

Kerangka Kerja AWS Well-Architected

# Pilar Optimasi Biaya



# Pilar Optimasi Biaya: Kerangka Kerja AWS Well-Architected

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

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

---

# Table of Contents

|  |    |
|--|----|
| Abstrak dan pengantar .....  | 1  |
| Abstrak .....  | 1  |
| Pengantar .....  | 1  |
| Optimasi biaya .....   | 3  |
| Prinsip desain .....   | 3  |
| Definisi .....   | 4  |
| Mempraktikkan Manajemen Keuangan Cloud .....                                       | 6  |
| COST01-BP01 Membangun kepemilikan pengoptimalan biaya .....                        | 9  |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP02 Buat kemitraan antara bagian keuangan dan teknologi .....              | 12 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP03 Tetapkan prakiraan dan anggaran cloud .....                            | 17 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP04 Mengimplementasikan kesadaran biaya dalam proses organisasi Anda ..... | 20 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP05 Melaporkan dan memberi tahu tentang pengoptimalan biaya .....          | 22 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP06 Memantau biaya secara proaktif .....                                   | 25 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP07 Mengikuti perkembangan rilis layanan baru .....                        | 27 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP08 Menciptakan budaya sadar biaya .....                                   | 29 |
| Panduan implementasi .....   | 9  |
| Sumber daya .....  | 11 |
| COST01-BP09 Menghitung nilai bisnis dari optimasi biaya .....                      | 31 |
| Panduan implementasi .....   | 9  |
| Sumber Daya .....  | 11 |

|   |    |
|---|----|
| Kesadaran akan penggunaan dan pengeluaran .....   | 34 |
| Tata Kelola .....   | 34 |
| COST02-BP01 Mengembangkan kebijakan berdasarkan keperluan organisasi Anda .....                                   | 35 |
| COST02-BP02 Mengimplementasikan sasaran dan target .....  | 39 |
| COST02-BP03 Mengimplementasikan struktur akun .....   | 42 |
| COST02-BP04 Mengimplementasikan grup dan peran .....  | 46 |
| COST02-BP05 Mengimplementasikan kontrol biaya .....   | 48 |
| COST02-BP06 Melacak siklus hidup proyek .....   | 51 |
| Pantau biaya dan penggunaan .....   | 53 |
| COST03-BP01 Mengonfigurasi sumber informasi yang mendetail .....  | 54 |
| COST03-BP02 Menambahkan informasi organisasi ke biaya dan penggunaan .....  | 56 |
| COST03-BP03 Mengidentifikasi kategori atribusi biaya .....  | 58 |
| COST03-BP04 Membangun metrik organisasi .....   | 62 |
| COST03-BP05 Mengonfigurasi alat manajemen penagihan dan biaya .....   | 63 |
| COST03-BP06 Mengalokasikan biaya berdasarkan metrik beban kerja .....   | 66 |
| Sumber daya penonaktifan .....  | 68 |
| COST04-BP01 Lacak sumber daya sepanjang masa pakainya .....   | 68 |
| COST04-BP02 Mengimplementasikan proses penonaktifan .....   | 70 |
| COST04-BP03 Menonaktifkan sumber daya .....   | 73 |
| COST04-BP04 Menonaktifkan sumber daya secara otomatis .....   | 74 |
| COST04-BP05 Menegakkan kebijakan retensi data .....   | 75 |
| Sumber daya hemat biaya .....   | 77 |
| Evaluasi biaya saat memilih layanan .....   | 77 |
| COST05-BP01 Identifikasi persyaratan organisasi untuk biaya .....   | 77 |
| COST05-BP02 Menganalisis semua komponen beban kerja .....   | 79 |
| COST05-BP03 Menjalankan analisis menyeluruh setiap komponen .....   | 81 |
| COST05-BP04 Memilih perangkat lunak dengan lisensi hemat biaya .....  | 84 |
| COST05-BP05 Memilih komponen beban kerja ini untuk mengoptimalkan biaya selaras dengan prioritas organisasi ..... | 85 |
| COST05-BP06 Melakukan analisis biaya untuk penggunaan berbeda seiring waktu .....                                 | 88 |
| Pilih tipe, ukuran, dan jumlah sumber daya yang tepat .....   | 90 |
| COST06-BP01 Melakukan pemodelan biaya .....   | 90 |
| COST06-BP02 Memilih jenis, ukuran, dan jumlah sumber daya berdasarkan data .....                                  | 92 |
| COST06-BP03 Pilih tipe, ukuran, dan jumlah sumber daya secara otomatis berdasarkan metrik .....                   | 95 |
| Memilih model harga terbaik .....   | 98 |

|  |     |
|--|-----|
| COST07-BP01 Melakukan analisis model harga .....                                       | 98  |
| COST07-BP02 Memilih Wilayah berdasarkan biaya .....                                    | 101 |
| COST07-BP03 Memilih perjanjian pihak ketiga dengan ketentuan hemat biaya .....         | 104 |
| COST07-BP04 Mengimplementasikan model harga untuk semua komponen beban kerja ini ..... | 105 |
| COST07-BP05 Melakukan analisis model harga di tingkat akun manajemen .....             | 108 |
| Merencanakan transfer data .....   | 110 |
| COST08-BP01 Melakukan pemodelan transfer data .....                                    | 111 |
| COST08-BP02 Memilih komponen untuk mengoptimalkan biaya transfer data .....            | 113 |
| COST08-BP03 Mengimplementasikan layanan untuk mengurangi biaya transfer data .....     | 115 |
| Kelola sumber daya pasokan dan permintaan .....  | 118 |
| COST09-BP01 Melakukan analisis pada permintaan beban kerja .....                       | 118 |
| Panduan implementasi .....   | 9   |
| Sumber daya .....  | 11  |
| COST09-BP02 Implementasikan buffer atau throttle untuk mengelola permintaan .....      | 121 |
| Panduan implementasi .....   | 9   |
| Sumber daya .....  | 11  |
| COST09-BP03 Menyediakan sumber daya secara dinamis .....                               | 124 |
| Panduan implementasi .....   | 9   |
| Langkah implementasi .....   | 10  |
| Sumber daya .....  | 11  |
| Pengoptimalan dari waktu ke waktu .....  | 132 |
| Tentukan proses peninjauan dan analisis beban kerja Anda secara rutin .....            | 132 |
| COST10-BP01 Mengembangkan proses peninjauan beban kerja .....                          | 132 |
| COST10-BP02 Meninjau dan menganalisis beban kerja ini secara teratur .....             | 135 |
| Otomatisasi operasi .....  | 137 |
| COST11-BP01 Melakukan otomatisasi untuk operasi .....                                  | 137 |
| Kesimpulan .....   | 141 |
| Kontributor .....  | 142 |
| Sumber Bacaan Lebih Lanjut .....   | 143 |
| Revisi dokumen .....   | 144 |

# Pilar Optimasi Biaya - Kerangka Kerja AWS Well-Architected

Tanggal publikasi: 6 Desember 2023 ([Revisi dokumen](#))

## Abstrak

Laporan resmi ini berfokus pada pilar optimasi biaya Kerangka Kerja Amazon Web Services (AWS) Well-Architected. Laporan ini menyediakan panduan untuk membantu pelanggan menerapkan praktik terbaik dalam desain, pengiriman, dan pemeliharaan lingkungan AWS.

Beban kerja dengan optimasi biaya menggunakan semua sumber daya sepenuhnya, mencapai hasil pada titik harga serendah mungkin, dan memenuhi persyaratan fungsional Anda. Laporan resmi ini memberikan panduan mendalam untuk membangun kemampuan dalam organisasi Anda, merancang beban kerja, memilih layanan, mengonfigurasi dan mengoperasikan layanan, serta menerapkan teknik optimasi biaya.

## Pengantar

Dengan [Kerangka Kerja AWS Well-Architected](#) Anda dapat memahami keputusan yang Anda ambil selama membangun beban kerja di AWS. Kerangka Kerja ini menyediakan praktik terbaik arsitektur untuk mendesain serta mengoperasikan beban kerja yang andal, aman, efisien, hemat biaya, dan ramah lingkungan di cloud. Kerangka Kerja ini menunjukkan cara untuk secara terus menerus menilai arsitektur Anda berdasarkan praktik terbaik dan mengidentifikasi area yang perlu diperbaiki. Kami percaya bahwa memiliki beban kerja yang didesain dengan baik akan meningkatkan peluang keberhasilan bisnis.

Enam pilar landasan kerangka kerja:

- Keunggulan Operasional
- Keamanan
- Keandalan
- Efisiensi Kinerja
- Optimasi Biaya
- Pelestarian Lingkungan

Laporan ini berfokus pada pilar optimasi biaya dan cara pembuatan arsitektur beban kerja menggunakan sumber daya dan layanan yang paling efektif untuk mencapai hasil bisnis dengan biaya serendah mungkin.

Anda akan mempelajari cara menerapkan praktik terbaik pilar optimasi biaya di organisasi Anda. Optimasi biaya dapat menjadi tantangan dalam solusi on-premise tradisional karena Anda harus memprediksi kapasitas dan kebutuhan bisnis di masa mendatang sekaligus menjalankan proses pengadaan yang kompleks. Mengadopsi praktik dalam laporan ini akan membantu organisasi Anda mencapai tujuan berikut:

- Mempraktikkan Manajemen Keuangan Cloud
- Kesadaran akan penggunaan dan pengeluaran
- Sumber daya hemat biaya
- Kelola sumber daya pasokan dan permintaan
- Pengoptimalan dari waktu ke waktu

Laporan ini ditujukan bagi Anda yang memiliki peran di bidang keuangan dan teknologi, seperti kepala pejabat teknologi (chief technology officer/CTO), kepala pejabat keuangan (chief financial officer/CFO), arsitek, developer, pengendali keuangan, perencana keuangan, analis bisnis, dan anggota tim operasional. Laporan ini tidak menyediakan detail implementasi atau pola arsitektural, tetapi menyertakan referensi ke sumber daya yang relevan.

# Optimasi biaya

Optimasi biaya adalah proses penyempurnaan dan perbaikan yang berkelanjutan selama siklus hidup beban kerja. Praktik dalam laporan ini membantu Anda membangun dan mengoperasikan beban kerja yang mempertimbangkan biaya yang mencapai hasil bisnis sambil meminimalkan biaya dan memungkinkan organisasi Anda memaksimalkan laba atas investasinya.

Topik

- [Prinsip desain](#)
- [Definisi](#)

## Prinsip desain

Pertimbangkan prinsip desain untuk optimasi biaya berikut ini:

**Implementasikan manajemen keuangan cloud:** Untuk mencapai keberhasilan keuangan dan mempercepat realisasi nilai bisnis di cloud, Anda harus berinvestasi dalam Manajemen Keuangan Cloud. Organisasi Anda harus mendedikasikan waktu dan sumber daya yang diperlukan untuk membangun kemampuan dalam domain teknologi baru ini dan dalam manajemen penggunaan. Serupa dengan kemampuan Keamanan atau Operasi Anda, Anda perlu membangun kemampuan melalui pembangunan pengetahuan, program, sumber daya, dan proses untuk membantu Anda menjadi organisasi yang hemat biaya.

**Adopsi model pemakaian:** Bayar hanya untuk sumber daya komputasi yang Anda gunakan, dan tingkatkan atau turunkan penggunaan tergantung kebutuhan bisnis Anda. Misalnya, lingkungan pengembangan dan pengujian umumnya hanya digunakan selama delapan jam sehari selama minggu operasional. Anda dapat menghentikan sumber daya ini ketika tidak digunakan untuk mendapatkan potensi penghematan biaya sebesar 75% (40 jam dibandingkan 168 jam).

**Ukur keseluruhan efisiensi:** Ukur output bisnis beban kerja serta biaya terkait pengiriman. Gunakan data ini untuk memahami keuntungan yang Anda peroleh dari peningkatan output, peningkatan fungsionalitas, dan penurunan biaya,

**Hentikan pembelanjaan untuk pekerjaan berat yang tidak terdiferensiasi:** AWS melakukan pekerjaan berat operasi pusat data seperti pembuatan rak, penumpukan, dan pemberian daya pada server. AWS juga menyingkirkan beban operasional berupa pengelolaan sistem operasi dan aplikasi dengan



layanan terkelola. Dengan demikian Anda dapat berkonsentrasi pada pelanggan dan proyek bisnis Anda dan mengalihkan fokus dari infrastruktur IT.

Analisis dan pengeluaran atribut: Cloud memudahkan identifikasi biaya dan penggunaan beban kerja secara akurat, yang kemudian memungkinkan atribusi biaya IT yang transparan ke aliran pendapatan dan setiap pemilik beban kerja. Hal ini membantu mengukur laba atas investasi (ROI) dan memberi pemilik beban kerja sebuah peluang untuk mengoptimalkan sumber daya mereka dan memangkas biaya.

## Definisi

Terdapat lima area fokus untuk optimasi biaya di cloud:

- Mempraktikkan Manajemen Keuangan Cloud
- Kesadaran akan penggunaan dan pengeluaran
- Sumber daya yang hemat biaya
- Mengelola sumber daya pasokan dan permintaan
- Pengoptimalan dari waktu ke waktu

Serupa dengan pilar lain di dalam Well-Architected Framework, terdapat kompromi yang perlu dipertimbangkan untuk optimasi biaya. Misalnya, pilihan untuk mengoptimalkan kecepatan masuk pasar, atau untuk biaya. Di beberapa kasus, pilihan terbaik adalah mengoptimalkan untuk kecepatan—memasuki pasar dengan cepat, mengirimkan fitur baru, atau memenuhi tenggat—daripada berinvestasi untuk optimasi biaya di awal.

Keputusan desain terkadang disetir oleh sikap terburu-buru, bukan oleh data, dan selalu ada godaan untuk melakukan kompensasi berlebihan, alih-alih meluangkan waktu untuk membandingkan opsi-opsi deployment dengan biaya paling optimal. Kompensasi yang berlebihan dapat berakibat pada deployment dengan pengadaan yang berlebihan dan kurang optimal. Namun, mungkin menjadi pilihan yang wajar jika Anda harus melakukan “angkat dan geser” sumber daya dari lingkungan on-premise ke cloud lalu melakukan optimasi setelahnya.

Menginvestasikan energi yang cukup dalam strategi optimasi biaya di awal memungkinkan Anda untuk lebih mudah mewujudkan manfaat cloud pada aspek ekonomi dengan memastikan kepatuhan yang konsisten terhadap praktik terbaik dan menghindari pengadaan berlebihan yang tidak diperlukan: Bagian berikutnya menyediakan teknik dan praktik terbaik untuk implementasi

Manajemen Keuangan Cloud dan optimasi biaya di awal dan yang sedang berjalan untuk beban kerja Anda.

# Mempraktikkan Manajemen Keuangan Cloud

Untuk mengelola keuangan cloud Anda perlu mengembangkan proses keuangan Anda saat ini agar dapat mapan dan beroperasi dengan transparansi, kontrol, perencanaan, dan pengoptimalan biaya untuk lingkungan AWS Anda.

Penerapan model perencanaan air terjun (waterfall planning) statis, penganggaran IT, dan evaluasi biaya konvensional ke penggunaan cloud dinamis dapat menimbulkan risiko, menyebabkan perencanaan yang tidak akurat, dan berakibat pada penurunan visibilitas. Akhirnya, hal ini menyebabkan hilangnya peluang untuk mengoptimalkan dan mengontrol biaya secara efektif dan mewujudkan nilai bisnis jangka panjang. Untuk menghindari hambatan-hambatan ini, kelola biaya secara aktif di sepanjang perjalanan cloud, baik Anda sedang membangun aplikasi secara native di cloud, memigrasikan beban kerja ke cloud, atau memperluas adopsi layanan cloud Anda.

Manajemen Keuangan Cloud (CFM) memungkinkan organisasi keuangan, produk, teknologi, dan bisnis untuk mengelola, mengoptimalkan, dan merencanakan biaya seiring penggunaan dan skala mereka di AWS bertambah. Tujuan utama CFM adalah memungkinkan pelanggan untuk mencapai hasil bisnis mereka dengan cara paling hemat biaya dan mempercepat penciptaan nilai ekonomi dan bisnis sambil menemukan keseimbangan yang tepat antara ketangkasan dan kontrol.

Solusi CFM membantu mentransformasi bisnis Anda melalui transparansi, kontrol, prakiraan, dan pengoptimalan biaya. Solusi-solusi tersebut juga dapat mewujudkan budaya sadar biaya yang mendorong akuntabilitas di semua tim dan fungsi. Tim keuangan dapat melihat asal biaya, menjalankan operasi dengan biaya tak terduga yang minimal, merencanakan penggunaan cloud dinamis, dan menghemat pengeluaran cloud di saat tim menskalakan adopsi mereka di cloud. Membagikan hal ini kepada tim rekayasawan dapat menyediakan konteks keuangan yang diperlukan untuk pemilihan, penggunaan, dan pengoptimalan sumber daya mereka.

AWS CFM menawarkan serangkaian kemampuan untuk mengelola, mengoptimalkan, dan merencanakan biaya cloud sambil mempertahankan ketangkasan bisnis. CFM tidak hanya penting untuk mengelola biaya secara efektif, tetapi juga untuk memverifikasi bahwa investasi telah mendorong hasil bisnis yang diharapkan. Berikut ini adalah empat pilar Kerangka Kerja Manajemen Keuangan Cloud di AWS Cloud: melihat, menghemat, merencanakan, dan menjalankan. Setiap pilar ini memiliki serangkaian aktivitas dan kemampuan.

Empat pilar Manajemen Keuangan Cloud.

- **Melihat:** Bagaimana saat ini Anda mengukur, memantau, dan menciptakan akuntabilitas untuk pengeluaran cloud Anda? Jika Anda baru di AWS atau berencana menggunakan AWS, apakah Anda memiliki rencana untuk membangun visibilitas biaya dan penggunaan?

Untuk memahami biaya AWS Anda dan mengoptimalkan pengeluaran, Anda perlu mengetahui dari mana semua biaya tersebut berasal. Untuk itu diperlukan sebuah struktur yang cermat untuk akun dan sumber daya Anda, yang membantu organisasi keuangan Anda melacak alur pengeluaran dan menjaga akuntabilitas tim terhadap bagian pendapatan mereka.

Layanan AWS: AWS Control Tower, AWS Organizations, Tag alokasi biaya, Kebijakan tag, AWS Resource Groups, Kategori Biaya AWS, AWS Cost Explorer, AWS Cost and Usage Report, RI dan SP

Sumber daya: Praktik Terbaik Pemberian Tag AWS, Kategori Biaya AWS

- **Penghematan:** Apa pendorong optimasi biaya yang saat ini Anda gunakan untuk mengoptimalkan pengeluaran Anda? Jika Anda tidak sedang menggunakan AWS, apakah Anda mengenal pengoptimalan berbasis penggunaan umum dan berbasis model harga?

Dalam prinsip penghematan, kami mengoptimalkan biaya dengan rekomendasi harga dan sumber daya. Mengoptimalkan biaya dimulai dengan memiliki strategi yang ditetapkan dengan baik untuk model operasi cloud baru Anda. Idealnya, hal ini harus dimulai sedini mungkin dalam perjalanan cloud Anda, menyiapkan fondasi untuk budaya sadar biaya yang diperkuat oleh proses dan perilaku yang tepat.

Anda memiliki banyak cara yang berbeda untuk mengoptimalkan biaya cloud. Salah satunya adalah memilih model pembelian yang tepat (RI dan SP) atau apakah beban kerja Anda tetap dan dalam kontainer sehingga Anda dapat mengadopsi Instans Spot Amazon EC2. Selain itu, skalakan beban kerja Anda menggunakan Grup Amazon EC2 Auto Scaling.

Layanan AWS: RI dan SP, Grup Amazon EC2 Auto Scaling, Instans Spot

Sumber daya: Instans Terpesan, Savings Plans, Praktik terbaik untuk menangani Amazon EC2

- **Rencana:** Bagaimana saat ini Anda merencanakan penggunaan dan pengeluaran cloud masa depan? Apakah Anda memiliki metodologi untuk menghitung penciptaan nilai untuk migrasi baru? Apakah Anda telah mengembangkan proses penganggaran dan prakiraan saat ini untuk mengadopsi berbagai penggunaan cloud?

Prinsip merencanakan artinya meningkatkan kualitas perencanaan Anda dengan penganggaran dan prakiraan yang fleksibel. Setelah Anda membangun visibilitas dan kontrol biaya, Anda

kemungkinan perlu merencanakan dan menetapkan ekspektasi untuk pengeluaran pada proyek-proyek cloud. AWS memberi Anda fleksibilitas untuk membangun proses prakiraan dan penganggaran yang dinamis agar Anda dapat terus mengetahui apakah biaya mengikuti, atau melampaui, batas anggaran.

Layanan AWS: AWS Cost Explorer, AWS Cost and Usage Report, AWS Budgets

Sumber daya: Prakiraan Berbasis Penggunaan, Laporan dan Pemberitahuan Anggaran AWS

- Menjalankan: Apa saja proses dan alat operasional yang saat ini Anda gunakan untuk mengelola pengeluaran cloud Anda, dan siapa yang memimpin upaya-upaya tersebut? Apakah Anda sudah memikirkan bagaimana operasi harian akan berjalan setelah Anda mulai menggunakan AWS?

Prinsip menjalankan sejatinya adalah mengelola tagihan dan kontrol biaya. Anda dapat membangun pagar pembatas dan menetapkan tata kelola untuk memastikan pengeluaran tetap selaras dengan anggaran. AWS menyediakan sejumlah alat untuk membantu Anda memulai.

Layanan AWS: Konsol AWS Billing and Cost Management, AWS Identity and Access Management, Kebijakan Kontrol Layanan (SCP), AWS Service Catalog, AWS Cost Anomaly Detection, AWS Budgets

Sumber daya: Mulai Menggunakan Konsol AWS Billing

Berikut ini adalah praktik-praktik terbaik Manajemen Keuangan Cloud:

Praktik terbaik

- [COST01-BP01 Membangun kepemilikan pengoptimalan biaya](#)
- [COST01-BP02 Buat kemitraan antara bagian keuangan dan teknologi](#)
- [COST01-BP03 Tetapkan prakiraan dan anggaran cloud](#)
- [COST01-BP04 Mengimplementasikan kesadaran biaya dalam proses organisasi Anda](#)
- [COST01-BP05 Melaporkan dan memberi tahu tentang pengoptimalan biaya](#)
- [COST01-BP06 Memantau biaya secara proaktif](#)
- [COST01-BP07 Mengikuti perkembangan rilis layanan baru](#)
- [COST01-BP08 Menciptakan budaya sadar biaya](#)
- [COST01-BP09 Menghitung nilai bisnis dari optimasi biaya](#)

## COST01-BP01 Membangun kepemilikan pengoptimalan biaya

Buat tim (tim Cloud Business Office, Cloud Center of Excellence, atau FinOps) yang bertanggung jawab membangun dan memelihara budaya sadar biaya (cost awareness) di seluruh organisasi Anda. Pemilik optimasi biaya dapat berupa individu atau tim (membutuhkan orang-orang dari tim keuangan, teknologi, dan bisnis) yang memahami seluruh organisasi dan keuangan cloud.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Ini adalah pengantar fungsi atau tim Cloud Business Office (CBO) atau Cloud Center of Excellence (CCOE) yang bertanggung jawab membangun dan memelihara budaya sadar biaya (cost awareness) di komputasi cloud. Fungsi ini dapat terdiri dari individu yang sudah ada, tim di dalam organisasi Anda, atau tim baru yang terdiri dari pemangku kepentingan utama organisasi, keuangan, dan teknologi dari seluruh organisasi.

Fungsi ini (individu atau tim) memprioritaskan dan menggunakan sebagian besar waktunya untuk aktivitas manajemen dan pengoptimalan biaya. Untuk organisasi kecil, fungsi ini mungkin memerlukan persentase waktu yang lebih sedikit dibandingkan dengan fungsi purnawaktu di korporasi yang lebih besar.

Fungsi ini (individu atau tim) memprioritaskan dan menggunakan sebagian besar waktunya untuk aktivitas manajemen dan pengoptimalan biaya. Untuk organisasi kecil, fungsi ini mungkin memerlukan persentase waktu yang lebih sedikit untuk aktivitas manajemen dan optimasi biaya dibandingkan dengan fungsi purnawaktu di korporasi yang lebih besar.

Fungsi ini memerlukan pendekatan multidisiplin, dengan kemampuan di bidang manajemen proyek, ilmu data, analisis keuangan, dan pengembangan perangkat lunak atau infrastruktur. Fungsi ini dapat meningkatkan efisiensi beban kerja dengan menjalankan pengoptimalan biaya dalam tiga kepemilikan yang berbeda:

- **Tersentralisasi:** Melalui tim yang ditunjuk seperti tim FinOps, tim Cloud Financial Management (CFM), Cloud Business Office (CBO), atau Cloud Center of Excellence (CCoE), pelanggan dapat merancang dan menerapkan mekanisme tata kelola dan mendorong praktik terbaik di seluruh perusahaan.
- **Terdesentralisasi:** Memengaruhi tim-tim teknologi untuk menjalankan optimasi biaya.
- **Hybrid:** Gabungan dari tim tersentralisasi dan terdesentralisasi dapat bekerja sama untuk menjalankan optimasi biaya.

Fungsi ini dapat diukur dari kemampuannya menjalankan dan menyampaikan tujuan pengoptimalan biaya (misalnya, metrik efisiensi beban kerja).

Anda harus mendapatkan sponsor eksekutif untuk fungsi ini, yang merupakan faktor keberhasilan utama. Sponsor ini dianggap sebagai penyokong penggunaan cloud hemat biaya, serta memberikan dukungan peningkatan kepada tim guna memastikan bahwa aktivitas pengoptimalan biaya ditangani menurut tingkat prioritas yang ditentukan oleh organisasi. Jika tidak, panduan dapat diabaikan dan peluang penghematan biaya tidak akan diprioritaskan. Bersama-sama, sponsor dan tim membantu organisasi Anda menggunakan cloud secara efisien dan menghadirkan nilai bisnis.

Jika Anda memiliki [paket dukungan](#) Business, Enterprise-On-Ramp, atau Enterprise dan memerlukan bantuan dalam membangun tim atau fungsi ini, hubungi ahli Cloud Financial Management (CFM) Anda melalui tim akun Anda.

## Langkah implementasi

- Tentukan anggota utama: Semua bagian yang relevan dari organisasi Anda harus berkontribusi dan berminat pada manajemen biaya. Tim umum di dalam organisasi biasanya berisi: tim keuangan, pemilik produk atau aplikasi, manajemen, dan teknis (DevOps). Beberapa dari mereka terlibat secara purnawaktu (keuangan atau teknis), sementara yang lain dilibatkan secara berkala sesuai kebutuhan. Individu atau tim yang melakukan CFM memerlukan rangkaian keahlian berikut:
  - Pengembangan perangkat lunak: apabila ada skrip dan otomatisasi yang sedang dibuat.
  - Rekayasa infrastruktur: untuk men-deploy skrip, mengotomatisasi proses, dan memahami bagaimana layanan atau sumber daya disediakan.
  - Kecakapan operasi: CFM adalah tentang beroperasi di cloud secara efisien dengan mengukur, memantau, memodifikasi, merencanakan, dan menskalakan penggunaan cloud secara efisien.
- Tentukan tujuan dan metrik: Fungsi harus memberikan nilai kepada organisasi dengan cara yang berbeda. Tujuan-tujuan tersebut ditetapkan dan terus berkembang seiring dengan perkembangan organisasi. Aktivitas umum mencakup: membuat dan menjalankan program edukasi tentang pengoptimalan biaya di seluruh organisasi, mengembangkan standar di seluruh organisasi, seperti pemantauan dan pelaporan untuk pengoptimalan biaya, dan menetapkan sasaran beban kerja dalam pengoptimalan. Fungsi juga harus melaporkan kemampuan pengoptimalan biaya mereka secara rutin kepada organisasi.

Anda dapat menentukan indikator kinerja utama (KPI) berbasis nilai atau biaya. Saat Anda menentukan KPI, Anda dapat menghitung biaya yang diharapkan sehubungan dengan efisiensi dan hasil bisnis yang diharapkan. KPI berbasis nilai mengaitkan metrik biaya dan penggunaan dengan pendorong nilai bisnis dan membantu merasionalisasi perubahan dalam pengeluaran

AWS. Langkah pertama untuk menyimpulkan KPI berbasis nilai adalah bekerja sama, lintas organisasi, untuk memilih dan menyepakati serangkaian KPI standar.

- Adakan koordinasi rutin: Grup (keuangan, teknologi, dan tim bisnis) harus melakukan rapat secara teratur untuk meninjau tujuan dan metrik mereka. Koordinasi ini biasanya membahas peninjauan status organisasi, program apa pun yang sedang berlangsung, serta metrik pengoptimalan dan keuangan secara keseluruhan. Selanjutnya, beban kerja utama dilaporkan dengan lebih mendetail.

Selama tinjauan rutin ini, Anda dapat meninjau efisiensi beban kerja (biaya) dan hasil bisnis. Misalnya, kenaikan biaya 20% untuk beban kerja dapat diselaraskan dengan peningkatan penggunaan pelanggan. Dalam kasus ini, kenaikan biaya 20% dapat diartikan sebagai investasi. Rapat koordinasi rutin ini dapat membantu tim mengidentifikasi KPI nilai yang memberikan makna bagi seluruh organisasi.

## Sumber daya

Dokumen terkait:

- [Blog CCOE AWS](#)
- [Membuat Kantor Bisnis Cloud](#)
- [CCOE - Cloud Center of Excellence](#)

Video terkait:

- [Kisah Sukses CCOE Vanguard](#)

Contoh terkait:

- [Menggunakan Cloud Center of Excellence \(CCOE\) untuk Mentransformasi Keseluruhan Perusahaan](#)
- [Membangun CCOE untuk mentransformasi keseluruhan perusahaan](#)
- [7 Perangkat yang Perlu Dihindari Saat Membangun CCOE](#)



## COST01-BP02 Buat kemitraan antara bagian keuangan dan teknologi

Libatkan tim keuangan dan teknologi dalam diskusi biaya dan penggunaan di semua tahap perjalanan cloud Anda. Tim secara teratur bertemu dan membahas topik-topik seperti target dan tujuan organisasi, kondisi biaya dan penggunaan saat ini, serta praktik keuangan dan akuntansi.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Tinggi

### Panduan implementasi

Tim teknologi berinovasi lebih cepat di cloud karena siklus deployment infrastruktur, pengadaan, dan persetujuan yang lebih pendek. Ini dapat membutuhkan penyesuaian untuk organisasi keuangan yang sebelumnya terbiasa menjalankan proses yang memerlukan waktu lama dan banyak sumber daya untuk mendapatkan dan melakukan deployment modal di lingkungan on-premise dan pusat data, dan alokasi biaya hanya berdasarkan persetujuan proyek.

Dari perspektif organisasi keuangan dan pengadaan, proses penganggaran modal, permintaan modal, persetujuan, pengadaan, dan pemasangan infrastruktur fisik adalah proses yang telah dipelajari dan distandardisasi selama beberapa dekade:

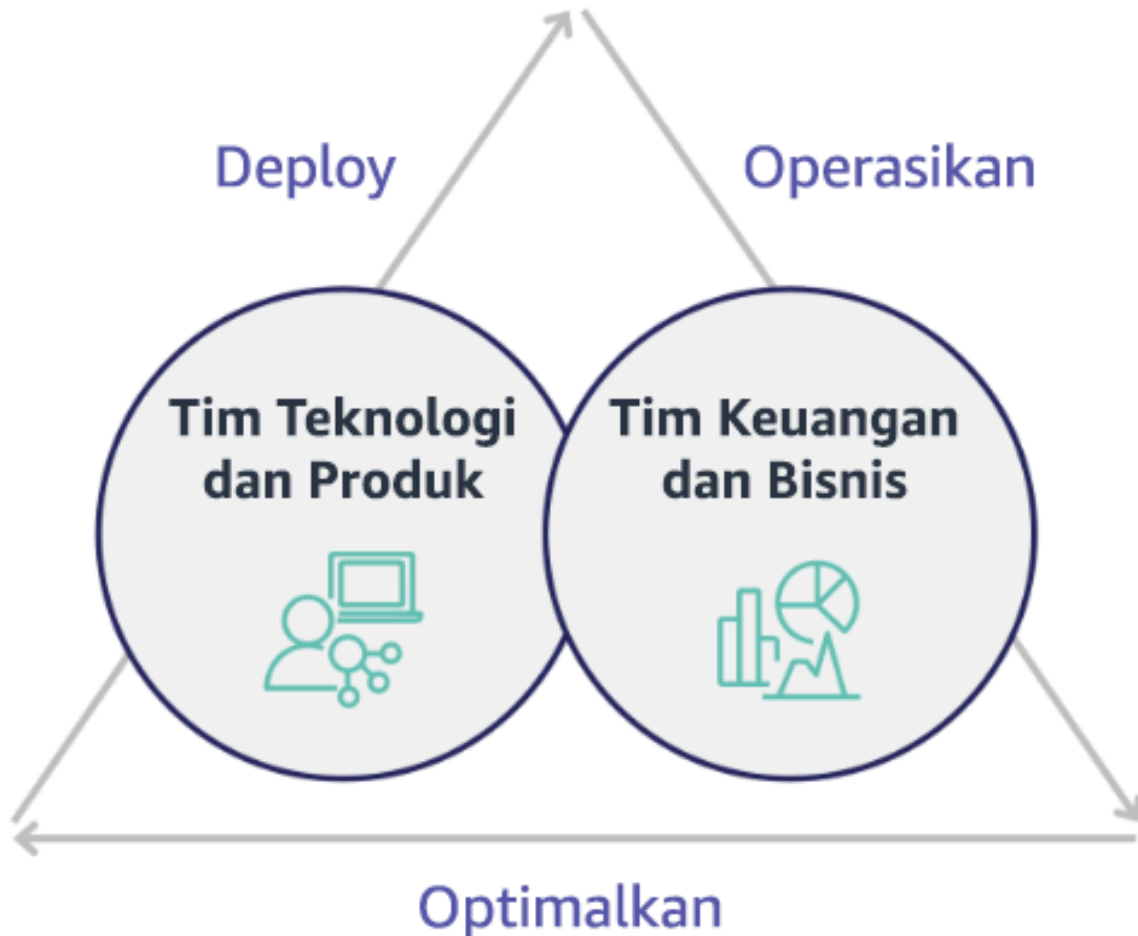
- Tim rekayasa atau IT biasanya adalah pemohon
- Berbagai tim keuangan bertindak sebagai pemberi persetujuan dan pengadaan
- Tim operasi mengumpulkan, menumpuk, dan menyerahkan infrastruktur siap pakai



Dengan penerapan cloud, pengadaan dan penggunaan infrastruktur tidak lagi terikat pada rantai dependensi. Dalam model cloud, tim teknologi dan produk tidak lagi hanya builder, tetapi operator dan pemilik produk mereka, bertanggung jawab atas sebagian besar aktivitas yang secara historis terkait dengan tim keuangan dan operasi, termasuk pengadaan dan deployment.

Hal yang diperlukan untuk menyediakan sumber daya cloud adalah akun pengguna, dan serangkaian izin yang tepat. Hal ini juga yang mengurangi risiko IT dan keuangan; yang berarti tim hanya perlu melakukan beberapa klik atau panggilan API untuk menghentikan sumber daya cloud yang tidak digunakan atau tidak diperlukan. Hal ini juga yang memungkinkan tim teknologi berinovasi lebih cepat – ketangkasan dan kemampuan untuk menjalankan dan kemudian menghentikan eksperimen. Sementara sifat variabel penggunaan cloud dapat memengaruhi prediktibilitas dari perspektif

penganggaran dan prakiraan modal, cloud memberi organisasi kemampuan untuk mengurangi biaya penyediaan yang berlebihan, serta mengurangi biaya peluang yang terkait dengan kekurangan penyediaan yang konservatif.



Buat kemitraan antara pemangku kepentingan penting dalam bidang keuangan dan teknologi untuk menghasilkan pemahaman bersama akan tujuan organisasi dan kembangkan mekanisme agar berhasil secara finansial dalam model pengeluaran variabel komputasi cloud. Tim yang relevan dalam organisasi Anda harus dilibatkan dalam diskusi biaya dan penggunaan di semua tahap perjalanan cloud Anda, termasuk:

- Kepala bagian keuangan: CFO, pengontrol keuangan, perencana keuangan, analis bisnis, bagian pengadaan, pengelolaan pemasok, dan utang usaha harus memahami model cloud dari proses penagihan bulanan, opsi pembelian, dan konsumsi. Tim keuangan perlu bermitra dengan tim teknologi untuk membuat dan menyosialisasikan kisah nilai IT, sehingga membantu tim bisnis memahami bagaimana pengeluaran teknologi terkait dengan hasil bisnis. Dengan cara ini,

pengeluaran teknologi tidak dipandang sebagai biaya, melainkan sebagai investasi. Karena perbedaan mendasar antara cloud (seperti laju perubahan dalam penggunaan, harga bayar sesuai pemakaian, harga berjenjang, model harga, dan informasi penggunaan serta penagihan mendetail) dibandingkan dengan operasi on-premise, maka penting organisasi keuangan memahami bagaimana penggunaan cloud dapat memengaruhi aspek bisnis, termasuk proses pengadaan, pelacakan insentif, alokasi biaya, dan laporan keuangan.

- Kepala bagian teknologi: Kepala bagian teknologi (termasuk pemilik aplikasi dan produk) harus sadar akan persyaratan keuangan (misalnya, batas anggaran) serta persyaratan bisnis (misalnya, perjanjian tingkat layanan). Ini memungkinkan beban kerja diimplementasikan untuk mencapai tujuan yang diinginkan organisasi.

Kemitraan antara bagian keuangan dan teknologi memberikan manfaat berikut:

- Tim keuangan dan teknologi memiliki visibilitas hampir dalam waktu nyata ke biaya dan penggunaan.
- Tim keuangan dan teknologi menetapkan prosedur pengoperasian standar untuk menangani varian pengeluaran cloud.
- Pemangku kepentingan dalam bidang keuangan berfungsi sebagai penasihat strategis dalam kaitannya dengan bagaimana modal digunakan untuk membeli diskon komitmen (misalnya, Instans Terpesan atau AWS Savings Plans), dan bagaimana cloud digunakan untuk mengembangkan organisasi.
- Proses pengadaan dan utang usaha yang ada digunakan dengan cloud.
- Tim keuangan dan teknologi berkolaborasi dalam memprediksi AWS biaya dan penggunaan di waktu mendatang untuk menyelaraskan dan menyusun anggaran organisasi.
- Komunikasi lintas organisasi yang lebih baik melalui bahasa bersama, dan pemahaman bersama akan konsep keuangan.

Pemangku kepentingan tambahan di dalam organisasi Anda yang harus dilibatkan dalam diskusi biaya dan penggunaan termasuk:

- Pemilik unit bisnis: Pemilik unit bisnis harus memahami model bisnis cloud sehingga mereka dapat memberikan pengarahan kepada unit bisnis dan seluruh perusahaan. Pengetahuan cloud ini sangat penting ketika ada kebutuhan untuk memprediksi pertumbuhan dan penggunaan beban kerja, dan ketika mengevaluasi opsi pembelian jangka lebih panjang, seperti Instans Terpesan atau Savings Plans.

- Tim rekayasa: Membangun kemitraan antara tim keuangan dan teknologi sangat penting untuk membangun budaya sadar biaya yang mendorong para rekayasawan untuk mengambil tindakan di Manajemen Finansial Cloud (CFM). Salah satu masalah umum dari CFM atau praktisi operasi keuangan dan tim keuangan adalah membuat para rekayasawan memahami seluruh bisnis di cloud, mengikuti praktik terbaik, dan mengambil tindakan yang direkomendasikan.
- Pihak ketiga: Jika organisasi Anda menggunakan pihak ketiga (misalnya, konsultan atau alat), pastikan mereka selaras dengan tujuan keuangan Anda, dan mereka dapat menunjukkan keselarasan tersebut melalui model keterlibatan dan laba atas investasi (ROI) mereka. Umumnya, pihak ketiga akan berkontribusi dalam pelaporan dan analisis beban kerja apa pun yang mereka kelola, dan mereka akan memberikan analisis biaya beban kerja apa pun yang mereka desain.

Menerapkan CFM dan mencapai kesuksesan memerlukan kolaborasi di seluruh tim keuangan, teknologi, dan bisnis, serta perubahan dalam cara pengeluaran cloud dikomunikasikan dan dievaluasi di seluruh organisasi. Ikut sertakan tim rekayasa sehingga mereka dapat menjadi bagian dari diskusi biaya dan penggunaan ini di semua tahap, dan dorong mereka untuk mengikuti praktik terbaik dan mengambil tindakan yang disepakati.

### Langkah implementasi

- Tentukan anggota utama: Verifikasi bahwa semua anggota yang relevan dari tim keuangan dan teknologi Anda ambil bagian dalam kemitraan. Anggota yang relevan dari tim keuangan adalah mereka yang memiliki interaksi dengan tagihan cloud. Umumnya ini adalah CFO, pengontrol keuangan, perencana keuangan, analis bisnis, bagian pengadaan, dan pengelolaan pemasok. Umumnya anggota bagian teknologi adalah pemilik aplikasi dan produk, manajer teknis, dan perwakilan dari semua tim yang membangun di cloud. Anggota-anggota lainnya dapat mencakup pemilik unit bisnis, misalnya bagian pemasaran, yang akan memengaruhi penggunaan produk, dan pihak ketiga seperti konsultan, untuk mencapai keselarasan dengan tujuan dan mekanisme Anda, dan untuk membantu dalam pelaporan.
- Tentukan topik untuk diskusi: Tentukan topik yang bersifat umum lintas tim, atau yang akan membutuhkan pemahaman bersama. Ikuti biaya dari waktu biaya dibuat, sampai tagihan dibayar. Catat setiap anggota yang terlibat, dan proses organisasi yang harus diterapkan. Pahami setiap langkah atau proses yang dilewatinya dan informasi terkait, seperti model harga yang tersedia, harga berjenjang, model diskon, penetapan anggaran, dan persyaratan keuangan.
- Adakan koordinasi rutin: Untuk menciptakan kemitraan keuangan dan teknologi, buatlah koordinasi komunikasi yang teratur untuk menciptakan dan mempertahankan keselarasan. Grup tersebut harus mengadakan pertemuan koordinasi secara rutin untuk membahas tujuan dan metrik.

