

Kerangka Kerja AWS Well-Architected

Pilar Optimalisasi Biaya



Pilar Optimalisasi Biaya: Kerangka Kerja AWS Well-Architected

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

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

Table of Contents

| | |
|---|----|
| Abstrak dan pengantar | 1 |
| Abstrak | 1 |
| Pengantar | 1 |
| Optimalisasi biaya | 3 |
| Prinsip desain | 3 |
| Definisi | 4 |
| Mempraktikkan Manajemen Keuangan Cloud | 6 |
| COST01-BP01 Menetapkan kepemilikan optimalisasi biaya | 9 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST01-BP02 Menjalin kemitraan antara 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 Menerapkan kesadaran biaya dalam proses organisasi Anda | 21 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST01-BP05 Melaporkan dan memberi tahu tentang pengoptimalan biaya | 23 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST01-BP06 Memantau biaya secara proaktif | 26 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST01-BP07 Tetap up-to-date dengan rilis layanan baru | 28 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST01-BP08 Menciptakan budaya sadar biaya | 29 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST01-BP09 Mengukur nilai bisnis dari optimalisasi biaya | 32 |
| 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 | 45 |
| COST02-BP04 Mengimplementasikan grup dan peran | 49 |
| COST02-BP05 Mengimplementasikan kontrol biaya | 51 |
| COST02-BP06 Lacak siklus hidup proyek | 54 |
| Pemantauan biaya dan penggunaan | 56 |
| COST03-BP01 Mengonfigurasi sumber informasi yang mendetail | 57 |
| COST03-BP02 Tambahkan informasi organisasi ke biaya dan penggunaan | 59 |
| COST03-BP03 Mengidentifikasi kategori atribusi biaya | 62 |
| COST03-BP04 Menetapkan metrik organisasi | 65 |
| COST03-BP05 Mengonfigurasi alat manajemen penagihan dan biaya | 66 |
| COST03-BP06 Mengalokasikan biaya berdasarkan metrik beban kerja | 70 |
| Nonaktifkan sumber daya | 72 |
| COST04-BP01 Lacak sumber daya selama masa pakainya | 72 |
| COST04-BP02 Menerapkan proses dekomisioning | 74 |
| COST04-BP03 Menonaktifkan sumber daya | 77 |
| COST04-BP04 Menonaktifkan sumber daya secara otomatis | 77 |
| COST04-BP05 Menegakkan kebijakan retensi data | 79 |
| Sumber daya hemat biaya | 81 |
| Evaluasi biaya saat Anda memilih layanan | 81 |
| COST05-BP01 Identifikasi persyaratan organisasi untuk biaya | 81 |
| COST05-BP02 Menganalisis semua komponen beban kerja | 83 |
| COST05-BP03 Melakukan analisis menyeluruh dari setiap komponen | 86 |
| COST05-BP04 Pilih perangkat lunak dengan lisensi hemat biaya | 88 |
| COST05-BP05 Pilih komponen beban kerja ini untuk mengoptimalkan biaya sesuai dengan prioritas organisasi | 90 |
| COST05-BP06 Lakukan analisis biaya untuk penggunaan yang berbeda dari waktu ke waktu | 92 |
| Pilih tipe, ukuran, dan jumlah sumber daya yang tepat | 94 |
| COST06-BP01 Lakukan pemodelan biaya | 95 |
| COST06-BP02 Pilih jenis sumber daya, ukuran, dan nomor berdasarkan data | 97 |
| COST06-BP03 Pilih tipe, ukuran, dan jumlah sumber daya secara otomatis berdasarkan metrik | 99 |

| | |
|---|-----|
| COST06-BP04 Pertimbangkan penggunaan sumber daya bersama | 102 |
| Memilih model penetapan harga terbaik | 105 |
| COST07-BP01 Lakukan analisis model harga | 112 |
| COST07-BP02 Memilih Wilayah berdasarkan biaya | 115 |
| COST07-BP03 Pilih perjanjian pihak ketiga dengan persyaratan hemat biaya | 117 |
| COST07-BP04 Menerapkan model penetapan harga untuk semua komponen beban kerja ini | 119 |
| COST07-BP05 Melakukan analisis model harga di tingkat akun manajemen | 121 |
| Merencanakan transfer data | 124 |
| COST08-BP01 Lakukan pemodelan transfer data | 124 |
| COST08-BP02 Pilih komponen untuk mengoptimalkan biaya transfer data | 126 |
| COST08-BP03 Mengimplementasikan layanan untuk mengurangi biaya transfer data | 128 |
| Kelola sumber daya pasokan dan permintaan | 131 |
| COST09-BP01 Melakukan analisis pada permintaan beban kerja | 131 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST09-BP02 Menerapkan buffer atau throttle untuk mengelola permintaan | 134 |
| Panduan implementasi | 9 |
| Sumber daya | 11 |
| COST09-BP03 Menyediakan sumber daya secara dinamis | 137 |
| Panduan implementasi | 9 |
| Langkah-langkah implementasi | 10 |
| Sumber daya | 11 |
| Pengoptimalan dari waktu ke waktu | 145 |
| Tentukan proses peninjauan dan analisis beban kerja Anda secara rutin | 145 |
| COST10-BP01 Mengembangkan proses peninjauan beban kerja | 145 |
| COST10-BP02 Tinjau dan analisis beban kerja ini secara teratur | 148 |
| Otomatisasi operasi | 150 |
| COST11-BP01 Lakukan otomatisasi untuk operasi | 150 |
| Kesimpulan | 155 |
| Kontributor | 156 |
| Sumber bacaan lebih lanjut | 157 |
| Revisi dokumen | 158 |
| Pemberitahuan | 160 |
| AWS Glosarium | 161 |

Pilar Pengoptimalan Biaya - Kerangka AWS Well-Architected

Tanggal publikasi: 27 Juni 2024 ([Revisi dokumen](#))

Abstrak

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

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

The [AWS Well-Architected](#) Framework membantu Anda memahami keputusan yang Anda buat saat membangun beban kerja. AWS Kerangka Kerja ini menyediakan praktik-praktik terbaik berkaitan dengan 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 meyakini bahwa memiliki beban kerja yang didesain dengan baik akan meningkatkan peluang keberhasilan bisnis.

Enam pilar landasan kerangka kerja:

- Keunggulan Operasional
- Keamanan
- Keandalan
- Efisiensi Kinerja
- Pengoptimalan Biaya
- Keberlanjutan

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

Paper ini ditujukan bagi mereka yang berada dalam peran teknologi dan keuangan, seperti chief technology officer (CTOs), chief financial officer (CFOs), arsitek, pengembang, pengontrol keuangan, perencana keuangan, analis bisnis, dan anggota tim operasi. Laporan ini tidak menyediakan detail implementasi atau pola arsitektural, tetapi menyertakan referensi ke sumber daya yang relevan.

Optimalisasi Biaya

Optimalisasi biaya adalah proses perbaikan dan peningkatan yang dilakukan terus-menerus selama siklus hidup sebuah beban kerja. Praktik-praktik yang diuraikan dalam dokumen ini akan membantu Anda membangun dan mengoperasikan beban kerja dengan mempertimbangkan biaya yang membantu Anda mencapai hasil bisnis sekaligus meminimalkan biaya dan memungkinkan organisasi Anda memaksimalkan laba atas investasi.

Topik

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

Prinsip desain

Pertimbangkan prinsip-prinsip desain berikut ini untuk optimalisasi biaya:

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

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

Ukur efisiensi keseluruhan: Ukur hasil akhir bisnis atas beban kerja serta biaya terkait dengan penyediaannya. Gunakan data ini untuk memahami keuntungan-keuntungan yang dapat Anda peroleh dengan melakukan peningkatan output, peningkatan fungsionalitas, dan penurunan biaya.

Berhenti mengeluarkan biaya untuk pekerjaan berat yang tidak menghasilkan: AWS melakukan pekerjaan berat pada operasional pusat data, seperti racking, stacking, serta memasok daya listrik

server. Ia juga menghilangkan beban operasional dalam mengelola sistem operasi dan aplikasi dengan memanfaatkan layanan terkelola. Dengan demikian, Anda dapat berkonsentrasi pada para pelanggan dan proyek bisnis Anda, bukan pada infrastruktur IT.

Analisis dan atribusikan pengeluaran: Cloud memudahkan Anda melakukan identifikasi biaya dan penggunaan beban kerja secara akurat, sehingga kemudian memungkinkan Anda melakukan atribusi biaya IT ke aliran pendapatan secara transparan ke pemilik beban kerja masing-masing. Hal ini akan membantu Anda dalam mengukur laba atas investasi (ROI) dan memberikan peluang kepada pemilik beban kerja untuk mengoptimalkan sumber daya mereka dan memangkas biaya.

Definisi

Terdapat lima area fokus untuk optimalisasi biaya di cloud:

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

Serupa dengan pilar-pilar lain yang ada di dalam Kerangka Kerja Well-Architected, terdapat kompromi yang perlu dipertimbangkan untuk melakukan optimalisasi biaya. Misalnya, Apakah kita memilih untuk mengoptimalkan kecepatan masuk pasar, atau untuk biaya. Di beberapa kasus, pilihan terbaiknya adalah mengoptimalkan kecepatan—memasuki pasar dengan cepat, mengirimkan fitur baru, atau memenuhi tenggat waktu—bukan berinvestasi untuk melakukan optimalisasi 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 berlebih 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 optimalisasi setelahnya.

Menginvestasikan upaya yang cukup dalam strategi optimalisasi biaya di muka memungkinkan Anda lebih mudah mewujudkan manfaat ekonomi dari cloud dengan memastikan kepatuhan yang konsisten terhadap praktik terbaik dan menghindari pengadaan berlebihan yang tidak diperlukan.

Bagian berikutnya terdapat teknik dan praktik terbaik untuk implementasi Manajemen Keuangan Cloud dan optimalisasi biaya di fase awal dan yang sedang berjalan untuk beban kerja Anda.

Mempraktikkan Manajemen Keuangan Cloud

Untuk mengelola keuangan cloud, Anda harus 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) akan memungkinkan organisasi keuangan, produk, teknologi, dan bisnis untuk mengelola, mengoptimalkan, dan merencanakan biaya seiring meningkatnya penggunaan dan skala mereka di AWS. Tujuan utama CFM adalah memungkinkan pelanggan untuk mencapai hasil bisnis mereka dengan cara paling hemat biaya dan mempercepat penciptaan nilai ekonomi dan bisnis sekaligus menemukan temuan yang menunjukkan 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 timbulnya 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 sekaligus 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 dalam AWS Cloud: lihat, hemat, rencanakan, dan jalankan. Setiap pilar ini memiliki serangkaian aktivitas dan kemampuan.



Empat pilar Manajemen Keuangan Cloud.

- **Lihat:** Bagaimana Anda saat ini mengukur, memantau, dan membuat akuntabilitas untuk pengeluaran cloud Anda? Jika Anda baru menggunakan 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, Tag kebijakan, AWS Resource Groups, AWS Cost Categories, AWS Cost Explorer, AWS Cost and Usage Report, RI dan SP

Sumber Daya: Praktik Terbaik Pemberian Tag AWS, AWS Cost Categories

- **Hemat:** Alat pengoptimalan biaya apa yang saat ini Anda gunakan untuk mengoptimalkan pengeluaran Anda? Jika Anda tidak sedang menggunakan AWS, apakah Anda mengenal optimalisasi 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 dengan 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

- **Rencanakan:** Bagaimana Anda saat ini merencanakan penggunaan dan pengeluaran cloud di 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 harus merencanakan dan menetapkan ekspektasi untuk pengeluaran pada proyek-proyek cloud. AWS akan 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: Peramalan Berbasis Penggunaan, Laporan Anggaran dan Peringatan AWS

- **Jalankan:** Apa saja proses operasional dan alat 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 akan menyediakan sejumlah alat untuk membantu Anda memulai.

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

Sumber Daya: Memulai dengan Konsol AWS Billing

Berikut ini adalah praktik-praktik terbaik Manajemen Keuangan Cloud:

Praktik terbaik

- [COST01-BP01 Menetapkan kepemilikan optimalisasi biaya](#)
- [COST01-BP02 Menjalin kemitraan antara keuangan dan teknologi](#)
- [COST01-BP03 Tetapkan prakiraan dan anggaran cloud](#)
- [COST01-BP04 Menerapkan kesadaran biaya dalam proses organisasi Anda](#)
- [COST01-BP05 Melaporkan dan memberi tahu tentang pengoptimalan biaya](#)
- [COST01-BP06 Memantau biaya secara proaktif](#)
- [COST01-BP07 Tetap up-to-date dengan rilis layanan baru](#)
- [COST01-BP08 Menciptakan budaya sadar biaya](#)
- [COST01-BP09 Mengukur nilai bisnis dari optimalisasi biaya](#)

COST01-BP01 Menetapkan kepemilikan optimalisasi biaya

Buat tim (Cloud Business Office, Cloud Center of Excellence, atau FinOps tim) yang bertanggung jawab untuk membangun dan memelihara kesadaran biaya 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 diterapkan: Tinggi

Panduan implementasi

Ini adalah pengenalan fungsi atau tim Cloud Business Office (CBO) atau Cloud Center of Excellence (CCOE) yang bertanggung jawab untuk membangun dan memelihara budaya kesadaran biaya dalam komputasi awan. Fungsi ini dapat terdiri dari individu yang sudah ada sekarang, 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 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:

- **Terpusat:** 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.
- **Hibrida:** 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](#) Bisnis, Enterprise-On-Ramp atau Perusahaan dan memerlukan bantuan untuk membangun tim atau fungsi ini, hubungi pakar Cloud Financial Management (CFM) Anda melalui tim akun Anda.

Langkah-langkah implementasi

- **Tentukan para anggota inti:** Semua bagian yang relevan dari organisasi Anda harus berkontribusi dan berminat pada manajemen biaya. Tim umum dalam organisasi biasanya meliputi: keuangan, aplikasi atau pemilik produk, manajemen, dan tim teknis (DevOps). Beberapa dari mereka terlibat secara purnawaktu (keuangan atau teknis), sementara yang lain dilibatkan secara berkala sesuai kebutuhan. Individu atau tim yang berkinerja CFM membutuhkan seperangkat keterampilan berikut:
 - **Keahlian pengembangan perangkat lunak:** jika skrip dan otomatisasi sedang dibuat.
 - **Rekayasa infrastruktur:** untuk men-deploy skrip, mengotomatisasi proses, dan memahami bagaimana layanan atau sumber daya disediakan.
 - **Ketajaman operasi:** CFM adalah tentang beroperasi di cloud secara efisien dengan mengukur, memantau, memodifikasi, merencanakan, dan menskalakan penggunaan cloud yang 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 kunci berbasis nilai atau biaya (KPIs). Ketika Anda menentukan KPIs, Anda dapat menghitung biaya yang diharapkan dalam hal efisiensi dan hasil bisnis yang diharapkan. Berbasis nilai KPIs mengikat metrik biaya dan penggunaan dengan pendorong nilai bisnis dan membantu merasionalisasi perubahan dalam pengeluaran. AWS Langkah pertama untuk mendapatkan berbasis nilai KPIs adalah bekerja sama, lintas organisasi, untuk memilih dan menyetujui seperangkat standar. KPIs

- Tetapkan jadwal pertemuan rutin: Departemen (keuangan, teknologi, dan tim bisnis) harus melakukan rapat secara teratur untuk meninjau sasaran 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. Panggilan irama reguler ini dapat membantu tim mengidentifikasi nilai KPIs yang memberi makna bagi seluruh organisasi.

Sumber daya

Dokumen terkait:

- [Blog AWS CCOE](#)
- [Membuat Kantor Bisnis Cloud](#)
- [CCOE- Pusat Keunggulan Cloud](#)

Video terkait:

- [Kisah Sukses Vanguard CCOE](#)

Contoh terkait:

- [Menggunakan Cloud Center of Excellence \(CCOE\) untuk Mengubah Seluruh Perusahaan](#)
- [Membangun CCOE untuk mengubah seluruh perusahaan](#)
- [7 Jebakan yang Harus Dihindari Saat Membangun CCOE](#)

COST01-BP02 Menjalin kemitraan antara 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 merakit, menumpuk, dan menyerahkan ready-to-use infrastruktur



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 sebuah akun, dan serangkaian izin yang tepat. Ini juga yang mengurangi risiko TI dan keuangan; yang berarti tim selalu hanya dengan beberapa klik atau API panggilan dari penghentian sumber daya cloud yang mengganggu atau tidak perlu. 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 prediktabilitas 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:

- **Prospek keuangan:** CFOs, pengontrol keuangan, perencana keuangan, analis bisnis, pengadaan, sumber, dan hutang dagang harus memahami model konsumsi cloud, opsi pembelian, dan proses faktur bulanan. 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, penetapan 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 keuangan bertindak sebagai penasihat strategis sehubungan dengan bagaimana modal digunakan untuk membeli diskon komitmen (misalnya, Instans Cadangan atau AWS Savings Plans), dan bagaimana cloud digunakan untuk menumbuhkan organisasi.
- Proses pengadaan dan utang usaha yang ada digunakan dengan cloud.
- Tim keuangan dan teknologi berkolaborasi dalam memperkirakan AWS biaya dan penggunaan masa depan untuk menyelaraskan dan membangun 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-opsi pembelian jangka lebih panjang, seperti Instans Terpesan atau Savings Plans.

- Tim teknik: Membangun kemitraan antara tim keuangan dan teknologi sangat penting untuk membangun budaya sadar biaya yang mendorong para insinyur untuk mengambil tindakan pada Cloud Financial Management (). CFM Salah satu masalah umum dari CFM praktisi operasi keuangan dan tim keuangan adalah membuat para insinyur 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 bahwa mereka selaras dengan tujuan keuangan Anda dan dapat menunjukkan keselarasan melalui model keterlibatan mereka dan laba atas investasi (ROI). 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 membutuhkan kolaborasi di seluruh keuangan, teknologi, dan tim bisnis, dan pergeseran dalam bagaimana belanja 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-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. Ini biasanya, pengontrol keuangan CFOs, perencana keuangan, analis bisnis, pengadaan, dan sumber. 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 yang akan dibahas: 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.
- Tetapkan jadwal pertemuan 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:

- [AWS Blog Berita](#)

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 diterapkan: Tinggi

Panduan implementasi

Dalam pengaturan IT on-premise konvensional, pelanggan sering menghadapi tantangan perencanaan biaya tetap yang hanya berubah sesekali, biasanya dengan pembelian perangkat keras dan layanan IT baru untuk memenuhi permintaan puncak. Sebaliknya, AWS Cloud mengadopsi pendekatan yang berbeda, di mana pelanggan membayar sumber daya yang mereka gunakan sesuai kebutuhan IT dan bisnis mereka yang sebenarnya. Di lingkungan cloud, permintaan dapat mengalami fluktuasi setiap bulan, setiap hari, atau bahkan setiap jam.

Dengan menggunakan cloud, pelanggan memperoleh efisiensi, kecepatan, dan ketangkasan, yang menghasilkan pola biaya dan penggunaan yang sangat bervariasi. Biaya dapat berkurang atau terkadang bertambah dikarenakan efisiensi beban kerja yang lebih besar atau deployment beban kerja dan fitur baru. Seiring meningkatnya skala beban kerja untuk melayani basis pelanggan yang berkembang, penggunaan dan biaya cloud juga meningkat disebabkan peningkatan aksesibilitas sumber daya. Fleksibilitas dalam layanan cloud ini meluas hingga ke biaya dan prakiraan, yang menciptakan elastisitas pada tingkat tertentu.

Sangat penting melakukan penyelarasan dengan perubahan kebutuhan bisnis dan faktor pendorong permintaan ini, dan menargetkan perencanaan yang seakurat mungkin. Proses anggaran organisasi konvensional perlu beradaptasi untuk mengakomodasi variabilitas ini.

Pertimbangkan pemodelan biaya saat Anda memprakirakan biaya untuk beban kerja baru.

