Membangun strategi untuk single, hybrid, dan multicloud dalam pendidikan

AWS Bimbingan Preskriptif

AWS Bimbingan Preskriptif: Membangun strategi untuk single, hybrid, dan multicloud dalam pendidikan

Table of Contents

Membangun strategi untuk single, hybrid, dan multicloud dalam pendidikan

Amazon Web Services (kontributor)

September 2023 (riwayat dokumen)

Institusi pendidikan berusaha untuk mendukung fungsi-fungsi seperti pembelajaran jarak jauh, penelitian, pengalaman siswa, wawasan data, dan administrasi dengan kelincahan, penghematan biaya, keamanan, dan ketahanan yang ditawarkan komputasi awan. Banyak organisasi menilai penyebaran hybrid dan multicloud sebagai bagian dari transformasi digital ini.

Paper ini memberikan panduan preskriptif tentang pembuatan teknologi dan strategi tata kelola tunggal, hibrida, dan multicloud, untuk para pemimpin eksekutif dan pembuat keputusan di lembaga pendidikan yang mengevaluasi opsi cloud mereka. Panduan ini didasarkan pada pengalaman kami dalam AWS bekerja dengan lebih dari 14.000 lembaga pendidikan dari semua ukuran di seluruh dunia—dari sekolah dasar dan menengah hingga pendidikan tinggi.

Ikhtisar

Ketika lembaga pendidikan bertransformasi secara digital untuk memberikan layanan dan pengalaman yang berbeda kepada siswa, orang tua, fakultas, staf, dan komunitas mereka, mereka menghadapi banyak keputusan teknis. Banyak organisasi telah membuat keputusan untuk mengadopsi cloud untuk meningkatkan kelincahan, elastisitas, ketahanan, keamanan, dan penghematan biaya. Berdasarkan hubungan dan investasi mereka yang ada di berbagai tim, sebagian besar organisasi menggunakan beberapa kombinasi pusat data lokal, fasilitas kolokasi, dan penyedia cloud. Mengingat ketersediaan beberapa opsi cloud, lembaga pendidikan harus sering memutuskan dari model penyebaran tunggal, hibrida, dan multicloud (didefinisikan di bagian Strategi penyebaran awan).

Multicloud, yang merupakan penggunaan layanan dari setidaknya dua penyedia layanan cloud, tidak jarang bagi banyak institusi saat ini. Tim TI Anda mungkin lebih memilih satu penyedia cloud, sedangkan grup, departemen, atau pengguna individu lain mungkin memilih atau sudah menggunakan penyedia alternatif. Institusi pendidikan yang tidak memiliki strategi yang jelas untuk memandu mereka ke model penyebaran cloud yang tepat menghadapi banyak tantangan. Ini termasuk kompleksitas yang tidak perlu, meningkatnya tuntutan staf, tata kelola yang tidak konsisten,

dan pendekatan common denominator terendah yang membatasi mereka pada subset kemampuan dasar yang umum di seluruh penyedia. Setiap tantangan menghambat inovasi dan memperlambat transformasi digital.

Sebaliknya, jika Anda memiliki strategi cloud yang memandu Anda untuk menggunakan single, hybrid, dan multicloud, Anda dapat memenuhi persyaratan misi pendidikan Anda sambil menyadari manfaat cloud dengan cara yang berkelanjutan secara operasional untuk kesuksesan jangka panjang. Untuk membuat strategi ini, kami merekomendasikan yang berikut:

- Pilih penyedia cloud utama dan strategis.
- Mendirikan Cloud Center of Excellence (CCoE).
- Membedakan antara aplikasi perangkat lunak sebagai layanan (SaaS) dan layanan cloud dasar.
- Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud.
- Mengadopsi solusi cloud-native, terkelola sedapat mungkin dan praktis.
- Menerapkan arsitektur hybrid ketika investasi lokal yang ada memberi insentif kepada penggunaan berkelanjutan.
- Cadangan multicloud hanya untuk beban kerja yang tidak dapat memenuhi persyaratan teknis atau bisnis melalui satu penyedia cloud.

Praktik terbaik ini dibahas secara rinci di bagian Rekomendasi dari paper ini. Setiap rekomendasi penting, tetapi prioritas institusi Anda akan tergantung pada tahap adopsi cloud-nya. Misalnya, jika Anda baru memulai adopsi cloud, fokuslah untuk memilih penyedia cloud strategis utama, membuat CCo E, dan mengadopsi solusi terkelola cloud-native. Jika Anda sudah menggunakan penyedia cloud tunggal, fokuslah untuk menetapkan persyaratan keamanan dan tata kelola inti, dan pertimbangkan arsitektur hybrid ketika investasi pusat data Anda yang ada memberi insentif untuk penggunaan berkelanjutan. Jika organisasi Anda sudah menggunakan beberapa penyedia cloud, fokuslah untuk membedakan aplikasi SaaS dan memesan penerapan multicloud ke beban kerja langka yang benar-benar membutuhkannya.

Daftar Isi

Strategi penyebaran cloud

AWS mendefinisikan komputasi awan sebagai pengiriman on-demand sumber daya TI melalui internet dengan pay-as-you-go harga. Alih-alih membeli, memiliki, dan memelihara pusat data fisik dan server, Anda dapat mengakses layanan teknologi, seperti daya komputasi, penyimpanan, dan database, sesuai kebutuhan dari penyedia cloud. Cloud computing memungkinkan institusi pendidikan untuk menghindari pengangkatan berat yang tidak berdiferensiasi seperti pengadaan perangkat keras, pemeliharaan, dan perencanaan kapasitas. Saat Anda mengadopsi dan menerapkan solusi cloud, Anda dapat memilih dari beberapa model: cloud tunggal, cloud hybrid, dan multicloud.

Awan tunggal

Model ini hanya menggunakan satu penyedia layanan cloud. Aplikasi dan beban kerja cloud tunggal dapat diimplementasikan secara langsung di cloud, atau sebelumnya dihosting di lingkungan lain dan dimigrasikan ke cloud. Beban kerja ini mungkin menggunakan layanan infrastruktur tingkat rendah dari penyedia cloud mereka atau juga memanfaatkan layanan terkelola tingkat tinggi. Terlepas dari itu, model ini mengadopsi penyedia cloud tunggal dan hanya menggunakan layanan cloud dari penyedia itu.

Awan hibrida

Model cloud hybrid mendistribusikan sumber daya di seluruh pusat data lokal organisasi dan setidaknya satu penyedia layanan cloud. Biasanya, tujuan dari model ini adalah untuk memperluas infrastruktur organisasi ke cloud sambil mempertahankan konektivitas pribadi dengan sistem internal yang ada yang berada di tempat.

Multicloud

Model multicloud mendistribusikan sumber daya di seluruh, dan menggunakan layanan dari, setidaknya dua penyedia layanan cloud. Sebuah organisasi mungkin memilih untuk menjadi multicloud, tetapi lebih sering ini adalah hasil yang tidak disengaja dari masing-masing tim, departemen, atau anggota staf yang memiliki preferensi mereka sendiri untuk penyedia cloud yang berbeda.

Rekomendasi

Sekarang setelah Anda memiliki pemahaman dasar tentang cloud tunggal, cloud hybrid, dan multicloud, bagian ini memberikan rekomendasi terperinci untuk memilih model.

- Pilih penyedia cloud utama dan strategis
- Menetapkan CCo E
- Membedakan antara aplikasi SaaS dan layanan cloud dasar
- Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud
- Mengadopsi layanan cloud-native, terkelola sedapat mungkin dan praktis
- Menerapkan arsitektur hibrida ketika ada, investasi lokal memberi insentif untuk penggunaan berkelanjutan
- Cadangan multicloud hanya untuk beban kerja yang tidak dapat memenuhi persyaratan teknis atau bisnis mereka melalui satu penyedia cloud

Pilih penyedia cloud utama dan strategis

Adopsi cloud memberikan banyak manfaat yang penting untuk modernisasi TI, efektivitas biaya, dan inovasi. Namun, mengadopsi teknologi cloud di luar aplikasi SaaS terbatas dapat menimbulkan tantangan yang harus direncanakan oleh lembaga pendidikan dengan hati-hati untuk menghindari biaya dan kompleksitas yang tidak perlu. Perubahan teknologi dan bisnis yang terlibat dalam penerapan beban kerja di cloud memerlukan pemberdayaan staf dan penyesuaian infrastruktur inti, termasuk jaringan, keamanan, tata kelola, dan operasi.

Pendekatan terbaik untuk mengatasi tantangan ini secara efektif, terutama jika organisasi Anda berada pada tahap awal perjalanan cloud-nya, adalah memilih penyedia cloud strategis utama untuk mendukung sebagian besar beban kerja Anda. Mulailah dengan adopsi terfokus yang berpusat pada penyedia itu sehingga Anda dapat menyederhanakan dan mempercepat realisasi manfaat cloud. Memilih penyedia cloud utama bukanlah keputusan eksklusif dan tidak dapat diubah. Ini memungkinkan organisasi Anda untuk mengembangkan adopsi cloud Anda secara berulang. Anda dapat mulai dengan berfokus pada beberapa layanan dan kemudian memperluas ke layanan cloud lainnya sesuai kebutuhan, tanpa menunda manfaat cloud secara keseluruhan. Pendekatan ini memaksimalkan kemampuan organisasi Anda untuk memanfaatkan kemampuan penyedia, berkonsentrasi dan mengembangkan keterampilan karyawan dan hubungan mitra pihak ketiga, dan menyederhanakan manajemen vendor.

Kami telah melihat pelanggan memulai perjalanan cloud mereka dengan mencoba secara bersamaan mengadopsi beberapa penyedia cloud tetapi kemudian menyesali keputusan itu dan kompleksitas yang diperkenalkannya. Gartner membagikan wawasan ini dalam artikel mereka, 6 Langkah untuk Merencanakan Strategi Cloud, di mana langkah 2 adalah "Prioritaskan penyedia utama dalam arsitektur multicloud."

Setiap penyedia cloud memperkenalkan model operasi dan dukungan yang berbeda, manajemen identitas dan akses, jaringan, operasi, kemampuan kepatuhan, dan banyak lagi. Lebih baik menguasai satu model operasi penyedia cloud sekaligus.Anda kemudian dapat menggabungkan layanan cloud tambahan secara berulang dan bertahap, jika dirasionalisasi. Banyak faktor yang dapat memengaruhi keputusan Anda untuk mengadopsi penyedia cloud utama, tetapi gunakan pertanyaan kunci berikut untuk memandu pilihan Anda.

- Berapa luas dan kedalaman layanan yang ditawarkan penyedia?

  Penyedia cloud yang berbeda menawarkan layanan yang berbeda. Minimal, pastikan bahwa penyedia utama Anda memiliki kemampuan yang diperlukan untuk mendukung semua persyaratan fungsional Anda serta kebutuhan operasional lintas sektoral Anda seperti keamanan, tata kelola, dan otomatisasi. Pilih penyedia yang memberikan kemampuan ini dengan rekam jejak inovasi dan keunggulan operasional yang terbukti. Pertimbangkan tidak hanya aplikasi Anda, tetapi juga data Anda. Pikirkan tentang integrasi data masa depan dan pola transfer untuk membatasi biaya, latensi, dan kompleksitas pemindahan data dalam jumlah besar antar penyedia. Pilih penyedia yang memiliki luas dan kedalaman layanan sebesar mungkin untuk memenuhi kebutuhan aplikasi dan data Anda saat ini, dan juga untuk membuka kasus penggunaan baru yang dapat memenuhi kebutuhan institusi Anda saat mereka berubah seiring waktu.
- Dapatkah penyedia mendukung semua kebutuhan keamanan dan kepatuhan Anda?

  Dalam pendidikan, keamanan dan kepatuhan sangat penting untuk penyebaran teknologi apa pun. Pilih penyedia cloud yang mampu memenuhi semua kebutuhan keamanan dan kepatuhan Anda. Alat seperti AWS Artifactdapat membantu Anda mengevaluasi penyedia dengan menawarkan sumber daya pusat untuk akses sesuai permintaan ke laporan keamanan dan kepatuhan. Pertimbangkan tidak hanya keamanan dan kepatuhan infrastruktur dan layanan penyedia cloud itu sendiri, tetapi juga betapa mudahnya bagi Anda untuk merancang solusi yang aman dan sesuai dengan menggunakan layanan tersebut. Pilih penyedia yang menawarkan beberapa kombinasi solusi bawaan, mulai cepat, dan panduan preskriptif untuk mempercepat adopsi cloud Anda yang aman.
- Apakah penyedia memiliki jaringan mitra yang kuat?

Tidak ada organisasi yang mengalami transformasi cloud sendirian. Untuk mempercepat adopsi, Anda harus menggunakan layanan dan keahlian penyedia cloud serta jaringan mitra mereka. Jaringan ini mencakup mitra teknologi yang menyediakan perangkat lunak yang berjalan, terintegrasi dengan, atau mendukung teknologi cloud, serta mitra konsultasi yang dapat membantu Anda merancang, membangun, menjalankan, dan mengelola aplikasi Anda sendiri di cloud. Anda akan menemukan bahwa banyak penyedia teknologi pendidikan, vendor perangkat lunak independen (ISVs), konsultan, dan reseller yang sudah bekerja dengan Anda adalah anggota jaringan mitra penyedia cloud. Lebih suka penyedia cloud yang memiliki jaringan mitra paling kuat dengan kompetensi yang diperiksa. Memiliki mitra dengan industri yang terbukti dan keahlian teknis sangat penting.

- Dukungan dan pemberdayaan apa yang ditawarkan penyedia?

  Agar berhasil mengadopsi teknologi baru apa pun, Anda memerlukan mekanisme untuk meminta pelatihan dan bantuan, termasuk rekomendasi praktik terbaik, panduan konfigurasi, dan penyelesaian masalah break-fix. Memilih penyedia cloud yang menawarkan opsi dukungan dan pelatihan yang kuat akan membuat Anda siap untuk sukses. Jelajahi model dan sumber daya dukungan resmi penyedia serta sumber daya pihak ketiga atau berbasis komunitas yang tersedia seperti blog, forum, video, dan panduan cara. Pertimbangkan tidak hanya program dukungan teknis penyedia, tetapi juga program yang berfokus pada transformasi bisnis dan budaya. Misalnya, AWS Cloud Adoption Framework (AWS CAF) membantu organisasi bertransformasi secara digital dengan berfokus pada perspektif yang mencakup proses bisnis dan orang, bukan hanya teknologi. Lebih suka penyedia cloud yang menawarkan opsi pelatihan ekstensif dan model dukungan dan komunitas yang terbukti dan andal.

Menetapkan CCo E

Pertimbangkan untuk mengembangkan fungsi kepemimpinan cloud Anda melalui kantor transformasi atau Cloud Center of Excellence (CCoE). A CCo E mengembangkan dan menginjili pendekatan untuk menerapkan teknologi cloud dalam skala besar di seluruh organisasi. Untuk adopsi cloud yang sukses, rancang CCo E Anda untuk menyertakan perwakilan yang dapat berbicara untuk tim dan departemen yang terlibat. Mulailah dari yang kecil dan kembangkan CCo E secara bertahap untuk memenuhi kebutuhan Anda saat Anda maju melalui perjalanan transformasi. Perwakilan penyedia cloud utama Anda, seperti manajer AWS akun dan arsitek solusi Anda, dapat menyediakan sumber daya untuk memandu Anda melalui pembuatan CCo E. A CCo E mempercepat kemampuan Anda untuk membangun keahlian materi pelajaran, mencapai pembelian, mendapatkan kepercayaan di

seluruh organisasi Anda, dan menetapkan pedoman yang efektif untuk memenuhi persyaratan misi Anda. Tidak ada struktur organisasi tunggal yang berfungsi untuk setiap institusi, tetapi pertanyaan-pertanyaan berikut akan membantu Anda merancang CCo E Anda sendiri.

- Siapa yang harus Anda sertakan dalam CCo E Anda?

  Pada awalnya, CCo E mungkin hanya mencakup segelintir pengadopsi awal dan juara cloud. CCoE mungkin tetap kecil, tetapi harus berkembang untuk memasukkan juara yang dapat berbicara untuk fungsi bisnis dan fungsi teknis yang dipengaruhi oleh adopsi cloud. Fungsi bisnis meliputi manajemen perubahan, persyaratan pemangku kepentingan, tata kelola, pelatihan, pengadaan, dan komunikasi. Fungsi-fungsi ini biasanya diwakili oleh anggota tim administrasi dan instruksional institusi Anda. Fungsi teknis meliputi infrastruktur, otomatisasi, alat operasional, keamanan, kinerja, dan ketersediaan. Fungsi-fungsi ini biasanya diwakili oleh anggota tim TI institusi Anda. CCoE juga harus berusaha melibatkan vendor dan mitra, jika perlu, untuk memberikan keahlian materi pelajaran. CCoE adalah organisasi yang hidup. Keanggotaan, bentuk, dan fungsinya kemungkinan akan berubah seiring waktu, dan bahkan mungkin bubar di beberapa titik kematangan masa depan.
- Bagaimana CCo E berinteraksi dengan para pemangku kepentingannya?

  CCoE melayani tim lain dan dimaksudkan hanya untuk menginformasikan dan memungkinkan adopsi cloud yang sukses. Lihatlah menyematkan bagian CCo E di berbagai departemen, sekolah, dan fungsi. Ini memungkinkan akses ke sumber daya yang lebih luas dan umpan balik internal yang lebih cepat. Fokus pada membangun kemitraan dan membuka jalur komunikasi antar pemangku kepentingan sejak dini untuk membangun kepercayaan dalam institusi dan memecah silo organisasi. CCoE harus memiliki mekanisme yang ditentukan untuk berkomunikasi dengan pemangku kepentingan, mengumpulkan umpan balik, dan melatih pengguna. Metrik keberhasilan CCo E harus mencerminkan kolaborasi dan komunikasi tersebut. Jika sebuah tim diukur hanya pada teknologi bangunan, lebih banyak teknologi akan dibangun, tetapi penggunaan dan hasilnya akan menjadi renungan. Metrik Anda seharusnya mengukur hal-hal seperti jumlah tim yang menjadi mandiri melalui pekerjaan CCo E, berapa kali CCo E berada di jalur kritis untuk inisiatif, jumlah acara pelatihan yang diadakan, atau luasnya adopsi output E. CCo CCoE yang dibangun dengan baik dan tepercaya dapat menjadi batu loncatan menuju transformasi organisasi yang lebih besar yang dibangun di atas kepercayaan.
- Bagaimana seharusnya Anda menetapkan CCo E?

Sebagian besar organisasi memulai adopsi cloud mereka dengan proyek percontohan yang spesifik dan ditargetkan. Menetapkan CCo E sebagai bagian dari proyek-proyek ini. Awal yang baik sangat penting dalam mendefinisikan keberhasilan seluruh perjalanan.

- Mulailah dengan masalah bisnis. Teknologi demi teknologi adalah strategi yang buruk. Jika Anda bereksperimen dengan teknologi cloud, identifikasi kasus penggunaan bisnis yang menarik tidak peduli seberapa kecil kelihatannya. Kemudian, kembalilah dari kasus penggunaan itu untuk menetapkan tujuan yang jelas tentang bagaimana teknologi dapat membantu. Jangan menerapkan solusi dalam silo. Ambil masukan konstan dari pemangku kepentingan bisnis sebelum dan selama implementasi proyek. Semua proyek cloud yang sukses bergantung pada kolaborasi erat dengan unit institusional yang akan menggunakan teknologi tersebut.
- Mulai dari yang kecil. Pilih proyek berisiko rendah yang menyediakan pintu dua arah. Ini berarti bahwa proyek tersebut dapat dibalik dan kesalahan apa pun dapat diperbaiki dengan cepat. Proyek percontohan adalah tentang eksperimen. Menghindari proyek berskala besar dan berisiko tinggi memberi Anda kontrol yang lebih baik atas implementasi dan hasil. Ini membantu untuk menargetkan masalah yang spesifik dan dapat ditentukan alih-alih tujuan berbasis luas. Misalnya, jika otomatisasi adalah tujuan akhir, bertujuan untuk mengotomatiskan tugas tertentu alih-alih seluruh pekerjaan.
- Tentukan dan ukur hasilnya. Tetapkan metrik yang jelas untuk menilai kemajuan dan kinerja setiap proyek. Tentukan keadaan akhir yang diinginkan jauh sebelumnya untuk menghindari harapan yang tidak cocok di antara para pemangku kepentingan. Bekerja sama dengan pemangku kepentingan bisnis dan pemimpin lain dalam organisasi untuk menentukan harapan dan keuntungan yang terukur. Penting juga untuk menerjemahkan hasilnya ke dalam bahasa non-teknis. Bicara dalam hal tujuan kelembagaan, seperti bagaimana proyek meningkatkan retensi dan mengurangi churn, bagaimana hal itu menurunkan biaya dan meningkatkan kecepatan pengiriman, dan sebagainya.
- Mulai dari zona nyaman. Pilih proyek dalam domain yang akrab dengan institusi Anda. Dengan cara ini Anda dapat memastikan bahwa proyek memiliki tujuan yang bermakna dan dapat dimengerti dengan dampak nyata. Proyek semacam itu akan membangun kepercayaan diri dan memiliki hasil jangka panjang yang lebih besar untuk organisasi Anda. Misalnya, jika Anda sudah memiliki keahlian dalam analitik data, Anda dapat memulai perjalanan cloud Anda sambil memanfaatkan keahlian yang ada dengan memulai dengan proyek analitik. Setiap institusi memiliki keahlian dan kebutuhan yang berbeda untuk menemukan komponen uniknya untuk menyusun strategi transformasi digital yang sukses.

Membedakan antara aplikasi SaaS dan layanan cloud dasar

Sebagian besar lembaga pendidikan telah mengadopsi aplikasi perangkat lunak sebagai layanan (SaaS). SaaS memberi institusi Anda solusi lengkap yang dijalankan dan dikelola oleh penyedia layanan. Aplikasi SaaS umum termasuk aplikasi produktivitas seperti pengolah kata dan email, tetapi opsi SaaS juga ada untuk banyak beban kerja mission-critical seperti perencanaan sumber daya perusahaan (ERP), sistem informasi siswa (SIS), dan sistem manajemen pembelajaran (LMS). Ketika institusi Anda mengadopsi penawaran SaaS, tim TI Anda tidak perlu memikirkan bagaimana layanan dipertahankan atau bagaimana infrastruktur dikelola — pengguna Anda hanya menggunakan layanan tersebut. Model pengiriman ini mengurangi beban manajemen pada staf TI Anda. Banyak institusi memilih untuk mengadopsi pendekatan “SaaS pertama” dalam strategi TI mereka, terutama jika tim TI mereka tidak memiliki waktu, sumber daya, atau keahlian untuk meng-host aplikasi yang sama secara memadai. Bahkan jika Anda memiliki sumber daya untuk menjadi tuan rumah sendiri, mungkin masih lebih hemat biaya untuk mengadopsi solusi SaaS dan berinvestasi dalam proyek lain sebagai gantinya.

Saat Anda menggunakan aplikasi SaaS, tim TI Anda tidak perlu mengelola infrastruktur yang mendasarinya, sehingga tempat vendor meng-host aplikasi (pusat data lokal, penyedia cloud utama, atau penyedia cloud alternatif) menjadi kurang penting. Setelah memilih penyedia cloud utama dan strategis, Anda dapat memilih untuk menggunakan penawaran SaaS yang dihosting di penyedia cloud lain atau di lokasi, di pusat data vendor. Sebaliknya, bahkan jika aplikasi SaaS Anda di-host di satu penyedia cloud, Anda dapat memilih penyedia cloud strategis utama yang berbeda berdasarkan kekuatan penyedia itu untuk beban kerja non-SaaS Anda. Perbedaan antara lingkungan hosting kurang penting untuk SaaS daripada untuk aplikasi yang di-host sendiri. Namun, Anda tetap harus mempertimbangkan pertanyaan kunci berikut saat mengevaluasi bagaimana SaaS cocok dengan cloud sebagai bagian dari strategi TI Anda.