Koordinasi ini biasanya membahas peninjauan status organisasi, program apa pun yang sedang berlangsung, serta metrik pengoptimalan dan keuangan secara keseluruhan. Kemudian beban kerja utama dilaporkan dengan lebih mendetail.

## Sumber daya

Dokumen terkait:

- [Blog Berita AWS](#)

## COST01-BP03 Tetapkan prakiraan dan anggaran cloud

Sesuaikan proses pemrakiraan dan penganggaran yang ada di organisasi agar kompatibel dengan biaya dan penggunaan cloud yang sangat bervariasi. Prosesnya harus dinamis menggunakan algoritma berbasis pendorong bisnis atau berbasis tren, atau kombinasi dari keduanya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

## Panduan implementasi

Pelanggan menggunakan cloud untuk efisiensi, kecepatan, dan ketangkasan, yang menimbulkan biaya dan penggunaan yang sangat bervariasi. Biaya dapat berkurang (atau terkadang bertambah) seiring dengan peningkatan efisiensi beban kerja, atau seiring dengan deployment fitur dan beban kerja baru. Beban kerja dapat diskalakan untuk melayani lebih banyak pelanggan, yang akan meningkatkan biaya dan penggunaan cloud Anda. Sumber daya sekarang lebih mudah diakses daripada sebelumnya. Elastisitas cloud juga menghadirkan elastisitas biaya dan prakiraan. Proses penganggaran organisasi yang sudah ada harus dimodifikasi untuk menyertakan variabilitas ini.

Anggaran biasanya disiapkan untuk satu tahun dan tidak berubah-ubah, sehingga membutuhkan kepatuhan yang ketat dari semua orang yang terlibat. Sebaliknya, pemrakiraan lebih fleksibel, sehingga memungkinkan penyesuaian ulang sepanjang tahun dan memberikan proyeksi dinamis selama periode satu, dua, atau tiga tahun. Baik penganggaran maupun pemrakiraan memainkan peran penting dalam membangun ekspektasi keuangan di kalangan berbagai pemangku kepentingan teknologi dan bisnis. Pemrakiraan dan implementasi yang akurat juga menghadirkan akuntabilitas bagi para pemangku kepentingan yang bertanggung jawab langsung atas penyediaan biaya sejak awal, serta dapat meningkatkan kesadaran biaya mereka secara keseluruhan.

Sesuaikan proses pemrakiraan dan penganggaran yang ada agar menjadi lebih dinamis menggunakan algoritma berbasis tren (menggunakan biaya historis sebagai masukan), atau menggunakan algoritma berbasis pendorong bisnis (misalnya, peluncuran produk baru atau ekspansi Regional), yang ideal untuk lingkungan penganggaran dinamis dan bervariasi, atau kombinasi antara pendorong bisnis dan tren.

Anda dapat menggunakan [AWS Cost Explorer](#) untuk pemrakiraan berbasis tren dalam rentang waktu mendatang yang ditentukan berdasarkan pengeluaran Anda sebelumnya. Mesin prakiraan AWS Cost Explorer menyegmentasikan data historis Anda berdasarkan jenis tagihan (misalnya, Instans Terpesan) dan menggunakan kombinasi machine learning dan model berbasis aturan untuk memprediksi pengeluaran di semua jenis tagihan satu per satu.

Identifikasi pendorong bisnis yang dapat berdampak pada biaya penggunaan Anda dan lakukan prakiraan untuk masing-masing secara terpisah untuk memastikan bahwa perkiraan penggunaan sudah diperhitungkan di awal. Beberapa pendorong terkait dengan tim IT dan produk di dalam organisasi. Pendorong bisnis lainnya, seperti acara pemasaran, promosi, merger, dan akuisisi, dikenal oleh pemimpin penjualan, pemasaran, dan bisnis Anda, dan penting juga untuk berkolaborasi dan memperhitungkan semua pendorong permintaan tersebut. Anda perlu bekerja sama dengan mereka untuk memahami dampaknya terhadap pendorong internal baru.

Setelah Anda menentukan prakiraan berbasis tren menggunakan Cost Explorer atau alat lainnya, gunakan [AWS Pricing Calculator](#) untuk memperkirakan kasus penggunaan AWS dan biaya mendatang berdasarkan pada penggunaan yang diharapkan (lalu lintas, permintaan per detik, atau instans Amazon EC2 yang diperlukan). Anda juga dapat menggunakannya untuk merencanakan pengeluaran, menemukan peluang penghematan biaya, dan membuat keputusan yang tepat saat menggunakan AWS. Penting untuk melacak keakuratan prakiraan tersebut karena anggaran harus ditetapkan berdasarkan perhitungan dan estimasi prakiraan ini.

Gunakan [AWS Budgets](#) untuk menetapkan anggaran khusus pada tingkat mendetail dengan menentukan periode waktu, pengulangan, atau jumlah (tetap atau variabel), dan menambahkan filter seperti layanan, Wilayah AWS, dan tag. Untuk tetap mendapatkan informasi tentang performa anggaran yang ada, Anda dapat membuat dan menjadwalkan [Laporan AWS Budgets](#) agar dikirim melalui email kepada Anda dan pemangku kepentingan secara teratur. Anda juga dapat membuat [Peringatan AWS Budgets](#) berdasarkan biaya aktual, yang bersifat reaktif, atau biaya yang diprakirakan, yang menyediakan waktu untuk menerapkan mitigasi terhadap potensi kelebihan biaya. Anda dapat menerima peringatan saat biaya atau penggunaan melampaui, atau jika diprakirakan akan melampaui, jumlah yang dianggarkan.

Gunakan [AWS Cost Anomaly Detection](#) untuk mencegah atau mengurangi biaya tak terduga dan meningkatkan kontrol tanpa memperlambat inovasi. AWS Cost Anomaly Detection memanfaatkan machine learning untuk mengidentifikasi pengeluaran yang tidak wajar dan akar penyebab, agar Anda dapat mengambil tindakan secara cepat. [Dengan tiga langkah sederhana](#), Anda dapat membuat pemantauan kontekstual Anda sendiri dan menerima peringatan saat pengeluaran yang tidak wajar terdeteksi.

Sebagaimana disebutkan dalam pilar optimasi biaya Well-Architected [bagian Kemitraan Keuangan dan Teknologi](#), penting untuk memiliki kemitraan dan koordinasi antara IT, keuangan, dan pemangku kepentingan lainnya untuk memverifikasi bahwa mereka semua menggunakan alat atau proses yang sama untuk konsistensi. Dalam kasus yang mungkin memerlukan perubahan anggaran, menambah frekuensi rapat singkat koordinasi dapat membantu bereaksi terhadap perubahan tersebut secara lebih cepat.

## Langkah implementasi

- Analisis prakiraan berbasis tren: Gunakan alat prakiraan berbasis tren pilihan seperti AWS Cost Explorer dan Amazon Forecast. Analisis biaya penggunaan Anda pada dimensi-dimensi yang berbeda seperti kategori layanan, akun, tag, dan biaya. Jika prakiraan lanjutan diperlukan, impor data AWS Cost and Usage Report Anda ke Amazon Forecast (yang menerapkan regresi linier sebagai bentuk machine learning ke prakiraan).
- Analisis prakiraan berbasis pendorong: Identifikasi dampak pendorong bisnis terhadap penggunaan cloud Anda dan lakukan prakiraan pada masing-masing secara terpisah untuk menghitung perkiraan biaya penggunaan di awal. Bekerjasamalah dengan pemilik unit bisnis dan pemangku kepentingan untuk memahami dampak pada pendorong baru dan menghitung perkiraan perubahan biaya untuk menentukan anggaran yang akurat.
- Perbarui proses pemrakiraan dan penganggaran yang ada: Tetapkan proses penganggaran prakiraan Anda berdasarkan metode prakiraan yang diadopsi seperti metode berbasis tren, berbasis pendorong bisnis, atau kombinasi keduanya. Anggaran harus terhitung dan realistis, berdasarkan proses pemrakiraan ini.
- Konfigurasi peringatan dan notifikasi: Gunakan Pemberitahuan AWS Budgets dan AWS Cost Anomaly Detection untuk mendapatkan pemberitahuan dan notifikasi.
- Lakukan peninjauan rutin dengan pemangku kepentingan utama: Misalnya, pemangku kepentingan di tim IT, keuangan, platform, dan bidang bisnis lainnya, harus menyelaraskan dengan perubahan dalam arah dan penggunaan bisnis.



## Sumber daya

Dokumen terkait:

- [AWS Cost Explorer](#)
- [AWS Cost and Usage Report](#)
- [Prakiraan Amazon QuickSight](#)
- [Amazon Forecast](#)
- [AWS Budgets](#)
- [Blog Berita AWS](#)

Video terkait:

- [Cara Menggunakan AWS Budgets untuk melacak pengeluaran dan penggunaan](#)
- [Serial Optimasi Biaya AWS: AWS Budgets](#)

Contoh terkait:

- [Memahami dan membangun prakiraan berbasis pendorong](#)
- [Cara membangun dan mendorong budaya prakiraan](#)
- [Cara meningkatkan prakiraan biaya cloud Anda](#)
- [Menggunakan alat yang tepat untuk prakiraan biaya cloud Anda](#)

## COST01-BP04 Mengimplementasikan kesadaran biaya dalam proses organisasi Anda

Implementasikan kesadaran biaya, transparansi, dan akuntabilitas biaya ke dalam proses baru atau yang sudah ada yang memengaruhi penggunaan, serta manfaatkan proses yang sudah ada untuk kesadaran biaya. Implementasikan kesadaran biaya ke dalam pelatihan karyawan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Tinggi

## Panduan implementasi

Kesadaran biaya harus diimplementasikan di dalam proses-proses baru dan yang sudah ada. Ini adalah salah satu kemampuan dasar dan prasyarat untuk praktik terbaik lainnya. Disarankan untuk menggunakan ulang dan memodifikasi proses yang sudah ada jika memungkinkan—hal ini meminimalkan dampak terhadap ketangkasan dan kecepatan. Laporkan biaya cloud kepada tim teknologi dan pengambil keputusan di tim bisnis dan keuangan untuk meningkatkan kesadaran biaya, dan tetapkan indikator kinerja utama (KPI) efisiensi untuk pemangku kepentingan keuangan dan bisnis. Saran berikut ini akan membantu mengimplementasikan kesadaran biaya di dalam beban kerja Anda:

- Pastikan manajemen perubahan mencakup pengukuran biaya untuk menghitung dampak keuangan dari perubahan Anda. Hal ini membantu Anda secara proaktif menangani masalah terkait biaya dan menyoroti penghematan biaya.
- Pastikan pengoptimalan biaya adalah komponen inti kemampuan operasional Anda. Misalnya, Anda dapat memanfaatkan proses manajemen insiden yang sudah ada untuk menyelidiki dan mengidentifikasi akar masalah untuk anomali biaya dan penggunaan atau kelebihan biaya.
- Percepat penghematan biaya dan realisasi nilai bisnis melalui otomatisasi atau penggunaan alat. Saat membahas biaya implementasi, atur percakapan agar turut membahas komponen laba atas investasi (ROI) guna mencari pembenaran atas investasi waktu atau uang.
- Alokasikan biaya cloud dengan menerapkan showback atau chargeback untuk pengeluaran cloud, termasuk pengeluaran untuk opsi pembelian berbasis komitmen, layanan bersama, dan pembelian marketplace untuk mendukung penggunaan cloud yang paling sadar biaya.
- Sertakan pelatihan tentang kesadaran biaya dalam program pelatihan dan pengembangan yang sudah ada di seluruh organisasi Anda. Disarankan untuk menyertakan pelatihan dan sertifikasi berkelanjutan di dalamnya. Hal ini akan menghasilkan sebuah organisasi yang mampu mengelola biaya dan penggunaan secara mandiri.
- Manfaatkan alat native AWS gratis seperti [AWS Cost Anomaly Detection](#), [AWS Budgets](#), dan [Laporan AWS Budgets](#).

Saat organisasi secara konsisten menerapkan praktik [Manajemen Finansial Cloud](#) (CFM), perilaku tersebut tertanam dalam cara kerja dan pengambilan keputusan. Hasilnya adalah budaya yang lebih sadar biaya, mulai dari pengembang yang merancang aplikasi yang baru di cloud, hingga manajer keuangan yang menganalisis ROI pada investasi cloud baru ini.

### Langkah implementasi

- Identifikasi proses organisasi yang relevan: Setiap unit organisasi meninjau proses mereka dan mengidentifikasi proses yang berdampak pada biaya dan penggunaan. Proses apa pun yang memunculkan pembuatan atau penghentian sumber daya perlu turut ditinjau. Cari proses yang dapat mendukung kesadaran biaya di dalam bisnis Anda, seperti manajemen dan pelatihan insiden.
- Membangun budaya swadaya dan sadar biaya: Pastikan semua pemangku kepentingan yang relevan menyelaraskan dengan penyebab perubahan dan dampak sebagai biaya sehingga mereka memahami biaya cloud. Ini akan memungkinkan organisasi Anda membangun budaya inovasi swadaya dan sadar biaya.
- Perbarui proses dengan kesadaran biaya: Setiap proses dimodifikasi agar menjadi proses yang sadar biaya. Proses mungkin memerlukan pemeriksaan awal tambahan, seperti penilaian dampak biaya, atau pemeriksaan akhir yang memvalidasi terjadinya perubahan yang diharapkan dalam hal biaya dan penggunaan. Proses pendukung seperti pelatihan dan manajemen insiden dapat dibuat agar menyertakan item-item untuk biaya dan penggunaan.

Untuk mendapatkan bantuan, hubungi pakar CFM melalui tim Akun Anda, atau jelajahi sumber daya dan dokumen terkait di bawah.

## Sumber daya

Dokumen terkait:

- [Manajemen Finansial Cloud AWS](#)

Contoh terkait:

- [Strategi untuk Manajemen Biaya Cloud Efisien](#)
- [Seri Blog Pengendalian Biaya No. 3: Cara Menangani Biaya Tak Terduga](#)
- [Panduan Pemula untuk AWS Cost Management](#)

## COST01-BP05 Melaporkan dan memberi tahu tentang pengoptimalan biaya

Siapkan anggaran cloud dan konfigurasi mekanisme untuk mendeteksi anomali dalam penggunaan. Konfigurasi alat-alat terkait untuk peringatan biaya dan penggunaan sesuai target

yang telah ditentukan sebelumnya dan terima notifikasi ketika terdapat penggunaan yang melebihi target tersebut. Lakukan pertemuan rutin untuk menganalisis efektivitas biaya beban kerja Anda dan meningkatkan kesadaran biaya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

## Panduan implementasi

Anda harus secara rutin melaporkan pengoptimalan biaya dan penggunaan di dalam organisasi Anda. Anda dapat mengimplementasikan sesi-sesi khusus untuk membahas kinerja biaya, atau sertakan pengoptimalan biaya di dalam siklus pelaporan operasional rutin untuk beban kerja Anda. Gunakan layanan dan alat untuk memantau kinerja biaya Anda secara teratur dan menerapkan peluang penghematan biaya.

Lihat biaya dan penggunaan Anda dengan beberapa filter dan perincian dengan menggunakan [AWS Cost Explorer](#), yang menyediakan dasbor dan laporan seperti biaya berdasarkan layanan atau akun, biaya harian, atau biaya pasar. Lacak progres biaya dan penggunaan Anda berdasarkan anggaran yang dikonfigurasi dengan [Laporan AWS Budgets](#).

Gunakan [AWS Budgets](#) untuk mengatur anggaran khusus dalam melacak biaya dan penggunaan Anda serta menanggapi dengan cepat pemberitahuan yang diterima dari email atau notifikasi Amazon Simple Notification Service (Amazon SNS) jika Anda melebihi ambang batas Anda. [Tetapkan periode anggaran sesuai keinginan Anda](#), yaitu harian, bulanan, triwulanan, atau tahunan, dan buat batas anggaran khusus agar terus mendapatkan informasi tentang progres biaya dan penggunaan aktual atau yang diperkirakan berdasarkan ambang batas anggaran Anda. Anda juga dapat mengatur [peringatan](#) dan [tindakan](#) terhadap pemberitahuan tersebut agar berjalan secara otomatis atau melalui proses persetujuan saat target anggaran terlampaui.

Implementasikan notifikasi biaya dan penggunaan untuk memastikan perubahan biaya dan penggunaan dapat ditindaklanjuti secara cepat jika tidak terduga. [AWS Cost Anomaly Detection](#) memungkinkan Anda mengurangi biaya tak terduga dan meningkatkan kontrol tanpa memperlambat inovasi. AWS Cost Anomaly Detection mengidentifikasi pengeluaran yang tidak wajar dan akar masalah, yang membantu mengurangi risiko penagihan tak terduga. Dengan tiga langkah mudah, Anda dapat membuat pemantauan kontekstual Anda sendiri dan menerima peringatan saat pengeluaran yang tidak wajar terdeteksi.

Anda juga dapat menggunakan [Amazon QuickSight](#) dengan data AWS Cost and Usage Report (CUR), untuk memberikan pelaporan yang sangat disesuaikan dengan data yang lebih mendetail. Amazon QuickSight memungkinkan Anda menjadwalkan laporan dan menerima email Laporan Biaya

berkala untuk biaya dan penggunaan historis, atau peluang penghematan biaya. Coba solusi [Dasbor Inteligensi Biaya](#) (CID) kami yang dibangun berdasarkan Amazon QuickSight, yang memberi Anda visibilitas lanjutan.

Gunakan [AWS Trusted Advisor](#), yang memberikan panduan untuk memverifikasi apakah sumber daya yang disediakan selaras dengan praktik terbaik AWS untuk pengoptimalan biaya.

Lihat rekomendasi Savings Plans Anda melalui grafik visual berdasarkan biaya dan penggunaan mendetail Anda. Grafik per jam menunjukkan pengeluaran Sesuai Permintaan beserta komitmen Savings Plans yang disarankan, sehingga memberikan wawasan tentang perkiraan penghematan, cakupan Savings Plans, dan pemanfaatan Savings Plans. Ini membantu organisasi untuk memahami bagaimana Savings Plans mereka berlaku untuk setiap jam pengeluaran tanpa harus menginvestasikan waktu dan sumber daya untuk membangun model-model analisis pengeluaran.

Buat laporan secara berkala yang berisi sorotan Savings Plans, Instans Terpesan, dan rekomendasi penyesuaian ukuran Amazon EC2 dari AWS Cost Explorer untuk mulai mengurangi biaya yang terkait dengan beban kerja berkondisi stabil serta sumber daya yang tidak aktif dan kurang dimanfaatkan. Identifikasi dan dapatkan kembali pengeluaran yang terkait dengan pemborosan cloud untuk sumber daya yang di-deploy. Pemborosan cloud terjadi saat sumber daya berukuran tidak tepat dibuat atau ditemukan pola penggunaan yang berbeda dari yang diharapkan. Ikuti praktik terbaik AWS untuk mengurangi pemborosan Anda atau minta tim akun dan partner Anda untuk membantu Anda [mengoptimalkan serta menghemat](#) biaya cloud Anda.

Buat laporan secara teratur untuk opsi pembelian yang lebih baik bagi sumber daya Anda guna menurunkan biaya unit untuk beban kerja Anda. Opsi pembelian seperti Savings Plans, Instans Terpesan, atau Instans Spot Amazon EC2 menawarkan penghematan biaya terbesar untuk beban kerja yang toleran terhadap kesalahan dan memungkinkan pemangku kepentingan (pemilik bisnis, tim keuangan, dan tim teknologi) untuk menjadi bagian dari diskusi komitmen ini.

Bagikan laporan yang berisi peluang atau pengumuman rilis baru yang dapat membantu Anda mengurangi total biaya kepemilikan (TCO) cloud. Terapkan layanan baru, Wilayah, fitur, solusi, atau cara baru untuk mencapai pengurangan biaya lebih lanjut.

## Langkah implementasi

- Mengonfigurasi AWS Budgets: Konfigurasi AWS Budgets di semua akun untuk beban kerja Anda. Tetapkan anggaran untuk keseluruhan pengeluaran akun, serta anggaran untuk beban kerja menggunakan tag.
- [Lab Well-Architected: Biaya dan Penggunaan Tata Kelola](#)

- Melaporkan pengoptimalan biaya: Siapkan siklus rutin untuk membahas dan menganalisis efisiensi beban kerja. Menggunakan metrik yang telah dibangun, buat laporan tentang metrik-metrik yang dicapai serta biaya untuk mencapainya. Identifikasi dan perbaiki tren negatif apa pun, serta tren positif yang dapat Anda pupuk di seluruh organisasi Anda. Pelaporan harus melibatkan perwakilan dari tim dan pemilik aplikasi, keuangan, dan pengambil keputusan utama sehubungan dengan pengeluaran cloud.

## Sumber daya

Dokumen terkait:

- [AWS Cost Explorer](#)
- [AWS Trusted Advisor](#)
- [AWS Budgets](#)
- [AWS Cost and Usage Report](#)
- [Praktik Terbaik AWS Budgets](#)
- [Analitik Amazon S3](#)

Contoh terkait:

- [Lab Well-Architected: Biaya dan Penggunaan Tata Kelola](#)
- [Cara utama untuk mulai mengoptimalkan biaya cloud AWS](#)

## COST01-BP06 Memantau biaya secara proaktif

Gunakan alat dan dasbor untuk memantau biaya beban kerja secara proaktif. Tinjau biaya secara teratur dengan alat yang dikonfigurasi atau alat siap pakai, jangan hanya memeriksa biaya dan kategori saat Anda menerima notifikasinya. Memantau dan menganalisis biaya secara proaktif akan membantu mengidentifikasi tren positif dan memungkinkan Anda mempromosikan tren tersebut ke seluruh organisasi Anda.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Sedang

## Panduan implementasi

Sebaiknya pantau biaya dan penggunaan di organisasi Anda secara proaktif, tidak hanya saat terdapat pengecualian atau anomali. Dasbor yang dapat dilihat dari kantor atau lingkungan kerja Anda memastikan orang yang memiliki peran penting dapat mengakses informasi yang diperlukan, serta menunjukkan bahwa organisasi Anda fokus pada pengoptimalan biaya. Dengan dasbor yang terlihat ini, Anda dapat mendorong hasil secara aktif dan menerapkannya di seluruh organisasi Anda.

Buat rutinitas harian atau berulang untuk menggunakan [AWS Cost Explorer](#) atau dasbor lain seperti [Amazon QuickSight](#) untuk melihat biaya dan menganalisis secara proaktif. Lakukan analisis penggunaan dan biaya layanan AWS pada tingkat akun AWS, tingkat beban kerja, atau tingkat layanan AWS tertentu dengan pengelompokan dan pemfilteran, dan lakukan validasi apakah penggunaan dan biaya tersebut diharapkan atau tidak. Gunakan perincian dan tag tingkat sumber daya dan per jam untuk memfilter dan mengidentifikasi biaya yang timbul untuk sumber daya teratas. Anda juga dapat membuat laporan Anda sendiri dengan [Dasbor Inteligensi Biaya](#), solusi [Amazon QuickSight](#) yang dibuat Arsitek Solusi AWS, dan bandingkan anggaran Anda dengan biaya dan penggunaan aktual.

### Langkah implementasi

- Melaporkan pengoptimalan biaya: Siapkan siklus rutin untuk membahas dan menganalisis efisiensi beban kerja. Menggunakan metrik yang dibuat, laporkan metrik-metrik yang dicapai serta biaya untuk mencapainya. Identifikasikan dan perbaiki tren negatif apa pun, serta identifikasi tren positif yang dapat Anda pupuk di seluruh organisasi Anda. Pelaporan harus melibatkan perwakilan dari tim dan pemilik aplikasi, keuangan, dan manajemen.
- Buat dan aktifkan [AWS Budgets](#) perincian harian agar biaya dan penggunaan mengambil tindakan tepat waktu untuk mencegah kemungkinan kelebihan biaya: AWS Budgets memungkinkan Anda mengonfigurasi notifikasi peringatan, sehingga Anda terus mendapatkan informasi terbaru jika salah satu jenis anggaran Anda berada di luar ambang batas yang sudah dikonfigurasi sebelumnya. Cara terbaik untuk memanfaatkan AWS Budgets adalah menetapkan biaya dan penggunaan yang diharapkan sebagai batas Anda, agar nilai apa pun di atas anggaran Anda dapat dianggap sebagai pengeluaran yang berlebihan.
- Buat AWS Cost Anomaly Detection untuk pemantauan biaya: [AWS Cost Anomaly Detection](#) menggunakan teknologi Machine Learning canggih untuk mengidentifikasi pengeluaran yang tidak wajar dan akar masalah, sehingga Anda dapat mengambil tindakan secara cepat. Hal ini memungkinkan Anda mengonfigurasi pemantauan biaya yang menentukan segmen pengeluaran yang ingin Anda evaluasi (misalnya, tiap-tiap layanan AWS, akun anggota, tag alokasi biaya,

dan kategori biaya), dan memungkinkan Anda mengatur kapan, di mana, dan bagaimana Anda menerima notifikasi peringatan. Untuk setiap pemantauan, kaitkan beberapa langganan peringatan untuk pemilik bisnis dan tim teknologi, termasuk nama, ambang batas dampak biaya, dan frekuensi peringatan (tiap-tiap peringatan, ringkasan harian, ringkasan mingguan) untuk setiap langganan.

- Gunakan AWS Cost Explorer atau integrasikan data AWS Cost and Usage Report (CUR) dengan dasbor Amazon QuickSight untuk memvisualisasikan biaya organisasi: AWS Cost Explorer memiliki antarmuka yang mudah digunakan yang memungkinkan Anda memvisualisasikan, memahami, dan mengelola biaya dan penggunaan AWS dari waktu ke waktu. Fitur [Dasbor Inteligensi Biaya](#) adalah dasbor yang dapat disesuaikan dan dapat diakses untuk membantu membuat landasan alat manajemen dan pengoptimalan biaya Anda sendiri.

## Sumber daya

Dokumen terkait:

- [AWS Budgets](#)
- [AWS Cost Explorer](#)
- [Anggaran Biaya dan Penggunaan Harian](#)
- [AWS Cost Anomaly Detection](#)

Contoh terkait:

- [Lab Well-Architected: Visualisasi](#)
- [Lab Well-Architected: Visualisasi Tingkat Lanjut](#)
- [Lab Well-Architected: Dasbor Inteligensi Cloud](#)
- [Lab Well-Architected: Visualisasi Biaya](#)
- [Peringatan AWS Cost Anomaly Detection dengan Slack](#)

## COST01-BP07 Mengikuti perkembangan rilis layanan baru

Berkonsultasilah secara rutin dengan ahli atau Partner AWS untuk mempertimbangkan layanan dan fitur mana yang dapat menekan biaya. Pelajari blog AWS dan sumber informasi lain.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Sedang



## Panduan implementasi

AWS terus menambahkan kemampuan baru sehingga Anda dapat memanfaatkan teknologi terkini untuk bereksperimen dan berinovasi secara lebih cepat. Anda mungkin dapat mengimplementasikan layanan dan fitur AWS baru untuk meningkatkan efisiensi biaya di beban kerja Anda. Tinjau secara rutin [Manajemen Biaya AWS](#), [Blog Berita AWS](#), [Blog Manajemen Biaya AWS](#), dan [Apa yang Baru dengan AWS](#) untuk mendapatkan informasi tentang rilis layanan dan fitur baru. Postingan Apa yang Baru memberikan gambaran singkat tentang semua pengumuman perluasan layanan, fitur, dan Wilayah AWS yang dirilis.

### Langkah implementasi

- Berlangganan ke blog: Kunjungi halaman blog AWS dan berlangganan ke Blog “Apa yang Baru” dan blog relevan lain. Anda dapat mendaftar di [halaman preferensi komunikasi](#) dengan alamat email Anda.
- Berlangganan ke Berita AWS: Tinjau secara rutin [Blog Berita AWS](#) dan [Apa yang Baru dengan AWS](#) untuk mendapatkan informasi tentang rilis layanan dan fitur baru. Berlangganan umpan RSS atau gunakan email Anda untuk mengikuti pengumuman dan rilis.
- Ikuti Penurunan Harga AWS: Potongan harga reguler pada semua layanan kami sudah menjadi cara standar bagi AWS untuk melimpahkan efisiensi ekonomi yang diperoleh dari skala kami kepada pelanggan. Sejak April 2022, AWS telah menurunkan harga 115 kali sejak diluncurkan pada 2006. Jika Anda memiliki keputusan bisnis yang tertunda karena masalah harga, Anda dapat meninjaunya kembali setelah pengurangan harga dan integrasi layanan baru. Anda dapat mempelajari tentang upaya pengurangan harga sebelumnya, termasuk instans Amazon Elastic Compute Cloud (Amazon EC2), dalam [kategori pengurangan harga dalam Blog Berita AWS](#).
- Acara dan pertemuan AWS: Hadiri AWS Summit, dan pertemuan lokal dengan organisasi lainnya dari area setempat. Jika Anda tidak dapat hadir secara langsung, coba hadiri acara virtual untuk mendapatkan berbagai informasi dari pakar AWS dan mengetahui kasus bisnis pelanggan lainnya.
- Bertemu dengan tim akun Anda: Jadwalkan koordinasi rutin dengan tim akun Anda, adakan pertemuan, serta diskusikan tren industri dan layanan AWS. Bicarakan dengan manajer akun, Arsitek Solusi, dan tim dukungan Anda.

## Sumber daya

### Dokumen terkait:

- [Manajemen Biaya AWS](#)

- [Apa yang Baru dengan AWS](#)
- [Blog Berita AWS](#)

Contoh terkait:

- [Amazon EC2 – Mengoptimalkan dan Menghemat Biaya IT Anda Selama 15 Tahun](#)
- [Blog Berita AWS - Pengurangan Harga](#)

## COST01-BP08 Menciptakan budaya sadar biaya

Implementasikan perubahan atau program untuk menciptakan budaya sadar biaya di seluruh organisasi. Sebaiknya mulai dari cakupan kecil. Kemudian, seiring meningkatnya kemampuan dan penggunaan cloud oleh organisasi Anda, implementasikan program dengan cakupan yang lebih luas dan besar.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Rendah

### Panduan implementasi

Budaya sadar biaya memungkinkan Anda menskalakan pengoptimalan biaya dan Manajemen Finansial Cloud (tim operasi keuangan, pusat keunggulan cloud, operasi cloud, dan sebagainya) melalui praktik terbaik yang dilakukan secara organik dan terdesentralisasi di seluruh organisasi Anda. Kesadaran biaya ini memungkinkan Anda membuat kemampuan tingkat tinggi di seluruh organisasi dengan upaya yang minimal, dibandingkan dengan pendekatan tersentralisasi dari atas ke bawah yang kaku.

Menghadirkan kesadaran biaya dalam komputasi cloud, terutama untuk pendorong biaya utama dalam komputasi cloud, memungkinkan tim memahami hasil yang diharapkan untuk setiap perubahan dalam perspektif biaya. Tim yang mengakses lingkungan cloud harus mengetahui model harga dan perbedaan antara pusat data on-premise tradisional dan komputasi cloud.

Manfaat utama budaya sadar biaya adalah bahwa tim teknologi mengoptimalkan biaya secara proaktif dan secara kontinu (misalnya, biaya tersebut dianggap sebagai persyaratan nonfungsional saat merancang beban kerja baru atau membuat perubahan pada beban kerja yang ada), bukan melakukan pengoptimalan biaya reaktif seperlunya.

Perubahan kecil dalam budaya bisa berdampak besar terhadap efisiensi beban kerja saat ini dan masa mendatang. Contohnya adalah:

- Memberikan visibilitas dan memunculkan kesadaran dalam tim rekayasa untuk memahami fungsi dan dampak budaya tersebut dari segi biaya.
- Menerapkan mekanisme yang kompetitif pada biaya dan penggunaan di seluruh organisasi. Hal ini dapat dilakukan melalui dasbor yang terlihat secara publik, atau laporan yang membandingkan biaya dan penggunaan normal di antara berbagai tim (misalnya, biaya per beban kerja, biaya per transaksi).
- Penghargaan atas efisiensi biaya. Berikan penghargaan atas pencapaian pengoptimalan biaya yang dilakukan dengan sukarela atau tanpa diminta baik secara publik maupun privat, dan belajar dari kesalahan agar tidak mengulanginya di masa depan.
- Membuat persyaratan organisasi dari atas ke bawah untuk beban kerja yang dijalankan dengan anggaran yang telah ditentukan sebelumnya.
- Mempertanyakan kebutuhan bisnis yang akan dipenuhi oleh perubahan, dan dampak biaya dari perubahan yang diminta pada infrastruktur arsitektur atau konfigurasi beban kerja untuk memastikan Anda hanya membayar sesuai kebutuhan.
- Memastikan perencana perubahan mengetahui perubahan yang diharapkan yang memiliki dampak biaya, dan bahwa perubahan tersebut dikonfirmasi oleh pemangku kepentingan untuk memberikan hasil bisnis dengan biaya yang efektif.

## Langkah implementasi

- Laporkan biaya cloud ke tim teknologi: Untuk meningkatkan kesadaran biaya, dan menetapkan KPI efisiensi bagi pemangku kepentingan keuangan dan bisnis.
- Beri tahu pemangku kepentingan atau anggota tim tentang perubahan yang direncanakan: Buat item agenda untuk membahas perubahan yang direncanakan dan dampak manfaat biaya pada beban kerja selama rapat perubahan mingguan.
- Bertemu dengan tim akun Anda: Koordinasikan rapat rutin dengan tim akun Anda, dan diskusikan tren industri dan layanan AWS. Bicarakan dengan manajer akun, arsitek, dan tim dukungan Anda.
- Bagikan kisah sukses: Bagikan kisah sukses tentang pengurangan biaya untuk beban kerja, Akun AWS, atau organisasi apa pun untuk mendukung sikap dan dorongan positif seputar pengoptimalan biaya.
- Pelatihan: Pastikan tim teknis atau anggota tim mendapatkan pelatihan terkait kesadaran biaya sumber daya di AWS Cloud.
- Acara dan pertemuan AWS: Hadiri AWS Summit lokal, dan pertemuan lokal dengan organisasi lainnya dari area setempat.

- Berlangganan ke blog: Buka halaman blog AWS dan berlangganan [Blog “Apa yang Baru”](#) dan blog lain yang relevan untuk mengikuti rilis baru, implementasi, contoh, dan perubahan yang diinformasikan oleh AWS.

## Sumber daya

Dokumen terkait:

- [Blog AWS](#)
- [Manajemen Biaya AWS](#)
- [Blog Berita AWS](#)

Contoh terkait:

- [Manajemen Finansial Cloud AWS](#)
- [Lab AWS Well-Architected: Manajemen Finansial Cloud](#)

## COST01-BP09 Menghitung nilai bisnis dari optimasi biaya

Penghitungan nilai bisnis dari optimasi biaya memungkinkan Anda memahami seluruh set keuntungan untuk organisasi Anda. Karena optimasi biaya adalah investasi yang diperlukan, penghitungan nilai bisnis memungkinkan Anda untuk menjelaskan laba atas investasi kepada pemangku kepentingan. Penghitungan nilai bisnis dapat membantu Anda memperoleh lebih banyak dukungan dari pemangku kepentingan untuk investasi optimasi biaya mendatang, dan menyediakan kerangka kerja untuk mengukur hasil bagi aktivitas optimasi biaya organisasi Anda.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

## Panduan implementasi

Menghitung nilai bisnis berarti mengukur manfaat yang diperoleh bisnis dari tindakan dan keputusan yang mereka ambil. Nilai bisnis dapat berwujud (seperti berkurangnya biaya atau meningkatnya keuntungan) atau tidak berwujud (seperti meningkatnya reputasi merek atau meningkatnya kepuasan pelanggan).

Menghitung nilai bisnis dari pengoptimalan biaya berarti menentukan seberapa besar nilai atau manfaat yang Anda dapatkan dari upaya Anda mengefisienkan pengeluaran. Misalnya, jika

perusahaan mengeluarkan 100.000 USD untuk melakukan deployment beban kerja di AWS lalu mengoptimalkannya, biaya baru menjadi hanya 80.000 USD tanpa mengorbankan kualitas atau output. Dalam skenario ini, nilai bisnis yang terhitung dari pengoptimalan biaya adalah penghematan sebesar 20.000 USD. Namun selain penghematan, bisnis mungkin juga menghitung nilai dalam hal waktu pengiriman yang lebih cepat, kepuasan pelanggan yang meningkat, atau metrik lain yang dihasilkan dari upaya pengoptimalan biaya. Pemangku kepentingan perlu membuat keputusan tentang potensi nilai pengoptimalan biaya, biaya untuk mengoptimalkan beban kerja, dan nilai labanya.

Selain melaporkan penghematan dari optimasi biaya, Anda direkomendasikan untuk menghitung nilai tambah yang dihadirkan. Keuntungan optimasi biaya biasanya dihitung dari biaya terendah per hasil bisnis. Sebagai contoh, Anda bisa menghitung penghematan biaya Amazon Elastic Compute Cloud(Amazon EC2) ketika Anda membeli Savings Plans, yang mengurangi biaya dan mempertahankan tingkat output beban kerja. Anda dapat menghitung pengurangan biaya pada pengeluaran untuk AWS ketika instans Amazon EC2 yang tidak aktif dihapus, atau volume Amazon Elastic Block Store (Amazon EBS) yang tidak dilampirkan dihapus.

Namun, keuntungan dari optimasi biaya bukan sekadar pengurangan atau peniadaan biaya. Pertimbangkan pengambilan data tambahan untuk mengukur peningkatan efisiensi dan nilai bisnis.

## Langkah implementasi

- **Evaluasi manfaat bisnis:** Ini adalah proses menganalisis dan menyesuaikan biaya AWS Cloud dengan cara yang memaksimalkan manfaat yang diterima dari setiap nominal yang dikeluarkan. Alih-alih berfokus pada pengurangan biaya tanpa nilai bisnis, pertimbangkan manfaat bisnis dan laba atas investasi untuk pengoptimalan biaya, yang dapat membawa lebih banyak nilai dari uang yang Anda keluarkan. Intinya adalah mengeluarkan uang dengan bijak dan melakukan investasi dan pengeluaran di area yang paling banyak menghasilkan laba.
- **Analisis prakiraan biaya AWS:** Dengan melakukan prakiraan, pemangku kepentingan keuangan dapat menetapkan ekspektasi dengan pemangku kepentingan organisasi internal dan eksternal lain, dan membantu meningkatkan prediktabilitas finansial organisasi Anda. [AWS Cost Explorer](#) dapat digunakan untuk melakukan prakiraan biaya dan penggunaan Anda.

## Sumber daya

Dokumen terkait:

- [Ekonomi AWS Cloud](#)

- [Blog AWS](#)
- [Manajemen Biaya AWS](#)
- [Blog Berita AWS](#)
- [laporan resmi Pilar Keandalan Well-Architected](#)
- [AWS Cost Explorer](#)

Video terkait:

- [Unlock Business Value with Windows on AWS](#)

Contoh terkait:

- [Measuring and Maximizing the Business Value of Customer 360](#)
- [The Business Value of Adopting Amazon Web Services Managed Databases](#)
- [The Business Value of Amazon Web Services for Independent Software Vendors](#)
- [Business Value of Cloud Modernization](#)
- [The Business Value of Migration to Amazon Web Services](#)

# Kesadaran akan penggunaan dan pengeluaran

Memahami pendorong dan biaya organisasi Anda sangat penting untuk mengelola biaya dan penggunaan Anda secara efektif, serta untuk mengenali peluang penghematan biaya. Umumnya organisasi mengoperasikan beberapa beban kerja yang dijalankan oleh beberapa tim. Tim-tim ini dapat berada dalam unit organisasi yang berlainan, masing-masing dengan aliran pendapatannya sendiri. Kemampuan untuk mengaitkan biaya sumber daya dengan beban kerja, masing-masing organisasi, atau pemilik produk mendorong perilaku penggunaan yang efisien dan membantu mengurangi pemborosan. Pemantauan penggunaan dan biaya yang akurat memungkinkan Anda untuk memahami seberapa menguntungkan unit organisasi dan produk, serta memungkinkan Anda untuk mengambil keputusan yang lebih bijaksana tentang di mana harus mengalokasikan sumber daya di dalam organisasi Anda. Kesadaran akan penggunaan di semua tingkatan di organisasi merupakan kunci untuk mendorong perubahan, karena perubahan dalam penggunaan mendorong perubahan dalam biaya.

Pertimbangkan untuk mengambil pendekatan dengan beberapa aspek untuk menyadari penggunaan dan pengeluaran Anda. Tim Anda harus mengumpulkan data, menganalisis, kemudian melaporkannya. Faktor-faktor utama yang harus dipertimbangkan antara lain:

## Topik

- [Tata Kelola](#)
- [Pantau biaya dan penggunaan](#)
- [Sumber daya penonaktifan](#)

## Tata Kelola

Untuk mengelola biaya Anda di cloud, Anda harus mengelola penggunaan Anda melalui area tata kelola berikut:

### Praktik terbaik

- [COST02-BP01 Mengembangkan kebijakan berdasarkan keperluan organisasi Anda](#)
- [COST02-BP02 Mengimplementasikan sasaran dan target](#)
- [COST02-BP03 Mengimplementasikan struktur akun](#)
- [COST02-BP04 Mengimplementasikan grup dan peran](#)

- [COST02-BP05 Mengimplementasikan kontrol biaya](#)
- [COST02-BP06 Melacak siklus hidup proyek](#)

## COST02-BP01 Mengembangkan kebijakan berdasarkan keperluan organisasi Anda

Kembangkan kebijakan yang menentukan cara organisasi Anda mengelola sumber daya dan periksa sumber daya secara berkala. Kebijakan harus mencakup aspek biaya sumber daya dan beban kerja, termasuk pembuatan, pengubahan, dan penonaktifan selama masa pakai sumber daya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Memahami pendorong dan biaya organisasi Anda sangat penting untuk mengelola biaya dan penggunaan Anda secara efektif, serta untuk mengenali peluang penghematan biaya. Umumnya organisasi mengoperasikan beberapa beban kerja yang dijalankan oleh beberapa tim. Tim-tim ini dapat berada dalam unit organisasi yang berlainan, masing-masing dengan aliran pendapatannya sendiri. Kemampuan untuk mengaitkan biaya sumber daya dengan beban kerja, tiap-tiap organisasi, atau pemilik produk mendorong perilaku penggunaan yang efisien dan membantu mengurangi pemborosan. Pemantauan biaya dan penggunaan yang akurat membantu Anda memahami seberapa optimal suatu beban kerja, serta seberapa menguntungkan produk dan unit organisasi. Pengetahuan ini memungkinkan pengambilan keputusan yang lebih tepat tentang target pengalokasian sumber daya di dalam organisasi Anda. Kesadaran tentang penggunaan pada semua tingkatan di organisasi sangat penting untuk mendorong perubahan, karena perubahan pada penggunaan akan mendorong perubahan pada biaya. Pertimbangkan untuk mengambil pendekatan dengan beberapa aspek untuk menyadari penggunaan dan pengeluaran Anda.

