



Strategi pengoptimalan biaya untuk SAP beban kerja di Cloud AWS

# AWS Bimbingan Preskriptif



# AWS Bimbingan Preskriptif: Strategi pengoptimalan biaya untuk SAP beban kerja di Cloud AWS

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

---

# Table of Contents

Pengantar .....	1
Audiens yang dituju .....	2
Hasil bisnis yang ditargetkan .....	2
Model harga .....	3
Sesuai Permintaan .....	3
Savings Plans .....	4
Host Khusus .....	4
Instans Cadangan .....	5
AWS layanan .....	6
Proses optimasi biaya .....	6
Layanan pengoptimalan biaya .....	7
AWS Cost Explorer .....	8
AWS Trusted Advisor .....	8
AWS Compute Optimizer .....	8
AWS Config .....	9
Layanan pengoptimalan penyimpanan .....	9
Manajer Volume Logis .....	10
Kebijakan siklus hidup Amazon S3 .....	10
Volume EBS Amazon .....	10
Kebijakan siklus hidup Amazon EFS .....	11
Pertanyaan yang Sering Diajukan .....	12
BagaimanaAWSpekerjaan penetapan harga? .....	12
Di mana saya dapat menemukan informasi lebih lanjut tentangAWSRencana Tabungan? .....	12
Bagaimana cara kerja SAPAWS? .....	12
Bagaimana cara memigrasikan beban kerja SAP keAWS? .....	12
Apakah mengotomatisasi operasi SAP mengurangi biaya? .....	12
Jenis layanan apa yang dilakukanAWSpenawaran untuk SAP? .....	13
Kasus penggunaan SAP mana yang melakukannyaAWSdukungan? .....	13
Langkah selanjutnya .....	14
Sumber daya .....	15
AWSmodel harga .....	15
Dokumentasi dukungan SAP .....	15
Sumber daya lainnya .....	15
Riwayat dokumen .....	16

Glosarium .....	17
# .....	17
A .....	18
B .....	21
C .....	23
D .....	26
E .....	30
F .....	32
G .....	33
H .....	34
I .....	35
L .....	38
M .....	39
O .....	43
P .....	45
Q .....	48
R .....	49
D .....	51
T .....	55
U .....	57
V .....	57
W .....	58
Z .....	59
.....	ix

# Strategi pengoptimalan biaya untuk beban kerja SAP di AWS Cloud

Chris Grudzinski dan Sergej Trisic, Amazon Web Services () AWS

Juli 2022 ([sejarah dokumen](#))

Migrasi beban kerja [SAP](#) ke perusahaan AWS Cloud dapat meningkatkan keandalan, skalabilitas, dan kelincahan serta dapat mengurangi biaya operasional yang terkait dengan menjalankan SAP. Menurut [Business Benefits of Running SAP Workloads on AWS](#) (IDG whitepaper), “Penghematan biaya adalah motivator utama bagi pelanggan untuk bergerak, dengan 96% pelanggan melaporkan pengurangan total biaya kepemilikan (TCO) dan penghematan keseluruhan rata-rata 26%”.

Whitepaper IDG yang sama mengungkapkan bahwa sebagian besar organisasi menyadari ekspektasi penghematan biaya mereka setelah dua tahun. 43% responden menghemat perangkat keras dan perangkat lunak, 37% mengutip penghematan operasional, dan 36% menyadari penghematan pada pemeliharaan berkelanjutan. Dari pengalaman kami, beban kerja SAP mengkonsumsi sebagian besar anggaran TI yang khas. Dengan menerapkan strategi dalam panduan ini, Anda dapat mengarahkan tabungan Anda ke inisiatif lain, seperti meningkatkan operasi TI dan memajukan inovasi.

Panduan ini membantu Anda mengoptimalkan biaya menjalankan SAP AWS dan menyoroti AWS layanan dan alat yang mendukung strategi ini. Umumnya, ada dua tuas yang memungkinkan Anda mengontrol pengeluaran Anda untuk AWS layanan:

- Model harga khusus — Anda biasanya memilih model harga di awal perjalanan migrasi Anda. Panduan ini secara singkat membahas model penetapan harga dan menyertakan referensi ke sumber daya tambahan sehingga Anda dapat menyelam lebih dalam. Karena beberapa model penetapan harga tunduk pada komitmen jangka waktu, bagian ini paling berlaku untuk perusahaan di awal perjalanan migrasi mereka atau perusahaan yang meninjau kembali model mereka saat ini.
- AWS layanan— Asli AWS layanan dan alat dapat memberikan analisis dan visualisasi terperinci ke dalam biaya aktual dan proyeksi Anda. Panduan ini berfokus utama pada tuas ini. Analisis dan visualisasi biaya sangat penting karena Anda atau penyedia layanan terkelola (MSP) dapat mengukur, bertindak, dan belajar secara interaktif dan gesit.

