Menerapkan AWS Well-Architected Framework ke WorkSpaces Aplikasi Amazon

AWS Panduan Preskriptif

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS Panduan Preskriptif: Menerapkan AWS Well-Architected Framework ke WorkSpaces Aplikasi Amazon

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

Table of Contents

Pengantar 1
Audiens yang dituju 1
Tujuan 2
Pilar keunggulan operasional 3
Mengatur tim di sekitar hasil bisnis 3
Menerapkan observabilitas untuk wawasan yang dapat ditindaklanjuti 4
Otomatiskan dengan aman jika memungkinkan 5
Buat perubahan yang sering, kecil, dan reversibel 7
Perbaiki prosedur operasi sesering mungkin 8
Mengantisipasi kegagalan 9
Belajar dari semua peristiwa dan metrik operasional 10
Gunakan layanan terkelola 10
Pilar keamanan 11
Menerapkan fondasi identitas yang kuat 11
Pertahankan ketertelusuran 12
Terapkan keamanan di semua lapisan 13
Mengotomatiskan praktik terbaik keamanan 14
Jauhkan orang dari data 14
Mempersiapkan acara keamanan 15
Pilar keandalan 17
Secara otomatis pulih dari kegagalan 17
Uji prosedur pemulihan 18
Skala horizontal untuk meningkatkan ketersediaan beban kerja agregat 18
Berhenti menebak kapasitas 19
Kelola perubahan melalui otomatisasi 19
Pilar efisiensi performa 20
Demokratisasi teknologi canggih 20
Menjadi global dalam hitungan menit 21
Gunakan arsitektur tanpa server 21
Eksperimen lebih sering 22
Pertimbangkan simpati mekanis 22
Pilar optimasi biaya 24
Menerapkan manajemen keuangan cloud 24
Tambahkan model konsumsi 24

Ukur efisiensi keseluruhan 25
Berhenti menghabiskan uang untuk angkat berat yang tidak berdiferensiasi 25
Analisis dan atribut pengeluaran 26
Pilar keberlanjutan 27
Pahami dampak Anda 27
Menetapkan tujuan keberlanjutan 27
Maksimalkan pemanfaatan 28
Mengantisipasi dan mengadopsi penawaran perangkat keras dan perangkat lunak baru yang lebih efisien 28
Layanan terkelola yang digunakan 28
Mengurangi dampak hilir beban kerja cloud Anda 29
Sumber daya 30
AWS dokumentasi 30
AWS posting blog 30
Riwayat dokumen 32
..... xxxiii

Menerapkan Kerangka Kerja AWS Well-Architected ke Aplikasi Amazon WorkSpaces

Mohamed Wali, Amazon Web Services

Juli 2025 (sejarah dokumen)

Panduan ini mencakup praktik terbaik untuk menerapkan Kerangka Kerja AWS Well-Architected saat Anda menggunakan Aplikasi Amazon. WorkSpaces WorkSpaces Aplikasi adalah layanan streaming aplikasi yang dikelola sepenuhnya yang memungkinkan Anda melakukan streaming aplikasi desktop ke pengguna tanpa menulis ulang.

AWS Well-Architected Framework membantu arsitek cloud membangun infrastruktur yang aman, berkinerja tinggi, tangguh, dan efisien untuk berbagai aplikasi dan beban kerja. Ini juga menyediakan pendekatan yang konsisten bagi pengguna dan AWS Mitra untuk mengevaluasi arsitektur dan menerapkan desain yang dapat diskalakan.

The AWS Well-Architected Framework dibangun di sekitar enam pilar:

- Keunggulan operasional
- Keamanan
- Keandalan
- Efisiensi kinerja
- Optimalisasi biaya
- Keberlanjutan

Panduan ini membahas bagaimana pilar dan praktik terbaik ini berlaku untuk menggunakan WorkSpaces Aplikasi.

Audiens yang dituju

Panduan ini untuk:

- Arsitek dan insinyur cloud yang merancang dan mengimplementasikan solusi WorkSpaces Aplikasi, dan perlu memastikan bahwa arsitekturnya mengikuti praktik terbaik AWS Well-Architected Framework.

- Tim operasi TI yang mengelola dan memelihara lingkungan WorkSpaces Aplikasi; menangani manajemen armada, penskalaan, dan pemantauan; dan perlu mengoptimalkan biaya dan kinerja.
- Organizations atau bisnis yang sedang mempertimbangkan atau sudah menggunakan WorkSpaces Applications, ingin melakukan streaming aplikasi desktop ke penggunanya, dan perlu membangun infrastruktur yang aman, berkinerja tinggi, tangguh, dan efisien.

Tujuan

Mengikuti praktik terbaik dalam panduan ini membantu Anda:

- Bangun infrastruktur yang aman, berkinerja tinggi, tangguh, dan efisien untuk streaming aplikasi desktop di aplikasi desktop. AWS Cloud
- Terapkan pendekatan yang konsisten saat mengevaluasi arsitektur WorkSpaces Aplikasi dan menerapkan desain yang dapat diskalakan.

Pilar keunggulan operasional

Operational Excellence (OE) mewakili dedikasi untuk menyusun solusi perangkat lunak berkualitas tinggi yang secara konsisten memenuhi dan melampaui harapan pengguna. Pilar keunggulan operasional dari AWS Well-Architected Framework mencakup strategi yang telah terbukti untuk organisasi tim yang efektif, desain beban kerja yang kuat, operasi skala besar yang efisien, dan adaptasi tanpa batas terhadap perubahan persyaratan dari waktu ke waktu. Dengan mematuhi prinsip-prinsip ini, organisasi dapat memastikan bahwa sistem mereka tetap tangguh, berkinerja, dan selaras dengan kebutuhan bisnis yang berkembang.

Area fokus utama untuk menerapkan pilar ini ke lingkungan streaming WorkSpaces Aplikasi Anda:

- Pemantauan dan observabilitas
- Otomatisasi dan DevOps
- Prosedur dan dokumentasi operasional
- Support dan manajemen insiden

Mengatur tim di sekitar hasil bisnis

Buat model operasi yang selaras dengan cloud dengan komitmen kepemimpinan yang kuat, di mana tujuan bisnis dan indikator kinerja utama (KPIs) mendorong transformasi organisasi melalui orang, proses, dan teknologi yang dioptimalkan.

- Struktur tim. Bangun tim khusus yang selaras dengan hasil streaming aplikasi. Contoh:
  - Tim manajemen gambar bertanggung jawab atas pengemasan aplikasi dan pengoptimalan gambar.
  - Tim operasi armada mengelola kapasitas, kinerja, dan penskalaan.
  - Tim pengalaman pengguna menangani dukungan dan kepuasan pengguna akhir.
- KPIs dan metrik. Tentukan dan lacak metrik yang selaras dengan bisnis seperti:
  - Tarif ketersediaan aplikasi
  - Saatnya menyebarkan aplikasi baru
  - Biaya per jam streaming aplikasi
- Model operasi. Buat proses yang jelas untuk:
  - Orientasi dan pembaruan aplikasi

- Manajemen kapasitas armada
- Penyediaan akses pengguna
- Respon dan resolusi insiden

Menerapkan observabilitas untuk wawasan yang dapat ditindaklanjuti

Menerapkan pemantauan dan observabilitas yang komprehensif untuk melacak KPIs dan kesehatan beban kerja. Prinsip ini memungkinkan keputusan berbasis data dan peningkatan proaktif di seluruh kinerja, keandalan, dan biaya.

