon\PhotonRoleCredentialProvider
\PhotonRoleCredentialProvider.exe'
$role = 'Machine'

$output = & $CMD --role=$role
$parsed = $output | ConvertFrom-Json

$access_key = $parsed.AccessKeyId
```

```
$secret_key = $parsed.SecretAccessKey
$session_token = $parsed.SessionToken
```

Melindungi bucket AppStream 2.0 S3 Anda

Jika beban kerja AppStream 2.0 Anda dikonfigurasi dengan Folder Rumah dan/atau Persistensi Aplikasi, maka praktik terbaik adalah melindungi bucket Amazon S3 tempat data persisten disimpan dari akses tidak sah atau penghapusan yang tidak disengaja. Lapisan perlindungan pertama adalah menambahkan kebijakan bucket Amazon S3 untuk mencegah penghapusan bucket secara tidak sengaja. Lapisan perlindungan kedua adalah menambahkan kebijakan bucket yang sejalan dengan prinsip hak istimewa terkecil. Menyelaraskan dengan prinsip dapat dilakukan dengan hanya mengizinkan akses bucket ke pihak-pihak yang diperlukan.

Strategi penskalaan otomatis armada

Memahami contoh AppStream 2.0

AppStream Instance armada 2.0 memiliki rasio instans pengguna ke armada 1:1. Ini berarti setiap pengguna memiliki instance streaming mereka sendiri. Jumlah pengguna yang Anda sambungkan secara bersamaan akan menentukan ukuran armada.

Kebijakan penskalaan

AppStream Armada 2.0 diluncurkan di Grup Application Auto Scaling. Hal ini memungkinkan armada untuk skala berdasarkan penggunaan untuk memenuhi permintaan. Saat penggunaan meningkat, armada akan meningkat, dan saat pengguna memutuskan sambungan, armada kembali masuk. Ini dikendalikan dengan menetapkan kebijakan penskalaan. Anda dapat mengatur kebijakan penskalaan berbasis terjadwal, penskalaan langkah, dan penskalaan pelacakan target. Untuk informasi selengkapnya tentang kebijakan penskalaan ini, lihat Fleet Auto Scaling untuk AppStream Amazon 2.0.

Penskalaan langkah

Kebijakan ini meningkatkan atau mengurangi kapasitas armada dengan persentase dari ukuran armada saat ini atau sejumlah contoh tertentu. Kebijakan penskalaan langkah dipicu oleh AppStream 2.0 CloudWatch metrik`Capacity Utilization`,`Available Capacity`, atau. `Insufficient Capacity Errors`

Saat menggunakan kebijakan penskalaan langkah, AWS sarankan Anda menambahkan persentase kapasitas dan bukan jumlah instans yang tetap. Ini memastikan bahwa tindakan penskalaan Anda sebanding dengan ukuran armada Anda. Ini akan membantu untuk menghindari situasi di mana Anda skala terlalu lambat (karena Anda menambahkan sejumlah kecil contoh relatif terhadap ukuran armada Anda) atau terlalu banyak contoh ketika armada Anda kecil.

Pelacakan target

Dengan kebijakan ini menetapkan tingkat pemanfaatan kapasitas untuk armada. Application Autoscaling membuat dan mengelola CloudWatch alarm yang memicu kebijakan penskalaan. Ini menambah atau menghapus kapasitas untuk menjaga armada pada, atau mendekati, nilai target yang ditentukan. Untuk memastikan ketersediaan aplikasi, armada Anda menskalakan secara proporsional dengan metrik secepat mungkin, tetapi skala lebih bertahap. Saat mengonfigurasi

pelacakan target, pertimbangkan cooldown penskalaan untuk memastikan penskalaan dan penskalaan terjadi dalam interval yang diinginkan.

Pelacakan target efektif untuk situasi churn tinggi. Churn adalah ketika sejumlah besar pengguna memulai, atau mengakhiri, sesi dalam waktu singkat. Anda dapat mengidentifikasi churn dengan memeriksa CloudWatch metrik untuk armada Anda. Periode waktu ketika armada Anda memiliki kapasitas tertunda non-nol tanpa perubahan (atau dengan sedikit perubahan) dalam kapasitas yang diinginkan menunjukkan bahwa churn tinggi kemungkinan terjadi. Dalam situasi churn tinggi, konfigurasikan kebijakan pelacakan target di mana (100 - persen pemanfaatan target) lebih dari tingkat churn dalam periode 15 menit. Misalnya, jika 10% armada Anda akan dihentikan dalam 15 menit karena omset pengguna, tetapkan target pemanfaatan kapasitas 90% atau kurang untuk mengimbangi churn tinggi.

Penskalaan berbasis terjadwal

Kebijakan ini memungkinkan Anda untuk mengatur kapasitas armada yang diinginkan berdasarkan jadwal berbasis waktu. Kebijakan ini efektif ketika Anda memahami perilaku login, dan dapat memprediksi perubahan permintaan.

Misalnya, pada awal hari kerja, Anda mungkin mengharapkan 100 pengguna untuk meminta koneksi streaming pada pukul 9:00 pagi. Anda dapat mengonfigurasi kebijakan penskalaan berbasis terjadwal untuk menetapkan ukuran armada minimum menjadi 100 pada pukul 8:40 pagi. Hal ini memungkinkan instance armada dibuat dan tersedia pada awal hari kerja, dan memungkinkan 100 pengguna untuk terhubung pada saat yang sama. Anda kemudian dapat menetapkan kebijakan terjadwal lain untuk menskalakan armada menjadi minimal sepuluh pada pukul 17:00. Ini memungkinkan Anda menghemat biaya, karena permintaan untuk sesi setelah jam kerja kurang dari pada hari kerja.

Kebijakan penskalaan dalam produksi

Anda dapat memilih untuk menggabungkan berbagai jenis kebijakan penskalaan dalam satu armada untuk membantu menentukan kebijakan penskalaan yang tepat untuk perilaku pengguna Anda. Pada contoh sebelumnya, Anda dapat menggabungkan kebijakan penskalaan terjadwal dengan kebijakan pelacakan target atau penskalaan langkah untuk mempertahankan tingkat pemanfaatan tertentu. Kombinasi penskalaan terjadwal dan penskalaan pelacakan target dapat membantu mengurangi dampak peningkatan tajam dalam tingkat pemanfaatan ketika kapasitas dibutuhkan segera.

Pengguna yang terhubung ke sesi streaming ketika kebijakan penskalaan mengubah jumlah instans yang diinginkan tidak terpengaruh oleh scale-in atau scale-out. Kebijakan penskalaan tidak akan

mengakhiri sesi streaming yang ada. Sesi yang ada akan terus berlanjut tanpa gangguan hingga sesi berakhir oleh pengguna atau kebijakan batas waktu armada.

Memantau penggunaan AppStream 2.0 dengan CloudWatch metrik dapat membantu Anda mengoptimalkan kebijakan penskalaan dari waktu ke waktu. Misalnya, adalah umum untuk menyediakan sumber daya yang berlebihan selama penyiapan awal dan Anda mungkin melihat periode pemanfaatan rendah yang lama. Atau, jika armada kurang disediakan, Anda mungkin melihat penggunaan kapasitas tinggi dan kesalahan “Kapasitas Tidak Cukup”. Meninjau CloudWatch metrik dapat membantu mendorong penyesuaian pada kebijakan penskalaan Anda untuk membantu mengurangi kesalahan ini. Untuk informasi selengkapnya, dan contoh kebijakan penskalaan AppStream 2.0 yang dapat Anda gunakan, lihat Menskalakan armada Amazon AppStream 2.0 Anda.

Praktik terbaik untuk desain kebijakan penskalaan

Menggabungkan kebijakan penskalaan

Banyak pelanggan memilih untuk menggabungkan berbagai jenis kebijakan penskalaan dalam satu armada untuk meningkatkan daya dan fleksibilitas Auto Scaling AppStream di 2.0. Misalnya, Anda dapat mengonfigurasi kebijakan penskalaan terjadwal untuk meningkatkan minimum armada Anda pada pukul 6:00 pagi untuk mengantisipasi pengguna memulai hari kerja mereka, dan untuk mengurangi minimum armada pada pukul 16:00 sebelum pengguna berhenti bekerja. Anda dapat menggabungkan kebijakan penskalaan terjadwal ini dengan pelacakan target atau kebijakan penskalaan langkah untuk mempertahankan tingkat pemanfaatan tertentu dan penskalaan atau - out pada siang hari untuk menangani penggunaan runcing. Kombinasi penskalaan terjadwal dan penskalaan pelacakan target dapat membantu mengurangi dampak peningkatan tajam dalam tingkat pemanfaatan ketika kapasitas dibutuhkan segera.

Hindari penskalaan churn

Pertimbangkan apakah armada Anda mungkin mengalami churn tingkat tinggi karena kasus penggunaan Anda. Churn terjadi ketika sejumlah besar pengguna memulai dan kemudian mengakhiri sesi dalam waktu singkat. Ini mungkin terjadi ketika banyak pengguna secara bersamaan mengakses aplikasi di armada Anda hanya beberapa menit sebelum menandatangani.

Dalam situasi seperti itu, ukuran armada Anda mungkin turun jauh di bawah kapasitas yang diinginkan, karena instance berakhir ketika pengguna mengakhiri sesi mereka. Kebijakan penskalaan langkah mungkin tidak menambahkan instance dengan cukup cepat untuk mengimbangi churn dan, akibatnya, armada Anda macet pada ukuran tertentu.

Anda dapat mengidentifikasi churn dengan memeriksa CloudWatch metrik untuk armada Anda. Periode waktu ketika armada Anda memiliki kapasitas tertunda bukan nol tanpa perubahan (atau dengan sedikit perubahan) dalam kapasitas yang diinginkan menunjukkan bahwa churn tinggi kemungkinan terjadi. Untuk memperhitungkan situasi churn yang tinggi, gunakan kebijakan penskalaan pelacakan target dan pilih pemanfaatan target sehingga (100 - persen pemanfaatan target) lebih dari tingkat churn dalam periode 15 menit. Misalnya, jika 10% armada Anda akan berakhir dalam periode 15 menit karena omset pengguna, tetapkan target pemanfaatan kapasitas 90% atau kurang untuk mengimbangi churn tinggi.

Memahami tingkat penyediaan maksimum

Pelanggan yang mengelola AppStream 2.0 armada untuk sejumlah besar pengguna harus mempertimbangkan batas tarif penyediaan. Batas ini akan berdampak pada seberapa cepat instance dapat ditambahkan ke armada atau di semua armada dalam sebuah armada. Akun AWS

Ada dua batasan yang perlu dipertimbangkan:

- Untuk satu armada, AppStream 2.0 ketentuan dengan kecepatan maksimum 20 instance per menit.
- Untuk satuAkun AWS, ketentuan AppStream 2.0 dengan kecepatan 60 instance per menit (dengan ledakan 100 instance per menit).

Jika lebih dari tiga armada ditingkatkan secara paralel, batas tingkat penyediaan akun dibagi di seluruh armada ini (misalnya, enam armada penskalaan secara paralel dapat masing-masing menyediakan hingga 10 instance per menit). Selain itu, pertimbangkan jumlah waktu instans streaming tertentu untuk menyelesaikan penyediaan sebagai respons terhadap peristiwa penskalaan. Untuk armada yang tidak bergabung dengan domain Active Directory, ini biasanya 15 menit. Untuk armada yang bergabung dengan domain Active Directory, ini bisa memakan waktu selama 25 menit.

Mengingat kendala tersebut, pertimbangkan contoh-contoh berikut:

- Jika Anda ingin menskalakan satu armada dari 0 hingga 1000 instans, dibutuhkan waktu 50 menit (1000 instans/20 instans per menit) agar penyediaan selesai, dan kemudian tambahan 15-25 menit agar semua instans tersedia untuk pengguna akhir, dengan total 65-75 menit.
- Jika Anda ingin secara bersamaan menskalakan tiga armada dari 0 hingga 333 instance (untuk total 999 instance dalamAkun AWS), dibutuhkan sekitar 17 menit (999/60 instans per menit) untuk semua armada untuk menyelesaikan penyediaan dan kemudian tambahan 15 menit agar instans tersebut tersedia untuk pengguna akhir, dengan total 32-42 menit.

Memanfaatkan beberapa Availability Zone

Pilih beberapa AZ di Wilayah untuk penyebaran armada Anda. Ketika Anda memilih beberapa AZ untuk armada Anda, Anda meningkatkan kemungkinan armada Anda akan dapat menambahkan instance sebagai respons terhadap peristiwa penskalaan. CloudWatch Metrik PendingCapacity ini merupakan titik awal untuk menilai seberapa optimal desain armada AZ dalam penyebaran armada besar. Nilai yang tinggi dan berkelanjutan untuk PendingCapacity dapat menunjukkan

kebutuhan untuk memperluas penskalaan horizontal (melintasi AZ). Untuk informasi selengkapnya, lihat Memantau Sumber Daya Amazon AppStream 2.0.

Misalnya, jika penskalaan otomatis mencoba menyediakan instans untuk meningkatkan ukuran armada Anda dan AZ yang dipilih memiliki kapasitas yang tidak mencukupi, penskalaan otomatis akan menambahkan instance di AZ lain yang telah Anda tentukan untuk armada Anda. Untuk informasi selengkapnya tentang Availability Zones dan desain AppStream 2.0, lihat Availability Zone dalam dokumen ini.

Pantau metrik Kesalahan Kapasitas Tidak Cukup

“Kesalahan Kapasitas Tidak Cukup” adalah CloudWatch metrik untuk armada AppStream 2.0. Metrik ini menentukan jumlah permintaan sesi yang ditolak karena kurangnya kapasitas.