### Important

Yang AWS layanan disebutkan dalam panduan ini mungkin membuat rekomendasi untuk mengoptimalkan biaya Anda. Namun, rekomendasi ini mungkin tidak kompatibel dengan beban kerja SAP yang beroperasi. Sebelum bertindak, konfirmasi bahwa rekomendasi tersebut kompatibel dengan mengoperasikan SAP pada AWS:

- [Aplikasi SAP pada AWS: Produk DB/OS dan Amazon EC2 yang didukung - 1656099](#) (catatan SAP)
- [SAP on AWS: Prasyarat Support - 1656250](#) (catatan SAP)

## Audiens yang dituju

Panduan ini dirancang untuk pemangku kepentingan senior SAP, seperti chief information officer (CIO); chief digital officer (CDO); wakil presiden (VP), petugas operasi keuangan cloud; dan direktur tim aplikasi perusahaan, pusat kompetensi SAP atau Enterprise Resource Planning (ERP), dan tim infrastruktur TI. Tujuan dari panduan ini adalah untuk membantu Anda memilih, menyesuaikan, dan menerapkan AWS layanan yang mengoptimalkan pengeluaran Anda dan menyeimbangkannya dengan kinerja, ketersediaan, fleksibilitas, dan tujuan kecepatan. Akibatnya, dokumen ini tidak fokus pada detail teknis atau implementasi. Namun, mungkin bermanfaat bagi konsultan teknis, arsitek solusi, administrator TI, dan staf TI lainnya yang bertanggung jawab untuk mengelola biaya. Panduan ini mencakup referensi ke konten teknis yang lebih dalam tentang teknologi SAP dan manajemen biaya. AWS

## Hasil bisnis yang ditargetkan

- Mengurangi biaya perangkat keras
- Mengurangi biaya sistem operasi (OS)
- Minimalkan biaya perawatan untuk perangkat keras dan OS
- Mengurangi biaya perangkat lunak
- Meningkatkan efisiensi biaya dengan memusatkan manajemen biaya
- Meningkatkan pembelajaran organisasi

# Memilih model AWS harga untuk beban kerja SAP

Salah satu manfaat utama menjalankan SAP AWS adalah model pay-as-you-go harga, yang berarti Anda hanya membayar untuk apa yang Anda gunakan. Beberapa model memungkinkan Anda mengurangi biaya dengan berkomitmen pada layanan atau sumber daya untuk jangka waktu tertentu. AWS juga menawarkan diskon berbasis volume sehingga Anda dapat mewujudkan penghematan saat penggunaan Anda meningkat. Untuk informasi selengkapnya, lihat [Bagaimana cara kerja AWS penetapan harga?](#) Anda dapat memperkirakan biaya menjalankan beban kerja SAP Anda AWS dengan menggunakan Kalkulator [Harga AWS](#).

Beban kerja SAP memiliki berbagai tingkat ketersediaan yang diperlukan. Oleh karena itu, di awal proses migrasi, Anda bekerja dengan tim akun Anda untuk menentukan model harga yang paling tepat. Seringkali model harga Anda terdiri dari kombinasi beberapa opsi, sebagaimana ditentukan oleh persyaratan ketersediaan Anda. Misalnya, Anda dapat memilih untuk menyertakan Savings Plan untuk sumber daya yang terkait dengan lanskap produksi, kualitas, dan pengembangan. Atau, karena sumber daya kotak pasir hanya dinyalakan saat diperlukan, Anda dapat memilih model sesuai permintaan untuk sumber daya di lingkungan tersebut.

Panduan ini memberikan ringkasan singkat dari model harga berikut, yang berlaku untuk beban kerja SAP pada: AWS

- [Sesuai Permintaan](#)
- [Savings Plans](#)
- [Host Khusus](#)
- [Instans Cadangan](#)

Model harga tambahan tersedia, tetapi tidak dibahas dalam panduan ini karena tidak berlaku untuk beban kerja SAP.