- Melaksanakan pemantauan kinerja. Konfigurasikan Amazon CloudWatch ke:
  - Pastikan kapasitas yang cukup untuk memenuhi permintaan. Misalnya, Anda dapat menggunakan metrik berikut:
    - `AvailableCapacity`untuk memantau instans streaming yang tersedia
    - `InUseCapacity`untuk melacak instance yang saat ini digunakan
    - `CapacityUtilization`untuk memantau persentase penggunaan armada
  - Pantau pengalaman dan kinerja pengguna.
  - Identifikasi dan atasi masalah layanan dengan segera.
- Lacak dan analisis laporan penggunaan WorkSpaces Aplikasi.
- Tangkap dan analisis log aplikasi. Untuk informasi selengkapnya, lihat posting AWS blog Menggunakan Agen Kinesis untuk Linux untuk melakukan streaming log WorkSpaces aplikasi di Aplikasi dan Menggunakan Agen Kinesis untuk Microsoft Windows untuk menyimpan log peristiwa WorkSpaces Aplikasi Windows.
- Pantau metrik dan acara WorkSpaces Aplikasi melalui notifikasi obrolan. Untuk informasi lebih lanjut, lihat posting AWS blog Memantau dan mengotomatiskan komputasi pengguna AWS akhir (EUC) dengan AWS Chatbot.
- Aktifkan manajemen sesi proaktif melalui isyarat visual. Untuk informasi selengkapnya, lihat posting AWS blog Kedaluwarsa sesi tampilan dan penghitung waktu mundur di Aplikasi Amazon. WorkSpaces
- Buat visualisasi untuk pola dan tren penggunaan. Untuk informasi selengkapnya, lihat posting AWS blog Ingest dan visualisasikan laporan penggunaan WorkSpaces Aplikasi Amazon di Layanan Amazon OpenSearch .

- Manfaatkan toolkit EUC untuk memantau sesi aktif, melacak inventaris armada, dan menghasilkan laporan sesi (ekspor CSV). Untuk informasi selengkapnya, lihat posting AWS blog Gunakan EUC Toolkit untuk mengelola WorkSpaces Aplikasi Amazon dan Amazon. WorkSpaces

Otomatiskan dengan aman jika memungkinkan

Terapkan prinsip infrastruktur sebagai kode (IAc) untuk mengotomatiskan semua aspek operasi beban kerja Anda. Gunakan pagar pembatas untuk membantu memastikan eksekusi yang aman dan konsisten sekaligus mengurangi intervensi manual.

- Otomatiskan pembuatan dan konfigurasi gambar WorkSpaces Aplikasi dengan menggunakan Image Assistant CLI. Untuk informasi selengkapnya, lihat Membuat gambar WorkSpaces Aplikasi Amazon Anda secara terprogram menggunakan operasi CLI Asisten Gambar di dokumentasi Aplikasi. WorkSpaces
  - Instalasi aplikasi: Gunakan Image Assistant CLI untuk mengotomatiskan instalasi aplikasi selama pembuatan gambar.
  - Pembuatan gambar: Buat gambar WorkSpaces Aplikasi secara terprogram dengan menggunakan perintah Image Assistant CLI.
  - Manajemen konfigurasi: Otomatiskan konfigurasi pengaturan aplikasi default dan parameter peluncuran.
- Otomatiskan kustomisasi gambar WorkSpaces Aplikasi. Untuk informasi selengkapnya, lihat posting AWS blog Membuat gambar Windows WorkSpaces Aplikasi yang disesuaikan secara otomatis.
- Terapkan IAc untuk menyebarkan infrastruktur dan komponen aplikasi untuk WorkSpaces Aplikasi. Untuk informasi selengkapnya, lihat posting AWS blog Otomatisasi infrastruktur dan penerapan aplikasi untuk WorkSpaces Aplikasi Amazon dengan Terraform.
- Menerapkan proses otomatis untuk manajemen armada, termasuk:
  - Penskalaan armada berdasarkan permintaan. Konfigurasikan kebijakan penskalaan otomatis untuk menyesuaikan kapasitas armada secara otomatis berdasarkan metrik pemanfaatan. Untuk informasi selengkapnya, lihat posting AWS blog Gunakan AWS Lambda untuk menyesuaikan langkah penskalaan dan ambang batas untuk Aplikasi Amazon. WorkSpaces
  - Pembaruan gambar dasar. Manfaatkan pembaruan otomatis ke gambar dasar WorkSpaces Aplikasi yang disediakan oleh AWS.

    - Optimalisasi kapasitas. Siapkan ambang batas penskalaan otomatis untuk mengoptimalkan penggunaan sumber daya berdasarkan pola permintaan.
- Konfigurasikan pagar pembatas untuk mengotomatiskan kontrol keselamatan:
    - Batas ukuran armada maksimum. Tetapkan batas atas kapasitas armada untuk mencegah penyediaan berlebih.
    - Konfigurasi kebijakan penskalaan. Menerapkan penskalaan langkah atau kebijakan penskalaan pelacakan target dengan ambang batas yang sesuai.
    - Kuota layanan. Gunakan kuota AWS layanan sebagai batas bawaan untuk mencegah alokasi sumber daya yang berlebihan.
    - Perlindungan skala. Konfigurasikan perlindungan penskalaan untuk mencegah penghapusan instance aktif selama peristiwa penskalaan.
- Lakukan pengujian dan validasi, termasuk pembuat gambar, armada, dan pengujian integrasi.
    - Pengujian pembuat gambar:
        - Uji aplikasi langsung di antarmuka pembuat gambar.
        - Verifikasi peluncuran dan fungsionalitas aplikasi.
        - Uji pengaturan dan konfigurasi pengguna.
        - Validasi kompatibilitas aplikasi.
    - Pengujian armada:
        - Uji sesi streaming dari perangkat klien yang berbeda.
        - Verifikasi hak dan akses pengguna.
        - Validasi kinerja aplikasi.
        - Uji pengalaman pengguna untuk elemen dan operasi seperti clipboard, transfer file, dan pencetakan.
    - Pengujian integrasi:
        - Uji Active Directory atau otentikasi berbasis SAM 2.0.
        - Uji folder rumah dan penyimpanan persisten.
        - Uji hak aplikasi.
        - Uji pengalihan perangkat USB (jika dikonfigurasi).
- Gunakan WorkSpaces Applications Applications Manager untuk mengotomatiskan pengemasan dan penyebaran aplikasi. Untuk informasi selengkapnya, lihat posting AWS blog Streamline onboarding aplikasi dengan manajer aplikasi untuk Aplikasi Amazon WorkSpaces .

- Otomatiskan penerapan versi aplikasi baru dengan menggunakan pipeline continuous integration dan continuous delivery (CI/CD). Untuk informasi selengkapnya, lihat posting AWS blog Screening Eagle: Optimalkan CI/CD dan pengalaman pengguna akhir di WorkSpaces Aplikasi Amazon.

Buat perubahan yang sering, kecil, dan reversibel

Bangun beban kerja yang digabungkan secara longgar dan dapat diskalakan yang memungkinkan penerapan otomatis skala kecil yang sering dengan risiko minimal dan kemampuan rollback yang mudah.

- Untuk pembaruan gambar, gunakan pembuatan gambar berversi dan pembaruan tambahan.
  - Pembuatan gambar berversi:
    - Buat gambar baru untuk setiap rangkaian perubahan dengan menggunakan pembuat gambar.
    - Pertahankan beberapa versi gambar untuk mendukung skenario rollback.
    - Gunakan strategi AWS penandaan untuk melacak versi dan atribut gambar.
  - Pembaruan tambahan:
    - Buat perubahan kecil dan bertahap pada aplikasi atau konfigurasi.
    - Uji pembaruan secara menyeluruh di pembuat gambar sebelum Anda membuat gambar baru.
    - Dokumentasikan semua perubahan yang Anda buat di setiap versi gambar baru.
- Untuk pembaruan armada kontrol:
  - Buat armada baru dengan gambar yang diperbarui untuk pengujian.
  - Ubah atribut armada yang ada tanpa mengganggu sesi aktif.
- Menetapkan prosedur manajemen perubahan untuk dokumentasi, protokol pengujian, alur kerja persetujuan, dan proses pemantauan.
  - Dokumentasi:
    - Pertahankan log perubahan terperinci untuk semua pembaruan gambar dan armada.
    - Dokumen prosedur pengujian dan hasil untuk setiap perubahan.
    - Gunakan AWS CloudTrailuntuk melacak dan mengaudit perubahan konfigurasi.
  - Protokol pengujian:
    - Menetapkan proses pengujian yang komprehensif untuk semua perubahan.
    - Sertakan fungsionalitas aplikasi, kinerja, dan pengujian pengalaman pengguna.
    - Lakukan pengujian di pembuat gambar sebelum Anda membuat gambar baru.