Pemodelan biaya menciptakan pemahaman dasar tentang perkiraan biaya cloud, yang membantu Anda melakukan analisis total biaya kepemilikan (TCO), laba atas investasi (ROI), dan analisis keuangan lainnya, menetapkan target dan harapan dengan para pemangku kepentingan, serta mengidentifikasi peluang untuk optimalisasi biaya.

Organisasi Anda harus memahami penetapan biaya dan pengelompokan yang diterima. Tingkat detail prakiraan Anda dapat bervariasi tergantung struktur organisasi dan alur kerja internal Anda. Pilih tingkat detail yang sesuai dengan kebutuhan khusus dan pengaturan organisasi Anda. Penting untuk memahami pada tingkat apa prakiraan dilakukan:

- Akun manajemen atau tingkat AWS Organizations: Akun manajemen adalah akun yang Anda gunakan untuk membuat AWS Organizations. Organisasi akan memiliki satu akun manajemen secara default.
- Akun tertaut atau akun anggota: Sebuah akun dalam Organisasi merupakan standar Akun AWS yang berisikan sumber daya AWS Anda dan identitas-identitas yang dapat mengakses sumber daya-sumber daya tersebut.
- Lingkungan: Sebuah lingkungan adalah sekumpulan sumber daya AWS yang menjalankan sebuah versi aplikasi. Lingkungan dapat dibuat dengan beberapa akun tertaut atau anggota.
- Proyek: Sebuah proyek adalah kombinasi dari sekumpulan tujuan atau tugas yang harus diselesaikan dalam satu periode yang sudah ditentukan. Penting untuk mempertimbangkan siklus hidup proyek selama prakiraan Anda.
- Layanan AWS: Grup atau kumpulan kategori seperti layanan komputasi atau penyimpanan tempat Anda dapat mengelompokkan layanan AWS untuk perkiraan Anda.
- Pengelompokan yang menyesuaikan kebutuhan: Anda dapat membuat pengelompokan yang menyesuaikan kebutuhan organisasi Anda, seperti unit bisnis, pusat biaya, tim, tag alokasi biaya, kategori biaya, akun tertaut, atau kombinasi dari semuanya.

Identifikasi pendorong bisnis yang dapat berdampak pada biaya penggunaan Anda, dan lakukan prakiraan untuk masing-masing secara terpisah untuk menghitung perkiraan penggunaan di awal. Beberapa pendorong dapat terkait dengan tim IT dan produk di dalam organisasi. Pendorong bisnis lainnya, seperti acara pemasaran, promosi, ekspansi geografis, merger, dan akuisisi, dikenal oleh pemimpin penjualan, pemasaran, dan bisnis Anda, dan penting juga untuk berkolaborasi dan memperhitungkan semua pendorong permintaan tersebut.

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.

Setelah Anda menetapkan proses prakiraan dan model pembuatan, Anda dapat menggunakan [AWS Budgets](#) untuk mengatur anggaran yang bisa disesuaikan di tingkat paling sederhana dengan menentukan periode waktu, pengulangan, atau jumlah (tetap atau berubah-ubah) dan menambahkan filter seperti layanan, Wilayah AWS dan tag. Anggaran biasanya disiapkan untuk satu tahun dan tidak berubah-ubah, yang membutuhkan kepatuhan yang ketat dari semua orang yang terlibat. Sebaliknya, prakiraan lebih fleksibel, yang memungkinkan penyesuaian ulang sepanjang tahun dan memberikan proyeksi dinamis selama periode satu, dua, atau tiga tahun. Baik anggaran maupun prakiraan memainkan peran penting ketika Anda membangun ekspektasi keuangan di kalangan berbagai pemangku kepentingan teknologi dan bisnis. Prakiraan 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.

Agar terus dapat memantau kinerja anggaran yang ada, Anda dapat membuat dan menjadwalkan laporan AWS Budgets untuk dikirim melalui email kepada Anda dan pemangku kepentingan Anda secara rutin. 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 benar-benar melampaui level tertentu atau jika diprakirakan akan melampaui jumlah yang dianggarkan.

Sesuaikan proses prakiraan dan anggaran yang ada agar menjadi lebih dinamis menggunakan algoritma berbasis tren (dengan biaya historis sebagai masukan), dan menggunakan algoritma berbasis pendorong bisnis (misalnya, peluncuran produk baru atau ekspansi Regional), yang ideal untuk lingkungan penganggaran dinamis dan bervariasi. Setelah Anda menentukan prakiraan Anda berdasarkan tren menggunakan Cost Explorer atau alat lainnya, gunakan [AWS Kalkulator Harga](#) untuk memperkirakan kasus penggunaan AWS Anda dan biaya di masa depan berdasarkan penggunaan yang diharapkan (lalu lintas, permintaan per detik, atau instans Amazon EC2 yang diperlukan).

Lacak keakuratan prakiraan tersebut, karena anggaran harus ditetapkan berdasarkan perhitungan dan estimasi prakiraan ini. Pantau keakuratan dan efektivitas prakiraan biaya cloud terintegrasi.

Tinjau secara rutin pengeluaran aktual dibandingkan dengan prakiraan Anda, dan sesuaikan dengan kebutuhan untuk meningkatkan presisi prakiraan. Lacak selisih prakiraan, dan lakukan analisis akar masalah pada selisih yang dilaporkan untuk bertindak dan menyesuaikan prakiraan.

Sebagaimana disebutkan dalam [COST01-BP02 Menjalin kemitraan antara keuangan dan teknologi](#), penting untuk membina 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, tingkatkan frekuensi rapat singkat koordinasi untuk bereaksi terhadap perubahan tersebut secara lebih cepat.

Langkah-langkah implementasi

- Tentukan bahasa biaya dalam organisasi: Buat bahasa biaya AWS yang umum dalam Organisasi dengan berbagai dimensi dan pengelompokan. Pastikan pemangku kepentingan memahami tingkat detail prakiraan, model harga, dan tingkat prakiraan biaya Anda.
- Analisis prakiraan berbasis tren: Gunakan alat prakiraan berbasis tren seperti AWS Cost Explorer dan Amazon Forecast. Analisis biaya penggunaan Anda pada beberapa dimensi-dimensi seperti kategori layanan, akun, tag, dan biaya.
- Analisis prakiraan berbasis faktor pendorong Tentukan dampak faktor-faktor bisnis pada penggunaan cloud Anda, dan prediksi setiap faktor bisnis satu-persatu untuk menghitung biaya yang dapat terakumulasi di kemudian hari. 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 prakiraan yang sudah ada dan proses penganggaran: Berdasarkan metode prakiraan yang telah digunakan seperti prakiraan berbasis tren, berbasis pendorong bisnis, atau kombinasi dari kedua metode prakiraan tersebut, tentukan prakiraan Anda dan proses penganggarnya. Anggaran harus terhitung, realistis, dan didasarkan pada proses prakiraan Anda.
- Konfigurasi peringatan dan notifikasi: Gunakan peringatan AWS Budgets dan deteksi anomali biaya untuk mendapatkan peringatan dan pemberitahuan.
- Lakukan peninjauan rutin bersama para pembangun jabatan inti: Misalnya, menyamakan persepsi tentang perubahan arah bisnis dan penggunaannya bersama para pihak penting di bagian IT, keuangan, platform, dan bidang bisnis lainnya.

Sumber daya

Dokumen terkait:

- [AWS Cost Explorer](#)
- [AWS Cost and Usage Report](#)
- [Melakukan Prakiraan dengan Cost Explorer](#)
- [Perkiraan QuickSight](#)
- [AWS Budgets](#)

Video terkait:

- [Cara Menggunakan AWS Budgets untuk melacak pengeluaran dan penggunaan saya](#)
- [Serial Pengoptimalan 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 Menerapkan 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 dalam tim bisnis dan keuangan untuk meningkatkan kesadaran biaya, dan menetapkan indikator kinerja utama efisiensi (KPIs) 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. Ketika memikirkan biaya implementasi, bingkai percakapan untuk memasukkan komponen pengembalian investasi (ROI) untuk membenarkan 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 AWS asli gratis seperti [AWS Cost Anomaly Detection](#), [AWS Budgets](#), dan [AWS Budgets Laporan](#).

Ketika organisasi secara konsisten mengadopsi praktik [Cloud Financial Management](#) (CFM), perilaku tersebut menjadi tertanam dalam cara kerja dan pengambilan keputusan. Hasilnya adalah budaya yang lebih sadar biaya, mulai dari pengembang yang merancang born-in-the-cloud aplikasi baru, hingga manajer keuangan yang menganalisis investasi cloud ROI baru ini.

Langkah-langkah implementasi

- Identifikasi proses-proses organisasi yang relevan: Setiap unit organisasi meninjau proses-proses mereka dan mengidentifikasi proses yang berdampak pada biaya dan penggunaan. Proses apa pun yang mengakibatkan pembuatan atau penghentian sumber daya perlu turut ditinjau. Cari proses yang dapat mendukung kesadaran biaya di dalam bisnis Anda, seperti manajemen insiden dan pelatihan insiden.
- Membangun budaya sadar biaya mandiri: Pastikan semua pemangku kepentingan yang relevan selaras dengan cause-of-change dan berdampak 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 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 CFM pakar melalui tim Akun Anda, atau jelajahi sumber daya dan dokumen terkait di bawah ini.

Sumber daya

Dokumen terkait:

- [AWS Manajemen Keuangan Cloud](#)

Contoh terkait:

- [Strategi untuk Manajemen Penghematan Biaya Cloud](#)
- [Seri Blog Pengendalian Biaya #3: Cara Mengatasi Lonjakan Biaya](#)
- [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 diterapkan: 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.

Periksa biaya dan penggunaan Anda dengan berbagai filter dan informasi terperinci dengan menggunakan [AWS Cost Explorer](#), yang menyediakan dasbor dan laporan seperti biaya berdasarkan layanan atau akun, biaya harian, atau biaya pasaran. Lacak kemajuan biaya dan penggunaan Anda dibandingkan dengan anggaran yang telah ditetapkan menggunakan [Laporan AWS Budgets](#).

Gunakan [AWS Budgets](#) untuk mengatur anggaran khusus untuk melacak biaya dan penggunaan Anda dan merespons dengan cepat peringatan yang diterima dari email atau notifikasi Amazon Simple Notification Service (Amazon SNS) jika Anda melebihi ambang batas Anda. [Tetapkan periode anggaran pilihan Anda](#) ke harian, bulanan, triwulanan, atau tahunan, dan buat batasan anggaran khusus untuk tetap mendapat informasi tentang bagaimana biaya aktual atau prakiraan dan kemajuan penggunaan terhadap ambang batasan anggaran Anda. Anda juga dapat mengatur [peringatan](#) dan [tindakan](#) atas pemberitahuan tersebut agar berjalan secara otomatis atau melalui proses persetujuan saat target anggaran terlampaui.

Terapkan pemberitahuan tentang biaya dan penggunaan untuk memastikan bahwa perubahan pada biaya dan penggunaan dapat ditindaklanjuti dengan cepat jika biaya dan penggunaan tak terkirakan. [AWS Cost Anomaly Detection](#) memungkinkan Anda mengurangi lonjakan biaya dan meningkatkan kendali tanpa memperlambat inovasi. AWS Cost Anomaly Detection mengidentifikasi pengeluaran yang tidak wajar dan akar penyebabnya, yang akan membantu mengurangi risiko lonjakan tagihan. Dengan tiga langkah mudah, Anda dapat membuat pemantauan kontekstual Anda sendiri dan menerima peringatan saat ada anomali pengeluaran yang terdeteksi.

Anda juga dapat menggunakan [QuickSight](#) dengan data (CUR) AWS Cost and Usage Report, untuk menghasilkan pelaporan yang sangat tersesuaikan dengan data yang lebih terperinci. QuickSight memungkinkan Anda untuk membuat jadwal laporan dan menerima email Laporan Biaya rutin berisikan historis biaya dan penggunaan atau peluang penghematan biaya. Periksa solusi [Dasbor Kecerdasan Biaya](#) (CID) kami yang dibangun di atas QuickSight, yang memberikan Anda visibilitas tingkat lanjut.

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 mitra Anda untuk membantu Anda [mengoptimalkan dan 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-langkah implementasi

- Konfigurasi 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: Penggunaan Biaya dan Tata Kelola](#)
- Laporan mengenai Optimalisasi 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](#)
- [Amazon Analitik S3](#)

Contoh terkait:

- [Cara-cara utama untuk mulai mengoptimalkan biaya cloud AWS Anda](#)

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 yang dilakukan dalam interval waktu berdekatan untuk menggunakan [AWS Cost Explorer](#) atau dasbor lainnya 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 Kecerdasan Biaya](#), sebuah solusi [Amazon QuickSight](#) yang dibuat oleh Solutions Architects AWS, dan bandingkan anggaran Anda dengan biaya dan penggunaan aktual.

Langkah-langkah implementasi

- Laporan mengenai Optimalisasi 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. 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 perincian harian [AWS Budgets](#) untuk memantau biaya dan penggunaan agar Anda dapat mengambil tindakan tepat waktu sehingga mencegah kemungkinan terjadinya kelebihan biaya yang harus dibayar: AWS Budgets memungkinkan Anda mengonfigurasi pemberitahuan peringatan, sehingga Anda tetap mendapat informasi jika ada jenis anggaran Anda yang keluar dari ambang batas yang telah ditetapkan 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 memantau biaya: [AWS Cost Anomaly Detection](#) menggunakan teknologi Machine Learning yang canggih untuk mengidentifikasi pengeluaran yang tidak wajar dan akar masalahnya, sehingga Anda dapat mengambil tindakan secara cepat. Hal ini akan 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 (CUR) AWS Cost and Usage Report Anda dengan dasbor Amazon QuickSight untuk memvisualisasikan biaya organisasi Anda: AWS Cost Explorer memiliki antarmuka yang mudah digunakan yang memungkinkan Anda memvisualisasikan, memahami, dan mengelola biaya dan penggunaan AWS Anda dari waktu ke waktu. [Dasbor Kecerdasan Biaya](#) adalah dasbor yang dapat disesuaikan dan dapat diakses untuk membantu Anda membuat landasan manajemen biaya dan alat optimalisasi Anda sendiri.

Sumber daya

Dokumen terkait:

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

Contoh terkait:

- [Peringatan AWS Cost Anomaly Detection dengan Slack](#)

COST01-BP07 Tetap up-to-date dengan rilis layanan baru

Konsultasikan secara teratur dengan para ahli atau AWS Mitra untuk mempertimbangkan layanan dan fitur mana yang memberikan biaya lebih rendah. Tinjau AWS blog dan sumber informasi lainnya.

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

Panduan implementasi

AWS terus menambahkan kemampuan baru sehingga Anda dapat memanfaatkan teknologi terbaru untuk bereksperimen dan berinovasi lebih cepat. Anda mungkin dapat menerapkan AWS layanan dan fitur baru untuk meningkatkan efisiensi biaya dalam beban kerja Anda. Lakukan peninjauan secara teratur [Manajemen Biaya AWS](#), [Blog Berita AWS](#), [Blog Manajemen Biaya AWS](#), dan [Apa yang Baru di AWS](#) untuk informasi tentang rilis layanan dan fitur-fitur baru. What's New post memberikan gambaran singkat tentang semua AWS layanan, fitur, dan pengumuman perluasan Wilayah saat dirilis.

Langkah-langkah implementasi

- Berlangganan blog: Buka halaman AWS blog dan berlangganan Blog Apa yang Baru dan blog lain yang relevan. Anda dapat mendaftar di halaman [preferensi komunikasi](#) dengan alamat email Anda.
- Berlangganan AWS Berita: Tinjau [Blog AWS Berita](#) dan [Apa yang Baru](#) secara teratur AWS untuk informasi tentang layanan baru dan rilis fitur. Berlangganan RSS feed, atau dengan email Anda untuk mengikuti pengumuman dan rilis.

- Ikuti Pengurangan AWS Harga: Pemotongan harga reguler pada semua layanan kami telah menjadi cara standar AWS untuk meneruskan efisiensi ekonomi kepada pelanggan kami yang diperoleh dari skala kami. Per 20 September 2023, AWS telah menurunkan harga 134 kali sejak 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 EC2 (Amazon), dalam [kategori pengurangan harga](#) di Blog Berita. AWS
- AWS acara dan pertemuan: Hadiri AWS pertemuan puncak lokal Anda, dan setiap pertemuan lokal dengan organisasi lain dari daerah Anda. Jika Anda tidak dapat hadir secara langsung, cobalah untuk menghadiri acara virtual untuk mendengar lebih banyak dari AWS para ahli dan kasus bisnis pelanggan lain.
- Temui tim akun Anda: Jadwalkan pertemuan rutin dengan tim akun Anda, adakan pertemuan lalu diskusikan tren industri dan layanan AWS . Bicarakan dengan manajer akun, Arsitek Solusi, dan tim dukungan Anda.

Sumber daya

Dokumen terkait:

- [AWS Manajemen Biaya](#)
- [Apa yang baru dengan AWS](#)
- [AWS Blog Berita](#)

Contoh terkait:

- [Amazon EC2 — 15 Tahun Mengoptimalkan dan Menghemat Biaya TI Anda](#)
- [AWS Blog Berita - 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 akan 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 akan memungkinkan Anda untuk 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 optimasi 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-langkah implementasi

- Laporkan biaya cloud ke tim departemen teknologi: Untuk meningkatkan kesadaran biaya, dan menetapkan KPI efisiensi bagi para pemangku kepentingan keuangan dan bisnis.
- Beritahukan para pemangku jabatan atau para anggota tim tentang perubahan yang sudah terencana: Buat item agenda untuk membahas perubahan yang direncanakan dan dampak manfaat biaya pada beban kerja selama rapat perubahan mingguan.
- Temui tim akun Anda: Jadwalkan pertemuan rutin dengan tim akun Anda, lalu 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 atau anggota teknis dilatih untuk memahami biaya sumber daya pada AWS Cloud.
- Pertemuan dan acara AWS: Hadiri Summit AWS dan pertemuan-pertemuan lokal lainnya di tempat Anda dengan organisasi lainnya dari area Anda.
- Berlangganan ke blog: Buka halaman blog AWS dan berlangganan ke [Blog Apa yang Baru](#) dan blog lainnya yang relevan untuk mengikuti rilis, implementasi, contoh, dan perubahan baru yang dibagikan oleh AWS.

Sumber daya

Dokumen terkait:

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

Contoh terkait:

- [Manajemen Finansial Cloud AWS](#)

COST01-BP09 Mengukur nilai bisnis dari optimalisasi biaya

Penghitungan nilai bisnis dari optimalisasi biaya memungkinkan Anda memahami seluruh rangkaian 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 diterapkan: 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 mengoptimalkan pengeluaran. Misalnya, jika sebuah perusahaan menghabiskan \$100.000 untuk menyebarkan beban kerja AWS dan kemudian mengoptimalkannya, biaya baru menjadi hanya \$80.000 tanpa mengorbankan kualitas atau output. Dalam skenario ini, nilai bisnis yang dihitung 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. Misalnya, Anda dapat mengukur penghematan biaya EC2 (Amazon Elastic Compute Cloud Amazon) saat membeli Savings Plans, yang mengurangi biaya dan mempertahankan tingkat output beban kerja. Anda dapat mengukur pengurangan biaya dalam AWS pengeluaran saat EC2 instans Amazon yang tidak aktif dihapus, atau volume Amazon Elastic Block Store (Amazon) yang tidak terpasang dihapus. EBS

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

Langkah-langkah implementasi

- Mengevaluasi manfaat bisnis: Ini adalah proses menganalisis dan menyesuaikan AWS Cloud biaya dengan cara yang memaksimalkan manfaat yang diterima dari setiap dolar yang dihabiskan. 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 AWS biaya peramalan: Peramalan membantu membiayai pemangku kepentingan menetapkan harapan dengan pemangku kepentingan organisasi internal dan eksternal lainnya, dan dapat meningkatkan prediktabilitas keuangan organisasi Anda. [AWS Cost Explorer](#) dapat digunakan untuk melakukan peramalan untuk biaya dan penggunaan Anda.