- Apakah aplikasi SaaS sangat tersedia dan dapat diskalakan?

  Banyak vendor telah membuat keputusan untuk mengadopsi cloud untuk penawaran SaaS mereka. Dengan demikian, vendor mampu mencapai manfaat cloud dari peningkatan ketersediaan dan skalabilitas. Selain itu, karena vendor dapat mengadopsi model tanggung jawab bersama cloud alih-alih mengelola dan memelihara infrastruktur fisik, mereka dapat menginvestasikan lebih banyak waktu dan sumber daya ke dalam pengiriman fitur baru. Karena manfaat ini, Anda harus memilih penyedia yang mengutamakan cloud dan menawarkan solusi yang dihosting di cloud.
- Dapatkah aplikasi SaaS memenuhi persyaratan keamanan Anda?

Saat mengevaluasi SaaS, penting untuk mengetahui data apa yang disimpan aplikasi, bagaimana data itu digunakan, dan kontrol keamanan mana yang ada untuk melindungi data tersebut. Meskipun Anda mungkin tidak memiliki kontrol langsung atas penyimpanan data seperti yang Anda lakukan di lingkungan Anda sendiri yang dihosting sendiri, Anda harus memastikan bahwa vendor memiliki mekanisme dan kontrol untuk menangani data Anda dengan tepat. Waspadai fitur keamanan mana yang ada di dalam solusi SaaS dan fitur mana yang memerlukan konfigurasi tambahan. Cloud memungkinkan penyedia SaaS untuk membangun solusi yang lebih tersedia dan terukur, dan mereka juga dapat membangun solusi yang lebih aman karena model tanggung jawab bersama. Anda harus memilih penyedia yang memanfaatkan alat dan layanan keamanan cloud sebagai bagian dari solusi mereka.

- Siapa yang memiliki data aplikasi SaaS dan bagaimana Anda bisa mengaksesnya?

  Saat Anda menggunakan SaaS, Anda mempercayai penyedia untuk menangani data institusi Anda dengan benar. Pastikan untuk meninjau persyaratan layanan dan perjanjian tingkat layanan untuk aplikasi SaaS untuk memahami faktor-faktor yang berkontribusi seperti kepemilikan data, ketersediaan, dan daya tahan. Evaluasi mekanisme untuk mencadangkan atau mengekspor data Anda; ini sangat penting jika Anda memutuskan untuk beralih penyedia atau penyedia menghentikan layanan.

- Dapatkah layanan Anda yang lain dan aplikasi yang dihosting sendiri terintegrasi dengan aplikasi SaaS, terlepas dari lingkungannya?

  Saat mengadopsi solusi SaaS, mudah untuk mengasumsikan bahwa layanan dan aplikasi yang berbagi lingkungan hosting yang sama (yaitu, aplikasi yang menggunakan penyedia cloud yang sama atau pusat data vendor yang sama) akan memiliki integrasi yang lebih mulus. Namun, sebagian besar solusi SaaS saat ini memiliki dukungan luas untuk API dan integrasi pihak ketiga, jadi jangan membatasi diri Anda pada solusi yang di-host di lingkungan yang sama. Jika integrasi yang diperlukan ada, solusi tidak harus berbagi lingkungan dasar yang sama. Misalnya, Anda menggunakan solusi SaaS seperti Google Drive atau Microsoft OneDrive untuk penyimpanan file pelajar berbasis cloud. Untuk menyediakan desktop virtual dan streaming aplikasi kepada siswa Anda, Anda dapat menentukan bahwa WorkSpaces Aplikasi Amazon paling sesuai dengan kebutuhan Anda. Meskipun layanan ini berjalan di lingkungan yang berbeda, WorkSpaces Aplikasi memiliki integrasi asli dengan Google Drive dan Microsoft OneDrive, sehingga siswa Anda dapat terus menggunakan penyimpanan yang ada.

- Apakah aplikasi SaaS mendukung manajemen identitas terpusat?

Untuk mencegah tim TI Anda dari keharusan mengelola toko identitas yang berbeda dan pengguna Anda dari keharusan mengingat beberapa set kredensional, pastikan bahwa solusi SaaS Anda mendukung integrasi dengan manajemen identitas yang ada atau solusi masuk tunggal. Manajemen identitas yang terfragmentasi mengurangi produktivitas dan dapat menyebabkan praktik keamanan yang buruk seperti creep hak istimewa dan kata sandi yang lemah. Jika solusi SaaS yang Anda inginkan tidak mendukung sistem masuk tunggal atau toko identitas Anda yang ada, evaluasi apakah nilai bisnis dari mengadopsi solusi melebihi beban yang meningkat pada pengguna dan staf.

- Bagaimana Anda bisa mengamankan komunikasi jaringan dengan aplikasi SaaS?

  Dalam beberapa kasus, Anda mungkin memerlukan aplikasi yang di-host sendiri untuk berkomunikasi dengan aplikasi SaaS. Biasanya, komunikasi ini akan melalui APIs yang diamankan dengan mekanisme otentikasi dan otorisasi yang sesuai. Namun, tergantung pada lingkungan hosting dari dua aplikasi, mekanisme alternatif atau tambahan mungkin diperlukan untuk menyederhanakan atau mengamankan komunikasi itu. Misalnya, jika Anda meng-host sendiri aplikasi dengan penyedia cloud dan perlu mengintegrasikannya dengan aplikasi SaaS yang di-host di penyedia cloud yang sama, vendor mungkin menyediakan beberapa opsi koneksi. Anda mungkin dapat menggunakan koneksi peering khusus cloud, antarmuka pribadi APIs, atau pribadi seperti AWS PrivateLinkuntuk mencegah komunikasi tersebut melintasi internet publik. Demikian pula, jika aplikasi lokal Anda memiliki koneksi jaringan khusus ke penyedia cloud melalui layanan seperti AWS Direct Connect, Anda dapat menggunakan koneksi yang sama untuk berkomunikasi dengan aplikasi SaaS yang di-host di penyedia cloud yang sama.

Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud

Institusi pendidikan memiliki berbagai tujuan kepatuhan, tata kelola, dan keamanan siber yang harus mereka capai. Risiko gagal memenuhi tujuan ini dapat mencakup hilangnya reputasi institusional, denda moneter, tebusan, pelanggaran data sensitif, pencurian kekayaan intelektual, dan hilangnya fungsi-fungsi penting misi yang terdegradasi atau lengkap. Karena model tanggung jawab bersama, institusi yang mengadopsi layanan cloud dapat mengurangi beban administrasi dengan melepaskan beberapa tanggung jawab untuk keamanan infrastruktur ke penyedia layanan cloud. Selain itu, Anda dapat memperoleh manfaat dari layanan keamanan cloud-native yang dibuat khusus yang menawarkan fitur yang seringkali tidak tersedia, sulit dikelola, atau mahal biaya dalam penerapan lokal. Contohnya termasuk layanan seperti AWS WAFuntuk perlindungan aplikasi web, AWS

Shielduntuk perlindungan penolakan layanan (DDoS) terdistribusi, dan Amazon GuardDuty untuk deteksi ancaman. Strategi keamanan dan tata kelola cloud yang sukses memungkinkan tim TI dan keamanan untuk fokus pada pembangunan sistem yang aman dengan desain, membantu institusi beradaptasi dengan cepat terhadap persyaratan misi yang berkembang, dan menyediakan lingkungan yang aman bagi fakultas dan peneliti untuk pembelajaran dan inovasi yang inovatif. Untuk mengevaluasi persyaratan keamanan dan tata kelola Anda, pertimbangkan pertanyaan-pertanyaan kunci berikut.

- Kerangka kerja kepatuhan mana yang harus disejajarkan dengan beban kerja Anda?

  Lembaga pendidikan harus mematuhi banyak kerangka kepatuhan karena banyaknya pemangku kepentingan dan beban kerja yang mereka dukung. Kerangka kerja kepatuhan ini termasuk Undang-Undang Hak dan Privasi Pendidikan Keluarga (FERPA), Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan (HIPAA), Program Manajemen Risiko dan Otorisasi Federal (FedRAMP), Sertifikasi Model Kematangan Keamanan Siber (CMMC), Peraturan Lalu Lintas Internasional dalam Senjata (ITAR), Layanan Informasi Peradilan Pidana (CJIS), dan Standar Keamanan Data Industri Kartu Pembayaran (PCI) DSS). Dalam beberapa kasus, seperti dengan CMMC, dana hibah penelitian tidak dirilis sampai beban kerja yang relevan disertifikasi sesuai. Setiap kerangka kerja unik dan mungkin hanya berlaku untuk subset beban kerja. Pastikan Anda tahu beban kerja mana yang harus mematuhi persyaratan mana dan bahwa Anda dapat mencapai persyaratan tersebut di setiap lingkungan beban kerja. Di lingkungan cloud, pastikan Anda memahami tanggung jawab Anda dibandingkan dengan tanggung jawab penyedia cloud. Anda harus memiliki pengetahuan, sumber daya, dan keahlian yang diperlukan untuk mencapai dan mempertahankan kepatuhan.
- Mekanisme apa yang Anda miliki untuk menegakkan kepatuhan di beberapa penyedia cloud tanpa menghambat inovasi?

  Jika institusi akademis Anda baru mengenal cloud, kami sarankan Anda memilih satu penyedia layanan cloud strategis utama dan fokus pada pemahaman bagaimana merancang, merekayasa, dan mengoperasikan lingkungan cloud yang aman menurut desain. Idealnya, kontrol keamanan yang secara otomatis tertanam dalam sistem swalayan memungkinkan pengguna untuk dengan cepat menyebarkan lingkungan cloud yang aman dengan jumlah intervensi minimum dari tim TI. Berfokus pada satu penyedia membatasi jumlah sumber daya dan waktu yang harus Anda investasikan untuk memastikan keamanan dan kepatuhan. Institusi yang paling sukses memilih penyedia layanan cloud yang dapat mendukung sebagian besar persyaratan kepatuhan, memiliki jaringan mitra yang kuat, menawarkan solusi kepatuhan bawaan, dan menyediakan otomatisasi layanan mandiri yang aman. Jika Anda harus memastikan keamanan dan kepatuhan

di beberapa penyedia cloud, investasi tambahan akan diperlukan untuk membangun keahlian dan sumber daya untuk mengelola kepatuhan untuk setiap lingkungan. Jika setiap penyedia cloud menggunakan lingkungan dasar, atau landing zone yang berbeda, Anda perlu memahami standar dan persyaratan kepatuhan mana yang dapat didukung oleh setiap landing zone, dan ini mungkin menentukan apakah beban kerja tertentu dapat di-host pada penyedia tersebut. Anda dapat mengelola kepatuhan untuk setiap penyedia secara terpisah atau menggunakan solusi yang dibuat khusus atau mitra yang dapat memusatkan manajemen di seluruh penyedia. AWS Marketplacemenyediakan solusi turnkey yang juga dapat memenuhi persyaratan kepatuhan Anda.

- Bagaimana Anda dapat menilai dan mengontrol biaya dan penggunaan di beberapa penyedia cloud?

  Jika institusi akademis Anda baru mengenal cloud, kami sarankan Anda menetapkan visibilitas biaya dan mekanisme kontrol untuk mendapatkan wawasan tentang layanan cloud mana yang digunakan, milik siapa sumber daya cloud tersebut, apa tujuan sumber daya cloud tersebut, dan potensi penghematan biaya apa yang dapat dicapai dengan mengoptimalkan konsumsi. Lembaga dapat mencapai laba atas investasi yang signifikan dengan bermitra dengan penyedia layanan cloud mereka untuk memigrasi dan memodernisasi sistem mission-critical, karena mereka dapat menegosiasikan perjanjian tingkat perusahaan, mendapatkan keuntungan dari harga volume, dan memanfaatkan keahlian penyedia layanan cloud. Jika Anda harus mengontrol biaya dan penggunaan di beberapa penyedia, pertimbangkan bagaimana Anda dapat mengumpulkan dan menganalisis biaya dan penggunaan dari masing-masing penyedia, baik dengan proses dan perkakas internal atau dengan menggunakan solusi mitra. Banyak organisasi mulai mengidentifikasi operasi keuangan cloud (FinOps) sebagai fungsi utama dan mendedikasikan sumber daya untuk menginjili dan menerapkan kemampuan untuk manajemen biaya cloud dan optimalisasi.
- Apakah Anda memiliki mekanisme untuk mengelola izin pengguna dengan mudah dari waktu ke waktu?

  Kami merekomendasikan agar lembaga akademis memahami kebutuhan pemangku kepentingan inti ketika mereka pertama kali mendekati cloud. Pengguna sistem kelembagaan termasuk mahasiswa, fakultas, peneliti, staf TI, administrasi, keamanan, masyarakat umum, dan kolaborator pihak ketiga. Anda harus mengidentifikasi kebutuhan inti pengguna ini dan memastikan bahwa Anda memiliki mekanisme yang tepat untuk memberi mereka akses ke layanan cloud. Jenis pengguna yang berbeda memerlukan berbagai jenis akses ke layanan cloud. Misalnya, mahasiswa, fakultas, dan masyarakat umum membutuhkan akses ke aplikasi; Staf TI, administrator, dan keamanan membutuhkan akses ke infrastruktur cloud; peneliti dan

kolaborator pihak ketiga mereka membutuhkan akses ke lingkungan penelitian yang aman; fakultas membutuhkan akses ke lingkungan pengajaran yang aman dan bahkan mungkin ingin memberi siswa akses langsung ke teknologi cloud. Anda harus memiliki alat untuk mengelola identitas ini secara terpusat secara otomatis, dan menggunakan proses yang telah ditetapkan untuk mengidentifikasi, memberikan, dan mencabut izin karena peran dan tanggung jawab berubah seiring waktu.

- Apakah Anda memiliki mekanisme untuk mengintegrasikan sistem baru dengan solusi manajemen identitas Anda secara tepat?

  Kami merekomendasikan bahwa lembaga akademik memudahkan untuk mengintegrasikan sistem baru dengan sistem manajemen identitas mereka. Hal ini memberikan lembaga fleksibilitas untuk mendukung berbagai fungsi mission-critical dengan memungkinkan pemangku kepentingan untuk mendapatkan dan membangun sistem yang dapat dengan mudah diintegrasikan ke dalam sistem manajemen identitas. Dengan menyederhanakan proses integrasi, pemangku kepentingan akan lebih kecil kemungkinannya untuk menggunakan langkah-langkah kontrol akses mereka sendiri, yang mungkin tidak menerapkan praktik terbaik keamanan seperti sistem masuk tunggal, kunci sandi, dan otentikasi multi-faktor (MFA). Pastikan bahwa sistem manajemen identitas Anda dapat beroperasi dengan sistem yang diperlukan melalui integrasi asli atau protokol standar industri.
- Apakah Anda memiliki mekanisme untuk memungkinkan deteksi dan respons insiden yang efektif?

  Institusi pendidikan sering menjadi sasaran serangan siber dan ransomware. Untuk membantu mendeteksi dan menanggapi insiden tersebut secara efektif, kami merekomendasikan pendekatan bercabang dua:

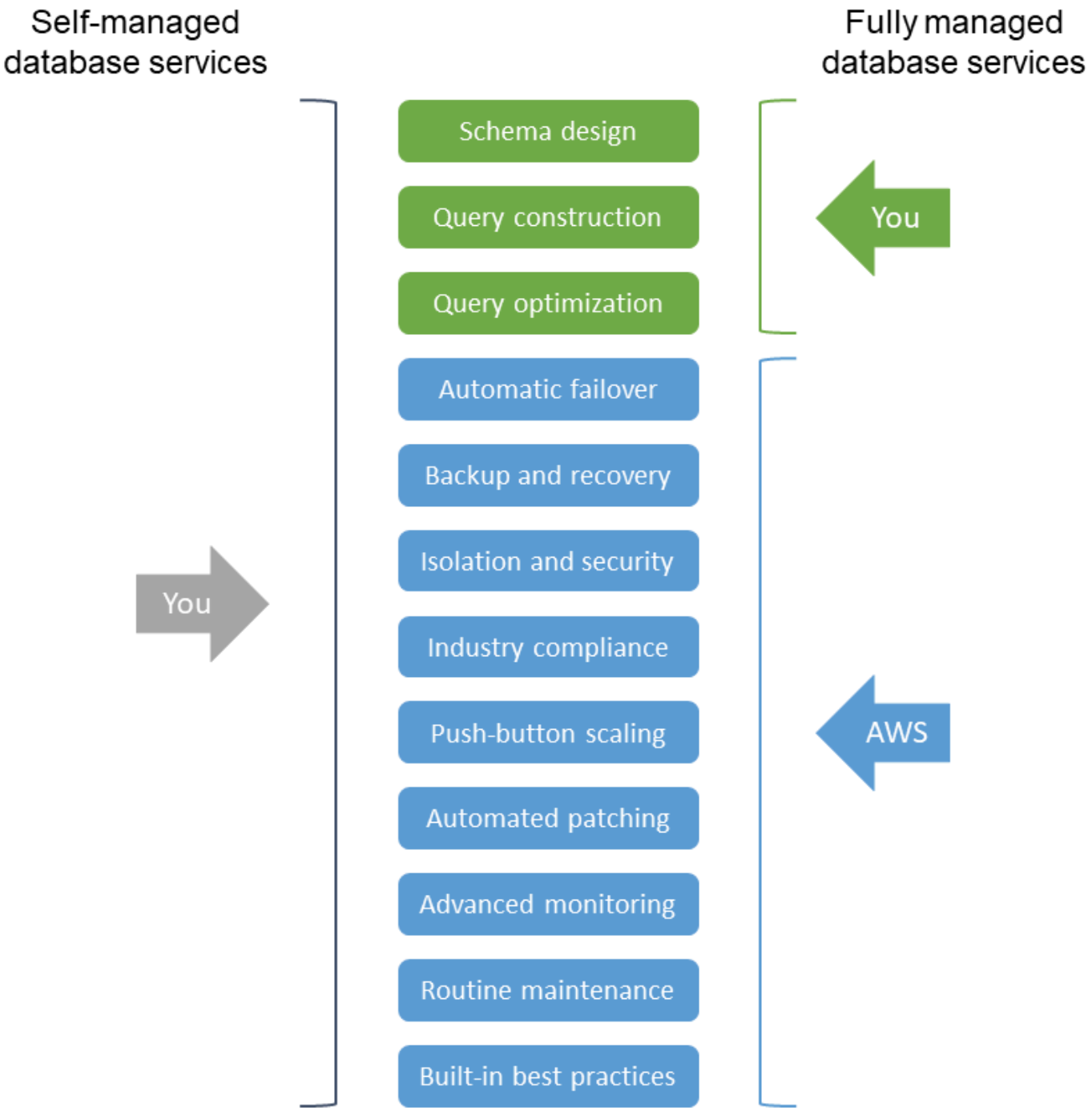
  - Fokuskan upaya Anda pada tindakan pencegahan dalam bentuk kontrol keamanan yang secara otomatis tertanam di lingkungan cloud.
  - Menerapkan kemampuan deteksi yang membantu responden insiden siber mendeteksi, menahan, dan mengurangi pelanggaran keamanan secara tepat waktu.

Seperti kepatuhan, Anda harus memastikan bahwa Anda memiliki sumber daya, keahlian, dan alat untuk mendeteksi, mencegah, dan menanggapi peristiwa di setiap lingkungan. Dengan berfokus pada satu penyedia cloud utama, Anda dapat membatasi sumber daya yang diperlukan. Lembaga akademis yang tidak memiliki tim operasi keamanan yang matang harus mencari vendor perangkat lunak independen, penyedia deteksi dan respons terkelola, dan konsultan keamanan siber untuk mendapatkan bantuan di bidang ini.

Mengadopsi layanan cloud-native, terkelola sedapat mungkin dan praktis

Ketika Anda awalnya mempertimbangkan bagaimana memanfaatkan layanan cloud, menggunakan layanan infrastruktur dan alat pengembangan yang akrab dengan tim Anda mungkin tampak seperti jalan terbaik ke depan. Namun, memilih layanan terkelola cloud-native, terutama opsi tanpa server, dapat sangat mengurangi biaya, usaha, dan kompleksitas.

Layanan cloud-native dan terkelola menghilangkan banyak tugas TI yang tidak terdiferensiasi yang membutuhkan waktu dan usaha dari staf Anda yang dapat dihabiskan dengan lebih baik untuk aktivitas yang berfokus pada misi. Selain itu, karena penyedia meningkatkan kemampuan layanan mereka, solusi Anda secara alami mewarisi peningkatan bertahap dalam efisiensi, keamanan, ketahanan, kinerja, dan karakteristik lainnya. Misalnya, layanan database yang dikelola sepenuhnya adalah sistem manajemen basis data relasional yang kaya fitur, tetapi Anda tidak perlu menyediakan dan mengelola server dan sistem operasi yang mendasari basis data berjalan. Ini menghilangkan tugas administratif yang biasanya diperlukan saat Anda memelihara database relasional di pusat data Anda sendiri atau di server virtual yang dikelola sendiri yang Anda sediakan di cloud. Diagram berikut menggambarkan perbedaan ini.



Manfaat menghilangkan manajemen infrastruktur jelas ketika Anda membandingkan layanan terkelola cloud-native dengan pendekatan pengelolaan mandiri yang sebanding. Akibatnya, kapan pun Anda perlu menerapkan komponen yang akan dijalankan oleh aplikasi yang dibeli atau dikembangkan khusus, Anda harus menggunakan layanan terkelola cloud-native untuk mengurangi waktu dan tenaga.

Saat tim Anda bertanggung jawab untuk membangun, menerapkan, atau mengelola solusi di cloud, gunakan layanan terkelola cloud-native untuk memanfaatkan sepenuhnya kemampuan dan inovasi penyedia cloud Anda yang berbeda. Strategi ini memungkinkan Anda untuk memilih, mengintegrasikan, dan menerapkan layanan cloud dengan cara yang mengurangi waktu dan upaya yang dibutuhkan proyek-proyek ini, sekaligus meningkatkan ketahanan dan keamanannya. Untuk strategi cloud yang sukses, pertimbangkan untuk mengadopsi blok bangunan cloud-native ini saat Anda memigrasikan solusi khusus ke cloud, mengembangkan solusi baru di cloud, atau menerapkan perangkat lunak berlisensi di cloud. Saat Anda mengevaluasi opsi untuk layanan terkelola cloud-native, pertimbangkan pertanyaan kunci berikut.

- Apakah Anda perlu lebih memfokuskan waktu dan upaya staf Anda pada fungsionalitas yang merupakan inti dari misi pendidikan Anda?