    - Lakukan pengujian tambahan pada armada non-produksi sebelum penyebaran penuh.
- Alur kerja persetujuan:
    - Menerapkan proses persetujuan untuk perubahan lingkungan produksi.
    - Tentukan kriteria untuk perubahan yang memerlukan persetujuan versus pembaruan standar.
    - Menetapkan peran dan tanggung jawab untuk persetujuan perubahan.
- Pemantauan dan validasi:
    - Gunakan Amazon CloudWatch untuk memantau kinerja armada dan aplikasi setelah perubahan.
    - Siapkan peringatan untuk metrik utama untuk mengidentifikasi masalah dengan cepat setelah pembaruan.
    - Lakukan tinjauan pasca-implementasi untuk memvalidasi keberhasilan perubahan dan mengumpulkan pembelajaran.

Perbaiki prosedur operasi sesering mungkin

Terus meningkatkan prosedur operasional melalui tinjauan rutin, pembaruan, dan keterlibatan tim untuk menjaga agar semua pemangku kepentingan tetap terinformasi dan selaras dengan praktik terbaik.

- Manajemen dokumentasi. Pertahankan dokumentasi prosedur WorkSpaces Aplikasi yang dikendalikan versi saat ini di lokasi pusat untuk memastikan konsistensi operasional dan berbagi pengetahuan di seluruh tim.
    - Dokumentasi yang diperlukan: Menjaga up-to-date dokumentasi untuk operasi WorkSpaces Aplikasi penting untuk pembuatan dan manajemen gambar, operasi armada, dan pemecahan masalah.
    - Tinjauan operasional: Memantau dan meninjau aspek operasional utama, termasuk metrik kinerja dan manajemen insiden.
- Perbaikan berkelanjutan. Secara sistematis meningkatkan operasi WorkSpaces Aplikasi dengan memasukkan Layanan AWS pembaruan, metrik operasional, dan praktik terbaik yang dipelajari ke dalam prosedur standar.
    - Pembaruan layanan: Memantau catatan rilis WorkSpaces Aplikasi untuk fitur baru, peningkatan layanan, pembaruan keamanan, dan ketersediaan Regional.

- Praktik terbaik: Tinjau dan gabungkan pembaruan AWS Well-Architected Framework WorkSpaces , praktik terbaik Aplikasi, arsitektur referensi AWS , dan rekomendasi keamanan. AWS
- Manajemen pengetahuan: Memelihara dan memperbarui prosedur operasi standar, runbook, panduan pemecahan masalah, dan dokumentasi dukungan pengguna.

Mengantisipasi kegagalan

Lakukan pengujian skenario kegagalan secara teratur untuk memahami risiko, memvalidasi prosedur respons, dan meningkatkan kesiapan tim untuk menangani insiden nyata.

- Pengujian kegagalan. Secara teratur mensimulasikan dan menguji kegagalan seperti kelelahan kapasitas armada, kegagalan peluncuran aplikasi, dan masalah konektivitas jaringan.
  - Kelelahan kapasitas armada:
    - Pantau dan uji perilaku penskalaan armada saat mendekati batas kapasitas.
    - Konfigurasikan CloudWatch alarm untuk `CapacityUtilization` dan `AvailableCapacity` metrik.
    - Menerapkan prosedur untuk menangani kendala kapasitas selama penggunaan puncak.
  - Kegagalan peluncuran aplikasi:
    - Uji perilaku peluncuran aplikasi pada instance streaming.
    - Validasi akses dan kinerja aplikasi di berbagai konfigurasi armada.
  - Masalah konektivitas jaringan:
    - Uji kinerja sesi streaming di berbagai kondisi jaringan.
    - Pantau `StreamingSessionLatency` masalah kualitas koneksi.
    - Pastikan konfigurasi pengaturan VPC dan grup keamanan yang tepat.
- Prosedur pemulihan. Mengembangkan dan menguji prosedur untuk:
  - Failover armada antara AWS Zona Ketersediaan. Selain itu, mendokumentasikan prosedur untuk penskalaan kapasitas armada, mengelola pembaruan armada, dan menanggapi masalah kesehatan misalnya.
  - Manajemen data pengguna:
    - Konfigurasikan dan uji solusi persistensi dan penyimpanan pengaturan aplikasi untuk folder rumah di Amazon Simple Storage Service (Amazon S3) untuk armada Windows dan sistem file bersama di Amazon Elastic File System (Amazon EFS) untuk armada Linux.

    - Validasi sinkronisasi data antar sesi.
  - Kontinuitas layanan. Menjaga prosedur untuk membuat instance armada baru, mengelola pembaruan gambar, dan menangani pemutusan sesi.
- Manajemen risiko. Identifikasi dan mitigasi:
  - Kendala kapasitas dengan menetapkan kapasitas minimum armada yang sesuai, mengonfigurasi kebijakan penskalaan otomatis berdasarkan pola permintaan, dan memantau tren pemanfaatan armada dengan menggunakan CloudWatch metrik seperti,, dan. `CapacityUtilization` `InUseCapacity` `AvailableCapacity`
  - Kemacetan kinerja dengan melacak metrik utama seperti `StreamingSessionLatency` dan mengonfigurasi alarm yang sesuai. CloudWatch

Belajar dari semua peristiwa dan metrik operasional

Menumbuhkan budaya perbaikan berkelanjutan dengan berbagi pelajaran yang dipetik dari peristiwa operasional dan kegagalan di seluruh organisasi. Tekankan dampaknya pada hasil bisnis.

- Analisis peristiwa. Mendokumentasikan dan menganalisis gangguan layanan, penurunan kinerja, keluhan pengguna, dan masalah kapasitas.
- Ulasan metrik. Menganalisis pola penggunaan, tren kinerja, metrik biaya, dan data kepuasan pengguna secara teratur.
- Berbagi pengetahuan. Menetapkan proses untuk sesi pembelajaran tim, dokumentasi praktik terbaik, transfer pengetahuan lintas tim, dan retrospektif insiden.

Gunakan layanan terkelola

Minimalkan overhead operasional dengan menggunakan layanan AWS terkelola dan membangun prosedur standar di sekitarnya. Integrasikan dengan layanan AWS terkelola berikut:

- AWS Systems Manageruntuk otomatisasi
- Amazon CloudWatch untuk pemantauan
- AWS Identity and Access Management (IAM) untuk kontrol akses
- Amazon S3 untuk penyimpanan pengguna untuk armada Windows
- Amazon EFS untuk penyimpanan pengguna untuk armada Linux
- AWS Directory Serviceuntuk otentikasi pengguna

Pilar keamanan

Pilar keamanan dari AWS Well-Architected Framework berfokus pada memanfaatkan kemampuan cloud untuk membantu membangun mekanisme perlindungan yang kuat untuk informasi, infrastruktur, dan sumber daya Anda. Prinsip-prinsip ini membantu meningkatkan postur keamanan Anda secara keseluruhan sekaligus memungkinkan inovasi.

Area fokus utama untuk menerapkan pilar ini ke lingkungan streaming WorkSpaces Aplikasi Anda:

- Integritas dan kerahasiaan data
- Mengelola izin pengguna
- Membangun kontrol untuk mendeteksi peristiwa keamanan

Menerapkan fondasi identitas yang kuat

Gunakan izin minimum yang diperlukan untuk mengakses AWS sumber daya sambil memusatkan manajemen identitas dan menghindari kredensi jangka panjang.