## Sesuai Permintaan

Harga On-Demand memungkinkan Anda membayar kapasitas komputasi yang benar-benar Anda gunakan, per jam, tanpa komitmen jangka panjang. Anda dapat menggunakan model penetapan harga ini untuk beban kerja dengan berbagai tingkat ketersediaan, seperti lingkungan kotak pasir atau lingkungan proyek (N+1) yang memiliki masa pakai terbatas. Jika persyaratan ketersediaan rata-

rata instans EC2 Anda lebih dari sekitar 70 jam setiap minggu, mungkin lebih menguntungkan untuk mempertimbangkan menempatkan lingkungan ini dalam model harga yang berbeda.

Untuk informasi selengkapnya tentang model harga ini, lihat Harga Sesuai [Permintaan Amazon EC2](#).

## Savings Plans

Savings Plans adalah model fleksibel yang menawarkan harga lebih rendah dibandingkan dengan harga On-Demand, dengan imbalan komitmen penggunaan tertentu selama periode satu tahun atau tiga tahun. Jenis Savings Plans berikut ini relevan dengan beban kerja SAP:

- **Compute Savings Plans** — Jenis ini memberikan fleksibilitas paling banyak dan membantu mengurangi biaya Anda hingga 66%. Paket ini berlaku untuk penggunaan instans Amazon Elastic Compute Cloud (Amazon EC2), terlepas dari keluarga instans, ukuran, Availability Zone, Region, sistem operasi, atau tenancy. Paket ini juga berlaku untuk AWS Fargate atau AWS Lambda penggunaan.
- **EC2 Instance Savings Plans** — Jenis ini memberikan harga terendah, menawarkan penghematan hingga 72% sebagai imbalan atas komitmen untuk penggunaan masing-masing keluarga instans EC2 dalam satu Wilayah AWS

Saat memilih Savings Plan, konfirmasi bahwa jenis paket dan tipe instans EC2 Anda didukung oleh produk, database, dan sistem operasi SAP Anda. Untuk informasi selengkapnya, lihat [Aplikasi SAP diAWS: Produk DB/OS dan Amazon EC2 yang didukung - 1656099](#) (catatan SAP).

Untuk informasi selengkapnya tentang model harga ini, lihat [Compute Savings Plans](#).

## Host Khusus

Dalam model ini, Anda membayar untuk setiap host khusus aktif, terlepas dari jumlah atau ukuran instance yang Anda luncurkan di atasnya. Harga Host Khusus bervariasi menurut keluarga instans, Wilayah, dan opsi pembayaran. Dibandingkan dengan harga On-Demand, model ini dapat mengurangi biaya hingga 70%. Model penetapan harga ini memungkinkan Anda untuk menggunakan lisensi aplikasi yang ada, yang mungkin didasarkan pada soket, inti, atau mesin virtual (VM). Contoh di mana model penetapan harga ini biasanya digunakan adalah untuk beban kerja hosting yang berjalan di SQL Server.

Untuk informasi selengkapnya tentang model harga ini, lihat Harga [Host Khusus Amazon EC2](#).

## Instans Cadangan

Ketika instans cadangan ditetapkan ke Availability Zone tertentu, mereka menyimpan kapasitas di zona tersebut. Ini dapat menguntungkan jika permintaan tinggi untuk Availability Zone karena memesan kapasitas berarti Anda dapat meluncurkan instance bila diperlukan. Harga Reserved Instance (RI) ditawarkan dalam dua opsi:

- Standar RI — Opsi ini biasanya digunakan untuk beban kerja yang memiliki sedikit variasi dalam kinerja selama beberapa bulan hingga bertahun-tahun.
- Convertible RI — Opsi ini biasanya digunakan untuk beban kerja yang membutuhkan fleksibilitas untuk mengubah keluarga instans, sistem operasi, atau tenansi, sehingga Anda dapat memanfaatkan peningkatan perangkat keras terbaru.

Untuk informasi selengkapnya tentang model penetapan harga ini, lihat Harga [Instans Cadangan Amazon EC2](#).

# Menggunakan AWS layanan untuk mengidentifikasi peluang optimasi biaya untuk beban kerja SAP

## Proses optimasi biaya

Optimalisasi biaya adalah proses berkelanjutan yang, ketika diterapkan, harus mengungkapkan wawasan yang signifikan dan potensi penghematan di muka yang besar. Proses berkelanjutan ini memberikan wawasan berkelanjutan dan membantu Anda bereaksi dengan cepat jika anomali terdeteksi dalam AWS layanan konsumsi yang dapat meningkatkan biaya.

Proses keseluruhan terdiri dari enam langkah, yang disorot pada gambar berikut.