Langkah pertama dalam menjalankan tata kelola adalah dengan menggunakan keperluan organisasi Anda untuk mengembangkan kebijakan penggunaan cloud Anda. Kebijakan tersebut menentukan cara organisasi Anda menggunakan cloud dan cara mengelola sumber daya. Kebijakan harus mencakup semua aspek dalam sumber daya dan beban kerja yang berkaitan dengan biaya dan penggunaan, termasuk pembuatan, pengubahan, serta penonaktifan selama masa pakai sumber daya. Verifikasikan bahwa kebijakan dan prosedur diikuti dan diterapkan untuk setiap perubahan di lingkungan cloud. Selama rapat manajemen perubahan IT Anda, ajukan pertanyaan untuk mengetahui dampak biaya dari perubahan yang direncanakan apakah meningkat atau menurun, justifikasi bisnis, dan hasil yang diharapkan.



Kebijakan harus dibuat sederhana agar dapat dipahami dengan mudah dan diimplementasikan secara efektif di seluruh organisasi. Kebijakan juga harus mudah diikuti dan ditafsirkan (sehingga digunakan) serta spesifik (tidak ada salah tafsir antar tim). Selain itu, kebijakan perlu diperiksa secara berkala (seperti mekanisme kami) dan diperbarui seiring perubahan kondisi atau prioritas bisnis pelanggan yang dapat menjadikan kebijakan usang.

Mulailah dengan kebijakan tingkat tinggi yang sangat umum, seperti Wilayah geografis yang digunakan atau waktu dalam sehari saat sumber daya harus dijalankan. Sempurnakan kebijakan-kebijakan tersebut secara bertahap untuk unit dan beban kerja organisasi yang beragam. Kebijakan yang umum mencakup layanan dan fitur yang dapat digunakan (misalnya, penyimpanan yang berperforma lebih rendah di lingkungan pengujian dan pengembangan), jenis sumber daya yang dapat digunakan oleh grup yang berbeda-beda (misalnya, ukuran sumber daya terbesar dalam akun pengembangan adalah ukuran sedang), dan jangka waktu sumber daya tersebut akan digunakan (sementara, jangka pendek, atau untuk jangka waktu tertentu).

### Contoh kebijakan

Berikut ini adalah contoh kebijakan yang dapat Anda kaji untuk membuat kebijakan tata kelola cloud Anda sendiri, yang berfokus pada pengoptimalan biaya. Pastikan Anda menyesuaikan kebijakan berdasarkan kebutuhan organisasi dan permintaan pemangku kepentingan Anda.

- Nama kebijakan: Tentukan nama kebijakan yang jelas, seperti Kebijakan Optimalisasi Sumber Daya dan Penurunan Biaya.
- Tujuan: Jelaskan alasan kebijakan ini harus digunakan serta hasil yang diharapkan. Tujuan dari kebijakan ini adalah untuk memverifikasi bahwa diperlukan biaya yang minim untuk men-deploy dan menjalankan beban kerja yang diinginkan untuk memenuhi persyaratan bisnis.
- Cakupan: Tentukan dengan jelas siapa yang harus menggunakan kebijakan ini dan kapan kebijakan ini harus digunakan, seperti Tim DevOps X menggunakan kebijakan ini pada pelanggan us-east untuk lingkungan X (produksi atau non-produksi).

### Pernyataan kebijakan

1. Pilih us-east-1 atau beberapa wilayah us-east berdasarkan lingkungan beban kerja dan kebutuhan bisnis Anda (pengembangan, pengujian penerimaan pengguna, praproduksi, atau produksi).
2. Jadwal instans Amazon EC2 dan Amazon RDS untuk berjalan antara pukul enam pagi dan delapan malam (Waktu Standar Timur (EST)).

3. Hentikan semua instans Amazon EC2 yang tidak digunakan setelah delapan jam dan instans Amazon RDS yang tidak digunakan setelah 24 jam tidak aktif.
4. Hentikan semua instans Amazon EC2 yang tidak digunakan setelah 24 jam tidak aktif di lingkungan non-produksi. Ingatkan pemilik instans Amazon EC2 (berdasarkan tag) untuk meninjau instans Amazon EC2 mereka yang dihentikan dalam produksi dan beri tahu mereka bahwa instans Amazon EC2 mereka akan diakhiri dalam waktu 72 jam jika tidak digunakan.
5. Gunakan keluarga dan ukuran instans umum seperti m5.large, kemudian ubah ukuran instans berdasarkan penggunaan CPU dan memori menggunakan AWS Compute Optimizer.
6. Prioritaskan menggunakan penskalaan otomatis untuk menyesuaikan jumlah instans yang berjalan secara dinamis berdasarkan lalu lintas.
7. Gunakan instans spot untuk beban kerja non-kritis.
8. Kaji persyaratan kapasitas untuk memberikan komitmen untuk paket penyimpanan atau instans cadangan untuk beban kerja yang dapat diprediksi dan beri tahu Tim Manajemen Keuangan Cloud.
9. Gunakan kebijakan siklus hidup Amazon S3 untuk memindahkan data yang jarang diakses ke tingkat penyimpanan yang lebih murah. Jika tidak ada kebijakan penyimpanan yang ditentukan, gunakan Amazon S3 Intelligent Tiering untuk memindahkan objek ke tingkat yang diarsipkan secara otomatis.
10. Pantau pemanfaatan sumber daya dan atur alarm untuk memicu peristiwa penskalaan menggunakan Amazon CloudWatch.
11. Untuk tiap Akun AWS, gunakan AWS Budgets untuk menetapkan anggaran biaya dan penggunaan untuk akun Anda berdasarkan unit bisnis dan pusat biaya.
12. Menggunakan AWS Budgets untuk menetapkan anggaran biaya dan penggunaan untuk akun Anda dapat membantu Anda terus memantau pengeluaran dan menghindari tagihan tidak terduga, sehingga Anda dapat lebih mengontrol biaya Anda.

Prosedur: Berikan prosedur mendetail untuk menerapkan kebijakan ini atau beri referensi ke dokumen lain yang menjelaskan cara menerapkan setiap pernyataan kebijakan. Bagian ini harus memberikan petunjuk langkah demi langkah untuk menjalankan persyaratan kebijakan.

Untuk menerapkan kebijakan ini, Anda dapat menggunakan berbagai alat pihak ketiga atau aturan AWS Config untuk memeriksa kepatuhan terhadap pernyataan kebijakan dan memicu tindakan perbaikan otomatis menggunakan fungsi AWS Lambda. Anda juga dapat menggunakan AWS Organizations untuk menegakkan kebijakan. Selain itu, Anda harus secara rutin meninjau

penggunaan sumber daya Anda dan menyesuaikan kebijakan apabila diperlukan untuk memastikan kebijakan tersebut terus memenuhi kebutuhan bisnis Anda.

## Langkah implementasi

- Bertemu dengan para pemangku kepentingan: Untuk mengembangkan kebijakan, mintalah pemangku kepentingan (kantor bisnis cloud, rekayasawan, atau pengambil keputusan fungsional untuk penegakan kebijakan) di dalam organisasi Anda untuk menentukan kebutuhan mereka dan mendokumentasikannya. Gunakan pendekatan berulang dengan memulai dari hal paling umum dan menyempurnakannya secara berkelanjutan hingga ke unit terkecil di setiap langkahnya. Anggota tim yang terlibat adalah mereka yang berkepentingan langsung dengan beban kerja, seperti unit organisasi atau pemilik aplikasi, serta grup pendukung, seperti tim keamanan dan keuangan.
- Minta konfirmasi: Pastikan semua tim menyetujui kebijakan terkait siapa yang dapat mengakses dan melakukan deployment ke AWS Cloud. Pastikan mereka mengikuti kebijakan organisasi Anda serta konfirmasikan bahwa pembuatan sumber daya mereka sejalan dengan kebijakan dan prosedur yang disetujui.
- Buat sesi pelatihan orientasi: Minta semua anggota organisasi baru untuk menyelesaikan kursus pelatihan orientasi untuk menciptakan kesadaran biaya dan membentuk persyaratan organisasi. Mereka boleh mempertimbangkan berbagai kebijakan dari pengalaman mereka sebelumnya atau tidak mempertimbangkannya sama sekali.
- Tentukan lokasi untuk beban kerja Anda: Tentukan lokasi pengoperasian beban kerja Anda, termasuk negara dan wilayah di dalam negara tersebut. Informasi ini digunakan untuk pemetaan ke Wilayah AWS dan Zona Ketersediaan.
- Tentukan dan kelompokkan layanan serta sumber daya: Tentukan layanan yang diperlukan beban kerja. Untuk setiap layanan, tentukan jenis, ukuran, dan jumlah sumber daya yang diperlukan. Tentukan grup sumber daya berdasarkan fungsi, seperti server aplikasi atau penyimpanan basis data. Sumber daya dapat dimiliki oleh beberapa grup.
- Tentukan dan kelompokkan pengguna berdasarkan fungsi: Tentukan pengguna yang berinteraksi dengan beban kerja, dengan berfokus pada apa yang mereka kerjakan dan cara mereka menggunakan beban kerja, bukan posisi atau jabatan mereka di organisasi. Kelompokkan pengguna atau fungsi yang serupa menjadi satu. Anda dapat menggunakan kebijakan yang dikelola AWS sebagai panduan.
- Tentukan tindakan: Dengan lokasi, sumber daya, dan pengguna yang telah diidentifikasi di awal, tentukan tindakan yang diperlukan oleh masing-masing untuk meraih hasil beban kerja dari masa pakainya (pengembangan, operasi, dan penonaktifan). Identifikasikan tindakannya berdasarkan

grup, bukan masing-masing elemen dalam grup, di setiap lokasi. Mulailah dari hal yang umum dengan membaca atau menulis, kemudian sesuaikan tindakan-tindakan tertentu untuk setiap layanan.

- Tentukan periode peninjauan: Beban kerja dan keperluan organisasi dapat berubah seiring waktu. Tentukan jadwal peninjauan beban kerja untuk memastikan ini tetap selaras dengan prioritas organisasi.
- Dokumentasikan kebijakan: Lakukan verifikasi bahwa kebijakan yang telah ditentukan dapat diakses saat dibutuhkan oleh organisasi Anda. Kebijakan-kebijakan ini digunakan untuk mengimplementasikan, memelihara, dan mengaudit akses lingkungan Anda.

## Sumber daya

Dokumen terkait:

- [Manajemen Perubahan di Cloud](#)
- [Kebijakan Terkelola AWS untuk berbagai Fungsi Pekerjaan](#)
- [Strategi penagihan beberapa akun AWS](#)
- [Tindakan, Sumber Daya, dan Kunci Kondisi untuk Layanan AWS](#)
- [Manajemen dan Tata Kelola AWS](#)
- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [Zona Ketersediaan \(AZ\) dan Wilayah Infrastruktur Global](#)

Video terkait:

- [Manajemen dan Tata Kelola AWS dalam Skala Besar](#)

Contoh terkait:

- [VMware - Apa yang Dimaksud dengan Kebijakan Cloud?](#)

## COST02-BP02 Mengimplementasikan sasaran dan target

Implementasikan sasaran serta target biaya dan penggunaan untuk beban kerja Anda. Sasaran memberikan arah untuk organisasi Anda tentang hasil yang diharapkan, dan target memberikan hasil terukur spesifik yang harus dicapai untuk beban kerja Anda.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

## Panduan implementasi

Kembangkan sasaran serta target biaya dan penggunaan untuk organisasi Anda. Sebagai organisasi yang sedang berkembang di AWS, penting untuk menetapkan dan melacak sasaran untuk optimalisasi biaya. Sasaran atau [indikator kinerja utama \(KPI\)](#) ini dapat mencakup hal-hal seperti persentase pengeluaran sesuai permintaan, atau adopsi layanan tertentu yang dioptimalkan seperti instans AWS Graviton atau tipe volume EBS gp3. Menetapkan sasaran yang jelas dan dapat dicapai dapat membantu Anda untuk terus mengukur peningkatan efisiensi yang penting bagi operasional bisnis yang sedang berlangsung. Sasaran memberikan panduan dan arah kepada organisasi Anda terkait hasil yang diharapkan. Target memberikan hasil yang jelas dan spesifik yang harus dicapai. Singkatnya, sasaran adalah arah yang ingin Anda tuju sedangkan target adalah seberapa jauh lagi jarak ke arah tersebut dan kapan sasaran tersebut harus dicapai (menggunakan panduan “spesifik, terukur, dapat ditetapkan, realistis, dan tepat waktu”, atau SMART (Specific, Measurable Assignable, Realistic, Timely)). Contoh sasaran adalah penggunaan platform harus meningkat secara signifikan, hanya dengan peningkatan biaya yang minim (nonlinier). Contoh target adalah peningkatan penggunaan platform 20%, dengan peningkatan biaya kurang dari lima persen. Contoh sasaran umum lainnya adalah beban kerja harus lebih efisien setiap enam bulan. Target yang menyertainya adalah metrik biaya per bisnis harus turun lima persen setiap enam bulan.

Sasaran untuk optimalisasi biaya adalah meningkatkan efisiensi beban kerja, yakni mengurangi biaya per hasil bisnis untuk beban kerja tersebut dari waktu ke waktu. Sebaiknya implementasikan sasaran ini ke semua beban kerja, serta tetapkan target seperti peningkatan efisiensi sebesar lima persen setiap enam bulan hingga satu tahun. Hal ini dapat dicapai di cloud melalui pengembangan kemampuan dalam optimalisasi biaya, serta melalui perlisian layanan dan fitur baru.

Penting untuk memiliki visibilitas hampir waktu nyata terhadap KPI dan peluang penghematan terkait serta melacak kemajuan Anda dari waktu ke waktu. Untuk mulai menetapkan dan melacak sasaran KPI, kami merekomendasikan dasbor KPI dari [Kerangka kerja Cloud Intelligence Dashboards \(CID\)](#). Berdasarkan data dari AWS Cost and Usage Report, dasbor KPI menyediakan serangkaian KPI optimalisasi biaya yang direkomendasikan dengan kemampuan untuk menetapkan sasaran khusus dan melacak kemajuan dari waktu ke waktu.

Jika Anda memiliki solusi lain yang memungkinkan Anda menetapkan dan melacak sasaran KPI, pastikan solusi tersebut diadopsi oleh semua pemangku kepentingan manajemen keuangan cloud di organisasi Anda.

## Langkah implementasi

- Tetapkan tingkat penggunaan yang diharapkan: Sebagai permulaan, fokuslah pada tingkat penggunaan. Berinteraksilah dengan pemilik aplikasi, pemasaran, dan tim bisnis yang lebih besar untuk memahami tingkat penggunaan yang diharapkan untuk beban kerja. Bagaimana permintaan pelanggan akan berubah seiring waktu, dan apakah akan ada perubahan akibat peningkatan musiman atau kampanye pemasaran?
- Tetapkan pengadaan sumber daya dan biaya beban kerja: Setelah tingkat penggunaan ditetapkan, hitung perubahan pada sumber daya beban kerja yang diperlukan untuk memenuhi tingkat penggunaan ini. Anda mungkin perlu meningkatkan ukuran atau jumlah sumber daya untuk suatu komponen beban kerja, meningkatkan transfer data, atau mengubah komponen beban kerja ke layanan lain pada tingkat tertentu. Tentukan berapa biaya yang akan dikeluarkan pada setiap poin utama ini, dan berapa perubahan biaya ketika terdapat perubahan pada penggunaan.
- Tetapkan sasaran bisnis: Dengan mengambil output dari perkiraan perubahan penggunaan dan biaya, gabungkan dengan perkiraan perubahan dalam teknologi, atau program apa pun yang sedang Anda jalankan, dan kembangkan sasaran untuk beban kerja. Sasaran harus mencakup penggunaan, biaya, serta hubungan di antara keduanya. Sasaran harus sederhana, berada pada tingkat tinggi, dan membantu orang memahami apa yang diharapkan bisnis dalam hal hasil (seperti memastikan sumber daya yang tidak terpakai dipertahankan di bawah tingkat biaya tertentu). Anda tidak perlu menentukan sasaran untuk setiap jenis sumber daya yang tidak terpakai atau menentukan biaya yang menyebabkan kerugian untuk sasaran dan target. Pastikan terdapat program organisasi (misalnya, pengembangan kemampuan seperti pelatihan dan pendidikan) jika terdapat perubahan yang diperkirakan pada biaya tanpa perubahan pada penggunaan.
- Tetapkan target: Untuk setiap sasaran yang telah ditetapkan, tentukan target yang terukur. Jika sasarannya adalah untuk meningkatkan efisiensi beban kerja, targetnya adalah menghitung jumlah peningkatan (biasanya pada output bisnis untuk setiap dolar yang dikeluarkan) dan kapan peningkatan tersebut akan tercapai. Misalnya, jika Anda menetapkan sasaran untuk meminimalkan pemborosan karena kelebihan penyediaan, target Anda dapat berupa pemborosan akibat kelebihan penyediaan komputasi di tingkat pertama beban kerja produksi tidak boleh melebihi 10% dari biaya komputasi tingkatan dan pemborosan akibat kelebihan penyediaan komputasi di tingkat kedua beban kerja produksi tidak boleh melebihi 5% dari biaya komputasi tingkatan.

## Sumber daya

Dokumen terkait:

- [Kebijakan terkelola AWS untuk berbagai fungsi pekerjaan](#)
- [Strategi multiakun AWS untuk zona landasan AWS Control Tower Anda](#)
- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [Sasaran SMART](#)

Video terkait:

- [Lab Well-Architected: Sasaran dan Target \(Tingkat 100\)](#)

Contoh terkait:

- [Lab Well-Architected: Menonaktifkan sumber daya \(Sasaran dan Target\)](#)
- [Lab Well-Architected: Jenis, Ukuran, dan Jumlah Sumber Daya \(Sasaran dan Target\)](#)

## COST02-BP03 Mengimplementasikan struktur akun

Implementasikan struktur akun yang memetakan ke organisasi Anda. Hal ini dapat membantu alokasi dan pengelolaan biaya di organisasi Anda.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

AWS Organizations memungkinkan Anda membuat beberapa Akun AWS yang dapat membantu Anda melakukan tata kelola lingkungan secara terpusat seiring Anda menskalakan beban kerja Anda di AWS. Anda dapat memodelkan hierarki organisasi Anda dengan mengelompokkan Akun AWS dalam struktur unit organisasi (OU) dan membuat beberapa Akun AWS di bawah setiap OU. Untuk membuat struktur akun, Anda perlu memutuskan mana dari Akun AWS Anda yang akan menjadi akun manajemen terlebih dahulu. Setelah itu, Anda dapat membuat Akun AWS baru atau memilih akun-akun yang sudah ada sebagai akun anggota berdasarkan struktur akun rancangan Anda dengan mengikuti [praktik terbaik akun manajemen](#) dan [praktik terbaik akun anggota](#).

Disarankan untuk memiliki setidaknya satu akun manajemen dengan satu akun anggota yang terhubung ke sana, terlepas dari ukuran organisasi Anda dan penggunaannya. Semua sumber daya beban kerja hanya boleh berada di akun anggota dan tidak boleh ada sumber daya yang dibuat di dalam akun manajemen. Jumlah Akun AWS yang harus Anda miliki akan bergantung pada

kebutuhan Anda. Evaluasi model biaya dan operasional Anda saat ini dan pada masa mendatang untuk memastikan bahwa struktur Akun AWS Anda sudah sesuai dengan tujuan organisasi Anda. Beberapa perusahaan membuat banyak Akun AWS untuk alasan bisnis, misalnya:

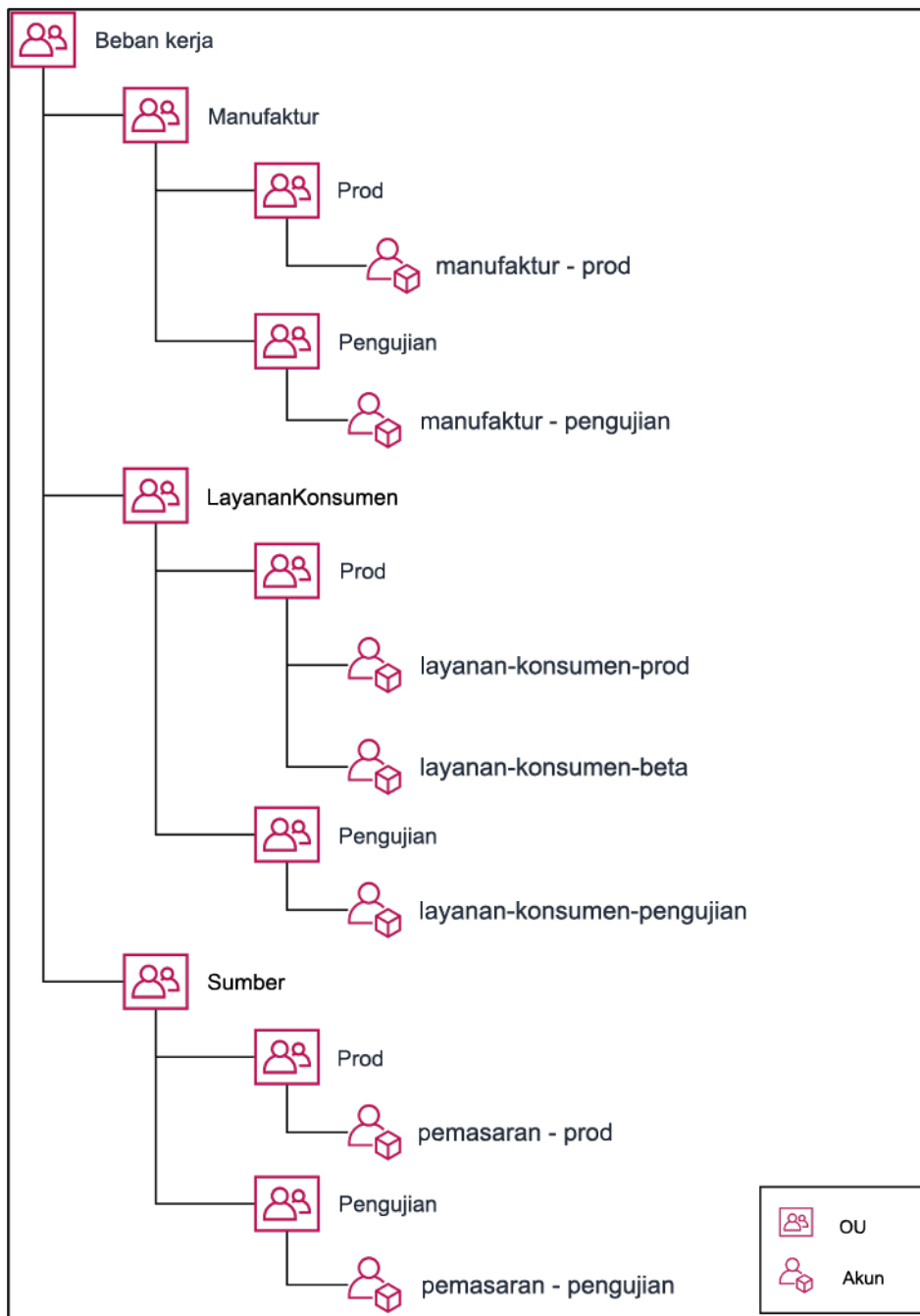
- Isolasi penagihan dan fiskal atau administratif diperlukan di antara unit organisasi, pusat biaya, atau beban kerja tertentu.
- Layanan AWS memiliki batasan tersendiri untuk beban kerja tertentu.
- Ada persyaratan untuk isolasi dan pemisahan antara beban kerja dan sumber daya.

Di dalam [AWS Organizations](#), [tagihan yang terkonsolidasi](#) membuat konstruksi antara satu atau beberapa akun anggota dan akun manajemen. Dengan akun anggota, Anda dapat mengisolasi serta membedakan biaya dan penggunaan berdasarkan grup. Praktik terbaik yang umum dilakukan adalah membuat akun anggota terpisah untuk setiap unit organisasi (seperti keuangan, pemasaran, dan penjualan), atau untuk setiap siklus hidup lingkungan (seperti pengembangan, pengujian, dan produksi), atau untuk setiap beban kerja (beban kerja a, b, dan c), kemudian menggabungkan akun yang terhubung ini menggunakan penagihan terkonsolidasi.

Penagihan terkonsolidasi membantu menggabungkan pembayaran untuk beberapa Akun AWS anggota dalam satu akun manajemen, dan tetap memberikan visibilitas untuk setiap aktivitas akun yang dihubungkan. Karena biaya dan penggunaan digabungkan dalam akun manajemen, Anda dapat memaksimalkan diskon volume layanan Anda, dan memaksimalkan penggunaan diskon komitmen Anda (Savings Plans dan Instans Terpesan) untuk mencapai diskon tertinggi.

Diagram berikut ini menunjukkan bagaimana Anda dapat menggunakan AWS Organizations dengan unit organisasi (OU) untuk mengelompokkan beberapa akun, dan menempatkan beberapa Akun AWS di setiap OU. Disarankan untuk menggunakan OU untuk berbagai kasus penggunaan dan beban kerja yang menyediakan pola untuk mengatur akun.





Contoh pengelompokan beberapa Akun AWS di bawah unit-unit organisasi.

[AWS Control Tower](#) dapat mengatur dan mengonfigurasi beberapa akun AWS dengan cepat, memastikan tata kelola sesuai dengan persyaratan organisasi Anda.

### Langkah implementasi

- Tentukan persyaratan pemisahan: Persyaratan pemisahan adalah kombinasi dari beberapa faktor, termasuk keamanan, keandalan, dan konstruksi keuangan. Siapkan setiap faktor secara berurutan

dan tentukan apakah beban kerja atau lingkungan beban kerja harus dipisahkan dari beban kerja lainnya. Keamanan meningkatkan perekatan pada persyaratan akses dan data. Keandalan mengelola batasan sehingga lingkungan dan beban kerja tidak memengaruhi hal lain. Pelajari pilar keamanan dan keandalan Kerangka Kerja Well-Architected secara berkala dan ikuti praktik-praktik terbaik yang disediakan. Konstruksi keuangan menciptakan pemisahan keuangan yang ketat (pusat biaya, kepemilikan beban kerja, dan akuntabilitas yang berbeda). Contoh umum pemisahan adalah menjalankan beban kerja pengujian dan produksi di akun terpisah, atau menggunakan akun terpisah agar data penagihan dan faktur dapat diberikan ke unit bisnis atau departemen individu di dalam organisasi atau pemangku kepentingan yang memiliki akun.

- Tentukan persyaratan pengelompokan: Persyaratan pengelompokan tidak menggantikan persyaratan pemisahan, tetapi digunakan untuk membantu manajemen. Kelompokkan lingkungan atau beban kerja serupa yang tidak perlu dipisah. Misalnya, kelompokkan beberapa lingkungan pengujian atau pengembangan dari satu atau beberapa beban kerja menjadi satu.
- Tentukan struktur akun: Dengan menggunakan pemisahan dan pengelompokan ini, tentukan akun untuk setiap kelompok dan penuhi persyaratan pemisahan. Akun-akun ini adalah akun anggota atau akun yang Anda hubungkan. Dengan mengelompokkan akun anggota ini ke dalam satu akun manajemen atau pembayar, Anda menggabungkan penggunaan untuk memperbesar volume diskon di semua akun, yang menyediakan satu tagihan untuk semua akun. Anda dapat memisahkan data penagihan dan menampilkannya secara terpisah untuk setiap akun anggota. Jika akun anggota tidak ingin memperlihatkan data penagihan atau penggunaannya kepada akun lain, atau harus ada tagihan terpisah dari AWS, tentukan beberapa akun manajemen atau pembayar. Dalam hal ini, setiap akun anggota memiliki akun manajemen atau pembayarnya masing-masing. Sumber daya harus selalu disimpan di akun anggota atau terkait. Akun manajemen atau pembayar hanya digunakan untuk manajemen.

## Sumber daya

Dokumen terkait:

- [Menggunakan Tag Alokasi Biaya](#)
- [Kebijakan terkelola AWS untuk fungsi tugas](#)
- [Strategi penagihan beberapa akun AWS](#)
- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [AWS Control Tower](#)
- [AWS Organizations](#)

- Praktik terbaik untuk [akun manajemen](#) dan [akun anggota](#)
- [Mengatur lingkungan AWS Anda Menggunakan Beberapa Akun](#)
- [Mengaktifkan instans terpesan bersama dan diskon Savings Plans](#)
- [Penagihan terkonsolidasi](#)
- [Penagihan terkonsolidasi](#)

Contoh terkait:

- [Memisahkan CUR dan Berbagi Akses](#)

Video terkait:

- [Memperkenalkan AWS Organizations](#)
- [Menyiapkan Lingkungan AWS Multiakun yang Menggunakan Praktik Terbaik untuk AWS Organizations](#)

Contoh terkait:

- [Lab Well-Architected: Membuat Organisasi AWS \(Level 100\)](#)
- [Memisahkan AWS Cost and Usage Report dan Berbagi Akses](#)
- [Menentukan Strategi Multiakun AWS untuk perusahaan telekomunikasi](#)
- [Praktik Terbaik untuk Mengoptimalkan Akun AWS](#)
- [Praktik Terbaik untuk Unit Organisasi dengan AWS Organizations](#)

## COST02-BP04 Mengimplementasikan grup dan peran

Implementasikan grup dan peran yang selaras dengan kebijakan Anda serta kontrol siapa saja yang dapat membuat, mengubah, atau menonaktifkan instans dan sumber daya di setiap grup. Misalnya, implementasikan grup pengembangan, pengujian, dan produksi. Ini berlaku untuk solusi pihak ketiga dan layanan AWS.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

## Panduan implementasi

Peran dan grup pengguna merupakan blok pembangun dasar dalam desain dan implementasi sistem yang aman dan efisien. Peran dan grup membantu organisasi menyeimbangkan perlunya kontrol dengan kebutuhan akan fleksibilitas dan produktivitas, yang pada akhirnya mendukung tujuan organisasi dan kebutuhan pengguna. Seperti yang direkomendasikan di bagian [Manajemen identitas dan akses](#) dalam Pilar Keamanan Kerangka Kerja AWS Well-Architected, Anda memerlukan manajemen identitas dan izin yang kuat untuk menyediakan akses ke sumber daya yang tepat untuk orang yang tepat dalam kondisi yang tepat. Pengguna hanya menerima akses yang diperlukan untuk menyelesaikan tugasnya. Ini meminimalkan risiko yang terkait dengan akses yang tidak sah atau penyalahgunaan.

Setelah mengembangkan kebijakan, Anda dapat membuat peran dan grup logis pengguna di organisasi Anda. Ini memungkinkan Anda untuk menetapkan izin, mengontrol penggunaan, dan membantu menerapkan mekanisme kontrol akses yang kuat, yang mencegah akses tidak sah ke informasi sensitif. Awali dengan pengelompokan orang tingkat tinggi. Hal ini biasanya selaras dengan unit organisasi dan peran pekerjaan (misalnya, administrator sistem di Departemen IT, pengontrol keuangan, atau analis bisnis). Grup tersebut mengategorikan orang yang memiliki tugas serupa dan memerlukan akses serupa. Peran menentukan aktivitas grup. Mengelola izin untuk grup dan peran lebih mudah dibandingkan untuk pengguna individu. Peran dan grup menetapkan izin secara konsisten dan sistematis bagi semua pengguna, sehingga mencegah kesalahan dan inkonsistensi.

Ketika peran pengguna berubah, administrator dapat menyesuaikan akses di tingkat peran atau grup, bukan mengonfigurasi ulang akun setiap pengguna. Misalnya, administrator sistem di TI memerlukan akses untuk membuat semua sumber daya, sedangkan tim analitik hanya perlu membuat sumber daya analitik.

### Langkah implementasi

- Implementasikan grup: Dengan grup pengguna yang ditentukan dalam kebijakan organisasi Anda, implementasikan grup yang sesuai, jika perlu. Untuk praktik terbaik tentang pengguna, grup, dan autentikasi, lihat [Pilar Keamanan](#) Kerangka Kerja AWS Well-Architected.
- Implementasikan peran dan kebijakan: Dengan menggunakan tindakan yang ditentukan dalam kebijakan organisasi Anda, buat kebijakan akses dan peran yang diperlukan. Untuk praktik terbaik tentang peran dan kebijakan, lihat [Pilar Keamanan](#) Kerangka Kerja AWS Well-Architected.

## Sumber daya

Dokumen terkait:

- [Kebijakan terkelola AWS untuk fungsi tugas](#)
- [Strategi penagihan beberapa akun AWS](#)
- [Pilar Keamanan Kerangka Kerja AWS Well-Architected](#)
- [AWS Identity and Access Management \(IAM\)](#)
- [Kebijakan AWS Identity and Access Management](#)

Video terkait:

- [Why use Identity and Access Management](#)

Contoh terkait:

- [Akses dan Identitas Dasar Lab Well-Architected](#)
- [Mengontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [Starting your Cloud Financial Management journey: Cloud cost operations](#)

## COST02-BP05 Mengimplementasikan kontrol biaya

Implementasikan kontrol berdasarkan kebijakan organisasi serta grup dan peran yang ditetapkan. Ini semua memastikan bahwa biaya hanya dikenakan sesuai yang ditetapkan oleh persyaratan organisasi seperti kontrol akses ke wilayah atau tipe sumber daya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Langkah pertama yang umum dalam mengimplementasikan kontrol biaya adalah mengatur notifikasi ketika peristiwa biaya atau penggunaan terjadi di luar kebijakan. Anda dapat bertindak cepat dan memverifikasi apakah diperlukan tindakan perbaikan, tanpa membatasi atau mengganggu beban kerja atau aktivitas baru. Setelah mengetahui batas beban kerja dan lingkungan, Anda dapat menegakkan tata kelola. [AWS Budgets](#) memungkinkan Anda untuk mengatur notifikasi dan menetapkan anggaran bulanan untuk biaya, penggunaan, diskon komitmen (Savings Plans dan Instans Terpesan) AWS Anda. Anda dapat membuat anggaran pada tingkat biaya agregat (misalnya

semua biaya), atau pada tingkat yang lebih mendetail yakni hanya menyertakan dimensi tertentu seperti akun yang dihubungkan, layanan, tag, atau Zona Ketersediaan.

Setelah Anda mengatur batas anggaran dengan AWS Budgets, gunakan [AWS Cost Anomaly Detection](#) untuk mengurangi biaya tidak terduga. AWS Cost Anomaly Detection adalah layanan manajemen biaya yang menggunakan machine learning untuk memantau biaya dan penggunaan Anda secara kontinu guna mendeteksi pengeluaran yang tidak wajar. Hal ini membantu Anda mengidentifikasi pengeluaran yang tidak wajar dan akar masalah, sehingga Anda dapat mengambil tindakan secara cepat. Pertama-tama, buat pemantau biaya di AWS Cost Anomaly Detection, lalu pilih preferensi pemberitahuan Anda dengan mengatur ambang batas biaya (seperti pemberitahuan anomali dengan dampak lebih dari \$1.000). Setelah Anda menerima pemberitahuan, Anda dapat menganalisis akar masalah penyebab anomali dan dampaknya pada biaya Anda. Anda juga dapat memantau dan melakukan analisis anomali Anda sendiri di AWS Cost Explorer.

Tegakkan kebijakan tata kelola di AWS melalui [AWS Identity and Access Management](#) dan [AWS Organizations Service Control Policies \(SCP\)](#). IAM memungkinkan Anda untuk mengelola akses secara aman ke layanan dan sumber daya AWS. Menggunakan IAM, Anda dapat mengontrol siapa yang dapat membuat atau mengelola sumber daya AWS, tipe sumber daya yang dapat dibuat, dan di mana sumber daya tersebut dapat dibuat. Hal ini meminimalkan kemungkinan sumber daya dibuat di luar kebijakan yang ditetapkan. Gunakan peran dan kelompok yang dibuat sebelumnya dan tetapkan [kebijakan IAM](#) untuk menegakkan penggunaan yang tepat. SCP menawarkan kontrol terpusat pada izin maksimum yang tersedia untuk semua akun di organisasi Anda, yang menjaga akun-akun Anda tetap berada di dalam pedoman kontrol akses Anda. SCP hanya tersedia di organisasi yang mengaktifkan semua fitur, dan Anda dapat mengonfigurasi SCP agar menolak atau mengizinkan tindakan untuk akun anggota secara default. Untuk detail selengkapnya tentang implementasi manajemen akses, lihat [laporan resmi Pilar Keamanan Well-Architected](#).

Tata kelola juga dapat diimplementasikan melalui manajemen [Kuota layanan AWS](#). Dengan memastikan kuota layanan diatur dengan biaya overhead minimum dan dipelihara secara akurat, Anda dapat meminimalkan pembuatan sumber daya di luar kebutuhan organisasi. Untuk meraih hal ini, Anda harus memahami seberapa cepat kebutuhan Anda dapat berubah, memahami proyek yang sedang berlangsung (baik pembuatan maupun penonaktifan sumber daya), dan mempertimbangkan seberapa cepat perubahan kuota dapat diimplementasikan. [Kuota layanan](#) dapat digunakan untuk meningkatkan kuota Anda saat diperlukan.

#### Langkah implementasi

- Implementasikan notifikasi pengeluaran: Menggunakan kebijakan organisasi yang Anda tetapkan, buat [AWS Budgets](#) untuk memberi tahu Anda saat pengeluaran berada di luar kebijakan Anda.

Konfigurasi beberapa anggaran biaya, satu untuk masing-masing akun, yang memberi tahu Anda tentang keseluruhan pengeluaran akun. Konfigurasi anggaran biaya tambahan di dalam masing-masing akun untuk unit yang lebih kecil di dalam akun. Unit-unit tersebut berbeda-beda tergantung struktur akun Anda. Beberapa contohnya adalah Wilayah AWS, beban kerja (menggunakan tag), atau layanan AWS. Konfigurasi daftar distribusi email sebagai penerima notifikasi, bukan akun email individu. Anda dapat mengonfigurasi anggaran riil ketika jumlah terlampaui, atau gunakan prakiraan anggaran untuk memberitahukan prakiraan penggunaan. Anda juga dapat melakukan konfigurasi Tindakan Anggaran AWS yang dapat menegakkan kebijakan IAM atau SCP, atau menghentikan instans Amazon EC2 atau Amazon RDS target. Tindakan Anggaran dapat dijalankan secara otomatis atau memerlukan persetujuan alur kerja.

- Implementasikan notifikasi pengeluaran anomali: Gunakan [AWS Cost Anomaly Detection](#) untuk mengurangi biaya tidak terduga di dalam organisasi Anda dan analisis akar masalah potensi pengeluaran anomali. Setelah Anda membuat pemantau biaya untuk mengidentifikasi pengeluaran tidak wajar dengan tingkat detail yang Anda tentukan dan mengonfigurasi notifikasi di AWS Cost Anomaly Detection, Anda akan menerima pemberitahuan ketika pengeluaran tidak wajar terdeteksi. Hal ini memungkinkan Anda untuk menganalisis akar masalah penyebab anomali tersebut dan memahami dampaknya terhadap biaya Anda. Gunakan Kategori Biaya AWS sambil melakukan konfigurasi AWS Cost Anomaly Detection untuk mengidentifikasi tim proyek atau tim unit bisnis mana yang dapat menganalisis akar masalah biaya tidak terduga dan mengambil tindakan yang diperlukan secara tepat waktu.
- Implementasikan kontrol pada penggunaan: Menggunakan kebijakan organisasi yang Anda tetapkan, implementasikan kebijakan dan peran IAM untuk menentukan tindakan apa yang dapat dilakukan oleh pengguna dan tindakan yang tidak dapat mereka lakukan. Beberapa kebijakan organisasi dapat disertakan di satu kebijakan AWS. Seperti saat Anda menetapkan kebijakan, mulailah secara umum lalu terapkan kontrol secara lebih mengerucut di masing-masing langkah. Batas layanan juga merupakan kontrol penggunaan yang efektif. Implementasikan batas layanan yang tepat pada semua akun Anda.

## Sumber daya

### Dokumen terkait:

- [Kebijakan terkelola AWS untuk fungsi tugas](#)
- [Strategi penagihan beberapa akun AWS](#)
- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [AWS Budgets](#)

- [AWS Cost Anomaly Detection](#)
- [Kontrol Biaya AWS Anda](#)

Video terkait:

- [Cara Menggunakan AWS Budgets untuk melacak pengeluaran dan penggunaan](#)

Contoh terkait:

- [Contoh kebijakan manajemen akses IAM](#)
- [Contoh kebijakan kontrol layanan](#)
- [Tindakan Anggaran AWS](#)
- [Buat Kebijakan IAM untuk mengontrol akses ke sumber daya Amazon EC2 menggunakan Tag](#)
- [Batasi akses Identitas IAM ke sumber daya Amazon EC2 tertentu](#)
- [Buat Kebijakan IAM untuk membatasi penggunaan Amazon EC2 berdasarkan kelompok](#)
- [Lab Well-Architected: Tata Kelola Biaya dan Penggunaan \(Level 100\)](#)
- [Lab Well-Architected: Tata Kelola Biaya dan Penggunaan \(Level 200\)](#)
- [Integrasi Slack untuk Cost Anomaly Detection menggunakan AWS Chatbot](#)

## COST02-BP06 Melacak siklus hidup proyek

Lacak, ukur, dan audit siklus hidup proyek, tim, dan lingkungan untuk menghindari penggunaan dan pembayaran sumber daya yang tidak perlu.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Pelacakan siklus hidup proyek secara efektif memungkinkan organisasi untuk memiliki kontrol yang lebih baik atas biaya melalui perencanaan, manajemen, dan pengoptimalan sumber daya, waktu, dan kualitas yang lebih baik. Wawasan yang diperoleh melalui pelacakan sangat berharga untuk membuat keputusan yang lebih tepat yang berperan dalam efektivitas biaya dan keberhasilan proyek secara keseluruhan.