Sumber daya

Dokumen terkait:

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

Video terkait:

- [Buka Kunci Nilai Bisnis dengan Windows aktif AWS](#)

Contoh terkait:

- [Mengukur dan Memaksimalkan Nilai Bisnis Customer 360](#)
- [Nilai Bisnis Dengan Mengadopsi Database yang Dikelola Layanan Web Amazon](#)
- [Nilai Bisnis dari Layanan Web Amazon untuk Vendor Software Mandiri](#)
- [Nilai Bisnis dari Modernisasi Cloud](#)
- [Nilai Bisnis dari Migrasi ke Layanan Web Amazon](#)

Kesadaran akan penggunaan dan pengeluaran

Memahami pendorong dan biaya organisasi Anda adalah yang sangat penting untuk mengelola biaya dan penggunaan Anda secara efektif, serta untuk mengidentifikasi setiap peluang untuk melakukan 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 Anda untuk mengaitkan biaya sumber daya dengan beban kerja, masing-masing organisasi, atau pemilik produk akan mendorong perilaku penggunaan yang efisien dan dapat membantu mengurangi pemborosan. Pemantauan penggunaan dan biaya yang akurat akan memungkinkan Anda untuk memahami seberapa menguntungkan unit organisasi dan produk, serta memungkinkan Anda untuk mengambil keputusan-keputusan yang lebih bijaksana mengenai di mana Anda harus mengalokasikan sumber daya di dalam organisasi Anda. Kesadaran Anda akan penggunaan di semua tingkatan dalam organisasi merupakan kunci yang akan mendorong perubahan, karena perubahan dalam penggunaan akan mendorong perubahan dalam biaya.

Pertimbangkan untuk mengambil pendekatan dengan beberapa aspek untuk menyadari penggunaan dan pengeluaran Anda. Tim Anda harus mengumpulkan data, melakukan analisis, dan membuat laporan. Faktor-faktor utama yang harus Anda pertimbangkan antara lain:

Topik

- [Tata kelola](#)
- [Pemantauan biaya dan penggunaan](#)
- [Nonaktifkan sumber daya](#)

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 Lacak 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 diterapkan: 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 Anda akan penggunaan di semua tingkatan dalam organisasi merupakan kunci yang akan mendorong perubahan, karena perubahan dalam penggunaan akan mendorong perubahan dalam 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 dengan mudah dipahami 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.
- Lingkup: 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. Jadwalkan 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 kelompok dan ukuran instans umum seperti m5.large, kemudian ubah ukuran instans berdasarkan penggunaan CPU dan memori dengan 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 (savings plans) atau instans terpesan untuk beban kerja yang dapat diprediksi dan beritahukan kepada 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-langkah implementasi

- Bertemulah dengan 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.
- Dapatkan 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 onboarding: 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 dan sumber daya: Tentukan layanan yang dibutuhkan 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 sesuai fungsi: Tentukan pengguna yang berinteraksi dengan beban kerja, dengan berfokus pada apa yang mereka kerjakan dan cara mereka menggunakan beban kerja, bukan pada 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 pada sepanjang masa pakainya (pengembangan, operasi, dan dekomisioning). Identifikasi tindakannya

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

- Tentukan periode peninjauan: Beban kerja dan persyaratan organisasi dapat berubah-ubah seiring waktu. Tentukan jadwal peninjauan beban kerja untuk memastikannya tetap selaras dengan prioritas organisasi.
- Buat dokumentasi 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 Fungsi Pekerjaan](#)
- [Strategi penagihan beberapa akun AWS](#)
- [Tindakan, Sumber Daya, dan Kunci Syarat untuk Layanan AWS.](#)
- [Manajemen dan Tata Kelola AWS](#)
- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [Wilayah Infrastruktur Global dan Zona Ketersediaan \(AZ\)](#)

Video terkait:

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

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 diterapkan: 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. Tujuan-tujuan atau [indikator-indikator performa utama \(KPI\)](#) ini dapat termasuk hal-hal seperti persentase pengeluaran berdasarkan permintaan atau adopsi layanan teroptimalkan lainnya seperti instans AWS Graviton atau jenis volume gp3 EBS. Tetapkan sasaran yang jelas dan dapat dicapai untuk membantu Anda mengukur peningkatan efisiensi yang penting bagi operasional bisnis Anda. 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 (gunakan 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. Gunakan metrik yang tepat, dan tetapkan KPI yang terhitung untuk organisasi Anda. Anda dapat memulai dengan KPI dasar lalu mengembangkannya berdasarkan kebutuhan bisnis.

Sasaran untuk optimalisasi biaya adalah meningkatkan efisiensi beban kerja, yang sejalan dengan penurunan biaya per hasil bisnis untuk beban kerja tersebut dari waktu ke waktu. Implementasikan sasaran ini ke semua beban kerja, dan tetapkan target seperti peningkatan efisiensi sebesar lima persen setiap enam bulan hingga satu tahun. Di cloud, Anda dapat mencapainya melalui pembentukan kemampuan dalam optimalisasi biaya, serta rilis layanan dan fitur baru.

Target adalah tolok ukur yang jelas yang ingin Anda capai untuk memenuhi sasaran dan tolok ukur membandingkan hasil aktual Anda dengan target. Tetapkan tolok ukur dengan KPI untuk biaya per unit layanan komputasi (seperti adopsi Spot, adopsi Graviton, jenis instans terbaru, dan cakupan Sesuai Permintaan), layanan penyimpanan (seperti adopsi EBS GP3, snapshot EBS usang, dan penyimpanan Amazon S3 standard), atau penggunaan layanan basis data (seperti mesin sumber terbuka RDS, adopsi Graviton, dan cakupan Sesuai Permintaan). Tolok ukur dan KPI ini dapat membantu Anda memverifikasi bahwa Anda menggunakan layanan AWS dengan cara yang paling hemat biaya.

Tabel berikut menyediakan daftar metrik AWS standar untuk referensi. Setiap organisasi dapat memiliki nilai target yang berbeda-beda untuk KPI ini.

| Kategori | KPI (%) | Deskripsi |
|------------|--------------------------------|--|
| Komputasi | Cakupan Penggunaan EC2 | Instans EC2 (dalam biaya atau jam) menggunakan SP +RI+Spot yang dibandingkan dengan total (dalam biaya atau jam) instans EC2 |
| Komputasi | Komputasikan pemanfaatan SP/RI | Jam pemanfaatan SP atau RI akan dibandingkan dengan total jam SP atau RI yang tersedia |
| Komputasi | Biaya EC2/Jam | Biaya EC2 dibagi dengan jumlah instans EC2 yang berjalan di jam tersebut |
| Komputasi | Biaya vCPU | Biaya per vCPU untuk semua instans |
| Komputasi | Generasi Instans Terbaru | Persentase instans pada Graviton (atau jenis instans generasi modern lainnya) |
| Basis Data | Cakupan RDS | Instans RDS (dalam biaya atau jam) menggunakan RI dibandingkan dengan total instans RDS (dalam biaya atau jam) |
| Basis Data | Pemanfaatan RDS | Jam pemanfaatan RI akan dibandingkan dengan total jam RI yang tersedia |

| Kategori | KPI (%) | Deskripsi |
|-------------|---------------------------|---|
| Basis Data | Waktu aktif RDS | Biaya RDS dibagi dengan jumlah instans RDS yang berjalan pada jam tersebut |
| Basis Data | Generasi Instans Terbaru | Persentase instans pada Graviton (atau tipe instans modern lainnya) |
| Penyimpanan | Pemanfaatan penyimpanan | Biaya penyimpanan yang dioptimalkan (misalnya Glacier, deep archive, atau Infrequent Access) dibagi dengan total biaya penyimpanan |
| Penandaan | Pelepasan tag sumber daya | <p>Cost Explorer</p> <ol style="list-style-type: none"> Memfilter kredit, diskon, pajak, pengembalian dana, pasar, dan salinan biaya bulanan terbaru Pilih Tampilkan hanya sumber daya yang tidak diberi tag di Cost Explorer Bagi jumlah dalam sumber daya yang tidak mendapat tag dengan biaya bulanan Anda. |

Menggunakan tabel ini, sertakan nilai target atau tolok ukur, yang harus dihitung berdasarkan sasaran organisasi Anda. Anda perlu mengukur metrik tertentu untuk bisnis Anda dan memahami hasil bisnis untuk beban kerja tersebut untuk menentukan KPI yang akurat dan realistis. Saat Anda mengevaluasi metrik performa di dalam organisasi, bedakan di antara berbagai jenis metrik yang memiliki fungsi berbeda-beda. Metrik-metrik tersebut terutama mengukur performa dan efisiensi infrastruktur teknis alih-alih mengukur secara langsung keseluruhan dampak bisnis. Misalnya,

metrik mungkin melacak waktu respons server, latensi jaringan, atau waktu aktif sistem. Metrik ini sangat penting untuk menilai performa infrastruktur dalam mendukung operasional teknis organisasi. Namun, metrik tersebut tidak memberikan wawasan langsung tentang sasaran bisnis yang lebih luas seperti kepuasan pelanggan, pertumbuhan pendapatan, atau pangsa pasar. Untuk mendapatkan pemahaman yang komprehensif tentang performa bisnis, lengkapi metrik efisiensi ini dengan metrik bisnis strategis yang berkorelasi langsung dengan hasil bisnis.

Miliki visibilitas mendekati waktu nyata terhadap KPI Anda dan peluang penghematan terkait serta lacak kemajuan Anda dari waktu ke waktu. Untuk mulai menetapkan dan melacak tujuan-tujuan KPI, kami merekomendasikan dasbor KPI dari [Dasbor Kecerdasan Cloud \(CID\)](#). Berdasarkan data dari Laporan Biaya dan Penggunaan (CUR), dasbor KPI ini 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 untuk menetapkan dan melacak sasaran KPI, pastikan metode ini diadopsi oleh semua pemangku kepentingan manajemen keuangan cloud di organisasi Anda.

Langkah-langkah implementasi

- Tentukan tingkat penggunaan yang diharapkan: Untuk memulai, 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 mungkin berubah seiring waktu, dan apa yang dapat berubah dikarenakan peningkatan musiman atau kampanye pemasaran?
- Tentukan sumber daya dan biaya untuk menjalankan 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 biaya yang akan dikeluarkan pada setiap poin utama ini, dan perkirakan perubahan biaya ketika terdapat perubahan pada penggunaan.
- Tentukan tujuan 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 dan 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 dapat menyebabkan kerugian pada 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.

- Tentukan sasaran: Untuk setiap sasaran yang telah ditetapkan, tentukan target yang terukur. Jika sasarannya adalah untuk meningkatkan efisiensi beban kerja, targetnya seharusnya adalah menghitung jumlah peningkatan (biasanya pada output bisnis untuk setiap dolar yang dikeluarkan) dan kapan peningkatan tersebut seharusnya tercapai. Misalnya, Anda dapat menetapkan sasaran untuk meminimalkan pemborosan karena kelebihan pasokan. Dengan sasaran ini, target Anda bisa jadi adalah pemborosan akibat kelebihan pasokan komputasi di tingkat pertama beban kerja produksi tidak boleh melebihi sepuluh persen dari biaya komputasi tingkat. Selain itu, target kedua bisa jadi adalah pemborosan akibat kelebihan pasokan komputasi di tingkat kedua beban kerja produksi tidak boleh melebihi lima persen dari biaya komputasi tingkat.

Sumber daya

Dokumen terkait:

- [Kebijakan terkelola AWS untuk fungsi pekerjaan](#)
- [Strategi penagihan beberapa akun AWS](#)
- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [Tujuan S.M.A.R.T.](#)
- [Cara melacak KPI pengoptimalan biaya Anda dengan Dasbor KPI CID](#)

Video terkait:

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

Contoh terkait:

- [Apa itu metrik satuan?](#)
- [Memilih metrik satuan untuk mendukung bisnis Anda](#)
- [Metrik unit dalam praktik – pelajaran yang dipetik](#)
- [Bagaimana metrik satuan membantu menciptakan keselarasan antarfungsi bisnis](#)

COST02-BP03 Mengimplementasikan struktur akun

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

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 Akun AWS baru atau memilih akun yang sudah ada sebagai akun anggota berdasarkan struktur akun yang dirancang dengan mengikuti [praktik terbaik akun manajemen](#) dan [praktik terbaik akun anggota](#).

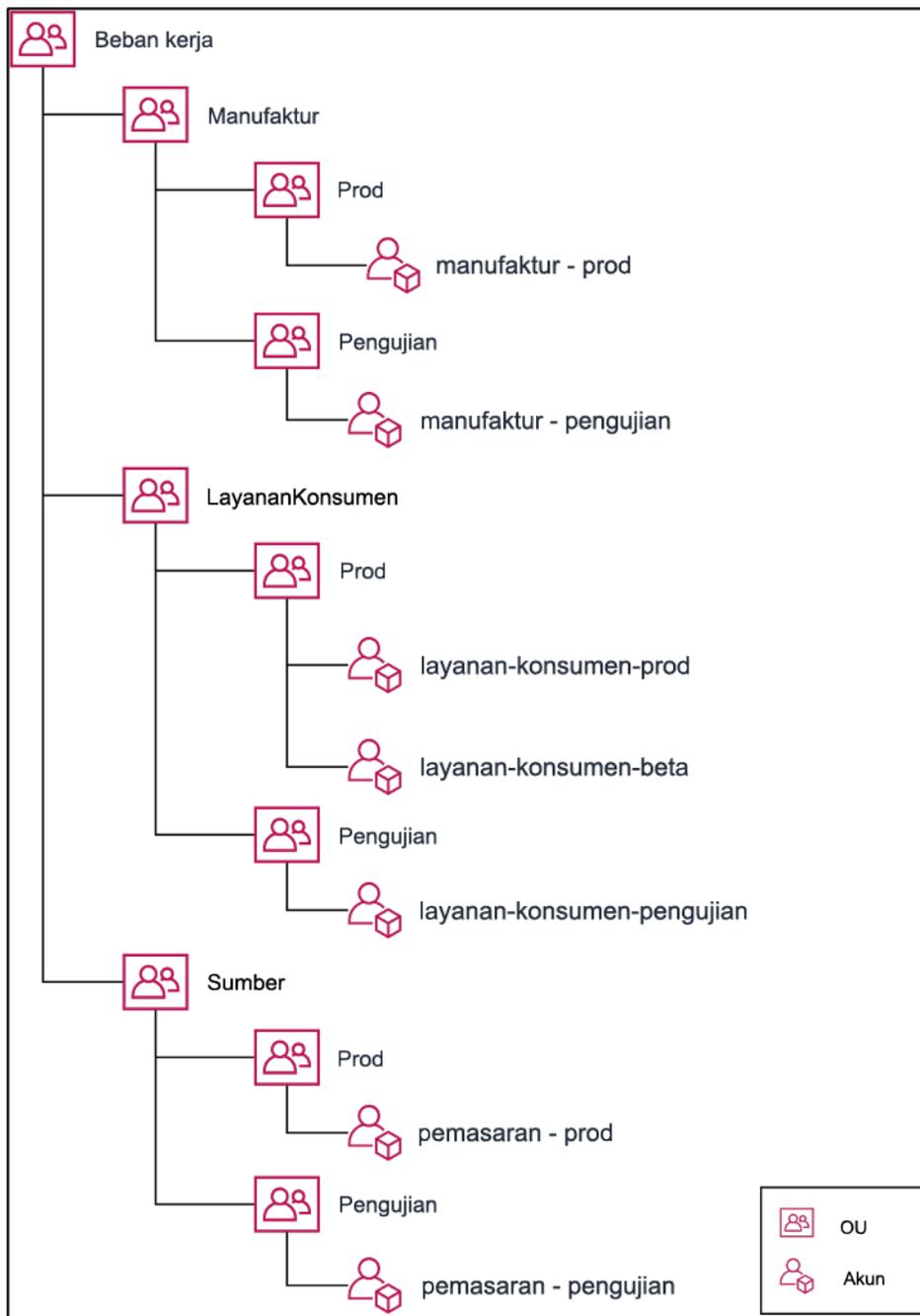
Disarankan untuk memiliki setidaknya satu akun manajemen dengan satu akun anggota yang terhubung ke sana, terlepas dari ukuran organisasi Anda dan penggunaannya. Semua sumber daya beban kerja hanya boleh berada di akun anggota dan tidak boleh ada sumber daya yang dibuat di dalam akun manajemen. Jumlah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

Dalam [AWS Organizations](#), [penagihan terkonsolidasi](#) menciptakan hubungan konsep 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 akan membantu untuk 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, jadi 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 organisasi.

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

Langkah-langkah implementasi

- Tentukan persyaratan pemisahan: Persyaratan untuk pemisahan adalah kombinasi dari beberapa faktor, termasuk konsep keamanan, konsep keandalan, dan konsep keuangan. Telusuri 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 menjadi satu berbagai lingkungan atau beban kerja serupa yang tidak perlu dipisahkan. 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 sebuah akun untuk setiap grup dan pastikan persyaratan pemisahan terpenuhi. Akun-akun ini adalah akun anggota atau akun tertaut Anda. 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 pekerjaan](#)
- [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 dengan Menggunakan Beberapa Akun](#)
- [Mengaktifkan diskon instans terpesan dan Savings Plans bersama](#)
- [Penagihan terkonsolidasi](#)
- [Penagihan terkonsolidasi](#)

Contoh terkait:

- [Memisahkan CUR dan Berbagi Akses](#)

Video terkait:

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

Contoh terkait:

- [Menetapkan Strategi Multi-Akun 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 melakukan dekomisioning 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 diterapkan: 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 [Identitas dan manajemen akses](#) dari Pilar Keamanan Kerangka Kerja Well-Architected AWS, Anda memerlukan manajemen identitas dan izin yang kuat untuk menyediakan akses ke sumber daya yang tepat bagi orang-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-langkah implementasi

- Terapkan kelompok: Dengan menggunakan grup pengguna yang ditentukan dalam kebijakan organisasi Anda, terapkan kelompok yang sesuai, jika perlu. Untuk praktik terbaik tentang pengguna, grup, dan otentikasi, lihat [Pilar Keamanan](#) dari Kerangka Kerja Well-Architected AWS.
- Terapkan peran dan kebijakan: Dengan menggunakan tindakan yang ditentukan dalam kebijakan organisasi Anda, buatlah peran dan kebijakan akses yang diperlukan. Untuk praktik terbaik tentang peran dan kebijakan, lihat [Pilar Keamanan](#) dari Kerangka Kerja Well-Architected AWS.

Sumber daya

Dokumen terkait:

- [Kebijakan terkelola AWS untuk fungsi pekerjaan](#)
- [Strategi penagihan beberapa akun AWS](#)

- [Pilar Keamanan Kerangka Kerja AWS Well-Architected](#)
- [\(IAM\) AWS Identity and Access Management](#)
- [kebijakan AWS Identity and Access Management](#)

Video terkait:

- [Mengapa menggunakan Manajemen Identitas dan Akses](#)

Contoh terkait:

- [Kontrol akses ke Wilayah AWS menggunakan kebijakan IAM](#)
- [Memulai perjalanan Manajemen Keuangan Cloud Anda: Operasi biaya cloud](#)

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 diterapkan: 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 Anda mengetahui batasan beban kerja dan lingkungan, Anda dapat menegakkan tata kelola. [AWS Budgets](#) memungkinkan Anda untuk mengatur notifikasi dan menentukan anggaran bulanan untuk biaya, penggunaan, dan diskon komitmen AWS Anda (Savings Plans dan Instans Terpesan). 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 Anda dengan AWS Budgets, gunakan [AWS Cost Anomaly Detection](#) untuk mengurangi biaya tak terduga Anda. AWS Cost Anomaly Detection adalah sebuah layanan manajemen biaya yang menggunakan machine learning untuk secara terus-menerus memantau biaya dan penggunaan Anda untuk mendeteksi pengeluaran yang tidak biasa. 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 USD). 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 AWS melalui [AWS Identity and Access Management](#) dan [Kebijakan Kontrol Layanan \(SCP\) AWS Organizations](#). IAM memungkinkan Anda untuk mengelola akses 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 menerapkan penggunaan yang benar. 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 penerapan manajemen akses, lihat [Laporan resmi Pilar Keamanan Well-Architected](#).