Ketika Anda membuat perubahan pada kebijakan penskalaan Anda, akan sangat membantu untuk membuat CloudWatch alarm untuk memberi tahu Anda ketika terjadi Kesalahan Kapasitas Tidak Cukup. Ini memungkinkan Anda menyesuaikan kebijakan penskalaan dengan cepat untuk mengoptimalkan ketersediaan bagi pengguna. Panduan administrasi memberikan langkah-langkah rinci untuk memantau sumber daya AppStream 2.0 Anda.

Metode koneksi

Saat streaming sesi di AppStream 2.0, pengguna memiliki dua metode koneksi yang tersedia:

- Akses Browser Web - Browser berkemampuan HTML5 apa pun didukung. Tidak diperlukan plug- in atau unduhan.
- AppStream 2.0 Klien Windows

Sebagai praktik terbaik, pertimbangkan persyaratan fitur dan perangkat untuk kasus penggunaan pengguna Anda untuk menyelaraskan browser, atau perangkat mana, yang paling mendukung persyaratan mereka.

> **Note**
>
> AppStream 2.0 tidak didukung pada perangkat yang memiliki resolusi layar lebih kecil dari 1024 x 768 piksel.

Fitur ringkasan dan dukungan perangkat

Tabel 3 - Fitur ringkasan dan dukungan perangkat

	Akses browser web	AppStream 2.0 Klien Windows
Beberapa monitor (hingga resolusi 2k)	Didukung	Didukung
Beberapa monitor (hingga resolusi 4k)	T/A	Didukung
Menggambar dukungan tablet	Didukung *	Didukung
Dukungan perangkat layar sentuh	Didukung	T/A
Dukungan perangkat passthrough USB	T/A	Didukung

	Akses browser web	AppStream 2.0 Klien Windows
Pintasan keyboard	Didukung	Didukung
Offset mouse relatif	Didukung	Didukung
Transfer file	Didukung	Didukung
Pengalihan printer lokal	T/A	Didukung
Pengalihan drive lokal	T/A	Didukung
Dukungan web-cam	Didukung	Didukung

*Hanya Google Chrome dan Mozilla Firefox

Akses browser web

AppStream Akses browser web 2.0 memungkinkan akses ke aplikasi tanpa perlu menginstal klien khusus. Pengguna dapat terhubung menggunakan browser berkemampuan HTML5 yang didukung. Tidak ada persyaratan untuk plugin atau ekstensi browser apa pun.

Akses browser web menyediakan berbagai pilihan sistem dan jenis operasi perangkat akhir.

AppStream 2.0 klien untuk Windows

Klien AppStream 2.0 untuk Windows adalah aplikasi yang Anda instal di PC Windows Anda. Aplikasi ini memberikan kemampuan tambahan yang tidak tersedia saat Anda mengakses AppStream 2.0 menggunakan browser web. Misalnya, AppStream klien memungkinkan Anda melakukan hal berikut:

- Gunakan lebih dari dua monitor atau resolusi 4K
- Gunakan perangkat USB Anda dengan aplikasi yang dialirkan melalui 2.0 AppStream
- Akses drive dan folder lokal Anda selama sesi streaming
- Arahkan ulang pekerjaan cetak dari aplikasi streaming Anda ke printer yang terhubung ke komputer lokal Anda
- Gunakan webcam lokal Anda untuk konferensi video dan audio dalam sesi streaming Anda
- Gunakan pintasan keyboard di aplikasi yang diakses selama sesi streaming

- Berinteraksi dengan aplikasi streaming jarak jauh Anda dengan cara yang sama seperti Anda berinteraksi dengan aplikasi yang diinstal secara lokal

AppStream 2.0 mode koneksi klien

Klien AppStream 2.0 menyediakan dua mode koneksi: mode aplikasi asli dan mode klasik. Mode koneksi yang Anda pilih menentukan opsi yang tersedia untuk Anda selama streaming aplikasi, dan bagaimana fungsi dan tampilan aplikasi streaming Anda. Administrator dapat mengontrol kemampuan pengguna untuk beralih antara mode aplikasi asli dan mode klasik.

- Mode klasik mengalirkan aplikasi di jendela sesi AppStream 2.0. Ini mirip dengan bagaimana pengguna akhir mengalirkan aplikasi di browser web. Gunakan mode klasik jika pengguna akhir lebih suka melakukan streaming aplikasi dengan cara yang sama seperti browser, sambil memanfaatkan fitur tambahan seperti koneksi untuk file lokal dan pengalihan printer. Mode klasik adalah mode koneksi default yang disarankan. Mode klasik adalah satu-satunya mode yang didukung untuk Tampilan Desktop.
- Mode aplikasi asli memungkinkan pengguna akhir untuk bekerja dengan aplikasi streaming jarak jauh dengan cara yang sama seperti aplikasi yang diinstal secara lokal lainnya. Jika pengguna akhir terbiasa bekerja dengan aplikasi yang diinstal secara lokal, mode aplikasi asli memberikan pengalaman yang mulus. Aplikasi streaming jarak jauh berfungsi dengan cara yang sama seperti aplikasi yang diinstal secara lokal. Ikon aplikasi ditampilkan di bilah tugas PC lokal Anda, seperti halnya ikon untuk aplikasi lokal Anda. Berbeda dengan ikon untuk aplikasi lokal Anda, ikon untuk aplikasi streaming Anda dalam mode aplikasi asli termasuk logo AppStream 2.0. Mode aplikasi asli adalah mode koneksi yang disarankan ketika pengguna ingin menggunakan pintasan keyboard aplikasi, dan mudah beralih antara aplikasi jarak jauh lokal dan individu menggunakan pintasan keyboard.

Penyebaran dan manajemen klien

Pengguna dapat menginstal klien AppStream 2.0 sendiri, atau administrator dapat menginstal klien AppStream 2.0 untuk mereka dengan menjalankan PowerShell skrip dari jarak jauh, atau mengemas ulang klien AppStream 2.0 dengan pengaturan yang disesuaikan.

Anda harus memenuhi syarat perangkat USB yang ingin Anda aktifkan pengguna Anda untuk digunakan dengan sesi streaming mereka. Jika perangkat USB mereka tidak memenuhi syarat, itu tidak akan terdeteksi oleh AppStream 2.0 dan tidak dapat dibagikan dengan sesi. Setelah perangkat

mereka memenuhi syarat, pengguna Anda harus berbagi perangkat dengan AppStream 2.0 setiap kali mereka memulai sesi streaming baru.

Saat menerapkan klien AppStream 2.0 dalam skala besar, AWS rekomendasikan untuk menggunakan Enterprise Deployment Tool. Enterprise Deployment Tool mencakup file instalasi AppStream klien dan template administratif Kebijakan Grup.

Domain kustom

Saat menerapkan AppStream 2.0 secara terprogram, dimungkinkan untuk membuat domain khusus yang dapat memberi pengguna pengalaman yang akrab untuk sesi streaming. Dalam penerapan AppStream SAMP 2.0 iDP 2.0, penting untuk menyoroti bahwa akses pengguna dimulai pada iDP, bukan 2.0. AppStream Pengguna tidak memerlukan AppStream 2.0 URL, karena ini disediakan oleh IDP setelah otentikasi. Oleh karena itu, nama domain khusus tidak diperlukan untuk penerapan IDP SAMP 2.0.

Autentikasi

Dengan AppStream 2.0, otentikasi dapat dilakukan di luar Amazon AppStream 2.0, atau sebagai bagian dari layanan AppStream 2.0. Memilih bagaimana otentikasi akan berlangsung untuk penerapan AppStream 2.0 Anda adalah pertimbangan mendasar dari desain Anda. Tidak jarang organisasi memiliki beberapa penerapan AppStream 2.0 untuk kasus penggunaan yang berbeda. Setiap kasus penggunaan dapat memiliki metode otentikasi yang berbeda.

Ada tiga jenis metode otentikasi untuk AppStream 2.0:

- SAMP 2.0
- Kolam Pengguna
- Terprogram

Menentukan metode yang dioptimalkan

Amazon AppStream 2.0 dirancang agar fleksibel untuk diterapkan pada sebagian besar persyaratan desain organisasi. Ketika menentukan metode yang dioptimalkan untuk otentikasi, itu adalah praktik terbaik untuk mempertimbangkan tujuan dan tujuan dari mereka yang mengkonsumsi layanan, dan kebijakan dan prosedur organisasi.

Berikut adalah beberapa contoh menggabungkan kasus penggunaan dengan tujuan organisasi.

Tabel 4 - Gunakan kasus dengan tujuan organisasi

Contoh	Deskripsi	Autentikasi
Instance armada yang bergabung dengan domain diperlukan	Aplikasi yang diinstal pada AppStream gambar hanya dapat diakses oleh sumber daya yang bergabung dengan domain.	SAML 2.0
Integrasi berat dengan layanan Microsoft	Ketergantungan organisasi pada pengembangan Kebijakan Microsoft Group dan infrastruktur backend	SAML 2.0

Contoh	Deskripsi	Autentikasi
Perusahaan yang ada Single Sign-on (SSO)	Semua layanan baru harus memanfaatkan solusi SSO perusahaan yang memiliki beberapa proses pelaporan dan keamanan yang ditetapkan.	SAML 2.0
Dukungan kartu pintar untuk aplikasi	Kartu pintar (seperti Verifikasi Identitas Pribadi dan kartu akses umum) untuk otentikasi dalam sesi ke aplikasi streaming melalui pembaca kartu pintar.	SAML 2.0
Tenaga kerja musiman dengan staf sementara	Beberapa bulan dalam setahun, pekerja sementara diberi satu set kecil aplikasi yang tidak termasuk sumber daya internal untuk menyelesaikan kegiatan.	Kolam Pengguna
Dukungan TI Terbatas	Organisasi yang lebih kecil dengan kurang dari 50 pengguna dan staf TI terbatas, ingin menghapus biaya pemeliharaan Penyedia Identitas (iDP)	Kolam Pengguna
Vendor Perangkat Lunak Independen (ISV)	Solusi berpemilik yang dibangun oleh organisasi Anda yang mencakup hak pengguna dan otentikasi, memperluas AppStream 2.0 sebagai bagian dari solusi Anda. *	Terprogram

Contoh	Deskripsi	Autentikasi
Pameran teknologi	Lingkungan yang sepenuhny a fana yang menampilkan teknologi eksklusif sebagai bagian dari tur berpemandu solusi Anda tanpa persyarat an untuk menyimpan informasi pengguna.	Terprogram
Pengalaman situs web interaktif	Jadikan situs web Anda interaktif dengan streaming aplikasi Windows. **	Terprogram

*Lihat vendor Perangkat Lunak: Kirimkan aplikasi Anda ke perangkat pengguna apa pun untuk informasi lebih lanjut.

**Lihat Sematkan Sesi Streaming AppStream 2.0 untuk informasi lebih lanjut.

Jika organisasi Anda memiliki kasus penggunaan atau kebijakan yang tidak tercantum dalam contoh yang diberikan sebelumnya, merupakan praktik terbaik untuk memperkirakan status akhir yang diinginkan dari konsumsi alur kerja AppStream 2.0 untuk memastikan solusi otentikasi tidak bertentangan dengannya.

Mengkonfigurasi penyedia identitas Anda

SAML 2.0

Security Assertion Markup Language (SAMP) 2.0 adalah opsi penerapan umum untuk memungkinkan pengguna menggunakan sumber daya. AWS Berbagai penyedia identitas SAMP 2.0 pihak ketiga mendukung AppStream 2.0. Apakah sumber daya AppStream 2.0 Anda bergabung dengan domain atau tidak, SAMP 2.0 iDP mengharuskan Anda untuk menggunakan IAM.

Karena sebagian besar IdPs menghasilkan metadata.xml unik dengan atribut SAMP tertentu untuk setiap aplikasi SAMP, setiap tumpukan AppStream 2.0 memerlukan Peran yang memiliki hubungan tepercaya dengan SAMP iDP dan Kebijakan yang memiliki satu izin ke AppStream: Stream dengan kondisi yang sesuai dengan persyaratan SAMP iDP dan ARN dari 2.0 Stack. AppStream

Panduan administrasi AppStream 2.0 memberikan contoh konfigurasi untuk desain tumpukan AppStream 2.0 tunggal. Untuk beberapa penerapan tumpukan, lihat langkah-langkah opsional untuk menggunakan katalog aplikasi multi-stack SAMP 2.0.

Kolam pengguna

Tab User Pool di AppStream 2.0 adalah opsi yang valid untuk bukti konsep kecil. Sebagai praktik terbaik, yang terbaik adalah menghindari kumpulan pengguna untuk kasus penggunaan dan organisasi apa pun yang menggunakan AppStream 2.0 untuk mengirimkan aplikasi produksi.

Satu hal penting yang perlu diperhatikan tentang kumpulan pengguna adalah bahwa alamat email pengguna peka huruf besar/kecil; oleh karena itu merupakan praktik terbaik untuk memastikan pengguna dididik tentang cara memasukkan kredensyal pengguna dengan benar.

URL streaming

Untuk penerapan yang memanggil sumber daya AppStream 2.0 dari layanan terpusat (biasanya ISV), otentikasi terprogram bergantung pada aplikasi untuk melakukan panggilan terprogram untuk meneruskan informasi secara dinamis dan membuat sesi 2.0 AWS untuk penggunanya. AppStream Gunakan metode otentikasi API (biasanya disebut sebagai 'programmatic') saat membuat URL streaming menggunakan operasi URL. CreateStreaming Pengguna yang melakukan `CreateStreamingURL` panggilan harus menggunakan pengguna atau peran yang valid dengan izin untuk`appstream:CreateStreamingURL`.

Saat membuat kebijakan untuk akses terprogram, adalah praktik terbaik untuk mengamankan akses dengan menentukan AppStream 2.0 Stack ARN yang tepat di bagian Sumber Daya sebagai pengganti default '*'. Sebagai contoh:

Example

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "appstream:createStreamingURL"
            ],
            "Resource": "arn:aws:appstream:us-east-
1:031421429609:stack/BestPracticesStack"
```

```
        }
    ]
}
```