1. Tinjau pengeluaran — Biasakan diri Anda dengan dasbor AWS Billing konsol. Dasbor ini memberikan wawasan terperinci tentang biaya yang Anda keluarkan untuk dikonsumsi. AWS layanan Identifikasi lima atau lebih sedikit layanan yang akan menjadi fokus siklus tinjauan optimasi biaya ini.

2. Temuan tindakan — Gunakan layanan yang disebutkan dalam [Layanan pengoptimalan biaya](#) untuk mengevaluasi layanan yang Anda targetkan. Setelah Anda menyelesaikan analisis penyelaman mendalam, buat dan terapkan rencana tindakan untuk mengurangi pengeluaran secara keseluruhan. [Layanan pengoptimalan biaya](#) Bagian ini mencakup rekomendasi.
3. Pantau tindakan — Validasi dan pantau perubahan untuk memastikan bahwa tidak ada dampak buruk terhadap lingkungan.
4. Pemeriksaan otomatis — Identifikasi otomatisasi yang dapat menyederhanakan proses mengidentifikasi, memantau, dan memvalidasi peluang pengoptimalan biaya. Misalnya, Anda dapat mengatur peringatan penagihan dengan menggunakan petunjuk di [Membuat alarm penagihan untuk memantau perkiraan AWS biaya](#) (dokumentasi Amazon CloudWatch ). Untuk wawasan tambahan tentang kemampuan otomatisasi standar, lihat [Layanan pengoptimalan biaya](#).
5. Dokumen — Dokumentasikan proses, layanan, dan biaya untuk menyelesaikan perbandingan dan memvalidasi bahwa biaya yang diharapkan telah dikurangi atau distabilkan.
6. Tetapkan irama — Tetapkan irama untuk mengulangi tinjauan pengoptimalan biaya.

Terus ulangi proses optimasi biaya ini dan bandingkan hasil yang didokumentasikan sebelumnya. Dengan menggunakan proses ini, Anda dapat mengurangi biaya secara berulang.

## Layanan pengoptimalan biaya

Bagian ini mengulas beberapa alat AWS layanan dan alat berikut, yang dapat memberikan wawasan dan membantu dalam proses pengoptimalan biaya:

- [AWS Cost Explorer](#)
- [AWS Trusted Advisor](#)
- [AWS Compute Optimizer](#)
- [AWS Config](#)

### Important

Ini AWS layanan mungkin membuat rekomendasi untuk mengoptimalkan biaya Anda. Namun, rekomendasi ini mungkin tidak kompatibel dengan beban kerja SAP yang beroperasi. Sebelum bertindak, konfirmasi bahwa rekomendasi tersebut kompatibel dengan mengoperasikan SAP pada AWS:

- [Aplikasi SAP pada AWS: Produk DB/OS dan Amazon EC2 yang didukung - 1656099](#) (catatan SAP)
- [SAP on AWS: Prasyarat Support - 1656250](#) (catatan SAP)

## AWS Cost Explorer

[AWS Cost Explorer](#) adalah alat dalam AWS Cost Management layanan. Ini membantu Anda memvisualisasikan biaya dan penggunaan yang terkait dengannya AWS layanan. Alat ini menyediakan laporan default yang menampilkan lima biaya AWS layanan teratas Anda yang bertambah. Laporan ini adalah tempat yang sangat baik untuk memulai upaya pengoptimalan biaya Anda.

Sebagai bagian dari strategi pengoptimalan biaya Anda, alat ini memberikan informasi yang diperlukan untuk mengidentifikasi layanan mana yang harus Anda targetkan untuk mengoptimalkan biaya terkait. Anda harus secara teratur menggunakan dan memantau alat ini untuk mengidentifikasi peluang pengoptimalan biaya dan memantau efek terkait biaya dari perubahan di lingkungan Anda.

## AWS Trusted Advisor

[AWS Trusted Advisor](#) memberikan rekomendasi yang membantu Anda mematuhi praktik terbaik dan mengurangi biaya Anda AWS layanan. Konfirmasikan bahwa Trusted Advisor rekomendasi kompatibel dengan lingkungan SAP. Anda harus secara teratur menggunakan Trusted Advisor rekomendasi tentang cara mengurangi biaya dan tetap mematuhi praktik terbaik.