Tata kelola juga dapat diimplementasikan melalui manajemen [Kuota Layanan \(Service Quotas\) AWS](#). Dengan memastikan kuota layanan (service quotas) 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 \(service quotas\)](#) dapat digunakan untuk meningkatkan kuota Anda saat diperlukan.

Langkah-langkah implementasi

- Terapkan pemberitahuan tentang pengeluaran: Untuk menggunakan kebijakan organisasi yang Anda tentukan, buatlah [AWS Budgets](#) untuk memberi tahu Anda ketika pengeluaran melonjak 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 contoh umumnya 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 dimulai secara otomatis atau memerlukan persetujuan alur kerja.

- Terapkan pemberitahuan tentang pengeluaran yang tidak biasa: Gunakan [AWS Cost Anomaly Detection](#) untuk mengurangi lonjakan biaya di organisasi Anda dan analisis akar penyebab potensi terjadinya pengeluaran tak biasa. 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 AWS Cost Categories 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 yang lebih mendetail 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 pekerjaan](#)
- [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 saya](#)

Contoh terkait:

- [Contoh kebijakan-kebijakan manajemen akses IAM](#)
- [Contoh kebijakan-kebijakan kontrol layanan](#)
- [Tindakan Anggaran AWS](#)
- [Buat Kebijakan IAM untuk mengontrol akses ke sumber daya Amazon EC2 dengan menggunakan Tag](#)
- [Batasi akses Identitas IAM ke sumber daya Amazon EC2 tertentu](#)
- [Integrasi Slack untuk Deteksi Anomali Biaya menggunakan Amazon Q Developer dalam aplikasi obrolan](#)

COST02-BP06 Lacak 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 diterapkan: Rendah

Panduan implementasi

Dengan melacak siklus hidup proyek secara efektif, organisasi dapat mencapai pengendalian biaya yang lebih baik melalui perencanaan, manajemen, dan optimalisasi sumber daya yang ditingkatkan. 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, mereka dapat dinonaktifkan atau diadopsi. Periksa tahapan beban kerja sebelumnya. Setelah beban kerja diproduksi, lingkungan sebelumnya bisa dinonaktifkan atau jauh dikurangi kapasitasnya sampai diperlukan lagi.

Anda dapat memberikan tag pada sumber daya dengan jangka waktu atau pengingat untuk menyematkan waktu peninjauan beban kerja. Misalnya, jika lingkungan pengembangan terakhir kali

ditinjau beberapa bulan lalu, sekarang mungkin adalah saat yang tepat untuk meninjaunya lagi untuk mempelajari apakah layanan baru dapat diadopsi atau apakah lingkungan sedang digunakan. Anda dapat mengelompokkan dan menandai aplikasi Anda AWS untuk mengelola dan melacak metadata seperti kekritisan, lingkungan, ulasan terakhir, dan pusat biaya. [myApplications](#) Anda dapat melacak siklus hidup beban kerja Anda dan memantau serta mengelola biaya, kesehatan, postur keamanan, dan kinerja aplikasi Anda.

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

Mirip dengan [Manajemen Siklus Hidup Aplikasi \(ALM\)](#), melacak siklus hidup proyek harus melibatkan beberapa proses, alat, dan tim yang bekerja sama, 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-langkah implementasi

- Tetapkan proses pemantauan siklus hidup proyek: [Tim Keunggulan Pusat Cloud](#) 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 peninjauan beban kerja: Seperti yang ditentukan oleh kebijakan organisasi Anda, siapkan jadwal rutin untuk mengaudit proyek yang ada dan lakukan peninjauan 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 menonaktifkan penuh atau sebagian.

Sumber daya

Dokumen terkait:

- [Panduan untuk Menandai AWS](#)
- [Apa itu ALM \(Manajemen Siklus Hidup Aplikasi\)?](#)
- [Kebijakan terkelola AWS untuk fungsi pekerjaan](#)

Contoh terkait:

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

Alat Terkait:

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

Pemantauan biaya dan penggunaan

Mampukan tim untuk mengambil tindakan atas biaya dan penggunaan mereka melalui visibilitas yang mendetail pada 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 yang akan datang, dan implementasi mekanisme yang memadai untuk menyelaraskan biaya dan penggunaan sesuai dengan tujuan organisasi Anda. Area-area berikut ini diperlukan untuk memantau biaya dan penggunaan Anda:

Praktik terbaik

- [COST03-BP01 Mengonfigurasi sumber informasi yang mendetail](#)
- [COST03-BP02 Tambahkan informasi organisasi ke biaya dan penggunaan](#)
- [COST03-BP03 Mengidentifikasi kategori atribusi biaya](#)
- [COST03-BP04 Menetapkan 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

Siapkan alat manajemen biaya dan pelaporan untuk meningkatkan analisis dan transparansi data biaya dan penggunaan. Konfigurasi beban kerja Anda untuk membuat entri log yang memfasilitasi pelacakan dan segregasi biaya dan penggunaan.

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

Anda dapat menggunakan Ekspor Data AWS untuk membuat ekspor AWS Cost and Usage Report (CUR) 2.0. Ini adalah cara baru dan direkomendasikan untuk menerima data biaya dan penggunaan mendetail Anda dari AWS. Cara ini menyediakan tingkat detail harian atau per jam, tarif, biaya, dan atribut penggunaan untuk semua layanan AWS yang dikenai biaya (informasi yang sama seperti CUR), beserta beberapa perbaikan. Semua dimensi yang memungkinkan ada di dalam CUR, seperti pemberian tag, lokasi, atribut sumber daya, dan ID akun.

Ada tiga jenis ekspor berdasarkan jenis ekspor yang ingin Anda buat: ekspor data standar, ekspor ke dasbor biaya dan penggunaan dengan integrasi QuickSight, atau ekspor data lama.

- Ekspor data standar: Ekspor tabel yang sudah disesuaikan yang dikirimkan ke Amazon S3 secara berulang.
- Dasbor biaya dan penggunaan: Ekspor dan integrasi ke QuickSight untuk men-deploy dasbor biaya dan penggunaan yang sudah dibuat sebelumnya.

- Ekspor data lama: Ekspor warisan AWS Cost and Usage Report (CUR).

Anda dapat membuat ekspor data dengan penyesuaian berikut:

- Sertakan ID sumber daya
- Data alokasi biaya terpisah
- Tingkat detail per jam
- Penentuan versi
- Jenis kompresi dan format file

Untuk beban kerja yang menjalankan kontainer di Amazon ECS atau Amazon EKS, aktifkan data alokasi biaya terpisah sehingga Anda dapat mengalokasikan biaya kontainer ke unit bisnis dan tim individual, berdasarkan cara beban kerja kontainer Anda menggunakan sumber daya komputasi dan memori bersama. Data alokasi biaya terpisah memasukkan data biaya dan penggunaan untuk sumber daya tingkat kontainer baru ke AWS Cost and Usage Report. Data alokasi biaya terpisah dihitung dengan menghitung biaya layanan dan tugas ECS individual yang berjalan di kluster.

Dasbor biaya dan penggunaan mengeksport tabel dasbor biaya dan penggunaan ke bucket S3 secara berulang dan melakukan deployment dasbor biaya dan penggunaan siap pakai ke QuickSight. Gunakan opsi ini jika Anda ingin melakukan deployment dasbor data biaya dan penggunaan dengan cepat tanpa kemampuan untuk penyesuaian.

Jika mau, Anda masih dapat mengeksport CUR dalam mode lama, di mana Anda dapat mengintegrasikan layanan-layanan pemrosesan lainnya seperti [AWS Glue](#) untuk menyiapkan data untuk menganalisis dan melakukan analisis data dengan [Amazon Athena](#) dengan menggunakan SQL untuk melakukan kueri data.

Langkah-langkah implementasi

- Buat ekspor data: Buat ekspor yang disesuaikan dengan data yang Anda inginkan dan kendalikan skema ekspor Anda. Buat ekspor data manajemen tagihan dan biaya menggunakan SQL dasar, dan visualisasikan data penagihan dan manajemen biaya Anda dengan mengintegrasikan dengan QuickSight. Anda juga dapat mengeksport data Anda dalam mode standar untuk menganalisis data Anda dengan alat pemrosesan lain seperti Amazon Athena.
- Konfigurasi laporan biaya dan penggunaan: Dengan 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 info detail per jam di Cost Explorer: Untuk mengakses data biaya dan penggunaan dengan perincian per jam selama 14 hari terakhir, pertimbangkan untuk mengaktifkan data per jam dan data tingkat sumber daya di konsol penagihan.
- Konfigurasi pencatatan log aktivitas dalam aplikasi: Pastikan bahwa aplikasi Anda mencatat setiap hasil bisnis yang dicapai sehingga bisa dilacak dan diukur. Pastikan tingkat detail data ini setidaknya per jam sehingga sesuai dengan data biaya dan penggunaan. Untuk detail lebih lanjut tentang pencatatan aktivitas dan pemantauan, lihat [Pilar Keunggulan Operasional Well-Architected](#).

Sumber daya

Dokumen terkait:

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

Contoh terkait:

- [Penyiapan Akun AWS](#)
- [Ekspor Data untuk Manajemen Penagihan dan Biaya AWS](#)
- [Kasus Penggunaan Umum AWS Cost Explorer](#)

COST03-BP02 Tambahkan 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 diterapkan: Sedang

Panduan implementasi

Terapkan [pemberian tag di AWS](#) untuk menambahkan informasi organisasi ke sumber daya Anda, yang kemudian akan ditambahkan ke informasi biaya dan penggunaan Anda. Tag adalah pasangan kunci-nilai — kunci perlu ditentukan dan harus unik di seluruh organisasi, dan nilainya unik untuk grup sumber daya. Sebuah contoh dari suatu pasangan data kunci dan nilai adalah kunci `Environment`, dengan nilai `Production`. Semua sumber daya dalam lingkungan produksi pasti memiliki pasangan kunci-nilai ini. Dengan pemberian tag, 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 Anda menerapkan tag ke AWS sumber daya Anda (seperti Amazon Elastic Compute Cloud instance atau Amazon Simple Storage Service bucket) dan mengaktifkan tag, AWS tambahkan informasi ini ke Laporan Biaya dan Penggunaan Anda. 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 menerapkan standar AWS penandaan di seluruh akun organisasi membantu Anda mengelola dan mengatur AWS lingkungan Anda secara konsisten dan seragam. Gunakan [Kebijakan Tag](#) AWS Organizations untuk menentukan aturan tentang bagaimana tag dapat digunakan pada AWS sumber daya di akun Anda AWS Organizations. Kebijakan Tag memungkinkan Anda untuk dengan mudah mengadopsi pendekatan standar untuk menandai sumber daya AWS

[AWS Tag Editor](#) memungkinkan Anda untuk menambahkan, menghapus, dan mengelola tag dari beberapa 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 untuk menetapkan arti organisasi untuk biaya Anda, 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-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.
- Tag sumber daya: Untuk menggunakan kategori atribusi biaya [tempatkan tag](#) di semua sumber daya di beban kerja Anda berdasarkan kategori. Gunakan alat seperti CLI, Editor Tag, atau AWS Systems Manager untuk meningkatkan efisiensi.
- Implementasikan AWS Cost Categories: Anda dapat membuat [Cost Categories](#) tanpa menerapkan penandaan. 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 bahwa semua sumber daya memiliki tag yang lengkap dan konsisten, otomatisasi proses penambahan tag pada sumber daya saat sumber daya dibuat. Gunakan layanan seperti [AWS CloudFormation](#) untuk memverifikasi bahwa sumber daya mendapat tag saat dibuat. Anda juga dapat membuat solusi yang dapat disesuaikan untuk menambahkan tag secara otomatis menggunakan fungsi Lambda atau gunakan layanan mikro yang akan memeriksa beban kerja secara rutin dan menghapus sumber daya yang tidak memiliki tag, yang mana merupakan kondisi ideal untuk pengujian dan pengembangan lingkungan.
- Awasi dan laporkan pemberian tag: Untuk memverifikasi bahwa Anda memberi tag yang lengkap dan konsisten di seluruh organisasi, awasi dan laporkan tag di seluruh beban kerja Anda. Anda dapat menggunakan [AWS Cost Explorer](#) untuk melihat biaya dari sumber daya yang mendapat tag dan tidak mendapat tag, atau gunakan layanan seperti [Editor Tag](#). 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](#)
- [AWS CloudFormation Tag Sumber Daya](#)

- [AWS Cost Categories](#)
- [Sumber daya penandaan AWS](#)
- [Menganalisis biaya Anda dengan AWS Anggaran](#)
- [Menganalisis biaya Anda dengan Cost Explorer](#)
- [Mengelola Laporan Biaya dan Penggunaan AWS](#)

Video terkait:

- [Bagaimana saya bisa menandai AWS sumber daya saya untuk membagi tagihan saya dengan pusat biaya atau proyek](#)
- [Sumber Daya Penandaan AWS](#)

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 menegakkan akuntabilitas pengeluaran, menciptakan kesadaran biaya, dan mendorong perilaku pemakaian yang efektif.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 tersentralisasi.

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 Anda dengan para pemangku kepentingan di organisasi Anda, gunakan [AWS Cost Categories](#) 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 Anda 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-kategori ini untuk berbagai produk di [AWS Manajemen Penagihan dan Biaya dan konsol 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).

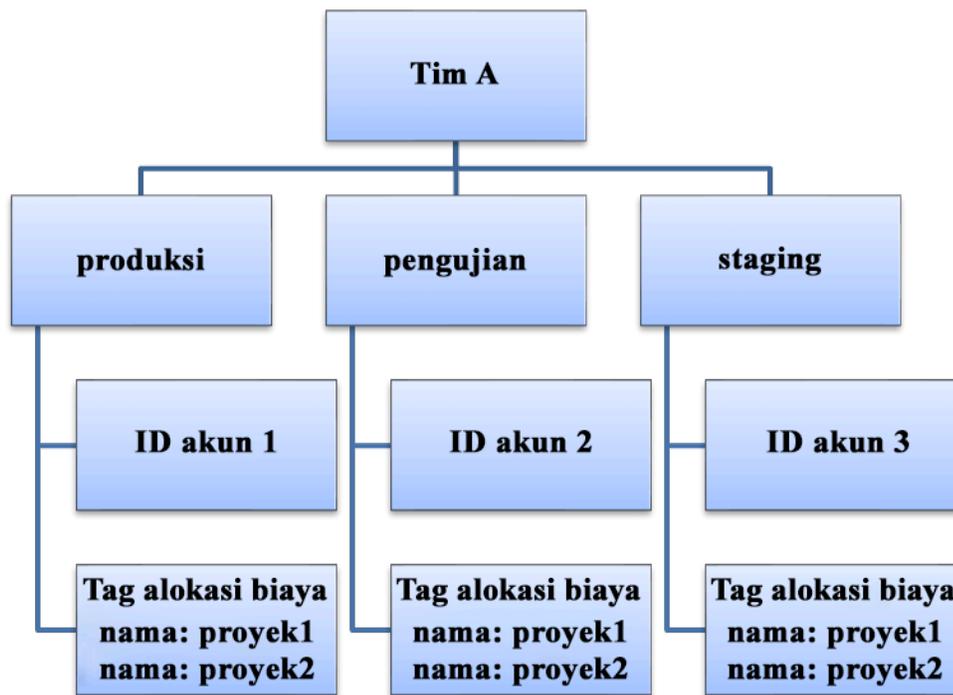


Diagram organisasi biaya dan penggunaan

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 menggunakan ini untuk memfilter nilai kategori biaya tertentu, atau grup menurut kategori biaya.

Langkah-langkah implementasi

- Tentukan kategori organisasi Anda: Temui 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: Temui 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.

- Tentukan AWS Cost Categories: Buat kategori biaya untuk mengatur informasi biaya dan penggunaan Anda dengan menggunakan [AWS Cost Categories](#) dan tentukan biaya dan penggunaan AWS Anda ke dalam [kategori yang bermakna](#). Beberapa kategori dapat ditetapkan untuk satu sumber daya, dan satu sumber daya dapat berada dalam beberapa kategori yang berbeda, jadi tentukan sebanyak kategori yang dibutuhkan sehingga Anda dapat [mengelola biaya Anda](#) ke dalam struktur yang dikategorikan menggunakan AWS Cost Categories.

Sumber daya

Dokumen terkait:

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

Contoh terkait:

- [Kelola biaya dan penggunaan Anda dengan AWS Cost Categories](#)
- [Mengelola biaya Anda dengan AWS Cost Categories](#)

COST03-BP04 Menetapkan metrik organisasi

Tetapkan 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 diterapkan: 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-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, tetapkan 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:

- [Sumber daya penandaan AWS](#)
- [Menganalisis biaya Anda dengan AWS Anggaran](#)
- [Menganalisis biaya Anda dengan Cost Explorer](#)
- [Mengelola Laporan Biaya dan Penggunaan AWS](#)

COST03-BP05 Mengonfigurasi alat manajemen penagihan dan biaya

Konfigurasi alat manajemen biaya yang 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 dengan sumber daya dan optimalisasi harga.

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

Panduan implementasi

Untuk membangun akuntabilitas yang kuat, pertimbangkan strategi akun Anda 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, berikan akses kepada pengguna ke alat yang memberikan visibilitas tentang biaya dan penggunaan mereka. AWS menyarankan Anda untuk mengonfigurasi semua beban kerja dan tim untuk tujuan berikut:

- **Atur:** Tetapkan acuan dasar alokasi dan tata kelola biaya dengan strategi pemberian tag dan taksonomi Anda sendiri. Buat beberapa Akun AWS dengan alat seperti AWS Control Tower atau AWS Organization. Berikan tag pada sumber daya AWS dan kategorikan sumber daya tersebut secara bermakna berdasarkan struktur organisasi Anda (berdasarkan unit bisnis, departemen, atau proyek). Beri tag pada nama akun untuk pusat-pusat biaya tertentu dan petakan dengan AWS Cost Categories untuk mengelompokkan akun unit bisnis untuk pusat biaya mereka sehingga pemilik unit bisnis dapat melihat konsumsi beberapa akun di satu tempat.
- **Akses:** Lacak informasi penagihan di seluruh organisasi dalam penagihan terkonsolidasi. 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, kebijakan IAM dan pemberitahuan anggaran. Misalnya, Anda dapat mengizinkan tim untuk membuat sumber daya tertentu di wilayah yang dipilih hanya dengan menggunakan mekanisme kontrol yang efektif dan mencegah pembuatan sumber daya tanpa tag khusus (seperti pusat biaya).
- **Kondisi saat ini:** Konfigurasi dasbor yang menampilkan tingkat biaya dan penggunaan saat ini. Dasbor harus tersedia di tempat yang mudah terlihat di dalam lingkungan kerja seperti dasbor operasi. Anda dapat mengeksport data dan menggunakan Dasbor Biaya dan Penggunaan dari Hub Optimisasi Biaya AWS atau produk apa pun yang didukung untuk membuat visibilitas ini. Anda mungkin perlu membuat dasbor yang berbeda untuk persona yang berbeda. Misalnya, dasbor manajer mungkin berbeda dari dasbor rekayasa.
- **Pemberitahuan:** Munculkan pemberitahuan ketika biaya atau penggunaan melebihi batas yang ditentukan dan terjadi ketidakwajaran dengan Anggaran AWS atau Deteksi Anomali Biaya AWS.