- Berikan izin istimewa paling sedikit untuk sumber daya WorkSpaces Aplikasi:
  - Buat peran IAM khusus untuk armada WorkSpaces Aplikasi dengan izin minimal yang diperlukan.
  - Konfigurasikan izin IAM terbatas untuk pembuat gambar.
  - Membatasi akses administratif ke fungsi manajemen WorkSpaces Aplikasi.
  - Tentukan izin granular untuk manajemen tumpukan dan armada.
- Menerapkan mekanisme otentikasi pengguna yang tepat:
  - Konfigurasikan federasi SAMP 2.0 untuk integrasi penyedia identitas perusahaan.
  - Siapkan Pusat Identitas AWS IAMuntuk manajemen pengguna.
  - Gunakan pialang identitas khusus hanya jika diperlukan untuk skenario otentikasi tertentu.
  - Menerapkan otentikasi multi-faktor (MFA) jika didukung.
- Kontrol akses pengguna ke aplikasi:
  - Konfigurasikan hak aplikasi untuk membatasi akses ke aplikasi tertentu.
  - Buat grup penugasan aplikasi berdasarkan peran pengguna.
  - Kelola akses aplikasi melalui izin tumpukan.

    - Menerapkan kebijakan sesi untuk mengontrol perilaku aplikasi.
- Amankan sesi pengguna dengan kontrol yang sesuai:
    - Konfigurasikan kebijakan batas waktu sesi.
    - Setel tindakan batas waktu pemutusan sambungan.
    - Menerapkan persyaratan ketekunan sesi.
    - Kontrol izin pengalihan sistem file.
- Konfigurasikan otentikasi berbasis sertifikat untuk Aplikasi. WorkSpaces Untuk informasi selengkapnya, lihat posting AWS blog Sederhanakan otentikasi berbasis sertifikat untuk WorkSpaces Aplikasi dan WorkSpaces dengan AWS Private CA Konektor untuk Direktori Aktif.
- Gunakan tag sesi untuk menerapkan kontrol akses berbutir halus. Untuk informasi selengkapnya, lihat posting AWS blog Gunakan tag sesi untuk menyederhanakan izin WorkSpaces Aplikasi.

Pertahankan ketertelusuran

Menerapkan pemantauan real-time dan sistem respons otomatis untuk semua perubahan dan aktivitas lingkungan.

- Konfigurasikan CloudWatch logging untuk log aplikasi untuk memantau peristiwa khusus aplikasi, termasuk peluncuran aplikasi, crash, dan kesalahan. Konfigurasikan log sesi untuk melacak informasi sesi streaming, termasuk sesi dimulai, berhenti, dan peristiwa koneksi pengguna.
- Aktifkan CloudTrail untuk mencatat semua panggilan API WorkSpaces Aplikasi dan untuk melacak peristiwa manajemen seperti pembuatan dan modifikasi armada, operasi pembuat gambar, konfigurasi tumpukan, dan aktivitas manajemen pengguna.
- Memantau aktivitas instance WorkSpaces Aplikasi:
    - Konfigurasikan pencatatan instance untuk menangkap peristiwa tingkat sistem.
    - Lacak peluncuran dan kegagalan aplikasi.
    - Memantau penggunaan dan kinerja sumber daya sistem.
- Lacak aktivitas pengguna:
    - Pantau upaya dan kegagalan otentikasi pengguna. Gunakan CloudWatch metrik dan CloudWatch Log untuk melacak upaya login pengguna, waktu mulai dan akhir sesi, dan peristiwa pemutusan sesi.

    - Lacak pola penggunaan aplikasi. Aktifkan laporan penggunaan WorkSpaces Aplikasi untuk mengambil informasi seperti durasi sesi, waktu mulai dan akhir, jenis instans yang digunakan, dan aplikasi yang diakses.
    - Rekam aktivitas sistem file melalui folder rumah yang diaktifkan.
    - Konfigurasikan pengaturan clipboard dan operasi pencetakan untuk mencapai tujuan pencegahan kehilangan data Anda.
- Konfigurasikan CloudWatchalarm untuk metrik terkait keamanan seperti otentikasi pengguna yang gagal, pola sesi yang tidak biasa, dan pelanggaran akses sumber daya.
- Gunakan toolkit EUC untuk melacak sesi dan status aktif, memantau alamat IP untuk sesi aktif yang sedang digunakan, dan mengekspor data sesi untuk audit. Untuk informasi selengkapnya, lihat posting AWS blog Gunakan toolkit EUC untuk mengelola WorkSpaces Aplikasi Amazon dan Amazon. WorkSpaces

Terapkan keamanan di semua lapisan

Terapkan beberapa lapisan kontrol keamanan di semua komponen infrastruktur Anda, dari tepi jaringan hingga kode aplikasi.

- Konfigurasikan keamanan lapisan jaringan:
    - Menerapkan aturan kelompok keamanan yang ketat.
    - Tempatkan instance armada WorkSpaces Aplikasi di subnet pribadi yang tidak memiliki akses internet langsung. Kontrol akses internet melalui perangkat NAT.
    - Gunakan titik akhir virtual private cloud (VPC) untuk mengakses yang didukung Layanan AWS (seperti Amazon S3).
    - Menerapkan daftar kontrol akses jaringan (ACLs) sebagai lapisan keamanan jaringan tambahan.
    - Batasi akses port streaming (TCP 8443 untuk HTTPS dan WebSocket Aman) ke rentang IP tertentu.
- Konfigurasikan keamanan lapisan akses:
    - Menerapkan kebijakan batas waktu sesi untuk secara otomatis memutuskan sambungan pengguna yang tidak aktif.
    - Menerapkan kontrol akses berbasis atribut dengan menggunakan tag sesi. Untuk informasi selengkapnya, lihat posting AWS blog Gunakan tag sesi untuk menyederhanakan izin WorkSpaces Aplikasi.

- Konfigurasikan keamanan lapisan aplikasi:
  - Konfigurasikan hak aplikasi untuk mengontrol pengguna mana yang dapat mengakses aplikasi tertentu.
  - Aktifkan kontrol pengalihan sistem file untuk membatasi akses ke drive lokal.
  - Konfigurasikan clipboard, transfer file, dan izin pencetakan berdasarkan persyaratan keamanan.
  - Siapkan kontrol akses perangkat USB sesuai dengan kebijakan keamanan.
- Konfigurasikan keamanan lapisan gambar:
  - Buat dan pertahankan gambar dasar yang diperkeras yang memenuhi persyaratan keamanan.
  - Tetap perbarui gambar dasar dengan tambalan keamanan terbaru.
  - Konfigurasikan pengaturan keamanan Windows dalam gambar dasar.
  - Nonaktifkan layanan dan fitur Windows yang tidak perlu dalam gambar dasar.

Mengotomatiskan praktik terbaik keamanan

Gunakan kontrol keamanan otomatis yang ditentukan kode dalam templat yang dikendalikan versi untuk mengaktifkan penerapan infrastruktur yang aman dan terukur.

- Gunakan infrastruktur sebagai kode (IAc) dengan menggunakan layanan seperti AWS CloudFormation untuk menerapkan konfigurasi keamanan yang konsisten di semua penerapan armada. Untuk informasi selengkapnya, lihat posting AWS blog Secara otomatis melampirkan grup keamanan tambahan ke WorkSpaces Aplikasi Amazon dan Amazon WorkSpaces.
- Otomatiskan proses keamanan pembuatan gambar dengan menggunakan Image Assistant CLI.
- Konfigurasikan respons otomatis untuk ambang batas pemanfaatan kapasitas yang terlampaui, upaya akses tidak sah, dan perubahan grup keamanan dengan menggunakan alarm Amazon, EventBridge aturan Amazon CloudWatch, dan fungsi untuk respons otomatis. AWS Lambda

Jauhkan orang dari data

Mengotomatiskan proses penanganan data untuk meminimalkan akses manusia secara langsung dan mengurangi risiko kesalahan atau kesalahan penanganan.