## AWS Compute Optimizer

[AWS Compute Optimizer](#) menggunakan data penggunaan aktual untuk membuat rekomendasi tentang cara menyeimbangkan kinerja dan biaya. Penyediaan sumber daya yang berlebihan menyebabkan biaya infrastruktur yang tidak perlu, dan sumber daya yang kurang penyediaan menyebabkan potensi masalah kinerja. Sebagai bagian dari strategi pengoptimalan biaya, layanan ini membantu Anda menganalisis apakah AWS sumber daya Anda, seperti instans Amazon Elastic Compute Cloud (Amazon EC2), sudah selesai atau kurang disediakan. Gunakan layanan ini saat mengoptimalkan biaya untuk instans EC2, grup Auto Scaling Amazon EC2, volume Amazon Elastic Block Store (Amazon EBS), atau fungsi. AWS Lambda

Rekomendasi dari Compute Optimizer harus direferensikan silang dengan dokumen dukungan SAP. Misalnya, jika Anda memiliki instans EC2 yang besar dan terlalu banyak disediakan di akun

Anda, Pengoptimal Biaya mungkin merekomendasikan untuk mengubah ke jenis instans yang lebih kecil. Konfirmasikan bahwa SAP mendukung jenis instans yang direkomendasikan oleh Cost Optimizer. Jika tidak didukung, Anda masih harus mempertimbangkan untuk mengubah jenis untuk mengoptimalkan biaya. Namun, Anda perlu memilih jenis berbeda yang didukung SAP. Untuk informasi selengkapnya tentang jenis instans yang tersedia, lihat [Jenis instans](#) (dokumentasi Amazon EC2).

## AWS Config

[AWS Config](#) terus memantau AWS sumber daya Anda dan memberikan pandangan terperinci tentang bagaimana AWS sumber daya di akun Anda dikonfigurasi. AWS Config dapat mengambil tindakan spesifik berdasarkan temuan yang dievaluasi. Meskipun layanan ini tidak terkait langsung dengan biaya, layanan ini memberikan solusi otomatis untuk memeriksa apakah lingkungan SAP Anda sesuai. Anda dapat menggunakan aturan yang ada atau membuat aturan khusus untuk mengatur peringatan. Misalnya, Anda dapat mengonfigurasi peringatan otomatis jika jenis instans EC2 diubah atau jika volume EBS terlepas. Anda juga dapat menyesuaikan aturan untuk mengotomatiskan remediasi dan mengembalikan sumber daya ke kepatuhan. Untuk informasi lebih lanjut tentang penggunaan AWS Config untuk lingkungan SAP, lihat [bagian I](#) dan [bagian II](#) dari Audit sistem SAP Anda dengan AWS Config (posting AWS blog).

Pertimbangkan untuk menggunakan layanan ini untuk deteksi otomatis skenario terkait biaya yang terkait dengan perubahan sumber daya. AWS

## Layanan pengoptimalan penyimpanan

Dengan penyimpanan, Anda harus menyeimbangkan kinerja dengan biaya. Bagian ini berisi beberapa rekomendasi untuk konfigurasi penyimpanan potensial, berdasarkan HANA dan ukuran instance yang dipilih.

Bagian ini berisi layanan dan alat berikut:

- [Manajer Volume Logis](#)
- [Kebijakan siklus hidup Amazon S3](#)
- [Volume EBS Amazon](#)
- [Kebijakan siklus hidup Amazon EFS](#)

## Manajer Volume Logis

Dalam beberapa kasus, Anda mungkin ingin menggunakan Logical Volume Manager dan menggabungkan (juga dikenal sebagai stripe) beberapa disk untuk menyediakan operasi input/output target per detik (IOPS) dan throughput. Konfigurasi ini biasanya lebih menguntungkan biaya daripada satu volume kelas yang lebih tinggi. Namun, pendekatan ini dapat menyebabkan beberapa ruang tambahan yang tidak terpakai yang harus Anda seimbangkan dengan pertimbangan ukuran dan biaya.

## Kebijakan siklus hidup Amazon S3

Anda dapat menggunakan bucket [Amazon Simple Storage Service \(Amazon S3\)](#) untuk penyimpanan cadangan basis data SAP jangka panjang dan jangka pendek. Ini membantu Anda menyeimbangkan biaya dengan kebutuhan akan akses dan ketahanan. Mengonfigurasi kebijakan siklus hidup mengotomatiskan kebijakan penyimpanan untuk file cadangan Anda. Dalam kebijakan siklus hidup, Anda dapat mengatur transisi dari satu kelas penyimpanan Amazon S3 ke kelas penyimpanan berbiaya lebih rendah untuk pencadangan yang memerlukan retensi jangka panjang.