- **Laporan:** Meringkas semua informasi biaya dan penggunaan Tingkatkan kesadaran dan akuntabilitas pengeluaran cloud Anda dengan data biaya yang mendetail dan dapat diatribusikan. Buat laporan yang relevan dengan tim yang memakai data tersebut dan berisi rekomendasi.
- **Pelacakan:** Tampilkan biaya dan penggunaan saat ini dan bandingkan dengan tujuan atau target yang telah ditentukan.
- **Analisis:** Beri izin anggota tim untuk melakukan analisis khusus dan mendalam hingga perincian per jam, per hari atau per bulan dengan filter yang berbeda-beda (sumber daya, akun, tag, dll.).
- **Inspeksi:** Ikuti terus perkembangan deployment sumber daya dan peluang untuk mengoptimalkan biaya Anda. Dapatkan notifikasi menggunakan Amazon CloudWatch, Amazon SNS, atau Amazon SES, untuk deployment sumber daya di tingkat organisasi. Tinjau rekomendasi optimalisasi biaya dengan AWS Trusted Advisor atau AWS Compute Optimizer.
- **Laporan tren:** Menampilkan keberagaman biaya dan penggunaan selama periode waktu yang diinginkan, dengan detail yang diperlukan.
- **Prakiraan:** Tampilkan prakiraan biaya di masa depan, prakiraan penggunaan sumber daya Anda, dan lacak pengeluaran dengan dasbor prakiraan yang Anda buat.

Anda dapat menggunakan [Hub Optimisasi Biaya AWS](#) untuk memahami potensi penghematan biaya yang dikonsolidasikan dari lokasi terpusat dan membuat ekspor data untuk integrasi dengan Amazon Athena. Anda juga dapat menggunakan Hub Optimisasi Biaya AWS untuk melakukan deployment Dasbor Biaya dan Penggunaan, yang memanfaatkan QuickSight untuk analisis biaya interaktif dan pembagian wawasan biaya yang aman.

Jika organisasi Anda tidak memiliki keterampilan utama atau sumber daya yang diperlukan, Anda dapat bekerja sama dengan [AWS ProServ](#), [AWS Managed Services \(AMS\)](#), atau [Mitra AWS](#). Anda juga dapat menggunakan alat pihak ketiga tetapi pastikan Anda memvalidasi proposisi nilainya.

Langkah-langkah implementasi

- **Beri izin akses berbasis tim ke alat-alat:** Konfigurasi akun Anda dan buat kelompok yang memiliki akses ke laporan biaya dan penggunaan yang diperlukan untuk mereka pakai 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.
- **Kelola Tag dan Kategori Biaya:** atur biaya Anda di seluruh tim, unit bisnis, aplikasi, lingkungan, dan proyek. Gunakan tag sumber daya untuk mengatur biaya, berdasarkan tag alokasi biaya.

Buat Kategori Biaya berdasarkan dimensi dengan menggunakan tag, akun, layanan, dll. untuk memetakan biaya Anda.

- Konfigurasi Anggaran AWS: [Konfigurasi Anggaran AWS](#) di semua akun untuk beban kerja Anda. Tetapkan anggaran untuk pengeluaran akun secara keseluruhan, serta anggaran untuk beban kerja menggunakan tag dan kategori biaya. Konfigurasi notifikasi di AWS Budgets untuk mendapatkan peringatan saat Anda melebihi jumlah yang dianggarkan, atau saat perkiraan biaya Anda melebihi anggaran.
- Tentukan Deteksi Ketidakwajaran Biaya AWS: Gunakan [Deteksi Ketidakwajaran Biaya AWS](#) untuk akun Anda, layanan inti atau kategori biaya yang Anda buat untuk memantau biaya dan penggunaan Anda dan lacak pengeluaran tak wajar. Anda dapat menerima 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.
- Gunakan alat analisis biaya: Tentukan [AWS Cost Explorer](#) untuk beban kerja dan akun Anda guna memvisualisasikan data biaya untuk analisis lebih lanjut. Buat dashboard untuk beban kerja yang melacak pengeluaran secara keseluruhan, metrik penggunaan utama untuk beban kerja, dan perkiraan biaya mendatang berdasarkan data biaya historis Anda.
- Gunakan alat analisis hemat biaya: Gunakan Hub Optimisasi Biaya AWS untuk mengidentifikasi peluang penghematan dengan rekomendasi yang disesuaikan termasuk menghapus sumber daya yang tidak digunakan, penyesuaian ukuran sesuai kebutuhan, komitmen Savings Plans, pencadangan, dan rekomendasi pengoptimalan komputasi.
- Sesuaikan fitur-fitur lanjutan yang tersedia: Anda dapat secara opsional membuat visualisasi untuk memudahkan analisis interaktif dan berbagi wawasan biaya. Dengan Ekspor Data di Hub Optimisasi Biaya AWS, Anda dapat membuat dashboard biaya dan penggunaan yang didukung oleh QuickSight untuk organisasi Anda yang memberikan detail dan tingkat detail tambahan. Anda juga dapat menerapkan kemampuan analisis lanjutan dengan menggunakan ekspor data di [Amazon Athena](#) untuk kueri lanjutan, dan membuat dashboard di [QuickSight](#). Bekerja sama dengan [Mitra AWS](#) untuk mengadopsi solusi manajemen cloud untuk pemantauan dan pengoptimalan penagihan cloud terkonsolidasi.

Sumber daya

Dokumen terkait:

- [Apa itu Manajemen AWS Manajemen Penagihan dan Biaya dan Biaya?](#)
- [Membangun lingkungan AWS praktik terbaik Anda](#)

- [Praktik Terbaik untuk Pemberian Tag Sumber Daya AWS](#)
- [Pemberian tag pada sumber daya AWS Anda](#)
- [AWS Cost Categories](#)
- [Menganalisis biaya Anda dengan AWS Budgets](#)
- [Menganalisis biaya Anda dengan AWS Cost Explorer](#)
- [Apa itu Ekspor Data AWS?](#)

Video terkait:

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

Contoh terkait:

- [Dasbor Biaya dan Penggunaan yang didukung](#) oleh QuickSight
- [Lokakarya Tata Kelola Biaya dan Penggunaan AWS](#)

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 dengan layanan analitik, yang dapat menyediakan wawasan dan kemampuan charge back.

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

Panduan implementasi

Optimisasi biaya adalah 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. Lakukan analisis ini pada level per jam. Efisiensi umumnya akan berubah jika Anda memiliki 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 membantu Anda fokus pada aktivitas optimalisasi Anda.

Membuat metrik beban kerja untuk sumber daya bersama mungkin sulit dibandingkan dengan sumber daya seperti aplikasi kontainer di Amazon Elastic Container Service (Amazon) ECS dan Amazon Gateway. API Namun, ada beberapa cara tertentu bagi Anda untuk mengkategorikan penggunaan dan melacak biaya. Jika Anda perlu melacak Amazon ECS dan sumber daya AWS Batch bersama, Anda dapat mengaktifkan data alokasi biaya terpisah. 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.

Langkah-langkah implementasi

- Alokasikan metrik biaya per beban kerja: Dengan menggunakan metrik dan tag terkonfigurasi yang sudah ditetapkan, buatlah sebuah metrik yang menggabungkan hasil akhir beban kerja dan biaya beban kerja. Gunakan layanan analitik seperti Amazon Athena dan Amazon QuickSight untuk membuat dasbor efisiensi untuk keseluruhan beban kerja dan komponen apa pun.

Sumber daya

Dokumen terkait:

- [Sumber daya penandaan AWS](#)
- [Menganalisis biaya Anda dengan AWS Anggaran](#)
- [Menganalisis biaya Anda dengan Cost Explorer](#)
- [Mengelola Laporan Biaya dan Penggunaan AWS](#)

Contoh terkait:

- [Meningkatkan visibilitas biaya Amazon ECS dan AWS Batch dengan AWS Split Cost Allocation Data](#)

Nonaktifkan sumber daya

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 ada pemiliknya.

Praktik terbaik

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

COST04-BP01 Lacak sumber daya selama 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 diterapkan: 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 fitur adalah `feature-X testing` untuk mengidentifikasi tujuan sumber daya dalam hal siklus hidup beban kerja. Contoh lain adalah menggunakan `LifeSpan` atau `TTL` untuk sumber daya, seperti nama kunci `to-be-deleted tag` dan nilai untuk menentukan periode waktu atau waktu tertentu untuk penonaktifan.

Langkah-langkah implementasi

- Terapkan skema pemberian tag: Terapkan skema pemberian tag yang mengidentifikasi beban kerja tempat sumber daya tersebut berada, verifikasi bahwa semua sumber daya dalam beban kerja diberi tag yang sesuai. Pemberian tag akan membantu Anda membuat kategori sumber daya berdasarkan tujuan, tim, lingkungan, atau kriteria lain yang relevan dengan bisnis Anda. Untuk detail selengkapnya tentang kasus, strategi, dan teknik penggunaan pemberian tag, lihat [Praktik Terbaik Pemberian Tag AWS](#).
- Menerapkan throughput beban kerja atau pemantauan output: Terapkan pemantauan atau peringatan throughput beban kerja, dimulai dari 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 lebih lanjut tentang sumber daya yang tidak digunakan atau kurang dimanfaatkan, lihat [pemeriksaan Pengoptimalan Biaya AWS Trusted Advisor](#).
- AWS Sumber daya grup: Buat grup untuk AWS sumber daya. Anda dapat menggunakan [AWS Resource Groups](#) untuk mengatur dan mengelola AWS sumber daya Anda yang sama Wilayah AWS. Anda dapat menambahkan tag ke sebagian besar sumber daya Anda untuk membantu mengidentifikasi dan menyortir sumber daya Anda di dalam organisasi Anda. Gunakan [Editor Tag](#) untuk menambahkan tag ke sumber daya yang didukung secara massal. Pertimbangkan untuk menggunakan [AWS Service Catalog](#) untuk membuat, mengelola dan mendistribusikan portofolio produk-produk yang disetujui ke para pengguna akhir dan kelola siklus hidup produk.

Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [AWS Trusted Advisor Pemeriksaan Optimalisasi Biaya](#)
- [Sumber daya penandaan AWS](#)
- [Memublikasikan Metrik Kustom](#)

Video terkait:

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

Contoh terkait:

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

COST04-BP02 Menerapkan proses dekomisioning

Implementasikan proses untuk mengidentifikasi dan menonaktifkan sumber daya tidak terpakai

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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-langkah implementasi

- Buat dan terapkan proses penonaktifan: Bekerjalalah dengan para pengembang 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 para pemilik beban kerja bahwa proses penonaktifan telah dimulai.

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 memenuhi syarat untuk dinonaktifkan di 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.
- Koordinasi dan komunikasi: Bekerja sama dengan para pemilik beban kerja untuk mengonfirmasi sumber daya yang akan dinonaktifkan

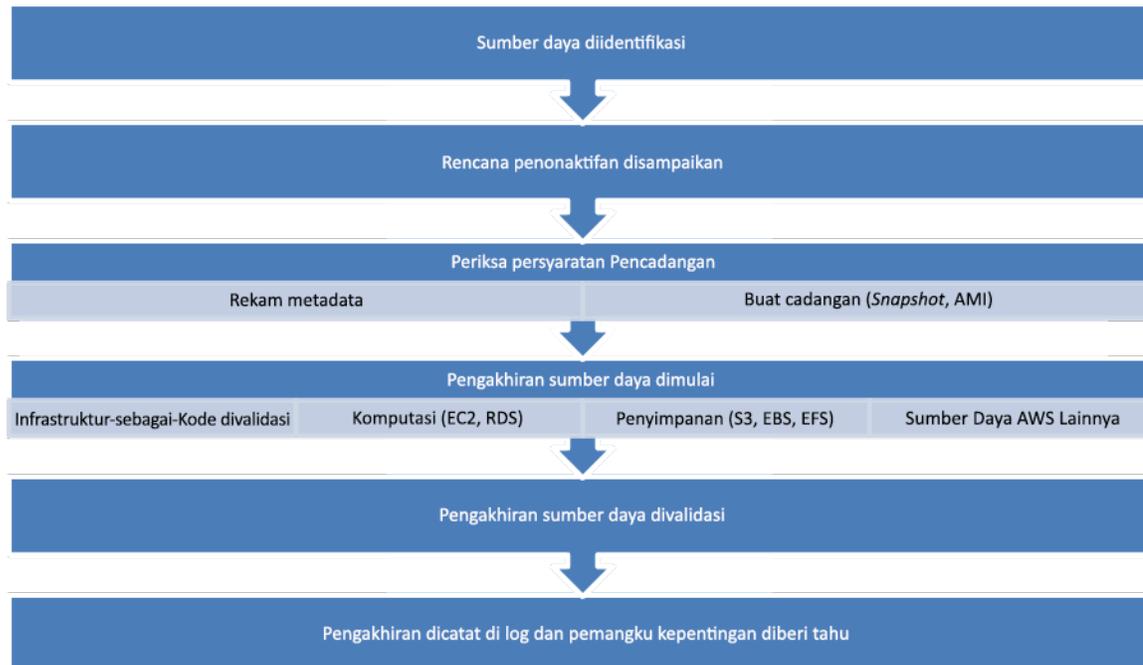
- Rekam metadata dan buat cadangan: Rekam metadata (seperti publikIPs, Wilayah, AZ,, SubnetVPC, dan Grup Keamanan) dan buat cadangan (seperti snapshot atau pengambilan Amazon Elastic Block Store, ekspor kunciAMI, dan ekspor Sertifikat) jika diperlukan untuk sumber daya di lingkungan produksi atau jika itu adalah sumber daya penting.
- Validasi infrastructure-as-code: Tentukan apakah sumber daya diterapkan dengan AWS CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK), atau alat infrastructure-as-code penerapan lainnya sehingga dapat diterapkan kembali jika perlu.
- Mencegah akses: Terapkan kendali batasan untuk jangka waktu tertentu, untuk mencegah penggunaan sumber daya saat Anda sedang memutuskan apakah sumber daya diperlukan. Lakukan verifikasi bahwa lingkungan sumber daya dapat dikembalikan ke status aslinya jika diperlukan.
- Ikuti proses penonaktifan internal Anda: Ikuti tugas administratif dan proses penonaktifan organisasi Anda, seperti menghapus sumber daya dari domain organisasi Anda, menghapus DNS catatan, dan menghapus sumber daya dari alat manajemen konfigurasi, alat pemantauan, alat otomatisasi, dan alat keamanan.

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

- Hentikan atau hentikan semua EC2 instans Amazon dan penyeimbang beban Anda. EC2Instans Amazon terlihat di konsol untuk waktu yang singkat setelah dihentikan. Anda tidak menerima tagihan untuk instans apa pun yang tidak memiliki status berjalan
- Menghapus infrastruktur penskalaan otomatis Anda
- Lepaskan semua Host Khusus.
- Hapus semua EBS volume Amazon dan EBS snapshot Amazon.
- Lepaskan semua Alamat IP elastis.
- Deregister semua Gambar Mesin Amazon (). AMIs
- Hentikan semua AWS Elastic Beanstalk lingkungan.

Jika sumber daya adalah sebuah 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 tergantung pada kelas penyimpanan yang digunakan. Untuk ringkasan durasi penyimpanan minimum untuk setiap kelas penyimpanan, lihat [Kinerja di seluruh kelas penyimpanan Amazon S3](#). Untuk detail tentang cara menghitung biaya penghapusan awal, 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.



Aliran penonaktifan sumber daya.

Sumber daya

Dokumen terkait:

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

Video terkait:

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

Contoh terkait:

- [Hapus atau hentikan sumber daya Amazon EC2](#)
- [Cari tahu pengguna mana yang meluncurkan EC2 instans Amazon](#)

COST04-BP03 Menonaktifkan sumber daya

Nonaktifkan sumber daya yang diinisiasi 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 diterapkan: 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 diinisiasi oleh perubahan status pada beban kerja, seperti produk yang mendekati akhir masa pakai atau mengalami penggantian. Pencarian dan peristiwa penonaktifan mungkin juga diinisiasi oleh peristiwa eksternal, seperti perubahan kondisi pasar atau penghentian produk.

Langkah-langkah implementasi

- Penonaktifan sumber daya: Ini adalah tahap depresiasi sumber daya AWS yang tidak lagi diperlukan atau telah berakhirnya 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](#)

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 diterapkan: 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 [Amazon EC2 Auto Scaling](#) atau [Penskalaan Otomatis Aplikasi](#) untuk melakukan proses penonaktifan. Anda juga dapat menerapkan kode khusus menggunakan [API atau SDK](#) untuk menonaktifkan sumber daya beban kerja secara otomatis.

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

Langkah-langkah implementasi

- Terapkan Amazon EC2 Auto Scaling atau Penskalaan Otomatis Aplikasi: Untuk sumber daya yang didukung, konfigurasi dengan Amazon EC2 Auto Scaling atau Penskalaan Otomatis Aplikasi. Layanan ini dapat membantu Anda mengoptimalkan pemanfaatan dan efisiensi biaya saat memakai layanan AWS. Ketika permintaan menurun, layanan-layanan ini akan menghapus kelebihan kapasitas sumber daya secara otomatis sehingga Anda dapat terhindar dari pengeluaran yang berlebihan.
- Konfigurasi CloudWatch untuk menghentikan instans: Instans-instans dapat dikonfigurasi untuk berhenti dengan menggunakan [alarm CloudWatch](#). Menggunakan metrik dari proses penonaktifan, implementasikan alarm dengan tindakan Amazon Elastic Compute Cloud. Verifikasi operasi di lingkungan non-produksi sebelum peluncuran.
- Terapkan kode dalam beban kerja: Anda dapat menggunakan SDK AWS 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 pembuatan [arsitektur nirserver](#) dan [dan arsitektur berbasis peristiwa](#) pada AWS untuk membangun dan menjalankan aplikasi Anda. AWS menawarkan beberapa layanan teknologi nirserver yang secara inheren menyediakan pemanfaatan sumber daya yang dioptimalkan secara otomatis dan penonaktifan otomatis (mengurangi skala (scale in) dan menambahkan skala (scale out)). Dengan aplikasi nirserver, pemanfaatan sumber daya dioptimalkan secara otomatis dan Anda tidak pernah membayar pengadaan yang berlebihan.

Sumber daya

Dokumen terkait:

- [Amazon EC2 Auto Scaling](#)
- [Memulai dengan Amazon EC2 Auto Scaling](#)
- [Penskalaan Otomatis Aplikasi](#)
- [AWS Trusted Advisor](#)
- [Nirserver di AWS](#)
- [Buat Alarm untuk Menghentikan, Mengakhiri, Menyalakan Ulang, atau Memulihkan sebuah Instans](#)
- [Menambahkan tindakan pengakhiran ke alarm Amazon CloudWatch](#)

Contoh terkait:

- [Jadwalkan penghapusan otomatis tumpukan AWS CloudFormation](#)

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 diterapkan: 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 Manajer Siklus Hidup Data Amazon 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 menerapkan kode kustom menggunakan [API atau SDK](#) untuk membuat kebijakan siklus hidup dan aturan kebijakan untuk objek yang akan dihapus secara otomatis.

Langkah-langkah implementasi

- Gunakan Manajer Siklus Hidup Data Amazon: Gunakan kebijakan siklus hidup di Manajer Siklus Hidup Data Amazon untuk mengotomatiskan penghapusan snapshot Amazon EBS dan AMI yang didukung Amazon EBS.
- Siapkan konfigurasi siklus hidup di bucket: Gunakan konfigurasi siklus hidup Amazon S3 di sebuah bucket untuk menentukan tindakan yang harus dilakukan Amazon S3 selama siklus hidup objek, serta penghapusan di akhir siklus hidup objek, berdasarkan kebutuhan bisnis Anda.