Melacak seluruh siklus kerja beban kerja membantu Anda memahami kapan beban kerja atau komponen beban kerja tidak lagi diperlukan. Beban kerja dan komponen yang ada mungkin tampak



sedang digunakan, tetapi ketika AWS merilis layanan atau fitur baru, beban kerja dan komponen tersebut dapat dinonaktifkan atau diadopsi. Periksa tahapan beban kerja sebelumnya. Setelah beban kerja diproduksi, lingkungan sebelumnya bisa dinonaktifkan atau jauh dikurangi kapasitasnya sampai diperlukan lagi.

AWS menyediakan beberapa layanan manajemen dan tata kelola yang bisa Anda gunakan untuk melacak siklus hidup entitas. Anda dapat menggunakan [AWS Config](#) atau [AWS Systems Manager](#) untuk menyediakan inventaris sumber daya dan konfigurasi AWS Anda yang mendetail. Anda disarankan untuk mengintegrasikan proyek atau sistem manajemen aset yang sudah ada agar proyek dan produk aktif di dalam organisasi Anda tetap terlacak. Dengan menggabungkan sistem Anda saat ini dengan set peristiwa dan metrik yang kaya yang disediakan oleh AWS, Anda dapat membangun tampilan peristiwa siklus hidup yang signifikan serta secara proaktif mengelola sumber daya untuk mengurangi biaya yang tidak perlu.

Serupa dengan [Manajemen Siklus Hidup Aplikasi \(ALM\)](#), pelacakan siklus hidup proyek harus melibatkan beberapa proses, alat, dan tim yang bekerja bersama, seperti desain dan pengembangan, pengujian, produksi, dukungan, dan redundansi beban kerja.

Dengan memantau setiap fase siklus hidup proyek secara cermat, organisasi memperoleh wawasan penting dan kontrol yang lebih baik, sehingga dapat memfasilitasi perencanaan, implementasi, dan penyelesaian proyek dengan baik. Pengawasan yang cermat ini memverifikasi bahwa proyek tidak hanya memenuhi standar kualitas, tetapi juga disampaikan tepat waktu dan tidak melampaui anggaran, sehingga meningkatkan efisiensi biaya secara keseluruhan.

Untuk detail selengkapnya tentang penerapan pelacakan siklus hidup entitas, lihat [laporan resmi Pilar Keunggulan Operasional AWS Well-Architected](#).

#### Langkah implementasi

- Tetapkan proses pemantauan siklus hidup proyek: [Tim Cloud Center of Excellence](#) harus menetapkan proses pemantauan siklus hidup proyek. Tetapkan pendekatan terstruktur dan sistematis untuk memantau beban kerja agar dapat meningkatkan kontrol, visibilitas, dan performa proyek. Jadikan proses pemantauan transparan, kolaboratif, dan berfokus pada peningkatan berkelanjutan untuk memaksimalkan efektivitas dan nilainya.
- Lakukan tinjauan beban kerja: Sebagaimana ditentukan oleh kebijakan organisasi Anda, tetapkan jadwal reguler untuk mengaudit proyek-proyek yang ada dan lakukan tinjauan beban kerja. Besarnya upaya yang dilakukan untuk audit harus sebanding dengan perkiraan risiko, nilai, atau biaya bagi organisasi. Area utama yang disertakan dalam audit adalah risiko insiden atau pemadaman terhadap organisasi, nilai atau kontribusi terhadap organisasi (diukur dalam bentuk

pendapatan atau reputasi merek), biaya beban kerja (diukur dalam bentuk total biaya sumber daya dan biaya operasional), dan penggunaan beban kerja (diukur dalam bentuk jumlah hasil organisasi per unit waktu). Jika area-area ini berubah selama siklus hidup, diperlukan penyesuaian beban kerja, seperti penonaktifan penuh atau sebagian.

## Sumber daya

Dokumen terkait:

- [Guidance for Tagging on AWS](#)
- [What Is ALM \(Application Lifecycle Management\)?](#)
- [Kebijakan terkelola AWS untuk fungsi tugas](#)

Contoh terkait:

- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)

Alat terkait:

- [AWS Config](#)
- [AWS Systems Manager](#)
- [AWS Budgets](#)
- [AWS Organizations](#)
- [AWS CloudFormation](#)

## Pantau biaya dan penggunaan

Mampukan tim untuk mengambil tindakan atas biaya dan penggunaan mereka melalui visibilitas mendetail ke beban kerja. Pengoptimalan biaya dimulai dengan pemahaman terperinci akan perincian biaya dan penggunaan, kemampuan untuk membuat model dan memprediksi fitur, penggunaan, dan pengeluaran di waktu mendatang, dan implementasi mekanisme yang memadai untuk menyelaraskan biaya dan penggunaan dengan tujuan organisasi Anda. Area berikut ini diperlukan untuk memantau biaya dan penggunaan Anda:

Praktik terbaik

- [COST03-BP01 Mengonfigurasi sumber informasi yang mendetail](#)
- [COST03-BP02 Menambahkan informasi organisasi ke biaya dan penggunaan](#)
- [COST03-BP03 Mengidentifikasi kategori atribusi biaya](#)
- [COST03-BP04 Membangun metrik organisasi](#)
- [COST03-BP05 Mengonfigurasi alat manajemen penagihan dan biaya](#)
- [COST03-BP06 Mengalokasikan biaya berdasarkan metrik beban kerja](#)

## COST03-BP01 Mengonfigurasi sumber informasi yang mendetail

Konfigurasi alat manajemen biaya dan pelaporan untuk tingkat detail per jam guna memberikan informasi biaya dan penggunaan yang mendetail, sehingga memungkinkan analitik dan transparansi yang lebih mendalam. Konfigurasi beban kerja Anda guna menghasilkan atau memperoleh entri log untuk setiap hasil bisnis yang dicapai.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Dengan informasi penagihan yang mendetail seperti tingkat detail per jam di dalam alat manajemen biaya, organisasi dapat melacak pemakaian mereka dengan lebih mendetail dan mengidentifikasi beberapa penyebab kenaikan biaya. Sumber-sumber data ini menyediakan tampilan paling akurat untuk biaya dan penggunaan di seluruh organisasi Anda.

AWS Cost and Usage Report menyediakan tingkat detail harian atau per jam, tarif, biaya, dan atribut penggunaan untuk semua layanan AWS yang dikenai biaya. Semua dimensi yang memungkinkan ada di dalam CUR, termasuk: pemberian tag, lokasi, atribut sumber daya, dan ID akun.

Konfigurasi CUR Anda dengan penyesuaian berikut ini:

- Sertakan ID sumber daya
- Segarkan CUR secara otomatis
- Tingkat detail per jam
- Versioning: Timpa laporan yang ada
- Integrasi data: Athena (Format dan kompresi Parquet)

Gunakan [AWS Glue](#) untuk menyiapkan data untuk analisis, dan gunakan [Amazon Athena](#) untuk melakukan analisis data, menggunakan SQL untuk mengkueri data. Anda juga dapat menggunakan

[Amazon QuickSight](#) untuk membangun visualisasi kustom dan kompleks serta mendistribusikannya ke seluruh organisasi Anda.

### Langkah implementasi

- Konfigurasi laporan biaya dan penggunaan: Menggunakan konsol penagihan, konfigurasi setidaknya satu laporan biaya dan penggunaan. Konfigurasi laporan dengan tingkat detail per jam yang menyertakan semua pengidentifikasi dan ID sumber daya. Anda juga dapat membuat laporan lain dengan tingkat detail berbeda untuk menyediakan informasi rangkuman dengan tingkat lebih tinggi.
- Konfigurasi tingkat detail per jam di Cost Explorer: Aktifkan Per jam dan Data Tingkat Sumber Daya untuk mengakses data biaya dan penggunaan dengan perincian per jam selama 14 hari terakhir dan perincian tingkat sumber daya.
- Konfigurasi pembuatan log aplikasi: Verifikasi bahwa aplikasi Anda membuat log untuk hasil bisnis yang dicapai sehingga hasil tersebut dapat dilacak dan diukur. Pastikan tingkat detail data ini setidaknya per jam sehingga sesuai dengan data biaya dan penggunaan. Untuk detail selengkapnya tentang pencatatan log dan pemantauan, lihat [Pilar Keunggulan Operasional Well-Architected](#).

### Sumber daya

#### Dokumen terkait:

- [AWS Cost and Usage Report](#)
- [AWS Glue](#)
- [Amazon QuickSight](#)
- [Harga Manajemen Biaya AWS](#)
- [Memberi tag pada sumber daya AWS](#)
- [Menganalisis biaya dengan AWS Budgets](#)
- [Menganalisis biaya dengan Cost Explorer](#)
- [Mengelola AWS Cost and Usage Report](#)
- [Pilar Keunggulan Operasional Well-Architected](#)

#### Contoh terkait:

- [Penyiapan Akun AWS](#)
- [Tampilan Baru dan Kasus Penggunaan Umum AWS Cost Explorer](#)

## COST03-BP02 Menambahkan informasi organisasi ke biaya dan penggunaan

Tentukan skema pemberian tag berdasarkan organisasi, atribut beban kerja, dan kategori alokasi biaya agar Anda dapat memfilter dan mencari sumber daya atau memantau biaya dan penggunaan di alat manajemen biaya. Implementasikan pemberian tag yang konsisten untuk semua sumber daya jika memungkinkan berdasarkan tujuan, tim, lingkungan, atau kriteria lain yang relevan dengan bisnis Anda.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Implementasikan [pemberian tag di AWS](#) untuk menambahkan informasi organisasi ke sumber daya Anda, yang kemudian akan ditambahkan ke informasi biaya dan penggunaan. Tag adalah pasangan kunci-nilai—kunci perlu ditentukan dan harus unik di seluruh organisasi, dan nilainya unik untuk grup sumber daya. Contoh pasangan kunci-nilai adalah `Environment` sebagai kunci dan `Production` sebagai nilai. Semua sumber daya dalam lingkungan produksi pasti memiliki pasangan kunci-nilai ini. Dengan penandaan, Anda dapat melacak dan mengelola biaya dengan informasi organisasi yang relevan dan bermanfaat. Anda dapat menerapkan tag yang merepresentasikan kategori organisasi (seperti pusat biaya, nama aplikasi, proyek, atau pemilik), dan mengidentifikasi beban kerja serta karakteristik beban kerja (misalnya pengujian atau produksi) untuk mengaitkan biaya dan penggunaan di seluruh organisasi.

Saat menerapkan tag ke sumber daya AWS (misalnya instans Amazon Elastic Compute Cloud atau bucket Amazon Simple Storage Service) dan mengaktifkan tag, AWS menambahkan informasi ini ke Laporan Biaya dan Penggunaan. Anda dapat menjalankan laporan dan melakukan analisis pada sumber daya yang diberi tag dan tidak diberi tag untuk meningkatkan kepatuhan terhadap kebijakan manajemen biaya dan memastikan atribusi yang akurat.

Membuat dan mengimplementasikan standar penandaan AWS di seluruh akun organisasi memungkinkan Anda untuk mengelola lingkungan AWS yang seragam dan konsisten. Gunakan [Kebijakan Tag](#) di AWS Organizations untuk menentukan aturan terkait cara tag dapat digunakan pada sumber daya AWS di akun AWS Organizations Anda. Kebijakan Tag memungkinkan Anda mengadopsi pendekatan terstandarisasi untuk pemberian tag sumber daya AWS

[AWS Tag Editor](#) memungkinkan Anda menambahkan, menghapus, dan mengelola tag di berbagai sumber daya. Dengan Tag Editor, Anda dapat mencari sumber daya yang ingin Anda beri tag, lalu mengelola tag untuk sumber daya tersebut dalam hasil pencarian Anda.

[AWS Cost Categories](#) memungkinkan Anda menetapkan makna organisasi ke biaya tanpa memerlukan tag pada sumber daya. Anda dapat memetakan informasi biaya dan penggunaan ke struktur organisasi internal yang unik. Anda menentukan aturan kategori untuk memetakan dan mengategorikan biaya menggunakan dimensi penagihan, seperti akun dan tag. Selain penandaan, hal ini memberikan kemampuan manajemen pada tingkat yang berbeda. Anda juga dapat memetakan akun dan tag spesifik untuk beberapa proyek.

### Langkah implementasi

- Tentukan skema pemberian tag: Kumpulkan semua pemangku kepentingan dari seluruh bisnis Anda untuk menentukan skema. Hal ini umumnya melibatkan orang-orang yang memiliki peran di bidang teknis, keuangan, dan manajemen. Kumpulkan daftar tag yang wajib dimiliki oleh semua sumber daya serta tag yang sebaiknya dimiliki oleh sumber daya. Verifikasikan bahwa nama dan nilai tag konsisten di seluruh organisasi.
- Berikan tag pada sumber daya: Dengan kategori atribusi biaya yang telah ditentukan, [tempatkan tag](#) pada semua sumber daya di beban kerja berdasarkan kategori. Gunakan alat seperti CLI, Tag Editor, atau AWS Systems Manager untuk meningkatkan efisiensi.
- Implementasikan AWS Cost Categories: Anda dapat membuat [Kategori Biaya](#) tanpa mengimplementasikan pemberian tag. Kategori biaya menggunakan dimensi biaya dan penggunaan yang sudah ada. Buat aturan kategori dari skema dan implementasikan ke kategori biaya.
- Pemberian tag otomatis: Untuk memastikan Anda mempertahankan pemberian tag tingkat tinggi pada semua sumber daya, otomatiskan pemberian tag agar sumber daya diberi tag secara otomatis saat dibuat. Gunakan layanan seperti [AWS CloudFormation](#) untuk memastikan sumber daya diberi tag saat dibuat. Anda juga dapat membuat solusi kustom untuk [memberikan tag secara otomatis](#) menggunakan fungsi Lambda atau menggunakan layanan mikro yang memindai beban kerja secara berkala dan menghapus semua sumber daya yang tidak diberi tag, yang ideal untuk lingkungan pengujian dan pengembangan.
- Pantau dan jalankan laporan pemberian tag: Untuk memastikan Anda mempertahankan pemberian tag tingkat tinggi di seluruh organisasi, jalankan laporan dan pantau tag pada seluruh beban kerja Anda. Anda dapat menggunakan [AWS Cost Explorer](#) untuk melihat biaya sumber daya yang diberi tag dan tidak diberi tag atau menggunakan layanan seperti [Tag Editor](#). Tinjau secara berkala

jumlah sumber daya yang tidak diberi tag dan ambil tindakan untuk menambahkan tag hingga tingkat pemberian tag yang diinginkan tercapai.

## Sumber daya

Dokumen terkait:

- [Praktik Terbaik Pemberian Tag](#)
- [Tag Sumber Daya AWS CloudFormation](#)
- [AWS Cost Categories](#)
- [Memberikan tag pada sumber daya AWS](#)
- [Menganalisis biaya dengan AWS Budgets](#)
- [Menganalisis biaya dengan Cost Explorer](#)
- [Mengelola Laporan Biaya dan Penggunaan AWS](#)

Video terkait:

- [Bagaimana cara memberikan tag pada sumber daya AWS saya untuk membagi tagihan berdasarkan pusat biaya atau proyek](#)
- [Memberikan Tag pada Sumber Daya AWS](#)

Contoh terkait:

- [Berikan tag pada sumber daya AWS baru secara otomatis berdasarkan identitas atau peran](#)

## COST03-BP03 Mengidentifikasi kategori atribusi biaya

Identifikasi kategori organisasi seperti unit bisnis, departemen, atau proyek yang dapat digunakan untuk mengalokasikan biaya di dalam organisasi Anda ke entitas pengonsumsi internal. Gunakan kategori tersebut untuk menegaskan akuntabilitas pengeluaran, menciptakan kesadaran biaya, dan mendorong perilaku pemakaian yang efektif.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

## Panduan implementasi

Proses kategorisasi biaya sangat penting dalam penganggaran, akuntansi, pelaporan keuangan, pengambilan keputusan, benchmarking, dan manajemen proyek. Dengan mengklasifikasikan dan membuat kategori pengeluaran, tim dapat lebih memahami jenis biaya yang mereka keluarkan selama perjalanan cloud mereka, sehingga membantu tim dalam mengambil keputusan yang tepat dan mengelola anggaran secara efektif.

Akuntabilitas pengeluaran cloud sangat bermanfaat untuk menghadirkan manajemen permintaan dan biaya yang disiplin. Hasilnya adalah penghematan biaya cloud yang jauh lebih besar untuk organisasi yang mengalokasikan sebagian besar pengeluaran cloud mereka untuk unit bisnis atau tim yang memakainya. Selain itu, pengalokasian pengeluaran cloud membantu organisasi dalam mengadopsi lebih banyak praktik terbaik tata kelola cloud yang terpusat.

Bekerjasamalah dengan tim keuangan Anda atau pemangku kepentingan lain yang relevan untuk memahami persyaratan tentang bagaimana biaya harus dialokasikan di dalam organisasi Anda selama rapat koordinasi rutin Anda. Biaya beban kerja harus dialokasikan pada seluruh siklus hidup, termasuk pengembangan, pengujian, produksi, dan penonaktifan. Pahami bagaimana biaya dikenakan untuk pembelajaran, pengembangan staf, dan pencetusan ide yang berkaitan dengan organisasi. Dengan begitu, akun yang akan digunakan untuk tujuan ini dapat dialokasikan dengan tepat ke anggaran pelatihan dan pengembangan, bukan anggaran biaya IT umum.

Setelah menentukan kategori atribusi biaya dengan pemangku kepentingan di organisasi, gunakan [Kategori Biaya AWS](#) untuk mengelompokkan informasi biaya dan penggunaan Anda ke dalam kategori yang bermakna di AWS Cloud seperti biaya untuk proyek tertentu atau Akun AWS untuk departemen atau unit bisnis. Anda dapat membuat kategori kustom dan memetakan informasi biaya dan penggunaan ke dalam kategori tersebut berdasarkan aturan yang Anda tentukan menggunakan berbagai dimensi seperti akun, tag, layanan, atau jenis biaya. Setelah kategori biaya disiapkan, Anda dapat melihat informasi biaya dan penggunaan berdasarkan kategori tersebut sehingga memungkinkan organisasi Anda membuat keputusan strategis dan pembelian yang lebih baik. Kategori tersebut juga akan terlihat di AWS Cost Explorer, AWS Budgets, dan AWS Cost and Usage Report.

Misalnya, buat kategori biaya untuk unit bisnis Anda (Tim DevOps), dan dalam setiap kategori, buat beberapa aturan (aturan untuk setiap subkategori) dengan beberapa dimensi (Akun AWS, tag alokasi biaya, layanan, atau jenis biaya) berdasarkan pengelompokan yang Anda tentukan. Dengan kategori biaya, Anda dapat mengatur biaya menggunakan mesin berbasis aturan. Aturan yang Anda konfigurasi akan mengatur biaya ke dalam kategori. Dalam aturan ini, Anda dapat memfilter menggunakan beberapa dimensi untuk setiap kategori seperti Akun AWS tertentu, layanan AWS,



atau jenis biaya. Anda kemudian dapat menggunakan kategori tersebut untuk berbagai produk di konsol [AWS Billing and Cost Management dan Manajemen Biaya](#) .. Produk ini mencakup AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report, dan AWS Cost Anomaly Detection.

Misalnya, diagram berikut menunjukkan cara mengelompokkan biaya dan informasi penggunaan di organisasi Anda dengan memiliki beberapa tim (kategori biaya), beberapa lingkungan (aturan), dan setiap lingkungan yang memiliki beberapa sumber daya atau aset (dimensi).

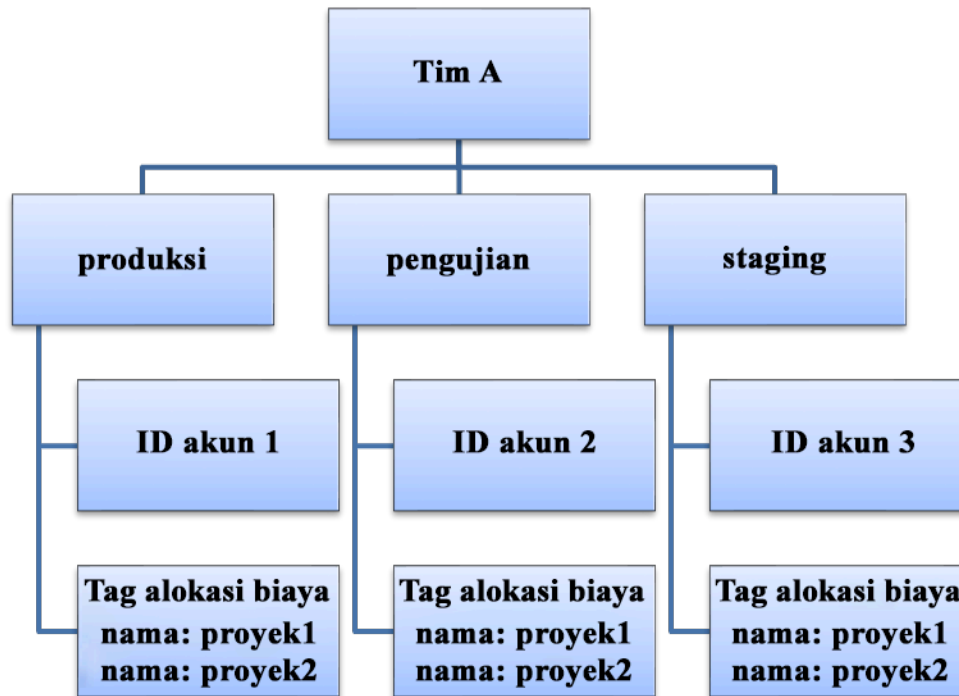


Diagram organisasi biaya dan penggunaan

Anda juga dapat membuat pengelompokan biaya menggunakan kategori biaya. Setelah Anda membuat kategori biaya (perlu waktu 24 jam setelah membuat kategori biaya agar catatan penggunaan Anda dapat diperbarui dengan nilai), kategori biaya tersebut akan muncul di [AWS Cost Explorer](#), [AWS Budgets](#), [AWS Cost and Usage Report](#), dan [AWS Cost Anomaly Detection](#). Di AWS Cost Explorer dan AWS Budgets, kategori biaya muncul dalam bentuk dimensi penagihan tambahan. Anda dapat menggunakannya untuk memfilter nilai kategori biaya tertentu atau mengelompokkan berdasarkan kategori biaya.

Langkah implementasi

- Tentukan kategori organisasi Anda: Lakukan rapat dengan para pemangku kepentingan dan unit bisnis internal untuk menentukan kategori yang sesuai dengan struktur dan persyaratan organisasi Anda. Kategori ini akan secara langsung dipetakan ke struktur kategori keuangan yang ada, seperti

unit bisnis, anggaran, pusat biaya, dan departemen. Lihat hasil yang diberikan cloud untuk bisnis Anda, seperti pelatihan dan edukasi, karena ini juga merupakan kategori organisasi.

- Tentukan kategori fungsional Anda: Lakukan rapat dengan para pemangku kepentingan dan unit bisnis internal untuk menentukan kategori yang sesuai dengan fungsi yang Anda miliki dalam bisnis Anda. Hal ini dapat berupa nama aplikasi atau beban kerja, serta jenis lingkungan, seperti produksi, pengujian, atau pengembangan.
- Menentukan Kategori Biaya AWS: Buat kategori biaya untuk mengatur informasi biaya dan penggunaan dengan menggunakan [Kategori Biaya AWS](#) dan memetakan biaya dan penggunaan AWS Anda ke [kategori yang bermakna](#). Beberapa kategori dapat ditetapkan pada satu sumber daya, dan satu sumber daya bisa masuk dalam kategori-kategori yang berlainan, jadi tentukan kategorinya sebanyak yang Anda butuhkan sehingga Anda dapat [mengelola biaya Anda](#) dalam struktur yang dikategorikan menggunakan Kategori Biaya AWS.

## Sumber daya

### Dokumen terkait:

- [Memberi tag pada sumber daya AWS](#)
- [Menggunakan Tag Alokasi Biaya](#)
- [Menganalisis biaya dengan AWS Budgets](#)
- [Menganalisis biaya dengan Cost Explorer](#)
- [Mengelola AWS Cost and Usage Report](#)
- [Kategori Biaya AWS](#)
- [Mengelola biaya Anda dengan Kategori Biaya AWS](#)
- [Membuat kategori biaya](#)
- [Memberikan tag pada kategori biaya](#)
- [Membagi biaya dalam kategori biaya](#)
- [Fitur Kategori Biaya AWS](#)

### Contoh terkait:

- [Mengatur biaya dan penggunaan Anda dengan Kategori Biaya AWS](#)
- [Mengelola biaya Anda dengan Kategori Biaya AWS](#)
- [Lab Well-Architected: Visualisasi Biaya dan Penggunaan](#)

- [Lab Well-Architected: Kategori Biaya](#)

## COST03-BP04 Membangun metrik organisasi

Bangun metrik-metrik organisasi yang diperlukan untuk beban kerja ini. Contoh metrik beban kerja adalah laporan pelanggan yang dibuat, atau halaman web yang disajikan untuk pelanggan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Pahami bagaimana output beban kerja Anda diukur berdasarkan keberhasilan bisnis. Masing-masing beban kerja umumnya memiliki satu set kecil berisi output-output utama yang mengindikasikan kinerja. Jika Anda memiliki beban kerja yang kompleks dengan banyak komponen, Anda dapat memprioritaskan daftar, atau menetapkan dan melacak metrik untuk setiap komponen. Bekerjalah dengan tim Anda untuk memahami metrik mana yang akan digunakan. Unit ini akan digunakan untuk memahami efisiensi beban kerja, atau biaya untuk masing-masing output bisnis.

### Langkah implementasi

- Tentukan hasil beban kerja: Lakukan pertemuan dengan pemangku kepentingan dalam bisnis dan tetapkan hasil untuk beban kerja. Ini adalah pengukur utama penggunaan pelanggan dan harus berupa metrik bisnis dan bukan metrik teknis. Harus ada sedikit metrik tingkat tinggi (kurang dari lima) per beban kerja. Jika beban kerja memunculkan beberapa hasil untuk kasus-kasus penggunaan yang berbeda, kelompokkan ke dalam satu metrik.
- Tentukan hasil komponen beban kerja: Opsi lainnya adalah jika Anda memiliki beban kerja besar dan kompleks, atau dapat dengan mudah mengurai beban kerja Anda ke dalam komponen (seperti layanan mikro) dengan input dan output yang ditetapkan dengan baik, tentukan metrik untuk masing-masing komponen. Upaya harus mencerminkan nilai dan biaya komponen. Mulailah dengan komponen yang paling besar menuju komponen yang lebih kecil.

### Sumber daya

Dokumen terkait:

- [Memberikan tag pada sumber daya AWS](#)
- [Menganalisis biaya dengan AWS Budgets](#)
- [Menganalisis biaya dengan Cost Explorer](#)

- [Mengelola Laporan Biaya dan Penggunaan AWS](#)

## COST03-BP05 Mengonfigurasi alat manajemen penagihan dan biaya

Konfigurasi alat manajemen biaya sesuai dengan kebijakan organisasi Anda untuk mengelola dan mengoptimalkan pengeluaran cloud. Hal ini mencakup layanan, alat, dan sumber daya untuk mengatur dan melacak data biaya dan penggunaan, mengoptimalkan kontrol melalui penagihan terkonsolidasi dan izin akses, meningkatkan perencanaan melalui penganggaran dan prakiraan, menerima notifikasi atau peringatan, serta menurunkan biaya secara lebih lanjut dengan sumber daya dan optimisasi harga.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Untuk membangun akuntabilitas yang kuat, strategi akun Anda harus dipertimbangkan terlebih dahulu sebagai bagian dari strategi alokasi biaya Anda. Jika benar di tahap ini, Anda mungkin tidak perlu berupaya lebih jauh. Jika tidak, dapat muncul ketidaksadaran dan titik-titik masalah lebih lanjut.

Untuk mendorong akuntabilitas pengeluaran cloud, pengguna harus memiliki akses ke alat yang memberikan visibilitas tentang biaya dan penggunaan mereka. Semua beban kerja dan tim sebaiknya memiliki beberapa alat yang telah dikonfigurasi untuk detail dan tujuan berikut:

- **Atur:** Menetapkan acuan dasar alokasi dan tata kelola biaya dengan strategi pemberian tag dan kategorisasi Anda sendiri.
- **Atur:** Tetapkan acuan dasar alokasi dan tata kelola biaya dengan strategi pemberian tag dan taksonomi Anda sendiri. Berikan tag pada sumber daya AWS dan kategorikan sumber daya tersebut secara bermakna berdasarkan struktur organisasi Anda (unit bisnis, departemen, atau proyek). Beri tag pada nama akun untuk pusat-pusat biaya tertentu dan petakan dengan Kategori Biaya AWS untuk mengelompokkan akun unit bisnis tertentu untuk pusat biaya mereka sehingga pemilik unit bisnis dapat melihat konsumsi beberapa akun di satu tempat.
- **Akses:** Melacak informasi penagihan di seluruh organisasi dalam [penagihan terkonsolidasi](#) dan verifikasi bahwa pemangku kepentingan yang tepat dan pemilik bisnis memiliki akses.
- **Kontrol:** Bangun mekanisme tata kelola yang efektif dengan pagar pembatas yang tepat untuk mencegah skenario tak terduga saat menggunakan Kebijakan Kontrol Layanan (SCP), kebijakan tag, dan pemberitahuan anggaran. Misalnya, Anda dapat mengizinkan tim untuk membuat sumber daya di Wilayah yang dipilih hanya dengan menggunakan mekanisme kontrol yang efektif.

- **Status Saat Ini:** Mengonfigurasi dasbor yang menampilkan tingkat biaya dan penggunaan saat ini. Dasbor harus tersedia di tempat yang mudah terlihat di dalam lingkungan kerja, mirip dengan dasbor operasi. Anda dapat menggunakan [Cloud Intelligence Dashboard \(CID\)](#) atau produk lain yang didukung untuk menciptakan visibilitas ini.
- **Notifikasi:** Berikan notifikasi saat biaya atau penggunaan melebihi batas yang ditentukan dan ketika anomali terjadi dengan AWS Budgets atau AWS Cost Anomaly Detection.
- **Laporan:** Merangkum semua informasi biaya dan penggunaan serta tingkatkan kesadaran dan akuntabilitas pengeluaran cloud Anda dengan data biaya yang mendetail dan dapat diatribusikan. Laporan harus relevan dengan tim yang memakainya dan idealnya harus berisi rekomendasi.
- **Pelacakan:** Menampilkan biaya dan penggunaan saat ini dibandingkan dengan tujuan atau target yang telah ditentukan.
- **Analisis:** Membantu anggota tim melakukan analisis kustom dan mendalam dengan detail per jam, menggunakan semua dimensi yang memungkinkan.
- **Inspeksi:** Terus memantau deployment sumber daya dan peluang optimisasi biaya Anda. Dapatkan notifikasi (menggunakan Amazon CloudWatch, Amazon SNS, atau Amazon SES) untuk deployment sumber daya di tingkat organisasi dan tinjau rekomendasi optimisasi biaya (misalnya, AWS Compute Optimizer atau AWS Trusted Advisor).
- **Tren:** Menampilkan variabilitas pada biaya dan penggunaan selama periode waktu yang diperlukan, dengan detail yang diperlukan.
- **Prakiraan:** Menampilkan perkiraan biaya mendatang, memperkirakan penggunaan sumber daya Anda, dan melakukan pengeluaran dengan dasbor prakiraan yang Anda buat.

Anda dapat menggunakan alat AWS seperti [AWS Cost Explorer](#), [AWS Billing and Cost Management](#), atau [AWS Budgets](#) untuk hal-hal inti, atau Anda dapat mengintegrasikan data CUR dengan [Amazon Athena](#) dan [Amazon QuickSight](#) untuk memberikan kemampuan ini untuk tampilan yang lebih mendetail. Jika Anda tidak memiliki keterampilan inti atau kapasitas di dalam organisasi Anda, Anda dapat bekerja sama dengan [AWS ProServ](#), [AWS Managed Services \(AMS\)](#), atau [AWS Partner](#) dan menggunakan alat-alat mereka. Anda juga dapat menggunakan alat pihak ketiga, tetapi pastikan terlebih dahulu bahwa biayanya akan memberikan nilai yang sepadan bagi organisasi Anda.

### Langkah implementasi

- **Izinkan akses berbasis tim ke alat:** Konfigurasi akun Anda dan buat grup yang memiliki akses ke laporan biaya dan penggunaan yang diperlukan untuk konsumsi mereka dan gunakan [AWS Identity and Access Management](#) untuk [mengontrol akses](#) ke alat-alat seperti AWS Cost Explorer. Grup tersebut harus menyertakan perwakilan dari semua tim yang memiliki atau mengelola sebuah

aplikasi. Hal ini bertujuan untuk memastikan setiap tim dapat mengakses informasi biaya dan penggunaan untuk melacak konsumsi mereka.

- Mengonfigurasi AWS Budgets: [Konfigurasi AWS Budgets](#) di semua akun untuk beban kerja Anda. Tetapkan anggaran untuk pengeluaran akun secara keseluruhan serta anggaran untuk beban kerja menggunakan tag. Konfigurasi notifikasi di AWS Budgets untuk mendapatkan peringatan saat Anda melebihi jumlah yang dianggarkan, atau saat perkiraan biaya Anda melebihi anggaran.
- Mengonfigurasi AWS Cost Explorer: Konfigurasi [AWS Cost Explorer](#) untuk beban kerja dan akun Anda guna memvisualisasikan data biaya untuk analisis lebih lanjut. Buat dasbor untuk beban kerja yang melacak pengeluaran secara keseluruhan, metrik penggunaan utama untuk beban kerja, dan perkiraan biaya mendatang berdasarkan data biaya historis Anda.
- Mengonfigurasi AWS Cost Anomaly Detection: Gunakan [AWS Cost Anomaly Detection](#) untuk akun, layanan inti, atau kategori biaya yang Anda buat guna memantau biaya dan penggunaan serta mendeteksi pengeluaran yang tidak biasa. Anda bisa mendapatkan peringatan satu per satu dalam laporan gabungan dan mendapatkan peringatan dalam email atau topik Amazon SNS yang memungkinkan Anda menganalisis dan menentukan akar penyebab anomali, serta mengidentifikasi faktor yang mendorong kenaikan biaya.
- Mengonfigurasi alat lanjutan: Anda memiliki opsi untuk membuat alat kustom untuk organisasi Anda guna memberikan detail dan granularitas tambahan. Anda dapat mengimplementasikan kemampuan analisis lanjutan menggunakan [Amazon Athena](#), dan dasbor menggunakan [Amazon QuickSight](#). Pertimbangkan untuk menggunakan [solusi CID](#) yang memiliki dasbor lanjutan yang telah dikonfigurasi. Tersedia juga [AWS Partner](#) yang dapat Anda ajak bekerja sama dan adopsi solusi manajemen cloud-nya untuk mengaktifkan pemantauan dan optimisasi tagihan cloud di satu lokasi yang praktis.

## Sumber daya

Dokumen terkait:

- [Manajemen Biaya AWS](#)
- [Memberi tag](#) Sumber daya AWS
- [Menganalisis biaya dengan AWS Budgets](#)
- [Menganalisis biaya dengan Cost Explorer](#)
- [Mengelola AWS Cost and Usage Report](#)
- [Kategori Biaya AWS](#)

- [Manajemen Keuangan Cloud dengan AWS](#)
- [Contoh kebijakan kontrol layanan](#)
- [Partner APN AWS – Manajemen Biaya](#)

Video terkait:

- [Melakukan Deployment Dasbor Inteligensi Cloud](#)
- [Dapatkan Peringatan tentang Metrik atau KPI FinOps atau Optimisasi Biaya](#)

Contoh terkait:

- [Lab Well-Architected - Pengaturan Akun AWS](#)
- [Lab Well-Architected: Visualisasi Penagihan](#)
- [Lab Well-Architected: Biaya dan Penggunaan Tata Kelola](#)
- [Lab Well-Architected: Analisis Biaya dan Penggunaan](#)
- [Lab Well-Architected: Visualisasi Biaya dan Penggunaan](#)
- [Lab Well-Architected: Dasbor Inteligensi Cloud](#)
- [Cara menggunakan SCP untuk mengatur pagar pembatas izin di seluruh akun](#)

## COST03-BP06 Mengalokasikan biaya berdasarkan metrik beban kerja

Alokasikan biaya beban kerja berdasarkan metrik penggunaan atau hasil bisnis untuk mengukur efisiensi biaya beban kerja. Implementasikan proses untuk menganalisis data biaya dan penggunaan, yang dapat menyediakan wawasan dan kemampuan charge back.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Optimalisasi biaya menghadirkan hasil bisnis pada titik harga terendah, yang hanya dapat dicapai dengan mengalokasikan biaya beban kerja berdasarkan metrik beban kerja (diukur berdasarkan efisiensi beban kerja). Pantau metrik beban kerja yang ditetapkan melalui file log atau pemantauan aplikasi lain. Kombinasikan data ini dengan biaya beban kerja, yang dapat diperoleh dengan melihat biaya dengan nilai tag khusus atau ID akun. Disarankan melakukan analisis ini pada level per jam. Efisiensi umumnya akan berubah jika Anda memiliki beberapa komponen biaya statis (misalnya basis

data backend yang berjalan secara permanen) dengan beragam laju permintaan (misalnya puncak penggunaan pada pukul sembilan pagi hingga lima sore, dengan sedikit permintaan pada malam hari). Memahami hubungan antara biaya statis dan variabel akan membantu Anda berfokus pada aktivitas optimalisasi Anda.

Membuat metrik beban kerja untuk sumber daya bersama mungkin lebih sulit dibandingkan dengan sumber daya seperti aplikasi dalam kontainer pada Amazon Elastic Container Service (Amazon ECS) dan Amazon API Gateway. Namun, ada beberapa cara tertentu bagi Anda untuk mengkategorikan penggunaan dan melacak biaya. Jika Anda perlu melacak sumber daya bersama Amazon ECS dan AWS Batch, Anda dapat mengaktifkan data alokasi biaya terpisah di AWS Cost Explorer. Dengan data alokasi biaya terpisah, Anda dapat memahami dan mengoptimalkan biaya serta penggunaan aplikasi dalam kontainer dan mengalokasikan biaya aplikasi kembali ke entitas bisnis masing-masing berdasarkan cara pemakaian sumber daya komputasi dan memori bersama mereka. Jika Anda memiliki penggunaan fungsi API Gateway dan AWS Lambda bersama, Anda dapat menggunakan [AWS Application Cost Profiler](#) untuk mengkategorikan konsumsi mereka berdasarkan ID Penyewa atau ID Pelanggan.

## Langkah implementasi

- Alokasikan biaya ke metrik beban kerja: Menggunakan metrik yang ditetapkan dan tag yang dikonfigurasi, buat metrik yang menggabungkan output beban kerja dan biaya beban kerja. Gunakan layanan analitik seperti Amazon Athena dan Amazon QuickSight untuk membuat dasbor efisiensi untuk beban kerja keseluruhan, dan komponen apa pun.

## Sumber daya

### Dokumen terkait:

- [Memberi tag pada sumber daya AWS](#)
- [Menganalisis biaya dengan AWS Budgets](#)
- [Menganalisis biaya dengan Cost Explorer](#)
- [Mengelola Laporan Biaya dan Penggunaan AWS](#)

### Contoh terkait:

- [Meningkatkan visibilitas biaya Amazon ECS dan AWS Batch dengan Data Alokasi Biaya Terpisah AWS](#)



## Sumber daya penonaktifan

Setelah Anda mengelola serangkaian sumber daya teknologi, karyawan, dan proyek, seiring waktu Anda akan dapat mengenali sumber daya mana yang tidak lagi digunakan, dan proyek mana yang tidak lagi memiliki pemilik.

### Praktik terbaik

- [COST04-BP01 Lacak sumber daya sepanjang masa pakainya](#)
- [COST04-BP02 Mengimplementasikan proses penonaktifan](#)
- [COST04-BP03 Menonaktifkan sumber daya](#)
- [COST04-BP04 Menonaktifkan sumber daya secara otomatis](#)
- [COST04-BP05 Menegakkan kebijakan retensi data](#)

## COST04-BP01 Lacak sumber daya sepanjang masa pakainya

Tentukan dan implementasikan metode untuk melacak sumber daya dan kaitannya dengan sistem sepanjang masa pakainya. Anda dapat menggunakan pemberian tag untuk mengidentifikasi beban kerja atau fungsi sumber daya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Nonaktifkan sumber daya beban kerja yang sudah tidak diperlukan. Contoh yang umum adalah sumber daya yang digunakan untuk pengujian: setelah pengujian selesai, sumber daya dapat dikeluarkan. Melacak sumber daya dengan tag (dan menjalankan laporan atas tag-tag tersebut) dapat membantu Anda mengidentifikasi aset untuk dinonaktifkan karena tidak digunakan atau yang lisensinya akan kedaluwarsa. Menggunakan tag merupakan cara efektif untuk melacak sumber daya, dengan memberi label sumber daya dengan fungsinya, atau tanggal kapan sumber daya dapat dinonaktifkan. Maka pelaporan dapat dijalankan atas tag ini. Contoh nilai untuk pemberian tag pada fitur adalah pengujian fitur X untuk mengidentifikasi tujuan sumber daya dalam siklus pakai beban kerja. Contoh lainnya adalah menggunakan LifeSpan atau TTL untuk sumber daya, seperti nama dan nilai kunci tag yang akan dihapus untuk menetapkan periode atau waktu tertentu untuk penonaktifan.

### Langkah implementasi

- Implementasikan skema pemberian tag: Implementasikan skema pemberian tag yang mengidentifikasi beban kerja untuk sumber daya, sambil verifikasi bahwa semua sumber daya dalam beban kerja sudah diberi tag dengan benar. Pemberian tag membantu Anda membuat kategori sumber daya berdasarkan tujuan, tim, lingkungan, atau kriteria lain yang relevan dengan bisnis Anda. Untuk detail selengkapnya tentang kasus penggunaan pemberian tag, strategi, dan teknik, lihat [Praktik Terbaik Pemberian Tag AWS](#).
- Implementasikan pemantauan throughput dan output beban kerja: Implementasikan pemantauan atau peringatan throughput beban kerja, yang dipicu oleh permintaan input atau penyelesaian output. Konfigurasi untuk memberikan notifikasi ketika permintaan beban kerja atau output menurun hingga nol, yang menandakan sumber daya beban kerja sudah tidak digunakan. Sertakan faktor waktu jika beban kerja secara berkala menurun hingga nol dalam kondisi normal. Untuk detail selengkapnya tentang sumber daya yang tidak digunakan atau kurang dimanfaatkan, lihat [Pemeriksaan Optimasi Biaya AWS Trusted Advisor](#).
- Kelompokkan sumber daya AWS: Buat grup untuk sumber daya AWS. Anda dapat menggunakan [AWS Resource Groups](#) untuk mengatur dan mengelola sumber daya AWS Anda yang berada di Wilayah AWS yang sama. Anda dapat menambahkan tag ke sebagian besar sumber daya Anda untuk membantu mengidentifikasi dan menyortir sumber daya Anda di dalam organisasi Anda. Gunakan [Tag Editor](#) untuk menambahkan tag ke sumber daya yang didukung secara massal. Pertimbangkan menggunakan [AWS Service Catalog](#) untuk membuat, mengelola, dan mendistribusikan portofolio produk-produk yang disetujui kepada pengguna akhir dan mengelola siklus hidup produk.