- Konfigurasikan hak aplikasi untuk mengontrol pengguna mana yang dapat mengakses aplikasi tertentu.

- Gunakan framework aplikasi dinamis untuk membangun penyedia aplikasi dinamis agar aplikasi tersedia secara dinamis berdasarkan atribut pengguna.
- Konfigurasikan pengalihan sistem file untuk mengontrol drive lokal mana yang dapat diakses pengguna, untuk membatasi akses ke folder tertentu, dan untuk mengelola izin transfer file antara sesi lokal dan streaming.
- Menerapkan pembatasan clipboard untuk menonaktifkan berbagi clipboard antara sesi lokal dan streaming, mengaktifkan aliran clipboard satu arah jika diperlukan, dan mencegah penyalinan data yang tidak sah.
- Konfigurasikan persistensi pengaturan aplikasi untuk menyimpan dan memulihkan konfigurasi aplikasi secara otomatis, menghilangkan kebutuhan konfigurasi manual, dan mempertahankan pengalaman pengguna yang konsisten.

Mempersiapkan acara keamanan

Kembangkan dan praktikkan rencana respons insiden dengan menggunakan alat otomatis untuk memungkinkan deteksi cepat, investigasi, dan pemulihan dari peristiwa keamanan.

- Siapkan CloudWatch alarm untuk upaya otentikasi yang gagal, perubahan pada grup keamanan armada, modifikasi konfigurasi gambar, dan pola sesi streaming yang tidak biasa.
- Prosedur respons dokumen untuk skenario keamanan WorkSpaces Aplikasi umum seperti:
  - Upaya akses tidak sah
    - Deteksi: Pantau kegagalan otentikasi.
    - Tanggapan: Cabut hak pengguna, tinjau log sesi, dan perbarui kebijakan akses.
  - Instans streaming yang dikompromikan
    - Deteksi: Pantau perilaku instance.
    - Respons: Hentikan sesi yang terpengaruh, ganti instance armada, dan tinjau konfigurasi grup keamanan.
  - Upaya eksfiltrasi data
    - Deteksi: Pantau aktivitas transfer file.
    - Tanggapan: Tinjau papan klip dan log transfer file, sesuaikan izin transfer file, dan perbarui kebijakan perlindungan data.
- Menerapkan proses pemulihan otomatis untuk penggantian instance armada, pemulihan grup keamanan, konfigurasi ulang akses pengguna, dan pemulihan pengaturan aplikasi.

- Gunakan Layanan AWS untuk manajemen keamanan, seperti AWS Security Hub CSPM untuk temuan keamanan dan Amazon GuardDuty untuk deteksi ancaman.

Pilar keandalan

Pilar keandalan dari AWS Well-Architected Framework membahas seberapa baik suatu sistem mempertahankan fungsionalitas dan tingkat kinerja yang dimaksudkan selama periode operasional yang diharapkan sepanjang masa pakainya. Ini memberikan pedoman komprehensif untuk membangun dan memelihara sistem yang dapat diandalkan AWS, termasuk strategi untuk pengujian dan validasi di semua tahapan siklus hidup beban kerja.

Area fokus utama untuk menerapkan pilar ini ke lingkungan streaming WorkSpaces Aplikasi Anda:

- Manajemen dan penskalaan armada
- Keandalan sesi
- Ketersediaan aplikasi
- Prosedur pemulihan

Secara otomatis pulih dari kegagalan

Pantau nilai bisnis KPIs untuk memicu respons otomatis yang dapat memprediksi, mencegah, atau memulihkan dari kegagalan sebelum berdampak pada operasi.

- Pastikan bahwa alokasi subnet IP Anda memperhitungkan ekspansi dan ketersediaan.
- Pantau CloudWatch metrik penting untuk memastikan ketersediaan dan kinerja layanan, termasuk metrik kapasitas armada seperti `AvailableCapacity` dan`InUseCapacity`, dan metrik kualitas streaming seperti. `StreamingSessionLatency`
- Konfigurasikan peringatan untuk ambang kapasitas, metrik kesehatan sesi, penurunan kinerja, dan perubahan status kesehatan armada.
- Gunakan kemampuan penskalaan otomatis WorkSpaces Aplikasi bawaan untuk:
  - Konfigurasikan kapasitas armada minimum dan maksimum.
  - Tetapkan kebijakan penskalaan berdasarkan pemanfaatan kapasitas.
  - Tentukan ambang batas scale-out dan scale-in berdasarkan metrik pengalaman pengguna dan persyaratan bisnis, bukan hanya metrik teknis.
- Bangun lingkungan pemulihan bencana untuk lingkungan WorkSpaces Aplikasi Anda. Untuk informasi lebih lanjut, lihat posting AWS blog Pertimbangan pemulihan bencana dengan WorkSpaces Aplikasi Amazon.

Uji prosedur pemulihan

Lingkungan cloud memungkinkan pengujian otomatis skenario kegagalan dan prosedur pemulihan. Kemampuan ini membantu Anda mengidentifikasi dan memperbaiki kerentanan sebelum kegagalan nyata terjadi.

- Pengujian pemulihan armada. Menerapkan pengujian pemulihan armada yang komprehensif di beberapa skenario:
  - Simulasikan penghentian instans untuk memverifikasi respons penskalaan otomatis.
  - Validasi pemeliharaan kapasitas minimum armada.
  - Uji waktu penggantian instance dan pengalihan pengguna.
  - Validasi efektivitas kebijakan penskalaan.
  - Uji batas kapasitas armada dan penanganan luapan.
- Pengujian pemulihan sesi. Menerapkan prosedur validasi pemulihan sesi:
  - Uji skenario pemutusan dan sambungkan kembali.
  - Verifikasi pelestarian status aplikasi.
  - Uji berbagai skenario gangguan jaringan.
  - Validasi perilaku batas waktu sesi.
  - Verifikasi persistensi otentikasi pengguna.
  - Verifikasi penanganan penyimpanan sementara.

Skala horizontal untuk meningkatkan ketersediaan beban kerja agregat

Distribusikan beban kerja Anda di beberapa sumber daya yang lebih kecil untuk meminimalkan dampak kegagalan individu dan untuk menghilangkan satu titik kegagalan.

- Menerapkan instance armada di beberapa Availability Zone.
- Konfigurasikan kapasitas armada minimum yang sesuai.
- Konfigurasikan penskalaan otomatis untuk armada dan tetapkan ambang batas penskalaan yang sesuai.
- Pantau pemanfaatan kapasitas di seluruh armada.

- Menerapkan tumpukan WorkSpaces Aplikasi di beberapa Wilayah. Untuk informasi selengkapnya, lihat posting AWS blog Optimalkan pengalaman pengguna dengan perutean berbasis latensi untuk Aplikasi Amazon. WorkSpaces

Berhenti menebak kapasitas

Gunakan kemampuan penskalaan otomatis cloud untuk menyesuaikan sumber daya secara dinamis berdasarkan permintaan. Ini membantu mencegah kejenuhan sumber daya sambil mempertahankan efisiensi optimal.

- Pantau metrik kunci seperti`CapacityUtilization`,`AvailableCapacity`, dan `InUseCapacity` untuk memahami kebutuhan kapasitas.
- Lacak tren pemanfaatan armada di berbagai periode waktu. Pantau pola harian, variasi mingguan, tren bulanan, dan puncak musiman.
- Siapkan kebijakan penskalaan dan konfigurasikan ambang batas penskalaan.
- Pastikan ada celah yang cukup antara kuota saat ini dan penggunaan maksimum untuk mengakomodasi failover.
- Mengakomodasi kuota dan kendala layanan tetap melalui arsitektur Anda.