Sumber daya

Dokumen terkait:

- [AWS Trusted Advisor](#)
- [Amazon Data Lifecycle Manager](#)
- [Cara untuk membuat konfigurasi siklus hidup bucket Amazon S3.](#)

Video terkait:

- [Otomatiskan Snapshot Amazon EBS dengan Manajer Siklus Hidup Data Amazon](#)
- [Kosongkan bucket Amazon S3 menggunakan peraturan konfigurasi siklus hidup](#)

Contoh terkait:

- [Kosongkan bucket Amazon S3 menggunakan peraturan konfigurasi siklus hidup](#)

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 Anda membuat sumber daya yang hemat biaya:

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

Topik

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

Evaluasi biaya saat Anda memilih layanan

Praktik terbaik

- [COST05-BP01 Identifikasi persyaratan organisasi untuk biaya](#)
- [COST05-BP02 Menganalisis semua komponen beban kerja](#)
- [COST05-BP03 Melakukan analisis menyeluruh dari setiap komponen](#)
- [COST05-BP04 Pilih perangkat lunak dengan lisensi hemat biaya](#)
- [COST05-BP05 Pilih komponen beban kerja ini untuk mengoptimalkan biaya sesuai dengan prioritas organisasi](#)
- [COST05-BP06 Lakukan analisis biaya untuk penggunaan yang berbeda dari waktu ke 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 diterapkan: 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. Buat keseimbangan antara optimasi biaya dan pilar AWS Well-Architected Framework lainnya, seperti kinerja 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 memperbaiki aplikasi Anda.

Langkah-langkah implementasi

- Identifikasi kebutuhan biaya organisasi: Bertemulah dengan anggota-anggota tim dari organisasi Anda, termasuk mereka dari tim 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.
- Tangani hutang teknis dan dokumentasikan: Selama peninjauan beban kerja, tangani hutang teknis. Dokumentasikan item backlog untuk mempertahankan beban kerja pada masa mendatang,

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

Sumber daya

Praktik-praktik terbaik terkait:

- [REL11-BP07 Arsitek produk Anda untuk memenuhi target ketersediaan dan perjanjian tingkat layanan uptime \(\) SLAs](#)
- [OPS01-BP06 Mengevaluasi pengorbanan](#)

Dokumen terkait:

- [AWS Total Biaya Kepemilikan \(TCO\) Kalkulator](#)
- [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 diterapkan: Tinggi

Panduan implementasi

Komponen beban kerja, yang dirancang untuk memberikan nilai bisnis kepada organisasi, dapat mencakup berbagai layanan. Untuk setiap komponen, seseorang dapat memilih AWS Cloud layanan khusus untuk memenuhi kebutuhan bisnis. Pilihan ini dapat dipengaruhi oleh faktor-faktor seperti pemahaman atau pengalaman sebelumnya dalam menggunakan layanan-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 optimalisasi komponen spesifik tersebut. Misalnya, apabila biaya sumber daya yang diajukan adalah 10 USD per bulan, dan berdasarkan beban yang diperkirakan, 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. Gunakan perkiraan berdasarkan data yang lebih cepat dan efisien untuk 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 EMR Tanpa Server mungkin merupakan pilihan yang tepat pada awalnya. Namun, karena konsumsi untuk layanan itu meningkat, transisi ke EMR on EC2 dapat mengurangi biaya untuk komponen beban kerja tersebut.

[AWS Cost Explorer](#) dan AWS Cost and Usage Reports ([CUR](#)) dapat menganalisis biaya bukti konsep (PoC) atau lingkungan berjalan. Anda juga dapat menggunakan [AWS Kalkulator Harga](#) untuk memperkirakan biaya beban kerja.

Tulis alur kerja yang perlu diikuti oleh tim teknis untuk meninjau beban kerja mereka. Jaga agar alur kerja ini tetap sederhana, tetapi juga mencakup semua langkah yang diperlukan untuk memastikan tim memahami setiap komponen beban kerja beserta harganya. Organisasi Anda kemudian dapat mengikuti dan menyesuaikan alur kerja ini berdasarkan kebutuhan khusus setiap tim.

1. Buat daftar setiap layanan yang digunakan untuk beban kerja Anda: Ini adalah titik awal yang baik. Identifikasi semua layanan yang sedang digunakan dan dari mana biaya berasal.
2. Pahami cara kerja penetapan harga untuk layanan tersebut: Pahami [model penetapan harga](#) dari masing-masing layanan. AWS Layanan yang berbeda memiliki model harga yang berbeda berdasarkan faktor-faktor seperti volume penggunaan, transfer data, dan harga khusus fitur.
3. Fokus pada layanan yang memiliki biaya beban kerja yang tidak terduga dan yang tidak sesuai dengan penggunaan yang diharapkan dan hasil bisnis Anda: Identifikasi outlier atau layanan di mana biaya tidak sebanding dengan nilai atau penggunaan dengan penggunaan atau s. AWS Cost Explorer AWS Cost and Usage Report Untuk memprioritaskan upaya optimalisasi, penting untuk mengaitkan biaya dengan hasil bisnis.
4. AWS Cost Explorer, CloudWatch Log, Log VPC Aliran, dan Lensa Penyimpanan Amazon S3 untuk memahami akar penyebab biaya tinggi tersebut: Alat ini berperan penting dalam diagnosis biaya tinggi. Setiap layanan menawarkan lensa yang berbeda untuk melihat dan menganalisis

penggunaan dan biaya. Misalnya, Cost Explorer membantu menentukan tren biaya secara keseluruhan, CloudWatch Log memberikan wawasan operasional, VPC Flow Logs menampilkan lalu lintas IP, dan Amazon S3 Storage Lens berguna untuk analitik penyimpanan.

5. Gunakan AWS Budgets untuk menetapkan anggaran untuk jumlah tertentu untuk layanan atau akun: Menetapkan anggaran adalah cara proaktif untuk mengelola biaya. Gunakan AWS Budgets untuk menetapkan ambang anggaran khusus dan menerima peringatan ketika biaya melebihi ambang batas tersebut.
6. Konfigurasi CloudWatch alarm Amazon untuk mengirim peringatan penagihan dan penggunaan: Siapkan pemantauan dan peringatan untuk metrik biaya dan penggunaan. CloudWatch alarm dapat memberi tahu Anda ketika ambang batas tertentu dilanggar, yang meningkatkan waktu respons intervensi.

Fasilitasi penyempurnaan dan penghematan keuangan yang jelas dari waktu ke waktu melalui tinjauan strategis pada semua komponen beban kerja dan terlepas dari atributnya saat ini. Upaya yang dicurahkan dalam proses tinjauan ini harus terencana, dengan pertimbangan yang cermat terhadap potensi keuntungan yang mungkin direalisasikan.

Langkah-langkah implementasi

- **Buat daftar komponen beban kerja:** Buat daftar komponen beban kerja Anda. Gunakan daftar ini untuk memverifikasi bahwa setiap komponen telah dianalisis. Upaya yang dilakukan harus sesuai dengan kekritisitas beban kerja sesuai prioritas organisasi Anda. Kelompokkan sumber daya menurut fungsinya untuk meningkatkan efisiensi (misalnya, penyimpanan basis data produksi, jika terdapat beberapa basis data).
- **Prioritaskan daftar komponen:** Ambil daftar komponen dan prioritaskan dalam urutan usaha. Daftar tersebut umumnya diurutkan berdasarkan 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 Kalkulator Harga](#)
- [AWS Cost Explorer](#)

- [Kelas penyimpanan Amazon S3](#)
- [Produk AWS Cloud](#)

Video terkait:

- [AWS Seri Optimalisasi Biaya: CloudWatch](#)

COST05-BP03 Melakukan analisis menyeluruh dari 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 diterapkan: 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 AWS menghilangkan beban operasional dan administrasi pemeliharaan layanan, seperti menambal atau meningkatkan OS, dan memungkinkan Anda untuk fokus pada inovasi dan bisnis.

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. Misalnya, Anda mungkin ingin mengurangi jumlah waktu yang Anda habiskan untuk mengelola instance database dengan bermigrasi ke database-as-a-service platform seperti [Amazon Relational Database Service \(AmazonRDS\)](#) atau memigrasikan aplikasi Anda ke platform yang dikelola sepenuhnya 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 memodifikasi atribut AWS Managed Services

menggunakan AWS Management Console atau AWS APIs dan SDKs untuk menyelaraskan kebutuhan sumber daya dengan permintaan yang berubah. Misalnya, Anda dapat menambah atau mengurangi jumlah node di EMR klaster Amazon (atau cluster Amazon Redshift) untuk skala atau masuk.

Anda juga dapat mengemas beberapa instance pada AWS sumber daya untuk mengaktifkan penggunaan kepadatan yang lebih tinggi. Misalnya, Anda dapat menyediakan beberapa database kecil pada satu instance database Amazon Relational Database Service (RDS Amazon). Seiring bertambahnya penggunaan, Anda dapat memigrasikan salah satu database ke instans RDS database Amazon khusus menggunakan proses snapshot dan restore.

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 berkelanjutan yang diperlukan untuk memodifikasi layanan dapat dikurangi menjadi hampir nol dengan menggunakan APIs dan SDKs yang terintegrasi dengan sistem dan alat pemantauan, seperti Amazon CloudWatch.

[Amazon RDS](#), [Amazon Redshift](#), dan [Amazon ElastiCache](#) menyediakan layanan database terkelola. [Amazon Athena](#), AmazonEMR, dan [Amazon OpenSearch Service menyediakan layanan](#) analitik terkelola.

[AMS](#) adalah layanan yang mengoperasikan AWS infrastruktur atas nama pelanggan dan mitra perusahaan. Layanan ini menyediakan lingkungan yang aman dan patuh untuk melakukan deployment beban kerja Anda. AMS menggunakan model operasi cloud perusahaan dengan otomatisasi untuk memungkinkan Anda memenuhi persyaratan organisasi Anda, pindah ke cloud lebih cepat, dan mengurangi biaya manajemen yang sedang berlangsung.

Langkah-langkah implementasi

- Lakukan analisis yang menyeluruh: Dengan menggunakan daftar komponen tersebut, kerjakan setiap komponen mulai dari prioritas tertinggi ke prioritas 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 yang dikelola dan tidak dikelola: Pertimbangkan biaya operasional untuk sumber daya yang Anda kelola dan bandingkan dengan sumber daya yang AWS dikelola.

Misalnya, tinjau database Anda yang berjalan di EC2 instans Amazon dan bandingkan dengan RDS opsi Amazon (layanan AWS terkelola) atau Amazon EMR dibandingkan dengan menjalankan Apache Spark di Amazon. EC2 Saat beralih dari beban kerja yang dikelola sendiri ke beban kerja yang dikelola AWS sepenuhnya, teliti opsi Anda dengan cermat. Tiga faktor terpenting yang perlu dipertimbangkan adalah [jenis 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:

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

Video terkait:

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

Contoh terkait:

- [Mengapa harus berpindah ke basis data terkelola](#)
- [Mengkonsolidasikan data dari database SQL Server identik ke dalam satu database Amazon RDS untuk SQL Server menggunakan AWS DMS](#)
- [Kirimkan data dalam skala besar ke Amazon Managed Streaming for Apache Kafka \(Amazon\) MSK](#)
- [Migrasi an. ASP NET aplikasi web untuk AWS Elastic Beanstalk](#)

COST05-BP04 Pilih perangkat lunak dengan lisensi hemat biaya

Perangkat lunak sumber terbuka meniadakan biaya lisensi perangkat lunak yang dapat menambah biaya yang besar pada beban kerja. Jika perangkat lunak berlisensi diperlukan, hindari lisensi yang

terikat pada atribut arbitrer seperti CPUs, cari lisensi yang terikat pada output atau hasil. Besar kecilnya biaya lisensi ini lebih sesuai dengan manfaat yang disediakan.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 insinyur, perkiraan penggunaan, atau ketergantungan teknologi lainnya, organisasi dapat mempertimbangkan untuk menggunakan perangkat lunak open source 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 diskalakan secara independen dari throughput atau penggunaan, seperti lisensi yang diskalakan oleh perangkat keras (lisensi CPU terikat). Lisensi jenis ini harus dihindari karena biaya dapat meningkat pesat tanpa hasil yang seimbang.

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

Ini [AWS Kalkulator Harga](#) menawarkan cara komprehensif untuk membandingkan biaya berbagai sumber daya dengan opsi lisensi yang berbeda, seperti RDS instans Amazon dan mesin basis data yang berbeda. Selain itu, AWS Cost Explorer memberikan perspektif yang tak ternilai untuk biaya beban kerja yang ada, terutama yang datang dengan lisensi yang berbeda. Untuk manajemen lisensi, [AWS License Manager](#) menawarkan metode efisien untuk mengawasi dan menangani lisensi perangkat lunak. Pelanggan dapat menerapkan dan mengoperasikan perangkat lunak sumber terbuka pilihan mereka di AWS Cloud.

Langkah-langkah implementasi

- Analisis opsi lisensi: Tinjau persyaratan 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 riwayat penetapan harga 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 di AWS](#)
- [AWS Total Biaya Kepemilikan \(TCO\) Kalkulator](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)

Contoh terkait:

- [Blog Sumber Terbuka](#)
- [AWS Blog Sumber Terbuka](#)
- [Evaluasi Optimalisasi dan Pemberian Lisensi](#)

COST05-BP05 Pilih komponen beban kerja ini untuk mengoptimalkan biaya sesuai dengan prioritas organisasi

Pertimbangkan biaya saat memilih semua komponen untuk beban kerja Anda. Termasuk di antaranya adalah menggunakan layanan terkelola dan tingkat aplikasi atau nirsumber, kontainer, atau arsitektur yang berbasis 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 diterapkan: Sedang

Panduan implementasi

Pertimbangkan biaya layanan dan opsi saat memilih semua komponen. Ini termasuk menggunakan tingkat aplikasi dan layanan terkelola, seperti [Amazon Relational Database Service \(Amazon\), RDS Amazon DynamoDB, Amazon Simple Notification Service \(Amazon\), dan SNS Amazon Simple Email Service \(SESAmazon\)](#) untuk mengurangi biaya organisasi secara keseluruhan.

Gunakan layanan nirserver dan kontainer untuk komputasi, seperti [AWS Lambda](#) dan [Amazon Simple Storage Service](#) (Amazon S3) untuk situs web statis. Kontainerisasi aplikasi Anda jika memungkinkan dan gunakan AWS Managed Container Services seperti [Amazon Elastic Container Service](#) (AmazonECS) 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 tanpa server atau tingkat aplikasi seperti Lambda, Amazon Simple Queue Service \(Amazon\), AmazonSQS, dan Amazon SNS 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.

Menggunakan [arsitektur berbasis peristiwa](#) juga dimungkinkan 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. Ini berarti lebih sedikit konsumsi bandwidth jaringan, CPU pemanfaatan yang lebih sedikit, kapasitas armada yang tidak aktif, dan lebih SSL TLS sedikit/jabat tangan.

Untuk informasi lebih lanjut tentang layanan nirserver, lihat [Laporan resmi lensa Aplikasi Well-Architected](#).

Langkah-langkah implementasi

- Pilih setiap layanan untuk mengoptimalkan biaya: Dengan menggunakan daftar dan analisis yang telah Anda prioritaskan, pilih setiap opsi yang menyediakan pilihan terbaik sesuai 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.

Misalnya, jika Anda perlu meninjau lalu lintas yang diharapkan untuk database Anda AWS, pertimbangkan untuk meningkatkan ukuran instans atau menggunakan ElastiCache layanan Amazon (Redis atau Memcached) untuk menyediakan mekanisme cache untuk database Anda.

- Evaluasi arsitektur berbasis peristiwa: Menggunakan arsitektur nirserver juga memungkinkan Anda membangun arsitektur berbasis peristiwa untuk aplikasi berbasis layanan mikro yang terdistribusi, yang membantu Anda membangun solusi yang dapat diskalakan, tangguh, gesit, dan hemat biaya.

Sumber daya

Dokumen terkait:

- [AWS Total Biaya Kepemilikan \(TCO\) Kalkulator](#)
- [Nirserver AWS](#)
- [Apa yang Dimaksud Arsitektur Berbasis Peristiwa?](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)
- [Amazon ElastiCache \(Redis\) OSS](#)

Contoh terkait:

- [Mulai menggunakan arsitektur berbasis peristiwa](#)
- [Arsitektur berbasis peristiwa](#)
- [Bagaimana Statsig menjalankan 100x lebih hemat biaya menggunakan Amazon \(Redis\) ElastiCache OSS](#)
- [Praktik terbaik untuk bekerja dengan AWS Lambda fungsi](#)

COST05-BP06 Lakukan analisis biaya untuk penggunaan yang berbeda dari waktu ke waktu

Beban kerja bisa berubah seiring 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 diterapkan: Sedang

Panduan implementasi

Saat AWS merilis layanan dan fitur baru, layanan optimal untuk beban kerja Anda dapat 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. Inisiasi 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 dalam AWS Cloud, Anda dapat memilih berbagai macam AWS penawaran layanan dan alat mitra untuk membantu Anda memigrasikan kumpulan data Anda, apakah itu file, database, gambar mesin, volume blok, atau bahkan backup tape. Misalnya, untuk memindahkan sejumlah besar data ke dan dari AWS atau memproses data di tepi, Anda dapat menggunakan salah satu perangkat yang AWS dibuat khusus untuk memindahkan petabyte data secara offline secara efektif. Contoh lain 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.

Lakukan analisis biaya untuk penggunaan yang berbeda dari waktu ke waktu dengan berdiskusi dengan para pemangku kepentingan yang ada di organisasi Anda dan gunakan fitur perkiraan [AWS Cost Explorer](#) untuk memprediksi dampak yang mungkin terjadi dari adanya perubahan layanan. Pantau peluncuran tingkat penggunaan menggunakan AWS Budgets, CloudWatch menagih alarm dan AWS Cost Anomaly Detection untuk mengidentifikasi dan menerapkan layanan yang paling hemat biaya lebih cepat.

Langkah-langkah implementasi

- Tentukan pola penggunaan yang terprediksi: Bersama organisasi Anda, seperti pemasaran dan pemilik produk, buatlah dokumentasi yang memuat pola penggunaan yang diharapkan dan 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 kalender, minggu, atau bulan di mana Anda mengharapkan lebih banyak pengguna untuk menggunakan AWS sumber daya Anda, yang menunjukkan bahwa Anda harus meningkatkan kapasitas sumber daya yang ada atau mengadopsi layanan tambahan untuk mengurangi biaya dan meningkatkan kinerja.

- Lakukan analisis biaya pada penggunaan yang terprediksi: Dengan menggunakan pola penggunaan yang sudah ditentukan, lakukan analisis pada masing-masing titik 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:

- [AWS Total Biaya Kepemilikan \(TCO\) Kalkulator](#)
- [Kelas penyimpanan Amazon S3](#)
- [Produk cloud](#)
- [EC2Auto Scaling Amazon](#)
- [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 sudah memenuhi persyaratan teknis dengan sumber daya berbiaya paling rendah. Aktivitas penyesuaian ukuran yang tepat ini harus mempertimbangkan semua sumber daya beban kerja, semua atribut setiap sumber daya, dan upaya yang digunakan dalam operasi penyesuaian ukuran. Penyesuaian ukuran bisa menjadi proses berulang-ulang, yang diinisiasi 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 yang didapatkan selama masa guna beban kerja.

Di AWS, ada beberapa pendekatan yang berbeda:

Praktik terbaik

- [COST06-BP01 Lakukan pemodelan biaya](#)
- [COST06-BP02 Pilih jenis sumber daya, ukuran, dan nomor berdasarkan data](#)
- [COST06-BP03 Pilih tipe, ukuran, dan jumlah sumber daya secara otomatis berdasarkan metrik](#)
- [COST06-BP04 Pertimbangkan penggunaan sumber daya bersama](#)

COST06-BP01 Lakukan pemodelan biaya

Identifikasi kebutuhan organisasi (seperti kebutuhan bisnis dan komitmen yang ada) dan lakukan pemodelan biaya (keseluruhan biaya) untuk 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 diterapkan: 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 AWS Well-Architected](#).