  Mengelola server, bahkan yang virtual, membutuhkan waktu dan perhatian untuk memastikan bahwa mereka tetap up to date dengan upgrade dan patch perangkat lunak sistem. Menggunakan layanan terkelola yang menangani tugas-tugas ini untuk Anda memungkinkan Anda mengarahkan waktu staf TI ke aktivitas yang lebih selaras langsung dengan misi institusi Anda. Misalnya, jika Anda perlu menyebarkan kontainer, pertimbangkan layanan terkelola tanpa server seperti AWS Fargatesehingga Anda tidak perlu mengonfigurasi dan memelihara server. Dengan menghilangkan kebutuhan untuk mendapatkan, menyediakan, dan mengelola infrastruktur yang mendasarinya, Anda dapat fokus untuk memberikan fungsionalitas baru, mengoptimalkan kinerja, dan meningkatkan pengalaman pengguna. Pertimbangkan manfaat ini ketika Anda mengevaluasi layanan terkelola terhadap opsi yang dikelola sendiri.
- Upaya apa yang diperlukan tim Anda untuk mengadopsi layanan terkelola cloud-native?

  Mungkin ada kurva pembelajaran untuk merancang dan menerapkan solusi dengan layanan cloud-native yang dikelola, tetapi upaya ini akan dilunasi dengan pengurangan biaya, waktu, dan kompleksitas selama masa pakai solusi. Karena pay-as-you-go sifat komputasi awan sesuai permintaan, layanan cloud-native memungkinkan Anda untuk dengan cepat mengulangi dan bereksperimen dengan cara yang lebih gesit sambil menghindari investasi di muka. Ini mengarah pada peningkatan inovasi dan jadwal proyek yang lebih pendek. Namun, untuk mewujudkan manfaat ini secara efektif, pertimbangkan apa yang mungkin diperlukan untuk mengadopsi dan menggunakan layanan, seperti pelatihan staf tentang pola penggunaan yang optimal dan refactoring kode untuk mengakomodasi layanan khusus. APIs Bahkan jika layanan menggunakan standar industri atau open source APIs, Anda mungkin perlu memfaktorkan ulang atau mengonfigurasi aplikasi Anda untuk menangani disparitas fitur atau ketidakcocokan versi.

- Bagaimana Anda saat ini menyebarkan dan mengelola infrastruktur? Apakah Anda perlu mempertahankan tingkat kontrol itu?

  Ada berbagai cara untuk meng-host dan mengelola infrastruktur di cloud, termasuk menggunakan host bare-metal, mesin virtual, layanan kontainer terkelola, dan penawaran tanpa server. Bahkan jika saat ini Anda menggunakan infrastruktur serupa, seperti mesin virtual atau kontainer, di lingkungan lokal Anda, pertimbangkan apakah pendekatan alternatif akan cocok untuk beban kerja tertentu. Misalnya, alih-alih menjalankan semua aplikasi di mesin virtual, pertimbangkan untuk mengkontainerisasi aplikasi Anda dan manfaatkan layanan kontainer terkelola seperti Amazon Elastic Container Service (Amazon ECS). Ini mungkin memerlukan refactoring, tetapi Anda dapat menggunakan alat seperti AWS App2Containeruntuk menyederhanakan dan membantu dengan containerization. Mengambil langkah ini lebih jauh, alih-alih menyebarkan server atau wadah untuk semua komponen, pertimbangkan opsi tanpa server sepenuhnya. Teknologi tanpa server menampilkan penskalaan otomatis, ketersediaan tinggi bawaan, dan model pay-for-use penagihan untuk meningkatkan kelincahan dan mengoptimalkan biaya. Pada saat yang sama, mereka menghilangkan kebutuhan untuk mengelola server dan merencanakan kapasitas. Layanan komputasi tanpa server seperti AWS Lambdamerupakan inti dari arsitektur tanpa server. Lambda mendukung bahasa pemrograman umum dan memungkinkan pengembang untuk fokus pada kode aplikasi alih-alih mengelola infrastruktur. Jelajahi opsi ini untuk setiap beban kerja, dan pertimbangkan faktor-faktor seperti kurva pembelajaran, overhead manajemen, biaya, dan perizinan.
- Apakah Anda harus menyebarkan dan mengelola infrastruktur untuk perangkat lunak berlisensi?

  Saat Anda menerapkan dan mengelola perangkat lunak berlisensi dari vendor perangkat lunak independen (ISVs), mungkin tampak logis untuk meniru penerapan lokal Anda dengan infrastruktur cloud. Misalnya, Anda dapat mempertimbangkan untuk mengganti mesin virtual lokal dengan mesin virtual yang dihosting cloud. Meskipun ini adalah opsi yang layak, pertimbangkan apakah Anda dapat mengganti komponen arsitektur apa pun dengan layanan terkelola cloud-native. Misalnya, Anda mungkin dapat mengganti server database yang dikelola sendiri dengan layanan database yang dikelola sepenuhnya yang mengurangi beban administrasi saat menjalankan mesin database yang sama. Banyak yang ISVs sudah menggunakan arsitektur cloud yang memanfaatkan layanan terkelola, dan bahkan mungkin menawarkan templat bawaan untuk menyederhanakan penerapan. Jika memungkinkan, Anda harus memilih ISVs yang menawarkan panduan preskriptif dan dukungan untuk penyebaran cloud. Sebelum Anda menyebarkan perangkat lunak berlisensi ke cloud, pastikan untuk berkonsultasi dengan ISV Anda untuk memahami bagaimana lisensi lingkungan cloud mungkin berbeda dari lisensi lokal.

- Apakah Anda khawatir bahwa menggunakan layanan terkelola dapat mengakibatkan penguncian vendor?

  Banyak layanan terkelola cloud-native dibangun untuk mendukung standar industri umum dan. APIs Misalnya, layanan analitik seperti AWS Gluedan Amazon EMR dibangun di atas kerangka pemrosesan dan penyimpanan standar industri seperti Apache Spark dan Apache Parquet. AWS Lambdanative mendukung Java, Go, Microsoft, Node.js PowerShell, C #, Python, dan kode Ruby. Amazon Relational Database Service (Amazon RDS) mendukung beberapa versi mesin database umum, termasuk SQL Server, Oracle, PostgreSQL, dan MySQL. Ketika layanan memiliki hak milik APIs, solusi asli atau mitra mungkin tersedia untuk berinteraksi APIs dengan menggunakan protokol cloud-agnostik umum. Misalnya, Amazon Simple Storage Service (Amazon S3) memiliki API khusus layanan untuk integrasi langsung, tetapi Anda juga dapat berinteraksi dengannya dengan menggunakan protokol penyimpanan standar seperti Network File System (NFS), Server Message Block (SMB), dan Internet Small Computer Systems Interface (iSCSI) saat Anda menggunakannya. AWS Storage Gateway Anda tetap harus fokus memilih layanan terkelola cloud-native yang paling sesuai dengan kebutuhan Anda sekaligus mengurangi overhead operasional secara maksimal, tetapi Anda mungkin lebih memilih layanan yang menggunakan atau menyediakan standar dan protokol industri umum.

Menerapkan arsitektur hibrida ketika ada, investasi lokal memberi insentif untuk penggunaan berkelanjutan

Sebagian besar institusi pendidikan telah berinvestasi di pusat data lokal dengan berbagai skala untuk meng-host aplikasi perusahaan, solusi penyimpanan data, lingkungan komputasi pengguna akhir (EUC), dan sumber daya komputasi bersama. Semua sumber daya di pusat data ini tunduk pada siklus penyegaran yang berbeda, di mana Anda harus mempertimbangkan pertumbuhan masa depan dan menyediakan kapasitas yang cukup untuk mengakomodasi skala puncak, yang mungkin diperlukan hanya beberapa kali dalam setahun. Akibatnya, sumber daya sering diam hingga siklus penyegaran berikutnya. Perencanaan, penganggaran, pengadaan, dan penyebaran perangkat keras baru dapat memakan waktu berminggu-minggu, jika tidak berbulan-bulan atau lebih lama. Proses yang panjang ini menghambat inovasi dan dapat menunda pembelajaran dan penelitian.

Cloud computing memecahkan banyak tantangan ini. Cloud menyediakan sumber daya TI sesuai permintaan, pay-as-you-go sehingga Anda dapat lebih mencocokkan kapasitas saat ini dengan permintaan aktual tanpa perencanaan dan investasi awal yang besar. Namun, jika Anda telah melakukan investasi yang signifikan dalam perangkat keras dan sumber daya lokal, Anda harus

berusaha memanfaatkan sumber daya tersebut secara efisien dan menambahkannya sesuai kebutuhan dengan teknologi cloud dalam model hybrid.

Strategi cloud hybrid yang sukses memanfaatkan investasi yang ada sambil memberikan kelincahan, skalabilitas, dan keandalan yang lebih besar daripada yang dapat didukung oleh investasi tersebut. Pertimbangan berikut dapat membantu Anda memulai.

- Ketika Anda harus meng-host beban kerja baru, apakah Anda berpikir tentang cloud terlebih dahulu?

  Bagaimana Anda menggunakan infrastruktur cloud publik dan pribadi bersama-sama mendefinisikan strategi cloud hybrid Anda. Pendekatan cloud pertama tidak berarti bahwa cloud adalah pilihan yang lebih baik untuk semua beban kerja Anda. Namun, ketika Anda merencanakan beban kerja baru, evaluasi cloud sebagai opsi pertama, terutama untuk beban kerja yang membutuhkan teknologi baru atau melebihi kapasitas penyimpanan dan komputasi yang tersedia di tempat. Beban kerja yang memiliki pola penggunaan sementara dan tidak konsisten, membutuhkan hasil yang cepat, mudah dibawa, atau memerlukan perangkat keras terbaru adalah kandidat ideal untuk skalabilitas dan elastisitas cloud. Juga, pertimbangkan apakah beban kerja akan mendapat manfaat dari layanan terkelola cloud-native yang tidak tersedia di tempat, bahkan jika Anda memiliki kapasitas yang tersedia.
- Apakah Anda memahami TCO lingkungan lokal Anda dan bermitra dengan CFO Anda saat melakukan investasi baru?

  Kami menyarankan Anda memahami total biaya kepemilikan (TCO) sebenarnya dari pemeliharaan pusat data lokal Anda sendiri. Ada banyak biaya tersembunyi yang terkait dengan memiliki dan mengoperasikan infrastruktur di tempat, termasuk tidak hanya perangkat keras, perangkat lunak, dan dukungan, tetapi juga fasilitas, utilitas, asuransi, dan jam staf. Biaya-biaya ini dapat berdampak negatif pada produktivitas staf, ketahanan operasional, dan kelincahan bisnis. Evaluasi struktur lisensi Anda saat ini dan periode pembaruan dan pemeliharaannya juga. Bermitra dengan Chief Financial Officer (CFO) Anda dapat membantu Anda mengidentifikasi semua biaya tersembunyi ketika Anda berencana untuk melakukan investasi baru. Beberapa lisensi mungkin menawarkan opsi Bring Your Own License (BYOL) di cloud, atau mungkin lebih atau kurang kondusif untuk layanan cloud. Memahami TCO sebenarnya dari infrastruktur Anda saat ini membantu Anda memprioritaskan adopsi cloud untuk beban kerja yang memiliki dampak terbesar pada total TCO organisasi Anda. Tim AWS akun Anda memiliki alat yang tersedia untuk membantu Anda lebih memahami TCO lokal Anda.
- Infrastruktur apa yang Anda perlukan untuk mendukung penerapan hybrid?

Untuk berhasil mengadopsi model hybrid, Anda memerlukan jaringan dasar, keamanan, dan perkakas infrastruktur. Pastikan Anda dapat mempertahankan konektivitas jaringan yang memadai dengan penyedia cloud Anda. Ini bisa melalui kombinasi konektivitas internet yang ada, jaringan pribadi virtual (VPNs), koneksi khusus seperti, penyedia konektivitas pihak ketiga Direct Connect, atau Internet2 dan jaringan penelitian dan pendidikan regional. Pastikan Anda memiliki identitas terpadu dan manajemen akses di lingkungan lokal dan cloud Anda. Menetapkan alat dan proses untuk menegakkan keamanan, biaya, dan pagar pembatas penggunaan yang konsisten.

- Apakah staf TI Anda siap untuk mengoperasikan penyebaran hybrid?

  Layanan cloud dapat memerlukan keahlian khusus yang mungkin tidak dimiliki tim Anda. Untuk membatasi pelatihan dan pemberdayaan yang diperlukan untuk meningkatkan keterampilan staf TI Anda untuk adopsi cloud yang efektif, pertimbangkan apakah penyedia cloud menawarkan layanan apa pun yang menggunakan kembali dan membangun keahlian yang ada di tempat dan cloud. Misalnya, jika Anda menggunakan dan akrab dengan Kubernetes, Anda dapat mempertimbangkan untuk menggunakan Amazon Elastic Kubernetes Service (Amazon EKS) atau Amazon EKS Anywhere. Jika Anda menggunakan dan terbiasa NetApp, Anda dapat mempertimbangkan untuk menggunakan Amazon FSx untuk NetApp ONTAP. Demikian pula, pertimbangkan juga apakah solusi mitra yang ada yang Anda gunakan memiliki integrasi asli atau dukungan untuk lingkungan cloud.
- Bisakah Anda membongkar penyimpanan jangka panjang atau komputasi penggunaan rendah dari tempat ke cloud?

  Penyimpanan cloud menyediakan beberapa opsi hemat biaya untuk penyimpanan data jangka panjang. Misalnya, Amazon Simple Storage Service (Amazon S3) menawarkan berbagai tingkatan penyimpanan yang dioptimalkan untuk berbagai kasus penggunaan. Jika institusi Anda diharuskan menyimpan data tertentu untuk jangka waktu yang lama, pertimbangkan solusi penyimpanan dingin seperti Amazon Glacier. Membongkar data ini ke penyimpanan cloud dapat membebaskan penyimpanan lokal berkinerja tinggi yang berharga. Layanan seperti AWS Storage Gatewaymemudahkan aplikasi lokal untuk mengakses tingkatan penyimpanan cloud melalui protokol standar seperti SMB, NFS, dan iSCSI. Demikian pula, pertimbangkan untuk membongkar tugas komputasi yang jarang digunakan atau rendah. Jika Anda memiliki server lokal yang didedikasikan untuk tugas tersebut, Anda dapat menggunakan layanan komputasi awan yang dapat diskalakan, tempat sumber daya disediakan sesuai permintaan dan Anda hanya membayar untuk apa yang Anda gunakan. Opsi komputasi berbiaya rendah, jangka panjang, dan penggunaan rendah juga menjadikan cloud ideal untuk pencadangan dan pemulihan bencana. Anda dapat menggunakan penyimpanan dan komputasi yang aman, tahan lama, terukur di cloud untuk

melindungi data Anda dan pulih dengan cepat jika terjadi bencana tanpa harus mempertahankan penyimpanan yang diperlukan dan menghitung infrastruktur sendiri.

- Apakah Anda memiliki kapasitas yang cukup di tempat untuk bereksperimen dan berinovasi?

  Kurangnya elastisitas dan kelincahan dalam lingkungan lokal dengan ukuran tetap dapat membatasi layanan dan teknologi yang tersedia bagi pengguna Anda. Jika Anda memiliki siklus penyegaran yang ketat, beban kerja baru mungkin harus menunggu hingga siklus berikutnya untuk implementasi. Model operasi ini dapat membatasi eksperimen dan inovasi lambat. Ketika Anda memiliki beban kerja baru atau baru yang perlu diuji, pertimbangkan untuk menggunakan layanan cloud elastis yang dapat diskalakan. Sumber daya cloud dapat disediakan dan dibatalkan sesuai permintaan dan Anda hanya membayar untuk apa yang Anda gunakan, sehingga Anda dapat bereksperimen dan gagal dengan cepat sambil meminimalkan risiko organisasi.
- Apakah Anda memiliki persyaratan kepatuhan atau kinerja unik yang memaksa Anda untuk menyimpan data di tempat?

  Beban kerja dengan persyaratan residensi atau latensi data yang ketat mungkin menentukan bahwa Anda menyimpan data di tempat atau sedekat mungkin dengan pengguna Anda. Untuk kasus penggunaan ini, Anda dapat memprioritaskan penggunaan sumber daya lokal yang ada. Namun, pertimbangkan apakah penyedia cloud Anda menawarkan layanan edge atau mekanisme untuk menggunakan teknologi berbasis cloud di tempat. Layanan Edge memberikan pemrosesan data, analisis, dan penyimpanan lebih dekat ke titik akhir Anda sendiri, dan memungkinkan Anda untuk menggunakan alat di luar pusat data penyedia cloud standar. Misalnya, AWS menawarkan layanan seperti AWS Local Zones dan AWS Wavelengthuntuk menyebarkan aplikasi di lokasi tertentu yang lebih dekat dengan pengguna akhir. Anda juga dapat membawa layanan dan fungsionalitas cloud ke pusat data yang ada dengan layanan seperti AWS Outposts, Amazon ECS Anywhere AWS Storage Gateway, dan Amazon EKS Anywhere.

Cadangan multicloud hanya untuk beban kerja yang tidak dapat memenuhi persyaratan teknis atau bisnis mereka melalui satu penyedia cloud

Multicloud mengacu pada penggunaan layanan cloud dari beberapa (dua atau lebih) penyedia layanan cloud. Memiliki strategi multicloud dapat menawarkan manfaat tertentu, seperti opsi untuk membuka kemampuan berbeda dari beberapa penyedia cloud atau kemampuan untuk memenuhi persyaratan kedaulatan data yang mungkin tidak dapat diakomodasi oleh satu penyedia cloud.

Namun, untuk setiap penyedia yang Anda gunakan, pastikan Anda memiliki orang, keterampilan, pelatihan, dan perangkat yang tepat untuk menggunakan penyedia itu secara efektif. Selain itu, jika Anda ingin menggunakan strategi multicloud untuk beban kerja tertentu, Anda akan memerlukan sumber daya tambahan untuk mengintegrasikan dan mengoperasikan layanan yang diperlukan dari setiap penyedia cloud. Kami menyarankan Anda mempertimbangkan multicloud hanya ketika manfaatnya lebih besar daripada peningkatan investasi. Untuk menentukan apakah Anda harus memilih strategi multicloud, pertimbangkan pertanyaan-pertanyaan kunci berikut.

- Apakah Anda memiliki sumber daya dan keahlian untuk menavigasi layanan yang ditawarkan oleh penyedia cloud yang berbeda?

  Ketika beberapa penyedia cloud menawarkan berbagai produk dan layanan, staf Anda membutuhkan keterampilan penting untuk menavigasi kemampuan masing-masing penyedia. Menggunakan satu layanan penyedia cloud saja dapat memerlukan peningkatan keterampilan dan pelatihan untuk staf Anda, tergantung pada layanan dan fitur yang Anda gunakan. Jika Anda mempertimbangkan strategi multicloud, evaluasi sumber daya yang ada untuk menentukan keahlian tambahan apa yang Anda perlukan untuk menggunakan layanan dari beberapa penyedia cloud secara efektif. Anda mungkin perlu menambah staf Anda atau menginvestasikan waktu dan uang tambahan dalam meningkatkan keterampilan dan pelatihan di luar apa yang diperlukan untuk satu penyedia cloud. Jika Anda sudah memiliki tim individu atau pengguna yang menggunakan penyedia cloud yang berbeda, pertimbangkan manfaat organisasi dari mengkonsolidasikan mereka ke penyedia cloud utama case-by-case berdasarkan.
- Overhead tambahan apa yang akan diperkenalkan oleh arsitektur multicloud tertentu?

  Driver umum untuk multicloud adalah keinginan untuk menggunakan layanan terkelola tertentu dari satu penyedia yang memiliki kemampuan yang dapat dibedakan dari layanan penyedia cloud lain. Misalnya, Anda mungkin ingin menggunakan satu penyedia cloud untuk kebutuhan infrastruktur Anda dan layanan terkelola penyedia lain untuk layanan domain dan direktori. Namun, bahkan jika layanan terkelola tunggal itu mengurangi beban administrasi dan menyederhanakan pengelolaan komponen arsitektur itu, itu dapat memperkenalkan overhead tambahan untuk beban kerja lainnya, seperti refactoring kode, kebutuhan konektivitas pribadi, atau pekerjaan integrasi manual. Identifikasi overhead tambahan ini di depan dan pastikan itu tidak mengimbangi atau melampaui manfaat yang diperoleh tim Anda dari layanan yang berbeda.
- Bagaimana Anda akan memusatkan pemantauan dan manajemen di seluruh penyedia cloud?

  Saat Anda mulai menerapkan aplikasi dan fungsionalitas dengan menggunakan sumber daya dari penyedia cloud yang berbeda, pertimbangkan bagaimana Anda akan menandai, memantau, dan

mengelola sumber daya tersebut. Setiap penyedia akan memiliki perkakas mereka sendiri, yang mungkin dapat Anda perluas ke lingkungan lain. Misalnya, Anda dapat menggunakan Amazon CloudWatch untuk memantau metrik dan log utama, membuat alarm, dan memvisualisasikan aplikasi dan infrastruktur Anda di lingkungan tunggal, hibrida, dan multicloud. Anda juga dapat menggunakannya AWS Systems Manageruntuk meningkatkan visibilitas dan kontrol sumber daya, mendiagnosis dan memulihkan masalah operasional dengan cepat, dan mengotomatiskan proses seperti memperbarui dan menambal mesin virtual di seluruh lingkungan. Jika Anda memiliki persyaratan yang tidak dapat didukung oleh alat penyedia, Anda dapat menjelajahi solusi mitra, tetapi ini dapat menambah biaya tambahan atau upaya integrasi.

- Bagaimana Anda bisa mengelola infrastruktur sebagai kode dengan otomatisasi saat menggunakan penyedia cloud yang berbeda?

  Saat Anda menjalankan sumber daya di cloud, penyediaan otomatis dan pengelolaan sumber daya membantu Anda mengelola berbagai lingkungan secara efisien. Alat otomatisasi asli APIs dan asli bervariasi di seluruh penyedia cloud. Jika memungkinkan, pertimbangkan untuk menggunakan seperangkat alat orkestrasi dan penyebaran umum yang dapat mengakomodasi sumber daya penyedia cloud yang berbeda. Ini memberikan fleksibilitas yang lebih besar dan menyederhanakan operasi di beberapa cloud. Namun, mungkin lebih mudah untuk menggunakan otomatisasi asli masing-masing penyedia secara terpisah dan menetapkan proses organisasi untuk memastikan penggunaan yang tepat.
- Apakah Anda memiliki persyaratan kepatuhan dan peraturan yang harus dipenuhi oleh setiap penyedia cloud?

  Anda mungkin memiliki pertimbangan peraturan yang menentukan bagaimana data harus disimpan dan ditangani. Fokus pada kebijakan standarisasi (seperti lalu lintas jaringan, penyimpanan, dan keamanan) yang dapat diterapkan secara otomatis ke setiap lingkungan cloud di seluruh penyedia cloud. Pertimbangkan bagaimana aplikasi Anda akan berkomunikasi dengan data mereka, dan host mereka di penyedia yang sama. Jika aplikasi Anda dan datanya terfragmentasi di seluruh penyedia, akan sulit untuk memastikan bahwa Anda memenuhi persyaratan kepatuhan dan peraturan. Seringkali yang terbaik adalah memiliki aplikasi sedekat mungkin dengan data untuk meminimalkan latensi jaringan, memaksimalkan throughput data, dan membatasi jalan keluar data sambil menyederhanakan keamanan dan kontrol akses.
- Apakah Anda dapat meminimalkan TCO dan memaksimalkan diskon harga saat Anda menerapkan aplikasi di seluruh penyedia cloud?

Penting untuk memperhitungkan total biaya kepemilikan (TCO) ketika mempertimbangkan multicloud. Menjalankan aplikasi Anda di beberapa penyedia cloud dapat meningkatkan biaya operasional dan overhead administratif untuk memelihara dan mengelola sumber daya di setiap lingkungan. Selain itu, penyebaran penggunaan di beberapa penyedia membuatnya lebih sulit untuk mengambil keuntungan dari diskon harga volume penyedia tertentu atau perjanjian perusahaan. Pertimbangkan faktor-faktor ini ketika Anda menentukan apakah manfaat multicloud menjamin peningkatan TCO.