Kelola perubahan melalui otomatisasi

Menerapkan perubahan infrastruktur melalui otomatisasi, termasuk perubahan yang dikendalikan versi pada kode otomatisasi itu sendiri.

- Gunakan IAc untuk konfigurasi armada.
- Menerapkan kebijakan penskalaan yang konsisten.
- Gunakan Image Assistant CLI untuk pembuatan gambar yang konsisten.

Pilar efisiensi performa

Pilar efisiensi kinerja dari AWS Well-Architected Framework berfokus pada pengoptimalan penggunaan sumber daya cloud untuk memenuhi atau melampaui tujuan kinerja sambil memastikan kemampuan beradaptasi terhadap permintaan yang berfluktuasi dan teknologi yang muncul. Ini menekankan pentingnya sistem fine-tuning terus menerus untuk mempertahankan efisiensi puncak dalam lingkungan cloud yang dinamis.

Area fokus utama untuk menerapkan pilar ini ke lingkungan streaming WorkSpaces Aplikasi Anda:

- Pemilihan dan pengoptimalan tipe instans
- Pengoptimalan kinerja streaming
- Manajemen kapasitas armada

Demokratisasi teknologi canggih

Manfaatkan layanan yang dikelola vendor cloud untuk teknologi yang kompleks sehingga tim Anda dapat fokus pada pengembangan produk alih-alih manajemen infrastruktur.

- Konfigurasikan jenis instans yang sesuai berdasarkan persyaratan aplikasi:
  - Pilih instans berkemampuan GPU untuk aplikasi intensif grafis.
  - Pilih keluarga GPU yang sesuai (seperti Graphics G4dn atau Graphics G5) berdasarkan kebutuhan aplikasi.
- Pilih dan konfigurasikan salah satu metode otentikasi berikut:
  - Siapkan integrasi dengan penyedia identitas berbasis SAMP 2.0.
  - Konfigurasikan pengaturan kumpulan pengguna.
  - Integrasikan dengan AWS Directory Service.
- Aktifkan dan konfigurasikan opsi penyimpanan berdasarkan kebutuhan pengguna:
  - Siapkan folder rumah di Amazon S3 untuk armada berbasis Windows.
  - Siapkan sistem file bersama di Amazon EFS untuk armada berbasis Linux.
  - Konfigurasikan izin penyimpanan persisten.
  - Aktifkan ketekunan pengaturan aplikasi.

Menjadi global dalam hitungan menit

Gunakan penyebaran multi-wilayah untuk meningkatkan pengalaman pengguna global melalui pengurangan latensi.

- Konfigurasikan armada dalam beberapa Wilayah AWS dengan menerapkan armada di Wilayah yang paling dekat dengan pengguna Anda sambil membuat tumpukan terpisah untuk setiap Wilayah.
- Terapkan pengalihan lintas wilayah untuk secara otomatis mengarahkan pengguna WorkSpaces Aplikasi ke AppStream tumpukan yang paling dekat dengan lokasi mereka saat ini.
- Jika Anda menggunakan salah satu fitur opsional dalam WorkSpaces Aplikasi, seperti persistensi pengaturan aplikasi, folder rumah, atau armada elastis, Anda perlu mengonfigurasi replikasi Amazon S3 Cross-region untuk data pengguna untuk armada berbasis Windows dan replikasi lintas wilayah untuk armada berbasis Linux.
- Replikasi gambar di seluruh Wilayah. Untuk informasi selengkapnya, lihat Menyalin gambar yang Anda miliki ke yang lain Wilayah AWS di WorkSpaces Aplikasi Amazon dalam AWS dokumentasi.
- Untuk armada yang bergabung dengan domain, pastikan infrastruktur Direktori Aktif, termasuk Layanan Federasi Direktori Aktif (AD FS) (kecuali jika Anda menggunakan SAMP 2.0 dan Amazon Cognito sebagai alternatif), dikonfigurasi dengan benar di Wilayah lain, dan yang Anda gunakan untuk kemampuan replikasi Multi-wilayah. AWS Directory Service for Microsoft Active Directory
- Arahkan pengguna ke titik akhir Aplikasi latensi terendah WorkSpaces . Untuk informasi selengkapnya, lihat posting AWS blog Optimalkan pengalaman pengguna dengan perutean berbasis latensi untuk Aplikasi Amazon. WorkSpaces

Gunakan arsitektur tanpa server

Arsitektur tanpa server menghilangkan overhead manajemen server dan mengurangi biaya dengan menggunakan layanan yang dikelola cloud untuk fungsi komputasi.

Gunakan layanan AWS tanpa server seperti berikut ini:

- AWS Lambdauntuk mengotomatiskan tugas dan mengintegrasikan logika khusus melalui fungsi yang digerakkan oleh peristiwa
- Amazon S3 menyediakan penyimpanan yang dapat diskalakan untuk data pengguna WorkSpaces Aplikasi, file aplikasi, dan artefak sesi

- Amazon CloudWatch menyediakan pemantauan, pencatatan, dan peringatan untuk metrik kinerja dan penggunaan WorkSpaces Aplikasi
- Amazon Cognito untuk memfasilitasi otentikasi pengguna dan kontrol akses untuk aplikasi Aplikasi WorkSpaces
- Amazon API Gateway RESTful APIs untuk membuat antarmuka antara WorkSpaces Aplikasi dan layanan lain atau aplikasi khusus

Eksperimen lebih sering

Infrastruktur cloud memungkinkan pengujian cepat berbagai konfigurasi sumber daya untuk mengoptimalkan kinerja dan biaya.

- Uji berbagai jenis instans untuk mengoptimalkan kinerja dan biaya:
  - Bandingkan kinerja streaming di berbagai kelompok instans.
  - Evaluasi instans GPU vs non-GPU untuk aplikasi grafis.
  - Uji instans yang dioptimalkan memori untuk aplikasi intensif memori.
- Menguji konfigurasi aplikasi dengan menggunakan Image Builder:
  - Buat gambar uji dengan konfigurasi aplikasi yang berbeda.
  - Validasi kinerja aplikasi sebelum penerapan.
  - Uji kompatibilitas aplikasi dengan jenis instans yang berbeda.
- Uji pengaturan armada dengan menggunakan konfigurasi kapasitas armada seperti kapasitas minimum dan maksimum, kebijakan penskalaan, pengaturan sesi seperti durasi sesi maksimum, dan pengaturan batas waktu pemutusan sambungan.

Pertimbangkan simpati mekanis

Pilih layanan cloud berdasarkan persyaratan dan pola penggunaan khusus beban kerja Anda untuk memastikan kinerja dan efisiensi yang optimal.

- Pilih instans Graphics G5 untuk aplikasi grafis intensif, aplikasi yang memerlukan DirectX, OpenGL, OpenCL, atau perangkat lunak visualisasi 3D.
- Pilih `stream.standard` instans untuk aplikasi bisnis, browser web, dan aplikasi grafis ringan
- Pantau dan sesuaikan protokol streaming berdasarkan CloudWatch metrik seperti`StreamingSessionLatency`.

- Konfigurasikan WorkSpaces Aplikasi VPCs yang paling dekat dengan pengguna Anda, dan gunakan bandwidth jaringan yang sesuai berdasarkan kebutuhan aplikasi Anda.
- Pilih jenis armada yang sesuai berdasarkan perilaku aplikasi. Misalnya, pilih armada sesi tunggal untuk aplikasi yang membutuhkan sumber daya khusus dan armada multi-sesi untuk aplikasi yang dapat berbagi sumber daya secara efisien.
- Pertimbangkan kompatibilitas aplikasi dengan lingkungan multi-sesi.
- Gunakan fitur pengalihan sistem file untuk menangani interaksi antara aplikasi jarak jauh dan lokal. Untuk informasi selengkapnya, lihat posting AWS blog Meluncurkan aplikasi lokal dari sesi streaming WorkSpaces Aplikasi Amazon.