Sebagai contoh, untuk membuat pemodelan biaya untuk beban kerja yang terdiri dari sumber daya komputasi, [AWS Compute Optimizer](#) dapat membantu pemodelan biaya untuk menjalankan beban kerja. Layanan ini menyediakan rekomendasi penyesuaian ukuran untuk sumber daya komputasi berdasarkan riwayat penggunaan. Pastikan CloudWatch Agen disebarkan ke EC2 instans Amazon untuk mengumpulkan metrik memori yang membantu Anda dengan rekomendasi yang lebih akurat di dalamnya. 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.

[Ada beberapa layanan yang dapat Anda gunakan dengan log kustom sebagai sumber data untuk operasi rightsizing untuk layanan lain dan komponen beban kerja, seperti, AWS Trusted Advisor Amazon dan CloudWatch Amazon Logs. CloudWatch](#) AWS Trusted Advisor memeriksa sumber daya dan menandai sumber daya dengan pemanfaatan rendah yang dapat membantu Anda mengukur sumber daya dengan tepat 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 AWS layanan yang tepat untuk beban kerja yang Anda rencanakan dengan mempertimbangkan komitmen Anda yang ada, model penetapan harga yang dipilih untuk beban kerja lainnya, dan kemampuan untuk berinovasi lebih cepat dan fokus pada nilai bisnis inti Anda.

Langkah-langkah implementasi

- Lakukan pemodelan biaya untuk sumber daya: Deploy beban kerja atau bukti konsep ke dalam sebuah akun terpisah dengan jenis 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 ke Ukuran yang Tepat](#)
- [CloudWatch Fitur Amazon](#)
- [Optimalisasi Biaya: Ukuran EC2 Tepat Amazon](#)
- [AWS Compute Optimizer](#)
- [AWS Kalkulator Harga](#)

Contoh terkait:

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

COST06-BP02 Pilih jenis sumber daya, ukuran, dan nomor 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 diterapkan: Sedang

Panduan implementasi

Amazon EC2 menyediakan berbagai pilihan jenis instans dengan berbagai tingkat CPU, memori, penyimpanan, dan kapasitas jaringan agar sesuai dengan kasus penggunaan yang berbeda. Jenis instans ini menampilkan perpaduan yang berbeda dari CPU, memori, penyimpanan, dan kemampuan jaringan, memberi Anda keserbagunaan 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 CPU core
- GPU inti

- Jumlah memori sistem (RAM)
- Jenis dan ruang penyimpanan
- Persyaratan bandwidth jaringan

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

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

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

Pengumpulan persyaratan sistem menjadi langkah yang 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 memiliki ukuran 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 AWS harga atau alat manajemen biaya dapat membantu dalam membuat keputusan berdasarkan informasi tentang jenis, ukuran, dan konfigurasi instans.

Langkah-langkah implementasi

- Pilih sumber daya berdasarkan data: Gunakan data pemodelan biaya Anda untuk memilih tingkat penggunaan beban kerja yang diantisipasi, dan pilih jenis dan ukuran sumber daya yang tertentu. Mengandalkan data pemodelan biaya, menentukan jumlah virtualCPUs, total memori (GiB), volume penyimpanan instans lokal (GB), volume EBS Amazon, dan tingkat kinerja jaringan, dengan

mempertimbangkan kecepatan transfer data yang diperlukan untuk instance tersebut. Selalu buat pilihan berdasarkan analisis mendetail dan data yang akurat untuk mengoptimalkan performa sekaligus mengelola biaya secara efektif.

Sumber daya

Dokumen terkait:

- [AWS Tipe instans](#)
- [AWS Auto Scaling](#)
- [CloudWatch Fitur Amazon](#)
- [Optimalisasi Biaya: Ukuran EC2 yang Tepat](#)

Video terkait:

- [Memilih EC2 instans Amazon yang tepat untuk beban kerja Anda](#)
- [Sesuaikan layanan Anda ke ukuran yang tepat](#)

Contoh terkait:

- [Semakin mudah untuk menemukan dan membandingkan jenis EC2 instans Amazon](#)

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 jenis yang tepat untuk mengoptimalkan biaya. Sediakan throughput, ukuran, dan penyimpanan secara tepat untuk layanan komputasi, penyimpanan, data, dan jaringan. Hal 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 diterapkan: 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 yang tepat untuk Anda. AWS juga menyediakan [API](#), [SDK](#), dan fitur-fitur yang memungkinkan sumber daya dapat dimodifikasi dengan sedikit usaha. 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 jenis atau ukuran otomatis di dalamnya, seperti [Amazon Simple Storage Service Intelligent-Tiering](#). Amazon S3 Intelligent-Tiering secara otomatis memindahkan data Anda antara dua tingkat akses, akses sering dan akses jarang, berdasarkan pola penggunaan Anda.

Langkah-langkah implementasi

- Tingkatkan observabilitas Anda dengan mengonfigurasi metrik beban kerja: Rekam metrik-metrik kunci untuk beban kerja. Metrik-metrik tersebut menyediakan indikasi pengalaman pelanggan, seperti output beban kerja, dan penyelarasan dengan perbedaan antartipe dan ukuran sumber daya, seperti penggunaan CPU dan memori. Untuk sumber daya komputasi, analisis data kinerja untuk menyesuaikan instans Amazon EC2 Anda ke ukuran yang tepat. Identifikasi instans tidak aktif dan instans dengan pemanfaatan terlalu rendah. Metrik utama yang harus diamati adalah penggunaan CPU dan pemanfaatan memori (misalnya, 40% pemanfaatan CPU di 90% dari keseluruhan waktu seperti yang dijelaskan dalam [Penyesuaian ukuran yang tepat dengan Mengaktifkan AWS Compute Optimizer dan Pemanfaatan Memori](#)). 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 di tingkat bucket, dan 14 hari data historis 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 yang tepat: Gunakan rekomendasi penyesuaian ukuran yang tepat di AWS Compute Optimizer dan alat pengatur ukuran Amazon EC2 di konsol Manajemen Biaya, atau tinjau penyesuaian ukuran AWS Trusted Advisor sumber daya Anda ke ukuran yang tepat untuk menyesuaikan beban kerja Anda. Penting untuk menggunakan [alat yang tepat](#) saat menyesuaikan sumber daya yang berbeda ke ukuran yang tepat dan mengikuti [pedoman penyesuaian ukuran yang tepat](#) apakah itu instans Amazon EC2, kelas penyimpanan AWS, atau jenis-jenis 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. Dengan menggunakan rekomendasi kontekstual yang diperoleh [Lensa Penyimpanan Amazon S3](#) dari analisis metrik di seluruh organisasi Anda, Anda dapat segera mengambil langkah untuk mengoptimalkan penyimpanan.

- Pilih jenis dan ukuran sumber daya secara otomatis berdasarkan metrik: Dengan 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 menyesuaikan armada Instans Sesuai Permintaan dan Instans Spot dalam satu grup Auto Scaling. Selain menerima diskon untuk menggunakan Spot Instance, Anda dapat menggunakan Instans Pesanan atau Savings Plan untuk menerima tarif diskon dengan harga biasa Instans Sesuai Permintaan. Semua kombinasi dari faktor-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 jenis instans berbasis atribut \(ABS\)](#) di [grup Auto Scaling \(ASG\)](#), yang akan membuat Anda dapat menyatakan persyaratan instans 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. Amazon EC2 Fleet dan Amazon EC2 Auto Scaling memilih dan meluncurkan instans yang cocok dengan atribut yang sudah ditentukan, sehingga menghilangkan kebutuhan untuk memilih jenis instans secara manual. Untuk sumber daya penyimpanan, Anda dapat menggunakan fitur [Amazon S3 Intelligent Tiering](#) dan fitur [Akses Jarang Amazon EFS](#), yang akan membuat Anda dapat memilih kelas penyimpanan secara otomatis yang memberikan penghematan biaya penyimpanan otomatis saat pola akses data berubah, tanpa memberikan dampak kinerja atau overhead operasional.

Sumber daya

Dokumen terkait:

- [AWS Auto Scaling](#)
- [Penyesuaian Ukuran yang Tepat untuk AWS](#)
- [AWS Compute Optimizer](#)
- [Fitur Amazon CloudWatch](#)
- [Menyiapkan CloudWatch](#)

- [CloudWatch Memublikasikan Metrik Kustom](#)
- [Memulai dengan Amazon EC2 Auto Scaling](#)
- [Lensa Penyimpanan Amazon S3](#)
- [Amazon S3 Intelligent-Tiering](#)
- [Akses Jarang Amazon EFS](#)
- [Meluncurkan Instans Amazon EC2 dengan Menggunakan SDK](#)

Video terkait:

- [Sesuaikan Layanan Anda ke Ukuran yang Tepat](#)

Contoh terkait:

- [Pemilihan Tipe Instans Berbasis Atribut untuk Penskalaan Otomatis untuk Amazon EC2 Fleet](#)
- [Mengoptimalkan biaya Amazon Elastic Container Service dengan menggunakan penskalaan terjadwal](#)
- [Penskalaan prediktif dengan Amazon EC2 Auto Scaling](#)
- [Mengoptimalkan Biaya dan Mendapatkan Visibilitas ke dalam Penggunaan dengan Lensa Penyimpanan Amazon S3](#)

COST06-BP04 Pertimbangkan penggunaan sumber daya bersama

Untuk layanan yang sudah di-deploy di tingkat organisasi untuk beberapa unit bisnis, pertimbangkan penggunaan sumber daya bersama untuk meningkatkan pemanfaatan dan mengurangi total biaya kepemilikan (TCO). Penggunaan sumber daya bersama dapat menjadi pilihan yang hemat biaya untuk memusatkan manajemen dan biaya dengan menggunakan solusi yang sudah ada, berbagi komponen, atau keduanya. Kelola fungsi-fungsi umum seperti pemantauan, pencadangan, dan konektivitas, baik dalam batasan akun maupun di sebuah akun khusus. Anda juga dapat mengurangi biaya dengan menerapkan standarisasi, mengurangi duplikasi, dan mengurangi kompleksitas.

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

Panduan implementasi

Apabila beberapa beban kerja menyebabkan fungsi yang sama, gunakan solusi yang sudah ada serta komponen bersama untuk memperbaiki manajemen dan mengoptimalkan biaya.

Pertimbangkan penggunaan sumber daya yang sudah ada (terutama sumber daya bersama), seperti server basis data nonproduksi atau layanan direktori, untuk mengurangi biaya cloud dengan mengikuti praktik terbaik keamanan dan peraturan organisasi. Untuk realisasi dan efisiensi nilai yang optimal, sangat penting untuk mengalokasikan biaya kembali (menggunakan showback dan chargeback) ke area terkait bisnis yang mendorong konsumsi.

Showback mengacu pada laporan yang memecah biaya cloud ke dalam kategori yang dapat diatribusikan, seperti konsumen, unit bisnis, akun buku besar, atau entitas yang bertanggung jawab lainnya. Tujuan showback adalah untuk menunjukkan kepada tim, unit bisnis, atau individu, biaya sumber daya cloud yang mereka gunakan.

Chargeback berarti mengalokasikan pengeluaran layanan pusat ke unit biaya berdasarkan strategi yang sesuai untuk proses manajemen keuangan tertentu. Untuk pelanggan, chargeback membebankan biaya yang dikeluarkan dari satu akun layanan bersama ke kategori biaya keuangan berbeda yang sesuai untuk suatu proses pelaporan pelanggan. Dengan menetapkan mekanisme chargeback, Anda dapat melaporkan biaya yang dikeluarkan oleh unit bisnis, produk, dan tim yang berbeda.

Beban kerja dapat dikategorikan sebagai kritis dan nonkritis. Berdasarkan klasifikasi ini, gunakan sumber daya bersama dengan konfigurasi umum untuk beban kerja yang kurang kritis. Agar lebih mengoptimalkan biaya, cadangkan server khusus semata-mata untuk beban kerja kritis. Bagikan sumber daya atau sediakan di beberapa akun agar dapat dikelola secara efisien. Bahkan dengan lingkungan pengembangan, pengujian, dan produksi yang berbeda, berbagi dapat dilakukan secara aman dan tidak membahayakan struktur organisasi.

Untuk meningkatkan pemahaman Anda dan mengoptimalkan biaya serta penggunaan untuk aplikasi dalam kontainer, gunakan data alokasi biaya terpisah yang membantu Anda mengalokasikan biaya ke entitas bisnis individual berdasarkan cara aplikasi memakai sumber daya komputasi dan memori bersama. Data alokasi biaya terbagi akan membantu Anda mencapai showback dan chargeback tingkat tugas dalam beban kerja kontainer yang berjalan di Amazon Elastic Container Service (Amazon ECS) atau Amazon Elastic Kubernetes Service (Amazon EKS).

Untuk arsitektur yang terdistribusi, buat VPC layanan bersama, yang menyediakan akses tersentralisasi ke layanan bersama yang diperlukan oleh beban kerja di setiap VPC. Layanan bersama ini dapat mencakup sumber daya seperti layanan direktori atau titik akhir VPC. Untuk mengurangi overhead dan biaya administrasi, bagikan sumber daya dari lokasi pusat, bukan membangunnya di setiap VPC.

Saat menggunakan sumber daya bersama, Anda dapat menghemat biaya operasional, memaksimalkan pemanfaatan sumber daya, dan meningkatkan konsistensi. Dalam desain multiakun, Anda dapat meng-host beberapa layanan AWS secara terpusat dan mengaksesnya dengan beberapa aplikasi dan akun di sebuah hub untuk menghemat biaya. Anda dapat menggunakan [AWS Resource Access Manager \(AWS RAM\)](#) untuk berbagi sumber daya umum lainnya, seperti [subnet VPC dan lampiran AWS Transit Gateway](#), [AWS Network Firewall](#), atau [Amazon SageMaker AI pipelines](#). Dalam sebuah lingkungan multi-akun, AWS RAM akan memungkinkan Anda untuk membuat sumber daya satu kali dan membagikannya kepada akun lain.

Organisasi harus memberikan tag pada biaya bersama secara efektif dan memverifikasi bahwa hanya sedikit biaya mereka yang tidak diberi tag atau dialokasikan. Jika Anda tidak mengalokasikan biaya bersama secara efektif dan tidak ada orang yang bertanggung jawab atas pengelolaan biaya bersama, biaya cloud bersama dapat meningkat. Anda harus tahu di mana Anda telah mengeluarkan biaya pada tingkat sumber daya, beban kerja, tim, atau organisasi, karena pengetahuan ini meningkatkan pemahaman Anda tentang nilai yang dihadirkan pada tingkat yang berlaku saat dibandingkan dengan hasil bisnis yang dicapai. Pada akhirnya, organisasi mendapat manfaat berupa penghematan biaya sebagai hasil dari berbagi infrastruktur cloud. Dorong alokasi biaya pada sumber daya cloud bersama untuk mengoptimalkan pengeluaran cloud.

Langkah-langkah implementasi

- Evaluasi sumber daya yang ada: Tinjau beban kerja yang ada yang menggunakan layanan serupa untuk beban kerja Anda. Tergantung komponen beban kerja, pertimbangkan platform yang ada jika logika bisnis atau persyaratan teknis memungkinkan.
- Gunakan berbagi sumber daya di AWS RAM dan batasi sesuai kebutuhan: Gunakan AWS RAM untuk berbagi sumber daya dengan akun AWS lain yang ada dalam organisasi Anda. Saat berbagi sumber daya, Anda tidak perlu menduplikasi sumber daya di beberapa akun, sehingga meminimalkan beban operasional pemeliharaan sumber daya. Proses ini juga membantu Anda secara aman membagikan sumber daya yang Anda buat kepada peran dan pengguna di akun Anda, serta kepada Akun AWS lainnya.
- Sumber daya tag: Berikan tag pada sumber daya yang merupakan kandidat untuk pelaporan biaya dan kategorikan sumber daya tersebut ke dalam kategori biaya. Aktifkan tag sumber daya terkait biaya tersebut untuk alokasi biaya agar dapat memberikan visibilitas penggunaan sumber daya AWS. Fokuslah menciptakan tingkat detail yang sesuai sehubungan dengan visibilitas biaya dan penggunaan, dan pengaruhi perilaku konsumsi cloud melalui pelaporan alokasi biaya dan pelacakan KPI.

Sumber daya

Praktik-praktik terbaik terkait:

- [SEC03-BP08 Membagikan sumber daya secara aman dalam organisasi Anda](#)

Dokumen terkait:

- [Apa itu AWS Resource Access Manager?](#)
- Layanan [AWS yang dapat Anda gunakan dengan AWS Organizations](#)
- [Sumber daya AWS yang dapat dibagikan](#)
- [Kueri AWS Cost and Usage Report \(CUR\)](#)

Video terkait:

- [AWS Resource Access Manager - Kontrol akses informasi terperinci dengan izin terkelola](#)
- [Bagaimana merancang strategi alokasi biaya AWS Anda](#)
- [AWS Cost Categories](#)

Contoh terkait:

- [Cara melakukan chargeback layanan bersama: Sebuah contoh AWS Transit Gateway](#)
- [Cara membuat model chargeback/showback untuk Savings Plans menggunakan CUR](#)
- [Menggunakan Berbagi VPC untuk Arsitektur Layanan Mikro Multi-Akun yang Hemat Biaya](#)
- [Tingkatkan visibilitas biaya Amazon EKS dengan Data Alokasi Biaya Terpisah AWS](#)
- [Tingkatkan visibilitas biaya Amazon ECS dan AWS Batch dengan Data Alokasi Biaya Terpisah AWS](#)

Memilih model penetapan harga terbaik

Lakukan pemodelan biaya beban kerja: Pertimbangkan persyaratan komponen beban kerja dan pahami model harga potensial. Tetapkan persyaratan ketersediaan komponen. 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.

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

Gunakan alat rekomendasi [AWS Cost Explorer](#) untuk menemukan peluang-peluang diskon komitmen.

Untuk menemukan peluang untuk beban kerja Spot, gunakan tampilan penggunaan secara keseluruhan per jam, dan cari periode penggunaan atau elastisitas yang berubah secara teratur.

Model Penetapan Harga: AWS memiliki banyak [model penetapan harga](#) yang memungkinkan Anda membayar sumber daya Anda dengan cara yang paling hemat biaya, yang sesuai dengan kebutuhan organisasi Anda. Bagian berikut ini akan menjelaskan masing-masing model pembelian:

- Instans Sesuai Permintaan
- Instans Spot
- Diskon komitmen - Savings Plans
- Diskon komitmen - Instans/Kapasitas Terpesan
- Pilihan geografis
- Perjanjian dan penetapan harga ketiga pihak ketiga

Instans Sesuai Permintaan: Ini adalah model penetapan harga default, yakni model penetapan harga bayar sesuai pemakaian. Saat Anda menggunakan sumber daya (misalnya, instans EC2 atau layanan seperti DynamoDB Sesuai Permintaan), Anda membayar tarif tetap, dan Anda tidak memiliki komitmen jangka panjang. Anda dapat menambah atau mengurangi kapasitas sumber daya atau layanan-layanan Anda berdasarkan permintaan aplikasi Anda. Instans Sesuai Permintaan memiliki tarif per jam, tetapi tergantung pada layanannya, dapat ditagihkan dengan penambahan satu detik (misalnya Amazon RDS, atau instans Linux EC2). Instans Sesuai Permintaan direkomendasikan untuk aplikasi dengan beban kerja jangka pendek (misalnya, proyek empat bulan), yang mengalami

lonjakan secara periodik, atau beban kerja yang tak terprediksi yang tidak boleh terinterupsi. Instans Sesuai Permintaan juga cocok untuk beban kerja, seperti lingkungan pra-produksi, yang memerlukan runtime tanpa interupsi, tetapi tidak berjalan cukup lama untuk masuk ke diskon komitmen (Savings Plans atau Instans Terpesan).