Contoh kasus penggunaan

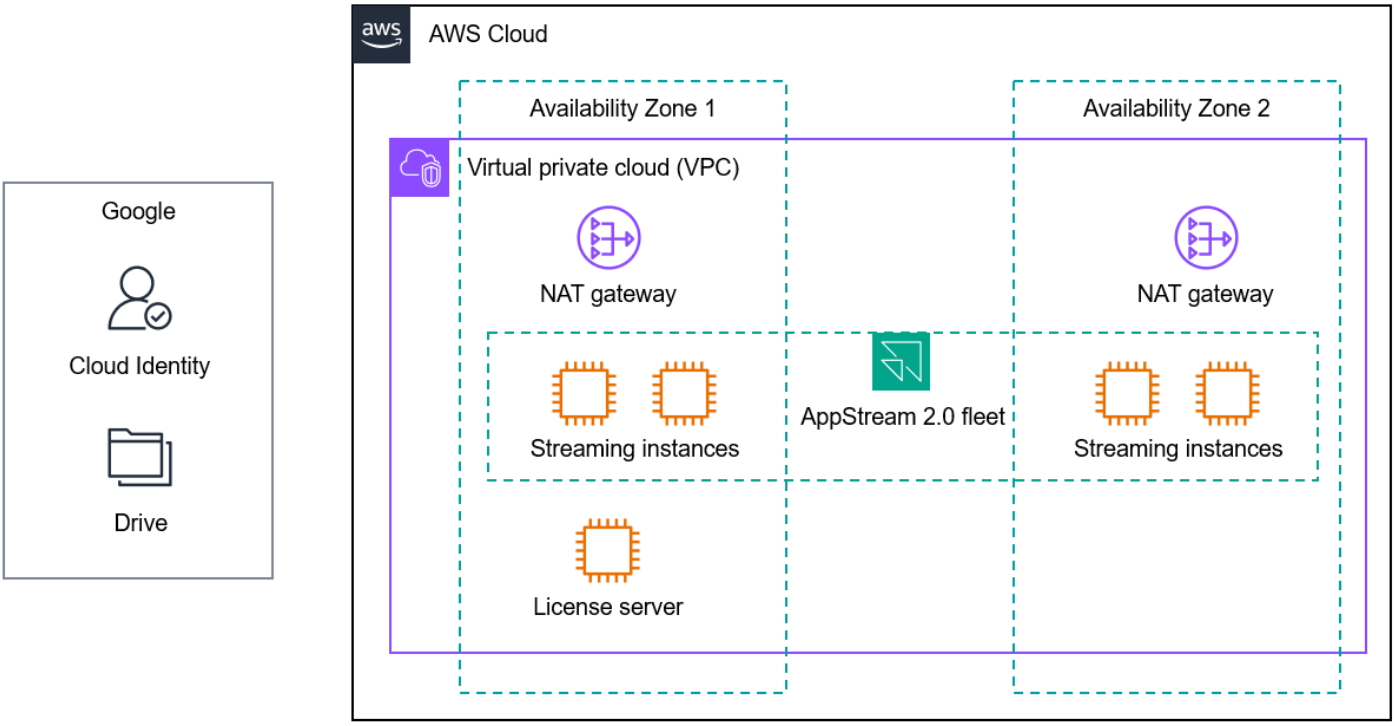
Untuk lebih memahami penerapan prinsip-prinsip ini dalam skenario yang berbeda, mari kita bahas beberapa contoh kasus penggunaan. Kasus penggunaan ini didasarkan pada bagaimana lembaga pendidikan dunia nyata mengadopsi layanan cloud.

- Laboratorium komputer virtual
- Memprediksi keberhasilan siswa
- Federasi identitas dan sistem masuk tunggal
- Cloud bursting untuk komputasi penelitian

Laboratorium komputer virtual

Terlepas dari popularitas alat pembelajaran berbasis web dan banyaknya perangkat pengguna seperti laptop, Chromebook, dan tablet, sebagian besar lembaga pendidikan memelihara laboratorium komputer fisik untuk aplikasi intensif sumber daya atau warisan. Laboratorium komputer ini sering menjadi kebutuhan untuk sains, teknologi, teknik, dan matematika (STEM), karir dan pendidikan teknis (CTE), media dan seni, teknik, dan kurikulum serupa. Sekolah dapat menambah atau mengganti laboratorium komputer fisik dengan desktop virtual berbasis cloud atau layanan streaming aplikasi untuk memastikan bahwa semua siswa memiliki akses ke aplikasi yang mereka butuhkan kapan saja, dari mana saja, dan di perangkat apa pun. Ini meningkatkan ekuitas digital, memungkinkan pembelajaran jarak jauh, memastikan pengalaman pengguna yang konsisten, dan mengamankan akses jarak jauh sekaligus menurunkan biaya.

Dalam pendidikan dasar dan menengah (K12), banyak sekolah di AS menggunakan Amazon WorkSpaces Applications, layanan streaming desktop dan aplikasi yang dikelola sepenuhnya, untuk menyediakan laboratorium komputer virtual untuk menyediakan akses ke Adobe Creative Cloud, perangkat lunak Autodesk, kurikulum STEM dan CTE seperti Project Lead the Way (PLTW), dan banyak lagi. Banyak organisasi K12 telah mengelola sistem masuk tunggal siswa dan penyimpanan file melalui Google Workspace dan Google Drive, yang merupakan aplikasi SaaS. Institusi ini dapat mengatur sistem masuk tunggal antara Google Workspace dan WorkSpaces Aplikasi melalui federasi SAFL 2.0. Mereka juga dapat mengonfigurasi integrasi asli antara WorkSpaces Aplikasi dan Google Drive sehingga siswa dapat menggunakan penyimpanan yang ada. Diagram berikut menggambarkan penerapan WorkSpaces Aplikasi untuk kasus penggunaan ini.



Arsitektur ini mengikuti rekomendasi ini:

- Pilih penyedia cloud utama dan strategis.Arsitektur ini menggunakan layanan cloud dari satu penyedia cloud utama. Meskipun termasuk integrasi dengan aplikasi SaaS yang tidak di-host pada penyedia yang sama, integrasi tersebut dilakukan melalui konfigurasi sederhana. Keahlian dan keahlian cloud diperlukan hanya untuk menyebarkan dan mengelola layanan dari penyedia cloud utama.
- Membedakan antara aplikasi SaaS dan layanan cloud dasar.Google Workspace dan Google Drive tidak dihosting di penyedia cloud yang sama dengan AppStream 2.0, tetapi itu dapat diterima karena penerapan ini menyediakan integrasi yang diperlukan. Single sign-on memungkinkan manajemen identitas terpusat dan dikonfigurasi dengan aman melalui SAFL 2.0. Mengaktifkan penyimpanan cloud persisten untuk siswa memerlukan perubahan konfigurasi sederhana di Google Drive dan WorkSpaces Aplikasi.
- Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud.Layanan dan integrasi yang digunakan dalam arsitektur ini membantu memenuhi persyaratan keamanan dan tata kelola institusi. Lalu lintas streaming dienkripsi. Federasi melalui Google Workspace memungkinkan manajemen identitas terpusat. Layanan jaringan seperti Amazon Virtual Private Cloud (Amazon VPC) mendukung konfigurasi subnet, routing, dan firewall. Anda dapat memfilter konten dengan menggunakan konfigurasi DNS, agen, peralatan virtual, atau layanan terkelola

seperti Amazon Route 53 Resolver DNS Firewall. Anda dapat menggunakan layanan seperti AWS Control Toweruntuk membantu memastikan bahwa akun AWS yang menghosting WorkSpaces Aplikasi mematuhi pagar dan kontrol organisasi standar.

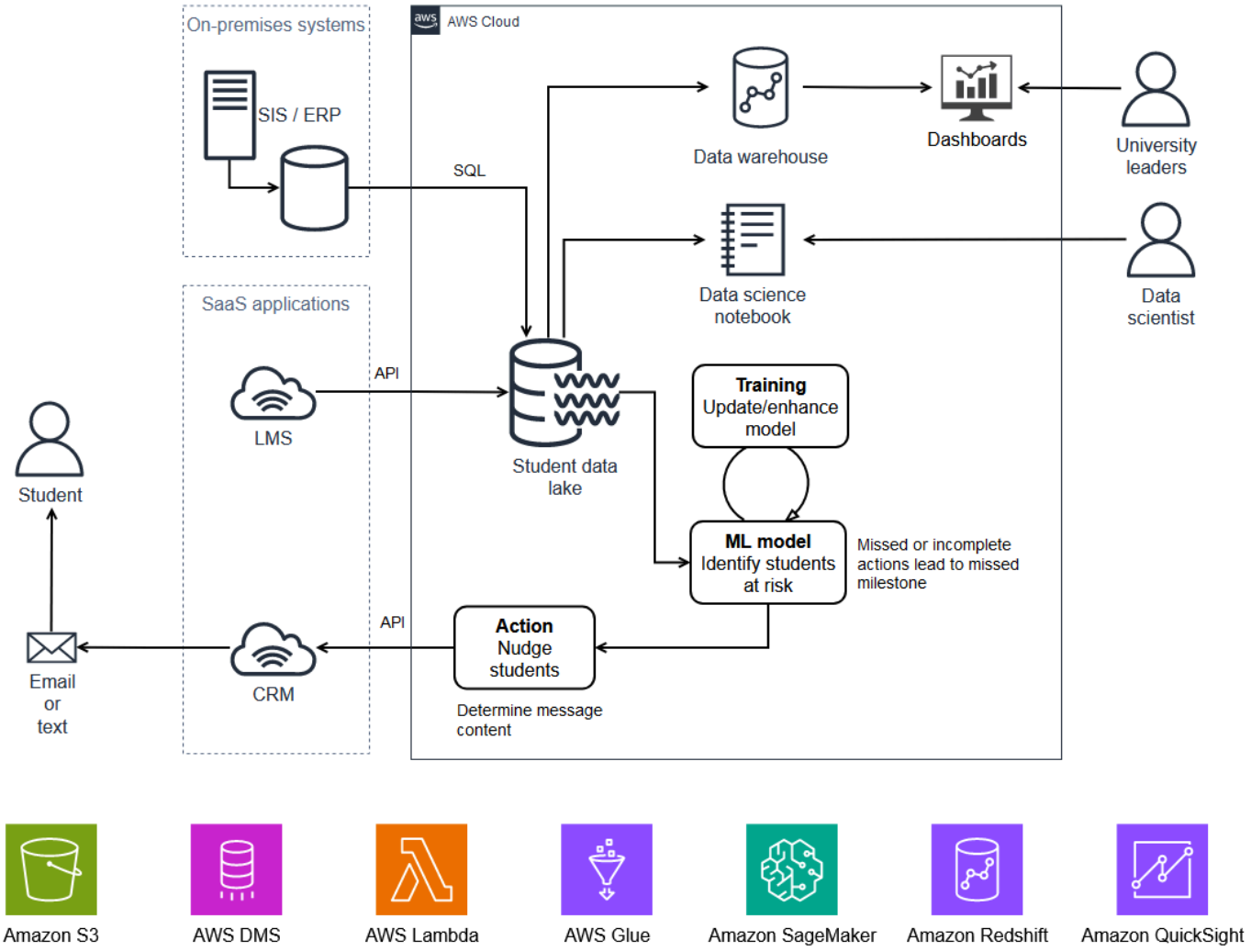
- Mengadopsi solusi cloud-native, terkelola sedapat mungkin dan praktis. WorkSpaces Aplikasi adalah layanan terkelola untuk streaming desktop dan aplikasi. Anda dapat melakukan streaming desktop dan aplikasi tanpa khawatir tentang penyediaan, penskalaan, atau pemeliharaan server. Anda menginstal aplikasi Anda, menghubungkan identitas, jaringan, dan solusi penyimpanan yang sesuai, dan kemudian mengelola dan mengalirkan aplikasi tersebut secara terpusat ke pengguna Anda. Ini menghilangkan banyak angkat berat yang tidak terdiferensiasi yang diperlukan untuk mengelola solusi streaming desktop virtual Anda sendiri.

Memprediksi keberhasilan siswa

Sebuah universitas Midwest di AS menemukan bahwa beberapa kegiatan utama untuk siswa tahun pertama yang masuk sangat memprediksi keberhasilan, baik di semester pertama kelas siswa maupun dalam mencapai gelar mereka. Universitas ingin menerapkan sistem yang mengawasi kegiatan ini diselesaikan, dan ketika tenggat waktu utama mendekati atau berlalu, mereka ingin mendorong siswa untuk menyelesaikan langkah-langkah ini.

Data sistem manajemen pembelajaran SaaS (LMS) adalah masukan utama untuk solusi ini, tetapi datanya terbukti menantang untuk diakses dan diproses dengan alat pergudangan data tim TI universitas. Selain itu, pesan kepada siswa harus dikirim melalui sistem manajemen hubungan pelanggan (CRM) berbasis cloud sekolah. Untuk membangun solusi fungsional dan menilai efektivitas petunjuk kepada siswa, universitas harus memulai pesan melalui CRM dan mengumpulkan data darinya.

Universitas mengembangkan dan menyebarkan solusi ke dalam lingkungan cloud tunggal. Solusinya adalah campuran layanan terkelola cloud-native, server cloud yang disediakan, dan integrasi dengan sistem lokal dan aplikasi SaaS berbasis cloud. Seperti yang ditunjukkan diagram berikut, solusi mencerna data dari sistem informasi siswa (SIS), LMS, dan CRM ke dalam danau data. Ini menggunakan data ini untuk mengidentifikasi siswa yang berada dalam bahaya kehilangan kegiatan utama, memulai pesan kepada mereka melalui CRM, dan menyediakan dasbor untuk kepemimpinan universitas.



Arsitektur ini mengikuti rekomendasi ini:

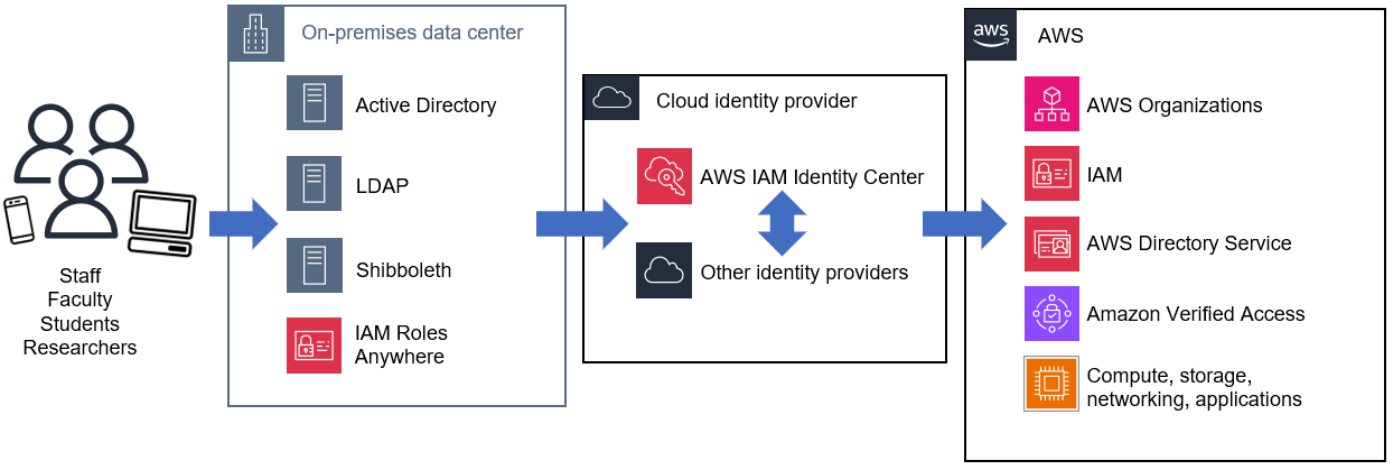
- Pilih penyedia cloud utama dan strategis.Penyedia cloud strategis universitas menampung seluruh solusi yang diterapkan. Hal ini memungkinkan staf TI dan bisnis untuk fokus pada pengembangan keterampilan dalam satu set kemampuan cloud yang terintegrasi.
- Membedakan antara aplikasi SaaS dan layanan cloud dasar.Universitas membedakan antara aplikasi SaaS dan layanan analisis cloud inti, dan menggunakan integrasi dengan aplikasi SaaS untuk mengumpulkan data dan memulai komunikasi yang sesuai.
- Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud.Universitas memastikan bahwa semua komponen arsitektur aman dengan menegakkan pagar pembatas dan kontrol, termasuk enkripsi dalam perjalanan dan saat istirahat, untuk menangani data siswa dengan tepat.

- Mengadopsi solusi cloud-native, terkelola sedapat mungkin dan praktis.Layanan terkelola cloud-native digunakan untuk konsumsi data, penyimpanan, database, dan fungsionalitas ekstrak, transformasi, dan pemuatan (ETL), yang mengurangi waktu untuk mengembangkan alur kerja pemrosesan data. end-to-end

Federasi identitas dan sistem masuk tunggal

Memastikan manajemen identitas yang konsisten di seluruh sistem inti adalah kunci untuk berhasil dan aman mengadopsi teknologi apa pun. Institusi pendidikan semakin mengadopsi identitas berbasis cloud dan solusi masuk tunggal seperti, AWS IAM Identity CenterMicrosoft Entra ID (sebelumnya Azure Active Directory), Okta,,, Ping Identity, dan CyberArk untuk menyederhanakan manajemen identitas JumpCloud, menurunkan beban operasional OneLogin, dan menegakkan praktik terbaik secara terpusat seperti otentikasi multi-faktor dan akses hak istimewa paling sedikit.

Banyak dari institusi ini masih mempertahankan manajemen identitas dan layanan direktori seperti Active Directory dan Shibboleth untuk lingkungan lokal mereka. Ini dapat diintegrasikan dengan solusi berbasis cloud untuk memungkinkan manajemen identitas terpusat dan sistem masuk tunggal untuk siswa, fakultas, dan staf Anda. Penyedia solusi cloud harus memiliki platform manajemen easy-to-integrate identitas yang kuat yang memungkinkan Anda menggabungkan identitas melalui penyedia identitas cloud ke aplikasi yang ada, solusi SaaS, dan layanan cloud Anda. Diagram berikut menunjukkan contoh arsitektur.



Arsitektur ini mengikuti rekomendasi ini:

- Pilih penyedia cloud utama dan strategis.Arsitektur ini digunakan AWS sebagai penyedia cloud utama. Dengan mengintegrasikan dengan penyedia identitas cloud dan manajemen identitas

dan layanan direktori yang ada di tempat, arsitektur ini mendukung penyediaan otomatis dan manajemen akses baik ke layanan penyedia cloud utama dan ke aplikasi lain dan solusi SaaS. Ini memastikan bahwa persyaratan keamanan dan tata kelola terpenuhi dengan cara yang konsisten dan mudah dikelola karena lebih banyak aplikasi dan layanan ditambahkan ke portofolio teknologi lembaga.

- Membedakan antara aplikasi SaaS dan layanan cloud dasar.Arsitektur ini mengintegrasikan beberapa jenis sistem identitas berbasis cloud, SaaS, dan lokal untuk menyediakan akses ke layanan dan aplikasi lainnya. AWS Cloud Banyak penyedia identitas berbasis cloud dan solusi masuk tunggal juga merupakan aplikasi SaaS, dan mereka dapat menggunakan integrasi asli dan protokol standar seperti SALL untuk bekerja di seluruh lingkungan.
- Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud.Arsitektur ini menganut panduan tentang identitas dan manajemen akses yang dikeluarkan oleh berbagai kerangka keamanan, termasuk National Institute of Standards and Technology (NIST) Cybersecurity Framework (CSF), NIST 800-171, dan NIST 800-53. Integrasi dengan AWS Organizations, AWS Identity and Access Management (IAM), dan layanan AWS keamanan, identitas, dan kepatuhan lainnya membantu menyediakan kontrol akses terperinci yang aman berdasarkan izin grup.
- Mengadopsi layanan cloud-native, terkelola sedapat mungkin dan praktis.Arsitektur ini menggunakan layanan terkelola berbasis cloud untuk manajemen identitas dan sistem masuk tunggal. Ini mengurangi waktu dan energi yang dihabiskan untuk manajemen infrastruktur dan membuatnya lebih mudah untuk mempertahankan sistem kritis ini.
- Menerapkan arsitektur hybrid ketika investasi lokal yang ada memberi insentif kepada penggunaan berkelanjutan.Arsitektur ini mengintegrasikan investasi lokal yang ada dalam infrastruktur untuk hosting Active Directory, Lightweight Directory Access Control (LDAP), dan beban kerja Shibboleth, dan menyediakan jalur untuk akhirnya memindahkan layanan identitas inti ke infrastruktur berbasis cloud. Selain itu, jika beban kerja lokal Anda memerlukan akses berbasis sertifikat ke AWS sumber daya, Anda dapat menggunakan Peran Di Mana Saja.AWS Identity and Access Management

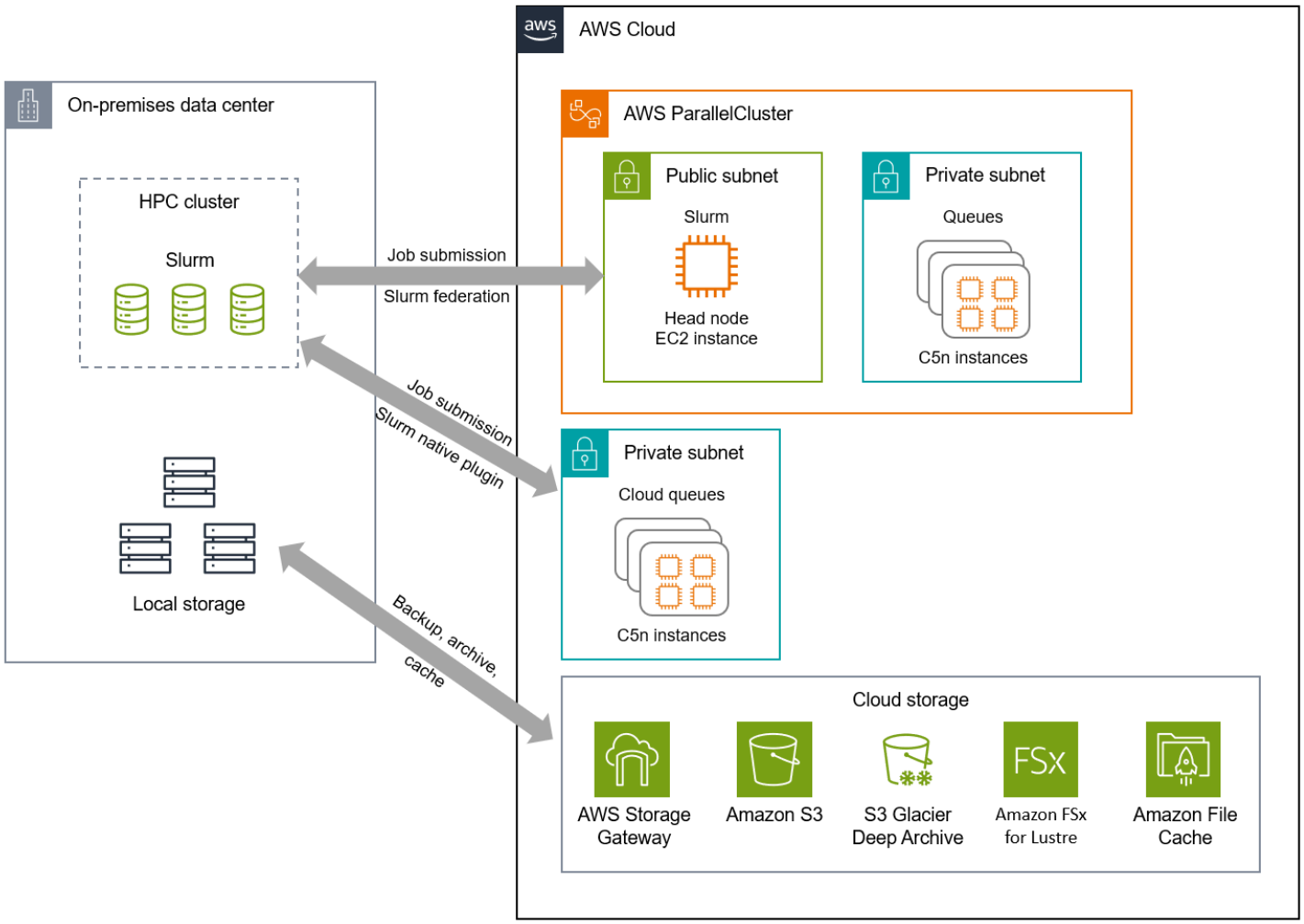
Cloud bursting untuk komputasi penelitian

Kelompok komputasi riset di lembaga penelitian R1 (Doctoral Universities — Very High Research Activity) di AS telah menjalankan cluster komputasi kinerja tinggi (HPC) lokal dengan penjadwal Slurm selama bertahun-tahun. Kecuali untuk beberapa minggu pemeliharaan terjadwal, cluster berjalan pada tingkat pemanfaatan 80-95 persen dengan sebagian besar antrian mereka penuh.