Pilar optimasi biaya

Pilar optimasi biaya dari AWS Well-Architected Framework berfokus pada memaksimalkan nilai bisnis sambil meminimalkan pengeluaran. Ini membantu memastikan bahwa setiap dolar yang Anda belanjakan untuk sumber daya cloud berkontribusi secara efektif untuk mencapai tujuan organisasi Anda.

Area fokus utama untuk menerapkan pilar ini ke lingkungan streaming WorkSpaces Aplikasi Anda:

- Manajemen kapasitas armada dan pemilihan tipe instans
- Optimalisasi penskalaan dan penjadwalan
- Pemantauan dan analisis pola penggunaan
- Alokasi dan pelacakan biaya

Menerapkan manajemen keuangan cloud

Membangun kemampuan organisasi khusus dalam manajemen keuangan cloud dan optimalisasi biaya melalui program dan proses terstruktur untuk memaksimalkan nilai dan efisiensi cloud.

- Pantau biaya WorkSpaces Aplikasi dengan menggunakan AWS Cost Explorerdan laporan penggunaan untuk melacak penggunaan jam streaming, menganalisis biaya instans armada, dan memantau distribusi biaya regional.
- Rencanakan dan tetapkan kontrol biaya dengan menggunakan AWS Budgetsuntuk menetapkan peringatan untuk keseluruhan biaya layanan WorkSpaces Aplikasi, membuat ambang anggaran untuk layanan, dan memantau pengeluaran aktual terhadap jumlah yang dianggarkan. Untuk informasi selengkapnya, lihat posting AWS blog Cara menggunakan otomatisasi untuk mengoptimalkan dan mengontrol biaya WorkSpaces Aplikasi Amazon.

Tambahkan model konsumsi

Skala sumber daya komputasi dan biaya berdasarkan pola penggunaan aktual. Misalnya, Anda dapat mematikan lingkungan non-produksi selama jam kerja untuk mengoptimalkan pengeluaran.

- Pilih model harga yang sesuai. Misalnya, gunakan armada yang selalu aktif untuk penggunaan yang konsisten dan armada sesuai permintaan untuk beban kerja variabel.

- Pilih jenis instans yang optimal. Misalnya, gunakan `stream.standard` instance untuk aplikasi umum dan gunakan instance grafis (G4dn) hanya bila diperlukan.

Ukur efisiensi keseluruhan

Menghitung dan melacak output bisnis untuk mengukur peningkatan efisiensi dan memandu upaya optimasi. cost-per-unit

- Lacak efisiensi sesi.
- Pantau pemanfaatan armada dengan menggunakan CloudWatch metrik berikut:
  - `AvailableCapacity`untuk melacak kapasitas yang tidak terpakai
  - `InUseCapacity`untuk mengukur penggunaan aktual
- Hitung dan lacak biaya per sesi seperti biaya per jam streaming, biaya per pengguna, dan biaya per aplikasi.
- Terapkan Pengoptimal Biaya untuk WorkSpaces Aplikasi untuk memantau pembangun Anda.
- Bandingkan biaya antar jenis armada. Misalnya, bandingkan:
  - Biaya lisensi untuk sesi tunggal dan multi-sesi
  - Tingkat pemanfaatan sumber daya
  - Kepadatan pengguna per instance
- Gunakan data pelacakan proses untuk mengidentifikasi aplikasi yang kurang dimanfaatkan atau tidak perlu. Untuk informasi selengkapnya, lihat posting AWS blog Melacak proses pengguna di sesi WorkSpaces Aplikasi Amazon.

Berhenti menghabiskan uang untuk angkat berat yang tidak berdiferensiasi

AWS mengelola operasi infrastruktur dan menawarkan layanan terkelola sehingga organisasi Anda dapat fokus pada tujuan bisnis alih-alih pemeliharaan TI.

- Membuat dan memelihara gambar aplikasi dengan menggunakan Image Builder untuk mengemas aplikasi Anda, mengkonfigurasi pengaturan aplikasi, dan menguji kompatibilitas aplikasi.
- Konfigurasikan spesifikasi armada dengan memilih jenis instans yang sesuai dan menentukan ambang batas penskalaan serta menetapkan batas kapasitas yang diinginkan.

- Siapkan opsi penyimpanan persisten dengan mengonfigurasi folder beranda di Amazon S3 untuk armada berbasis Windows dan sistem file bersama di Amazon EFS untuk armada berbasis Linux. Tetapkan izin penyimpanan dan tentukan kebijakan penyimpanan.

Analisis dan atribut pengeluaran

Cloud memungkinkan pelacakan penggunaan sumber daya dan biaya per beban kerja secara akurat, yang memungkinkan pengukuran laba atas investasi (ROI) yang akurat dan peluang pengoptimalan yang ditargetkan.

- Menerapkan strategi penandaan komprehensif untuk armada untuk alokasi biaya, gambar untuk pelacakan aset, pembuat gambar untuk penunjukan lingkungan, dan tumpukan untuk pengelompokan organisasi.
- Gunakan Laporan AWS Biaya dan Penggunaan (AWS CUR) untuk memecah biaya WorkSpaces Aplikasi dengan sumber daya yang ditandai, dan menganalisis biaya per armada, tumpukan, dan gambar.
- Gunakan AWS Cost Exploreruntuk memvisualisasikan tren pengeluaran WorkSpaces Aplikasi, dan membandingkan biaya di berbagai dimensi seperti Wilayah dan jenis instans.
- Memantau dan menganalisis tingkat pemanfaatan armada, efisiensi jenis instans, dan jam streaming berdasarkan aplikasi.
- Lacak kapasitas cadangan yang tidak terpakai, armada atau tumpukan yang kurang dimanfaatkan, dan periode idle dalam penggunaan armada.
- Hitung dan lacak biaya per pengguna untuk setiap aplikasi, jam streaming per aplikasi, dan tarif adopsi pengguna untuk aplikasi streaming.
- Siapkan analisis penggunaan terperinci dengan mengonfigurasi laporan penggunaan WorkSpaces Aplikasi, menggunakan Amazon Athena untuk menanyakan data penggunaan, dan membuat visualisasi di Amazon Quick untuk wawasan biaya dan penggunaan.
- Mengevaluasi pertimbangan biaya total seperti lisensi Windows Server, model lisensi aplikasi, dan lisensi per pengguna dibandingkan dengan lisensi per perangkat.
- Gunakan Amazon Athena untuk menanyakan dan menganalisis biaya penyimpanan folder rumah dan pola penggunaan oleh pengguna. Untuk informasi selengkapnya, lihat posting AWS blog Cara melaporkan penggunaan folder beranda WorkSpaces Aplikasi Amazon dengan Amazon Athena.

Pilar keberlanjutan

Pilar keberlanjutan dari AWS Well-Architected Framework menekankan meminimalkan jejak lingkungan Anda dan mengoptimalkan penggunaan dan efisiensi energi. Ini memandu arsitek untuk membuat keputusan sadar lingkungan dalam desain sistem dan strategi alokasi sumber daya mereka.

Area fokus utama untuk menerapkan pilar ini ke lingkungan streaming WorkSpaces Aplikasi Anda:

- Memahami dan mengoptimalkan alokasi sumber daya agar sesuai dengan permintaan aktual dan meminimalkan pemborosan di lingkungan streaming
- Menganalisis dan beradaptasi dengan pola konsumsi pengguna untuk meningkatkan efisiensi pengiriman aplikasi dan sesi streaming
- Memilih dan menggunakan konfigurasi perangkat keras yang sesuai untuk memaksimalkan efisiensi energi sambil memenuhi persyaratan kinerja
- Menggunakan kemampuan layanan AWS terkelola untuk mendapatkan keuntungan dari skala ekonomi dan fitur efisiensi bawaan yang ditawarkan oleh layanan ini

Pahami dampak Anda

Pantau dan optimalkan dampak lingkungan beban kerja Anda dengan mengukur efisiensi sumber daya dan emisi per unit output. Gunakan data ini untuk membangun KPIs dan memandu peningkatan keberlanjutan.