## Sumber daya

### Dokumen terkait:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [Pemeriksaan Optimasi Biaya AWS Trusted Advisor](#)
- [Memberikan tag pada sumber daya AWS](#)
- [Memublikasikan Metrik Kustom](#)

### Video terkait:

- [Cara mengoptimalkan biaya menggunakan AWS Trusted Advisor](#)

Contoh terkait:

- [Mengatur sumber daya AWS](#)
- [Mengoptimalkan biaya menggunakan AWS Trusted Advisor](#)

## COST04-BP02 Mengimplementasikan proses penonaktifan

Implementasikan proses untuk mengidentifikasi dan menonaktifkan sumber daya tidak terpakai

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Implementasikan proses standar di organisasi Anda untuk mengidentifikasi dan menyingkirkan sumber daya yang tidak digunakan. Proses tersebut harus menetapkan frekuensi pelaksanaan pencarian, dan proses untuk menyingkirkan sumber daya guna memverifikasi terpenuhinya semua persyaratan organisasi.

#### Langkah implementasi

- Buat dan implementasikan proses penonaktifan: Bekerjalah dengan developer dan pemilik beban kerja untuk membangun proses penonaktifan untuk beban kerja dan sumber dayanya. Proses tersebut harus mencakup metode untuk memverifikasi apakah beban kerja sedang digunakan, begitu juga dengan setiap sumber daya beban kerja. Buat detail langkah-langkah yang diperlukan untuk menonaktifkan sumber daya, yakni menghapusnya dari layanan sambil memastikan kepatuhan terhadap semua persyaratan peraturan. Semua sumber daya terkait harus disertakan, seperti lisensi atau penyimpanan terlampir. Beri tahu pemilik beban kerja bahwa proses penonaktifan telah dijalankan.

Gunakan langkah-langkah penonaktifan berikut ini untuk memandu Anda terkait hal-hal yang harus diperiksa sebagai bagian dari proses Anda:

- Identifikasi sumber daya yang akan dinonaktifkan: Identifikasi sumber daya yang layak dinonaktifkan di dalam AWS Cloud Anda. Rekam semua informasi yang diperlukan dan jadwalkan penonaktifan. Dalam lini waktu Anda, pastikan untuk mempertimbangkan apakah (dan kapan) masalah yang tidak terduga muncul selama proses.
- Lakukan koordinasi dan komunikasi: Bekerjalah dengan pemilik beban kerja untuk mengonfirmasikan sumber daya yang akan dinonaktifkan.

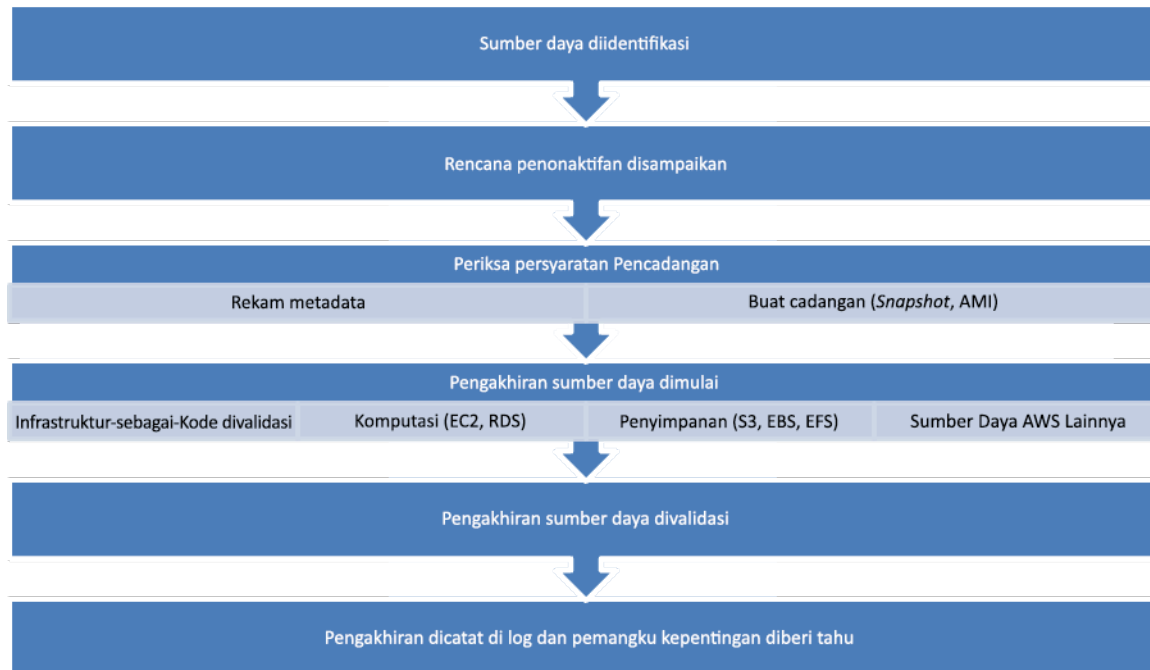
- Rekam metadata dan buat cadangan: Rekam metadata (seperti IP publik, Wilayah, AZ, VPC, Subnet, dan Grup Keamanan) dan buat cadangan (seperti snapshot Amazon Elastic Block Store atau mengambil AMI, laporan kunci, dan ekspor Sertifikat) jika diperlukan untuk sumber daya di dalam lingkungan produksi atau jika sumber daya tersebut sangat penting.
- Validasikan infrastruktur sebagai kode: Tentukan apakah sumber daya di-deploy dengan AWS CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK), atau alat deployment infrastruktur sebagai kode lainnya sehingga dapat di-deploy ulang jika perlu.
- Cegah akses: Terapkan kontrol restriktif selama jangka waktu tertentu, untuk mencegah penggunaan sumber daya ketika Anda menentukan apakah sumber daya tersebut diperlukan. Verifikasi bahwa lingkungan sumber daya dapat dikembalikan ke status aslinya jika diperlukan.
- Ikuti proses penonaktifan internal Anda: Ikuti tugas-tugas administratif dan proses penonaktifan organisasi Anda, seperti menghilangkan sumber daya dari domain organisasi Anda, menghilangkan catatan DNS, dan menghilangkan sumber daya dari alat manajemen konfigurasi Anda, alat pemantauan, alat otomatisasi, dan alat keamanan.

Jika sumber daya adalah instans Amazon EC2, pelajari daftar berikut ini. [Untuk detail selengkapnya, lihat Bagaimana cara menghapus atau mengakhiri sumber daya Amazon EC2 saya?](#)

- Hentikan atau akhiri semua instans dan penyeimbang beban Amazon EC2 Anda. Instans Amazon EC2 dapat dilihat di konsol dalam waktu singkat setelah diakhiri. Anda tidak menerima tagihan untuk instans apa pun yang tidak memiliki status berjalan
- Hapus infrastruktur Auto Scaling Anda.
- Lepaskan semua Host Khusus.
- Hapus semua volume Amazon EBS dan snapshot Amazon EBS.
- Lepaskan semua Alamat IP elastis.
- Batalkan pendaftaran semua Amazon Machine Image (AMI).
- Akhiri lingkungan AWS Elastic Beanstalk Anda.

Jika sumber daya adalah objek di dalam penyimpanan Amazon S3 Glacier dan jika Anda menghapus sebuah arsip sebelum memenuhi durasi penyimpanan minimum, Anda akan dikenakan biaya penghapusan dini prorata. Durasi penyimpanan minimum Amazon S3 Glacier bergantung pada kelas penyimpanan yang digunakan. Untuk ringkasan durasi penyimpanan minimum untuk setiap kelas penyimpanan, lihat [Kinerja di kelas-kelas penyimpanan Amazon S3](#). Untuk detail tentang penghitungan biaya penghapusan dini, lihat [harga Amazon S3](#).

Bagan alur proses penonaktifan sederhana berikut ini menguraikan langkah-langkah penonaktifan. Sebelum menonaktifkan sumber daya, verifikasi bahwa sumber daya yang telah Anda identifikasi untuk dinonaktifkan tidak sedang digunakan oleh organisasi.



Alur penonaktifan sumber daya.

## Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [AWS CloudTrail](#)

Video terkait:

- [Hapus tumpukan CloudFormation tetapi pertahankan beberapa sumber daya](#)
- [Temukan pengguna mana yang meluncurkan instans Amazon EC2](#)

Contoh terkait:

- [Hapus atau akhiri sumber daya Amazon EC2](#)
- [Temukan pengguna mana yang meluncurkan instans Amazon EC2](#)

## COST04-BP03 Menonaktifkan sumber daya

Nonaktifkan sumber daya yang dipicu oleh peristiwa seperti audit berkala, atau perubahan penggunaan. Penonaktifan umumnya dilakukan secara berkala dan dapat dilakukan secara manual atau otomatis.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Frekuensi dan upaya untuk mencari sumber daya yang tidak digunakan harus mencerminkan potensi penghematan, sehingga akun dengan biaya kecil harus dianalisis lebih jarang daripada akun dengan biaya yang lebih besar. Pencarian dan peristiwa penonaktifan bisa dipicu oleh perubahan status pada beban kerja, seperti produk yang mendekati akhir masa pakai atau mengalami penggantian. Pencarian dan peristiwa penonaktifan mungkin juga dipicu oleh peristiwa eksternal, seperti perubahan kondisi pasar atau penghentian produk.

### Langkah implementasi

- Nonaktifkan sumber daya: Ini adalah tahap depresiasi sumber daya AWS yang sudah tidak diperlukan atau mengakhiri perjanjian lisensi. Selesaikan semua pemeriksaan akhir sebelum beralih ke tahap penghapusan dan menonaktifkan sumber daya untuk mencegah gangguan yang tidak diinginkan seperti mengambil snapshot atau cadangan. Nonaktifkan setiap sumber daya yang telah teridentifikasi tidak terpakai menggunakan proses penonaktifan.

### Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)

Contoh terkait:

- [Lab Well-Architected: Menonaktifkan sumber daya \(Level 100\)](#)

## COST04-BP04 Menonaktifkan sumber daya secara otomatis

Rancang beban kerja Anda agar menangani pengakhiran sumber daya secara anggun ketika Anda mengidentifikasi dan menonaktifkan sumber daya non-kritis, sumber daya yang tidak diperlukan, atau sumber daya dengan pemanfaatan yang rendah.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Gunakan otomatisasi untuk mengurangi atau menyingkirkan biaya terkait untuk proses penonaktifan. Dengan merancang beban kerja agar menjalankan penonaktifan otomatis, Anda akan mengurangi biaya beban kerja secara keseluruhan selama masa pakainya. Anda dapat menggunakan [AWS Auto Scaling](#) untuk melakukan proses penonaktifan. Anda juga dapat mengimplementasikan kode kustom menggunakan [API atau SDK](#) untuk menonaktifkan sumber daya beban kerja secara otomatis.

[Aplikasi modern](#) dibangun dengan prioritas nirserver, yakni strategi yang mengutamakan adopsi layanan nirserver. AWS mengembangkan [layanan nirserver](#) untuk ketiga lapisan tumpukan Anda: komputasi, integrasi, dan tempat penyimpanan data. Menggunakan arsitektur nirserver, Anda dapat menghemat biaya selama periode lalu lintas rendah dengan menaikkan dan menurunkan skala secara otomatis.

### Langkah implementasi

- Implementasikan AWS Auto Scaling: Untuk sumber daya yang didukung, konfigurasi dengan [AWS Auto Scaling](#). AWS Auto Scaling dapat membantu Anda mengoptimalkan pemanfaatan dan efisiensi biaya Anda saat memakai layanan AWS. Ketika permintaan menurun, AWS Auto Scaling akan menghapus kelebihan kapasitas sumber daya secara otomatis sehingga Anda dapat terhindar dari pengeluaran yang berlebihan.
- Konfigurasi CloudWatch untuk mengakhiri instans: Instans dapat dikonfigurasi agar berakhir menggunakan [alarm CloudWatch](#). Menggunakan metrik dari proses penonaktifan, implementasikan alarm dengan tindakan Amazon Elastic Compute Cloud. Verifikasi operasi di lingkungan non-produksi sebelum peluncuran.
- Implementasikan kode di dalam beban kerja: Anda dapat menggunakan AWS SDK atau AWS CLI untuk menonaktifkan sumber daya beban kerja. Implementasikan kode di dalam aplikasi yang terintegrasi dengan AWS dan akhiri atau hapus sumber daya yang sudah tidak digunakan.
- Gunakan layanan nirserver: Prioritaskan pembangunan [arsitektur nirserver](#) dan [arsitektur berbasis peristiwa](#) di AWS untuk membangun dan menjalankan aplikasi Anda. AWS menawarkan beberapa

layanan teknologi nirserver yang secara bawaan menyediakan pemanfaatan sumber daya yang dioptimalkan secara otomatis dan penonaktifan otomatis (penskalaan ke dalam dan penskalaan ke luar). Dengan aplikasi nirserver, pemanfaatan sumber daya dioptimalkan secara otomatis dan Anda tidak pernah membayar pengadaan yang berlebihan.

## Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [Nirserver di AWS](#)
- [Membuat Alarm untuk Menghentikan, Mengakhiri, Mem-boot Ulang, atau Memulihkan Instans](#)
- [Mulai Menggunakan Amazon EC2 Auto Scaling](#)
- [Menambahkan tindakan mengakhiri ke alarm Amazon CloudWatch](#)

Contoh terkait:

- [Menjadwalkan penghapusan otomatis tumpukan AWS CloudFormation](#)
- [Lab Well-Architected – Nonaktifkan sumber daya secara otomatis \(Level 100\)](#)
- [Servian AWS Auto Cleanup](#)

## COST04-BP05 Menegakkan kebijakan retensi data

Tetapkan kebijakan retensi data pada sumber daya yang didukung untuk menangani penghapusan objek sesuai persyaratan organisasi Anda. Identifikasi dan hapus sumber daya yang tidak diperlukan atau tidak digunakan dan objek yang sudah tidak diperlukan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

Gunakan kebijakan retensi data dan kebijakan siklus hidup guna mengurangi biaya terkait untuk proses penonaktifan dan biaya penyimpanan untuk sumber daya yang diidentifikasi. Penetapan kebijakan retensi data dan kebijakan siklus hidup untuk menjalankan migrasi dan penghapusan kelas penyimpanan otomatis akan mengurangi keseluruhan biaya penyimpanan di sepanjang masa pakainya. Anda dapat menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan pembuatan dan penghapusan snapshot Amazon Elastic Block Store dan Amazon Machine Image



(AMI) yang didukung Amazon EBS, dan menggunakan Amazon S3 Intelligent-Tiering atau konfigurasi siklus hidup Amazon S3 untuk mengelola siklus hidup objek Amazon S3 Anda. Anda juga dapat mengimplementasikan kode kustom menggunakan [API atau SDK](#) untuk menciptakan kebijakan siklus hidup dan aturan kebijakan untuk objek yang akan dihapus secara otomatis.

### Langkah implementasi

- Gunakan Amazon Data Lifecycle Manager: Gunakan kebijakan siklus hidup pada Amazon Data Lifecycle Manager untuk mengotomatiskan penghapusan snapshot Amazon EBS dan AMI yang didukung Amazon EBS.
- Atur konfigurasi siklus hidup pada bucket: Gunakan konfigurasi siklus hidup Amazon S3 pada bucket untuk menetapkan tindakan bagi Amazon S3 yang akan dijalankan selama siklus hidup suatu objek, serta penghapusan pada akhir siklus hidup objek tersebut, berdasarkan persyaratan bisnis Anda.

### Sumber daya

#### Dokumen terkait:

- [AWS Trusted Advisor](#)
- [Amazon Data Lifecycle Manager](#)
- [Cara mengatur konfigurasi siklus hidup pada bucket Amazon S3](#)

#### Video terkait:

- [Mengotomatiskan Snapshot Amazon EBS dengan Amazon Data Lifecycle Manager](#)
- [Mengosongkan bucket Amazon S3 menggunakan aturan konfigurasi siklus hidup](#)

#### Contoh terkait:

- [Mengosongkan bucket Amazon S3 menggunakan aturan konfigurasi siklus hidup](#)
- [Lab Well-Architected: Menonaktifkan sumber daya secara otomatis \(Level 100\)](#)

# Sumber daya hemat biaya

Penggunaan layanan, sumber daya, dan konfigurasi yang tepat untuk beban kerja Anda adalah kunci penghematan biaya. Pertimbangkan hal-hal berikut saat membuat sumber daya yang hemat biaya:

Anda dapat menggunakan Arsitek Solusi AWS, Solusi AWS, Arsitektur Rujukan AWS, dan Partner APN untuk membantu Anda memilih arsitektur berdasarkan apa yang telah Anda pelajari.

## Topik

- [Evaluasi biaya saat memilih layanan](#)
- [Pilih tipe, ukuran, dan jumlah sumber daya yang tepat](#)
- [Memilih model harga terbaik](#)
- [Merencanakan transfer data](#)

# Evaluasi biaya saat memilih layanan

## Praktik terbaik

- [COST05-BP01 Identifikasi persyaratan organisasi untuk biaya](#)
- [COST05-BP02 Menganalisis semua komponen beban kerja](#)
- [COST05-BP03 Menjalankan analisis menyeluruh setiap komponen](#)
- [COST05-BP04 Memilih perangkat lunak dengan lisensi hemat biaya](#)
- [COST05-BP05 Memilih komponen beban kerja ini untuk mengoptimalkan biaya selaras dengan prioritas organisasi](#)
- [COST05-BP06 Melakukan analisis biaya untuk penggunaan berbeda seiring waktu](#)

## COST05-BP01 Identifikasi persyaratan organisasi untuk biaya

Bekerja dengan anggota tim untuk menentukan keseimbangan antara pengoptimalan biaya dan pilar lainnya, seperti keandalan dan performa, untuk beban kerja ini.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

## Panduan implementasi

Di sebagian besar organisasi, departemen teknologi informasi (IT) terdiri dari beberapa tim kecil, masing-masing dengan agenda dan area fokusnya sendiri, yang mencerminkan spesialisasi dan keterampilan anggota timnya. Anda perlu memahami tujuan keseluruhan, prioritas, dan sasaran organisasi Anda, serta bagaimana setiap departemen atau proyek berkontribusi terhadap tujuan ini. Mengategorikan semua sumber daya penting, termasuk personel, peralatan, teknologi, bahan, dan layanan eksternal, sangat penting untuk mencapai tujuan organisasi dan perencanaan anggaran yang komprehensif. Mengadopsi pendekatan sistematis terhadap identifikasi dan pemahaman biaya ini merupakan hal yang sangat penting untuk menyusun rencana biaya yang realistis dan matang untuk organisasi.

Ketika memilih layanan untuk beban kerja Anda, penting bagi Anda untuk memahami prioritas organisasi Anda. Ciptakan keseimbangan antara pengoptimalan biaya dan pilar Kerangka Kerja AWS Well-Architected lainnya, seperti performa dan keandalan. Proses ini harus dilakukan secara sistematis dan teratur untuk mencerminkan perubahan dalam tujuan organisasi, kondisi pasar, dan dinamika operasional. Beban kerja yang biayanya dioptimalkan penuh adalah solusi yang paling selaras dengan persyaratan organisasi Anda, tidak selalu berarti biaya yang paling rendah. Bertemulah dengan semua tim dalam organisasi Anda, seperti tim produk, bisnis, teknis, dan keuangan, untuk mengumpulkan informasi. Evaluasi dampak kompromi antar kepentingan yang bertentangan atau pendekatan alternatif, untuk membantu mengambil keputusan yang lebih tepat saat menentukan ke mana upaya perlu difokuskan atau saat memilih rencana tindakan.

Misalnya, meningkatkan kecepatan masuk pasar untuk fitur baru dapat diprioritaskan daripada optimalisasi biaya, atau Anda bisa memilih basis data relasional untuk data non-relasional guna menyederhanakan upaya migrasi sistem, dibandingkan bermigrasi ke basis data yang dioptimalkan untuk tipe data Anda dan memperbarui aplikasi Anda.

### Langkah implementasi

- Identifikasi persyaratan organisasi untuk biaya: Bertemulah dengan anggota-anggota tim dari organisasi Anda, termasuk yang ada di pengelolaan produk, pemilik aplikasi, tim pengembangan dan operasional, serta peran manajemen dan keuangan. Prioritaskan pilar Well-Architected untuk beban kerja ini dan komponennya. Output-nya harus berupa daftar pilar secara berurutan. Anda juga dapat menambahkan bobot pada masing-masing pilar untuk menunjukkan berapa fokus tambahan yang dimiliki sebuah pilar, atau seberapa serupa fokus antara dua pilar.
- Atasi dan dokumentasikan utang teknis: Selama tinjauan beban kerja, atasi utang teknis. Dokumentasikan item backlog untuk meninjau kembali beban kerja pada masa mendatang,

dengan tujuan memfaktor ulang atau merancang ulang untuk mengoptimalkannya lebih lanjut. Sangat penting untuk secara jelas mengkomunikasikan kompromi yang dilakukan kepada pemangku kepentingan lainnya.

## Sumber daya

Praktik Terbaik Terkait:

- [REL11-BP07 Merancang produk Anda agar memenuhi target ketersediaan dan perjanjian tingkat layanan \(SLA\) waktu aktif](#)
- [OPS01-BP06 Mengevaluasi kompromi](#)

Dokumen terkait:

- [Kalkulator Total Biaya Kepemilikan \(TCO\) AWS](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)

## COST05-BP02 Menganalisis semua komponen beban kerja

Verifikasi bahwa setiap beban kerja telah dianalisis, terlepas dari ukuran atau biaya saat ini. Upaya peninjauan harus menggambarkan manfaat potensial, seperti biaya saat ini dan yang diperkirakan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Komponen beban kerja, yang dirancang untuk memberikan nilai bisnis kepada organisasi, dapat mencakup berbagai layanan. Untuk setiap komponen, Anda mungkin memilih layanan AWS Cloud tertentu guna memenuhi kebutuhan bisnis. Pilihan ini dapat dipengaruhi oleh faktor-faktor seperti pemahaman atau pengalaman sebelumnya dalam menggunakan layanan tersebut.

Setelah mengidentifikasi persyaratan organisasi Anda (seperti yang disebutkan dalam [COST05-BP01 Identifikasi persyaratan organisasi untuk biaya](#)), lakukan analisis menyeluruh pada semua komponen dalam beban kerja Anda. Analisis setiap komponen dengan mempertimbangkan biaya dan ukuran saat ini serta proyeksinya. Pertimbangkan biaya analisis terhadap potensi penghematan beban kerja selama siklus hidupnya. Upaya yang dikeluarkan untuk menganalisis semua komponen beban kerja

ini harus sesuai dengan potensi penghematan atau peningkatan yang diantisipasi dari pengoptimalan komponen spesifik tersebut. Misalnya, apabila biaya sumber daya yang diajukan adalah 10 USD per bulan, dan dalam prakiraan beban tidak akan melebihi 15 USD per bulan, mengerahkan usaha sehari-hari penuh untuk mengurangi biaya hingga 50% (lima dolar per bulan) dapat melampaui manfaat potensial selama masa pakai sistem. Menggunakan perkiraan berdasarkan data yang lebih cepat dan efisien akan memberikan hasil terbaik secara keseluruhan untuk komponen ini.

Beban kerja dapat berubah seiring waktu, dan rangkaian layanan yang tepat dapat menjadi tidak optimal jika penggunaan atau arsitektur beban kerja berubah. Analisis pilihan layanan harus menggabungkan tingkat penggunaan dan status beban kerja saat ini serta di masa mendatang. Mengimplementasikan layanan untuk penggunaan atau status beban kerja di masa mendatang dapat menghemat biaya keseluruhan dengan meminimalkan atau tanpa memerlukan usaha untuk membuat perubahan di masa mendatang. Misalnya, menggunakan Amazon EMR Serverless mungkin pada awalnya merupakan pilihan yang tepat. Namun, karena konsumsi untuk layanan tersebut meningkat, transisi ke Amazon EMR di Amazon EC2 dapat mengurangi biaya untuk komponen beban kerja tersebut.

Tinjauan strategis dari semua komponen beban kerja, terlepas dari atributnya saat ini, berpotensi menghasilkan peningkatan penting dan penghematan keuangan seiring waktu. Upaya yang dicurahkan dalam proses tinjauan ini harus terencana, dengan pertimbangan yang cermat terhadap potensi keuntungan yang mungkin direalisasikan.

[AWS Cost Explorer](#) dan [AWS Cost and Usage Report](#) (CUR) dapat menganalisis biaya Bukti Konsep (PoC) atau lingkungan yang sedang berjalan. Anda juga dapat menggunakan [AWS Pricing Calculator](#) untuk memperkirakan biaya beban kerja.

### Langkah implementasi

- **Buat daftar komponen beban kerja:** Buat daftar komponen beban kerja Anda. Ini digunakan sebagai verifikasi untuk memastikan bahwa setiap komponen telah dianalisis. Upaya yang dilakukan harus sesuai dengan kekritisitas beban kerja sesuai prioritas organisasi Anda. Mengelompokkan sumber daya menurut fungsinya dapat meningkatkan efisiensi (misalnya penyimpanan basis data produksi, jika terdapat beberapa basis data).
- **Prioritaskan daftar komponen:** Ambil daftar komponen dan prioritaskan sesuai urutan upaya. Ini biasanya diurutkan berdasarkan pada biaya komponen, dari yang paling mahal ke yang paling murah, atau diurutkan sesuai kekritisitas sebagaimana ditentukan oleh prioritas organisasi Anda.
- **Lakukan analisis:** Untuk setiap komponen di dalam daftar, tinjau opsi dan layanan yang tersedia, kemudian pilih opsi yang paling sesuai dengan prioritas organisasi Anda.

## Sumber daya

Dokumen terkait:

- [AWS Pricing Calculator](#)
- [AWS Cost Explorer](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)

## COST05-BP03 Menjalankan analisis menyeluruh setiap komponen

Lihat keseluruhan biaya organisasi dari setiap komponen. Hitung total biaya kepemilikan dengan mempertimbangkan faktor biaya operasi dan manajemen, terutama jika menggunakan layanan terkelola oleh penyedia cloud. Hasil upaya peninjauan harus menggambarkan manfaat potensial (misalnya, waktu yang digunakan untuk menganalisis sebanding dengan biaya komponen).

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Pertimbangkan waktu yang dapat dihemat yang memungkinkan tim Anda untuk berfokus pada penghentian utang teknis, inovasi, fitur yang menambah nilai, dan membangun sesuatu yang membedakan bisnis Anda dari yang lain. Misalnya, Anda mungkin perlu mengangkat dan menggeser (juga disebut host ulang) basis data Anda dari lingkungan on-premise Anda ke cloud secepat mungkin dan kemudian mengoptimalkannya. Sebaiknya cari tahu kemungkinan penghematan yang diperoleh menggunakan layanan terkelola di AWS yang dapat menghilangkan atau mengurangi biaya lisensi. Layanan terkelola di AWS menghilangkan beban administratif dan operasional pemeliharaan layanan, seperti patching atau upgrading OS, dan memungkinkan Anda untuk berfokus pada inovasi.

Karena layanan terkelola beroperasi dengan skala cloud, biaya yang ditawarkan per transaksi atau layanan dapat lebih rendah. Anda dapat melakukan optimalisasi potensial untuk mencapai beberapa manfaat nyata, tanpa mengubah arsitektur inti aplikasi. Sebagai contoh, Anda mungkin berusaha mengurangi waktu yang digunakan untuk mengelola instans basis data dengan cara bermigrasi ke platform basis data sebagai layanan seperti [Amazon Relational Database Service \(Amazon RDS\)](#) atau memigrasikan aplikasi Anda ke platform yang terkelola penuh seperti [AWS Elastic Beanstalk](#).

Biasanya, layanan terkelola memiliki atribut yang dapat Anda atur untuk memastikan kapasitas yang memadai. Anda harus mengatur dan memantau atribut ini agar kapasitas Anda yang berlebih

diminimalkan dan kinerja dimaksimalkan. Anda dapat mengubah atribut AWS Managed Services menggunakan AWS Management Console atau API dan SDK AWS untuk menyelaraskan kebutuhan sumber daya dengan perubahan permintaan. Misalnya, Anda dapat menambah atau mengurangi jumlah simpul di kluster Amazon EMR (atau kluster Amazon Redshift) untuk menskalakan ke luar atau ke dalam.

Anda juga dapat mengemas beberapa instans dalam sumber daya AWS untuk memungkinkan penggunaan densitas yang lebih tinggi. Misalnya, Anda dapat menyiapkan beberapa basis data kecil dalam satu instans basis data Amazon Relational Database Service (Amazon RDS). Seiring dengan meningkatnya penggunaan, Anda dapat memigrasikan satu dari beberapa basis data ke instans basis data khusus Amazon RDS menggunakan proses pemulihan dan snapshot.

Ketika menyiapkan beban kerja dalam layanan terkelola, Anda harus memahami persyaratan untuk menyesuaikan kapasitas layanan. Persyaratan ini biasanya berupa waktu, upaya, dan dampak apa pun terhadap operasi beban kerja normal. Sumber daya yang disiapkan harus memberikan waktu untuk perubahan apa pun, siapkan biaya tambahan yang diperlukan untuk melakukan hal ini. Upaya kontinu yang diperlukan untuk mengubah layanan dapat dikurangi hingga menjadi hampir nol menggunakan API dan SDK yang diintegrasikan dengan alat pemantauan dan sistem, seperti Amazon CloudWatch.

[Amazon RDS](#), [Amazon Redshift](#), dan [Amazon ElastiCache](#) menyediakan layanan basis data terkelola. [Amazon Athena](#), [Amazon EMR](#), dan [Amazon OpenSearch Service](#) menyediakan layanan analitik terkelola.

[AMS](#) adalah layanan yang mengoperasikan infrastruktur AWS atas nama partner dan pelanggan perusahaan. Layanan ini menyediakan lingkungan yang aman dan patuh sebagai tempat deployment beban kerja Anda. AMS menggunakan model operasi cloud perusahaan dengan otomatisasi untuk memungkinkan Anda memenuhi persyaratan perusahaan, memindahkan ke cloud dengan lebih cepat, serta mengurangi biaya untuk manajemen berkelanjutan.

### Langkah implementasi

- Jalankan analisis yang menyeluruh: Menggunakan daftar komponen, kerjakan setiap komponen mulai dari prioritas tertinggi ke yang terendah. Untuk komponen yang diprioritaskan dan membutuhkan biaya mahal, jalankan analisis tambahan dan evaluasi semua opsi yang ada serta dampaknya dalam jangka panjang. Untuk komponen dengan prioritas rendah, ukur apakah perubahan penggunaan akan mengubah prioritas komponen, kemudian jalankan analisis upaya yang sesuai.

- Bandingkan sumber daya terkelola dan tidak terkelola: Pertimbangkan biaya operasional untuk sumber daya yang Anda kelola dan bandingkan dengan sumber daya terkelola AWS. Misalnya, tinjau basis data Anda yang berjalan di instans Amazon EC2 dan bandingkan dengan opsi-opsi Amazon RDS (layanan terkelola AWS) atau Amazon EMR dibandingkan dengan menjalankan Apache Spark di Amazon EC2. Saat beralih dari beban kerja yang dikelola mandiri ke beban kerja terkelola penuh AWS, pelajari opsi-opsi Anda secara cermat. Tiga faktor paling penting untuk dipertimbangkan adalah [tipe layanan terkelola](#) yang ingin Anda gunakan, proses yang akan Anda gunakan untuk [memigrasikan data Anda](#) dan memahami [model tanggung jawab bersama AWS](#).

## Sumber daya

### Dokumen terkait:

- [Kalkulator Total Biaya Kepemilikan \(TCO\) AWS](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk AWS Cloud](#)
- [Model Tanggung Jawab Bersama AWS](#)

### Video terkait:

- [Mengapa beralih ke basis data terkelola?](#)
- [Apa itu Amazon EMR dan bagaimana cara menggunakannya untuk memproses data?](#)

### Contoh terkait:

- [Alasan beralih ke basis data terkelola](#)
- [Konsolidasikan data dari beberapa basis data SQL Server yang identik ke dalam satu basis data Amazon RDS for SQL Server tunggal menggunakan AWS DMS](#)
- [Kirimkan data dalam skala besar ke Amazon Managed Streaming for Apache Kafka \(Amazon MSK\)](#)
- [Memigrasikan aplikasi web ASP.NET ke AWS Elastic Beanstalk](#)



## COST05-BP04 Memilih perangkat lunak dengan lisensi hemat biaya

Perangkat lunak sumber terbuka meniadakan biaya lisensi perangkat lunak yang dapat menambah biaya yang besar pada beban kerja. Ketika perangkat lunak berlisensi diperlukan, hindari lisensi yang terikat ke atribut arbitrer seperti CPU, carilah lisensi yang terikat dengan output atau hasil. Besar kecilnya biaya lisensi ini lebih sesuai dengan manfaat yang disediakan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Sumber terbuka berasal dari konteks pengembangan perangkat lunak untuk menunjukkan bahwa perangkat lunak tersebut memenuhi kriteria distribusi gratis tertentu. Perangkat lunak sumber terbuka terdiri dari kode sumber yang dapat diperiksa, dimodifikasi, dan disempurnakan oleh siapa pun. Berdasarkan persyaratan bisnis, keterampilan rekayasawan, perkiraan penggunaan, atau ketergantungan teknologi lainnya, organisasi dapat mempertimbangkan untuk menggunakan perangkat lunak sumber terbuka di AWS untuk meminimalkan biaya lisensi mereka. Dengan kata lain, biaya lisensi perangkat lunak dapat dikurangi melalui penggunaan [perangkat lunak sumber terbuka](#). Penggunaan jenis perangkat lunak ini dapat memberikan pengaruh besar pada biaya beban kerja seiring berubahnya ukuran beban kerja.

Ukur manfaat perangkat lunak berlisensi terhadap total biaya untuk mengoptimalkan beban kerja Anda. Modelkan perubahan dalam lisensi dan bagaimana pengaruhnya terhadap biaya beban kerja Anda. Jika vendor mengubah biaya lisensi basis data Anda, selidiki bagaimana pengaruhnya terhadap keseluruhan efisiensi beban kerja Anda. Pertimbangkan riwayat pengumuman harga dari vendor Anda untuk mengetahui tren perubahan lisensi di seluruh produk mereka. Biaya lisensi juga dapat berubah terlepas dari throughput atau penggunaan, seperti lisensi yang berubah berdasarkan perangkat keras (lisensi terikat CPU). Lisensi jenis ini harus dihindari karena biaya dapat meningkat pesat tanpa hasil yang seimbang.

Misalnya, mengoperasikan instans Amazon EC2 di us-east-1 dengan sistem operasi Linux memungkinkan Anda memangkas biaya sekitar 45%, dibandingkan dengan menjalankan instans Amazon EC2 lain yang berjalan di Windows.

[AWS Pricing Calculator](#) menawarkan cara komprehensif untuk membandingkan biaya berbagai sumber daya dengan opsi lisensi yang berbeda, seperti instans Amazon RDS dan mesin basis data yang berbeda. Selain itu, AWS Cost Explorer memberikan perspektif yang tak ternilai untuk biaya beban kerja yang ada, terutama beban kerja dengan lisensi yang berbeda. Untuk manajemen lisensi, [AWS License Manager](#) menawarkan metode yang efisien untuk mengawasi dan menangani lisensi

perangkat lunak. Pelanggan dapat menerapkan dan mengoperasikan perangkat lunak sumber terbuka pilihan mereka di AWS Cloud.

### Langkah implementasi

- Analisis opsi lisensi: Tinjau ketentuan lisensi perangkat lunak yang tersedia. Cari versi sumber terbuka yang memiliki fungsionalitas yang diperlukan, dan cari tahu apakah manfaat dari perangkat lunak berlisensi lebih besar daripada biayanya. Ketentuan yang menguntungkan adalah yang menyelaraskan biaya dengan manfaat yang disediakan.
- Analisis penyedia perangkat lunak: Tinjau harga historis atau perubahan lisensi dari vendor. Cari perubahan yang tidak selaras dengan hasil, seperti ketentuan merugikan yang mengharuskan perangkat lunak dijalankan di perangkat keras atau platform vendor tertentu. Selain itu, cari tahu bagaimana mereka melakukan audit, dan sanksi yang dapat dikenakan.

### Sumber daya

#### Dokumen terkait:

- [Open Source at AWS](#)
- [Kalkulator Total Biaya Kepemilikan \(TCO\) AWS](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)

#### Contoh terkait:

- [Blog Sumber Terbuka](#)
- [Blog Sumber Terbuka AWS](#)
- [Penilaian Perizinan dan Optimisasi](#)

## COST05-BP05 Memilih komponen beban kerja ini untuk mengoptimalkan biaya selaras dengan prioritas organisasi

Pertimbangkan biaya saat memilih semua komponen untuk beban kerja Anda. Termasuk di antaranya adalah menggunakan layanan terkelola dan tingkat aplikasi atau nirserver, kontainer, atau arsitektur yang didorong peristiwa agar dapat menekan keseluruhan biaya. Minimalkan biaya lisensi

menggunakan perangkat lunak sumber terbuka, perangkat lunak yang tidak memiliki biaya lisensi, atau alternatif untuk menekan biaya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

## Panduan implementasi

Pertimbangkan biaya layanan dan opsi saat memilih semua komponen. Termasuk di dalamnya adalah menggunakan layanan tingkat aplikasi dan terkelola, seperti [Amazon Relational Database Service](#) (Amazon RDS), [Amazon DynamoDB](#), [Amazon Simple Notification Service](#) (Amazon SNS), dan [Amazon Simple Email Service](#) (Amazon SES) untuk mengurangi biaya organisasi keseluruhan.

Gunakan nirserver dan kontainer untuk komputasi, seperti [AWS Lambda](#) dan [Amazon Simple Storage Service](#) (Amazon S3) untuk situs web statis. Lakukan kontainerisasi aplikasi Anda jika memungkinkan dan gunakan Layanan Kontainer Terkelola AWS seperti [Amazon Elastic Container Service](#) (Amazon ECS) atau [Amazon Elastic Kubernetes Service](#) (Amazon EKS).

Minimalkan biaya lisensi dengan menggunakan perangkat lunak sumber terbuka, atau perangkat lunak yang tidak memiliki ongkos lisensi (misalnya Amazon Linux untuk beban kerja komputasi atau migrasikan basis data ke Amazon Aurora).

Anda dapat menggunakan layanan nirserver atau tingkat aplikasi seperti [Lambda](#), [Amazon Simple Queue Service \(Amazon SQS\)](#), [Amazon SNS](#), dan [Amazon SES](#). Semua layanan ini menyingkirkan kebutuhan Anda untuk mengelola sumber daya, dan menyediakan fungsi eksekusi kode, layanan pengantrean, dan pengiriman pesan. Manfaat lain layanan-layanan ini adalah menskalakan kinerja dan biaya sesuai dengan penggunaan, sehingga memungkinkan alokasi dan atribusi biaya yang efisien.

Jika menggunakan [arsitektur yang didorong peristiwa](#) juga memungkinkan dengan layanan nirserver. Arsitektur yang didorong peristiwa didasarkan pada push, sehingga semuanya terjadi sesuai permintaan saat peristiwa muncul di dalam router. Dengan demikian, Anda tidak akan membayar polling yang terjadi terus-menerus untuk memeriksa peristiwa. Hasilnya adalah konsumsi bandwidth jaringan berkurang, penggunaan CPU berkurang, kapasitas armada tidak aktif berkurang, dan handshake SSL/TLS berkurang.

Untuk informasi selengkapnya tentang nirserver, lihat [Laporan resmi lensa Aplikasi Nirserver Well-Architected](#)

## Langkah implementasi

- Pilih tiap layanan untuk mengoptimalkan biaya: Menggunakan daftar dan analisis yang Anda prioritaskan, pilih setiap opsi yang menyediakan pasangan terbaik untuk prioritas organisasi Anda. Alih-alih meningkatkan kapasitas untuk memenuhi permintaan, pertimbangkan opsi-opsi lain yang dapat memberi Anda kinerja yang lebih baik dengan biaya yang lebih rendah. Sebagai contoh, jika Anda perlu meninjau lalu lintas basis data yang Anda perkirakan di AWS, pertimbangkan untuk meningkatkan ukuran instans atau menggunakan layanan Amazon ElastiCache (Redis atau Memcached) untuk menyediakan mekanisme dalam cache untuk basis data Anda.
- Evaluasi arsitektur yang didorong peristiwa: Dengan menggunakan arsitektur nirserver, Anda juga dapat membangun arsitektur yang didorong peristiwa untuk aplikasi berbasis layanan mikro terdistribusi, yang membantu Anda membangun solusi yang dapat diskalakan, tangguh, tangkas, dan hemat biaya.

## Sumber daya

### Dokumen terkait:

- [AWS Kalkulator Total Biaya Kepemilikan \(TCO\)](#)
- [AWS Nirserver](#)
- [Apa Itu Arsitektur yang Didorong Peristiwa?](#)
- [Amazon S3 kelas penyimpanan](#)
- [Produk cloud](#)
- [Amazon ElastiCache for Redis](#)

### Contoh terkait:

- [Mulai menggunakan arsitektur yang didorong peristiwa](#)
- [Arsitektur yang didorong peristiwa](#)
- [Bagaimana Statsig beroperasi 100x lebih hemat biaya menggunakan Amazon ElastiCache for Redis](#)
- [Praktik terbaik menggunakan fungsi AWS Lambda](#)

## COST05-BP06 Melakukan analisis biaya untuk penggunaan berbeda seiring waktu

Beban kerja bisa berubah dari waktu ke waktu. Beberapa layanan atau fitur lebih hemat biaya pada tingkat penggunaan yang berbeda. Dengan melakukan analisis pada setiap komponen dari waktu ke waktu serta pada penggunaan yang diperkirakan, beban kerja tetap hemat biaya di sepanjang masa pakainya.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Ketika AWS merilis layanan dan fitur baru, layanan optimal untuk beban kerja Anda mungkin berubah. Upaya yang diperlukan harus mencerminkan keuntungan potensial. Frekuensi peninjauan beban kerja tergantung pada persyaratan organisasi Anda. Jika beban kerja memiliki biaya yang signifikan, penerapan layanan baru lebih dini akan memaksimalkan penghematan biaya, sehingga manfaatnya bisa lebih besar jika peninjauan lebih sering dilakukan. Pemicu peninjauan lainnya adalah perubahan pola penggunaan. Perubahan yang signifikan pada penggunaan bisa menandakan bahwa layanan alternatif akan lebih optimal.