Meningkatnya jumlah kegiatan penelitian di lembaga memperkenalkan tantangan kapasitas dan kemampuan. Beberapa peneliti profil tinggi selalu melakukan simulasi jangka panjang pada antrian tertentu, yang meningkatkan waktu tunggu untuk pengguna lain. Fakultas yang baru dipekerjakan perlu menjalankan sejumlah besar simulasi cuaca untuk membangun model kecerdasan buatan dan pembelajaran mesin (AI/ML) baru untuk prakiraan cuaca, tetapi mereka membutuhkan kapasitas lebih dari yang tersedia. Kelompok komputasi riset juga mendapatkan lebih banyak permintaan untuk unit pemrosesan grafis terbaru (GPUs) untuk melatih model pembelajaran mesin. Bahkan dengan pendanaan untuk yang baru GPUs, tim perlu menunggu berbulan-bulan untuk mendapatkan persetujuan untuk memperluas ruang rak di pusat data.

Banyak peneliti tidak mau menghapus data lama, sehingga kapasitas penyimpanan lokal juga menjadi tantangan. Opsi penyimpanan jangka panjang yang lebih skalabel diperlukan untuk membebaskan penyimpanan yang berharga dan berkinerja tinggi di tempat.

Cloud mengatasi tantangan ini dengan solusi komputasi dan penyimpanan hybrid yang memungkinkan Anda menjalankan komputasi riset ke cloud saat kapasitas lokal tidak cukup. Diagram arsitektur berikut menggambarkan beberapa pendekatan ledakan komputasi dan penyimpanan, menggunakan alat seperti dan. AWS ParallelClusterAWS Storage Gateway



Arsitektur ini mengikuti rekomendasi ini:

- Pilih penyedia cloud utama dan strategis.Arsitektur ini menggunakan satu penyedia cloud utama untuk menghindari pembatasan oleh pendekatan penyebut yang paling tidak umum. Dengan cara ini, institusi dapat memanfaatkan inovasi dan layanan komputasi dan penyimpanan asli yang ditawarkan penyedia cloud utama. Tim komputasi riset dapat fokus pada pengoptimalan beban kerja di lingkungan yang disediakan oleh penyedia cloud utama, bukan bagaimana bekerja di lingkungan cloud yang berbeda.
- Menetapkan persyaratan keamanan dan tata kelola untuk setiap penyedia layanan cloud.Setiap layanan dan alat yang digunakan dalam arsitektur ini dapat dikonfigurasi untuk memenuhi persyaratan keamanan dan tata kelola tim komputasi riset, termasuk konektivitas pribadi, enkripsi data dalam perjalanan dan saat istirahat, pencatatan aktivitas, dan banyak lagi.
- Mengadopsi layanan cloud-native, terkelola sedapat mungkin dan praktis.Arsitektur ini menyediakan kemampuan untuk menggunakan penyimpanan terkelola dan layanan komputasi

serta alat untuk menyederhanakan manajemen cluster. Dengan cara ini, tim komputasi riset tidak perlu khawatir tentang mengelola cluster atau infrastruktur yang mendasarinya sendiri, yang dapat menjadi kompleks dan memakan waktu.

- Menerapkan arsitektur hybrid ketika investasi lokal yang ada memberi insentif kepada penggunaan berkelanjutan.Arsitektur ini memungkinkan institusi untuk terus menggunakan sumber daya lokal dan memanfaatkan cloud untuk meningkatkan kapasitas dan memperluas daya komputasi sesuai permintaan. Dengan cloud, institusi dapat mengukur jenis komputasi dengan tepat untuk memaksimalkan kinerja harga dan mengakses teknologi terbaru untuk mempromosikan inovasi tanpa investasi awal yang besar dalam perangkat keras lokal tambahan.

Langkah selanjutnya

Memilih model penerapan yang tepat untuk beban kerja cloud memerlukan pertimbangan yang cermat. Gunakan rekomendasi yang diuraikan dalam paper ini untuk memandu pengambilan keputusan Anda dan untuk menghindari jebakan umum seperti kompleksitas yang tidak perlu, meningkatnya tuntutan staf, tata kelola yang tidak konsisten, dan pendekatan common denominator terendah. Dengan mengikuti praktik terbaik ini, Anda dapat mempercepat adopsi cloud Anda untuk memenuhi dan melampaui tujuan kelembagaan Anda secara lebih efektif.

Ingatlah untuk memilih penyedia cloud utama dan strategis, dan mendirikan Cloud Center of Excellence (CCoE) untuk membantu mendorong kematangan organisasi guna memastikan kesuksesan jangka panjang Anda. Bedakan antara aplikasi SaaS dan layanan cloud dasar, dan identifikasi persyaratan keamanan dan tata kelola inti untuk masing-masing aplikasi. Jika memungkinkan, adopsi layanan cloud-native, terkelola, dan terapkan arsitektur hibrid saat investasi pusat data Anda yang ada memberi insentif untuk terus digunakan. Terakhir, pesan multicloud hanya untuk beban kerja yang benar-benar membutuhkannya.

AWS diposisikan dengan baik untuk membantu Anda mengelola lingkungan tunggal, hibrida, dan multicloud. Institusi Anda dapat menggunakan solusi AWS manajemen dan observabilitas seperti AWS Systems Manager, AWS Config, dan Amazon CloudWatch untuk menyederhanakan dan memusatkan pengelolaan dan pemantauan infrastruktur dan aplikasi Anda, terlepas dari lingkungan Anda. Dengan layanan data dan analitik seperti Amazon Athena, dan AWS GlueAWS DataSync, Anda dapat memperoleh wawasan dari semua data Anda, di mana pun data tersebut disimpan. Solusi hybrid seperti AWS Outposts, AWS Wavelength, dan AWS Snow Familymemungkinkan Anda membawa AWS infrastruktur dan layanan ke mana pun mereka dibutuhkan. Alat seperti Amazon EKS Distro membantu Anda membangun cluster Kubernetes yang dikelola sendiri di, di tempat AWS, atau di cloud lain.

Saat Anda menentukan strategi cloud Anda, pertimbangkan langkah-langkah berikut:

1. Tinjau AWS Cloud Adoption Framework (AWS CAF) untuk mengidentifikasi dan memprioritaskan peluang transformasi, mengevaluasi dan meningkatkan kesiapan cloud Anda, dan mengembangkan peta jalan transformasi Anda secara berulang.
2. Identifikasi sistem untuk implementasi cloud untuk memulai sebagai bukti konsep. Ini akan membantu Anda menentukan fondasi cloud atau kerangka kerja untuk memvalidasi asumsi apa pun, dan juga akan memungkinkan implementasi cloud masa depan.

3. Libatkan tim AWS akun Anda untuk mendiskusikan sasaran implementasi cloud Anda. Tim AWS akun dapat membantu memberikan klarifikasi, menyarankan pendekatan, mengidentifikasi dependensi, dan juga bekerja dengan tim Anda untuk memetakan perjalanan Anda dari konsep awal hingga implementasi.

Kontributor

Kontributor untuk panduan ini meliputi:

- Kevin Arand, Manajer Senior, Arsitektur Solusi, Pendidikan, AWS
- Kevin McCandless, Arsitek Solusi Senior, Pendidikan K-12, AWS
- Craig Yordania, Arsitek Solusi Utama, Pendidikan, AWS
- Jesse Roberts, Arsitek Solusi Utama, Pendidikan SLG & K-12, AWS
- Jianjun Xu, Arsitek Solusi Utama, Pendidikan, AWS
- Josh Badal, Arsitek Solusi Senior, Pendidikan, AWS
- Raj Chery, Arsitek Solusi Senior, Pendidikan, AWS

Sumber bacaan lebih lanjut

Untuk informasi tambahan, baca:

- AWS Pusat Arsitektur
- Transformasi Cloud Sektor Publik
- AWS Kerangka Adopsi Cloud (AWS CAF)
- AWS Solusi untuk Hybrid dan Multicloud

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	—	15 September 2023

AWS Glosarium Panduan Preskriptif

Berikut ini adalah istilah yang umum digunakan dalam strategi, panduan, dan pola yang disediakan oleh Panduan AWS Preskriptif. Untuk menyarankan entri, silakan gunakan tautan Berikan umpan balik di akhir glosarium.

Nomor

7 Rs

Tujuh strategi migrasi umum untuk memindahkan aplikasi ke cloud. Strategi ini dibangun di atas 5 Rs yang diidentifikasi Gartner pada tahun 2011 dan terdiri dari yang berikut:

- Refactor/Re-Architect — Memindahkan aplikasi dan memodifikasi arsitekturnya dengan memanfaatkan sepenuhnya fitur cloud-native untuk meningkatkan kelincahan, kinerja, dan skalabilitas. Ini biasanya melibatkan porting sistem operasi dan database. Contoh: Migrasikan database Oracle lokal Anda ke Amazon Aurora PostgreSQL Compatible Edition.
- Replatform (angkat dan bentuk ulang) — Pindahkan aplikasi ke cloud, dan perkenalkan beberapa tingkat pengoptimalan untuk memanfaatkan kemampuan cloud. Contoh: Memigrasikan database Oracle lokal Anda ke Amazon Relational Database Service (Amazon RDS) untuk Oracle di. AWS Cloud
- Pembelian kembali (drop and shop) - Beralih ke produk yang berbeda, biasanya dengan beralih dari lisensi tradisional ke model SaaS. Contoh: Migrasikan sistem manajemen hubungan pelanggan (CRM) Anda ke Salesforce.com.
- Rehost (lift dan shift) — Pindahkan aplikasi ke cloud tanpa membuat perubahan apa pun untuk memanfaatkan kemampuan cloud. Contoh: Migrasikan database Oracle lokal Anda ke Oracle pada instans EC2 di. AWS Cloud
- Relokasi (hypervisor-level lift and shift) — Pindahkan infrastruktur ke cloud tanpa membeli perangkat keras baru, menulis ulang aplikasi, atau memodifikasi operasi yang ada. Anda memigrasikan server dari platform lokal ke layanan cloud untuk platform yang sama. Contoh: Migrasikan Microsoft Hyper-V aplikasi ke AWS.
- Pertahankan (kunjungi kembali) - Simpan aplikasi di lingkungan sumber Anda. Ini mungkin termasuk aplikasi yang memerlukan refactoring besar, dan Anda ingin menunda pekerjaan itu sampai nanti, dan aplikasi lama yang ingin Anda pertahankan, karena tidak ada pembenaran bisnis untuk memigrasikannya.

- Pensiun — Menonaktifkan atau menghapus aplikasi yang tidak lagi diperlukan di lingkungan sumber Anda.

A

ABAC

Lihat kontrol akses berbasis atribut.

layanan abstrak

Lihat layanan terkelola.

ASAM

Lihat atomisitas, konsistensi, isolasi, daya tahan.

migrasi aktif-aktif

Metode migrasi database di mana database sumber dan target tetap sinkron (dengan menggunakan alat replikasi dua arah atau operasi penulisan ganda), dan kedua database menangani transaksi dari menghubungkan aplikasi selama migrasi. Metode ini mendukung migrasi dalam batch kecil yang terkontrol alih-alih memerlukan pemotongan satu kali. Ini lebih fleksibel tetapi membutuhkan lebih banyak pekerjaan daripada migrasi aktif-pasif.

migrasi aktif-pasif

Metode migrasi database di mana database sumber dan target disimpan dalam sinkron, tetapi hanya database sumber yang menangani transaksi dari menghubungkan aplikasi sementara data direplikasi ke database target. Basis data target tidak menerima transaksi apa pun selama migrasi.

fungsi agregat

Fungsi SQL yang beroperasi pada sekelompok baris dan menghitung nilai pengembalian tunggal untuk grup. Contoh fungsi agregat meliputi `SUM` dan`MAX`.

AI

Lihat kecerdasan buatan.

AIOps

Lihat operasi kecerdasan buatan.

anonimisasi

Proses menghapus informasi pribadi secara permanen dalam kumpulan data. Anonimisasi dapat membantu melindungi privasi pribadi. Data anonim tidak lagi dianggap sebagai data pribadi.

anti-pola

Solusi yang sering digunakan untuk masalah berulang di mana solusinya kontra-produktif, tidak efektif, atau kurang efektif daripada alternatif.

kontrol aplikasi

Pendekatan keamanan yang memungkinkan penggunaan hanya aplikasi yang disetujui untuk membantu melindungi sistem dari malware.

portofolio aplikasi

Kumpulan informasi rinci tentang setiap aplikasi yang digunakan oleh organisasi, termasuk biaya untuk membangun dan memelihara aplikasi, dan nilai bisnisnya. Informasi ini adalah kunci untuk penemuan portofolio dan proses analisis dan membantu mengidentifikasi dan memprioritaskan aplikasi yang akan dimigrasi, dimodernisasi, dan dioptimalkan.

kecerdasan buatan (AI)

Bidang ilmu komputer yang didedikasikan untuk menggunakan teknologi komputasi untuk melakukan fungsi kognitif yang biasanya terkait dengan manusia, seperti belajar, memecahkan masalah, dan mengenali pola. Untuk informasi lebih lanjut, lihat Apa itu Kecerdasan Buatan?

operasi kecerdasan buatan (AIOps)

Proses menggunakan teknik pembelajaran mesin untuk memecahkan masalah operasional, mengurangi insiden operasional dan intervensi manusia, dan meningkatkan kualitas layanan. Untuk informasi selengkapnya tentang cara AIOps digunakan dalam strategi AWS migrasi, lihat panduan integrasi operasi.

enkripsi asimetris

Algoritma enkripsi yang menggunakan sepasang kunci, kunci publik untuk enkripsi dan kunci pribadi untuk dekripsi. Anda dapat berbagi kunci publik karena tidak digunakan untuk dekripsi, tetapi akses ke kunci pribadi harus sangat dibatasi.

atomisitas, konsistensi, isolasi, daya tahan (ACID)

Satu set properti perangkat lunak yang menjamin validitas data dan keandalan operasional database, bahkan dalam kasus kesalahan, kegagalan daya, atau masalah lainnya.

kontrol akses berbasis atribut (ABAC)

Praktik membuat izin berbutir halus berdasarkan atribut pengguna, seperti departemen, peran pekerjaan, dan nama tim. Untuk informasi selengkapnya, lihat ABAC untuk AWS dokumentasi AWS Identity and Access Management (IAM).

sumber data otoritatif

Lokasi di mana Anda menyimpan versi utama data, yang dianggap sebagai sumber informasi yang paling dapat diandalkan. Anda dapat menyalin data dari sumber data otoritatif ke lokasi lain untuk tujuan pemrosesan atau modifikasi data, seperti menganonimkan, menyunting, atau membuat nama samaran.

Zona Ketersediaan

Lokasi berbeda di dalam Wilayah AWS yang terisolasi dari kegagalan di Availability Zone lainnya dan menyediakan konektivitas jaringan latensi rendah yang murah ke Availability Zone lainnya di Wilayah yang sama.

AWS Kerangka Adopsi Cloud (AWS CAF)

Kerangka pedoman dan praktik terbaik AWS untuk membantu organisasi mengembangkan rencana yang efisien dan efektif untuk bergerak dengan sukses ke cloud. AWS CAF mengatur panduan ke dalam enam area fokus yang disebut perspektif: bisnis, orang, tata kelola, platform, keamanan, dan operasi. Perspektif bisnis, orang, dan tata kelola fokus pada keterampilan dan proses bisnis; perspektif platform, keamanan, dan operasi fokus pada keterampilan dan proses teknis. Misalnya, perspektif masyarakat menargetkan pemangku kepentingan yang menangani sumber daya manusia (SDM), fungsi kepegawaian, dan manajemen orang. Untuk perspektif ini, AWS CAF memberikan panduan untuk pengembangan, pelatihan, dan komunikasi orang untuk membantu mempersiapkan organisasi untuk adopsi cloud yang sukses. Untuk informasi lebih lanjut, lihat situs web AWS CAF dan whitepaper AWS CAF.

AWS Kerangka Kualifikasi Beban Kerja (AWS WQF)

Alat yang mengevaluasi beban kerja migrasi database, merekomendasikan strategi migrasi, dan memberikan perkiraan kerja. AWS WQF disertakan dengan AWS Schema Conversion Tool ()AWS SCT. Ini menganalisis skema database dan objek kode, kode aplikasi, dependensi, dan karakteristik kinerja, dan memberikan laporan penilaian.

B

bot buruk

Bot yang dimaksudkan untuk mengganggu atau menyebabkan kerugian bagi individu atau organisasi.

BCP

Lihat perencanaan kontinuitas bisnis.

grafik perilaku

Pandangan interaktif yang terpadu tentang perilaku dan interaksi sumber daya dari waktu ke waktu. Anda dapat menggunakan grafik perilaku dengan Amazon Detective untuk memeriksa upaya logon yang gagal, panggilan API yang mencurigakan, dan tindakan serupa. Untuk informasi selengkapnya, lihat Data dalam grafik perilaku di dokumentasi Detektif.

sistem big-endian

Sistem yang menyimpan byte paling signifikan terlebih dahulu. Lihat juga endianness.

klasifikasi biner

Sebuah proses yang memprediksi hasil biner (salah satu dari dua kelas yang mungkin). Misalnya, model ML Anda mungkin perlu memprediksi masalah seperti “Apakah email ini spam atau bukan spam?” atau “Apakah produk ini buku atau mobil?”

filter mekar

Struktur data probabilistik dan efisien memori yang digunakan untuk menguji apakah suatu elemen adalah anggota dari suatu himpunan.

deployment biru/hijau

Strategi penyebaran tempat Anda membuat dua lingkungan yang terpisah namun identik. Anda menjalankan versi aplikasi saat ini di satu lingkungan (biru) dan versi aplikasi baru di lingkungan lain (hijau). Strategi ini membantu Anda dengan cepat memutar kembali dengan dampak minimal.

bot

Aplikasi perangkat lunak yang menjalankan tugas otomatis melalui internet dan mensimulasikan aktivitas atau interaksi manusia. Beberapa bot berguna atau bermanfaat, seperti perayap web yang mengindeks informasi di internet. Beberapa bot lain, yang dikenal sebagai bot buruk, dimaksudkan untuk mengganggu atau membahayakan individu atau organisasi.

botnet

Jaringan bot yang terinfeksi oleh malware dan berada di bawah kendali satu pihak, yang dikenal sebagai bot herder atau operator bot. Botnet adalah mekanisme paling terkenal untuk skala bot dan dampaknya.

cabang

Area berisi repositori kode. Cabang pertama yang dibuat dalam repositori adalah cabang utama. Anda dapat membuat cabang baru dari cabang yang ada, dan Anda kemudian dapat mengembangkan fitur atau memperbaiki bug di cabang baru. Cabang yang Anda buat untuk membangun fitur biasanya disebut sebagai cabang fitur. Saat fitur siap dirilis, Anda menggabungkan cabang fitur kembali ke cabang utama. Untuk informasi selengkapnya, lihat Tentang cabang (GitHub dokumentasi).

akses break-glass

Dalam keadaan luar biasa dan melalui proses yang disetujui, cara cepat bagi pengguna untuk mendapatkan akses ke Akun AWS yang biasanya tidak memiliki izin untuk mengaksesnya. Untuk informasi lebih lanjut, lihat indikator Implementasikan prosedur break-glass dalam panduan Well-Architected AWS .

strategi brownfield

Infrastruktur yang ada di lingkungan Anda. Saat mengadopsi strategi brownfield untuk arsitektur sistem, Anda merancang arsitektur di sekitar kendala sistem dan infrastruktur saat ini. Jika Anda memperluas infrastruktur yang ada, Anda dapat memadukan strategi brownfield dan greenfield.

cache penyangga

Area memori tempat data yang paling sering diakses disimpan.

kemampuan bisnis

Apa yang dilakukan bisnis untuk menghasilkan nilai (misalnya, penjualan, layanan pelanggan, atau pemasaran). Arsitektur layanan mikro dan keputusan pengembangan dapat didorong oleh kemampuan bisnis. Untuk informasi selengkapnya, lihat bagian Terorganisir di sekitar kemampuan bisnis dari Menjalankan layanan mikro kontainer di whitepaper. AWS

perencanaan kelangsungan bisnis (BCP)

Rencana yang membahas dampak potensial dari peristiwa yang mengganggu, seperti migrasi skala besar, pada operasi dan memungkinkan bisnis untuk melanjutkan operasi dengan cepat.

C

KAFE

Lihat Kerangka Adopsi AWS Cloud.

penyebaran kenari

Rilis versi yang lambat dan bertahap untuk pengguna akhir. Ketika Anda yakin, Anda menyebarkan versi baru dan mengganti versi saat ini secara keseluruhan.

CCoE

Lihat Cloud Center of Excellence.

CDC

Lihat mengubah pengambilan data.

ubah pengambilan data (CDC)

Proses melacak perubahan ke sumber data, seperti tabel database, dan merekam metadata tentang perubahan tersebut. Anda dapat menggunakan CDC untuk berbagai tujuan, seperti mengaudit atau mereplikasi perubahan dalam sistem target untuk mempertahankan sinkronisasi.

rekayasa kekacauan

Sengaja memperkenalkan kegagalan atau peristiwa yang mengganggu untuk menguji ketahanan sistem. Anda dapat menggunakan AWS Fault Injection Service (AWS FIS) untuk melakukan eksperimen yang menekankan AWS beban kerja Anda dan mengevaluasi responsnya.

CI/CD

Lihat integrasi berkelanjutan dan pengiriman berkelanjutan.

klasifikasi

Proses kategorisasi yang membantu menghasilkan prediksi. Model ML untuk masalah klasifikasi memprediksi nilai diskrit. Nilai diskrit selalu berbeda satu sama lain. Misalnya, model mungkin perlu mengevaluasi apakah ada mobil dalam gambar atau tidak.