Sebagai bagian dari strategi pengoptimalan biaya Anda, otomatisasi ini menyediakan aturan yang diperlukan untuk menjaga biaya Amazon S3 Anda selaras dengan kebutuhan retensi spesifik Anda. Tinjau otomatisasi ini pada jadwal berkala untuk mengonfirmasi bahwa konfigurasi dioptimalkan dan konten bucket S3 mematuhi aturan konfigurasi yang diharapkan.

## Volume EBS Amazon

Volume [Amazon Elastic Block Store \(Amazon EBS\)](#) digunakan untuk menyimpan data persisten Anda seperti sistem operasi, file biner SAP, dan file database SAP Anda. Ukuran volume ini sangat penting bila diimbangi dengan optimasi biaya. Sangat mudah untuk meningkatkan ukuran volume, oleh karena itu Anda harus menyediakan volume dengan ruang yang cukup untuk menampung volume data aktual ditambah persentase kecil untuk mengakomodasi pertumbuhan. Volume overprovisioned ukuran yang tepat dapat menjadi padat karya dan membutuhkan waktu henti yang direncanakan. Jika ini adalah di mana persentase besar dari pengeluaran Anda terjadi meskipun akan masuk akal untuk memulai kegiatan semacam itu.

Sebagai bagian dari strategi pengoptimalan biaya Anda, peninjauan volume EBS yang dlebih-lebihkan harus dilakukan setiap siklus peninjauan lainnya. Jika ada peluang berkelanjutan di area khusus ini, otomatisasi dapat dikembangkan untuk meringankan beberapa upaya manual yang terlibat.

## Kebijakan siklus hidup Amazon EFS

[Amazon Elastic File System \(Amazon EFS\)](#) secara otomatis menskalakan penyimpanan saat file ditambahkan atau dihapus. Dalam lingkungan SAP, ini biasanya digunakan sebagai transportasi, antarmuka, dan sistem file cluster Anda. Selain itu, Anda dapat menggunakannya untuk menyimpan sementara perangkat lunak untuk proyek aktif. Meskipun, dalam jangka panjang, Anda disarankan untuk menyimpan perangkat lunak dalam ember S3.

Untuk sistem file antarmuka SAP, pertimbangkan untuk mengonfigurasi kebijakan siklus [hidup EFS](#). Dengan kebijakan siklus hidup, Anda dapat mengurangi biaya penyimpanan dengan mentransisikan file dengan persyaratan retensi tertentu ke kelas penyimpanan berbiaya lebih rendah. Jika Anda menggunakan estimasi yang diterima industri, 80% data Anda jarang diakses. Memindahkan data ini ke kelas penyimpanan berbiaya lebih rendah adalah peluang utama untuk mengurangi biaya Anda. Untuk informasi selengkapnya, lihat [Amazon EFS Infrequent Access](#).

Sebagai bagian dari strategi pengoptimalan biaya Anda, tinjau kebijakan siklus hidup EFS setiap siklus peninjauan lainnya. Jika ada peluang berkelanjutan, aktifkan kebijakan siklus hidup EFS untuk mengotomatiskan proses dan menerapkan rutinitas pembersihan.

## Pertanyaan yang Sering Diajukan

Bagian ini memberikan jawaban atas pertanyaan yang sering diajukan tentang pengoptimalan biaya, migrasi, dan otomatisasi untuk beban kerja SAPAWS.

### BagaimanaAWSpekerjaan penetapan harga?

denganAWS, Anda hanya membayar untuk layanan individual yang Anda gunakan. Model harga tersedia yang memberikan diskon berdasarkan komitmen dan volume. Untuk informasi selengkapnya, lihat [AWS Harga](#).

### Di mana saya dapat menemukan informasi lebih lanjut tentangAWSRencana Tabungan?

AWSmenawarkan berbagai[Paket Tabungan](#)dirancang untuk memenuhi kebutuhan dan anggaran Anda.

### Bagaimana cara kerja SAPAWS?

Untuk gambaran umum tentang cara menggunakan solusi SAPAWS, lihat[Memulai SAP diAWS](#).

### Bagaimana cara memigrasikan beban kerja SAP keAWS?

Anda harus mulai mempertimbangkan peluang pengoptimalan biaya sebelum migrasi. Misalnya, memperkirakan total biaya kepemilikan (TCO) adalah perhitungan penting ketika mendefinisikan arsitektur konseptual, negara masa depan Anda selama fase penilaian migrasi. Untuk ikhtisar tentang cara memigrasikan beban kerja SAP Anda keAWS, lihat[SAP padaAWSmetodologi migrasi](#).AWS Layanan Profesional danAWS Partner Networkmenawarkan layanan untuk membantu Anda dengan migrasi Anda.