Jika Anda perlu memindahkan data ke AWS Cloud, Anda dapat memilih berbagai macam layanan yang ditawarkan oleh AWS dan alat-alat partner untuk membantu Anda memigrasikan set data Anda, baik berupa file, basis data, citra mesin, volume blok, bahkan cadangan berupa pita. Misalnya, untuk memindahkan data dalam jumlah besar ke dan dari AWS atau memproses data di edge, Anda dapat menggunakan salah satu perangkat yang dibuat khusus AWS untuk memindahkan data offline berukuran petabita secara hemat biaya. Contoh lainnya adalah untuk kecepatan transfer data yang lebih tinggi, layanan koneksi langsung mungkin lebih murah daripada VPN, yang menyediakan konektivitas konsisten yang diperlukan untuk bisnis Anda.

Berdasarkan analisis biaya untuk berbagai penggunaan seiring waktu, tinjau aktivitas penskalaan Anda. Analisis hasilnya untuk melihat apakah kebijakan penskalaan dapat disesuaikan untuk menambahkan instans dengan beberapa tipe instans dan opsi pembelian. Tinjau pengaturan Anda untuk melihat apakah pengaturan minimum dapat dikurangi untuk melayani permintaan pengguna tetapi dengan ukuran armada yang lebih kecil, dan tambahkan lebih banyak sumber daya untuk memenuhi permintaan tinggi yang diperkirakan.

Jalankan analisis biaya untuk berbagai penggunaan seiring waktu dengan berdiskusi dengan pemangku kepentingan di dalam organisasi Anda dan gunakan fitur prakiraan dari [AWS Cost](#)

[Explorer](#) untuk memprediksi potensi dampak perubahan layanan. Pantau pemicu tingkat penggunaan menggunakan AWS Budgets, alarm penagihan CloudWatch, dan AWS Cost Anomaly Detection untuk mengidentifikasi dan mengimplementasikan layanan yang paling hemat biaya lebih dini.

### Langkah implementasi

- Tentukan pola penggunaan yang diprediksi: Bersama organisasi Anda, seperti pemasaran dan pemilik produk, dokumentasikan seperti apa pola penggunaan yang diperkirakan dan yang diprediksi untuk beban kerja. Bersama pemangku kepentingan bisnis, diskusikan riwayat dan prakiraan peningkatan biaya dan penggunaan dan pastikan peningkatan tersebut sesuai dengan persyaratan bisnis. Identifikasi hari, minggu, atau bulan kalender yang Anda perkirakan ada lebih banyak pengguna yang menggunakan sumber daya AWS Anda, yang menandakan bahwa Anda harus meningkatkan kapasitas sumber daya yang ada atau mengadopsi layanan tambahan untuk menekan biaya dan meningkatkan kinerja.
- Jalankan analisis biaya pada penggunaan yang diprediksi: Menggunakan pola penggunaan yang telah ditetapkan, jalankan analisis pada setiap poin ini. Upaya analisis harus mencerminkan hasil potensial. Sebagai contoh, jika ada perubahan besar pada penggunaan, analisis yang mendalam harus dilakukan untuk memastikan biaya dan perubahan yang terjadi. Dengan kata lain, saat biaya meningkat, penggunaan untuk bisnis juga harus meningkat.

### Sumber daya

#### Dokumen terkait:

- [Kalkulator Total Biaya Kepemilikan \(TCO\) AWS](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)
- [Amazon EC2 Auto Scaling](#)
- [Migrasi Data Cloud](#)
- [AWS Snow Family](#)

#### Video terkait:

- [AWS OpsHub for Snow Family](#)

## Pilih tipe, ukuran, dan jumlah sumber daya yang tepat

Dengan memilih tipe, ukuran, dan jumlah sumber daya terbaik, Anda memenuhi persyaratan teknis dengan sumber daya berbiaya terendah. Aktivitas penyesuaian ukuran mempertimbangkan semua sumber daya beban kerja, semua atribut setiap sumber daya, dan upaya yang terlibat dalam operasi penyesuaian ukuran. Penyesuaian ukuran bisa menjadi proses iteratif, yang dipicu oleh perubahan pola penggunaan dan faktor eksternal, seperti turunnya harga AWS atau tipe sumber daya AWS baru. Penyesuaian ukuran juga dapat dilakukan satu kali jika biaya penyesuaian ukuran lebih tinggi daripada potensi penghematan selama masa guna beban kerja.

Di AWS, terdapat sejumlah pendekatan yang berbeda:

Praktik terbaik

- [COST06-BP01 Melakukan pemodelan biaya](#)
- [COST06-BP02 Memilih jenis, ukuran, dan jumlah sumber daya berdasarkan data](#)
- [COST06-BP03 Pilih tipe, ukuran, dan jumlah sumber daya secara otomatis berdasarkan metrik](#)

### COST06-BP01 Melakukan pemodelan biaya

Identifikasi persyaratan organisasi (seperti kebutuhan bisnis dan komitmen yang ada) dan lakukan pemodelan biaya (keseluruhan biaya) dari beban kerja serta setiap komponennya. Lakukan aktivitas tolok ukur untuk beban kerja di bawah berbagai beban yang diprediksi lalu bandingkan biayanya. Upaya pemodelan harus mencerminkan manfaat potensial. Misalnya, waktu yang digunakan sebanding dengan biaya komponen.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

#### Panduan implementasi

Lakukan pemodelan biaya untuk beban kerja Anda dan setiap komponennya guna memahami keseimbangan antar sumber daya, dan temukan ukuran yang tepat untuk setiap sumber daya di dalam beban kerja, dengan tingkat kinerja tertentu. Pemahaman tentang pertimbangan biaya dapat menjadi landasan untuk kasus bisnis dan proses pengambilan keputusan organisasi Anda saat mengevaluasi hasil realisasi nilai untuk deployment beban kerja yang direncanakan.

Lakukan aktivitas tolok ukur untuk beban kerja di bawah berbagai beban yang diprediksi lalu bandingkan biayanya. Hasil upaya pemodelan harus menggambarkan manfaat potensial, misalnya, waktu yang digunakan sebanding dengan biaya komponen atau penghematan yang diprediksi. Untuk

praktik terbaik, lihat [bagian Tinjauan dari Pilar Efisiensi Kinerja Kerangka Kerja Well-Architected AWS](#).

Sebagai contoh, untuk membuat pemodelan biaya untuk suatu beban kerja yang terdiri dari sumber daya komputasi, [AWS Compute Optimizer](#) dapat membantunya dengan pemodelan biaya untuk beban kerja yang sedang berjalan. Layanan ini menyediakan rekomendasi penyesuaian ukuran untuk sumber daya komputasi berdasarkan riwayat penggunaan. Pastikan CloudWatch Agents di-deploy ke instans Amazon EC2 agar dapat mengumpulkan metrik memori yang membantu Anda mendapatkan rekomendasi yang lebih akurat di dalam AWS Compute Optimizer. Ini adalah sumber data ideal untuk sumber daya komputasi karena ini adalah layanan gratis, yang menggunakan machine learning untuk memberikan beberapa rekomendasi tergantung tingkat risiko.

Terdapat [beberapa layanan](#) yang dapat Anda gunakan dengan log kustom sebagai sumber data untuk menyesuaikan ukuran operasi untuk layanan dan komponen beban kerja lain, seperti [AWS Trusted Advisor](#), [Amazon CloudWatch](#) dan [Amazon CloudWatch Logs](#). AWS Trusted Advisor memeriksa sumber daya dan menandai sumber daya dengan penggunaan rendah yang dapat membantu Anda menyesuaikan ukuran sumber daya Anda dan membuat pemodelan biaya.

Berikut ini adalah beberapa rekomendasi untuk data dan metrik pemodelan biaya:

- Pemantauan harus mencerminkan pengalaman pengguna secara akurat. Pilih tingkat detail yang tepat untuk periode waktu dan dengan cermat pilih persentil maksimum atau ke-99, bukan rata-rata.
- Pilih tingkat detail yang tepat untuk periode waktu analisis yang diperlukan untuk mencakup siklus beban kerja apa pun. Sebagai contoh, jika dilakukan analisis dua minggu, Anda mungkin mengabaikan siklus pemanfaatan tinggi bulanan, yang dapat menyebabkan pengadaan yang terlalu rendah.
- Pilih layanan AWS yang tepat untuk beban kerja yang Anda rencanakan dengan mempertimbangkan komitmen Anda saat ini, model harga yang dipilih untuk beban kerja lain, dan kemampuan untuk berinovasi lebih cepat dan berfokus pada nilai bisnis inti Anda.

### Langkah implementasi

- Lakukan pemodelan biaya untuk sumber daya: Deploy beban kerja atau bukti konsep ke dalam akun terpisah dengan tipe dan ukuran sumber daya tertentu untuk diuji. Jalankan beban kerja dengan data pengujian dan rekam hasil output, beserta data biaya untuk periode pengujian. Kemudian, deploy ulang beban kerja atau ubah tipe dan ukuran sumber daya lalu jalankan ulang pengujian. Sertakan biaya lisensi produk apa pun yang mungkin Anda gunakan dengan sumber



daya ini serta perkiraan biaya operasi (tenaga kerja atau teknisi) untuk men-deploy dan mengelola sumber daya ini selama membuat pemodelan biaya. Pertimbangkan pemodelan biaya untuk periode tertentu (jam, harian, bulanan, tahunan, atau tiga tahun).

## Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [Mengidentifikasi Peluang untuk Menyesuaikan Ukuran](#)
- [Fitur Amazon CloudWatch](#)
- [Optimalisasi Biaya: Penyesuaian Ukuran Amazon EC2](#)
- [AWS Compute Optimizer](#)
- [Kalkulator Harga AWS](#)

Contoh terkait:

- [Lakukan Pemodelan Biaya yang Didorong Data](#)
- [Hitung biaya konfigurasi sumber daya AWS yang direncanakan](#)
- [Pilih alat AWS yang tepat](#)

## COST06-BP02 Memilih jenis, ukuran, dan jumlah sumber daya berdasarkan data

Pilih jenis atau ukuran sumber daya berdasarkan data tentang karakteristik sumber daya dan beban kerja. Misalnya, intensif komputasi, memori, throughput, atau tulis. Pilihan ini biasanya dibuat menggunakan beban kerja versi sebelumnya (on-premise), menggunakan dokumentasi, atau menggunakan sumber informasi lain untuk beban kerja.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Amazon EC2 menyediakan berbagai pilihan jenis instans dengan tingkat CPU, memori, penyimpanan, dan kapasitas jaringan yang berbeda untuk kasus penggunaan yang berbeda. Jenis-jenis instans ini memadukan kemampuan CPU, memori, penyimpanan, dan jaringan yang

berbeda-beda, sehingga Anda bisa lebih fleksibel saat memilih kombinasi sumber daya yang tepat untuk proyek Anda. Setiap jenis instans tersedia dalam berbagai ukuran, sehingga Anda dapat menyesuaikan sumber daya berdasarkan tuntutan beban kerja Anda. Untuk menentukan jenis instans yang Anda butuhkan, kumpulkan detail tentang persyaratan sistem aplikasi atau perangkat lunak yang akan Anda jalankan pada instans Anda. Detail ini sebaiknya mencakup hal-hal berikut:

- Sistem operasi
- Jumlah core CPU
- Core GPU
- Jumlah memori sistem (RAM)
- Jenis dan ruang penyimpanan
- Persyaratan bandwidth jaringan

Identifikasi tujuan persyaratan komputasi dan instans mana yang diperlukan, lalu jelajahi berbagai rangkaian instans Amazon EC2. Amazon menawarkan rangkaian jenis instans berikut:

- Instans Tujuan Umum
- Komputasi Dioptimalkan
- Memori Dioptimalkan
- Penyimpanan Dioptimalkan
- Komputasi Dipercepat
- HPC Dioptimalkan

Untuk pemahaman yang lebih mendalam tentang tujuan spesifik dan kasus penggunaan yang dapat dipenuhi oleh rangkaian instans Amazon EC2 tertentu, lihat [Jenis Instans AWS](#).

Pengumpulan persyaratan sistem sangat penting bagi Anda untuk memilih rangkaian instans yang spesifik dan jenis instans yang paling sesuai dengan kebutuhan Anda. Nama jenis instans terdiri dari nama rangkaian dan ukuran instans. Misalnya, instans t2.micro berasal dari rangkaian T2 dan berukuran mikro.

Pilih jenis atau ukuran sumber daya berdasarkan karakteristik sumber daya dan beban kerja (misalnya, intensif komputasi, memori, throughput, atau tulis). Pilihan ini biasanya dibuat menggunakan pemodelan biaya, beban kerja versi sebelumnya (seperti versi on-premise), menggunakan dokumentasi, atau menggunakan sumber informasi lain tentang beban kerja (laporan

resmi atau solusi yang dipublikasikan). Menggunakan kalkulator harga atau alat manajemen biaya AWS dapat membantu dalam membuat keputusan yang lebih tepat tentang jenis, ukuran, dan konfigurasi instans.

### Langkah implementasi

- Pilih sumber daya berdasarkan data: Gunakan data pemodelan biaya Anda untuk memilih tingkat penggunaan beban kerja yang diantisipasi, dan pilih jenis serta ukuran sumber daya yang ditentukan. Dengan mengandalkan data pemodelan biaya, tentukan jumlah CPU virtual, total memori (GiB), volume penyimpanan instans lokal (GB), volume Amazon EBS, dan tingkat performa jaringan, dengan mempertimbangkan kecepatan transfer data yang diperlukan untuk instans. Selalu buat pilihan berdasarkan analisis mendetail dan data yang akurat untuk mengoptimalkan performa sekaligus mengelola biaya secara efektif.

### Sumber daya

#### Dokumen terkait:

- [Jenis Instans AWS](#)
- [AWS Auto Scaling](#)
- [Fitur Amazon CloudWatch](#)
- [Optimisasi Biaya: Rightsizing EC2](#)

#### Video terkait:

- [Selecting the right Amazon EC2 instance for your workloads](#)
- [Right Size Your Service](#)

#### Contoh terkait:

- [It just got easier to discover and compare Amazon EC2 instance types](#)

## COST06-BP03 Pilih tipe, ukuran, dan jumlah sumber daya secara otomatis berdasarkan metrik

Gunakan metrik dari beban kerja yang sedang berjalan untuk memilih ukuran dan tipe yang tepat untuk mengoptimalkan biaya. Sediakan throughput, ukuran, dan penyimpanan secara tepat untuk layanan komputasi, penyimpanan, data, dan jaringan. Ini dapat dilakukan dengan loop umpan balik seperti penskalaan otomatis atau dengan kode kustom di dalam beban kerja.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Buat loop umpan balik di dalam beban kerja yang menggunakan metrik aktif dari beban kerja yang berjalan untuk melakukan perubahan pada beban kerja tersebut. Anda dapat menggunakan layanan terkelola, seperti [AWS Auto Scaling](#), yang Anda konfigurasi untuk melakukan operasi penyesuaian ukuran untuk Anda. AWS juga menyediakan [API](#), [SDK](#), dan fitur-fitur yang memungkinkan agar sumber daya dapat dimodifikasi dengan upaya minimal. Anda dapat memprogram beban kerja agar menghentikan dan menjalankan instans Amazon EC2 untuk memungkinkan perubahan ukuran atau tipe instans. Hal ini menyediakan manfaat penyesuaian ukuran sambil menghilangkan hampir semua biaya operasional yang diperlukan untuk melakukan perubahan.

Beberapa layanan AWS memiliki pemilihan tipe atau ukuran otomatis secara bawaan, seperti [Amazon Simple Storage Service Intelligent-Tiering](#). Intelligent-Tiering secara otomatis memindahkan data Anda antara dua tingkat akses, akses sering dan akses jarang, berdasarkan pola penggunaan Anda.

### Langkah implementasi

- Tingkatkan observabilitas dengan mengonfigurasi metrik beban kerja: Rekam metrik-metrik utama untuk beban kerja. Metrik-metrik tersebut menyediakan indikasi pengalaman pelanggan, seperti output beban kerja, dan penyesuaian dengan perbedaan antartipe dan ukuran sumber daya, seperti penggunaan CPU dan memori. Untuk sumber daya komputasi, analisis data kinerja untuk menyesuaikan ukuran instans Amazon EC2 Anda. Identifikasi instans tidak aktif dan instans dengan pemanfaatan terlalu rendah. Metrik utama yang dicari adalah penggunaan CPU dan pemanfaatan memori (misalnya, 40% pemanfaatan CPU pada 90% waktu seperti yang dijelaskan dalam [Menyesuaikan Ukuran dengan AWS Compute Optimizer dan Pemanfaatan Memori Diaktifkan](#)). Identifikasi instans dengan penggunaan CPU maksimum dan pemanfaatan

memori kurang dari 40% selama jangka waktu empat minggu. Ini adalah instans untuk disesuaikan ukurannya guna memangkas biaya. Untuk sumber daya penyimpanan seperti Amazon S3, Anda dapat menggunakan [Lensa Penyimpanan Amazon S3](#), yang memungkinkan Anda melihat 28 metrik di berbagai kategori pada tingkat bucket, dan data riwayat 14 hari di dasbor secara default. Anda dapat memfilter dasbor Lensa Penyimpanan Amazon S3 Anda berdasarkan ringkasan dan pengoptimalan biaya atau peristiwa untuk menganalisis metrik-metrik tertentu.

- Lihat rekomendasi penyesuaian ukuran: Gunakan rekomendasi penyesuaian ukuran di AWS Compute Optimizer dan alat penyesuaian ukuran Amazon EC2 di konsol Manajemen Biaya, atau tinjau AWS Trusted Advisor yang menyesuaikan ukuran sumber daya Anda untuk melakukan penyesuaian pada beban kerja Anda. Penting untuk menggunakan [alat yang tepat](#) ketika menyesuaikan ukuran sumber daya yang berbeda dan ikuti [pedoman penyesuaian ukuran](#) baik untuk instans Amazon EC2, kelas penyimpanan AWS, atau tipe instans Amazon RDS. Untuk sumber daya penyimpanan, Anda dapat menggunakan Lensa Penyimpanan Amazon S3, yang memberi Anda visibilitas tentang penggunaan penyimpanan objek dan tren aktivitas, serta memberikan rekomendasi yang dapat ditindaklanjuti untuk mengoptimalkan biaya dan menerapkan praktik terbaik perlindungan data. Menggunakan rekomendasi kontekstual yang diambil oleh [Lensa Penyimpanan Amazon S3](#) dari analisis metrik di seluruh organisasi Anda, Anda dapat mengambil langkah-langkah langsung untuk mengoptimalkan penyimpanan.
- Pilih tipe dan ukuran sumber daya secara otomatis berdasarkan metrik: Menggunakan metrik beban kerja, pilih sumber daya beban kerja Anda secara manual atau otomatis. Untuk sumber daya komputasi, konfigurasi AWS Auto Scaling atau implementasi kode di dalam aplikasi Anda dapat menghemat energi yang diperlukan jika diperlukan perubahan yang sering, dan ini dapat berpotensi mengimplementasikan perubahan lebih awal daripada proses manual. Anda dapat meluncurkan dan secara otomatis menskalakan armada Instans Sesuai Permintaan dan Instans Spot di dalam satu grup Auto Scaling. Selain menerima diskon karena menggunakan Instans Spot, Anda dapat menggunakan Instans Terpesan atau Savings Plan untuk menerima tarif diskon untuk harga Instans Sesuai Permintaan biasa. Semua kombinasi faktor tersebut membantu Anda mengoptimalkan penghematan biaya Anda untuk instans Amazon EC2 dan menentukan skala dan kinerja yang diinginkan untuk aplikasi Anda. Anda juga dapat menggunakan strategi [pemilihan tipe berbasis atribut \(ABS\)](#) di [Auto Scaling Groups \(ASG\)](#), yang memungkinkan Anda untuk menyatakan persyaratan instans Anda dalam bentuk rangkaian atribut, seperti vCPU, memori, dan penyimpanan. Anda dapat menggunakan tipe instans generasi lebih baru secara otomatis ketika tipe tersebut dirilis dan mengakses cakupan kapasitas yang lebih luas dengan Instans Spot Amazon EC2. Armada Amazon EC2 dan Amazon EC2 Auto Scaling memilih dan meluncurkan instans yang cocok dengan atribut yang ditentukan, sehingga tipe instans tidak perlu dipilih secara manual. Untuk sumber daya penyimpanan, Anda dapat menggunakan fitur [Amazon S3 Intelligent](#)

[Tiering](#) dan [Amazon EFS Infrequent Access](#), yang memungkinkan Anda untuk memilih kelas penyimpanan secara otomatis yang menghadirkan penghematan biaya penyimpanan otomatis saat pola akses data berubah, tanpa dampak kinerja atau biaya overhead operasional.

## Sumber daya

### Dokumen terkait:

- [AWS Auto Scaling](#)
- [Penyesuaian Ukuran AWS](#)
- [AWS Compute Optimizer](#)
- [Fitur Amazon CloudWatch](#)
- [CloudWatch Penyiapan](#)
- [CloudWatch Memublikasikan Metrik Kustom](#)
- [Mulai Menggunakan Amazon EC2 Auto Scaling](#)
- [Lensa Penyimpanan Amazon S3](#)
- [Amazon S3 Intelligent-Tiering](#)
- [Amazon EFS Infrequent Access](#)
- [Meluncurkan Instans Amazon EC2 Menggunakan SDK](#)

### Video terkait:

- [Menyesuaikan Ukuran Layanan Anda](#)

### Contoh terkait:

- [Pemilihan Tipe Instans berbasis atribut untuk Auto Scaling untuk Armada Amazon EC2](#)
- [Mengoptimalkan Amazon Elastic Container Service untuk biaya menggunakan penskalaan terjadwal](#)
- [Penskalaan Prediktif dengan Amazon EC2 Auto Scaling](#)
- [Mengoptimalkan Biaya dan Memperoleh Visibilitas tentang Penggunaan dengan Lensa Penyimpanan Amazon S3](#)
- [Lab Well-Architected: Rekomendasi Penyesuaian Ukuran \(Level 100\)](#)

- [Lab Well-Architected: Menyesuaikan Ukuran dengan Mengaktifkan AWS Compute Optimizer dan Pemanfaatan Memori \(Level 200\)](#)

## Memilih model harga terbaik

### Praktik terbaik

- [COST07-BP01 Melakukan analisis model harga](#)
- [COST07-BP02 Memilih Wilayah berdasarkan biaya](#)
- [COST07-BP03 Memilih perjanjian pihak ketiga dengan ketentuan hemat biaya](#)
- [COST07-BP04 Mengimplementasikan model harga untuk semua komponen beban kerja ini](#)
- [COST07-BP05 Melakukan analisis model harga di tingkat akun manajemen](#)

## COST07-BP01 Melakukan analisis model harga

Analisis setiap komponen beban kerja. Tentukan apakah komponen dan sumber daya akan dijalankan dalam waktu yang lama (untuk diskon komitmen), atau bersifat dinamis dan dijalankan dalam waktu singkat (untuk spot atau sesuai permintaan). Lakukan analisis pada beban kerja menggunakan rekomendasi dalam alat manajemen biaya dan terapkan aturan bisnis pada rekomendasi tersebut untuk mendapatkan hasil tinggi.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

AWS memiliki beberapa [model harga](#) yang memungkinkan Anda membayar sumber daya Anda dengan cara yang paling hemat biaya, yang sesuai dengan kebutuhan organisasi Anda dan bergantung pada produk. Bekerjalah dengan tim Anda untuk menentukan model harga yang paling sesuai. Sering kali model harga terdiri dari kombinasi beberapa opsi, yang ditentukan oleh ketersediaan Anda

Instans Sesuai Permintaan memungkinkan Anda membayar kapasitas basis data atau komputasi per jam atau per detik (minimum 60 detik), tergantung pada instans mana yang Anda jalankan, tanpa komitmen jangka panjang atau pembayaran di muka.

Savings Plans merupakan model harga fleksibel yang menawarkan harga murah untuk penggunaan Amazon EC2, Lambda, dan AWS Fargate (Fargate), sebagai imbalan atas komitmen terhadap

sejumlah penggunaan yang konsisten (diukur dalam dolar per jam) selama jangka waktu satu tahun atau tiga tahun.

Instans Spot adalah mekanisme harga Amazon EC2 yang memungkinkan Anda meminta kapasitas komputasi cadangan dengan tarif diskon per jam (diskon hingga 90% dari harga sesuai permintaan) tanpa komitmen di muka.

Instans Terpesan memberi Anda diskon hingga 75 persen dengan pembayaran kapasitas di muka. Untuk detail selengkapnya, lihat [Mengoptimalkan biaya dengan reservasi](#).

Anda mungkin memilih untuk menyertakan Savings Plan untuk sumber daya yang terkait dengan lingkungan produksi, kualitas, dan pengembangan. Alternatifnya, karena sumber daya sandbox hanya diaktifkan saat diperlukan, Anda mungkin memilih model sesuai permintaan untuk sumber daya dalam lingkungan tersebut. Gunakan [Instans Spot](#) Amazon untuk mengurangi biaya Amazon EC2 atau gunakan [Savings Plans Komputasi](#) untuk mengurangi biaya Amazon EC2, Fargate, dan Lambda. Alat rekomendasi [AWS Cost Explorer](#) memberikan peluang untuk diskon komitmen dengan Savings Plans.

Jika Anda pernah membeli [Instans Terpesan](#) untuk Amazon EC2 di masa lalu atau Anda telah menjalankan praktik alokasi biaya di dalam organisasi Anda, Anda dapat terus menggunakan Instans Terpesan Amazon EC2 sementara ini. Tetapi, kami menyarankan agar Anda membuat strategi untuk menggunakan Savings Plans di waktu mendatang sebagai mekanisme penghematan biaya yang lebih fleksibel. Anda dapat menyegarkan Rekomendasi Savings Plans (SP) di AWS Cost Management untuk membuat Rekomendasi Savings Plans baru kapan saja. Gunakan Instans Terpesan (RI) untuk mengurangi biaya Amazon RDS, Amazon Redshift, Amazon ElastiCache, dan Amazon OpenSearch Service. Savings Plans dan Instans Terpesan tersedia dalam tiga opsi: semua pembayaran di muka, sebagian pembayaran di muka, dan tanpa pembayaran di muka. Gunakan rekomendasi yang diberikan di AWS Cost Explorer rekomendasi pembelian RI dan SP.

Untuk menemukan peluang untuk beban kerja Spot, gunakan tampilan penggunaan secara keseluruhan per jam, dan cari periode penggunaan atau elastisitas yang berubah secara teratur. Anda dapat menggunakan Instans Spot untuk berbagai aplikasi yang fleksibel dan toleran terhadap kesalahan. Contohnya antara lain server web tanpa status, titik akhir API, aplikasi analitik dan big data, beban kerja terkontainerisasi, CI/CD, dan beban kerja fleksibel lainnya.

Analisis instans Amazon EC2 dan Amazon RDS Anda untuk mengetahui apakah instans tersebut dapat dimatikan ketika Anda tidak menggunakannya (sesuai jam kerja dan akhir pekan). Pendekatan ini akan memungkinkan Anda mengurangi biaya hingga 70% atau lebih dibandingkan penggunaan 24/7. Jika Anda memiliki kluster Amazon Redshift yang hanya perlu disediakan pada waktu tertentu,



Anda dapat menghentikan sejenak klaster dan melanjutkannya nanti. Ketika klaster Amazon Redshift atau Instans Amazon EC2 dan Amazon RDS dihentikan, penagihan komputasi terhenti dan hanya biaya penyimpanan yang berlaku.

Perhatikan bahwa [reservasi Kapasitas Sesuai Permintaan](#) (ODCR) bukanlah diskon harga. Reservasi Kapasitas dikenakan biaya sesuai tarif Sesuai Permintaan yang setara, baik Anda menjalankan instans dalam kapasitas terpesan atau tidak. Reservasi ini harus dipertimbangkan saat Anda harus memberikan cukup banyak kapasitas untuk sumber daya yang menurut rencana akan Anda jalankan. ODCR tidak harus terikat dengan komitmen jangka panjang, karena ODCR dapat dibatalkan saat Anda tidak lagi membutuhkannya, tetapi ODCR juga dapat mendapat manfaat dari diskon yang diberikan Savings Plans atau Instans Terpesan.

### Langkah implementasi

- Analisis elastisitas beban kerja: Menggunakan granularitas per jam dalam Cost Explorer atau dasbor kustom, analisis elastisitas beban kerja Anda. Cari perubahan teratur dalam jumlah instans yang dijalankan. Instans berdurasi pendek merupakan kandidat untuk Instans Spot atau Armada Spot.
  - [Well-Architected Lab: Cost Explorer](#)
  - [Well-Architected Lab: Visualisasi Biaya](#)
- Tinjau kontrak harga yang ada: Tinjau komitmen atau kontrak saat ini untuk kebutuhan jangka panjang. Analisis apa yang Anda miliki saat ini dan berapa banyak komitmen tersebut digunakan. Manfaatkan diskon kontrak atau perjanjian perusahaan yang sudah ada sebelumnya. [Perjanjian Perusahaan](#) memberikan kepada pelanggan opsi untuk menyesuaikan perjanjian yang paling sesuai dengan kebutuhan mereka. Untuk komitmen jangka panjang, pertimbangkan diskon harga terpesan, Instans Terpesan atau Savings Plans untuk jenis instans, kelompok instans, Wilayah AWS, dan Zona Ketersediaan tertentu.
- Lakukan analisis diskon komitmen: Menggunakan Cost Explorer di akun Anda, tinjau rekomendasi Savings Plans dan Instans Terpesan. Untuk memverifikasi bahwa Anda mengimplementasikan rekomendasi yang benar dengan risiko dan diskon yang diperlukan, ikuti [Well-Architected Lab](#).

### Sumber daya

#### Dokumen terkait:

- [Mengakses rekomendasi Instans Terpesan](#)
- [Opsi pembelian instans](#)

- [AWS Perusahaan](#)

Video terkait:

- [Hemat hingga 90% dan jalankan beban kerja produksi di Spot](#)

Contoh terkait:

- [Well-Architected Lab: Cost Explorer](#)
- [Well-Architected Lab: Visualisasi Biaya](#)
- [Well-Architected Lab: Model Harga](#)

## COST07-BP02 Memilih Wilayah berdasarkan biaya

Praktik terbaik ini diperbarui dengan panduan baru pada 13 Juli 2023.

Penetapan harga sumber daya dapat berbeda di setiap Wilayah. Identifikasi perbedaan biaya sesuai Wilayah dan lakukan deployment di Wilayah dengan biaya yang lebih tinggi hanya untuk memenuhi persyaratan latensi, residensi data, dan kedaulatan data. Dengan mempertimbangkan biaya Wilayah Anda dapat memperoleh harga keseluruhan yang paling rendah untuk beban kerja ini.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Infrastruktur [AWS Cloud](#) bersifat global, di-host di [berbagai lokasi di seluruh dunia](#), dan dibangun berdasarkan Wilayah AWS, Zona Ketersediaan, Zona Lokal, AWS Outposts, dan Zona Wavelength. Wilayah adalah lokasi fisik di dunia dan setiap Wilayah merupakan area geografis yang terpisah dengan beberapa Zona Ketersediaan yang dimiliki AWS. Zona Ketersediaan yang merupakan beberapa lokasi terpisah di dalam setiap Wilayah terdiri dari satu atau beberapa pusat data khusus, masing-masing dengan daya, jaringan, dan konektivitas redundan.

Setiap Wilayah AWS beroperasi sesuai kondisi pasar lokal, dan harga sumber daya berbeda-beda di setiap Wilayah karena perbedaan biaya seperti lahan, fiber, listrik, dan pajak. Pilih Wilayah tertentu untuk mengoperasikan komponen atau seluruh solusi sehingga Anda dapat menjalankannya dengan harga serendah mungkin secara global. Gunakan [Kalkulator AWS](#) untuk menghitung perkiraan biaya

beban kerja Anda di berbagai Wilayah dengan mencari layanan berdasarkan tipe lokasi (Wilayah, zona wavelength, dan zona lokal) dan Wilayah.

Saat Anda merancang solusi, praktik terbaik yang dapat diterapkan adalah berusaha menempatkan sumber daya komputasi lebih dekat dengan pengguna untuk memberikan latensi yang lebih rendah dan kedaulatan data yang kuat. Pilih lokasi geografis berdasarkan persyaratan bisnis, privasi data, kinerja, dan keamanan Anda. Untuk aplikasi dengan pengguna akhir global, gunakan beberapa lokasi.

Gunakan Wilayah yang menyediakan harga lebih rendah untuk layanan AWS untuk men-deploy beban kerja Anda jika Anda tidak memiliki keharusan memenuhi persyaratan bisnis, privasi data, dan keamanan. Sebagai contoh, jika Wilayah default Anda adalah ap-southeast-2 (Sydney), dan jika tidak ada pembatasan (misalnya privasi data dan keamanan) untuk menggunakan Wilayah lain, men-deploy instans Amazon EC2 nonkritik (pengembangan dan pengujian) di Wilayah north-east-1 (N. Virginia) akan lebih murah.

|                  | <i>Kepatuhan</i> | <i>Latensi</i>      | <i>Biaya</i> | <i>Layanan/Fitur</i> |
|------------------|------------------|---------------------|--------------|----------------------|
| <i>Wilayah 1</i> | ✓                | 15 milidetik        | \$\$         | ✓                    |
| <i>Wilayah 2</i> | ✓                | 20 milidetik        | \$\$\$       | X                    |
| <i>Wilayah 3</i> | ✓                | 80 milidetik        | \$           | ✓                    |
| <i>Wilayah 4</i> | ✓                | 15 milidetik        | \$\$         | ✓                    |
| <i>Wilayah 5</i> | ✓                | 20 milidetik        | \$\$\$       | X                    |
| <b>Wilayah 6</b> | ✓                | <b>15 milidetik</b> | <b>\$</b>    | ✓                    |
| <i>Wilayah 7</i> | ✓                | 80 milidetik        | \$           | ✓                    |
| <i>Wilayah 8</i> | ✓                | 15 milidetik        | \$           | X                    |

Tabel matriks fitur Wilayah

Tabel matriks di atas menunjukkan kepada kita bahwa Wilayah 4 adalah opsi terbaik untuk skenario yang diberikan ini karena latensinya rendah dibandingkan dengan wilayah lain, layanannya tersedia, dan ini adalah Wilayah yang paling murah.

## Langkah implementasi

- Tinjau harga Wilayah AWS: Analisis biaya beban kerja di Wilayah saat ini. Mulai dengan harga tertinggi dari jenis layanan dan penggunaan, lalu hitung biayanya di Wilayah yang tersedia. Jika perkiraan penghematan lebih banyak dari biaya pemindahan komponen atau beban kerja, lakukan migrasi ke Wilayah baru.
- Tinjau persyaratan ulasan untuk deployment multi-Wilayah: Analisis persyaratan dan kewajiban bisnis Anda (privasi data, keamanan, atau kinerja) untuk mencari tahu apakah ada pembatasan bagi Anda untuk menggunakan beberapa Wilayah. Gunakan beberapa Wilayah jika tidak ada kewajiban yang membatasi Anda untuk menggunakan Wilayah tunggal.
- Analisis transfer data yang diperlukan: Pertimbangkan biaya transfer data saat memilih Wilayah. Simpan data Anda dekat dengan sumber dayanya dan dengan pelanggan Anda. Pilih Wilayah AWS berbiaya lebih rendah di mana data mengalir dan transfer data hanya sedikit. Tergantung kebutuhan bisnis Anda untuk transfer data, Anda dapat menggunakan [Amazon CloudFront](#), [AWS PrivateLink](#), [AWS Direct Connect](#), dan [AWS Virtual Private Network](#) untuk mengurangi biaya jaringan Anda, meningkatkan kinerja, dan menambah keamanan.

## Sumber daya

### Dokumen terkait:

- [Mengakses rekomendasi Instans Terpesan](#)
- [Penetapan harga Amazon EC2](#)
- [Opsi pembelian instans](#)
- [Tabel Wilayah](#)

### Video terkait:

- [Hemat hingga 90% dan jalankan beban kerja produksi di Spot](#)

### Contoh terkait:

- [Ikhtisar Biaya Transfer Data untuk Arsitektur Umum](#)
- [Pertimbangan Biaya untuk Deployment Global](#)
- [Hal-Hal yang Perlu Dipertimbangkan saat Memilih Wilayah untuk Beban Kerja](#)

- [Lab Well-Architected: Membatasi penggunaan layanan berdasarkan Wilayah \(Level 200\)](#)

## COST07-BP03 Memilih perjanjian pihak ketiga dengan ketentuan hemat biaya

Perjanjian dan ketentuan yang hemat biaya memastikan biaya layanan ini dapat disesuaikan dengan manfaat yang disediakan. Pilih perjanjian dan harga yang dapat diskalakan ketika ada keuntungan tambahan yang disediakan untuk organisasi Anda.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Ada beberapa produk di pasar yang dapat membantu Anda mengelola biaya di lingkungan cloud Anda. Produk-produk tersebut mungkin memiliki fitur yang berbeda-beda sesuai kebutuhan pelanggan; misalnya, beberapa di antaranya berfokus pada tata kelola biaya atau visibilitas biaya, sedangkan produk lainnya berfokus pada pengoptimalan biaya. Salah satu faktor penting dalam pengoptimalan dan tata kelola biaya yang efektif adalah penggunaan alat yang tepat dengan fitur yang sesuai kebutuhan dan model harga yang tepat. Model harga untuk produk-produk ini berbeda. Beberapa produk mengenakan persentase tertentu dari tagihan bulanan Anda, sementara produk lainnya mengenakan persentase dari realisasi penghematan Anda. Idealnya, Anda seharusnya hanya membayar sesuai apa yang Anda butuhkan.

Saat Anda menggunakan solusi atau layanan pihak ketiga di cloud, penting untuk memastikan bahwa struktur harganya selaras dengan hasil yang Anda inginkan. Harga harus sesuai dengan hasil dan nilai yang disediakan. Contohnya, dalam perangkat lunak yang mengambil persentase dari penghematan yang dihasilkannya, makin banyak Anda menghemat (hasil), makin tinggi juga biayanya. Perjanjian lisensi yang mengharuskan Anda membayar lebih banyak ketika pengeluaran Anda meningkat mungkin bukan cara terbaik bagi Anda untuk mengoptimalkan biaya. Namun, jika vendor tersebut menawarkan manfaat yang jelas untuk semua elemen dalam tagihan Anda, biaya yang meningkat ini mungkin dapat dijustifikasi.

Misalnya, solusi yang menyediakan rekomendasi untuk Amazon EC2 dan mengenakan sekian persen dari seluruh tagihan Anda akan menjadi makin mahal jika Anda menggunakan layanan lain yang tidak memberikan manfaat. Contoh lainnya adalah layanan terkelola yang dikenakan biaya sekian persen dari biaya sumber daya yang dikelola. Ukuran instans yang lebih besar tidak selalu memerlukan lebih banyak upaya manajemen, tetapi dapat dikenakan biaya lebih banyak. Pastikan

bahwa perjanjian harga layanan ini mencakup fitur atau program pengoptimalan biaya di layanannya untuk mendorong efisiensi.

Pelanggan mungkin merasa produk-produk yang ada di pasar ini lebih canggih atau mudah digunakan. Anda perlu mempertimbangkan biaya produk ini dan memikirkan potensi hasil pengoptimalan biaya dalam jangka panjang.

#### Langkah implementasi

- Analisis perjanjian dan ketentuan pihak ketiga: Tinjau harga dalam perjanjian pihak ketiga. Lakukan pemodelan untuk berbagai tingkat penggunaan Anda, dan pertimbangkan biaya baru seperti penggunaan layanan baru, atau peningkatan pada layanan saat ini akibat pertumbuhan beban kerja. Tentukan apakah biaya tambahan memberikan keuntungan yang diperlukan pada bisnis Anda.

#### Sumber daya

Dokumen terkait:

- [Mengakses rekomendasi Instans Terpesan](#)
- [Opsi pembelian instans](#)

Video terkait:

- [Hemat hingga 90% dan jalankan beban kerja produksi di Spot](#)

## COST07-BP04 Mengimplementasikan model harga untuk semua komponen beban kerja ini

Sumber daya yang berjalan secara permanen harus menggunakan kapasitas terpesan seperti Savings Plans atau Instans Terpesan. Kapasitas jangka pendek dikonfigurasi untuk menggunakan Instans Spot, atau Armada Spot. Instans Sesuai Permintaan hanya digunakan untuk beban kerja jangka pendek yang tidak bisa dihentikan dan tidak berjalan cukup lama untuk kapasitas terpesan, yakni antara 25% sampai 75% dari periode, tergantung pada tipe sumber daya.

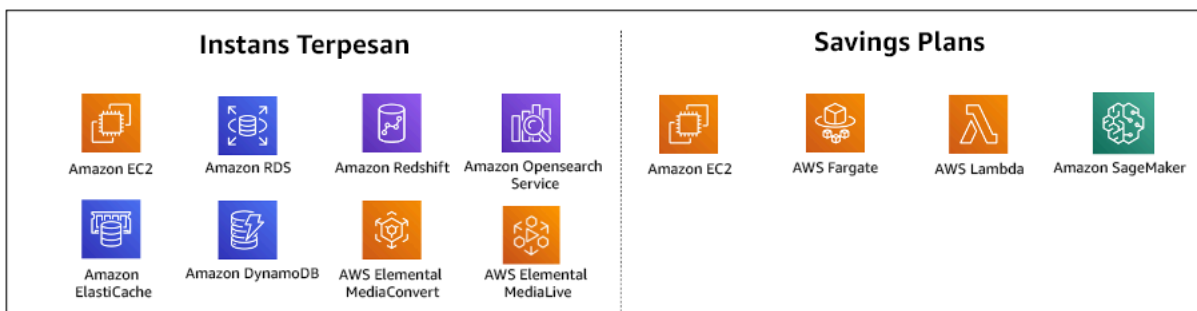
Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

## Panduan implementasi

Untuk meningkatkan efisiensi biaya, AWS memberikan beberapa rekomendasi komitmen berdasarkan riwayat penggunaan Anda. Anda dapat menggunakan rekomendasi ini untuk memahami apa yang dapat Anda hemat, dan bagaimana penggunaan komitmen tersebut. Anda dapat menggunakan layanan ini dengan cara Sesuai Permintaan, Spot, atau membuat komitmen untuk jangka waktu tertentu dan mengurangi biaya sesuai permintaan dengan Instans Terpesan (RI) dan Savings Plans (SP). Selain memahami setiap komponen beban kerja dan beberapa layanan AWS, Anda juga perlu memahami diskon komitmen, opsi pembelian, dan Instans Spot untuk layanan ini agar dapat mengoptimalkan beban kerja Anda.