> **Note**
>
> Anda dapat dengan cepat mengambil ARN AppStream 2.0 Stacks Anda dengan menggunakan describe stacks API atau AWS CLI.

AppStream 2.0 instance harus dimulai sebagai contoh generik. Melalui informasi yang diteruskan kepadanya dari aplikasi, instance AppStream 2.0 menetapkan lingkungan menggunakan konteks sesi untuk membuat hal-hal dinamis bagi pengguna.

Meskipun GPO lokal dapat digunakan untuk menentukan pengaturan saat login pengguna, konteks sesi adalah praktik terbaik saat menggunakan`CreateStreamingURL`, dan meneruskan atribut kunci seperti ID Pelanggan atau pengaturan koneksi database, untuk digunakan dalam sesi. AppStream

Hak aplikasi

AppStream 2.0 dapat secara dinamis membangun katalog aplikasi yang disajikan kepada pengguna. Hak aplikasi didasarkan pada atribut SAMP 2.0, atau dengan menggunakan AppStream 2.0 Dynamic Application Framework.

Hak aplikasi berbasis atribut menggunakan SAMP 2.0 direkomendasikan di sebagian besar skenario. Untuk mengelola pengiriman paket aplikasi, Dynamic Application Framework direkomendasikan.

Integrasi dengan Microsoft Active Directory

Amazon AppStream 2.0 Image Builders dan armada dapat diintegrasikan dengan Microsoft Active Directory. Ini memungkinkan Anda untuk menyediakan metode terpusat untuk otentikasi pengguna, otorisasi, dan untuk menerapkan kebijakan Grup Direktori Aktif ke instance 2.0 yang bergabung dengan AppStream domain. Menggunakan AppStream armada yang bergabung dengan domain memberikan manfaat administratif yang sama dengan lingkungan lokal. Ini termasuk manajemen terpusat dari berbagi file jaringan, hak pengguna-aplikasi, profil roaming, akses printer, dan pengaturan berbasis kebijakan lainnya.

Saat mengintegrasikan lingkungan AppStream 2.0 dengan Active Directory, penting untuk dicatat bahwa otentikasi awal ke tumpukan AppStream 2.0 masih dikelola oleh IDP SAML2.0. Setelah pengguna berhasil diautentikasi ke IDP, ketika pengguna meluncurkan sesi, mereka harus memasukkan kata sandi domain mereka atau otentikasi kartu pintar untuk domain Active Directory.

Saat merancang lingkungan Active Directory Domain Services (ADDS) yang akan digunakan dengan AppStream 2.0, ada dua opsi layanan dan banyak skenario penyebaran yang tersedia. Juga, pastikan bahwa jaringan AppStream 2.0 ditinjau dengan pemilik topologi situs Active Directory Anda.

Opsi layanan

Active Directory juga dapat digunakan menggunakan AWSManaged Microsoft Active Directory (AD). AWS Microsoft AD yang dikelola adalah layanan yang dikelola sepenuhnya yang memungkinkan Anda menjalankan Microsoft Active Directory. Microsoft Active Directory juga dapat digunakan di lingkungan yang dihosting sendiri, berjalan di EC2 atau lokal.

Skenario penyebaran

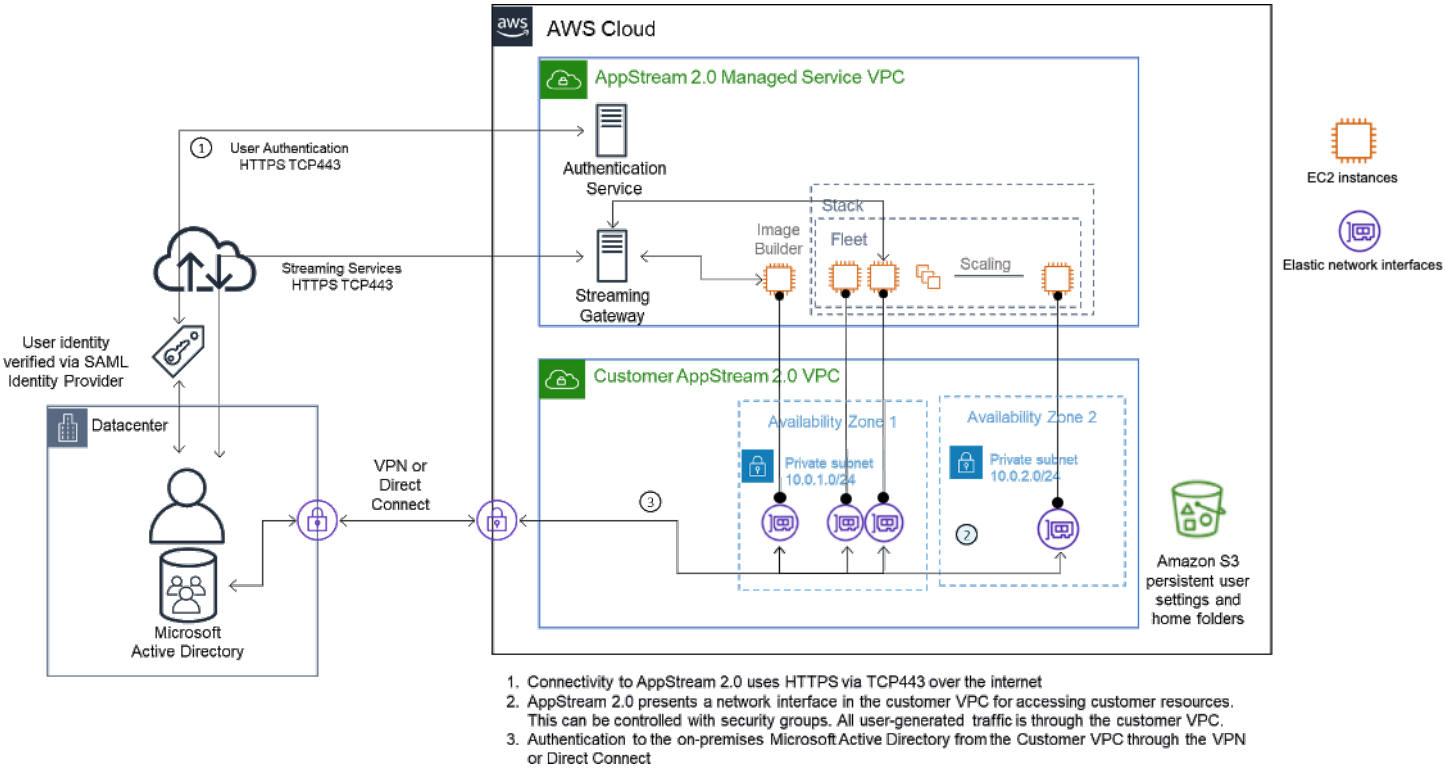
Skenario penerapan berikut yang tercantum adalah opsi integrasi yang umum digunakan dan direkomendasikan untuk AppStream 2.0 dengan Microsoft Managed AD atau Active Directory yang dikelola sendiri oleh pelanggan. Semua diagram arsitektur yang tercantum di bawah ini menggunakan konstruksi inti Amazon.

- Amazon Virtual Private Cloud (VPC) - Pembuatan VPC Amazon yang didedikasikan untuk layanan AppStream 2.0 dengan setidaknya empat subnet pribadi yang tersebar di empat AZ. Dua subnet pribadi digunakan untuk AppStream armada dan Image Builder. Dua subnet yang tersisa digunakan untuk pengontrol domain pada EC2 atau Microsoft Managed AD).

- Set Opsi Dynamic Host Configuration Protocol (DHCP) - Menyediakan standar untuk meneruskan info konfigurasi ke armada AppStream 2.0 dan Pembuat Gambar yang akan disediakan di VPC. Set Opsi DHCP didefinisikan pada tingkat VPC. Hal ini memungkinkan pelanggan untuk menentukan nama domain tertentu dan pengaturan DNS yang akan digunakan dengan AppStream 2.0 instance saat sedang disediakan.
- AWSLayanan Direktori — Amazon Microsoft Managed AD dapat digunakan ke dalam dua subnet pribadi yang akan digunakan bersama dengan AppStream beban kerja 2.0.
- AppStream Armada 2.0 - Armada AppStream 2.0 atau Pembangun Gambar di-host di AWS VPC Terkelola. Setiap instance AppStream 2.0 memiliki dua Elastic Network Interfaces (ENI). Antarmuka utama (`eth0`) digunakan untuk tujuan manajemen dan menengahi koneksi pengguna akhir ke instance melalui gateway streaming. Antarmuka sekunder (`eth1`) disuntikkan ke VPC pelanggan dan dapat digunakan untuk mengakses sumber daya lain di VPC atau lokal yang dipesan lebih dahulu.