### Apakah mengotomatisasi operasi SAP mengurangi biaya?

Otomatisasi dapat membantu Anda mengurangi biaya infrastruktur cloud dan mengurangi biaya yang terkait dengan administrator SAP Anda yang sering melakukan tugas yang manual, berulang, dan memakan waktu. Misalnya, Anda dapat menggunakanAWS Systems Manageruntuk secara otomatis

memulai dan menghentikan sistem SAP, mengurangi biaya operasi Anda. Selain itu, Anda dapat menggunakan Amazon EC2 Auto Scaling untuk secara otomatis menambah atau menghapus instans Amazon EC2 sesuai dengan permintaan server aplikasi SAP, mengoptimalkan kinerja sekaligus meminimalkan biaya. Untuk informasi selengkapnya tentang peluang otomatisasi, lihat [Strategi otomatisasi untuk operasi SAP di AWS Cloud](#).

## Jenis layanan apa yang dilakukan AWS penawaran untuk SAP?

AWS mengelola infrastruktur fisik sistem SAP hingga lapisan virtualisasi. Anda mengelola sistem operasi dan aplikasi SAP dan database yang berjalan di atas lapisan virtualisasi. Jika layanan terkelola diperlukan, AWS memiliki jaringan mitra yang menawarkan konsultasi SAP dan layanan terkelola AWS. Untuk menemukan AWS Partner yang telah menunjukkan kemahiran teknis dan keberhasilan yang terbukti dalam implementasi dan migrasi SAP, lihat [AWS Mitra Kompetensi SAP](#).

## Kasus penggunaan SAP mana yang melakukannya AWS dukungan?

Pelanggan dan mitra SAP menggunakan AWS untuk kasus penggunaan mulai dari menjalankan sistem uji SAP tunggal hingga hosting lingkungan produksi SAP yang lengkap. Untuk mempelajari lebih lanjut tentang bagaimana pelanggan dan mitra SAP menggunakan AWS, lihat [Memulai SAP di AWS](#).

## Langkah selanjutnya

Seperti halnya proses baru, beberapa siklus pertama menggunakan [Proses optimasi biaya](#). Dalam panduan ini mungkin memakan waktu. Namun, prosesnya dapat menghasilkan penghematan yang signifikan. Mulailah proses dengan mengidentifikasi dua hingga tiga layanan yang memiliki biaya tinggi atau melebihi biaya yang Anda harapkan. Dalam waktu singkat, Anda dapat memiliki pegangan yang lebih baik pada pengeluaran, dengan cepat mengidentifikasi layanan yang harganya lebih dari yang diharapkan, dan menerapkan strategi untuk mengurangi biaya tersebut.

Seiring waktu, banyak perusahaan terus menyadari peningkatan tambahan dalam efisiensi dan biaya. [Manfaat Bisnis Menjalankan Beban Kerja SAP di AWS](#), IDG MarketPulse whitepaper, melaporkan peningkatan efisiensi dari 28% menjadi 43% setelah dua tahun mengoperasikan SAP AWS. Kami menyarankan Anda terus mengulangi proses pengoptimalan biaya dalam panduan ini sehingga Anda dapat mewujudkan penghematan di luar peluang yang diidentifikasi sebelumnya.

Jika Anda membutuhkan panduan tambahan, Anda dapat melibatkan tim SAP Global Specialty Practice dari AWS Layanan Profesional untuk membantu Anda menentukan dan mengimplementasikan rencana tindakan Anda untuk mengoptimalkan biaya menjalankan beban kerja SAP Anda AWS.

# Sumber daya

## AWSmodel harga

- [BagaimanaAWSpekerjaan penetapan harga?](#)
- [AWSPaket Tabungan](#)
- [Harga Sesuai Permintaan Amazon EC2](#)
- [Harga Host Khusus Amazon EC2](#)
- [Harga Instans Cadangan Amazon EC2](#)

## Dokumentasi dukungan SAP

- [Aplikasi SAP padaAWS: Produk DB/OS dan Amazon EC2 yang didukung - 1656099](#)(Catatan SAP)
- [SAP padaAWS: Prasyarat dukungan - 1656250](#)(Catatan SAP)