Pertimbangkan persyaratan komponen beban kerja Anda, dan pahami model harga yang berbeda untuk layanan ini. Tentukan persyaratan ketersediaan komponen ini. Tentukan apakah ada beberapa sumber daya independen yang melakukan fungsinya pada beban kerja, serta apa persyaratan beban kerja dari waktu ke waktu. Bandingkan biaya sumber daya menggunakan model harga Sesuai Permintaan default dan model lain yang dapat diterapkan. Pertimbangkan potensi perubahan apa pun pada sumber daya atau komponen beban kerja.

Sebagai contoh, mari kita lihat Arsitektur Aplikasi Web ini di AWS. Beban kerja sampel ini terdiri dari beberapa layanan AWS, seperti Amazon Route 53, AWS WAF, Amazon CloudFront, instans Amazon EC2, instans Amazon RDS, Penyeimbang Beban, penyimpanan Amazon S3, dan Amazon Elastic File System (Amazon EFS). Anda perlu meninjau setiap layanan ini, dan mengidentifikasi peluang penghematan biaya potensial dengan model harga yang berbeda. Beberapa di antaranya mungkin memenuhi syarat untuk RI atau SP, sedangkan beberapa lainnya mungkin hanya tersedia berdasarkan permintaan. Seperti yang ditunjukkan gambar berikut, beberapa layanan AWS dapat dibeli dengan komitmen menggunakan RI atau SP.



Layanan AWS dibeli dengan komitmen menggunakan Instans Terpesan dan Savings Plans

## Langkah implementasi

- Terapkan model harga: Menggunakan hasil analisis, beli Savings Plans, Instans Terpesan, atau terapkan Instans Spot. Jika ini adalah pembelian komitmen pertama Anda, pilih lima atau 10 rekomendasi teratas dalam daftar, lalu pantau dan analisis hasilnya selama satu atau dua bulan ke depan. AWS Cost Management Console membantu Anda dalam proses ini. Tinjau rekomendasi RI atau SP dari konsol, sesuaikan rekomendasi (jenis, pembayaran, serta jangka waktu), dan tinjau komitmen per jam (misalnya \$20 per jam), lalu tambahkan ke keranjang. Diskon berlaku secara otomatis untuk penggunaan yang memenuhi syarat. Beli sejumlah kecil diskon komitmen dalam siklus reguler (misalnya setiap 2 pekan atau setiap bulan). Implementasikan Instans Spot untuk beban kerja yang bisa dihentikan atau stateless. Terakhir, pilih instans Amazon EC2 sesuai permintaan dan alokasikan sumber daya untuk persyaratan yang tersisa.
- Siklus peninjauan beban kerja: Implementasikan siklus peninjauan untuk beban kerja yang secara spesifik menganalisis cakupan model harga. Setelah beban kerja memiliki cakupan yang diperlukan, beli diskon komitmen tambahan secara parsial (setiap beberapa bulan), atau sesuai dengan perubahan penggunaan organisasi Anda.

## Sumber daya

### Dokumen terkait:

- [Memahami rekomendasi Savings Plans Anda](#)
- [Mengakses rekomendasi Instans Terpesan](#)
- [Cara Membeli Instans Terpesan](#)
- [Opsi pembelian instans](#)
- [Instans Spot](#)
- [Model reservasi untuk layanan AWS lainnya](#)
- [Layanan Savings Plans yang Didukung](#)

### Video terkait:

- [Hemat hingga 90% dan jalankan beban kerja produksi di Spot](#)

### Contoh terkait:

- [Apa yang harus Anda pertimbangkan sebelum membeli Savings Plans?](#)



- [Bagaimana cara menggunakan Cost Explorer untuk menganalisis pengeluaran dan penggunaan saya?](#)

## COST07-BP05 Melakukan analisis model harga di tingkat akun manajemen

Periksa alat manajemen biaya dan tagihan dan lihat diskon yang direkomendasikan dengan komitmen dan reservasi untuk melakukan analisis secara teratur di tingkat akun manajemen.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Melakukan pemodelan biaya secara teratur membantu Anda mengimplementasikan peluang untuk mengoptimalkan di beberapa beban kerja. Misalnya, jika beberapa beban kerja menggunakan Instans Sesuai Permintaan, di tingkat agregat, risiko perubahan akan lebih rendah, dan implementasi diskon berbasis komitmen dapat menghasilkan biaya keseluruhan yang lebih rendah. Disarankan agar analisis dilakukan dalam siklus waktu dua minggu hingga satu bulan secara teratur. Ini memungkinkan Anda untuk melakukan pembelian penyesuaian kecil, sehingga cakupan model harga Anda terus berevolusi dengan beban kerja yang terus berubah serta komponennya.

Gunakan alat rekomendasi [AWS Cost Explorer](#) untuk menemukan peluang untuk diskon komitmen dalam akun manajemen Anda. Rekomendasi di tingkat akun manajemen dihitung dengan mempertimbangkan penggunaan di semua akun di organisasi AWS Anda yang memiliki diskon Instans Terpesan (RI) atau Savings Plans (SP). Rekomendasi tersebut juga dihitung ketika berbagi diskon diaktifkan untuk merekomendasikan komitmen yang memaksimalkan penghematan di seluruh akun.

Meskipun pembelian di tingkat akun manajemen mengoptimalkan penghematan maksimum pada banyak kasus, mungkin terdapat situasi di mana Anda dapat mempertimbangkan pembelian SP di tingkat akun tertaut, seperti ketika Anda ingin agar diskon diterapkan terlebih dahulu untuk penggunaan pada akun tertaut tersebut. Rekomendasi akun anggota dihitung di tingkat akun individu, untuk memaksimalkan penghematan untuk setiap akun terisolasi. Jika akun Anda memiliki komitmen RI dan SP, urutan penerapannya adalah:

1. RI Zona
2. RI Standar
3. RI Konvertibel
4. Instance Savings Plan

## 5. Compute Savings Plan

Jika Anda membeli SP di tingkat akun manajemen, penghematan akan diterapkan berdasarkan persentase diskon tertinggi hingga terendah. SP di tingkat akun manajemen melihat semua akun yang ditautkan dan menerapkan penghematan di tempat-tempat dengan nilai diskon tertinggi. Jika Anda ingin membatasi tempat-tempat penghematan diterapkan, Anda dapat membeli Savings Plan di tingkat akun tertaut dan setiap kali akun tersebut menjalankan layanan komputasi yang memenuhi syarat, diskon akan diterapkan di sana terlebih dahulu. Ketika akun tersebut tidak menjalankan layanan komputasi yang memenuhi syarat, diskon akan dibagikan ke akun tertaut lainnya di bawah akun manajemen yang sama. Berbagi diskon diaktifkan secara default, tetapi dapat dinonaktifkan jika diperlukan.

Dalam Keluarga Penagihan Gabungan, Savings Plans diterapkan terlebih dahulu ke penggunaan akun pemilik, kemudian ke penggunaan akun lain. Ini hanya terjadi jika Anda mengaktifkan fitur berbagi. Savings Plans Anda diterapkan ke persentase penghematan tertinggi Anda terlebih dahulu. Jika ada beberapa penggunaan dengan persentase penghematan yang sama, Savings Plans diterapkan pada penggunaan pertama dengan tarif Savings Plans terendah. Savings Plans terus berlaku sampai tidak ada lagi penggunaan yang tersisa atau komitmen Anda habis. Sisa penggunaan lainnya akan ditagih dengan tarif Sesuai Permintaan. Anda dapat menyegarkan Rekomendasi Savings Plans di Manajemen Biaya AWS untuk membuat Rekomendasi Savings Plans baru kapan saja.

Setelah menganalisis fleksibilitas instans, Anda dapat memberikan komitmen dengan mengikuti rekomendasi. Buat pemodelan biaya dengan menganalisis biaya jangka pendek beban kerja dengan potensi opsi sumber daya yang berlainan, menganalisis model harga AWS dan menyelaraskannya dengan persyaratan bisnis Anda untuk menemukan peluang [optimisasi biaya](#) dan total biaya kepemilikan.

### Langkah implementasi

Lakukan analisis diskon komitmen: Gunakan Cost Explorer di akun Anda, tinjau rekomendasi Savings Plans dan Instans Terpesan. Pastikan Anda memahami rekomendasi Savings Plans, dan perkirakan pengeluaran bulanan serta penghematan bulanan Anda. Tinjau rekomendasi di tingkat akun manajemen yang dihitung dengan mempertimbangkan penggunaan di semua akun anggota di organisasi AWS Anda yang disertai pengaktifan pembagian diskon RI atau Savings Plans, untuk memaksimalkan penghematan di seluruh akun. Anda dapat memverifikasi bahwa Anda telah mengimplementasikan rekomendasi yang benar dengan risiko dan diskon yang diperlukan dengan mengikuti lab Well-Architected.

## Sumber daya

Dokumen terkait:

- [Bagaimana cara kerja harga AWS?](#)
- [Opsi pembelian instans](#)
- [Ikhtisar Savings Plan](#)
- [Rekomendasi Savings Plan](#)
- [Mengakses rekomendasi Instans Terpesan](#)
- [Memahami rekomendasi Savings Plans Anda](#)
- [Bagaimana Savings Plans diterapkan pada penggunaan AWS Anda?](#)
- [Savings Plans dengan Penagihan Gabungan](#)
- [Mengaktifkan instans terpesan bersama dan diskon Savings Plans](#)

Video terkait:

- [Hemat hingga 90% dan jalankan beban kerja produksi di Spot](#)

Contoh terkait:

- [Lab Well-Architected AWS: Model Harga \(Tingkat 200\)](#)
- [Lab Well-Architected AWS: Analisis Model Harga \(Tingkat 200\)](#)
- [Apa yang harus saya pertimbangkan sebelum membeli Savings Plans?](#)
- [Bagaimana cara menggunakan Savings Plans bergulir untuk mengurangi risiko komitmen?](#)
- [Kapan Menggunakan Instans Spot](#)

## Merencanakan transfer data

Salah satu keunggulan cloud adalah layanan jaringan terkelolanya. Tidak ada lagi kebutuhan untuk mengelola dan mengoperasikan armada switch, router, dan perlengkapan jaringan terkait lainnya. Sumber daya jaringan di cloud digunakan dan dibayar seperti Anda membayar CPU dan penyimpanan—Anda hanya membayar sesuai yang Anda gunakan. Penggunaan sumber daya jaringan yang efisien diperlukan untuk optimasi biaya di cloud.

Praktik terbaik

- [COST08-BP01 Melakukan pemodelan transfer data](#)
- [COST08-BP02 Memilih komponen untuk mengoptimalkan biaya transfer data](#)
- [COST08-BP03 Mengimplementasikan layanan untuk mengurangi biaya transfer data](#)

## COST08-BP01 Melakukan pemodelan transfer data

Kumpulkan persyaratan organisasi dan lakukan pemodelan transfer data beban kerja dan setiap komponennya. Ini mengidentifikasi titik biaya terendah untuk persyaratan transfer data saat ini.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

### Panduan implementasi

Saat merancang solusi di cloud, biaya transfer data biasanya diabaikan karena sudah terbiasa merancang arsitektur menggunakan pusat data on-premise atau ketidaktahuan. Biaya transfer data di AWS ditentukan oleh sumber, tujuan, dan volume lalu lintas. Memperhitungkan biaya-biaya ini dalam fase desain dapat menghasilkan penghematan biaya. Sangat penting untuk memahami di mana transfer data terjadi dalam beban kerja Anda, berapa biaya transfer, dan manfaat terkaitnya agar total biaya kepemilikan (TCO) dapat diperkirakan secara akurat. Dengan demikian, Anda dapat membuat keputusan yang lebih tepat untuk mengubah atau menerima keputusan arsitektur. Misalnya, Anda memiliki konfigurasi Multi-Zona Ketersediaan yang datanya Anda replikasi antara Zona Ketersediaan tersebut.

Anda membuat model komponen layanan yang mentransfer data dalam beban kerja Anda, dan memutuskan bahwa ini adalah biaya yang dapat diterima (mirip dengan membayar komputasi dan penyimpanan di kedua Zona Ketersediaan) untuk mencapai keandalan dan ketahanan yang diperlukan. Buat model biaya untuk berbagai tingkat penggunaan. Penggunaan beban kerja dapat berubah dari waktu ke waktu, dan beberapa layanan bisa menjadi lebih hemat biaya pada level tertentu.

Saat membuat model transfer data Anda, pikirkan berapa banyak data yang diserap dan dari mana data tersebut berasal. Selain itu, pertimbangkan berapa banyak data yang diproses dan berapa banyak penyimpanan atau kapasitas komputasi yang dibutuhkan. Selama pemodelan, ikuti praktik terbaik jaringan untuk arsitektur beban kerja Anda guna mengoptimalkan potensi biaya transfer data Anda.

AWS Pricing Calculator dapat membantu Anda melihat perkiraan biaya untuk layanan AWS tertentu dan transfer data yang diharapkan. Jika Anda memiliki beban kerja yang sudah berjalan (untuk tujuan

pengujian atau di lingkungan pra-produksi), gunakan [AWS Cost Explorer](#) atau [AWS Cost and Usage Report](#) (CUR) untuk memahami dan membuat model biaya transfer data Anda. Konfigurasi bukti konsep (PoC) atau uji beban kerja Anda, dan lakukan pengujian dengan simulasi beban realistis. Anda dapat membuat model biaya untuk berbagai permintaan beban kerja.

## Langkah implementasi

- Identifikasi persyaratan: Apa tujuan utama dan persyaratan bisnis untuk transfer data yang direncanakan antara sumber dan tujuan? Apa hasil bisnis yang diharapkan akhirnya nanti? Kumpulkan persyaratan bisnis dan tentukan hasil yang diharapkan.
- Identifikasi sumber dan tujuan: Apa sumber data dan tujuan transfer data, misalnya di dalam Wilayah AWS, ke layanan AWS, atau ke internet?
  - [Transfer data dalam sebuah Wilayah AWS](#)
  - [Transfer data antara Wilayah AWS](#)
  - [Transfer data ke internet](#)
- Identifikasi klasifikasi data: Apa klasifikasi data untuk transfer data ini? Apa jenis data tersebut? Seberapa besar datanya? Seberapa sering data harus ditransfer? Apakah data sensitif?
- Identifikasi layanan atau alat AWS yang akan digunakan: Layanan AWS mana yang digunakan untuk transfer data ini? Apakah layanan yang sudah disediakan untuk beban kerja lain dapat digunakan?
- Hitung biaya transfer data: Gunakan [Harga AWS](#) pemodelan transfer data yang Anda buat sebelumnya guna menghitung biaya transfer data untuk beban kerja. Hitung biaya transfer data di berbagai tingkat penggunaan, baik saat penggunaan beban kerja berkurang maupun bertambah. Jika ada banyak opsi arsitektur beban kerja, maka hitung biaya untuk setiap opsi sebagai perbandingan.
- Hubungkan biaya dengan hasil: Untuk setiap biaya transfer data yang dikenakan, tentukan hasil yang akan dicapai untuk beban kerja. Jika transfer antarkomponen, hasilnya mungkin untuk pemisahan. Jika dilakukan di antara Zona Ketersediaan, tujuannya mungkin untuk redundansi.
- Buat pemodelan transfer data: Setelah mengumpulkan semua informasi, buat pemodelan transfer data dasar konseptual untuk sejumlah kasus penggunaan dan beban kerja yang berbeda.

## Sumber daya

### Dokumen terkait:

- [Solusi caching AWS](#)

- [Harga AWS](#)
- [Harga Amazon EC2](#)
- [Harga Amazon VPC](#)
- [Memahami biaya transfer data](#)

Video terkait:

- [Memantau dan Mengoptimalkan Biaya Transfer Data Anda](#)
- [S3 Transfer Acceleration](#)

Contoh terkait:

- [Ikhtisar Biaya Transfer Data untuk Arsitektur Umum](#)
- [Panduan Preskriptif AWS untuk Jaringan](#)

## COST08-BP02 Memilih komponen untuk mengoptimalkan biaya transfer data

Semua komponen diseleksi, dan arsitektur didesain untuk mengurangi biaya transfer data. Termasuk di dalamnya adalah menggunakan komponen seperti optimalisasi jaringan area luas (WAN) dan konfigurasi Multi-Zona Ketersediaan (AZ)

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Perancangan untuk transfer data meminimalkan biaya transfer data. Ini mungkin melibatkan penggunaan jaringan pengiriman untuk menemukan data yang lebih dekat dengan pengguna, atau penggunaan tautan jaringan khusus dari lokasi Anda ke AWS. Anda juga dapat menggunakan optimalisasi WAN dan optimalisasi aplikasi untuk mengurangi jumlah data yang ditransfer antar komponen.

Saat mentransfer data ke atau di dalam AWS Cloud, penting untuk mengetahui tujuan berdasarkan berbagai kasus penggunaan, sifat data, dan sumber daya jaringan yang tersedia untuk memilih layanan AWS yang tepat guna mengoptimalkan transfer data. AWS menawarkan berbagai layanan transfer data yang disesuaikan untuk beragam persyaratan migrasi data. Pilih opsi [penyimpanan data](#) dan [transfer data](#) yang tepat berdasarkan kebutuhan bisnis dalam organisasi Anda.

Saat merencanakan atau meninjau arsitektur beban kerja Anda, pertimbangkan hal berikut:

- Gunakan titik akhir VPC dalam AWS: Titik akhir VPC memungkinkan koneksi pribadi antara VPC Anda dan layanan AWS yang didukung. Hal ini memungkinkan Anda untuk menghindari penggunaan internet publik, yang dapat menimbulkan biaya transfer data.
- Gunakan gateway NAT: Gunakan [gateway NAT](#) agar instans di subnet privat dapat terhubung ke internet atau ke layanan di luar VPC Anda. Periksa apakah sumber daya di belakang gateway NAT yang mengirimkan lalu lintas terbanyak berada di Zona Ketersediaan yang sama dengan gateway NAT. Jika tidak, buat gateway NAT baru di Zona Ketersediaan yang sama dengan sumber daya untuk mengurangi biaya transfer data lintas AZ.
- Gunakan AWS Direct Connect: AWS Direct Connect melewati internet publik dan membuat koneksi privat langsung antara jaringan on-premise Anda dan AWS. Hal ini bisa lebih hemat biaya dan konsisten daripada mentransfer data dalam jumlah besar melalui internet.
- Hindari transfer data melintasi batas-batas Regional: Transfer data antar-Wilayah AWS (dari satu Wilayah ke Wilayah lainnya) biasanya dikenakan biaya. Jika ingin memilih jalur multi-Wilayah, sebaiknya putuskan dengan cermat. Untuk detail selengkapnya, lihat [Skenario Multi-Wilayah](#).
- Pantau transfer data: Gunakan Amazon CloudWatch dan [log aliran VPC](#) untuk menangkap detail tentang transfer data dan penggunaan jaringan Anda. Analisis informasi lalu lintas jaringan yang ditangkap di VPC Anda, seperti alamat IP atau jangkauan ke dan dari antarmuka jaringan.
- Analisis penggunaan jaringan Anda: Gunakan alat pengukuran dan pelaporan seperti AWS Cost Explorer, CUDOS Dashboards, atau CloudWatch untuk memahami biaya transfer data beban kerja Anda.

## Langkah implementasi

- Pilih komponen untuk transfer data: Gunakan pemodelan transfer data yang dijelaskan di [COST08-BP01 Melakukan pemodelan transfer data](#), fokuslah pada di mana biaya terbesar saat ini untuk transfer data, atau di mana kira-kira biaya terbesarnya jika penggunaan beban kerja berubah. Cari arsitektur alternatif, atau komponen tambahan yang menghapus atau mengurangi kebutuhan untuk transfer data, atau menghemat biayanya.

## Sumber daya

### Praktik Terbaik Terkait:

- [COST08-BP01 Melakukan pemodelan transfer data](#)

- [COST08-BP03 Mengimplementasikan layanan untuk mengurangi biaya transfer data](#)

Dokumen terkait:

- [Migrasi Data Cloud](#)
- [Solusi caching AWS](#)
- [Kirim konten lebih cepat dengan Amazon CloudFront](#)

Contoh terkait:

- [Ikhtisar Biaya Transfer Data untuk Arsitektur Umum](#)
- [AWS Network Optimization Tips](#)
- [Mengoptimalkan performa dan mengurangi biaya untuk analitik jaringan dengan VPC Flow Log dalam format Apache Parquet](#)

## COST08-BP03 Mengimplementasikan layanan untuk mengurangi biaya transfer data

Implementasikan layanan untuk mengurangi transfer data. Misalnya, gunakan lokasi edge atau jaringan pengiriman konten (CDN) untuk mengirimkan konten ke pengguna akhir, membangun lapisan caching di depan server aplikasi atau basis data Anda, dan gunakan koneksi jaringan khusus alih-alih VPN untuk terhubung ke cloud.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Ada berbagai layanan AWS yang dapat membantu Anda mengoptimalkan penggunaan transfer data jaringan Anda. Tergantung komponen beban kerja, jenis, dan arsitektur cloud Anda, semua layanan ini dapat membantu Anda mengompresi, melakukan caching, serta berbagi dan mendistribusikan lalu lintas Anda di cloud.

- [Amazon CloudFront](#) adalah jaringan pengiriman konten global yang mengirimkan data dengan latensi rendah dan kecepatan transfer yang tinggi. Layanan ini meng-cache data di lokasi edge di seluruh dunia, yang mengurangi beban di sumber daya Anda. Dengan menggunakan CloudFront, Anda dapat mengurangi upaya administratif dalam pengiriman konten ke sejumlah besar pengguna



di seluruh dunia, dengan latensi minimum. Paket [penghematan keamanan](#) dapat membantu Anda menghemat hingga 30% pada penggunaan CloudFront Anda jika Anda berencana meningkatkan penggunaan Anda dari waktu ke waktu.

- [AWS Direct Connect](#) memungkinkan Anda untuk menetapkan koneksi jaringan khusus ke AWS. Ini dapat mengurangi biaya jaringan, meningkatkan bandwidth, dan menyediakan pengalaman jaringan yang lebih konsisten daripada koneksi berbasis internet.
- [AWS VPN](#) memungkinkan Anda untuk membangun koneksi aman dan privat antara jaringan privat Anda dan jaringan global AWS. Layanan ini ideal untuk kantor kecil atau partner bisnis karena menyediakan konektivitas yang disederhanakan, serta dengan layanan yang dikelola penuh dan elastis.
- [Titik akhir VPC](#) memungkinkan konektivitas antara layanan AWS lewat jaringan privat dan dapat digunakan untuk mengurangi transfer data publik serta [Gateway NAT](#). [Titik akhir VPC gateway](#) tidak memiliki tarif per jam, serta mendukung Amazon S3 dan Amazon DynamoDB. [Titik akhir VPC antarmuka](#) disediakan oleh [AWS PrivateLink](#) dan memiliki tarif per jam serta biaya penggunaan per GB.
- [biaya gateway NAT](#) menyediakan penskalaan dan manajemen bawaan untuk mengurangi biaya dibandingkan dengan instans NAT mandiri. Tempatkan gateway NAT di dalam Zona Ketersediaan yang sama dengan instans lalu lintas tinggi dan pertimbangkan menggunakan titik akhir VPC untuk instans yang perlu mengakses Amazon DynamoDB atau Amazon S3 guna mengurangi biaya transfer dan pemrosesan data.
- Gunakan perangkat [AWS Snow Family](#) yang memiliki sumber daya komputasi untuk mengumpulkan dan memproses data di edge. Perangkat AWS Snow Family ([Snowcone](#), [Snowball](#) dan [Snowmobile](#)) memungkinkan Anda memindahkan data seukuran petabyte ke AWS Cloud secara hemat biaya dan offline.

## Langkah implementasi

- Implementasikan layanan: Pilih layanan jaringan AWS yang berlaku berdasarkan layanan Anda, jenis beban kerja yang menggunakan pemodelan transfer data, dan meninjau Log Aliran VPC. Amati area dengan biaya terbesar dan alur volume tertinggi. Tinjau layanan AWS dan nilai apakah ada layanan yang mengurangi atau menghapus transfer, khususnya jaringan dan pengiriman konten. Cari juga layanan caching yang menyediakan akses berulang ke data, atau data dalam jumlah besar.

## Sumber daya

### Dokumen terkait:

- [AWS Direct Connect](#)
- [AWS Jelajahi Produk Kami](#)
- [solusi caching AWS](#)
- [Amazon CloudFront](#)
- [AWS Snow Family](#)
- [Paket Penghematan Keamanan Amazon CloudFront](#)

### Video terkait:

- [Memantau dan Mengoptimalkan Biaya Transfer Data Anda](#)
- [Serial Pengoptimalan Biaya AWS: CloudFront](#)
- [Bagaimana cara mengurangi biaya transfer data untuk gateway NAT saya?](#)

### Contoh terkait:

- [Cara melakukan tolak bayar layanan bersama: Contoh AWS Transit Gateway](#)
- [Memahami detail transfer data AWS secara mendalam dari laporan biaya dan penggunaan menggunakan kueri Athena dan QuickSight](#)
- [Ikhtisar Biaya Transfer Data untuk Arsitektur Umum](#)
- [Menggunakan AWS Cost Explorer untuk menganalisis biaya transfer data](#)
- [Mengoptimalkan Biaya arsitektur AWS Anda dengan memanfaatkan fitur Amazon CloudFront](#)
- [Bagaimana cara mengurangi biaya transfer data untuk gateway NAT saya?](#)

# Kelola sumber daya pasokan dan permintaan

Ketika Anda beralih ke cloud, Anda hanya perlu membayar sesuai dengan yang Anda butuhkan. Anda dapat memasok sumber daya sesuai dengan permintaan beban kerja pada saat dibutuhkan — sehingga menghilangkan pemborosan biaya dan penyediaan berlebih yang tidak terpakai. Anda juga dapat mengubah permintaan dengan throttle, buffer, atau antrean, untuk memperlancar permintaan serta memenuhinya dengan sumber daya yang lebih sedikit.

Manfaat ekonomi dari pemasokan yang tepat waktu harus diimbangi dengan kebutuhan penyediaan untuk memperhitungkan kegagalan sumber daya, ketersediaan tinggi, dan waktu penyediaan. Bergantung pada apakah permintaan Anda sudah pasti atau bervariasi, rencanakan untuk membuat metrik dan otomatisasi yang membantu meminimalkan manajemen lingkungan Anda – bahkan saat skala berubah. Saat mengubah permintaan, Anda harus mengetahui penundaan maksimum yang dapat diterima dan dapat ditangani beban kerja.

Di AWS, Anda dapat menggunakan berbagai pendekatan yang berbeda untuk mengelola sumber daya pasokan dan permintaan. Praktik terbaik berikut akan menjelaskan cara menggunakan pendekatan ini:

## Praktik terbaik

- [COST09-BP01 Melakukan analisis pada permintaan beban kerja](#)
- [COST09-BP02 Implementasikan buffer atau throttle untuk mengelola permintaan](#)
- [COST09-BP03 Menyediakan sumber daya secara dinamis](#)

## COST09-BP01 Melakukan analisis pada permintaan beban kerja

Analisis permintaan beban kerja dari waktu ke waktu. Verifikasikan bahwa analisis ini mencakup tren musiman dan merepresentasikan secara akurat kondisi operasi di sepanjang masa pakai beban kerja penuh. Upaya analisis harus mencerminkan potensi manfaat, misalnya, waktu yang digunakan sebanding dengan biaya beban kerja.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Tinggi

## Panduan implementasi

Analisis permintaan beban kerja untuk komputasi cloud melibatkan pemahaman pola dan karakteristik tugas komputasi yang dimulai di lingkungan cloud. Analisis ini membantu pengguna

mengoptimalkan alokasi sumber daya, mengelola biaya, dan memastikan tingkat kinerja yang diperlukan.

Ketahui persyaratan beban kerja. Persyaratan organisasi Anda harus menunjukkan waktu respons beban kerja untuk permintaan. Waktu respons dapat digunakan untuk menentukan apakah permintaan dikelola, atau apakah pasokan sumber daya harus berubah untuk memenuhi permintaan.

Analisis harus mencakup prediktabilitas dan pengulangan permintaan, tingkat perubahan dalam permintaan, serta jumlah perubahan dalam permintaan. Lakukan analisis dalam periode yang cukup lama sehingga menyertakan variasi musiman apa pun, seperti pemrosesan akhir bulan atau puncak liburan.

Upaya analisis harus mencerminkan potensi manfaat implementasi penskalaan. Amati perkiraan total biaya komponen serta peningkatan atau penurunan penggunaan dan biaya di sepanjang masa pakai beban kerja.

Berikut ini adalah beberapa aspek kunci yang perlu dipertimbangkan saat melakukan analisis permintaan beban kerja untuk komputasi cloud:

1. Metrik pemanfaatan sumber daya dan performa: Analisis bagaimana sumber daya AWS digunakan dari waktu ke waktu. Tentukan pola penggunaan puncak dan di luar puncak untuk mengoptimalkan alokasi sumber daya dan strategi penskalaan. Pantau metrik kinerja seperti waktu respons, latensi, throughput, dan tingkat kesalahan. Metrik-metrik ini membantu menilai kondisi dan efisiensi infrastruktur cloud secara keseluruhan.
2. Perilaku penskalaan pengguna dan aplikasi: Pahami perilaku pengguna dan bagaimana hal tersebut memengaruhi permintaan beban kerja. Pemeriksaan pola lalu lintas pengguna bermanfaat dalam meningkatkan pengiriman konten dan responsivitas aplikasi. Analisis bagaimana beban kerja diskalakan seiring meningkatnya permintaan. Tentukan apakah parameter penskalaan otomatis dikonfigurasi dengan benar dan efektif untuk menangani fluktuasi beban.
3. Jenis beban kerja: Identifikasi berbagai jenis beban kerja yang berjalan di cloud, seperti pemrosesan batch, pemrosesan data waktu nyata, aplikasi web, basis data, atau machine learning. Setiap jenis beban kerja mungkin memiliki persyaratan sumber daya dan profil kinerja yang berbeda-beda.
4. Perjanjian tingkat layanan (SLA): Bandingkan kinerja aktual dengan SLA untuk memastikan kepatuhan dan mengidentifikasi area yang perlu ditingkatkan.

Anda dapat menggunakan [Amazon CloudWatch](#) untuk mengumpulkan dan melacak metrik, memantau file log, mengatur alarm, dan secara otomatis bereaksi terhadap perubahan sumber daya

Anda AWS. Anda dapat menggunakan Amazon CloudWatch untuk memperoleh visibilitas di seluruh sistem tentang pemanfaatan sumber daya, kinerja aplikasi, dan kondisi operasional.

Dengan [AWS Trusted Advisor](#), Anda dapat menyediakan sumber daya Anda sesuai praktik terbaik untuk meningkatkan kinerja dan keandalan sistem, meningkatkan keamanan, dan mencari peluang penghematan biaya. Anda juga dapat menonaktifkan instans non-produksi dan menggunakan Amazon CloudWatch dan Auto Scaling agar sesuai dengan peningkatan atau penurunan permintaan.

Akhirnya, Anda dapat menggunakan [AWS Cost Explorer](#) atau [Amazon QuickSight](#) dengan file AWS Cost and Usage Report (CUR) atau log aplikasi Anda untuk melakukan analisis permintaan beban kerja lanjutan.

Secara keseluruhan, analisis permintaan beban kerja yang komprehensif memungkinkan organisasi untuk mengambil keputusan yang berdasar tentang penyediaan sumber daya, penskalaan, dan pengoptimalan, yang menghasilkan perbaikan kinerja, efisiensi biaya, dan kepuasan pengguna.

## Langkah implementasi

- Analisis data beban kerja yang ada: Analisis data dari beban kerja yang ada, versi beban kerja sebelumnya, atau pola penggunaan yang diprediksi. Gunakan Amazon CloudWatch, file log, dan data pemantauan untuk mendapatkan wawasan tentang bagaimana beban kerja digunakan. Analisis siklus beban kerja secara penuh, dan kumpulkan data untuk perubahan musiman apa pun seperti peristiwa akhir bulan atau akhir tahun. Upaya yang tercermin dalam analisis harus mencerminkan karakteristik beban kerja. Upaya terbesar harus ditempatkan pada beban kerja bernilai tinggi dengan perubahan permintaan terbesar. Upaya terkecil harus ditempatkan pada beban kerja bernilai rendah dengan perubahan permintaan yang minim.
- Prakirakan pengaruh luar: Temui anggota tim dari seluruh organisasi yang dapat memengaruhi atau mengubah permintaan pada beban kerja. Tim umum terdiri dari penjualan, pemasaran, atau pengembangan bisnis. Bekerjalah dengan mereka untuk mengetahui siklus operasi mereka, dan apakah ada peristiwa yang akan mengubah permintaan beban kerja. Prakirakan permintaan beban kerja dengan data ini.

## Sumber daya

Dokumen terkait:

- [Amazon CloudWatch](#)
- [AWS Trusted Advisor](#)

- [AWS X-Ray](#)
- [AWS Auto Scaling](#)
- [Penjadwal Instans AWS](#)
- [Memulai dengan Amazon SQS](#)
- [AWS Cost Explorer](#)
- [Amazon QuickSight](#)

Video terkait:

Contoh terkait:

- [Pantau, Lacak, dan Analisis untuk pengoptimalan biaya](#)
- [Mencari dan menganalisis log di CloudWatch](#)

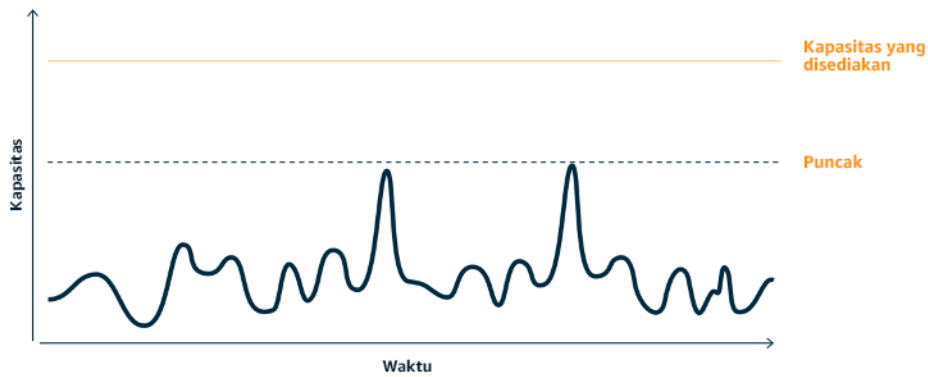
## COST09-BP02 Implementasikan buffer atau throttle untuk mengelola permintaan

Buffering dan throttling memodifikasi permintaan di beban kerja Anda, meratakan fluktuasi. Implementasikan throttling ketika klien Anda mencoba ulang. Implementasikan buffering untuk menyimpan permintaan dan menunda pemrosesan ke lain waktu. Pastikan throttle dan buffer Anda didesain sehingga klien menerima respons dalam waktu yang diperlukan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

Mengimplementasikan buffer atau throttle sangat penting dalam komputasi cloud untuk mengelola permintaan dan mengurangi penyediaan kapasitas yang diperlukan untuk beban kerja Anda. Untuk performa yang optimal, penting agar Anda mengukur total permintaan, termasuk puncak, laju perubahan permintaan, dan waktu respons yang diperlukan. Ketika klien memiliki kemampuan untuk mengirim ulang permintaan mereka, maka penerapan throttling menjadi praktis. Sebaliknya, untuk klien yang tidak memiliki fungsi coba ulang, mengimplementasikan solusi buffer merupakan pendekatan yang ideal. Buffer tersebut itu merampingkan masuknya permintaan dan mengoptimalkan interaksi aplikasi dengan kecepatan operasional yang bervariasi.



Kurva permintaan dengan dua puncak terpisah yang memerlukan penyediaan kapasitas tinggi

Asumsikan beban kerja dengan kurva permintaan yang ditunjukkan pada gambar berikut. Beban kerja ini memiliki dua puncak. Untuk menangani puncak-puncak ini, kapasitas sumber daya sebagaimana ditunjukkan oleh garis oranye disediakan. Sumber daya dan energi yang digunakan untuk beban kerja ini tidak ditunjukkan dengan area kurva permintaan, tetapi dengan area garis kapasitas yang disediakan, karena kapasitas yang disediakan diperlukan untuk menangani kedua puncak ini. Meratakan kurva permintaan beban kerja dapat membantu Anda mengurangi kapasitas tersedia untuk beban kerja dan mengurangi dampaknya pada lingkungan. Untuk memperlancar puncak, pertimbangkan untuk mengimplementasikan solusi throttling atau buffering.

Untuk memahaminya lebih lanjut, mari kita bahas apa itu throttling dan buffering.

**Throttling:** Jika sumber permintaan memiliki kemampuan coba ulang, Anda dapat mengimplementasikan throttling. Throttling memberi tahu sumber bahwa jika sistem saat ini tidak dapat melayani permintaan, sumber harus mencoba lagi nanti. Sumber menunggu untuk jangka waktu tertentu, lalu mencoba kembali permintaan. Implementasi throttling memiliki manfaat membatasi jumlah maksimum sumber daya dan biaya beban kerja. Di AWS, Anda dapat menggunakan [Amazon API Gateway](#) untuk mengimplementasikan throttling.

**Berbasis buffer:** Pendekatan berbasis buffer menggunakan produsen (komponen yang mengirim pesan ke antrian), konsumen (komponen yang menerima pesan dari antrian), dan antrian (yang menyimpan pesan) untuk menyimpan pesan. Pesan dibaca oleh konsumen dan diproses, sehingga pesan dapat dijalankan dengan tingkat yang memenuhi persyaratan bisnis konsumen. Dengan menggunakan metodologi buffer-sentris, pesan dari produsen disimpan dalam antrian atau aliran, siap diakses oleh konsumen dalam kecepatan yang selaras dengan tuntutan operasional mereka.

Di AWS, Anda dapat memilih dari berbagai layanan untuk mengimplementasikan pendekatan buffering. [Amazon Simple Queue Service\(Amazon SQS\)](#) adalah layanan terkelola yang menyediakan

antrean yang memungkinkan satu konsumen membaca pesan individual. [Amazon Kinesis](#) menyediakan aliran yang memungkinkan banyak konsumen membaca pesan yang sama.

Buffering dan throttling dapat memperlancar setiap puncak dengan memodifikasi permintaan pada beban kerja Anda. Gunakan throttling saat klien mencoba kembali tindakan dan gunakan buffering untuk menahan permintaan dan memprosesnya nanti. Ketika menggunakan pendekatan berbasis buffer, rancang dan konfigurasi beban kerja untuk melayani permintaan dalam waktu yang diperlukan, verifikasi bahwa Anda dapat menangani permintaan kerja duplikat. Analisis permintaan secara keseluruhan, tingkat perubahan, dan waktu respons yang diperlukan untuk ukuran throttle atau buffer yang tepat.

## Langkah implementasi

- Analisis persyaratan klien: Analisis permintaan klien untuk menentukan apakah mereka dapat melakukan percobaan ulang. Untuk klien yang tidak dapat melakukan percobaan ulang, buffer perlu diimplementasikan. Analisis permintaan secara keseluruhan, tingkat perubahan, dan waktu respons yang diperlukan untuk menentukan ukuran throttle atau buffer yang diperlukan.
- Implementasikan buffer atau throttle: Implementasikan buffer atau throttle dalam beban kerja. Antrean seperti Amazon Simple Queue Service (Amazon SQS) dapat memberikan buffer untuk komponen beban kerja Anda. Amazon API Gateway dapat memberikan throttling untuk komponen beban kerja Anda.

## Sumber daya

Praktik Terbaik Terkait:

- [SUS02-BP06 Mengimplementasikan buffering atau throttling untuk meratakan kurva permintaan](#)
- [REL05-BP02 Membatasi \(throttling\) permintaan](#)

Dokumen terkait:

- [AWS Auto Scaling](#)
- [Penjadwal Instans AWS](#)
- [Amazon API Gateway](#)
- [Amazon Simple Queue Service](#)
- [Mulai menggunakan Amazon SQS](#)



- [Amazon Kinesis](#)

Video terkait:

- [Memilih Layanan Olah Pesan yang Tepat untuk Aplikasi Terdistribusi Anda](#)

Contoh terkait:

- [Mengelola dan memanfaatkan throttling API di dalam beban kerja Anda](#)
- [Throttling a tiered, multi-tenant REST API at scale using API Gateway](#)
- [Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway](#)
- [Integrasi Aplikasi Menggunakan Antrean dan Pesan](#)

## COST09-BP03 Menyediakan sumber daya secara dinamis

Sumber daya disediakan sesuai dengan perencanaan. Penyediaan dapat berdasarkan permintaan, yaitu melalui penskalaan otomatis, atau berdasarkan waktu, yaitu permintaan dapat diprediksi dan sumber daya disediakan berdasarkan waktu. Metode ini dapat meminimalkan penyediaan yang terlalu banyak atau terlalu sedikit.