Skenario 1: Layanan Domain Direktori Aktif (ADDS) diterapkan di lokasi

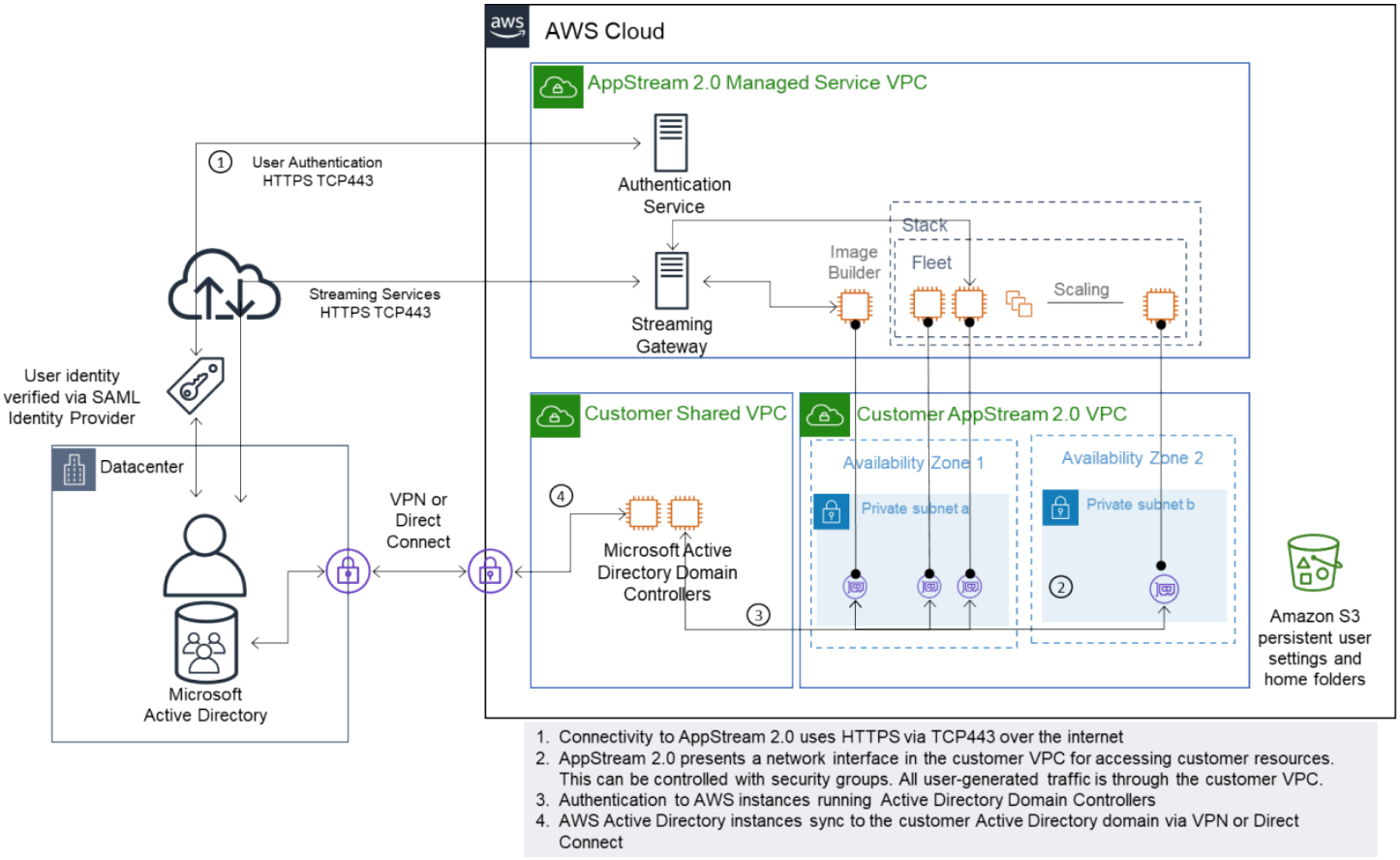
Semua lalu lintas otentikasi melintasi koneksi VPN atau Direct Connect dari VPC pelanggan ke gateway pelanggan. Keuntungan dari skenario ini adalah manfaat menggunakan lingkungan AD yang mungkin sudah diterapkan tanpa harus menyediakan pengontrol domain tambahan di VPC pelanggan. Kerugiannya adalah ketergantungan tunggal pada VPN atau Direct Connect untuk mengautentikasi dan mengotorisasi pengguna untuk armada 2.0. AppStream Jika ada masalah konektivitas jaringan, armada AppStream 2.0 atau Image Builders akan terkena dampak langsung. Menyediakan terowongan VPN ganda atau koneksi Direct Connect dengan jalur yang berbeda mengurangi potensi risiko ini.



Skenario 1 — Layanan Domain Direktori Aktif (ADDS) digunakan di lokasi

Skenario 2: Perluas Layanan Domain Aktif (ADDS) ke AWS VPC pelanggan

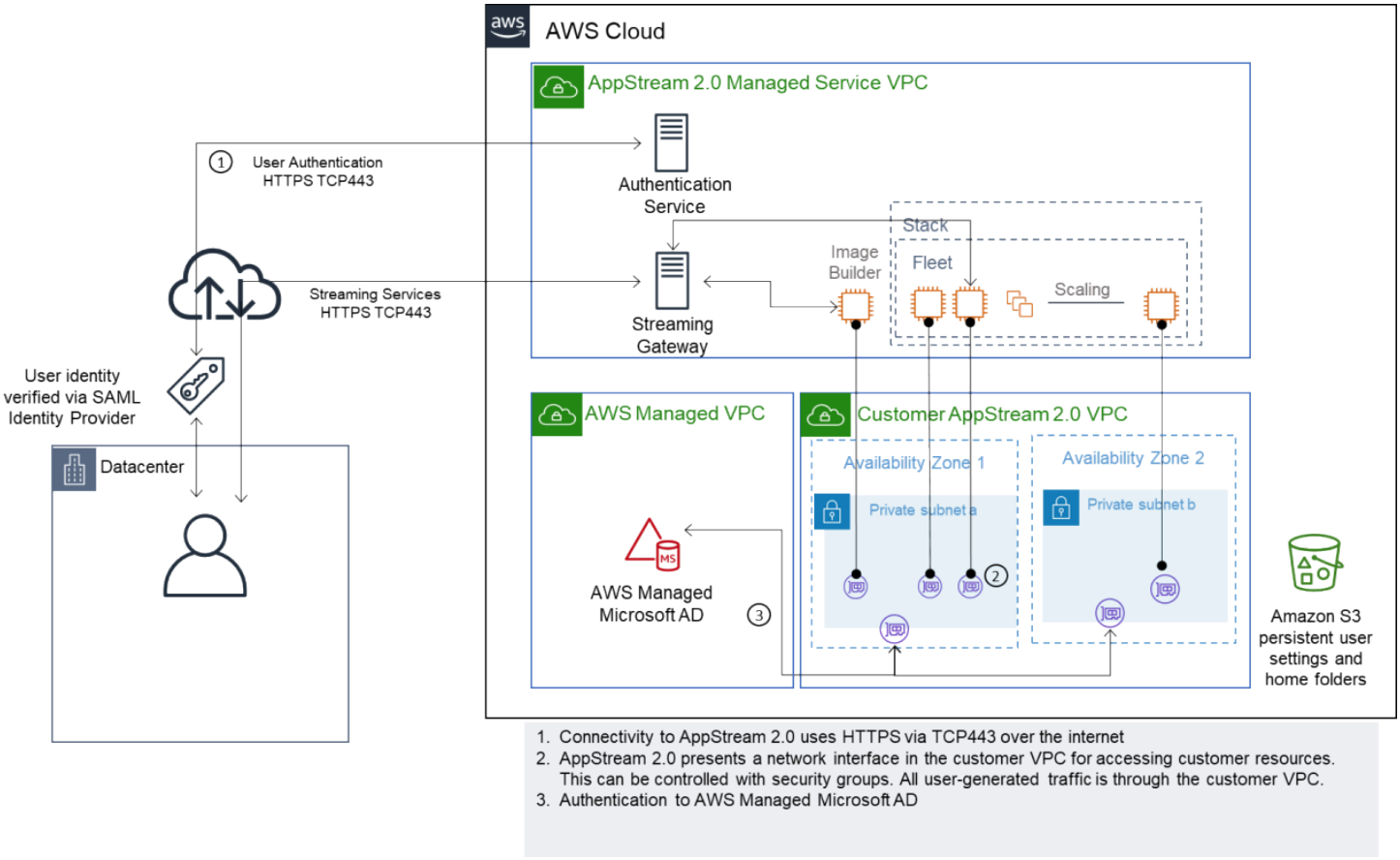
Direktori Aktif diperluas ke VPC pelanggan Anda. Situs Direktori Aktif harus dibuat untuk pengontrol domain baru di VPC pelanggan. Lalu lintas otentikasi diarahkan ke pengontrol domain di AWS VPC pelanggan alih-alih melintasi koneksi VPN atau Direct Connect.



Skenario 2 — Memperluas Layanan Domain Aktif ke Cloud Pribadi Virtual AWS pelanggan

Skenario 3: Direktori Aktif Microsoft yang AWS Dikelola

AWSMicrosoft AD yang dikelola digunakan di AWS Cloud dan digunakan sebagai domain identitas dan sumber daya untuk armada AppStream 2.0 dan Pembuat Gambar.



Skenario 3 - Direktori Aktif AWS Terkelola

Topologi Situs Active Directory Service

Topologi situs layanan Active Directory adalah representasi logis dari jaringan fisik Anda.

Topologi situs membantu Anda secara efisien merutekan kueri klien dan lalu lintas replikasi Direktori Aktif. Topologi situs yang dirancang dan dipelihara dengan baik membantu organisasi Anda mencapai manfaat berikut:

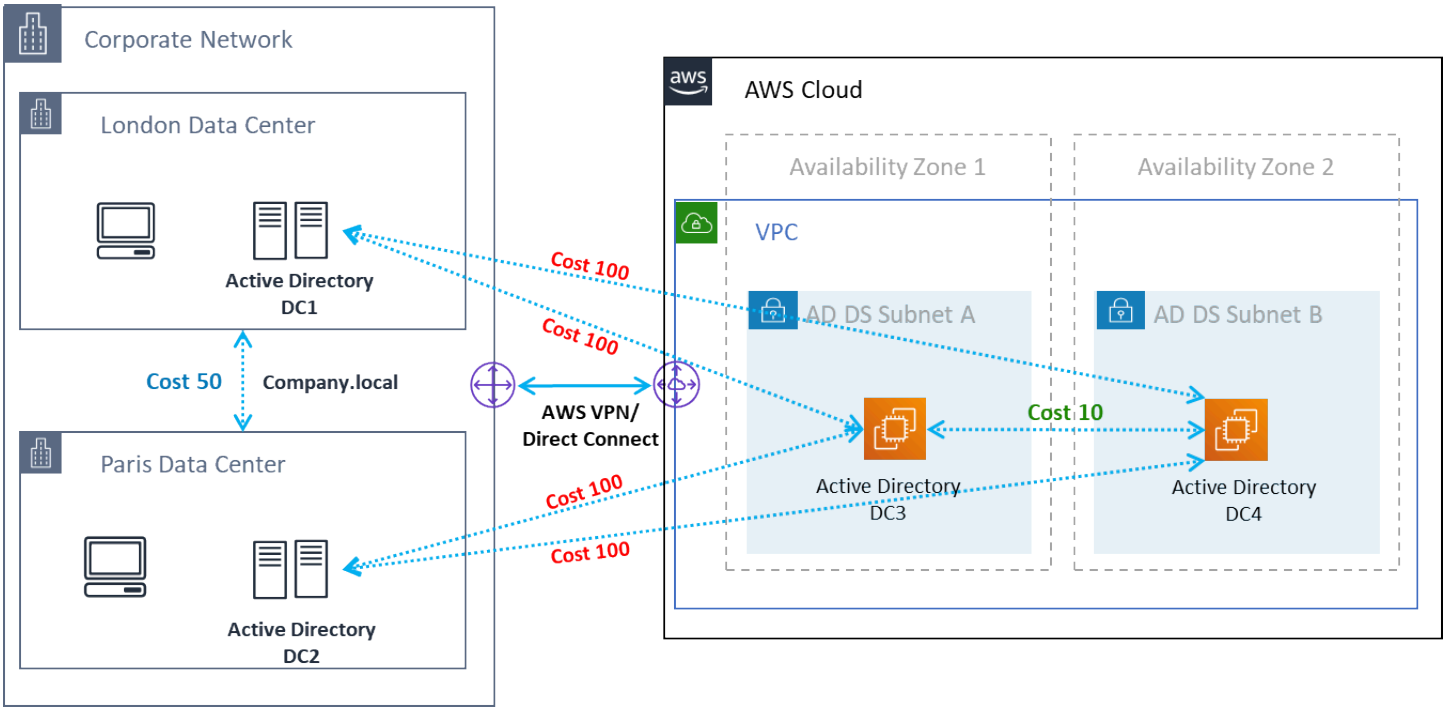
- Minimalkan biaya replikasi data Active Directory saat menyinkronkan antara lokal dan. AWS Cloud
- Optimalkan kemampuan komputer klien untuk menemukan sumber daya terdekat, seperti pengontrol domain. Ini membantu mengurangi lalu lintas jaringan melalui tautan jaringan area luas (WAN) yang lambat, meningkatkan proses logon dan logoff, dan mempercepat operasi akses sumber daya.

Saat memperkenalkan layanan AppStream 2.0, pastikan bahwa rentang alamat yang digunakan untuk subnet instance AppStream 2.0 ditetapkan ke situs yang benar untuk lingkungan Anda.

Untuk Skenario 1 dan Skenario 2, situs dan layanan adalah komponen penting untuk pengalaman pengguna terbaik dalam hal waktu masuk, dan waktu untuk akses sumber daya Direktori Aktif.

Topologi situs mengontrol replikasi Direktori Aktif antara pengontrol domain dalam situs yang sama dan melintasi batas situs.

Mendefinisikan topologi situs yang benar memastikan afinitas klien, yang berarti bahwa klien (dalam hal ini, instance streaming AppStream 2.0) menggunakan pengontrol domain lokal pilihan mereka.



Situs dan layanan Active Directory — afinitas klien

> **Tip**
>
> Sebagai praktik terbaik, tentukan biaya tinggi untuk tautan situs antara AD DS lokal dan AWS Cloud. Angka sebelumnya adalah contoh biaya yang harus Anda tetapkan ke tautan situs (biaya 100) untuk memastikan afinitas klien yang tidak bergantung pada situs.