Enkripsi sisi klien

Enkripsi data secara lokal, sebelum target Layanan AWS menerimanya.

Pusat Keunggulan Cloud (CCoE)

Tim multi-disiplin yang mendorong upaya adopsi cloud di seluruh organisasi, termasuk mengembangkan praktik terbaik cloud, memobilisasi sumber daya, menetapkan jadwal migrasi, dan memimpin organisasi melalui transformasi skala besar. Untuk informasi selengkapnya, lihat posting CCo E di Blog Strategi AWS Cloud Perusahaan.

komputasi cloud

Teknologi cloud yang biasanya digunakan untuk penyimpanan data jarak jauh dan manajemen perangkat IoT. Cloud computing umumnya terhubung ke teknologi edge computing.

model operasi cloud

Dalam organisasi TI, model operasi yang digunakan untuk membangun, mematangkan, dan mengoptimalkan satu atau lebih lingkungan cloud. Untuk informasi selengkapnya, lihat Membangun Model Operasi Cloud Anda.

tahap adopsi cloud

Empat fase yang biasanya dilalui organisasi ketika mereka bermigrasi ke AWS Cloud:

- Proyek — Menjalankan beberapa proyek terkait cloud untuk bukti konsep dan tujuan pembelajaran
- Foundation — Melakukan investasi dasar untuk meningkatkan adopsi cloud Anda (misalnya, membuat landing zone, mendefinisikan CCo E, membuat model operasi)
- Migrasi — Migrasi aplikasi individual
- Re-invention — Mengoptimalkan produk dan layanan, dan berinovasi di cloud

Tahapan ini didefinisikan oleh Stephen Orban dalam posting blog The Journey Toward Cloud-First & the Stages of Adoption di blog Strategi Perusahaan. AWS Cloud Untuk informasi tentang bagaimana kaitannya dengan strategi AWS migrasi, lihat panduan kesiapan migrasi.

CMDB

Lihat database manajemen konfigurasi.

repositori kode

Lokasi di mana kode sumber dan aset lainnya, seperti dokumentasi, sampel, dan skrip, disimpan dan diperbarui melalui proses kontrol versi. Repositori cloud umum termasuk GitHub atau. Bitbucket Cloud Setiap versi kode disebut cabang. Dalam struktur layanan mikro, setiap repositori

dikhususkan untuk satu bagian fungsionalitas. Pipa CI/CD tunggal dapat menggunakan beberapa repositori.

cache dingin

Cache buffer yang kosong, tidak terisi dengan baik, atau berisi data basi atau tidak relevan. Ini mempengaruhi kinerja karena instance database harus membaca dari memori utama atau disk, yang lebih lambat daripada membaca dari cache buffer.

data dingin

Data yang jarang diakses dan biasanya historis. Saat menanyakan jenis data ini, kueri lambat biasanya dapat diterima. Memindahkan data ini ke tingkat penyimpanan atau kelas yang berkinerja lebih rendah dan lebih murah dapat mengurangi biaya.

visi komputer (CV)

Bidang AI yang menggunakan pembelajaran mesin untuk menganalisis dan mengekstrak informasi dari format visual seperti gambar dan video digital. Misalnya, Amazon SageMaker AI menyediakan algoritma pemrosesan gambar untuk CV.

konfigurasi drift

Untuk beban kerja, konfigurasi berubah dari status yang diharapkan. Ini dapat menyebabkan beban kerja menjadi tidak patuh, dan biasanya bertahap dan tidak disengaja.

database manajemen konfigurasi (CMDB)

Repositori yang menyimpan dan mengelola informasi tentang database dan lingkungan TI, termasuk komponen perangkat keras dan perangkat lunak dan konfigurasinya. Anda biasanya menggunakan data dari CMDB dalam penemuan portofolio dan tahap analisis migrasi.

paket kesesuaian

Kumpulan AWS Config aturan dan tindakan remediasi yang dapat Anda kumpulkan untuk menyesuaikan kepatuhan dan pemeriksaan keamanan Anda. Anda dapat menerapkan paket kesesuaian sebagai entitas tunggal di Akun AWS dan Wilayah, atau di seluruh organisasi, dengan menggunakan templat YAMM. Untuk informasi selengkapnya, lihat Paket kesesuaian dalam dokumentasi. AWS Config

integrasi berkelanjutan dan pengiriman berkelanjutan (CI/CD)

Proses mengotomatiskan sumber, membangun, menguji, pementasan, dan tahap produksi dari proses rilis perangkat lunak. CI/CD biasanya digambarkan sebagai pipa. CI/CD dapat membantu

Anda mengotomatiskan proses, meningkatkan produktivitas, meningkatkan kualitas kode, dan memberikan lebih cepat. Untuk informasi lebih lanjut, lihat Manfaat pengiriman berkelanjutan. CD juga dapat berarti penerapan berkelanjutan. Untuk informasi selengkapnya, lihat Continuous Delivery vs Continuous Deployment.

CV

Lihat visi komputer.

D

data saat istirahat

Data yang stasioner di jaringan Anda, seperti data yang ada di penyimpanan.

klasifikasi data

Proses untuk mengidentifikasi dan mengkategorikan data dalam jaringan Anda berdasarkan kekritisan dan sensitivitasnya. Ini adalah komponen penting dari setiap strategi manajemen risiko keamanan siber karena membantu Anda menentukan perlindungan dan kontrol retensi yang tepat untuk data. Klasifikasi data adalah komponen pilar keamanan dalam AWS Well-Architected Framework. Untuk informasi selengkapnya, lihat Klasifikasi data.

penyimpangan data

Variasi yang berarti antara data produksi dan data yang digunakan untuk melatih model ML, atau perubahan yang berarti dalam data input dari waktu ke waktu. Penyimpangan data dapat mengurangi kualitas, akurasi, dan keadilan keseluruhan dalam prediksi model ML.

data dalam transit

Data yang aktif bergerak melalui jaringan Anda, seperti antara sumber daya jaringan.

jala data

Kerangka arsitektur yang menyediakan kepemilikan data terdistribusi dan terdesentralisasi dengan manajemen dan tata kelola terpusat.

minimalisasi data

Prinsip pengumpulan dan pemrosesan hanya data yang sangat diperlukan. Mempraktikkan minimalisasi data di dalamnya AWS Cloud dapat mengurangi risiko privasi, biaya, dan jejak karbon analitik Anda.

perimeter data

Satu set pagar pembatas pencegahan di AWS lingkungan Anda yang membantu memastikan bahwa hanya identitas tepercaya yang mengakses sumber daya tepercaya dari jaringan yang diharapkan. Untuk informasi selengkapnya, lihat Membangun perimeter data pada AWS.

prapemrosesan data

Untuk mengubah data mentah menjadi format yang mudah diuraikan oleh model ML Anda. Preprocessing data dapat berarti menghapus kolom atau baris tertentu dan menangani nilai yang hilang, tidak konsisten, atau duplikat.

asal data

Proses melacak asal dan riwayat data sepanjang siklus hidupnya, seperti bagaimana data dihasilkan, ditransmisikan, dan disimpan.

subjek data

Individu yang datanya dikumpulkan dan diproses.

gudang data

Sistem manajemen data yang mendukung intelijen bisnis, seperti analitik. Gudang data biasanya berisi sejumlah besar data historis, dan biasanya digunakan untuk kueri dan analisis.

bahasa definisi database (DDL)

Pernyataan atau perintah untuk membuat atau memodifikasi struktur tabel dan objek dalam database.

bahasa manipulasi basis data (DHTML)

Pernyataan atau perintah untuk memodifikasi (memasukkan, memperbarui, dan menghapus) informasi dalam database.

DDL

Lihat bahasa definisi database.

ansambel yang dalam

Untuk menggabungkan beberapa model pembelajaran mendalam untuk prediksi. Anda dapat menggunakan ansambel dalam untuk mendapatkan prediksi yang lebih akurat atau untuk memperkirakan ketidakpastian dalam prediksi.

pembelajaran mendalam

Subbidang ML yang menggunakan beberapa lapisan jaringan saraf tiruan untuk mengidentifikasi pemetaan antara data input dan variabel target yang diinginkan.

defense-in-depth

Pendekatan keamanan informasi di mana serangkaian mekanisme dan kontrol keamanan dilapisi dengan cermat di seluruh jaringan komputer untuk melindungi kerahasiaan, integritas, dan ketersediaan jaringan dan data di dalamnya. Saat Anda mengadopsi strategi ini AWS, Anda menambahkan beberapa kontrol pada lapisan AWS Organizations struktur yang berbeda untuk membantu mengamankan sumber daya. Misalnya, defense-in-depth pendekatan mungkin menggabungkan otentikasi multi-faktor, segmentasi jaringan, dan enkripsi.

administrator yang didelegasikan

Di AWS Organizations, layanan yang kompatibel dapat mendaftarkan akun AWS anggota untuk mengelola akun organisasi dan mengelola izin untuk layanan tersebut. Akun ini disebut administrator yang didelegasikan untuk layanan itu. Untuk informasi selengkapnya dan daftar layanan yang kompatibel, lihat Layanan yang berfungsi dengan AWS Organizations AWS Organizations dokumentasi.

deployment

Proses pembuatan aplikasi, fitur baru, atau perbaikan kode tersedia di lingkungan target. Deployment melibatkan penerapan perubahan dalam basis kode dan kemudian membangun dan menjalankan basis kode itu di lingkungan aplikasi.

lingkungan pengembangan

Lihat lingkungan.

kontrol detektif

Kontrol keamanan yang dirancang untuk mendeteksi, mencatat, dan memperingatkan setelah suatu peristiwa terjadi. Kontrol ini adalah garis pertahanan kedua, memperingatkan Anda tentang peristiwa keamanan yang melewati kontrol pencegahan yang ada. Untuk informasi selengkapnya, lihat Kontrol Detektif dalam Menerapkan kontrol keamanan pada. AWS

pemetaan aliran nilai pengembangan (DVSM)

Sebuah proses yang digunakan untuk mengidentifikasi dan memprioritaskan kendala yang mempengaruhi kecepatan dan kualitas dalam siklus hidup pengembangan perangkat lunak. DVSM memperluas proses pemetaan aliran nilai yang awalnya dirancang untuk praktik

manufaktur ramping. Ini berfokus pada langkah-langkah dan tim yang diperlukan untuk menciptakan dan memindahkan nilai melalui proses pengembangan perangkat lunak.

kembar digital

Representasi virtual dari sistem dunia nyata, seperti bangunan, pabrik, peralatan industri, atau jalur produksi. Kembar digital mendukung pemeliharaan prediktif, pemantauan jarak jauh, dan optimalisasi produksi.

tabel dimensi

Dalam skema bintang, tabel yang lebih kecil yang berisi atribut data tentang data kuantitatif dalam tabel fakta. Atribut tabel dimensi biasanya bidang teks atau angka diskrit yang berperilaku seperti teks. Atribut ini biasanya digunakan untuk pembatasan kueri, pemfilteran, dan pelabelan set hasil.

musibah

Peristiwa yang mencegah beban kerja atau sistem memenuhi tujuan bisnisnya di lokasi utama yang digunakan. Peristiwa ini dapat berupa bencana alam, kegagalan teknis, atau akibat dari tindakan manusia, seperti kesalahan konfigurasi yang tidak disengaja atau serangan malware.

pemulihan bencana (DR)

Strategi dan proses yang Anda gunakan untuk meminimalkan downtime dan kehilangan data yang disebabkan oleh bencana. Untuk informasi selengkapnya, lihat Disaster Recovery of Workloads on AWS: Recovery in the Cloud in the AWS Well-Architected Framework.

DML~

Lihat bahasa manipulasi basis data.

desain berbasis domain

Pendekatan untuk mengembangkan sistem perangkat lunak yang kompleks dengan menghubungkan komponennya ke domain yang berkembang, atau tujuan bisnis inti, yang dilayani oleh setiap komponen. Konsep ini diperkenalkan oleh Eric Evans dalam bukunya, Domain-Driven Design: Tackling Complexity in the Heart of Software (Boston: Addison-Wesley Professional, 2003). Untuk informasi tentang cara menggunakan desain berbasis domain dengan pola gambar pencekik, lihat Memodernisasi layanan web Microsoft ASP.NET (ASMX) lama secara bertahap menggunakan container dan Amazon API Gateway.

DR

Lihat pemulihan bencana.

deteksi drift

Melacak penyimpangan dari konfigurasi dasar. Misalnya, Anda dapat menggunakan AWS CloudFormation untuk mendeteksi penyimpangan dalam sumber daya sistem, atau Anda dapat menggunakannya AWS Control Tower untuk mendeteksi perubahan di landing zone yang mungkin memengaruhi kepatuhan terhadap persyaratan tata kelola.

DVSM

Lihat pemetaan aliran nilai pengembangan.

E

EDA

Lihat analisis data eksplorasi.

EDI

Lihat pertukaran data elektronik.

komputasi tepi

Teknologi yang meningkatkan daya komputasi untuk perangkat pintar di tepi jaringan IoT. Jika dibandingkan dengan komputasi awan, komputasi tepi dapat mengurangi latensi komunikasi dan meningkatkan waktu respons.

pertukaran data elektronik (EDI)

Pertukaran otomatis dokumen bisnis antar organisasi. Untuk informasi selengkapnya, lihat Apa itu Pertukaran Data Elektronik.

enkripsi

Proses komputasi yang mengubah data plaintext, yang dapat dibaca manusia, menjadi ciphertext.

kunci enkripsi

String kriptografi dari bit acak yang dihasilkan oleh algoritma enkripsi. Panjang kunci dapat bervariasi, dan setiap kunci dirancang agar tidak dapat diprediksi dan unik.

endianness

Urutan byte disimpan dalam memori komputer. Sistem big-endian menyimpan byte paling signifikan terlebih dahulu. Sistem little-endian menyimpan byte paling tidak signifikan terlebih dahulu.

titik akhir

Lihat titik akhir layanan.

layanan endpoint

Layanan yang dapat Anda host di cloud pribadi virtual (VPC) untuk dibagikan dengan pengguna lain. Anda dapat membuat layanan endpoint dengan AWS PrivateLink dan memberikan izin kepada prinsipal lain Akun AWS atau ke AWS Identity and Access Management (IAM). Akun atau prinsipal ini dapat terhubung ke layanan endpoint Anda secara pribadi dengan membuat titik akhir VPC antarmuka. Untuk informasi selengkapnya, lihat Membuat layanan titik akhir di dokumentasi Amazon Virtual Private Cloud (Amazon VPC).

perencanaan sumber daya perusahaan (ERP)

Sistem yang mengotomatiskan dan mengelola proses bisnis utama (seperti akuntansi, MES, dan manajemen proyek) untuk suatu perusahaan.

enkripsi amplop

Proses mengenkripsi kunci enkripsi dengan kunci enkripsi lain. Untuk informasi selengkapnya, lihat Enkripsi amplop dalam dokumentasi AWS Key Management Service (AWS KMS).

lingkungan

Sebuah contoh dari aplikasi yang sedang berjalan. Berikut ini adalah jenis lingkungan yang umum dalam komputasi awan:

- Development Environment — Sebuah contoh dari aplikasi yang berjalan yang hanya tersedia untuk tim inti yang bertanggung jawab untuk memelihara aplikasi. Lingkungan pengembangan digunakan untuk menguji perubahan sebelum mempromosikannya ke lingkungan atas. Jenis lingkungan ini kadang-kadang disebut sebagai lingkungan pengujian.
- lingkungan yang lebih rendah — Semua lingkungan pengembangan untuk aplikasi, seperti yang digunakan untuk build awal dan pengujian.
- lingkungan produksi — Sebuah contoh dari aplikasi yang berjalan yang pengguna akhir dapat mengakses. Dalam sebuah CI/CD pipeline, lingkungan produksi adalah lingkungan penyebaran terakhir.

- lingkungan atas — Semua lingkungan yang dapat diakses oleh pengguna selain tim pengembangan inti. Ini dapat mencakup lingkungan produksi, lingkungan praproduksi, dan lingkungan untuk pengujian penerimaan pengguna.

epik

Dalam metodologi tangkas, kategori fungsional yang membantu mengatur dan memprioritaskan pekerjaan Anda. Epik memberikan deskripsi tingkat tinggi tentang persyaratan dan tugas implementasi. Misalnya, epos keamanan AWS CAF mencakup manajemen identitas dan akses, kontrol detektif, keamanan infrastruktur, perlindungan data, dan respons insiden. Untuk informasi selengkapnya tentang epos dalam strategi AWS migrasi, lihat panduan implementasi program.

ERP

Lihat perencanaan sumber daya perusahaan.

analisis data eksplorasi (EDA)

Proses menganalisis dataset untuk memahami karakteristik utamanya. Anda mengumpulkan atau mengumpulkan data dan kemudian melakukan penyelidikan awal untuk menemukan pola, mendeteksi anomali, dan memeriksa asumsi. EDA dilakukan dengan menghitung statistik ringkasan dan membuat visualisasi data.

F

tabel fakta

Tabel tengah dalam skema bintang. Ini menyimpan data kuantitatif tentang operasi bisnis. Biasanya, tabel fakta berisi dua jenis kolom: kolom yang berisi ukuran dan yang berisi kunci asing ke tabel dimensi.

gagal cepat

Filosofi yang menggunakan pengujian yang sering dan bertahap untuk mengurangi siklus hidup pengembangan. Ini adalah bagian penting dari pendekatan tangkas.

batas isolasi kesalahan

Dalam AWS Cloud, batas seperti Availability Zone, Wilayah AWS, control plane, atau data plane yang membatasi efek kegagalan dan membantu meningkatkan ketahanan beban kerja. Untuk informasi selengkapnya, lihat Batas Isolasi AWS Kesalahan.

cabang fitur

Lihat cabang.

fitur

Data input yang Anda gunakan untuk membuat prediksi. Misalnya, dalam konteks manufaktur, fitur bisa berupa gambar yang diambil secara berkala dari lini manufaktur.

pentingnya fitur

Seberapa signifikan fitur untuk prediksi model. Ini biasanya dinyatakan sebagai skor numerik yang dapat dihitung melalui berbagai teknik, seperti Shapley Additive Explanations (SHAP) dan gradien terintegrasi. Untuk informasi lebih lanjut, lihat Interpretabilitas model pembelajaran mesin dengan. AWS

transformasi fitur

Untuk mengoptimalkan data untuk proses ML, termasuk memperkaya data dengan sumber tambahan, menskalakan nilai, atau mengekstrak beberapa set informasi dari satu bidang data. Hal ini memungkinkan model ML untuk mendapatkan keuntungan dari data. Misalnya, jika Anda memecah tanggal “2021-05-27 00:15:37" menjadi “2021", “Mei”, “Kamis”, dan “15", Anda dapat membantu algoritme pembelajaran mempelajari pola bernuansa yang terkait dengan komponen data yang berbeda.

beberapa tembakan mendorong

Menyediakan LLM dengan sejumlah kecil contoh yang menunjukkan tugas dan output yang diinginkan sebelum memintanya untuk melakukan tugas serupa. Teknik ini adalah aplikasi pembelajaran dalam konteks, di mana model belajar dari contoh (bidikan) yang tertanam dalam petunjuk. Beberapa bidikan dapat efektif untuk tugas-tugas yang memerlukan pemformatan, penalaran, atau pengetahuan domain tertentu. Lihat juga zero-shot prompting.

FGAC

Lihat kontrol akses berbutir halus.

kontrol akses berbutir halus (FGAC)

Penggunaan beberapa kondisi untuk mengizinkan atau menolak permintaan akses.

migrasi flash-cut

Metode migrasi database yang menggunakan replikasi data berkelanjutan melalui pengambilan data perubahan untuk memigrasikan data dalam waktu sesingkat mungkin, alih-alih

menggunakan pendekatan bertahap. Tujuannya adalah untuk menjaga downtime seminimal mungkin.

FM

Lihat model pondasi.

model pondasi (FM)

Jaringan saraf pembelajaran mendalam yang besar yang telah melatih kumpulan data besar-besaran data umum dan tidak berlabel. FMs mampu melakukan berbagai tugas umum, seperti memahami bahasa, menghasilkan teks dan gambar, dan berbicara dalam bahasa alami. Untuk informasi selengkapnya, lihat Apa itu Model Foundation.

G

AI generatif

Subset model AI yang telah dilatih pada sejumlah besar data dan yang dapat menggunakan prompt teks sederhana untuk membuat konten dan artefak baru, seperti gambar, video, teks, dan audio. Untuk informasi lebih lanjut, lihat Apa itu AI Generatif.

pemblokiran geografis

Lihat pembatasan geografis.

pembatasan geografis (pemblokiran geografis)

Di Amazon CloudFront, opsi untuk mencegah pengguna di negara tertentu mengakses distribusi konten. Anda dapat menggunakan daftar izinkan atau daftar blokir untuk menentukan negara yang disetujui dan dilarang. Untuk informasi selengkapnya, lihat Membatasi distribusi geografis konten Anda dalam dokumentasi. CloudFront

Alur kerja Gitflow

Pendekatan di mana lingkungan bawah dan atas menggunakan cabang yang berbeda dalam repositori kode sumber. Alur kerja Gitflow dianggap warisan, dan alur kerja berbasis batang adalah pendekatan modern yang lebih disukai.

gambar emas

Sebuah snapshot dari sistem atau perangkat lunak yang digunakan sebagai template untuk menyebarkan instance baru dari sistem atau perangkat lunak itu. Misalnya, di bidang manufaktur,

gambar emas dapat digunakan untuk menyediakan perangkat lunak pada beberapa perangkat dan membantu meningkatkan kecepatan, skalabilitas, dan produktivitas dalam operasi manufaktur perangkat.

strategi greenfield

Tidak adanya infrastruktur yang ada di lingkungan baru. Saat mengadopsi strategi greenfield untuk arsitektur sistem, Anda dapat memilih semua teknologi baru tanpa batasan kompatibilitas dengan infrastruktur yang ada, juga dikenal sebagai brownfield. Jika Anda memperluas infrastruktur yang ada, Anda dapat memadukan strategi brownfield dan greenfield.

pagar pembatas

Aturan tingkat tinggi yang membantu mengatur sumber daya, kebijakan, dan kepatuhan di seluruh unit organisasi ()OUs. Pagar pembatas preventif menegakkan kebijakan untuk memastikan keselarasan dengan standar kepatuhan. Mereka diimplementasikan dengan menggunakan kebijakan kontrol layanan dan batas izin IAM. Detective guardrails mendeteksi pelanggaran kebijakan dan masalah kepatuhan, dan menghasilkan peringatan untuk remediasi. Mereka diimplementasikan dengan menggunakan AWS Config, AWS Security Hub CSPM, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector, dan pemeriksaan khusus AWS Lambda .