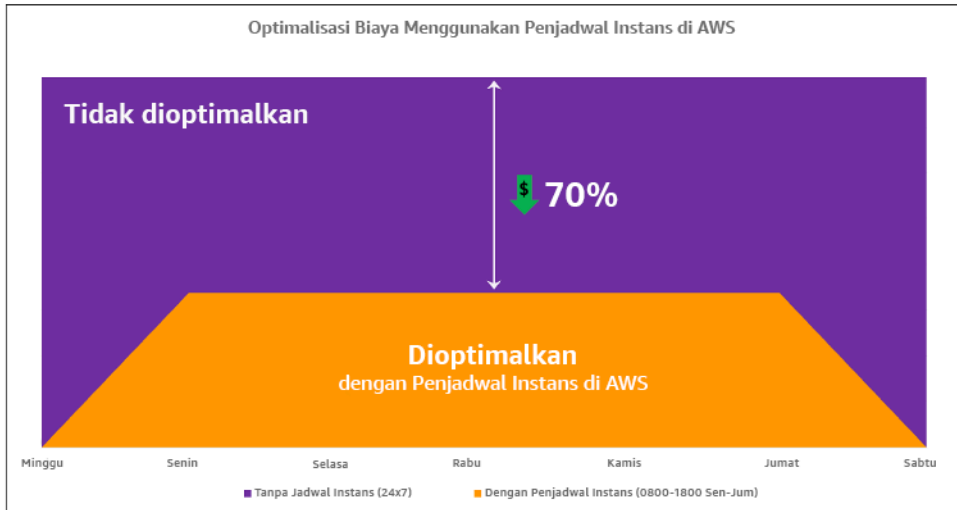
Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

### Panduan implementasi

Ada sejumlah cara bagi pelanggan AWS untuk meningkatkan sumber daya yang tersedia untuk aplikasi mereka dan menyediakan sumber daya untuk memenuhi permintaan. Salah satu opsinya adalah dengan menggunakan Penjadwal Instans AWS, yang mengotomatiskan pengaktifan dan penghentian instans Amazon Elastic Compute Cloud (Amazon EC2) dan Amazon Relational Database Service (Amazon RDS). Opsi lainnya adalah menggunakan AWS Auto Scaling, yang memungkinkan Anda untuk secara otomatis menskalakan sumber daya komputasi berdasarkan permintaan aplikasi atau layanan Anda. Memasok sumber daya berdasarkan permintaan akan memungkinkan Anda membayar sumber daya yang Anda gunakan saja, menekan biaya dengan meluncurkan sumber daya hanya saat dibutuhkan, dan menghentikannya saat tidak dibutuhkan.

[Penjadwal Instans AWS](#) memungkinkan Anda untuk mengonfigurasi penghentian dan pengaktifan instans Amazon EC2 dan Amazon RDS Anda pada waktu yang telah ditentukan sehingga Anda

dapat memenuhi permintaan untuk sumber daya yang sama dalam pola waktu yang konsisten, misalnya setiap hari pengguna mengakses instans Amazon EC2 pada pukul delapan pagi dan mereka tidak perlu mengaksesnya setelah pukul enam sore. Solusi ini membantu mengurangi biaya operasional dengan menghentikan sumber daya yang tidak digunakan dan memulainya saat diperlukan.



Optimalisasi biaya dengan Penjadwal Instans AWS.

Anda juga dapat dengan mudah mengonfigurasi jadwal untuk instans Amazon EC2 Anda di seluruh akun dan Wilayah Anda dengan antarmuka pengguna (UI) yang sederhana menggunakan Pengaturan Cepat AWS Systems Manager. Anda dapat menjadwalkan instans Amazon EC2 atau Amazon RDS dengan Penjadwal Instans AWS dan Anda dapat menghentikan dan memulai instans yang ada. Namun, Anda tidak dapat menghentikan dan memulai instans yang merupakan bagian dari grup Auto Scaling (ASG) Anda atau yang mengelola layanan seperti Amazon Redshift atau Amazon OpenSearch Service. Grup Auto Scaling memiliki penjadwalan sendiri untuk instans di dalam grup dan instans-instans tersebut adalah hasil buatan.

[AWS Auto Scaling](#) membantu menyesuaikan kapasitas Anda untuk menjaga kinerja yang stabil dan terprediksi dengan biaya serendah mungkin untuk memenuhi permintaan yang berubah. Ini adalah layanan gratis dan terkelola penuh untuk menskalakan kapasitas aplikasi Anda yang terintegrasi dengan instans Amazon EC2 dan Armada Spot, Amazon ECS, Amazon DynamoDB, dan Amazon Aurora. Auto Scaling menyediakan pencarian sumber daya secara otomatis yang dapat dikonfigurasi untuk membantu Anda menemukan sumber daya dalam beban kerja Anda, dilengkapi dengan strategi penskalaan bawaan untuk mengoptimalkan kinerja, biaya, atau keseimbangan antara keduanya, serta memberikan penskalaan prediktif untuk membantu menangani lonjakan yang rutin terjadi.

Ada beberapa opsi penskalaan yang tersedia untuk menskalakan grup Auto Scaling Anda:

- Pertahankan level instans saat ini setiap saat
- Skalikan secara manual
- Skalikan berdasarkan jadwal
- Skalikan berdasarkan permintaan
- Gunakan penskalaan prediktif

Kebijakan Auto Scaling berbeda-beda dan dapat dikategorikan ke dalam kebijakan penskalaan dinamis dan terjadwal. Kebijakan dinamis ditujukan untuk penskalaan manual atau dinamis, sedangkan kebijakan terjadwal ditujukan untuk penskalaan terjadwal atau prediktif. Anda dapat menggunakan kebijakan penskalaan untuk penskalaan dinamis, terjadwal, dan prediktif. Anda juga dapat menggunakan metrik atau alarm dari [Amazon CloudWatch](#) untuk memicu peristiwa penskalaan untuk beban kerja Anda. Sebaiknya Anda menggunakan [templat peluncuran](#), yang memungkinkan Anda untuk mengakses fitur dan perbaikan terbaru. Tidak semua fitur Auto Scaling tersedia saat Anda menggunakan konfigurasi peluncuran. Misalnya, Anda tidak dapat membuat grup Auto Scaling yang meluncurkan Instans Spot dan Sesuai Permintaan atau yang menentukan beberapa jenis instans. Anda harus menggunakan templat peluncuran untuk mengonfigurasi fitur-fitur ini. Saat menggunakan templat peluncuran, kami sarankan Anda melakukan versioning masing-masing. Dengan versioning templat peluncuran, Anda dapat membuat subset dari set lengkap parameter. Kemudian, Anda dapat menggunakannya kembali untuk membuat versi lain dari templat peluncuran yang sama.

Anda dapat menggunakan AWS Auto Scaling atau menggabungkan penskalaan di dalam kode Anda dengan [API atau SDK AWS](#). Hal ini akan menghemat biaya beban kerja secara keseluruhan dengan menghilangkan biaya operasional yang diperlukan untuk membuat perubahan secara manual di lingkungan Anda, dan perubahan dapat dilakukan dengan lebih cepat. Hal ini juga menyesuaikan pengadaan sumber daya beban kerja Anda dengan permintaan Anda kapan saja. Agar dapat mengikuti praktik terbaik ini dan menyediakan sumber daya secara dinamis untuk organisasi Anda, Anda harus memahami penskalaan horizontal dan vertikal di AWS Cloud, serta sifat aplikasi yang berjalan di instans Amazon EC2. Sebaiknya tim Manajemen Keuangan Cloud Anda bekerja dengan tim teknis untuk mengikuti praktik terbaik ini.

[Elastic Load Balancing \(Elastic Load Balancing\)](#) membantu penskalaan dengan mendistribusikan permintaan ke beberapa sumber daya. Dengan menggunakan ASG dan Elastic Load Balancing, Anda dapat mengelola permintaan masuk dengan merutekan lalu lintas secara optimal sehingga tidak ada instans yang kewalahan di dalam grup Auto Scaling. Permintaan akan didistribusikan di

antara semua target dari grup target secara round-robin tanpa mempertimbangkan kapasitas atau pemanfaatan.

Metrik yang umum dapat berupa metrik Amazon EC2 standar, seperti pemanfaatan CPU, throughput jaringan, dan latensi respons atau permintaan terobservasi Elastic Load Balancing. Jika memungkinkan, Anda harus menggunakan metrik yang menggambarkan pengalaman pelanggan, biasanya berupa metrik kustom yang berasal dari kode aplikasi di dalam beban kerja Anda. Untuk menguraikan cara memenuhi permintaan secara dinamis dalam dokumen ini, kami akan mengelompokkan Auto Scaling ke dalam dua kategori yakni model penyediaan berdasarkan permintaan dan berdasarkan waktu, dan kami akan menjelaskan masing-masing.

Penyediaan berdasarkan permintaan: Manfaatkan elastisitas cloud untuk memasok sumber daya guna memenuhi permintaan yang berubah dengan mengandalkan kondisi permintaan yang mendekati waktu nyata. Untuk penyediaan berdasarkan permintaan, gunakan fitur layanan atau API untuk mengelompokkan jumlah sumber daya cloud secara terprogram di arsitektur Anda. Hal ini memungkinkan Anda untuk menskalakan komponen di arsitektur Anda, serta meningkatkan jumlah sumber daya saat permintaan melonjak guna mempertahankan kinerja, dan mengurangi kapasitas saat permintaan menurun untuk mengurangi biaya.

## Pasokan Berbasis Permintaan (Kebijakan Penskalaan Dinamis)



Kebijakan penskalaan dinamis berdasarkan permintaan

- Penskalaan Sederhana/Langkah: Memantau metrik dan menambahkan/menghapus instans sesuai langkah yang ditentukan oleh pelanggan secara manual.

- **Pelacakan Target:** Mekanisme kontrol mirip termostat yang secara otomatis menambahkan atau menghapus instans untuk mempertahankan metrik pada target yang ditentukan pelanggan.

Jika Anda menggunakan pendekatan berdasarkan permintaan saat merancang, selalu pertimbangkan dua hal yang utama. Pertama, ketahui seberapa cepat Anda harus menyediakan sumber daya baru. Kedua, ketahui bahwa ukuran margin antara penyediaan dan permintaan akan berubah. Anda harus siap menangani rasio perubahan dalam permintaan dan juga siap dengan kegagalan sumber daya.

**Penyediaan berdasarkan waktu:** Pendekatan berdasarkan waktu selaras dengan kapasitas sumber daya untuk permintaan yang dapat diprediksi atau telah ditentukan berdasarkan waktu. Pendekatan ini biasanya tidak bergantung pada tingkat pemanfaatan sumber daya. Pendekatan berdasarkan waktu memastikan ketersediaan sumber daya pada waktu tertentu saat diperlukan serta dapat disediakan tanpa penundaan yang disebabkan sistem dan prosedur menghidupkan atau pemeriksaan konsistensi. Menggunakan pendekatan berdasarkan waktu, Anda dapat menyediakan sumber daya tambahan atau meningkatkan kapasitas selama periode sibuk.

### Pasokan Berbasis Waktu (Kebijakan Penskalaan Terjadwal dan Prediktif)



Kebijakan penskalaan berdasarkan waktu

Anda dapat menggunakan penskalaan otomatis yang terjadwal atau prediktif untuk menerapkan pendekatan berdasarkan waktu. Beban kerja dapat dijadwalkan untuk penskalaan ke luar atau ke

dalam pada waktu yang ditentukan (misalnya, awal jam kerja), sehingga sumber daya tersedia saat pengguna datang atau permintaan meningkat. Penskalaan prediktif menggunakan pola untuk menskalakan ke luar sedangkan penskalaan terjadwal menggunakan waktu yang ditetapkan di awal untuk menskalakan ke luar. Anda juga dapat menggunakan [strategi pemilihan jenis instans berbasis atribut \(ABS\)](#) di grup Auto Scaling, yang memungkinkan Anda untuk menyatakan persyaratan instans Anda dalam bentuk rangkaian atribut, seperti vCPU, memori, dan penyimpanan. Ini juga memungkinkan Anda untuk menggunakan tipe instans generasi lebih baru secara otomatis ketika tipe tersebut dirilis dan mengakses cakupan kapasitas yang lebih luas dengan Instans Spot Amazon EC2. Armada Amazon EC2 dan Amazon EC2 Auto Scaling memilih dan meluncurkan instans yang cocok dengan atribut yang ditentukan, sehingga tipe instans tidak perlu dipilih secara manual.

Anda juga dapat memanfaatkan [API serta SDK AWS](#) dan [AWS CloudFormation](#) untuk menyiapkan atau menarik secara otomatis seluruh lingkungan saat Anda memerlukannya. Pendekatan ini ideal untuk lingkungan pengujian atau pengembangan yang hanya berjalan pada jam kerja atau periode waktu yang ditentukan. Anda dapat menggunakan API untuk menskalakan ukuran sumber daya di dalam suatu lingkungan (penskalaan vertikal). Misalnya, Anda dapat menaikkan skala beban kerja produksi dengan mengubah ukuran atau kelas instans. Hal ini dapat dicapai dengan menghentikan lalu memulai instans, kemudian memilih kelas dan ukuran instans yang berbeda. Teknik ini juga dapat diterapkan ke sumber daya lain, seperti Volume Elastis Amazon EBS, yang dapat diubah untuk meningkatkan ukuran, menyesuaikan kinerja (IOPS), atau mengubah jenis volume saat sedang digunakan.

Jika Anda menggunakan pendekatan berdasarkan waktu saat merancang, selalu pertimbangkan dua hal utama. Pertama, seberapa konsisten pola penggunaannya? Kedua, apa dampak dari perubahan pola tersebut? Anda dapat meningkatkan akurasi prediksi dengan memantau beban kerja Anda dan menggunakan kecerdasan bisnis. Jika Anda mendapati perubahan yang signifikan dalam pola penggunaan, Anda dapat menyesuaikan waktu untuk memastikan bahwa cakupan tersedia.

## Langkah implementasi

- Konfigurasi penskalaan terjadwal: Untuk perubahan permintaan yang dapat diprediksi, penskalaan berdasarkan waktu dapat memberikan jumlah sumber daya yang benar pada waktu yang tepat. Hal ini juga bermanfaat jika pembuatan dan konfigurasi sumber daya tidak cukup cepat untuk merespons perubahan permintaan. Menggunakan analisis beban kerja konfigurasi penskalaan terjadwal menggunakan AWS Auto Scaling. Untuk mengonfigurasi penjadwalan berdasarkan waktu, Anda dapat menggunakan penskalaan prediktif dari penskalaan terjadwal untuk menambah jumlah instans Amazon EC2 di dalam grup Auto Scaling Anda sebelumnya berdasarkan perkiraan dan prediksi perubahan beban.

- Konfigurasi penskalaan prediktif: Penskalaan prediktif memungkinkan Anda untuk menambah jumlah instans Amazon EC2 di dalam grup Auto Scaling Anda sebelum pola harian atau mingguan dalam arus lalu lintas. Jika Anda sering mengalami lonjakan lalu lintas dan aplikasi Anda perlu waktu lama untuk memulai, Anda sebaiknya mempertimbangkan untuk menggunakan penskalaan prediktif. Penskalaan prediktif dapat membantu Anda melakukan penskalaan lebih cepat dengan memulai kapasitas sebelum beban yang diperkirakan dibandingkan dengan penskalaan dinamis saja, yang memiliki sifat reaktif. Sebagai contoh, jika pengguna mulai menggunakan beban kerja Anda pada awal jam kerja dan tidak menggunakannya setelah jam kerja, maka penskalaan prediktif dapat menambah kapasitas sebelum jam kerja, sehingga tidak terjadi keterlambatan penskalaan dinamis dalam menanggapi perubahan lalu lintas.
- Konfigurasi penskalaan otomatis dinamis: Untuk mengonfigurasi penskalaan berdasarkan metrik beban kerja yang aktif, gunakan Auto Scaling. Gunakan analisis dan konfigurasi Auto Scaling untuk melakukan peluncuran pada tingkat sumber daya yang benar, dan pastikan beban kerja menyesuaikan skala pada waktu yang diinginkan. Anda dapat meluncurkan dan secara otomatis menskalakan armada Instans Sesuai Permintaan dan Instans Spot di dalam satu grup Auto Scaling. Selain menerima diskon karena menggunakan Instans Spot, Anda dapat menggunakan Instans Terpesan atau Savings Plan untuk menerima tarif diskon untuk harga Instans Sesuai Permintaan biasa. Semua kombinasi faktor tersebut membantu Anda mengoptimalkan penghematan biaya Anda untuk instans Amazon EC2 dan membantu Anda mendapatkan skala dan kinerja yang diinginkan untuk aplikasi Anda.

## Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [Penjadwal Instans AWS](#)
- Skalakan ukuran grup Auto Scaling Anda
- [Memulai dengan Amazon EC2 Auto Scaling](#)
- [Memulai dengan Amazon SQS](#)
- [Penskalaan Terjadwal untuk Amazon EC2 Auto Scaling](#)
- [Penskalaan prediktif untuk Amazon EC2 Auto Scaling](#)

Video terkait:

- [Kebijakan Penskalaan Pelacakan Target untuk Auto Scaling](#)
- [Penjadwal Instans AWS](#)

Contoh terkait:

- [Pemilihan Tipe Instans Berbasis Atribut untuk Auto Scaling untuk Armada Amazon EC2](#)
- [Mengoptimalkan Amazon Elastic Container Service untuk biaya menggunakan penskalaan terjadwal](#)
- [Penskalaan Prediktif dengan Amazon EC2 Auto Scaling](#)
- [Bagaimana cara menggunakan Penjadwal Instans dengan AWS CloudFormation untuk menjadwalkan instans Amazon EC2?](#)



# Pengoptimalan dari waktu ke waktu

Di AWS, Anda dapat melakukan pengoptimalan dari waktu ke waktu dengan meninjau layanan baru dan mengimplementasikannya di beban kerja Anda.

Saat AWS merilis layanan dan fitur baru, praktik terbaik yang dapat diterapkan adalah meninjau keputusan arsitektur Anda agar tetap hemat biaya. Seiring dengan perubahan persyaratan, jangan ragu untuk menonaktifkan sumber daya, komponen, dan beban kerja yang sudah tidak diperlukan. Pertimbangkan praktik terbaik berikut untuk membantu Anda melakukan pengoptimalan dari waktu ke waktu.

Selagi mengoptimalkan beban kerja dari waktu ke waktu dan meningkatkan budaya [CFM](#) di organisasi Anda, lakukan evaluasi biaya usaha untuk pengoperasian di cloud, tinjau operasi cloud yang menghabiskan banyak waktu, dan otomatiskan operasi tersebut untuk mengurangi tenaga manusia dan biaya dengan menerapkan layanan AWS, produk pihak ketiga, atau alat kustom yang sesuai (seperti SDK [AWS CLI](#) atau [AWS](#)).

Topik

- [Tentukan proses peninjauan dan analisis beban kerja Anda secara rutin](#)
- [Otomatisasi operasi](#)

## Tentukan proses peninjauan dan analisis beban kerja Anda secara rutin

Praktik terbaik

- [COST10-BP01 Mengembangkan proses peninjauan beban kerja](#)
- [COST10-BP02 Meninjau dan menganalisis beban kerja ini secara teratur](#)

### COST10-BP01 Mengembangkan proses peninjauan beban kerja

Kembangkan proses yang menentukan kriteria dan proses untuk peninjauan beban kerja. Upaya peninjauan harus mencerminkan potensi manfaat. Misalnya, beban kerja inti atau beban kerja dengan nilai di atas sepuluh persen dari tagihan ditinjau setiap kuartal atau setiap enam bulan, sementara beban kerja di bawah sepuluh persen ditinjau setiap tahun.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Tinggi

## Panduan implementasi

Untuk memiliki beban kerja yang paling hemat biaya, Anda harus meninjau beban kerja secara rutin untuk mengetahui apakah ada peluang untuk menerapkan layanan, fitur, dan komponen baru. Untuk mendapatkan biaya yang secara keseluruhan lebih rendah, proses harus seimbang dengan potensi jumlah penghematan. Misalnya, beban kerja yang biayanya 50% dari seluruh pengeluaran Anda harus ditinjau secara lebih rutin, dan lebih menyeluruh, daripada beban kerja yang hanya lima persen dari seluruh pengeluaran Anda. Pertimbangkan beberapa faktor eksternal atau volatilitas. Jika beban kerja melayani lokasi geografis atau segmen pasar spesifik, dan perubahan pada area tersebut diprediksi, peninjauan yang lebih sering dapat menghasilkan penghematan biaya. Faktor lain dalam peninjauan adalah upaya untuk mengimplementasikan perubahan. Jika ada biaya besar dalam pengujian dan validasi perubahan, peninjauan harus dilakukan lebih jarang.

Pertimbangkan biaya jangka panjang dalam memelihara komponen dan sumber daya yang usang dan warisan serta ketidakmampuan untuk mengimplementasikan fitur-fitur baru di dalamnya. Biaya pengujian dan validasi saat ini mungkin lebih besar dari manfaat yang diajukan. Namun, seiring waktu, biaya untuk membuat perubahan mungkin meningkat secara signifikan karena meningkatnya kesenjangan antara beban kerja dan teknologi saat ini, sehingga biayanya menjadi makin besar. Misalnya, biaya pindah ke bahasa pemrograman baru saat ini mungkin tidak hemat biaya. Namun, dalam jangka waktu lima tahun, biaya orang memiliki keterampilan bahasa tersebut mungkin meningkat, dan akibat pertumbuhan beban kerja, Anda akan memindahkan sistem yang lebih besar ke bahasa baru tersebut sehingga memerlukan upaya yang lebih banyak daripada sebelumnya.

Uraikan beban kerja Anda ke dalam komponen, tentukan biaya komponen (perkiraan saja sudah cukup), kemudian buat daftar faktor (misalnya, upaya dan pasar eksternal) di sebelah setiap komponen. Gunakan indikator ini untuk menentukan frekuensi peninjauan untuk setiap beban kerja. Misalnya, Anda mungkin memiliki server web dengan biaya yang tinggi, upaya perubahan yang rendah, dan faktor eksternal yang tinggi, sehingga frekuensi peninjauannya tinggi. Basis data pusat mungkin memerlukan biaya sedang, upaya perubahannya tinggi, dan faktor eksternalnya rendah, sehingga frekuensi peninjauannya sedang.

Tetapkan proses untuk mengevaluasi layanan, pola desain, tipe sumber daya, dan konfigurasi baru untuk mengoptimalkan biaya beban kerja Anda saat sudah tersedia. Serupa dengan proses [peninjauan pilar kinerja](#) dan [peninjauan pilar keandalan](#), identifikasi, validasi, dan prioritaskan aktivitas optimalisasi dan perbaikan serta penyelesaian masalah dan sertakan hal ini ke dalam backlog Anda.

### Langkah implementasi

- Tentukan frekuensi peninjauan: Tentukan seberapa sering beban kerja dan komponennya harus ditinjau. Alokasikan waktu dan sumber daya untuk perbaikan kontinu dan tinjau frekuensi untuk meningkatkan efisiensi dan optimalisasi beban kerja Anda. Ini adalah kombinasi faktor yang mungkin berbeda-beda dari satu beban kerja ke beban kerja lain dalam organisasi Anda dan antara komponen-komponen dalam beban kerja tersebut. Faktor yang umum antara lain tingkat kepentingan bagi organisasi yang diukur dalam hal pendapatan dan merek, biaya total untuk menjalankan beban kerja (termasuk biaya operasi dan sumber daya), kompleksitas beban kerja, tingkat kesulitan implementasi perubahan, perjanjian lisensi perangkat lunak apa pun, dan apakah perubahan akan mendatangkan peningkatan yang signifikan pada biaya lisensi akibat lisensi yang merugikan. Komponen dapat ditentukan secara fungsi atau secara teknis, seperti server web dan basis data, atau sumber daya komputasi dan penyimpanan. Seimbangkan faktor sesuai kebutuhan dan kembangkan periode bagi beban kerja serta komponennya. Anda mungkin memutuskan untuk meninjau beban kerja penuh setiap 18 bulan, meninjau server web setiap enam bulan, basis data setiap 12 bulan, komputasi dan penyimpanan jangka pendek setiap enam bulan, serta penyimpanan jangka panjang setiap 12 bulan.
- Tentukan ketelitian peninjauan: Tentukan seberapa besar upaya yang dikeluarkan untuk meninjau beban kerja atau komponen beban kerja. Seperti halnya frekuensi peninjauan, ini adalah keseimbangan antara beberapa faktor. Evaluasi dan prioritaskan peluang untuk peningkatan guna memfokuskan upaya ke hal-hal yang memberikan manfaat paling besar sekaligus memperkirakan seberapa besar upaya yang diperlukan untuk aktivitas-aktivitas tersebut. Jika hasil yang diperkirakan tidak memenuhi tujuan, dan upaya yang diperlukan memerlukan biaya lebih besar, lakukan iterasi menggunakan tindakan alternatif. Proses peninjauan Anda harus mencakup waktu dan sumber daya yang didedikasikan untuk melakukan peningkatan yang bertambah terus menerus. Sebagai contoh, Anda mungkin memutuskan untuk menghabiskan satu pekan untuk menganalisis komponen basis data, satu minggu analisis untuk sumber daya komputasi, dan empat jam untuk peninjauan penyimpanan.

## Sumber daya

### Dokumen terkait:

- [Blog Berita AWS](#)
- [Jenis Komputasi Cloud](#)
- [Yang Baru dengan AWS](#)

### Contoh terkait:

- [Layanan Proaktif Dukungan AWS](#)
- [Peninjauan beban kerja rutin untuk beban kerja SAP](#)

## COST10-BP02 Meninjau dan menganalisis beban kerja ini secara teratur

Beban kerja yang sudah ada ditinjau secara teratur berdasarkan setiap proses yang ditetapkan untuk mengetahui apakah layanan baru dapat diadopsi, layanan yang sudah ada dapat diganti, atau beban kerja dapat dirancang ulang.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Sedang

### Panduan implementasi

AWS secara terus-menerus menambahkan fitur baru sehingga Anda dapat bereksperimen dan berinovasi dengan lebih cepat dengan teknologi terbaru. [AWS Yang Baru](#) memerinci bagaimana AWS melakukan tindakan ini dan memberikan ikhtisar ringkas tentang pengumuman ekspansi Regional, fitur, dan layanan AWS pada saat dirilis. Anda dapat mendalami peluncuran yang telah diumumkan dan menggunakannya untuk meninjau dan menganalisis beban kerja Anda yang sudah ada. Untuk mewujudkan manfaat dari fitur dan layanan AWS baru, Anda meninjau di beban kerja Anda dan mengimplementasikan fitur serta layanan baru sesuai yang diperlukan. Ini artinya Anda mungkin harus mengganti layanan yang sudah ada yang Anda gunakan untuk beban kerja, atau memodernisasikan beban kerja Anda untuk mengadopsi layanan AWS baru ini. Contohnya, Anda mungkin meninjau beban kerja Anda dan mengganti komponen pesan dengan Amazon Simple Email Service. Ini menghilangkan biaya operasi dan pemeliharaan armada instans, sekaligus memberikan semua fungsionalitas dengan harga yang lebih murah.

Untuk menganalisis beban kerja Anda dan menyorot potensi peluang, Anda juga harus mempertimbangkan cara baru untuk membangun solusi, bukan hanya tentang layanan baru. Tinjau video [Ini adalah Arsitektur Saya](#) di AWS untuk mempelajari tentang desain arsitektur pelanggan lainnya, tantangan mereka, dan solusi mereka. Lihat [seri Sepenuhnya](#) untuk mengetahui aplikasi layanan AWS dalam dunia nyata dan kisah pelanggan. Anda juga dapat menonton seri video [Kembali ke Hal Mendasar](#) yang menjelaskan, memeriksa, dan memerinci praktik terbaik pola arsitektur cloud dasar. Sumber daya lain yakni video [Cara Membangun Ini](#), yang didesain untuk membantu orang dengan ide-ide besar terkait cara mewujudkan produk layak minimum (MVP) menggunakan layanan AWS. Ini adalah cara bagi builder dari seluruh dunia yang memiliki ide kuat untuk mendapatkan panduan arsitektural dari Solutions Architects AWS. Terakhir, Anda dapat meninjau materi sumber daya [Memulai](#), yang memiliki tutorial langkah demi langkah.

Sebelum menjalankan proses peninjauan Anda, ikuti persyaratan bisnis Anda untuk persyaratan beban kerja, keamanan, dan privasi data untuk menggunakan persyaratan performa dan Wilayah atau layanan tertentu sekaligus mengikuti proses peninjauan yang telah disepakati.

### Langkah implementasi

- Tinjau beban kerja secara teratur: Menggunakan proses yang telah ditetapkan, lakukan peninjauan dengan frekuensi yang telah ditentukan. Pastikan Anda memberikan upaya yang cukup di setiap komponen. Proses ini serupa dengan proses desain awal ketika Anda memilih layanan untuk pengoptimalan biaya. Analisis layanan dan manfaatnya, kali ini perhitungkan biaya perubahan, tidak hanya manfaat jangka panjang saja.
- Implementasikan layanan baru: Jika hasil analisis mengatakan perubahan harus diimplementasikan, pertama, buat garis dasar beban kerja guna mengetahui biaya saat ini untuk setiap output. Implementasikan perubahan, kemudian lakukan analisis untuk mengonfirmasi biaya baru untuk setiap output.

### Sumber daya

#### Dokumen terkait:

- [Blog Berita AWS](#)
- [Yang Baru dengan AWS](#)
- [Dokumentasi AWS](#)
- [AWS Memulai](#)
- [Sumber Daya Umum AWS](#)

#### Video terkait:

- [AWS - Ini adalah Arsitektur Saya](#)
- [AWS - Kembali ke Hal Mendasar](#)
- [AWS - Seri Sepenuhnya](#)
- [Cara Membangun Proses Ini](#)

# Otomatisasi operasi

## Praktik terbaik

- [COST11-BP01 Melakukan otomatisasi untuk operasi](#)

## COST11-BP01 Melakukan otomatisasi untuk operasi

Evaluasi biaya dari upaya untuk operasi di cloud. Kuantifikasikan pengurangan waktu dan upaya untuk tugas administrasi, deployment, dan operasi lainnya menggunakan otomatisasi. Evaluasi waktu dan biaya yang diperlukan untuk upaya operasi dan otomatisasi tugas administrasi untuk mengurangi upaya manusia apabila mungkin.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak dijalankan: Rendah

Mengotomatiskan operasi akan meningkatkan konsistensi dan skalabilitas, memberikan visibilitas, keandalan, dan fleksibilitas lebih tinggi, mengurangi biaya, dan mempercepat inovasi dengan membebaskan sumber daya manusia dan meningkatkan metrik. Otomatisasi mengurangi frekuensi tugas manual, meningkatkan efisiensi, dan menguntungkan perusahaan dengan memberikan pengalaman yang konsisten dan andal saat melakukan deployment, memberikan, atau mengoperasikan beban kerja. Anda dapat membebaskan sumber daya infrastruktur dari tugas operasional manual dan menggunakannya untuk inovasi dan tugas nilai lebih tinggi, sehingga meningkatkan hasil bisnis. Perusahaan memerlukan cara yang terbukti dan teruji untuk mengelola beban kerja mereka di cloud. Solusi tersebut harus aman, cepat, dan hemat biaya, dengan risiko minimum dan keandalan maksimum.

Mulai dengan memprioritaskan operasi Anda berdasarkan upaya yang diperlukan dengan melihat biaya operasi secara keseluruhan di cloud. Contohnya, berapa lama waktu yang diperlukan untuk melakukan deployment sumber daya baru di cloud, membuat perubahan optimisasi pada yang sudah ada, atau mengimplementasikan konfigurasi yang diperlukan? Lihat biaya total dari tindakan manusia dengan memperhitungkan biaya operasi dan manajemen. Prioritaskan otomatisasi untuk tugas administrasi guna mengurangi upaya manusia. Upaya peninjauan harus mencerminkan potensi manfaat. Contohnya, waktu yang diluangkan untuk melakukan tugas secara manual dibandingkan secara otomatis. Prioritaskan otomatisasi aktivitas bernilai tinggi yang repetitif. Aktivitas yang menimbulkan risiko lebih tinggi dalam hal kesalahan manusia biasanya merupakan tempat yang baik untuk memulai otomatisasi karena sering kali risikonya menimbulkan biaya operasional tambahan yang tidak diinginkan (seperti jam kerja tambahan untuk tim operasi).

Dengan menggunakan layanan AWS, alat, atau produk pihak ketiga, Anda dapat memilih otomatisasi AWS mana yang akan diimplementasikan dan disesuaikan untuk persyaratan spesifik Anda. Tabel berikut menunjukkan beberapa kemampuan dan fungsi operasi inti yang dapat Anda capai dengan layanan AWS untuk mengotomatiskan administrasi dan operasi:

- [AWS Audit Manager](#): Terus audit penggunaan AWS Anda untuk menyederhanakan penilaian kepatuhan dan risiko
- [AWS Backup](#): Kelola dan otomatisasi perlindungan data secara terpusat.
- [AWS Config](#): Konfigurasi sumber daya komputasi, nilai, audit, dan evaluasi konfigurasi dan inventaris sumber daya.
- [AWS CloudFormation](#): Luncurkan sumber daya yang mudah tersedia dengan infrastruktur sebagai kode.
- [AWS CloudTrail](#): Manajemen perubahan IT, kepatuhan, dan kontrol.
- [Amazon EventBridge](#): Jadwalkan acara dan picu AWS Lambda untuk mengambil tindakan.
- [AWS Lambda](#): Otomatiskan proses repetitif dengan memicunya menggunakan acara atau dengan menjalankannya sesuai jadwal tetap dengan Amazon EventBridge.
- [AWS Systems Manager](#): Mulai dan hentikan beban kerja, patch sistem operasi, otomatisasi konfigurasi, dan lakukan manajemen yang berkelanjutan.
- [AWS Step Functions](#): Jadwalkan pekerjaan dan otomatisasi alur kerja.
- [AWS Service Catalog](#): Pemakaian templat dan infrastruktur sebagai kode dengan kepatuhan dan kontrol.

Pertimbangkan penghematan waktu yang memungkinkan tim Anda untuk berfokus pada penghentian utang teknis, inovasi, dan fitur yang menambah nilai. Misalnya, Anda mungkin perlu mengangkat dan menggeser lingkungan on-premise Anda ke cloud secepat mungkin dan kemudian mengoptimalkannya. Sebaiknya cari tahu penghematan apa saja yang dapat Anda realisasikan dengan layanan terkelola penuh menggunakan AWS yang menghilangkan atau mengurangi biaya lisensi seperti [Amazon Relational Database Service](#), [Amazon EMR](#), [Amazon WorkSpaces](#), dan [Amazon SageMaker](#). Layanan terkelola menghilangkan beban administratif dan operasional pemeliharaan layanan, sehingga Anda dapat berfokus pada inovasi. Selain itu, karena layanan terkelola beroperasi di skala cloud, biaya yang ditawarkan per transaksi atau layanan dapat lebih rendah.

Jika Anda ingin mengadopsi otomatisasi dengan segera menggunakan layanan dan produk AWS dan jika tidak ada keterampilannya dalam organisasi Anda, hubungi [AWS Managed Services](#)



(AMS), [Layanan Profesional AWS](#), atau [Partner AWS](#) untuk meningkatkan adopsi otomatisasi dan meningkatkan keunggulan operasional Anda di cloud.

[AWS Managed Services \(AMS\)](#) adalah layanan yang mengoperasikan infrastruktur AWS atas nama partner dan pelanggan perusahaan. Layanan ini menyediakan lingkungan yang aman dan patuh sebagai tempat deployment beban kerja Anda. AMS menggunakan model operasi cloud perusahaan dengan otomatisasi untuk memungkinkan Anda memenuhi persyaratan perusahaan, memindahkan ke cloud dengan lebih cepat, serta mengurangi biaya untuk manajemen berkelanjutan.

[AWS Layanan Profesional](#) juga dapat membantu Anda mendapatkan hasil bisnis yang diinginkan dan mengotomatiskan operasi dengan AWS. AWS Layanan Profesional memberikan praktik spesialisasi global untuk mendukung upaya Anda dalam fokus area komputasi cloud perusahaan. Praktik spesialisasi memberikan panduan bertarget melalui praktik terbaik, kerangka kerja, alat, dan layanan untuk semua area subjek industri, teknologi, dan solusi. Praktik ini membantu pelanggan melakukan deployment operasi IT otomatis yang andal dan tangkas, dan kemampuan tata kelola yang dioptimalkan untuk pusat cloud.

#### Langkah implementasi

- Bangun satu kali dan lakukan banyak deployment: Gunakan infrastruktur sebagai kode seperti AWS CloudFormation, AWS SDK, atau AWS Command Line Interface (AWS CLI) untuk melakukan deployment satu kali dan gunakan berulang kali untuk lingkungan yang sama atau untuk skenario pemulihan bencana. Beri tanda saat melakukan deployment untuk melacak pemakaian Anda sebagaimana ditetapkan dalam praktik terbaik lainnya. Gunakan [AWS Launch Wizard](#) untuk mengurangi waktu deployment sejumlah besar beban kerja populer perusahaan. AWS Launch Wizard memandu Anda dalam penentuan ukuran, konfigurasi, dan deployment beban kerja perusahaan mengikuti praktik terbaik AWS. Anda juga dapat menggunakan [AWS Service Catalog](#), yang membantu Anda membuat dan mengelola templat yang disetujui infrastruktur sebagai kode untuk digunakan di AWS sehingga siapa pun dapat menemukan sumber daya cloud yang disetujui dengan layanan mandiri.
- Otomatiskan operasi: Jalankan operasi rutin secara otomatis tanpa intervensi manusia. Dengan menggunakan alat dan layanan AWS, Anda dapat memilih otomatisasi AWS mana yang akan diimplementasikan dan disesuaikan untuk persyaratan spesifik Anda. Contohnya, gunakan [EC2 Image Builder](#) untuk membangun, menguji, dan melakukan deployment mesin virtual dan citra kontainer untuk digunakan di AWS atau on-premise. Jika tindakan yang Anda inginkan tidak dapat dilakukan dengan layanan AWS atau Anda memerlukan tindakan lebih kompleks dengan sumber daya penyaring, maka otomatiskan operasi Anda menggunakan [AWS CLI](#) atau alat SDK AWS. AWS CLI memberikan kemampuan untuk mengotomatiskan seluruh proses pengontrolan dan



pengelolaan layanan AWS lewat skrip tanpa menggunakan Konsol AWS. Pilih SDK AWS yang Anda sukai untuk berinteraksi dengan layanan AWS. Untuk contoh kode lainnya, lihat [repositori contoh Kode SDK AWS](#).

## Sumber daya

Dokumen terkait:

- [Memodernisasikan operasi di AWS Cloud](#)
- [Layanan AWS untuk Otomatisasi](#)
- [Otomatisasi AWS Systems Manager](#)
- [Otomatisasi AWS untuk operasi dan administrasi SAP](#)
- [AWS Managed Services](#)
- [Layanan Profesional AWS](#)
- [Infrastruktur dan otomatisasi](#)

Contoh terkait:

- [Menginovasikan kembali operasi otomatis \(Bagian I\)](#)
- [Menginovasikan kembali operasi otomatis \(Bagian II\)](#)
- [Otomatisasi AWS untuk operasi dan administrasi SAP](#)
- [Otomatisasi IT dengan AWS Lambda](#)
- [Repositori Contoh Kode AWS](#)
- [Sampel AWS](#)

## Kesimpulan

Optimasi Biaya dan Manajemen Keuangan Cloud adalah upaya yang berkelanjutan. Anda harus secara rutin bekerja sama dengan tim keuangan dan teknologi, meninjau pendekatan arsitektural, serta memperbarui pilihan komponen Anda.

AWS berupaya membantu Anda menghemat biaya saat membangun deployment yang tangguh, responsif, dan adaptif. Untuk menerapkan optimasi biaya deployment Anda, manfaatkan alat, teknik, dan praktik terbaik yang dijelaskan dalam tulisan ini.

# Kontributor

Kontributor dokumen ini termasuk:

- Ben Mergen, Cost Optimization Pillar Lead, Well-Architected, Amazon Web Services
- Keith Jarrett, Business Development Lead – Optimasi Biaya, Amazon Web Services
- Arthur Basbaum, Business Developer Manager, Amazon Web Services
- Jarman Hauser, Commercial Architect, Amazon Web Services

## Sumber Bacaan Lebih Lanjut

Untuk informasi tambahan, buka:

- [AWS Well-Architected Framework](#)
- [Pusat Arsitektur AWS](#)

## Revisi Dokumen

Berlangganan umpan RSS untuk memperoleh pemberitahuan tentang pembaruan laporan resmi ini.

| Perubahan   | Deskripsi   | Tanggal           |
|---|---|-------------------|
| <a href="#">Panduan praktik terbaik yang diperbarui</a> | Pembaruan praktik terbaik kecil di berbagai bagian.   | December 6, 2023  |
| <a href="#">Panduan praktik terbaik yang diperbarui</a> | Praktik terbaik diperbarui dengan panduan baru di seluruh pilar.  | October 3, 2023   |
| <a href="#">Panduan praktik terbaik yang diperbarui</a> | Praktik terbaik diperbarui dengan panduan baru di area-area berikut: <a href="#">Tata kelola</a> , <a href="#">Memantau biaya dan penggunaan</a> , <a href="#">Memilih model harga terbaik</a> , dan <a href="#">Mengelola sumber daya pasokan dan permintaan</a> . | July 13, 2023     |
| <a href="#">Pembaruan untuk Kerangka Kerja baru</a>     | Praktik terbaik diperbarui dengan panduan preskriptif dan praktik terbaik baru ditambahkan. Pertanyaan COST 11 ditambahkan dengan praktik terbaik baru COST11-BP01.   | April 10, 2023    |
| <a href="#">Laporan resmi diperbarui</a>                | Praktik terbaik diperbarui dengan panduan implementasi baru.  | December 15, 2022 |
| <a href="#">Laporan resmi diperbarui</a>                | Praktik terbaik diperluas dan rencana pengembangan ditambahkan.   | October 20, 2022  |

---

|   |  |                  |
|---|--|------------------|
| <a href="#">Pembaruan kecil</a>                     | Penambahan Pilar Pelestarian Lingkungan ke pengantar.  | December 2, 2021 |
| <a href="#">Pembaruan kecil</a>                     | Tautan diperbarui.   | April 25, 2021   |
| <a href="#">Pembaruan kecil</a>                     | Tautan diperbarui.   | March 10, 2021   |
| <a href="#">Pembaruan untuk Kerangka Kerja baru</a> | Diperbarui untuk menggabungkan CFM, layanan baru, dan juga integrasi dengan Well-Architected.                  | July 8, 2020     |
| <a href="#">Laporan resmi diperbarui</a>            | Diperbarui untuk mencerminkan perubahan pada AWS dan menyertakan pembelajaran dari tinjauan bersama pelanggan. | July 1, 2018     |
| <a href="#">Laporan resmi diperbarui</a>            | Diperbarui untuk mencerminkan perubahan pada AWS dan menyertakan pembelajaran dari tinjauan bersama pelanggan. | November 1, 2017 |
| <a href="#">Publikasi awal</a>                      | Pilar Optimasi Biaya - Kerangka Kerja AWS Well-Architected dipublikasikan.                                     | November 1, 2016 |