Untuk informasi lebih lanjut tentang topologi situs, lihat Merancang Topologi Situs.

Unit Organisasi Direktori Aktif

AWS merekomendasikan untuk menyimpan Unit Organisasi (OU) yang dikonfigurasi dalam satu objek Config Direktori AppStream 2.0. Ini adalah praktik terbaik untuk setiap tumpukan AppStream

2.0 untuk memiliki OU sendiri. Ini memungkinkan Anda fleksibilitas untuk memiliki GPO spesifik per tumpukan. Pastikan bahwa OU didedikasikan untuk AppStream 2.0 objek komputer untuk menghindari pencampuran kebijakan AppStream spesifik 2.0 dengan desktop lokal. Pertimbangkan untuk menggunakan Sub-OUS untuk setiap yang Wilayah AWS Anda gunakan AppStream 2.0.

Pembersihan objek komputer Active Directory

AppStream 2.0 instance bersifat fana. Armada membuat dan menggunakan kembali objek komputer Active Directory saat armada skala dan skala masuk.

AWSmerekomendasikan untuk membuat proses pembersihan AD untuk menghapus objek komputer Active Directory basi yang dapat ada setelah AppStream armada dihapus.

Keamanan

Keamanan cloud di Amazon Web Services (AWS) merupakan prioritas tertinggi. Keamanan dan kepatuhan adalah tanggung jawab bersama antara AWS dan pelanggan. Untuk informasi lebih lanjut, lihat Model Tanggung Jawab Bersama. Sebagai pelanggan AWS dan AppStream 2.0, penting untuk menerapkan langkah-langkah keamanan pada lapisan yang berbeda seperti tumpukan, armada, gambar, dan jaringan.

Karena sifatnya yang fana, AppStream 2.0 sering lebih disukai sebagai solusi aman untuk aplikasi dan pengiriman desktop. Pertimbangkan apakah solusi antivirus yang biasa digunakan dalam penerapan Windows relevan dalam kasus penggunaan Anda untuk lingkungan yang telah ditentukan sebelumnya dan dibersihkan di akhir sesi pengguna. Antivirus menambahkan overhead ke instance virtual, menjadikannya praktik terbaik untuk mengurangi aktivitas yang tidak perlu. Misalnya, memindai volume sistem (yang bersifat sementara) saat boot, misalnya, tidak menambah keamanan keseluruhan 2.0. AppStream

Dua pertanyaan kunci untuk keamanan AppStream 2.0 berpusat pada:

- Apakah status pengguna yang bertahan di luar sesi merupakan persyaratan?
- Berapa banyak akses yang harus dimiliki pengguna dalam satu sesi?

Mengamankan data persisten

Penerapan AppStream 2.0 dapat mengharuskan status pengguna untuk bertahan dalam beberapa bentuk. Mungkin untuk mempertahankan data untuk pengguna individu, atau untuk mempertahankan data untuk kolaborasi menggunakan folder bersama. AppStreamPenyimpanan instance 2.0 bersifat sementara dan tidak memiliki opsi enkripsi.

AppStream 2.0 menyediakan persistensi status pengguna melalui folder rumah dan pengaturan aplikasi di Amazon S3. Beberapa kasus penggunaan memerlukan kontrol yang lebih besar atas persistensi status pengguna. Untuk kasus penggunaan ini, AWS rekomendasikan untuk menggunakan berbagi file Server Message Block (SMB).

Status pengguna dan data

Karena sebagian besar aplikasi Windows berkinerja terbaik dan paling aman ketika ditempatkan bersama dengan data aplikasi yang dibuat oleh pengguna, ini adalah praktik terbaik untuk menyimpan data ini Wilayah AWS sama dengan armada AppStream 2.0. Mengenkripsi data ini

adalah praktik terbaik. Perilaku default folder home pengguna adalah mengenkripsi file dan folder saat istirahat menggunakan kunci enkripsi terkelola Amazon S3 dari layanan manajemen AWS kunci ().AWS KMS Penting untuk dicatat bahwa Pengguna AWS Administratif dengan akses ke bucket AWS Console atau Amazon S3 akan dapat mengakses file-file tersebut secara langsung.

Dalam desain yang memerlukan target Server Message Block (SMB) dari Windows File Share untuk menyimpan file dan folder pengguna, prosesnya otomatis atau memerlukan konfigurasi.

Tabel 5 — Opsi untuk mengamankan data pengguna

SMBtarget	E ncryption-at-rest	E ncryption-in-transit	Antivirus (AV)
FSxuntuk Windows File Server	Otomatis melalui AWS KMS	Otomatis melalui SMB enkripsi	AV yang diinstal pada instance jarak jauh melakukan pemindaian pada drive yang dipetakan
Gateway File, AWS Storage Gateway	Secara default, semua data yang disimpan oleh AWS Storage Gateway dalam S3 dienkripsi sisi server dengan Amazon S3-Managed Encryption Keys (-S3). SSE Anda dapat secara opsional mengonfigurasi berbagai jenis gateway untuk mengenkripsi data yang disimpan dengan AWS Key Management Service () KMS	Semua data yang ditransfer antara semua jenis alat gateway dan AWS penyimpanan dienkripsi menggunakan. SSL	AV yang diinstal pada instance jarak jauh melakukan pemindaian pada drive yang dipetakan
EC2Server Berkas Windows berbasis	Aktifkan EBS enkripsi	PowerShell; `Set-SmbServer`	AV yang diinstal pada server melakukan

SMBtarget	E ncryption-at-rest	E ncryption-in-transit	Antivirus (AV)
		`Configuration - EncryptData $True`	pemindaian pada drive lokal

Keamanan titik akhir dan antivirus

Sifat singkat instans Amazon AppStream 2.0 dan kurangnya persistensi data berarti diperlukan pendekatan yang berbeda untuk memastikan pengalaman dan kinerja pengguna tidak terganggu oleh aktivitas yang akan diperlukan pada desktop persisten. Agen Endpoint Security diinstal dalam gambar AppStream 2.0 ketika ada kebijakan organisasi atau ketika digunakan dengan masuknya data eksternal misalnya email, masuknya file, penjelajahan web eksternal.

Menghapus pengidentifikasi unik

Agen Endpoint Security mungkin memiliki identifier (GUID) yang unik secara global yang harus disetel ulang selama proses pembuatan instance armada. Vendor memiliki instruksi untuk menginstal produk mereka dalam gambar yang akan memastikan yang baru GUID dihasilkan untuk setiap instance yang dihasilkan dari gambar.

Untuk memastikan tidak GUID dihasilkan, instal agen Endpoint Security sebagai tindakan terakhir sebelum menjalankan Asisten AppStream 2.0 untuk menghasilkan gambar.

Optimalisasi kinerja

Vendor Keamanan Endpoint menyediakan sakelar dan pengaturan yang mengoptimalkan kinerja 2.0. AppStream Pengaturan bervariasi antara vendor dan dapat ditemukan dalam dokumentasi mereka, biasanya di bagian di. VDI Beberapa pengaturan umum termasuk tetapi tidak terbatas pada adalah:

- Matikan pemindaian boot up untuk memastikan pembuatan instans, waktu startup, dan login diminimalkan
- Matikan pemindaian terjadwal untuk mencegah pemindaian yang tidak perlu
- Matikan cache tanda tangan untuk mencegah pencacahan file
- Aktifkan pengaturan IO yang VDI dioptimalkan
- Pengecualian yang diperlukan oleh aplikasi untuk memastikan kinerja

Vendor keamanan endpoint memberikan instruksi untuk digunakan dengan lingkungan desktop virtual yang mengoptimalkan kinerja.

- Dukungan Pemindaian Kantor Trend Micro untuk Infrastruktur Desktop Virtual - Apex One/ OfficeScan (trendmicro.com)
- CrowdStrike dan Cara Memasang CrowdStrike Falcon di Pusat Data
- Sophos dan Sophos Central Endpoint: Cara menginstal pada gambar emas untuk menghindari identitas duplikat dan Sophos Central: Praktik terbaik saat menginstal Titik Akhir Windows di Lingkungan Desktop Virtual
- McAfee dan Penyediaan dan penyebaran McAfee Agen pada sistem Infrastruktur Desktop Virtual
- Microsoft Endpoint Security dan Mengkonfigurasi Microsoft Defender Antivirus untuk mesin yang tidak persisten VDI - Microsoft Tech Community

Pengecualian pemindaian

Jika perangkat lunak keamanan diinstal dalam kasus AppStream 2.0, perangkat lunak keamanan tidak boleh mengganggu proses berikut.

Tabel 6 — AppStream 2.0 memproses perangkat lunak keamanan tidak boleh mengganggu proses berikut.

Layanan	Proses
AmazonCloudWatchAgent	“C:\Program File\ Amazon\AmazonCloud WatchAgent\ mulai-amazon- cloudwatch-agent.exe”
A mazonSSMAgent	“C:\Program File\ Amazon\SSM\ amazon-ssm-agent .exe”
NICE DCV	“C:\Program FilesNICE\DCV\\ Server\ bin\ dcvserver.exe” "C:\Program File\NICE\DCV\ Server\ bin\ dcvagent.exe”
AppStream 2.0	“C:\ProgramFiles\ Amazon\ AppStream 2\StorageConnector\ StorageConnector .exe” Di folder "C:\Program Files\ Amazon\ Photon\”

Layanan	Proses
	“. \ Agen\ PhotonAgent .exe” “. \ Agen\ s5cmd.exe” “. \WebServer\ PhotonAgentWebServer .exe” “. \CustomShell\ PhotonWindowsAppSwitcher .exe” “. \CustomShell\ PhotonWindowsCustomShell .exe” “. \CustomShell\ PhotonWindowsCustomShellBackground .exe”

Folder

Jika perangkat lunak keamanan diinstal dalam kasus AppStream 2.0, perangkat lunak tidak boleh mengganggu folder berikut:

Example

```
C:\Program Files\Amazon\*
C:\ProgramData\Amazon\*
C:\Program Files (x86)\AWS Tools\*
C:\Program Files (x86)\AWS SDK for .NET\*
C:\Program Files\NICE\*
C:\ProgramData\NICE\*
C:\AppStream\*
C:\Program Files\Internet Explorer\*
C:\Program Files\nodejs\
```

Kebersihan konsol keamanan titik akhir

Amazon AppStream 2.0 akan membuat instance unik baru setiap kali pengguna terhubung di luar batas waktu idle dan memutuskan sambungan. Instans akan memiliki nama yang unik dan akan dibangun di kondoles manajemen keamanan endpoint. Menyetel mesin tua yang tidak digunakan di atas 4 hari atau lebih (atau lebih rendah tergantung pada batas waktu sesi AppStream 2.0) untuk dihapus akan meminimalkan jumlah instance kedaluwarsa di konsol.

Pengecualian jaringan

Rentang jaringan manajemen AppStream 2.0 (`198.19.0.0/16`) dan port dan alamat berikut tidak boleh diblokir oleh solusi keamanan/firewall atau antivirus apa pun dalam AppStream 2.0 instance.

Tabel 7 - Port dalam instance streaming AppStream 2.0 perangkat lunak keamanan tidak boleh mengganggu

Port	Penggunaan
8300, 3128	Ini digunakan untuk membuat koneksi streaming
8000	Ini digunakan untuk mengelola instance streaming dengan AppStream 2.0
8443	Ini digunakan untuk mengelola instance streaming dengan AppStream 2.0
53	DNS

Tabel 8 — AppStream 2.0 alamat layanan terkelola perangkat lunak keamanan tidak boleh mengganggu

Port	Penggunaan
169.254.169.123	NTP
169.254.169.249	NVIDIAGRIDLayanan Lisensi

Port	Penggunaan
169.254.169.250	KMS
169.254.169.251	KMS
169.254.169.253	DNS
169.254.169.254	Metadata

Mengamankan sesi AppStream

Membatasi kontrol aplikasi dan sistem operasi

AppStream 2.0 memberi administrator kemampuan untuk menentukan dengan tepat aplikasi mana yang dapat diluncurkan dari halaman web dalam mode streaming aplikasi. Namun, ini tidak menjamin bahwa hanya aplikasi yang ditentukan yang dapat dijalankan.

Utilitas dan aplikasi Windows dapat diluncurkan melalui sistem operasi melalui cara tambahan. AWS merekomendasikan penggunaan Microsoft AppLocker untuk memastikan bahwa hanya aplikasi yang dibutuhkan organisasi Anda yang dapat dijalankan. Aturan default harus dimodifikasi, karena memberikan akses jalur kepada semua orang ke direktori sistem kritis.

> **Note**
>
> Windows Server 2016 dan 2019 mengharuskan layanan Identitas Aplikasi Windows berjalan untuk menegakkan AppLocker aturan. Akses aplikasi dari AppStream 2.0 menggunakan Microsoft AppLocker dirinci dalam Panduan AppStream Admin.