H

HA

Lihat ketersediaan tinggi.

migrasi database heterogen

Memigrasi database sumber Anda ke database target yang menggunakan mesin database yang berbeda (misalnya, Oracle ke Amazon Aurora). Migrasi heterogen biasanya merupakan bagian dari upaya arsitektur ulang, dan mengubah skema dapat menjadi tugas yang kompleks. AWS menyediakan AWS SCT yang membantu dengan konversi skema.

ketersediaan tinggi (HA)

Kemampuan beban kerja untuk beroperasi terus menerus, tanpa intervensi, jika terjadi tantangan atau bencana. Sistem HA dirancang untuk gagal secara otomatis, secara konsisten memberikan kinerja berkualitas tinggi, dan menangani beban dan kegagalan yang berbeda dengan dampak kinerja minimal.

modernisasi sejarawan

Pendekatan yang digunakan untuk memodernisasi dan meningkatkan sistem teknologi operasional (OT) untuk melayani kebutuhan industri manufaktur dengan lebih baik. Sejarawan adalah jenis database yang digunakan untuk mengumpulkan dan menyimpan data dari berbagai sumber di pabrik.

data penahanan

Sebagian dari data historis berlabel yang ditahan dari kumpulan data yang digunakan untuk melatih model pembelajaran mesin. Anda dapat menggunakan data penahanan untuk mengevaluasi kinerja model dengan membandingkan prediksi model dengan data penahanan.

migrasi database homogen

Memigrasi database sumber Anda ke database target yang berbagi mesin database yang sama (misalnya, Microsoft SQL Server ke Amazon RDS for SQL Server). Migrasi homogen biasanya merupakan bagian dari upaya rehosting atau replatforming. Anda dapat menggunakan utilitas database asli untuk memigrasi skema.

data panas

Data yang sering diakses, seperti data real-time atau data translasi terbaru. Data ini biasanya memerlukan tingkat atau kelas penyimpanan berkinerja tinggi untuk memberikan respons kueri yang cepat.

perbaikan terbaru

Perbaikan mendesak untuk masalah kritis dalam lingkungan produksi. Karena urgensinya, perbaikan terbaru biasanya dibuat di luar alur kerja DevOps rilis biasa.

periode hypercare

Segera setelah cutover, periode waktu ketika tim migrasi mengelola dan memantau aplikasi yang dimigrasi di cloud untuk mengatasi masalah apa pun. Biasanya, periode ini panjangnya 1-4 hari. Pada akhir periode hypercare, tim migrasi biasanya mentransfer tanggung jawab untuk aplikasi ke tim operasi cloud.

I

IAc

Lihat infrastruktur sebagai kode.

kebijakan berbasis identitas

Kebijakan yang dilampirkan pada satu atau beberapa prinsip IAM yang mendefinisikan izin mereka dalam lingkungan. AWS Cloud

aplikasi idle

Aplikasi yang memiliki penggunaan CPU dan memori rata-rata antara 5 dan 20 persen selama periode 90 hari. Dalam proyek migrasi, adalah umum untuk menghentikan aplikasi ini atau mempertahankannya di tempat.

IIoT

Lihat Internet of Things industri.

infrastruktur yang tidak dapat diubah

Model yang menyebarkan infrastruktur baru untuk beban kerja produksi alih-alih memperbarui, menambal, atau memodifikasi infrastruktur yang ada. Infrastruktur yang tidak dapat diubah secara inheren lebih konsisten, andal, dan dapat diprediksi daripada infrastruktur yang dapat berubah. Untuk informasi selengkapnya, lihat praktik terbaik Deploy using immutable infrastructure di AWS Well-Architected Framework.

masuk (masuknya) VPC

Dalam arsitektur AWS multi-akun, VPC yang menerima, memeriksa, dan merutekan koneksi jaringan dari luar aplikasi. Arsitektur Referensi AWS Keamanan merekomendasikan pengaturan akun Jaringan Anda dengan inbound, outbound, dan inspeksi VPCs untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

migrasi inkremental

Strategi cutover di mana Anda memigrasikan aplikasi Anda dalam bagian-bagian kecil alih-alih melakukan satu cutover penuh. Misalnya, Anda mungkin hanya memindahkan beberapa layanan mikro atau pengguna ke sistem baru pada awalnya. Setelah Anda memverifikasi bahwa semuanya berfungsi dengan baik, Anda dapat secara bertahap memindahkan layanan mikro atau pengguna tambahan hingga Anda dapat menonaktifkan sistem lama Anda. Strategi ini mengurangi risiko yang terkait dengan migrasi besar.

Industri 4.0

Sebuah istilah yang diperkenalkan oleh Klaus Schwab pada tahun 2016 untuk merujuk pada modernisasi proses manufaktur melalui kemajuan dalam konektivitas, data real-time, otomatisasi, analitik, dan AI/ML.

infrastruktur

Semua sumber daya dan aset yang terkandung dalam lingkungan aplikasi.

infrastruktur sebagai kode (IAc)

Proses penyediaan dan pengelolaan infrastruktur aplikasi melalui satu set file konfigurasi. IAc dirancang untuk membantu Anda memusatkan manajemen infrastruktur, menstandarisasi sumber daya, dan menskalakan dengan cepat sehingga lingkungan baru dapat diulang, andal, dan konsisten.

Internet of Things industri (IIoT)

Penggunaan sensor dan perangkat yang terhubung ke internet di sektor industri, seperti manufaktur, energi, otomotif, perawatan kesehatan, ilmu kehidupan, dan pertanian. Untuk informasi lebih lanjut, lihat Membangun strategi transformasi digital Internet of Things (IIoT) industri.

inspeksi VPC

Dalam arsitektur AWS multi-akun, VPC terpusat yang mengelola inspeksi lalu lintas jaringan antara VPCs (dalam yang sama atau berbeda Wilayah AWS), internet, dan jaringan lokal. Arsitektur Referensi AWS Keamanan merekomendasikan pengaturan akun Jaringan Anda dengan inbound, outbound, dan inspeksi VPCs untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

Internet of Things (IoT)

Jaringan objek fisik yang terhubung dengan sensor atau prosesor tertanam yang berkomunikasi dengan perangkat dan sistem lain melalui internet atau melalui jaringan komunikasi lokal. Untuk informasi selengkapnya, lihat Apa itu IoT?

interpretasi

Karakteristik model pembelajaran mesin yang menggambarkan sejauh mana manusia dapat memahami bagaimana prediksi model bergantung pada inputnya. Untuk informasi lebih lanjut, lihat Interpretabilitas model pembelajaran mesin dengan. AWS

IoT

Lihat Internet of Things.

Perpustakaan informasi TI (ITIL)

Serangkaian praktik terbaik untuk memberikan layanan TI dan menyelaraskan layanan ini dengan persyaratan bisnis. ITIL menyediakan dasar untuk ITSM.

Manajemen layanan TI (ITSM)

Kegiatan yang terkait dengan merancang, menerapkan, mengelola, dan mendukung layanan TI untuk suatu organisasi. Untuk informasi tentang mengintegrasikan operasi cloud dengan alat ITSM, lihat panduan integrasi operasi.

ITIL

Lihat perpustakaan informasi TI.

ITSM

Lihat manajemen layanan TI.

L

kontrol akses berbasis label (LBAC)

Implementasi kontrol akses wajib (MAC) di mana pengguna dan data itu sendiri masing-masing secara eksplisit diberi nilai label keamanan. Persimpangan antara label keamanan pengguna dan label keamanan data menentukan baris dan kolom mana yang dapat dilihat oleh pengguna.

landing zone

Landing zone adalah AWS lingkungan multi-akun yang dirancang dengan baik yang dapat diskalakan dan aman. Ini adalah titik awal dari mana organisasi Anda dapat dengan cepat meluncurkan dan menyebarkan beban kerja dan aplikasi dengan percaya diri dalam lingkungan keamanan dan infrastruktur mereka. Untuk informasi selengkapnya tentang zona pendaratan, lihat Menyiapkan lingkungan multi-akun AWS yang aman dan dapat diskalakan.

model bahasa besar (LLM)

Model AI pembelajaran mendalam yang dilatih sebelumnya pada sejumlah besar data. LLM dapat melakukan beberapa tugas, seperti menjawab pertanyaan, meringkas dokumen, menerjemahkan teks ke dalam bahasa lain, dan menyelesaikan kalimat. Untuk informasi lebih lanjut, lihat Apa itu LLMs.

migrasi besar

Migrasi 300 atau lebih server.

LBAC

Lihat kontrol akses berbasis label.

hak istimewa paling sedikit

Praktik keamanan terbaik untuk memberikan izin minimum yang diperlukan untuk melakukan tugas. Untuk informasi selengkapnya, lihat Menerapkan izin hak istimewa terkecil dalam dokumentasi IAM.

angkat dan geser

Lihat 7 Rs.

sistem endian kecil

Sebuah sistem yang menyimpan byte paling tidak signifikan terlebih dahulu. Lihat juga endianness.

LLM

Lihat model bahasa besar.

lingkungan yang lebih rendah

Lihat lingkungan.

M

pembelajaran mesin (ML)

Jenis kecerdasan buatan yang menggunakan algoritma dan teknik untuk pengenalan pola dan pembelajaran. ML menganalisis dan belajar dari data yang direkam, seperti data Internet of Things (IoT), untuk menghasilkan model statistik berdasarkan pola. Untuk informasi selengkapnya, lihat Machine Learning.

cabang utama

Lihat cabang.

malware

Perangkat lunak yang dirancang untuk membahayakan keamanan atau privasi komputer. Malware dapat mengganggu sistem komputer, membocorkan informasi sensitif, atau mendapatkan akses yang tidak sah. Contoh malware termasuk virus, worm, ransomware, Trojan horse, spyware, dan keyloggers.

layanan terkelola

Layanan AWS yang AWS mengoperasikan lapisan infrastruktur, sistem operasi, dan platform, dan Anda mengakses titik akhir untuk menyimpan dan mengambil data. Amazon Simple Storage Service (Amazon S3) dan Amazon DynamoDB adalah contoh layanan terkelola. Ini juga dikenal sebagai layanan abstrak.

sistem eksekusi manufaktur (MES)

Sistem perangkat lunak untuk melacak, memantau, mendokumentasikan, dan mengendalikan proses produksi yang mengubah bahan baku menjadi produk jadi di lantai toko.

PETA

Lihat Program Percepatan Migrasi.

mekanisme

Proses lengkap di mana Anda membuat alat, mendorong adopsi alat, dan kemudian memeriksa hasilnya untuk melakukan penyesuaian. Mekanisme adalah siklus yang memperkuat dan meningkatkan dirinya sendiri saat beroperasi. Untuk informasi lebih lanjut, lihat Membangun mekanisme di AWS Well-Architected Framework.

akun anggota

Semua Akun AWS selain akun manajemen yang merupakan bagian dari organisasi di AWS Organizations. Akun dapat menjadi anggota dari hanya satu organisasi pada suatu waktu.

MES

Lihat sistem eksekusi manufaktur.

Transportasi Telemetri Antrian Pesan (MQTT)

Protokol komunikasi ringan machine-to-machine (M2M), berdasarkan pola terbitkan/berlangganan, untuk perangkat IoT yang dibatasi sumber daya.

layanan mikro

Layanan kecil dan independen yang berkomunikasi dengan jelas APIs dan biasanya dimiliki oleh tim kecil yang mandiri. Misalnya, sistem asuransi mungkin mencakup layanan mikro yang memetakan kemampuan bisnis, seperti penjualan atau pemasaran, atau subdomain, seperti pembelian, klaim, atau analitik. Manfaat layanan mikro termasuk kelincahan, penskalaan yang fleksibel, penyebaran yang mudah, kode yang dapat digunakan kembali, dan ketahanan. Untuk informasi selengkapnya, lihat Mengintegrasikan layanan mikro dengan menggunakan layanan tanpa AWS server.

arsitektur microservices

Pendekatan untuk membangun aplikasi dengan komponen independen yang menjalankan setiap proses aplikasi sebagai layanan mikro. Layanan mikro ini berkomunikasi melalui antarmuka yang terdefinisi dengan baik dengan menggunakan ringan. APIs Setiap layanan mikro dalam arsitektur ini dapat diperbarui, digunakan, dan diskalakan untuk memenuhi permintaan fungsi tertentu dari suatu aplikasi. Untuk informasi selengkapnya, lihat Menerapkan layanan mikro di AWS.

Program Percepatan Migrasi (MAP)

AWS Program yang menyediakan dukungan konsultasi, pelatihan, dan layanan untuk membantu organisasi membangun fondasi operasional yang kuat untuk pindah ke cloud, dan untuk membantu mengimbangi biaya awal migrasi. MAP mencakup metodologi migrasi untuk mengeksekusi migrasi lama dengan cara metodis dan seperangkat alat untuk mengotomatisasi dan mempercepat skenario migrasi umum.

migrasi dalam skala

Proses memindahkan sebagian besar portofolio aplikasi ke cloud dalam gelombang, dengan lebih banyak aplikasi bergerak pada tingkat yang lebih cepat di setiap gelombang. Fase ini menggunakan praktik dan pelajaran terbaik dari fase sebelumnya untuk mengimplementasikan pabrik migrasi tim, alat, dan proses untuk merampingkan migrasi beban kerja melalui otomatisasi dan pengiriman tangkas. Ini adalah fase ketiga dari strategi AWS migrasi.

pabrik migrasi

Tim lintas fungsi yang merampingkan migrasi beban kerja melalui pendekatan otomatis dan gesit. Tim pabrik migrasi biasanya mencakup operasi, analis dan pemilik bisnis, insinyur migrasi, pengembang, dan DevOps profesional yang bekerja di sprint. Antara 20 dan 50 persen portofolio aplikasi perusahaan terdiri dari pola berulang yang dapat dioptimalkan dengan pendekatan pabrik. Untuk informasi selengkapnya, lihat diskusi tentang pabrik migrasi dan panduan Pabrik Migrasi Cloud di kumpulan konten ini.

metadata migrasi

Informasi tentang aplikasi dan server yang diperlukan untuk menyelesaikan migrasi. Setiap pola migrasi memerlukan satu set metadata migrasi yang berbeda. Contoh metadata migrasi termasuk subnet target, grup keamanan, dan akun. AWS

pola migrasi

Tugas migrasi berulang yang merinci strategi migrasi, tujuan migrasi, dan aplikasi atau layanan migrasi yang digunakan. Contoh: Rehost migrasi ke Amazon EC2 AWS dengan Layanan Migrasi Aplikasi.

Penilaian Portofolio Migrasi (MPA)

Alat online yang menyediakan informasi untuk memvalidasi kasus bisnis untuk bermigrasi ke. AWS Cloud MPA menyediakan penilaian portofolio terperinci (ukuran kanan server, harga, perbandingan TCO, analisis biaya migrasi) serta perencanaan migrasi (analisis data aplikasi dan pengumpulan data, pengelompokan aplikasi, prioritas migrasi, dan perencanaan gelombang). Alat MPA (memerlukan login) tersedia gratis untuk semua AWS konsultan dan konsultan APN Partner.

Penilaian Kesiapan Migrasi (MRA)

Proses mendapatkan wawasan tentang status kesiapan cloud organisasi, mengidentifikasi kekuatan dan kelemahan, dan membangun rencana aksi untuk menutup kesenjangan yang diidentifikasi, menggunakan CAF. AWS Untuk informasi selengkapnya, lihat panduan kesiapan migrasi. MRA adalah tahap pertama dari strategi AWS migrasi.

strategi migrasi

Pendekatan yang digunakan untuk memigrasikan beban kerja ke file. AWS Cloud Untuk informasi lebih lanjut, lihat entri 7 Rs di glosarium ini dan lihat Memobilisasi organisasi Anda untuk mempercepat migrasi skala besar.

ML

Lihat pembelajaran mesin.

modernisasi

Mengubah aplikasi usang (warisan atau monolitik) dan infrastrukturnya menjadi sistem yang gesit, elastis, dan sangat tersedia di cloud untuk mengurangi biaya, mendapatkan efisiensi, dan memanfaatkan inovasi. Untuk informasi selengkapnya, lihat Strategi untuk memodernisasi aplikasi di. AWS Cloud

penilaian kesiapan modernisasi

Evaluasi yang membantu menentukan kesiapan modernisasi aplikasi organisasi; mengidentifikasi manfaat, risiko, dan dependensi; dan menentukan seberapa baik organisasi dapat mendukung keadaan masa depan aplikasi tersebut. Hasil penilaian adalah cetak biru arsitektur target, peta jalan yang merinci fase pengembangan dan tonggak untuk proses modernisasi, dan rencana aksi untuk mengatasi kesenjangan yang diidentifikasi. Untuk informasi lebih lanjut, lihat Mengevaluasi kesiapan modernisasi untuk aplikasi di. AWS Cloud

aplikasi monolitik (monolit)

Aplikasi yang berjalan sebagai layanan tunggal dengan proses yang digabungkan secara ketat. Aplikasi monolitik memiliki beberapa kelemahan. Jika satu fitur aplikasi mengalami lonjakan permintaan, seluruh arsitektur harus diskalakan. Menambahkan atau meningkatkan fitur aplikasi monolitik juga menjadi lebih kompleks ketika basis kode tumbuh. Untuk mengatasi masalah ini, Anda dapat menggunakan arsitektur microservices. Untuk informasi lebih lanjut, lihat Mengurai monolit menjadi layanan mikro.

MPA

Lihat Penilaian Portofolio Migrasi.

MQTT

Lihat Transportasi Telemetri Antrian Pesan.

klasifikasi multiclass

Sebuah proses yang membantu menghasilkan prediksi untuk beberapa kelas (memprediksi satu dari lebih dari dua hasil). Misalnya, model ML mungkin bertanya “Apakah produk ini buku, mobil, atau telepon?” atau “Kategori produk mana yang paling menarik bagi pelanggan ini?”

infrastruktur yang bisa berubah

Model yang memperbarui dan memodifikasi infrastruktur yang ada untuk beban kerja produksi. Untuk meningkatkan konsistensi, keandalan, dan prediktabilitas, AWS Well-Architected Framework merekomendasikan penggunaan infrastruktur yang tidak dapat diubah sebagai praktik terbaik.

O

OAC

Lihat kontrol akses asal.

OAI

Lihat identitas akses asal.

OCM

Lihat manajemen perubahan organisasi.

migrasi offline

Metode migrasi di mana beban kerja sumber diturunkan selama proses migrasi. Metode ini melibatkan waktu henti yang diperpanjang dan biasanya digunakan untuk beban kerja kecil dan tidak kritis.

OI

Lihat integrasi operasi.

OLA

Lihat perjanjian tingkat operasional.

migrasi online

Metode migrasi di mana beban kerja sumber disalin ke sistem target tanpa diambil offline. Aplikasi yang terhubung ke beban kerja dapat terus berfungsi selama migrasi. Metode ini melibatkan waktu henti nol hingga minimal dan biasanya digunakan untuk beban kerja produksi yang kritis.

OPC-UA

Lihat Komunikasi Proses Terbuka - Arsitektur Terpadu.

Komunikasi Proses Terbuka - Arsitektur Terpadu (OPC-UA)

Protokol komunikasi machine-to-machine (M2M) untuk otomasi industri. OPC-UA menyediakan standar interoperabilitas dengan enkripsi data, otentikasi, dan skema otorisasi.

perjanjian tingkat operasional (OLA)

Perjanjian yang menjelaskan apa yang dijanjikan kelompok TI fungsional untuk diberikan satu sama lain, untuk mendukung perjanjian tingkat layanan (SLA).

Tinjauan Kesiapan Operasional (ORR)

Daftar pertanyaan dan praktik terbaik terkait yang membantu Anda memahami, mengevaluasi, mencegah, atau mengurangi ruang lingkup insiden dan kemungkinan kegagalan. Untuk informasi lebih lanjut, lihat Ulasan Kesiapan Operasional (ORR) dalam Kerangka Kerja Well-Architected AWS .

teknologi operasional (OT)

Sistem perangkat keras dan perangkat lunak yang bekerja dengan lingkungan fisik untuk mengendalikan operasi industri, peralatan, dan infrastruktur. Di bidang manufaktur, integrasi sistem OT dan teknologi informasi (TI) adalah fokus utama untuk transformasi Industri 4.0.

integrasi operasi (OI)

Proses modernisasi operasi di cloud, yang melibatkan perencanaan kesiapan, otomatisasi, dan integrasi. Untuk informasi selengkapnya, lihat panduan integrasi operasi.

jejak organisasi

Jejak yang dibuat oleh AWS CloudTrail itu mencatat semua peristiwa untuk semua Akun AWS dalam organisasi di AWS Organizations. Jejak ini dibuat di setiap Akun AWS bagian organisasi dan melacak aktivitas di setiap akun. Untuk informasi selengkapnya, lihat Membuat jejak untuk organisasi dalam CloudTrail dokumentasi.

manajemen perubahan organisasi (OCM)

Kerangka kerja untuk mengelola transformasi bisnis utama yang mengganggu dari perspektif orang, budaya, dan kepemimpinan. OCM membantu organisasi mempersiapkan, dan transisi ke, sistem dan strategi baru dengan mempercepat adopsi perubahan, mengatasi masalah transisi, dan mendorong perubahan budaya dan organisasi. Dalam strategi AWS migrasi, kerangka kerja ini disebut percepatan orang, karena kecepatan perubahan yang diperlukan dalam proyek adopsi cloud. Untuk informasi lebih lanjut, lihat panduan OCM.

kontrol akses asal (OAC)

Di CloudFront, opsi yang disempurnakan untuk membatasi akses untuk mengamankan konten Amazon Simple Storage Service (Amazon S3) Anda. OAC mendukung semua bucket S3 di semua Wilayah AWS, enkripsi sisi server dengan AWS KMS (SSE-KMS), dan dinamis dan permintaan ke bucket S3. `PUT` `DELETE`

identitas akses asal (OAI)

Di CloudFront, opsi untuk membatasi akses untuk mengamankan konten Amazon S3 Anda. Saat Anda menggunakan OAI, CloudFront buat prinsipal yang dapat diautentikasi oleh Amazon S3. Prinsipal yang diautentikasi dapat mengakses konten dalam bucket S3 hanya melalui distribusi tertentu. CloudFront Lihat juga OAC, yang menyediakan kontrol akses yang lebih terperinci dan ditingkatkan.

ORR

Lihat tinjauan kesiapan operasional.

OT

Lihat teknologi operasional.

keluar (jalan keluar) VPC

Dalam arsitektur AWS multi-akun, VPC yang menangani koneksi jaringan yang dimulai dari dalam aplikasi. Arsitektur Referensi AWS Keamanan merekomendasikan pengaturan akun Jaringan Anda dengan inbound, outbound, dan inspeksi VPCs untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

P

batas izin

Kebijakan manajemen IAM yang dilampirkan pada prinsipal IAM untuk menetapkan izin maksimum yang dapat dimiliki pengguna atau peran. Untuk informasi selengkapnya, lihat Batas izin dalam dokumentasi IAM.

Informasi Identifikasi Pribadi (PII)

Informasi yang, jika dilihat secara langsung atau dipasangkan dengan data terkait lainnya, dapat digunakan untuk menyimpulkan identitas individu secara wajar. Contoh PII termasuk nama, alamat, dan informasi kontak.

PII

Lihat informasi yang dapat diidentifikasi secara pribadi.

buku pedoman

Serangkaian langkah yang telah ditentukan sebelumnya yang menangkap pekerjaan yang terkait dengan migrasi, seperti mengirimkan fungsi operasi inti di cloud. Buku pedoman dapat berupa skrip, runbook otomatis, atau ringkasan proses atau langkah-langkah yang diperlukan untuk mengoperasikan lingkungan modern Anda.

PLC

Lihat pengontrol logika yang dapat diprogram.