- Pantau pola pemanfaatan armada.
- Lacak jam streaming per pengguna.
- Menganalisis tren penggunaan kapasitas armada.

Menetapkan tujuan keberlanjutan

Tetapkan tujuan keberlanjutan yang terukur untuk setiap beban kerja yang selaras dengan tujuan organisasi. Fokus pada pengurangan intensitas sumber daya per transaksi saat Anda menskalakan.

- Tetapkan target untuk tingkat pemanfaatan armada, efisiensi jenis instans, dan pengoptimalan jam streaming.

- Rencanakan kapasitas berdasarkan pola penggunaan aktual.

Maksimalkan pemanfaatan

Optimalkan efisiensi beban kerja dengan mengukur sumber daya yang tepat dan memaksimalkan pemanfaatan. Mengurangi kapasitas idle untuk meminimalkan konsumsi energi dan meningkatkan keberlanjutan.

- Konfigurasikan penskalaan otomatis agar sesuai dengan permintaan aktual.
- Kapasitas armada ukuran yang tepat berdasarkan pola penggunaan.
- Menerapkan batas kapasitas minimum dan maksimum yang sesuai.
- Pilih jenis instans yang sesuai untuk beban kerja.
- Pantau dan optimalkan kepadatan sesi streaming.
- Kurangi kapasitas idle selama jam-jam di luar sibuk.

Mengantisipasi dan mengadopsi penawaran perangkat keras dan perangkat lunak baru yang lebih efisien

Tetap mendapat informasi tentang, dan segera adopsi, teknologi efisien baru dari mitra dan pemasok untuk terus meningkatkan dampak lingkungan beban kerja Anda.

- Gunakan jenis instance generasi saat ini.
- Tingkatkan ke jenis instans yang lebih baru bila tersedia.
- Optimalkan pengaturan streaming aplikasi.
- Konfigurasikan protokol streaming yang sesuai.
- Perbarui ke fitur WorkSpaces Aplikasi terbaru.

Layanan terkelola yang digunakan

Manfaatkan layanan cloud bersama dan solusi terkelola untuk memaksimalkan efisiensi pemanfaatan sumber daya sekaligus meminimalkan dampak lingkungan melalui penskalaan otomatis dan manajemen siklus hidup.

- Gunakan Amazon S3 untuk penyimpanan pengguna untuk armada berbasis Windows dan Amazon EFS untuk sistem file bersama untuk armada berbasis Linux.
- Implementasikan CloudWatchuntuk pemantauan.
- Konfigurasikan IAM untuk manajemen akses.

Mengurangi dampak hilir beban kerja cloud Anda

Merancang layanan untuk meminimalkan kebutuhan sumber daya sisi klien, untuk mengurangi konsumsi energi dan memperpanjang masa pakai perangkat bagi pengguna.

- Sesuaikan durasi sesi maksimum untuk mencegah konsumsi sumber daya yang tidak perlu.
- Konfigurasikan batas waktu sesi yang sesuai.
- Tetapkan kebijakan batas waktu pemutusan sambungan.
- Menerapkan kebijakan persistensi sesi jika diperlukan.

Sumber daya

AWS dokumentasi

- AWS Well-ArchitectedKerangka
- Panduan Administrasi WorkSpaces Aplikasi Amazon
- Panduan CloudWatch Pengguna Amazon
- Panduan Pengguna Amazon EFSS
- Panduan Pengguna Amazon S3
- Panduan Pengguna IAM

AWS posting blog

- Penargetan WorkSpaces Aplikasi Berbasis Keanggotaan Grup Direktori Aktif
- Buat Penyedia Identitas Tunggal untuk semua Tumpukan WorkSpaces Aplikasi Amazon Anda dengan Azure AD
- Mengkonfigurasi Bantuan Jarak Jauh Windows untuk Aplikasi Amazon WorkSpaces dan Amazon WorkSpaces
- Membuat data AS2TrustedDomains DNS TXT untuk mengarahkan klien asli Aplikasi WorkSpaces Amazon ke penyedia identitas pihak ketiga
- Membuat pencatatan khusus dan CloudWatch peringatan Amazon di Aplikasi Amazon WorkSpaces
- Cross-Region pengalihan dengan Aplikasi Geo Targetly dan Amazon WorkSpaces
- Cross-account sumber daya dan WorkSpaces Aplikasi Amazon
- Aktifkan federasi dengan Bio-key PortalGuard dan WorkSpaces Aplikasi Amazon
- Mengaktifkan Federasi dengan Aplikasi SimpleSamlPHP dan Amazon WorkSpaces
- Mengaktifkan federasi identitas dengan Aplikasi Duo Single dan Sign-On Amazon WorkSpaces
- Mengaktifkan Federasi Identitas dengan Aplikasi Shibboleth dan Amazon WorkSpaces
- Strategi failover untuk VDI lokal dengan Komputasi Pengguna Akhir Amazon
- Bagaimana Amazon Menggunakan WorkSpaces Aplikasi Amazon untuk Memberikan Ilmuwan Data dan Analis Akses ke Data Sensitif

- Cara mengonfigurasi otentikasi berbasis sertifikat untuk Aplikasi Amazon WorkSpaces
- Cara menggunakan klaim Okta dengan hak aplikasi untuk Aplikasi Amazon WorkSpaces
- Mengelola Lab Komputer di WorkSpaces Aplikasi Amazon dengan Manajemen Aplikasi Virtual Sumber Terbuka
- Metode mengalokasikan biaya WorkSpaces Aplikasi Anda ke unit bisnis Anda
- Memantau WorkSpaces Aplikasi Amazon dengan OpenSearch Layanan Amazon dan Amazon Kinesis Data Firehose
- Pemisahan Jaringan dan Sanitasi Data menggunakan Amazon, WorkSpaces Aplikasi WorkSpaces Amazon, dan Amazon Macie
- OneLogin SSO dengan Aplikasi Amazon WorkSpaces
- Optimalkan jalur audio panggilan Connect Customer Anda dengan WorkSpaces Aplikasi Amazon
- Penyimpanan persisten untuk WorkSpaces Aplikasi Amazon Armada Linux di Amazon Elastic File System
- Arahkan ulang aplikasi Okta SAMP ke klien asli Aplikasi Amazon WorkSpaces
- Sederhanakan manajemen gambar WorkSpaces Aplikasi Amazon dengan Masking Aplikasi
- Streaming aplikasi dengan biaya lebih rendah dengan armada Amazon WorkSpaces Applications Elastic dan kompatibilitas Linux
- Streaming dari titik akhir VPC antarmuka untuk lingkungan yang Diatur dengan Aplikasi WorkSpaces
- Menggunakan hak WorkSpaces aplikasi Aplikasi Amazon dengan Azure AD
- Reporter Masalah Pengguna untuk Aplikasi Amazon WorkSpaces
- Menggunakan hak WorkSpaces aplikasi Aplikasi Amazon dengan Google Workspace
- Menggunakan Auth0 dengan Microsoft Active Directory di Aplikasi Amazon WorkSpaces
- Menggunakan Microsoft AppLocker untuk mengelola pengalaman aplikasi di WorkSpaces Aplikasi Amazon
- Menggunakan Python untuk memberi daya pada GUI Asisten Pencitraan Linux WorkSpaces Aplikasi
- Opsi pengalihan aplikasi web untuk Klien WorkSpaces Aplikasi

Riwayat dokumen

Tabel berikut menjelaskan perubahan signifikan pada panduan ini. Jika Anda ingin diberi tahu tentang pembaruan masa depan, Anda dapat berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Publikasi awal	—	Juli 23, 2025

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