Untuk instance armada yang bergabung dengan domain Active Directory, gunakan Group Policy Objects (GPOs) untuk memberikan setelan pengguna dan sistem guna mengamankan akses aplikasi dan sumber daya pengguna.

Firewall dan routing

Saat membuat armada AppStream 2.0, subnet dan Grup Keamanan harus ditetapkan. Subnet memiliki penugasan Network Access Control Lists (NACLs) dan tabel rute yang ada. Anda dapat

mengaitkan hingga lima grup keamanan saat meluncurkan pembuat gambar baru atau saat membuat armada baru Grup Keamanan dapat memiliki hingga lima tugas dari Grup Keamanan yang ada. Untuk setiap grup keamanan, Anda menambahkan aturan yang mengontrol lalu lintas jaringan keluar dan masuk dari dan ke instans Anda

A NACL adalah lapisan keamanan opsional untuk Anda VPC yang bertindak sebagai firewall stateless untuk mengendalikan lalu lintas masuk dan keluar dari satu atau lebih subnet. Anda dapat mengatur jaringan ACLs dengan aturan yang mirip dengan grup keamanan Anda untuk menambahkan lapisan keamanan tambahan ke AndaVPC. Untuk informasi selengkapnya tentang perbedaan antara grup keamanan dan jaringanACLs, lihat membandingkan grup dan NACLs halaman keamanan.

Saat merancang dan menerapkan Grup Keamanan dan NACL aturan, pertimbangkan praktik terbaik yang Dirancang AWS dengan Baik untuk mendapatkan hak istimewa paling sedikit. Keistimewaan paling sedikit adalah prinsip pemberian hanya izin yang diperlukan untuk menyelesaikan tugas.

Untuk pelanggan yang memiliki jaringan pribadi berkecepatan tinggi yang menghubungkan lingkungan di lokasi mereka ke AWS (melalui AWS Direct Connect), Anda dapat mempertimbangkan untuk menggunakan VPC Endpoint for AppStream, yang berarti lalu lintas streaming akan diarahkan melalui konektivitas jaringan pribadi Anda daripada melalui internet publik. Untuk informasi lebih lanjut tentang topik ini, lihat bagian VPC titik akhir antarmuka streaming AppStream 2.0 dari dokumen ini.

Pencegahan kehilangan data

Kita akan melihat dua jenis pencegahan kehilangan data.

Klien ke AppStream 2.0 Kontrol Transfer Data Instance

Tabel 9 — Panduan untuk mengontrol masuknya data dan keluar

Pengaturan	Pilihan	Bimbingan
Papan klip	• Salin dan tempel hanya ke sesi jarak jauh • Salin ke perangkat lokal saja • Nonaktif	Menonaktifkan pengatura n ini tidak menonaktifkan salin dan tempel dalam sesi. Jika menyalin data ke sesi diperlukan, pilih Tempel

Pengaturan	Pilihan	Bimbingan
		ke sesi jarak jauh hanya untuk meminimalkan potensi kebocoran data.
Transfer file	• Unggah dan unduh • Unggah saja • Unduh saja • Nonaktif	Hindari mengaktifkan pengaturan ini untuk mencegah kebocoran data.
Mencetak ke perangkat lokal	• Diaktifkan • Dinonaktifkan	Jika pencetakan diperluka n, gunakan printer yang dipetakan jaringan yang dikontrol dan dipantau oleh organisasi Anda.

Pertimbangkan keuntungan dari solusi transfer data organisasi yang ada dibandingkan pengaturan tumpukan. Konfigurasi ini tidak dirancang untuk menggantikan solusi transfer data aman yang komprehensif.

Mengontrol lalu lintas keluar dari instance 2.0 AppStream

Di mana kehilangan data menjadi perhatian, penting untuk menutupi apa yang dapat diakses Pengguna begitu mereka berada di dalam instance AppStream 2.0 mereka. Seperti apa jalur keluar (atau jalan keluar) jaringan? Merupakan persyaratan umum untuk memiliki akses internet publik yang tersedia untuk pengguna akhir di dalam instance AppStream 2.0 mereka, sehingga menempatkan Solusi Penyaringan Konten WebProxy atau di jalur jaringan perlu dipertimbangkan. Pertimbangan lain termasuk aplikasi Antivirus lokal dan langkah-langkah keamanan endpoint lainnya di dalam AppStream instance (lihat bagian “Endpoint Security and Antivirus” untuk informasi lebih lanjut).

Menggunakan AWS layanan

AWS Identity and Access Management

Menggunakan IAM peran untuk mengakses AWS layanan, dan spesifik dalam IAM kebijakan yang melekat padanya, adalah praktik terbaik yang hanya menyediakan pengguna dalam sesi

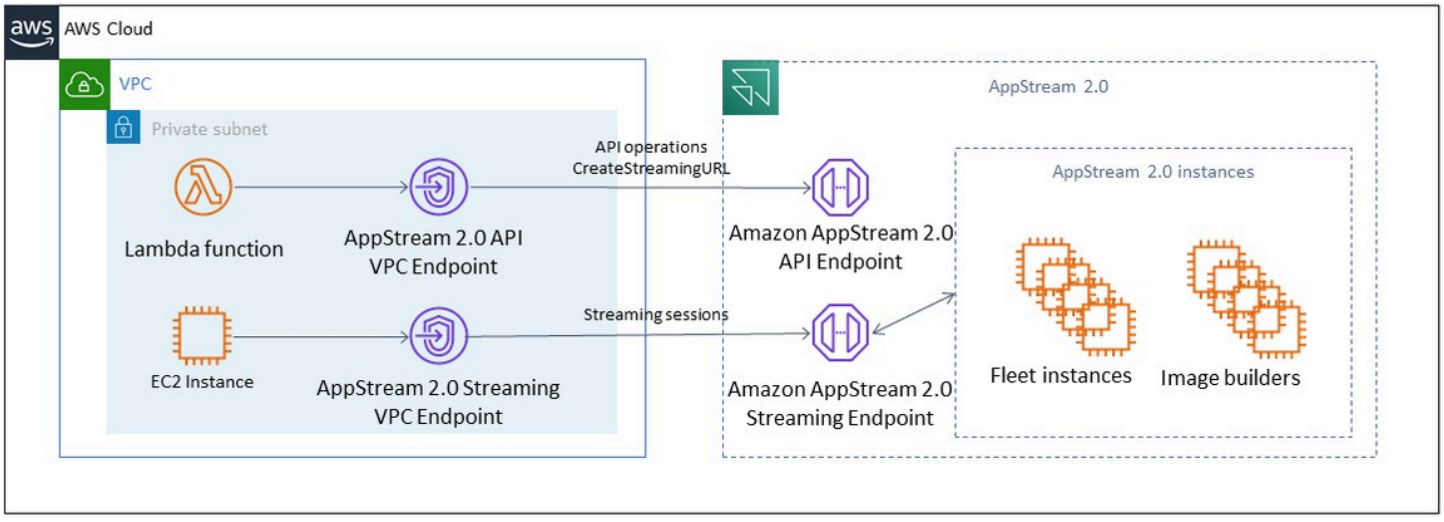
AppStream 2.0 yang memiliki akses tanpa mengelola kredensil tambahan. Ikuti praktik terbaik untuk menggunakan IAM Peran dengan AppStream 2.0.

Buat IAMkebijakan untuk melindungi bucket Amazon S3 yang dibuat untuk mempertahankan data pengguna di folder beranda dan persistensi pengaturan aplikasi. Ini mencegah administrator AppStream non-2.0 dari akses.

VPCtitik akhir

VPCTitik akhir memungkinkan koneksi pribadi antara layanan Anda VPC dan layanan yang didukung serta AWS layanan VPC titik akhir yang didukung oleh. AWS PrivateLink AWS PrivateLink adalah teknologi yang memungkinkan Anda mengakses layanan secara pribadi dengan menggunakan alamat IP pribadi. Lalu lintas antara Anda VPC dan layanan lainnya tidak meninggalkan jaringan Amazon. Jika akses internet publik hanya diperlukan untuk AWS layanan, VPC titik akhir menghapus persyaratan untuk gateway dan NAT gateway internet sama sekali.

Di lingkungan di mana rutinitas otomatisasi atau pengembang memerlukan API panggilan untuk AppStream 2.0, buat VPC titik akhir antarmuka untuk operasi AppStream 2.0 API. Misalnya, jika ada EC2 instance di subnet pribadi tanpa akses internet publik, VPC titik akhir untuk AppStream 2.0 API dapat digunakan untuk memanggil API operasi AppStream 2.0 seperti. CreateStreamingURL Diagram berikut menunjukkan contoh pengaturan di mana AppStream 2.0 API dan VPC titik akhir streaming dikonsumsi oleh fungsi EC2 dan instance Lambda.



VPCtitik akhir

VPCTitik akhir streaming memungkinkan Anda melakukan streaming sesi melalui titik VPC akhir. Titik akhir antarmuka streaming mempertahankan lalu lintas streaming di dalam AndaVPC. Lalu lintas streaming meliputi pikselUSB, input pengguna, audio, clipboard, unggahan dan unduhan file, dan

lalu lintas printer. Untuk menggunakan VPC titik akhir, pengaturan VPC titik akhir harus diaktifkan pada tumpukan AppStream 2.0. Ini berfungsi sebagai alternatif untuk streaming sesi pengguna melalui internet publik dari lokasi yang memiliki akses internet terbatas dan akan mendapat manfaat dari mengakses melalui instance Direct Connect. Streaming sesi pengguna melalui VPC titik akhir memerlukan hal berikut:

- Grup Keamanan yang terkait dengan titik akhir antarmuka harus mengizinkan akses masuk ke port `443` (TCP) dan port `1400-1499` (TCP) dari rentang alamat IP tempat pengguna Anda terhubung.
- Daftar Kontrol Akses Jaringan untuk subnet harus mengizinkan lalu lintas keluar dari port jaringan sementara `1024-65535` (TCP) ke rentang alamat IP dari mana pengguna Anda terhubung.
- Konektivitas internet diperlukan untuk mengautentikasi pengguna dan mengirimkan aset web yang dibutuhkan AppStream 2.0 agar berfungsi.

Untuk mempelajari lebih lanjut tentang membatasi lalu lintas ke AWS layanan dengan AppStream 2.0, lihat panduan administrasi untuk membuat dan streaming dari titik VPC akhir.

Ketika akses internet publik penuh diperlukan, itu adalah praktik terbaik untuk menonaktifkan Internet Explorer Enhanced Security Configuration (ESC) pada Image Builder. Untuk informasi selengkapnya, lihat panduan administrasi AppStream 2.0 untuk menonaktifkan konfigurasi keamanan Internet Explorer yang disempurnakan.

Pemulihan bencana

Amazon AppStream 2.0 telah membangun redundansi di hingga tiga zona ketersediaan. Ini berarti bahwa jika pengguna memiliki sesi aktif di zona ketersediaan yang menjadi terdegradasi, mereka dapat dengan mudah memutuskan dan menyambung kembali yang akan memesan sesi mereka di zona ketersediaan yang sehat dengan asumsi Anda memiliki kapasitas. Meskipun ini memberikan ketersediaan tinggi di Wilayah, itu tidak memberikan solusi pemulihan bencana jika layanan mengalami masalah di tingkat regional.

Untuk menyediakan rencana pemulihan bencana bagi pengguna AppStream 2.0 Anda, Anda harus terlebih dahulu membangun lingkungan AppStream 2.0 di Wilayah sekunder Anda. Dari perspektif desain, lingkungan ini harus memiliki koneksi berlebihan ke lingkungan lokal Anda, jika berlaku, dan seharusnya tidak memiliki ketergantungan pada Wilayah utama. Misalnya, jika armada AppStream 2.0 Anda bergabung dengan domain, Anda harus memiliki pengontrol domain tambahan di Wilayah sekunder dengan Situs dan Layanan yang dikonfigurasi. Dari perspektif AppStream 2.0, lingkungan ini harus terdiri dari pengaturan armada dan tumpukan yang sama dengan yang Anda miliki di Wilayah utama Anda. Armada itu sendiri harus menjalankan gambar dasar Anda yang sama, yang dapat disalin ke Wilayah sekunder Anda melalui konsol atau secara terprogram. Jika aplikasi yang berjalan dalam sesi AppStream 2.0 Anda memiliki ketergantungan backend yang terkait dengan Wilayah utama Anda, itu juga harus memiliki redundansi regional untuk memastikan pengguna masih dapat mengakses backend aplikasi jika Wilayah utama turun. Batas tingkat layanan Anda di Wilayah tujuan Anda harus sesuai dengan Wilayah utama Anda.