PLM

Lihat manajemen siklus hidup produk.

kebijakan

Objek yang dapat menentukan izin (lihat kebijakan berbasis identitas), menentukan kondisi akses (lihat kebijakan berbasis sumber daya), atau menentukan izin maksimum untuk semua akun di organisasi (lihat kebijakan kontrol layanan). AWS Organizations

persistensi poliglot

Secara independen memilih teknologi penyimpanan data microservice berdasarkan pola akses data dan persyaratan lainnya. Jika layanan mikro Anda memiliki teknologi penyimpanan data yang sama, mereka dapat menghadapi tantangan implementasi atau mengalami kinerja yang buruk. Layanan mikro lebih mudah diimplementasikan dan mencapai kinerja dan skalabilitas yang lebih baik jika mereka menggunakan penyimpanan data yang paling sesuai dengan kebutuhan mereka.

penilaian portofolio

Proses menemukan, menganalisis, dan memprioritaskan portofolio aplikasi untuk merencanakan migrasi. Untuk informasi selengkapnya, lihat Mengevaluasi kesiapan migrasi.

predikat

Kondisi kueri yang mengembalikan `true` atau`false`, biasanya terletak di `WHERE` klausa.

predikat pushdown

Teknik optimasi kueri database yang menyaring data dalam kueri sebelum transfer. Ini mengurangi jumlah data yang harus diambil dan diproses dari database relasional, dan meningkatkan kinerja kueri.

kontrol preventif

Kontrol keamanan yang dirancang untuk mencegah suatu peristiwa terjadi. Kontrol ini adalah garis pertahanan pertama untuk membantu mencegah akses tidak sah atau perubahan yang tidak diinginkan ke jaringan Anda. Untuk informasi selengkapnya, lihat Kontrol pencegahan dalam Menerapkan kontrol keamanan pada. AWS

principal

Entitas AWS yang dapat melakukan tindakan dan mengakses sumber daya. Entitas ini biasanya merupakan pengguna root untuk Akun AWS, peran IAM, atau pengguna. Untuk informasi selengkapnya, lihat Prinsip dalam istilah dan konsep Peran dalam dokumentasi IAM.

privasi berdasarkan desain

Pendekatan rekayasa sistem yang memperhitungkan privasi melalui seluruh proses pengembangan.

zona host pribadi

Container yang menyimpan informasi tentang bagaimana Anda ingin Amazon Route 53 merespons kueri DNS untuk domain dan subdomainnya dalam satu atau lebih. VPCs Untuk informasi selengkapnya, lihat Bekerja dengan zona yang dihosting pribadi di dokumentasi Route 53.

kontrol proaktif

Kontrol keamanan yang dirancang untuk mencegah penyebaran sumber daya yang tidak sesuai. Kontrol ini memindai sumber daya sebelum disediakan. Jika sumber daya tidak sesuai dengan kontrol, maka itu tidak disediakan. Untuk informasi selengkapnya, lihat panduan referensi Kontrol dalam AWS Control Tower dokumentasi dan lihat Kontrol proaktif dalam Menerapkan kontrol keamanan pada AWS.

manajemen siklus hidup produk (PLM)

Manajemen data dan proses untuk suatu produk di seluruh siklus hidupnya, mulai dari desain, pengembangan, dan peluncuran, melalui pertumbuhan dan kematangan, hingga penurunan dan penghapusan.

lingkungan produksi

Lihat lingkungan.

pengontrol logika yang dapat diprogram (PLC)

Di bidang manufaktur, komputer yang sangat andal dan mudah beradaptasi yang memantau mesin dan mengotomatiskan proses manufaktur.

rantai cepat

Menggunakan output dari satu prompt LLM sebagai input untuk prompt berikutnya untuk menghasilkan respons yang lebih baik. Teknik ini digunakan untuk memecah tugas yang kompleks menjadi subtugas, atau untuk secara iteratif memperbaiki atau memperluas respons awal. Ini membantu meningkatkan akurasi dan relevansi respons model dan memungkinkan hasil yang lebih terperinci dan dipersonalisasi.

pseudonimisasi

Proses penggantian pengidentifikasi pribadi dalam kumpulan data dengan nilai placeholder. Pseudonimisasi dapat membantu melindungi privasi pribadi. Data pseudonim masih dianggap sebagai data pribadi.

publish/subscribe (pub/sub)

Pola yang memungkinkan komunikasi asinkron antara layanan mikro untuk meningkatkan skalabilitas dan daya tanggap. Misalnya, dalam MES berbasis layanan mikro, layanan mikro dapat mempublikasikan pesan peristiwa ke saluran yang dapat berlangganan layanan mikro lainnya. Sistem dapat menambahkan layanan mikro baru tanpa mengubah layanan penerbitan.

Q

rencana kueri

Serangkaian langkah, seperti instruksi, yang digunakan untuk mengakses data dalam sistem database relasional SQL.

regresi rencana kueri

Ketika pengoptimal layanan database memilih rencana yang kurang optimal daripada sebelum perubahan yang diberikan ke lingkungan database. Hal ini dapat disebabkan oleh perubahan statistik, kendala, pengaturan lingkungan, pengikatan parameter kueri, dan pembaruan ke mesin database.

R

Matriks RACI

Lihat bertanggung jawab, akuntabel, dikonsultasikan, diinformasikan (RACI).

LAP

Lihat Retrieval Augmented Generation.

ransomware

Perangkat lunak berbahaya yang dirancang untuk memblokir akses ke sistem komputer atau data sampai pembayaran dilakukan.

Matriks RASCI

Lihat bertanggung jawab, akuntabel, dikonsultasikan, diinformasikan (RACI).

RCAC

Lihat kontrol akses baris dan kolom.

replika baca

Salinan database yang digunakan untuk tujuan read-only. Anda dapat merutekan kueri ke replika baca untuk mengurangi beban pada database utama Anda.

arsitek ulang

Lihat 7 Rs.

tujuan titik pemulihan (RPO)

Jumlah waktu maksimum yang dapat diterima sejak titik pemulihan data terakhir. Ini menentukan apa yang dianggap sebagai kehilangan data yang dapat diterima antara titik pemulihan terakhir dan gangguan layanan.

tujuan waktu pemulihan (RTO)

Penundaan maksimum yang dapat diterima antara gangguan layanan dan pemulihan layanan.

refactor

Lihat 7 Rs.

Region

Kumpulan AWS sumber daya di wilayah geografis. Masing-masing Wilayah AWS terisolasi dan independen dari yang lain untuk memberikan toleransi kesalahan, stabilitas, dan ketahanan. Untuk informasi selengkapnya, lihat Menentukan Wilayah AWS akun yang dapat digunakan.

regresi

Teknik ML yang memprediksi nilai numerik. Misalnya, untuk memecahkan masalah "Berapa harga rumah ini akan dijual?" Model ML dapat menggunakan model regresi linier untuk memprediksi harga jual rumah berdasarkan fakta yang diketahui tentang rumah (misalnya, luas persegi).

rehost

Lihat 7 Rs.

melepaskan

Dalam proses penyebaran, tindakan mempromosikan perubahan pada lingkungan produksi.

memindahkan

Lihat 7 Rs.

memplatform ulang

Lihat 7 Rs.

pembelian kembali

Lihat 7 Rs.

ketahanan

Kemampuan aplikasi untuk melawan atau pulih dari gangguan. Ketersediaan tinggi dan pemulihan bencana adalah pertimbangan umum ketika merencanakan ketahanan di. AWS Cloud Untuk informasi lebih lanjut, lihat AWS Cloud Ketahanan.

kebijakan berbasis sumber daya

Kebijakan yang dilampirkan ke sumber daya, seperti bucket Amazon S3, titik akhir, atau kunci enkripsi. Jenis kebijakan ini menentukan prinsipal mana yang diizinkan mengakses, tindakan yang didukung, dan kondisi lain yang harus dipenuhi.

matriks yang bertanggung jawab, akuntabel, dikonsultasikan, diinformasikan (RACI)

Matriks yang mendefinisikan peran dan tanggung jawab untuk semua pihak yang terlibat dalam kegiatan migrasi dan operasi cloud. Nama matriks berasal dari jenis tanggung jawab yang

didefinisikan dalam matriks: bertanggung jawab (R), akuntabel (A), dikonsultasikan (C), dan diinformasikan (I). Jenis dukungan (S) adalah opsional. Jika Anda menyertakan dukungan, matriks disebut matriks RASCI, dan jika Anda mengecualikannya, itu disebut matriks RACI.

kontrol responsif

Kontrol keamanan yang dirancang untuk mendorong remediasi efek samping atau penyimpangan dari garis dasar keamanan Anda. Untuk informasi selengkapnya, lihat Kontrol responsif dalam Menerapkan kontrol keamanan pada AWS.

melestarikan

Lihat 7 Rs.

pensiun

Lihat 7 Rs.

Retrieval Augmented Generation (RAG)

Teknologi AI generatif di mana LLM mereferensikan sumber data otoritatif yang berada di luar sumber data pelatihannya sebelum menghasilkan respons. Misalnya, model RAG mungkin melakukan pencarian semantik dari basis pengetahuan organisasi atau data kustom. Untuk informasi lebih lanjut, lihat Apa itu RAG.

rotasi

Proses memperbarui rahasia secara berkala untuk membuatnya lebih sulit bagi penyerang untuk mengakses kredensil.

kontrol akses baris dan kolom (RCAC)

Penggunaan ekspresi SQL dasar dan fleksibel yang telah menetapkan aturan akses. RCAC terdiri dari izin baris dan topeng kolom.

RPO

Lihat tujuan titik pemulihan.

RTO

Lihat tujuan waktu pemulihan.

buku runbook

Satu set prosedur manual atau otomatis yang diperlukan untuk melakukan tugas tertentu. Ini biasanya dibangun untuk merampingkan operasi berulang atau prosedur dengan tingkat kesalahan yang tinggi.

D

SAML 2.0

Standar terbuka yang digunakan oleh banyak penyedia identitas (IdPs). Fitur ini memungkinkan sistem masuk tunggal gabungan (SSO), sehingga pengguna dapat masuk ke Konsol Manajemen AWS atau memanggil operasi AWS API tanpa Anda harus membuat pengguna di IAM untuk semua orang di organisasi Anda. Untuk informasi lebih lanjut tentang federasi berbasis SAMP 2.0, lihat Tentang federasi berbasis SAMP 2.0 dalam dokumentasi IAM.

PENIPUAN

Lihat kontrol pengawasan dan akuisisi data.

SCP

Lihat kebijakan kontrol layanan.

Rahasia

Dalam AWS Secrets Manager, informasi rahasia atau terbatas, seperti kata sandi atau kredensi pengguna, yang Anda simpan dalam bentuk terenkripsi. Ini terdiri dari nilai rahasia dan metadata-nya. Nilai rahasia dapat berupa biner, string tunggal, atau beberapa string. Untuk informasi selengkapnya, lihat Apa yang ada di rahasia Secrets Manager? dalam dokumentasi Secrets Manager.

keamanan dengan desain

Pendekatan rekayasa sistem yang memperhitungkan keamanan melalui seluruh proses pengembangan.

kontrol keamanan

Pagar pembatas teknis atau administratif yang mencegah, mendeteksi, atau mengurangi kemampuan pelaku ancaman untuk mengeksploitasi kerentanan keamanan. Ada empat jenis kontrol keamanan utama: preventif, detektif, responsif, dan proaktif.

pengerasan keamanan

Proses mengurangi permukaan serangan untuk membuatnya lebih tahan terhadap serangan. Ini dapat mencakup tindakan seperti menghapus sumber daya yang tidak lagi diperlukan, menerapkan praktik keamanan terbaik untuk memberikan hak istimewa paling sedikit, atau menonaktifkan fitur yang tidak perlu dalam file konfigurasi.

sistem informasi keamanan dan manajemen acara (SIEM)

Alat dan layanan yang menggabungkan sistem manajemen informasi keamanan (SIM) dan manajemen acara keamanan (SEM). Sistem SIEM mengumpulkan, memantau, dan menganalisis data dari server, jaringan, perangkat, dan sumber lain untuk mendeteksi ancaman dan pelanggaran keamanan, dan untuk menghasilkan peringatan.

otomatisasi respons keamanan

Tindakan yang telah ditentukan dan diprogram yang dirancang untuk secara otomatis merespons atau memulihkan peristiwa keamanan. Otomatisasi ini berfungsi sebagai kontrol keamanan detektif atau responsif yang membantu Anda menerapkan praktik terbaik AWS keamanan. Contoh tindakan respons otomatis termasuk memodifikasi grup keamanan VPC, menambal instans Amazon EC2, atau memutar kredensil.

enkripsi sisi server

Enkripsi data di tujuannya, oleh Layanan AWS yang menerimanya.

kebijakan kontrol layanan (SCP)

Kebijakan yang menyediakan kontrol terpusat atas izin untuk semua akun di organisasi. AWS Organizations SCPs menentukan pagar pembatas atau menetapkan batasan pada tindakan yang dapat didelegasikan oleh administrator kepada pengguna atau peran. Anda dapat menggunakan SCPs daftar izin atau daftar penolakan, untuk menentukan layanan atau tindakan mana yang diizinkan atau dilarang. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam AWS Organizations dokumentasi.

titik akhir layanan

URL titik masuk untuk file Layanan AWS. Anda dapat menggunakan endpoint untuk terhubung secara terprogram ke layanan target. Untuk informasi selengkapnya, lihat Layanan AWS titik akhir di Referensi Umum AWS.

perjanjian tingkat layanan (SLA)

Perjanjian yang menjelaskan apa yang dijanjikan tim TI untuk diberikan kepada pelanggan mereka, seperti waktu kerja dan kinerja layanan.

indikator tingkat layanan (SLI)

Pengukuran aspek kinerja layanan, seperti tingkat kesalahan, ketersediaan, atau throughputnya.

tujuan tingkat layanan (SLO)

Metrik target yang mewakili kesehatan layanan, yang diukur dengan indikator tingkat layanan.

model tanggung jawab bersama

Model yang menjelaskan tanggung jawab yang Anda bagikan AWS untuk keamanan dan kepatuhan cloud. AWS bertanggung jawab atas keamanan cloud, sedangkan Anda bertanggung jawab atas keamanan di cloud. Untuk informasi selengkapnya, lihat Model tanggung jawab bersama.

SIEM

Lihat informasi keamanan dan sistem manajemen acara.

titik kegagalan tunggal (SPOF)

Kegagalan dalam satu komponen penting dari aplikasi yang dapat mengganggu sistem.

SLA

Lihat perjanjian tingkat layanan.

SLI

Lihat indikator tingkat layanan.

SLO

Lihat tujuan tingkat layanan.

split-and-seed model

Pola untuk menskalakan dan mempercepat proyek modernisasi. Ketika fitur baru dan rilis produk didefinisikan, tim inti berpisah untuk membuat tim produk baru. Ini membantu meningkatkan kemampuan dan layanan organisasi Anda, meningkatkan produktivitas pengembang, dan

mendukung inovasi yang cepat. Untuk informasi lebih lanjut, lihat Pendekatan bertahap untuk memodernisasi aplikasi di. AWS Cloud

SPOF

Lihat satu titik kegagalan.

skema bintang

Struktur organisasi database yang menggunakan satu tabel fakta besar untuk menyimpan data transaksional atau terukur dan menggunakan satu atau lebih tabel dimensi yang lebih kecil untuk menyimpan atribut data. Struktur ini dirancang untuk digunakan dalam gudang data atau untuk tujuan intelijen bisnis.

pola ara pencekik

Pendekatan untuk memodernisasi sistem monolitik dengan menulis ulang secara bertahap dan mengganti fungsionalitas sistem sampai sistem warisan dapat dinonaktifkan. Pola ini menggunakan analogi pohon ara yang tumbuh menjadi pohon yang sudah mapan dan akhirnya mengatasi dan menggantikan inangnya. Pola ini diperkenalkan oleh Martin Fowler sebagai cara untuk mengelola risiko saat menulis ulang sistem monolitik. Untuk contoh cara menerapkan pola ini, lihat Memodernisasi layanan web Microsoft ASP.NET (ASMX) lama secara bertahap menggunakan container dan Amazon API Gateway.

subnet

Rentang alamat IP dalam VPC Anda. Subnet harus berada di Availability Zone tunggal.

kontrol pengawasan dan akuisisi data (SCADA)

Di bidang manufaktur, sistem yang menggunakan perangkat keras dan perangkat lunak untuk memantau aset fisik dan operasi produksi.

enkripsi simetris

Algoritma enkripsi yang menggunakan kunci yang sama untuk mengenkripsi dan mendekripsi data.

pengujian sintetis

Menguji sistem dengan cara yang mensimulasikan interaksi pengguna untuk mendeteksi potensi masalah atau untuk memantau kinerja. Anda dapat menggunakan Amazon CloudWatch Synthetics untuk membuat tes ini.

sistem prompt

Teknik untuk memberikan konteks, instruksi, atau pedoman ke LLM untuk mengarahkan perilakunya. Permintaan sistem membantu mengatur konteks dan menetapkan aturan untuk interaksi dengan pengguna.

T

tag

Pasangan nilai kunci yang bertindak sebagai metadata untuk mengatur sumber daya Anda. AWS Tanda membantu Anda mengelola, mengidentifikasi, mengatur, dan memfilter sumber daya. Untuk informasi selengkapnya, lihat Menandai sumber daya AWS.

variabel target

Nilai yang Anda coba prediksi dalam ML yang diawasi. Ini juga disebut sebagai variabel hasil. Misalnya, dalam pengaturan manufaktur, variabel target bisa menjadi cacat produk.

daftar tugas

Alat yang digunakan untuk melacak kemajuan melalui runbook. Daftar tugas berisi ikhtisar runbook dan daftar tugas umum yang harus diselesaikan. Untuk setiap tugas umum, itu termasuk perkiraan jumlah waktu yang dibutuhkan, pemilik, dan kemajuan.

lingkungan uji

Lihat lingkungan.

pelatihan

Untuk menyediakan data bagi model ML Anda untuk dipelajari. Data pelatihan harus berisi jawaban yang benar. Algoritma pembelajaran menemukan pola dalam data pelatihan yang memetakan atribut data input ke target (jawaban yang ingin Anda prediksi). Ini menghasilkan model ML yang menangkap pola-pola ini. Anda kemudian dapat menggunakan model ML untuk membuat prediksi pada data baru yang Anda tidak tahu targetnya.

gerbang transit

Hub transit jaringan yang dapat Anda gunakan untuk menghubungkan jaringan Anda VPCs dan lokal. Untuk informasi selengkapnya, lihat Apa itu gateway transit dalam AWS Transit Gateway dokumentasi.

alur kerja berbasis batang

Pendekatan di mana pengembang membangun dan menguji fitur secara lokal di cabang fitur dan kemudian menggabungkan perubahan tersebut ke cabang utama. Cabang utama kemudian dibangun untuk pengembangan, praproduksi, dan lingkungan produksi, secara berurutan.

akses tepercaya

Memberikan izin ke layanan yang Anda tentukan untuk melakukan tugas di organisasi Anda di dalam AWS Organizations dan di akunnya atas nama Anda. Layanan tepercaya menciptakan peran terkait layanan di setiap akun, ketika peran itu diperlukan, untuk melakukan tugas manajemen untuk Anda. Untuk informasi selengkapnya, lihat Menggunakan AWS Organizations dengan AWS layanan lain dalam AWS Organizations dokumentasi.

penyetelan

Untuk mengubah aspek proses pelatihan Anda untuk meningkatkan akurasi model ML. Misalnya, Anda dapat melatih model ML dengan membuat set pelabelan, menambahkan label, dan kemudian mengulangi langkah-langkah ini beberapa kali di bawah pengaturan yang berbeda untuk mengoptimalkan model.

tim dua pizza

Sebuah DevOps tim kecil yang bisa Anda beri makan dengan dua pizza. Ukuran tim dua pizza memastikan peluang terbaik untuk berkolaborasi dalam pengembangan perangkat lunak.

U

waswas

Sebuah konsep yang mengacu pada informasi yang tidak tepat, tidak lengkap, atau tidak diketahui yang dapat merusak keandalan model ML prediktif. Ada dua jenis ketidakpastian: ketidakpastian epistemik disebabkan oleh data yang terbatas dan tidak lengkap, sedangkan ketidakpastian aleatorik disebabkan oleh kebisingan dan keacakan yang melekat dalam data.

tugas yang tidak terdiferensiasi

Juga dikenal sebagai angkat berat, pekerjaan yang diperlukan untuk membuat dan mengoperasikan aplikasi tetapi itu tidak memberikan nilai langsung kepada pengguna akhir atau memberikan keunggulan kompetitif. Contoh tugas yang tidak terdiferensiasi termasuk pengadaan, pemeliharaan, dan perencanaan kapasitas.

lingkungan atas

Lihat lingkungan.

V

menyedot debu

Operasi pemeliharaan database yang melibatkan pembersihan setelah pembaruan tambahan untuk merebut kembali penyimpanan dan meningkatkan kinerja.

kendali versi

Proses dan alat yang melacak perubahan, seperti perubahan kode sumber dalam repositori.

Peering VPC

Koneksi antara dua VPCs yang memungkinkan Anda untuk merutekan lalu lintas dengan menggunakan alamat IP pribadi. Untuk informasi selengkapnya, lihat Apa itu peering VPC di dokumentasi VPC Amazon.

kerentanan

Kelemahan perangkat lunak atau perangkat keras yang membahayakan keamanan sistem.

W

cache hangat

Cache buffer yang berisi data terkini dan relevan yang sering diakses. Instance database dapat membaca dari cache buffer, yang lebih cepat daripada membaca dari memori utama atau disk.

data hangat

Data yang jarang diakses. Saat menanyakan jenis data ini, kueri yang cukup lambat biasanya dapat diterima.

fungsi jendela

Fungsi SQL yang melakukan perhitungan pada sekelompok baris yang berhubungan dengan catatan saat ini. Fungsi jendela berguna untuk memproses tugas, seperti menghitung rata-rata bergerak atau mengakses nilai baris berdasarkan posisi relatif dari baris saat ini.

beban kerja

Kumpulan sumber daya dan kode yang memberikan nilai bisnis, seperti aplikasi yang dihadapi pelanggan atau proses backend.

aliran kerja

Grup fungsional dalam proyek migrasi yang bertanggung jawab atas serangkaian tugas tertentu. Setiap alur kerja independen tetapi mendukung alur kerja lain dalam proyek. Misalnya, alur kerja portofolio bertanggung jawab untuk memprioritaskan aplikasi, perencanaan gelombang, dan mengumpulkan metadata migrasi. Alur kerja portofolio mengirimkan aset ini ke alur kerja migrasi, yang kemudian memigrasikan server dan aplikasi.

CACING

Lihat menulis sekali, baca banyak.

WQF

Lihat AWS Kerangka Kualifikasi Beban Kerja.

tulis sekali, baca banyak (WORM)

Model penyimpanan yang menulis data satu kali dan mencegah data dihapus atau dimodifikasi. Pengguna yang berwenang dapat membaca data sebanyak yang diperlukan, tetapi mereka tidak dapat mengubahnya. Infrastruktur penyimpanan data ini dianggap tidak dapat diubah.

Z

eksploitasi zero-day

Serangan, biasanya malware, yang memanfaatkan kerentanan zero-day.

kerentanan zero-day

Cacat atau kerentanan yang tak tanggung-tanggung dalam sistem produksi. Aktor ancaman dapat menggunakan jenis kerentanan ini untuk menyerang sistem. Pengembang sering menyadari kerentanan sebagai akibat dari serangan tersebut.

bisikan zero-shot

Memberikan LLM dengan instruksi untuk melakukan tugas tetapi tidak ada contoh (tembakan) yang dapat membantu membimbingnya. LLM harus menggunakan pengetahuan pra-terlatih untuk

menangani tugas. Efektivitas bidikan nol tergantung pada kompleksitas tugas dan kualitas prompt. Lihat juga beberapa bidikan yang diminta.

aplikasi zombie

Aplikasi yang memiliki CPU rata-rata dan penggunaan memori di bawah 5 persen. Dalam proyek migrasi, adalah umum untuk menghentikan aplikasi ini.

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