## Sumber daya lainnya

- [AWSKalkulator Harga](#)
- [AWSLayanan Profesional](#)
- [AWSMitra Kompetensi SAP](#)
- [Memulai SAP diAWS](#)
- [AWSuntuk SAP](#)(AWSblog)
- [Strategi otomatisasi untuk operasi SAP diAWS Cloud](#)(AWSBimbingan Preskriptif)
- [SAP padaAWSmetodologi migrasi](#)(AWSBimbingan Preskriptif)
- [Manfaat Bisnis Menjalankan Beban Kerja SAPAWS](#)(IDGMarketPulsewhitepaper)

## Riwayat dokumen

Tabel berikut menjelaskan perubahan signifikan pada panduan ini. Jika Anda ingin diberi tahu tentang pembaruan di masa mendatang, Anda dapat berlangganan [Umpan RSS](#).

Perubahan	Deskripsi	Tanggal
<a href="#">Publikasi awal</a>	—	Juli 18, 2022

# 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: Migrasikan 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 memproses atau memodifikasi 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 AWS layanan menerimanya.

## Cloud Center of Excellence (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 CCoE](#) 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 CCoE, 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 AWS CodeCommit 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, AWS Panorama menawarkan perangkat yang menambahkan CV ke jaringan kamera lokal, dan Amazon SageMaker menyediakan algoritme 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 umumnya 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 kekritisannya 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 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 di tempat. 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 database](#).

## 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](#).

### 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.

### 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 pipa CI/CD, 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.

## 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.

## G

### 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.

## 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 (OU). 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, 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.

### 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.

|

### 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.

|

## IloT

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 VPC masuk, keluar, dan inspeksi 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 selengkapnya, 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 VPC (dalam hal yang sama atau berbeda Wilayah AWS), internet, dan jaringan lokal. [Arsitektur Referensi AWS Keamanan](#) merekomendasikan pengaturan akun Jaringan Anda dengan VPC masuk, keluar, dan inspeksi 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?](#)

## interpretabilitas

Karakteristik model pembelajaran mesin yang menggambarkan sejauh mana manusia dapat memahami bagaimana prediksi model bergantung pada inputnya. Untuk informasi selengkapnya, 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](#).

### 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](#).

### 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

AWS layanan 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 melalui API yang terdefinisi dengan baik 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 API ringan. 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 terbaik dan pelajaran yang dipetik 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 VPC masuk, keluar, dan inspeksi 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 dalam organisasi di \(lihat kebijakan kontrol layanan\). AWS Organizations](#)

### persistensi poliglott

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. Untuk informasi selengkapnya, lihat [Mengaktifkan persistensi data di layanan mikro](#).

### 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 pengoptimalan 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 oleh Desain

Pendekatan dalam rekayasa sistem yang memperhitungkan privasi di seluruh proses rekayasa.

## zona host pribadi

Container yang menyimpan informasi tentang bagaimana Anda ingin Amazon Route 53 merespons kueri DNS untuk domain dan subdomainnya dalam satu atau beberapa VPC. 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.

## pseudonimisasi

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

## terbitkan/berlangganan (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 \(RACI\)](#).

### 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 hilangnya 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](#).

### Wilayah

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 prinsip 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](#).

## rotasi

Proses memperbarui [rahasia](#) secara berkala untuk membuatnya lebih sulit bagi penyerang untuk mengakses kredensial.

## 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 AWS Management Console 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 kredensial pengguna, yang Anda simpan dalam bentuk terenkripsi. Ini terdiri dari nilai rahasia dan metadatanya. 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.

## 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, menambah instans Amazon EC2, atau memutar kredensial.

enkripsi sisi server

Enkripsi data di tujuannya, oleh AWS layanan yang menerimanya.

kebijakan kontrol layanan (SCP)

Kebijakan yang menyediakan kontrol terpusat atas izin untuk semua akun di organisasi. AWS Organizations SCP menentukan pagar pembatas atau menetapkan batasan pada tindakan yang dapat didelegasikan oleh administrator kepada pengguna atau peran. Anda dapat menggunakan SCP sebagai 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 AWS layanan. Anda dapat menggunakan endpoint untuk terhubung secara terprogram ke layanan target. Untuk informasi selengkapnya, lihat [AWS layanan 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.

# T

## tag

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

## 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 VPC dan jaringan lokal Anda. 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. Untuk informasi lebih lanjut, lihat panduan [Mengukur ketidakpastian dalam sistem pembelajaran mendalam](#).

### 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 VPC yang memungkinkan Anda 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 [Kerangka Kualifikasi Beban Kerja AWS](#).

## 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.

### 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.