Perutean identitas

Ada dua metode berbeda untuk menyediakan akses ke aplikasi dalam skenario DR. Pada tingkat tinggi, kedua metode berbeda dengan bagaimana pengguna diarahkan ke Wilayah failover. Metode pertama dilakukan dengan konfigurasi aplikasi AppStream 2.0 tunggal di iDP Anda dan metode kedua memiliki dua konfigurasi aplikasi terpisah.

Metode 1: Mengubah status relai aplikasi Anda

Saat pengguna masuk ke AppStream 2.0 dari Penyedia Identitas (iDP), setelah otentikasi mereka, mereka diteruskan ke URL tertentu yang sejajar dengan Wilayah dan tumpukan yang dimaksudkan untuk diakses. Untuk informasi selengkapnya seputar URL Status Relay, lihat Panduan Administrasi Amazon AppStream 2.0. Administrator dapat mengonfigurasi tumpukan Lintas wilayah yang

dibangun pada gambar AppStream 2.0 yang sama dengan Wilayah utama bagi pengguna untuk failover. Administrator dapat mengontrol failover ini hanya dengan memperbarui URL Relay State untuk menunjuk ke tumpukan failover. Agar metode ini dapat beroperasi dengan benar, kebijakan IAM terkait perlu mencerminkan akses ke kedua tumpukan; primer dan failover. Untuk detail selengkapnya tentang bagaimana kebijakan IAM ini harus dikonfigurasi, lihat contoh kebijakan berikut.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "appstream:Stream",
            "Resource": [
            "arn:aws:appstream:PrimaryRegion:190836837966:stack/StackName",
            "arn:aws:appstream:FailoverRegion:190836837966:stack/StackName"
            ],
            "Condition": {
                "StringEquals": {
                    "appstream:userId": "${saml:sub}"
                }
            }
        }
    ]
}
```

Metode 2: Mengkonfigurasi dua aplikasi AppStream 2.0 dalam IDP Anda

Metode ini mengharuskan administrator untuk membangun dua aplikasi terpisah untuk AppStream 2.0 dalam iDP. Mereka kemudian dapat menyajikan kedua aplikasi dan membiarkan pengguna memilih ke mana harus pergi, atau mereka mengunci/menyembunyikan aplikasi sampai tiba waktunya untuk failover. Metode ini lebih selaras dengan kasus penggunaan memiliki pengguna global yang sering berpindah-pindah. Pengguna tersebut harus melakukan streaming dari titik akhir terdekat, oleh karena itu memiliki kedua aplikasi yang ditetapkan memberi mereka opsi untuk memilih aplikasi yang dikonfigurasi untuk Wilayah terdekat mereka. Ini juga dapat diotomatisasi, untuk informasi lebih lanjut lihat posting blog ini.

Ketekunan penyimpanan

Saat memanfaatkan fitur persistensi data AppStream 2.0 yang disertakan, seperti Persistensi Aplikasi dan Sinkronisasi Folder Rumah, Anda perlu mereplikasi data tersebut ke wilayah failover Anda. Fitur-fitur ini menyimpan data persisten dalam bucket Amazon S3 di wilayah AppStream 2.0 yang diberikan. Agar data tetap ada lintas wilayah, Anda harus mereplikasi semua perubahan pada bucket sumber ke bucket failover region AppStream 2.0. Ini dapat dilakukan dengan fitur Amazon S3 asli, seperti replikasi lintas wilayah Amazon S3. Setiap pengguna data persisten akan berada di bawah folder nama pengguna yang di-hash mereka. Karena nama pengguna akan di-hash lintas wilayah yang sama, cukup mereplikasi data akan memberikan persistensi data di wilayah sekunder Anda. Untuk informasi lebih lanjut tentang bucket Amazon S3 yang digunakan oleh AppStream 2.0, lihat panduan ini.

Pemantauan

Menggunakan dasbor

Pemantauan pemanfaatan armada adalah kegiatan rutin yang dapat dilakukan melalui CloudWatch metrik dan membuat dasbor. Atau, dari konsol AppStream 2.0, gunakan tab Penggunaan Armada. Pantau penggunaan armada Anda secara teratur, karena perilaku pengguna tidak selalu dapat diprediksi, dan permintaan dapat melebihi perencanaan awal tingkat pertama sekalipun. Daftar lengkap AppStream 2.0 metrik dan dimensi untuk CloudWatch dapat ditemukan di panduan administrasi AppStream 2.0 di bawah Sumber Daya Pemantauan.

Mengantisipasi pertumbuhan

Setiap kali ada lompatan besar`PendingCapacity`, peristiwa penskalaan otomatis telah terjadi. Penting untuk mengonfirmasi hal itu `AvailableCapacity` dan `PendingCapacity` memiliki hubungan terbalik sementara instance armada AppStream 2.0 baru tersedia untuk meng-host sesi pengguna. Buat CloudWatch Alarm `InsufficientCapacityError` untuk setiap armada AppStream 2.0 untuk memberi tahu administrator guna memastikan penskalaan otomatis tidak ketinggalan permintaan.

Jika permintaan melebihi kapasitas dan nilai `InsufficientCapacityError` metrik adalah hal yang umum, pertimbangkan untuk meningkatkan kapasitas minimum melalui kebijakan Penskalaan Terjadwal untuk awal hari kerja. Selain itu, miliki kebijakan Penskalaan Terjadwal kedua untuk menurunkan kapasitas minimum setelah permintaan terpenuhi. Perlu diingat bahwa menurunkan nilai untuk kapasitas minimum tidak memengaruhi sesi yang ada. Menurunkan kapasitas minimum sebelum akhir hari kerja secara efektif memungkinkan skala berfungsi sebagaimana dimaksud dengan menurunkan nilai untuk`ActualCapacity`. Ini mengoptimalkan biaya.

Jika permintaan secara konsisten tidak dapat diprediksi, gunakan kebijakan penskalaan Target Tracking untuk memastikan bahwa ada cukup `AvailableCapacity` dalam armada AppStream 2.0 untuk memenuhi permintaan sambil menentukan pola penggunaan. Terus memantau karena Pelacakan Target menggunakan persentase konsumsi armada. Ketika jumlah total instance armada bertambah, jumlah total instance armada yang tidak digunakan berlipat ganda. Ini bisa menjadi boros kecuali kapasitas maksimum diatur ke nilai konservatif. Gunakan beberapa jenis kebijakan penskalaan (misalnya, Pelacakan Terjadwal dan Target) untuk menyeimbangkan keandalan dengan pengoptimalan biaya.

Memantau penggunaan pengguna

Memantau pengguna unik, karena ada biaya yang terkait untuk itu dalam bentuk biaya pengguna. Biaya biaya pengguna ini disebabkan oleh lisensi akses pelanggan (SAL) Image Assistant (RDS). Mengevaluasi pengguna unik dapat dilakukan melalui pelaporan dari IDP tempat otentikasi dilakukan, atau melalui laporan penggunaan.

Laporan penggunaan disimpan sebagai `.csv` file terpisah di bucket S3 Anda, yang dapat Anda unduh dan analisis menggunakan alat intelijen bisnis (BI) pihak ketiga. Anda dapat menganalisis data penggunaan AWS tanpa mengunduh laporan atau membuat laporan melalui rentang tanggal khusus tanpa menggabungkan beberapa file. `.csv` Misalnya, Anda dapat menggunakan Amazon Athena dan Amazon QuickSight untuk membuat laporan kustom dan visualisasi data penggunaan 2.0 Anda AppStream .

Aplikasi yang bertahan dan log peristiwa Windows

Ketika sesi instance AppStream 2.0 selesai, instance berakhir. Ini berarti semua log peristiwa aplikasi dan Windows yang digunakan dalam sesi hilang. Jika ada persyaratan untuk mempertahankan aplikasi ini dan log peristiwa Windows, salah satu metode adalah dengan menggunakan Amazon Data Firehose untuk mengirimkannya secara real-time ke S3 dan mencari dengan OpenSearch Amazon Service (ServiceOpenSearch). Jika kueri tidak diantisipasi sering, untuk mengoptimalkan biaya, gunakan Amazon Athena untuk mencari sebagai lawan menjalankan Layanan Amazon. OpenSearch

Jaringan audit dan aktivitas administrasi

Jika belum diatur, ini adalah praktik terbaik AWS CloudTrailuntuk mengonfigurasi Akun AWS dengan Amazon AppStream 2.0. Untuk mengaudit panggilan API AppStream 2.0 secara khusus, gunakan sumber peristiwa filter dengan nilai`appstream.amazonaws.com`.

Aktifkan log aliran VPC untuk mengaudit akses ke sumber daya yang dikelola pelanggan. Log aliran VPC dapat dipublikasikan ke CloudWatch Log untuk melakukan kueri saat audit diperlukan.

Pemantauan alokasi IP subnet penting karena armada AppStream 2.0 tumbuh. Laporkan penugasan IP dengan menjalankan CLI subnet deskripsikan untuk melaporkan alamat IP yang tersedia di setiap subnet yang ditetapkan ke armada. Pastikan bahwa organisasi Anda memiliki kapasitas alamat IP yang cukup untuk memenuhi permintaan semua armada yang berjalan pada kapasitas maksimum.

Optimalisasi biaya

Optimalisasi biaya berfokus pada menghindari biaya yang tidak dibutuhkan. Topik utama termasuk memahami dan mengendalikan di mana uang dihabiskan, dan memilih jumlah jenis sumber daya yang paling tepat dan benar. Analisis pengeluaran dari waktu ke waktu dan penskalaan untuk memenuhi kebutuhan bisnis. Sumber daya AppStream 2.0 berikut dikenakan pay-as-you-go biaya:

- Instans armada yang Selalu Aktif
- Instans armada On-Demand
- On-Demand menghentikan biaya instans
- Instans image builder
- Biaya pengguna

Untuk informasi harga saat ini, lihat situs AWS web untuk harga Amazon AppStream 2.0.

Merancang penerapan AppStream 2.0 yang hemat biaya

Langkah pertama dalam perencanaan dan desain penerapan AppStream 2.0 adalah menggunakan alat penetapan harga sederhana untuk memperkirakan dasar AWS biaya Anda terkait dengan penggunaan Anda. Berikan jumlah total pengguna Anda, penggunaan bersamaan aktual per jam, jenis instans, dan pemanfaatan armada, dan alat penetapan harga memperkirakan harga per pengguna Anda. Ini juga menunjukkan perkiraan penghematan harga saat Anda menggunakan armada On-Demand alih-alih armada Always-On.

Pelanggan menyukai model harga AppStream 2.0 yang hanya membayar untuk contoh yang mereka sediakan untuk memenuhi kebutuhan streaming pengguna mereka. Model ini berbeda dari lingkungan streaming aplikasi yang ada. Ini biasanya didasarkan pada penyediaan untuk kapasitas puncak, bahkan pada malam hari, akhir pekan, dan hari libur, ketika beban lebih rendah. Alat Harga Amazon AppStream 2.0 hanya memberikan perkiraan biaya AWS Anda yang terkait dengan penggunaan AppStream 2.0, dan tidak termasuk pajak apa pun yang mungkin berlaku. Biaya aktual Anda bergantung pada berbagai faktor, termasuk penggunaan layanan AWS Anda yang sebenarnya.

Alat Harga AppStream 2.0 disediakan sebagai spreadsheet Microsoft Excel atau OpenOffice Calc yang memungkinkan Anda memasukkan informasi dasar tentang armada Anda, kemudian memberikan perkiraan biaya untuk lingkungan AppStream 2.0 untuk armada sesuai permintaan dan selalu aktif berdasarkan pola penggunaan Anda. Anda dapat mensimulasikan biaya berdasarkan

tren penggunaan historis atau yang diantisipasi. Armada elastis membebaskan administrator dari kebutuhan untuk memprediksi penggunaan, membuat, memelihara kebijakan penskalaan dan gambar dengan memiliki fitur-fitur ini bawaan. Armada elastis dan instans yang menjalankan Amazon Linux 2 (semua jenis armada) ditagih untuk durasi sesi streaming, dalam hitungan detik, dengan minimal 15 menit.

Mengoptimalkan biaya dengan pilihan jenis instans

Untuk instance armada dan pembuat gambar, ada berbagai jenis dan jenis instans berbeda yang tersedia yang Anda pilih untuk aplikasi Anda.

Pengujian pengguna akhir — Langkah selanjutnya adalah meluncurkan armada AppStream 2.0 ke sekelompok pengguna pilot untuk pengujian guna memvalidasi pilihan jenis instans kami. Penting untuk meminta pengguna pilot untuk menguji semua alur kerja reguler dan berat mereka untuk menangkap metrik di sekitar memori, CPU, dan grafik sehingga Anda dapat menangkap metrik kinerja dasar. Grup percontohan harus berisi berbagai peran pengguna yang menggunakan aplikasi untuk memastikan Anda mengujinya dari beberapa pengalaman pengguna. Pengujian penerimaan pengguna memungkinkan Anda mengumpulkan umpan balik tentang pengalaman sesi streaming. Saat membuat atau memperbarui tumpukan, ada opsi untuk menggunakan URL umpan balik khusus. Pengguna diarahkan ke URL ini setelah mereka memilih tautan Kirim Umpan Balik untuk mengirimkan umpan balik tentang pengalaman streaming aplikasi mereka. Jika ada hambatan kinerja, gunakan metrik kinerja Windows untuk menganalisis kendala sumber daya. Misalnya, jika tipe instance armada saat ini stream.standard.medium menunjukkan batasan sumber daya, maka tingkatkan jenis instance ke stream.standard.large. Sebaliknya, jika metrik kinerja menunjukkan tingkat penggunaan sumber daya yang kurang tinggi, pertimbangkan untuk menurunkan jenis instans.

Mengoptimalkan biaya dengan pilihan tipe armada

Saat membuat armada AppStream 2.0 baru, pengembang harus memilih jenis armada Always-On atau On-Demand. Saat memilih jenis instans dari perspektif harga, penting untuk memahami bagaimana AppStream 2.0 mengelola instance armada. Untuk armada Always-On, instance armada tetap dalam kondisi berjalan. Oleh karena itu, ketika pengguna mencoba melakukan streaming sesi, instance armada selalu siap untuk memulai sesi streaming.

Untuk armada On-Demand, setelah instance armada diluncurkan, mereka disimpan dalam keadaan berhenti. Biaya instans berhenti lebih rendah daripada biaya instans berjalan, yang dapat membantu

mengurangi biaya. Instance armada On-Demand harus dimulai dari status berhenti. Pengguna harus menunggu sekitar dua menit agar sesi streaming mereka tersedia.

Armada elastis adalah kandidat yang baik untuk aplikasi yang mandiri dan dapat diinstal ke hard drive virtual yang disimpan dalam bucket Amazon Simple Storage Service (Amazon S3). Armada elastis selanjutnya dapat mengurangi biaya untuk beberapa kasus penggunaan karena tagihan per detik yang dibebankan hanya selama durasi streaming. Tarif adalah fungsi dari jenis dan ukuran instans dan sistem operasi yang Anda pilih saat membuat armada.

Jika pengguna akhir membutuhkan instance armada selama jam kerja, lebih baik menyimpan sesi streaming yang sama. Ini karena instans armada dibebankan per jam, dan setiap kali sesi streaming baru dimulai, itu menimbulkan biaya instans armada lain.

Tabel 10 - AppStream 2.0 perbandingan tipe armada

Jenis armada	Keuntungan	Pertimbangan
Selalu Aktif	Kurang waktu tunggu untuk sesi streaming	Pengguna membayar biaya instans per jam karena tidak ada opsi untuk menyimpan instance dalam status berhenti.
Sesuai Permintaan	Penghematan biaya karena instans tetap dalam keadaan berhenti	Waktu tunggu yang lebih lama untuk sesi streaming
Elastis	Penagihan per detik mungkin berguna untuk kasus penggunaan yang memiliki pola penggunaan sporadis untuk aplikasi yang dapat diinstal pada hard disk virtual	Karena ukuran hard disk virtual aplikasi menjadi lebih besar, waktu yang dibutuhkan untuk memasangnya ke instance streaming bisa lama

AppStream 2.0 memantau pemanfaatan armada Anda dan melakukan penyesuaian otomatis pada kapasitas armada untuk memenuhi permintaan pengguna Anda dengan biaya serendah mungkin. Penyesuaian kapasitas dibuat berdasarkan kebijakan penskalaan yang Anda tentukan, baik berdasarkan pemanfaatan saat ini atau berdasarkan jadwal. Tinjau metrik penggunaan armada

secara berkala untuk memvalidasi bahwa kebijakan penskalaan armada tidak memiliki kapasitas cadangan tingkat tinggi.

Kebijakan penskalaan

Fleet Auto Scaling memungkinkan Anda untuk mengoptimalkan sumber daya armada dengan tidak harus melakukan terlalu banyak sumber daya menunggu pengguna untuk login. Administrator dapat menyesuaikan ukuran armada berdasarkan berbagai pemanfaatan agar sesuai dengan permintaan pengguna. Gunakan CloudWatch AppStream 2.0 Fleet Metrics atau alat pemantauan pihak ketiga untuk mempelajari aktivitas pengguna dan mengonfigurasi kebijakan penskalaan untuk memperluas atau mengecilkan armada AppStream 2.0 berdasarkan penggunaan yang diharapkan. Log pengguna adalah mekanisme penting untuk mendapatkan pemahaman tentang penggunaan nyata. Wawasan ini dapat digunakan untuk mengubah ukuran armada secara dinamis berdasarkan Auto Scaling.

Dalam banyak kasus, armada AppStream 2.0 dibuat berdasarkan jumlah maksimum pengguna dan tidak disesuaikan untuk waktu yang berbeda dalam sehari dan minggu seperti malam dan akhir pekan. Sering kali, jumlah pengguna bersamaan dari aplikasi streaming kurang dari jumlah total pengguna terutama ketika pengguna memiliki fleksibilitas untuk bekerja dari jarak jauh. Penting untuk mempertimbangkan faktor-faktor ini saat memproyeksikan pola penggunaan. Melebih-lebihkan menyebabkan penyediaan berlebih dari AppStream 2.0 instans yang mengakibatkan biaya tambahan. Untuk mendapatkan konfigurasi yang optimal, Anda mungkin perlu menggabungkan satu atau beberapa kebijakan penskalaan terjadwal dengan kebijakan skala keluar.

Untuk mempelajari selengkapnya tentang penerapan Kebijakan Penskalaan, tinjau Penskalaan armada Amazon AppStream 2.0 Anda.

Biaya pengguna

Biaya pengguna dibebankan per pengguna, per bulan di masing-masing Wilayah AWS tempat pengguna melakukan streaming aplikasi dari AppStream 2.0 instance armada. Alih-alih menghasilkan ID pengguna yang berbeda, memiliki ID pengguna yang konsisten untuk pengguna AppStream 2.0. Biaya pengguna tidak dikenakan biaya saat menghubungkan ke pembuat gambar.

Sekolah, universitas, dan lembaga publik tertentu mungkin memenuhi syarat untuk mengurangi biaya pengguna Microsoft RDS SAL sebesar $0,44 per pengguna per bulan. Untuk persyaratan kualifikasi, lihat Persyaratan dan Dokumen Lisensi Microsoft.

Jika Anda memiliki Microsoft License Mobility, Anda mungkin memenuhi syarat untuk membawa Lisensi Akses Klien Microsoft RDS (CAL) Anda sendiri dan menggunakannya dengan Amazon

2.0. AppStream Jika Anda ditanggung oleh lisensi Anda sendiri, Anda tidak akan dikenakan biaya pengguna bulanan. Untuk informasi selengkapnya tentang apakah Anda dapat menggunakan lisensi Microsoft RDS CAL yang ada dengan Amazon AppStream 2.0, lihat panduan Mobilitas AWS Lisensi, atau konsultasikan dengan perwakilan lisensi Microsoft Anda.

Penggunaan Image Builder

AppStream 2.0 Instans Image Builder dikenakan biaya per jam. Biaya instans Image Builder mencakup komputasi, penyimpanan, dan lalu lintas jaringan apa pun yang digunakan oleh protokol streaming. Semua instance Image Builder yang sedang berjalan dikenakan biaya instans berjalan yang berlaku. Biaya ini didasarkan pada jenis dan ukuran instans, bahkan ketika tidak ada administrator yang terhubung.

Sebagai praktik terbaik untuk mengoptimalkan biaya, matikan instance Image Builder saat tidak digunakan. CloudWatch Aturan acara dapat digunakan untuk menjadwalkan pekerjaan sehari-hari, seperti menjalankan fungsi Lambda untuk menghentikan instance pembuat gambar.

Anda dapat menyimpan gambar AppStream 2.0 Anda up-to-date dengan menggunakan pembaruan gambar AppStream 2.0 terkelola. Metode pembaruan ini menyediakan pembaruan sistem operasi Windows terbaru dan pembaruan driver, dan perangkat lunak agen AppStream 2.0 terbaru. Saat menggunakan metode ini untuk memperbarui gambar, Image Builder secara otomatis dimulai, dan dihentikan, sebagai bagian dari proses layanan terkelola.

Kesimpulan

Dengan AppStream 2.0, Anda dapat dengan mudah menambahkan aplikasi desktop yang ada ke AWS dan memungkinkan pengguna Anda untuk mengalirkannya secara instan. Pengguna Windows dapat menggunakan klien AppStream 2.0 atau browser web berkemampuan HTML5 untuk streaming aplikasi. Anda dapat mempertahankan satu versi dari setiap aplikasi Anda, yang membuat manajemen aplikasi lebih mudah. Pengguna Anda selalu mengakses versi terbaru dari aplikasi mereka. Aplikasi Anda berjalan pada sumber daya AWS komputasi, dan data tidak pernah disimpan di perangkat pengguna, yang berarti mereka selalu mendapatkan pengalaman berkinerja tinggi dan aman.

Tidak seperti solusi lokal tradisional untuk streaming aplikasi desktop, AppStream menawarkan pay-as-you-go harga, tanpa investasi di muka dan tidak ada infrastruktur untuk dipelihara. Anda dapat menskalakan secara instan dan global, memastikan bahwa pengguna Anda selalu memiliki pengalaman yang luar biasa.

Amazon AppStream 2.0 dirancang untuk diintegrasikan ke dalam sistem dan proses TI yang ada, dan whitepaper ini menjelaskan praktik terbaik untuk melakukan ini. Hasil dari mengikuti pedoman dalam whitepaper ini adalah penerapan desktop cloud hemat biaya yang dapat disesuaikan dengan bisnis Anda pada infrastruktur global dengan aman. AWS

Kontributor

Kontributor dokumen ini meliputi:

- Andrew Wood, Arsitek Solusi Sr., Amazon Web Services
- Andrew Morgan, Spesialis EUC SA, Amazon Web Services
- Arun PC, Spesialis Sr EUC SA, Amazon Web Services
- Asriel Agronin, Arsitek Solusi Sr., Amazon Web Services
- Dustin Shelton, Spesialis Sr EUC SA, Amazon Web Services
- Jeremy Schiefer, Arsitek Solusi Sr, Amazon Web Services
- Navi Magea, Arsitek Solusi Utama, Amazon Web Services
- Pete Fergus, Insinyur Dukungan Cloud Sr, Amazon Web Services
- Phil Persson, Spesialis Utama EUC SA, Amazon Web Services
- Richard Spaven, Sr Spesialis EUC SA, Amazon Web Services
- Spencer DeBrosse, Sr. Solusi Arsitek, Amazon Web Services
- Stephen Stetler, Arsitek Solusi Sr., Amazon Web Services
- Taka Matsumoto, Insinyur Dukungan Cloud Sr, Amazon Web Services
- Vasant Sirsat, Spesialis Sr EUC SA, Amazon Web Services

Bacaan lebih lanjut

Untuk informasi tambahan, lihat:

- Panduan Administrasi Amazon AppStream 2.0
- Amazon AppStream API Referensi
- Gunakan Amazon FSx for Windows File Server dan FSLogix untuk Mengoptimalkan Persistensi Pengaturan Aplikasi di Amazon 2.0 AppStream
- Memantau Amazon AppStream 2.0 dengan Amazon ElasticSearch dan Amazon Firehose
- Analisis Laporan Penggunaan Amazon AppStream 2.0 Anda Menggunakan Amazon Athena dan Amazon QuickSight
- Skala armada Amazon AppStream 2.0 Anda
- Menggunakan Microsoft AppLocker untuk mengelola pengalaman aplikasi di Amazon AppStream 2.0
- Menggunakan domain khusus dengan Amazon AppStream 2.0
- Bagaimana cara menggunakan Microsoft RDS CAL saya sendiri dengan 2.0? AppStream
- Alat Harga Amazon AppStream 2.0
- Buat Uji Coba Perangkat Lunak Online dengan AppStream 2.0
- Buat Portal SaaS dengan Amazon 2.0 AppStream

Revisi dokumen

Untuk diberitahu tentang pembaruan pada whitepaper ini, berlangganan RSS feed.

Perubahan	Deskripsi	Tanggal
Dokumen diperbarui	Pembaruan untuk menyertakan armada Elastic, dengan hak aplikasi berbasis tribute, katalog aplikasi multi-stack, armada berbasis Linux, masuknya dan keluar data, pemulihan bencana, dan pembaruan lainnya.	Juni 14, 2022
Dokumen diperbarui	Versi HTML diterbitkan.	Januari 19, 2022
Publikasi awal	Whitepaper diterbitkan.	8 Juni 2021

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. AWSproduk 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.

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