Instans Spot: Sebuah [Instans Spot](#) adalah kapasitas komputasi Amazon EC2 cadangan yang tersedia dengan harga diskon hingga 90% dari harga Instans Sesuai Permintaan tanpa ada kewajiban komitmen jangka panjang. Dengan Instans Spot, Anda dapat secara signifikan mengurangi biaya untuk menjalankan aplikasi atau memperbesar skala kapasitas komputasi aplikasi Anda dengan anggaran biaya yang sama. Tidak seperti Instans Sesuai Permintaan, Instans Spot boleh diinterupsi dengan peringatan 2-menit jikalau Amazon EC2 memerlukan kapasitasnya kembali, atau jika harga Instans Spot melampaui harga yang telah diatur sebelumnya. Rata-rata, Instans Spot mengalami gangguan kurang dari 5% dari seluruh waktu.

Instans Spot adalah instans yang paling ideal ketika ada antrean atau terjadi buffer, atau di mana ada beberapa sumber daya yang bekerja secara independen untuk memproses permintaan (misalnya, pemrosesan data Hadoop). Biasanya beban kerja ini toleran terhadap kesalahan, tidak memandang geografis, dan fleksibel, seperti pemrosesan batch, big data dan analitik, lingkungan kontainer unit mandiri, dan komputasi performa tinggi (HPC). Beban kerja non-kritis seperti lingkungan pengujian dan pengembangan juga merupakan kandidat beban kerja untuk Instans Spot.

Instans Spot juga terintegrasi ke dalam beragam layanan AWS, seperti grup Amazon EC2 Auto Scaling, Amazon EMR, Amazon Elastic Container Service (Amazon ECS), dan AWS Batch.

Saat sebuah Instans Spot perlu diklaim ulang, Amazon EC2 mengirimkan sebuah peringatan dua menit melalui pemberitahuan interupsi Instans Spot yang dikirimkan melalui CloudWatch Events, serta di metadata instans. Selama periode dua menit tersebut, aplikasi Anda dapat memanfaatkan waktu yang tersedia itu untuk menyimpan statusnya, mengosongkan kontainer yang sedang berjalan, mengunggah file log akhir, atau menghapus dirinya sendiri dari sebuah penyeimbang beban. Pada akhir dua menit, Anda memiliki opsi untuk hibernasi, menghentikan, atau menghentikan Instans Spot.

Pertimbangkan praktik terbaik berikut saat Anda mengadopsi Instans Spot di beban kerja Anda:

- Bersikaplah fleksibel di sebanyak mungkin jenis instans: Bersikaplah fleksibel di kedua famili dan ukuran jenis instans, untuk meningkatkan kemungkinan pemenuhan persyaratan kapasitas target Anda, untuk mendapatkan biaya serendah mungkin, dan minimalisir dampak interupsi.
- Bersikaplah fleksibel tentang di mana beban kerja Anda akan berjalan: Kapasitas yang tersedia dapat bervariasi sesuai Zona Ketersediaan. Hal ini akan meningkatkan kecenderungan

pemenuhan kapasitas target Anda dengan memanfaatkan beberapa kumpulan kapasitas cadangan, dan menyediakan biaya yang serendah mungkin.

- Desain untuk keberlanjutan: Rancanglah desain beban kerja Anda untuk kebebasan batas geografis dan toleransi kesalahan, sehingga jika ada beberapa kapasitas EC2 Anda yang mengalami interupsi, hal itu tidak akan berdampak pada ketersediaan atau kinerja beban kerja.
- Kami merekomendasikan Anda untuk menggunakan Instans Spot yang dikombinasikan dengan Instans Sesuai Permintaan dan Savings Plans/Instans Terpesan untuk memaksimalkan optimalisasi biaya beban kerja dengan kinerja.

Diskon komitmen – Savings Plans: AWS menyediakan sejumlah cara bagi Anda untuk mengurangi biaya dengan memesan atau membeli komitmen untuk menggunakan sejumlah sumber daya tertentu, dan menerima tarif diskon untuk sumber daya Anda. Paket [Savings Plans](#) memungkinkan Anda membuat komitmen pengeluaran per jam selama satu atau tiga tahun, dan menerima diskon harga di seluruh sumber daya Anda. Paket Savings Plans menyediakan diskon untuk layanan Komputasi AWS seperti Amazon EC2, AWS Fargate, dan AWS Lambda. Ketika Anda membeli sebuah komitmen, Anda membayar jumlah komitmen itu setiap jam, dan jumlah tersebut akan dikurangi dari penggunaan Sesuai Permintaan Anda dengan persenan diskon. Misalnya, Anda membeli komitmen sebesar 50 USD per jam, dan memiliki 150 USD per jam penggunaan Sesuai Permintaan. Dengan mempertimbangkan skema harga Savings Plans, penggunaan spesifik Anda memiliki persenan diskon sebesar 50%. Jadi, komitmen 50 USD Anda mencakup 100 USD dari penggunaan Sesuai Permintaan. Anda akan membayar 50 USD (komitmen) dan 50 USD dari sisa penggunaan Sesuai Permintaan.

Paket [Compute Savings Plans](#) adalah paket yang paling fleksibel dan bisa memberikan Anda diskon hingga 66%. Paket-paket harga ini berlaku secara otomatis di seluruh Zona Ketersediaan, seluruh ukuran instans, famili instans, sistem operasi, penghunian, Wilayah, dan layanan komputasi.

Paket [Savings Plans Instans](#) memiliki fleksibilitas yang lebih sedikit tetapi memberikan persenan diskon yang lebih tinggi (hingga 72%). Paket-paket harga ini secara otomatis berlaku di seluruh Zona Ketersediaan, seluruh ukuran instans, sistem operasi, dan penghunian.

Ada tiga opsi pembayaran, yaitu:

- Tanpa pembayaran di muka: Tanpa pembayaran di muka; Anda hanya harus membayar tarif per jam yang berkurang setiap bulan untuk total jam penggunaan dalam sebulan.

- Pembayaran di muka sebagian: Memberikan besaran diskon yang lebih tinggi daripada opsi tanpa pembayaran di muka. Sebagian penggunaan dibayar di muka: kemudian Anda hanya harus membayar tarif per jam yang berkurang setiap bulan untuk total jam penggunaan dalam sebulan.
- Pelunasan di muka: Penggunaan untuk seluruh periode dibayar di muka, dan tidak ada biaya lain yang dikeluarkan untuk sisa jangka waktu penggunaan yang ditanggung oleh paket komitmen ini.

Anda dapat menerapkan kombinasi apa pun dari ketiga opsi pembelian ini di seluruh beban kerja Anda.

Paket Savings Plans berlaku pertama untuk penggunaan di akun yang digunakan dalam pembelian paket, dari persentase diskon paling tinggi hingga paling rendah, kemudian paket tersebut akan berlaku untuk penggunaan gabungan di semua akun lain, dari persentase diskon paling tinggi hingga paling rendah.

Sebaiknya Anda membeli semua paket Savings Plans untuk satu akun dengan tanpa penggunaan atau pun sumber daya, seperti akun manajemen. Hal ini akan memastikan paket Savings Plans berlaku untuk persentase diskon paling tinggi di semua penggunaan Anda, yang akan memaksimalkan jumlah diskon.

Beban kerja dan penggunaan biasanya akan berubah dari waktu ke waktu. Anda sebaiknya melakukan pembelian sejumlah kecil komitmen Savings Plans dari waktu ke waktu secara berkelanjutan. Hal ini dapat memastikan bahwa Anda akan mempertahankan cakupan tingkat tinggi untuk memaksimalkan diskon Anda, dan paket yang Anda beli akan sangat sesuai dengan beban kerja dan persyaratan organisasi Anda setiap saat.

Jangan tetapkan cakupan target di akun Anda, karena adanya bermacam-macam diskon yang mungkin akan berlaku. Cakupan yang rendah tidak selalu menunjukkan potensi adanya penghematan yang besar. Anda mungkin memiliki cakupan yang rendah di akun Anda, tetapi jika penggunaan Anda terdiri dari instans-instans kecil, dengan sistem operasi yang sudah mempunyai lisensi, maka potensi penghematan yang Anda dapatkan bisa jadi sangat rendah, hanya beberapa persen saja. Sebagai gantinya, lakukan pelacakan dan pemantauan atas potensi penghematan yang tersedia di alat rekomendasi Savings Plans. Sering-seringlah meninjau rekomendasi Savings Plans di Cost Explorer (untuk melakukan analisis rutin) dan lanjutkan membeli komitmen sampai penghematan yang diperkirakan berada di bawah diskon yang diperlukan untuk organisasi. Misalnya, lacak dan pantau apakah diskon potensial Anda berada tetap di bawah 20%, jika lebih dari itu, maka Anda harus melakukan pembelian.

Lakukan pantauan pemanfaatan dan cakupan, tetapi hanya untuk mendeteksi perubahan. Jangan menargetkan persentase penggunaan atau cakupan tertentu, karena hal ini tidak selalu berbanding lurus dengan penghematan. Pastikan bahwa pembelian Savings Plans yang Anda lakukan menghasilkan peningkatan cakupan, dan jika terjadi penurunan cakupan atau pemanfaatan, maka Anda harus memastikan bahwa penurunan ini terukur dan diketahui. Misalnya, Anda melakukan migrasi terhadap sumber daya beban kerja ke jenis instans yang lebih baru, yang mengurangi pemanfaatan paket yang sudah ada, tetapi manfaat kinerja melebihi pengurangan penghematan.

Diskon komitmen – Instans/Komitmen Cadangan: Mirip dengan Savings Plans, [Instans Terpesan](#) (RI) yang menawarkan diskon hingga 72% untuk sebuah paket komitmen untuk menjalankan sumber daya dengan jumlah minimum. Instans Terpesan tersedia untuk Amazon RDS, Amazon OpenSearch Service, Amazon ElastiCache, Amazon Redshift, dan DynamoDB. Amazon CloudFront dan AWS Elemental MediaConvert juga memberikan diskon saat Anda mencapai komitmen penggunaan minimum Anda. Instans Terpesan saat ini tersedia untuk Amazon EC2, namun Savings Plans juga menawarkan tingkat diskon yang sama dengan peningkatan fleksibilitas dan tidak ada biaya overhead manajemen.

Instans Terpesan menawarkan opsi harga yang sama tanpa pembayaran di muka, pembayaran sebagian di muka, dan pelunasan di muka, dan persyaratan yang sama untuk satu atau tiga tahun.

Instans Terpesan dapat dibeli di sebuah Wilayah atau Zona Ketersediaan tertentu. Mereka menyediakan reservasi kapasitas saat Instans Terpesan dibeli di sebuah Zona Ketersediaan.

Amazon EC2 memiliki fitur RI yang dapat dikonversi, namun paket Savings Plans harus diterapkan ke semua instans EC2 untuk peningkatan fleksibilitas dan pengurangan biaya operasional.

Proses dan metrik yang sama harus digunakan untuk melakukan pelacakan dan pembelian Instans Terpesan. Anda sebaiknya tidak melacak cakupan RI di seluruh akun Anda. Anda juga sebaiknya tidak melakukan pelacakan dan pemantauan persentase pemanfaatan, sebagai gantinya, Anda dapat melihat laporan pemanfaatan di Cost Explorer, dan menggunakan kolom penghematan bersih yang ada pada tabel. Jika penghematan bersih bernilai angka negatif yang sangat besar, maka Anda harus mengambil tindakan untuk melakukan pemulihan RI yang tidak digunakan.

Armada EC2: [Armada EC2](#) adalah sebuah fitur yang memungkinkan Anda menentukan kapasitas komputasi target, lalu menentukan jenis instans dan saldo Instans Sesuai Permintaan dan Instans Spot untuk armada tersebut. Armada EC2 akan secara otomatis meluncurkan kombinasi harga paling rendah dari banyak sumber daya untuk memenuhi kapasitas yang telah ditentukan.

Pemilihan geografis: Saat Anda merancang arsitektur solusi Anda, praktik terbaik yang dapat diterapkan adalah berusaha menempatkan sumber daya komputasi lebih dekat dengan para

pengguna untuk memberikan latensi yang lebih rendah dan kedaulatan data yang kuat. Untuk audiens global, Anda harus menggunakan beberapa lokasi untuk memenuhi kebutuhan-kebutuhan ini. Anda harus memilih lokasi geografis yang dapat meminimalkan biaya Anda.

Infrastruktur AWS Cloud dibangun di sekitar [Wilayah dan Zona Ketersediaan](#). Sebuah Wilayah adalah lokasi fisik di dunia yang terdiri dari beberapa Zona Ketersediaan. Zona Ketersediaan (AZ) terdiri dari satu atau beberapa pusat data terpisah, yang masing-masing pusat data itu memiliki daya, jaringan, dan konektivitas redundan yang ditempatkan di fasilitas-fasilitas terpisah juga.

Setiap Wilayah AWS dioperasikan dalam kondisi pasar setempat, dan harga sumber daya di setiap Wilayah tentunya berbeda. Pilihlah sebuah Wilayah tertentu untuk mengoperasikan komponen atau seluruh solusi Anda sehingga Anda dapat menjalankannya dengan harga serendah mungkin secara global. Anda dapat menggunakan Kalkulator Bulanan Sederhana AWS untuk membuat estimasi biaya beban kerja Anda di berbagai Wilayah.

Perjanjian dan penentuan harga pihak ketiga: Ketika Anda menggunakan solusi atau layanan pihak ketiga di cloud, Anda harus memastikan bahwa struktur harga selaras dengan hasil-hasil Optimasi Biaya yang Anda dapatkan. Skema harga harus diskalakan sesuai dengan hasil dan nilai yang disediakan oleh solusi atau layanan tersebut. Contoh dari hal ini adalah perangkat lunak yang mengambil persentase dari penghematan yang disediakan, makin banyak Anda penghematannya (hasil), makin banyak juga biayanya. Perjanjian yang diskalakan sesuai dengan tagihan Anda biasanya tidak selaras dengan Optimasi Biaya, kecuali jika perjanjian tersebut menyediakan hasil untuk setiap bagian tagihan spesifik Anda. Misalnya, solusi yang menyediakan rekomendasi untuk Amazon EC2 dan mengenakan biaya persentase dari seluruh tagihan Anda akan meningkat jika Anda menggunakan layanan-layanan lain yang tidak menerima manfaat darinya. Contoh lainnya adalah sebuah layanan terkelola yang dikenakan biaya dengan persentase dari biaya sumber daya yang dikelola. Sebuah instans yang memiliki ukuran lebih besar tidak selalu memerlukan lebih banyak upaya manajemen, tetapi akan dikenakan lebih banyak biaya. Pastikan bahwa perjanjian harga layanan ini mencakup program optimalisasi biaya atau fitur-fitur yang ada pada layanan mereka untuk mendorong efisiensi.

Praktik terbaik

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

COST07-BP01 Lakukan 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 yang tinggi.

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 tergantung 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 untuk kapasitas komputasi atau basis data sesuai jam atau detik (minimum 60 detik) tergantung pada instans mana yang Anda jalankan, tanpa perlu ada komitmen jangka panjang atau pembayaran di muka.

Savings Plans adalah model penetapan harga fleksibel yang menawarkan harga rendah di AmazonEC2, Lambda, dan AWS Fargate penggunaan, dengan imbalan komitmen terhadap jumlah penggunaan yang konsisten (diukur dalam dolar per jam) selama satu tahun atau tiga tahun.

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

Instans Terpesan akan memungkinkan Anda mendapat diskon hingga 75 persen dengan membayar di muka atas kapasitas yang didapat. Untuk detail selengkapnya, lihat [Mengoptimalkan biaya dengan reservasi](#).

Anda mungkin memilih untuk menyertakan Savings Plans 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 EC2 biaya Amazon atau menggunakan [Compute Savings](#) Plans untuk mengurangi biaya AmazonEC2, Fargate,

dan Lambda. Alat rekomendasi [AWS Cost Explorer](#) memberikan peluang untuk diskon komitmen dengan paket Savings Plans.

Jika Anda telah membeli [Instans Cadangan](#) untuk Amazon EC2 di masa lalu atau telah menetapkan praktik alokasi biaya di dalam organisasi Anda, Anda dapat terus menggunakan Instans EC2 Cadangan Amazon untuk saat 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) AWS Cost Management untuk menghasilkan Rekomendasi Savings Plans baru kapan saja. Gunakan Instans Cadangan (RI) untuk mengurangi biaya AmazonRDS, Amazon Redshift, Amazon, dan ElastiCache OpenSearch Amazon Service. Savings Plans dan Instans Terpesan tersedia dalam tiga opsi, yakni: semua pembayaran di muka, sebagian pembayaran di muka, dan tanpa pembayaran di muka. Gunakan rekomendasi yang diberikan dalam rekomendasi pembelian AWS Cost Explorer 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 Spot Instance untuk berbagai aplikasi yang toleran terhadap kesalahan dan fleksibel. Contohnya termasuk server web stateless, API endpoint, data besar dan aplikasi analitik, beban kerja kontainer, CI/CD, dan beban kerja fleksibel lainnya.

Analisis RDS instans Amazon EC2 dan Amazon Anda apakah mereka dapat dimatikan saat Anda tidak menggunakannya (setelah 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 kluster dan melanjutkannya nanti. Saat kluster Amazon Redshift atau Amazon dan RDS Instans EC2 Amazon dihentikan, tagihan komputasi terhenti dan hanya biaya penyimpanan yang berlaku.

Perhatikan bahwa [reservasi Kapasitas Sesuai Permintaan](#) (ODCR) bukanlah diskon harga. Reservasi Kapasitas ditagih dengan tarif Sesuai Permintaan yang setara, baik Anda menjalankan instans di 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 dapat dibatalkan saat Anda tidak lagi membutuhkannya, tetapi mereka juga bisa mendapatkan keuntungan dari diskon yang diberikan oleh Savings Plans atau Reserved Instances.

Langkah-langkah implementasi

- Analisis elastisitas beban kerja: Dengan menggunakan perincian per jam di Cost Explorer atau dasbor kustom, lakukan analisis terhadap elastisitas beban kerja Anda. Cari perubahan teratur dalam jumlah instans yang dijalankan. Instans berdurasi pendek merupakan kandidat untuk Instans Spot atau Armada Spot.
 - [Lab Well-Architected: Cost Explorer](#)
 - [Lab Well-Architected: Visualisasi Biaya](#)
- Tinjau kontrak harga yang ada saat ini: Tinjau kontrak atau komitmen 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](#) memberi pelanggan pilihan untuk menyesuaikan perjanjian yang paling sesuai dengan kebutuhan mereka. Untuk komitmen jangka panjang, pertimbangkan diskon harga cadangan, Instans Cadangan, atau Savings Plans untuk jenis instans tertentu, keluarga instans Wilayah AWS, dan Availability Zone.
- Lakukan analisis discount komitmen: Dengan 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 [lab Well-Architected](#).

Sumber daya

Dokumen terkait:

- [Mengakses Rekomendasi Instans Terpesan](#)
- [Opsi pembelian instans](#)
- [AWS Perusahaan](#)

Video terkait:

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

Contoh terkait:

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

COST07-BP02 Memilih Wilayah berdasarkan biaya

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 diterapkan: Sedang

Panduan implementasi

[Infrastruktur AWS Cloud](#) bersifat global, di-hosting di [beberapa lokasi di seluruh dunia](#), dan terdiri dari Wilayah AWS, Zona Ketersediaan, Local Zone, AWS Outposts, dan Wavelength Zones. 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 memiliki 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. Pilihlah sebuah Wilayah tertentu untuk mengoperasikan komponen atau seluruh solusi Anda 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 kewajiban memenuhi persyaratan bisnis, privasi data, dan keamanan. Misalnya, jika Wilayah default Anda adalah Asia Pasifik (Sydney) (ap-southwest-2), dan jika tidak ada batasan (privasi data, keamanan, misalnya) untuk menggunakan Wilayah lain, menerapkan instans Amazon EC2 yang tidak kritis (pengembangan dan pengujian) di AS Timur (Virginia Utara) (us-east-1) akan dikenakan biaya 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 |
| Wilayah 6 | ✓ | 15 milidetik | \$ | ✓ |
| <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 6 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-langkah implementasi

- Tinjau penetapan 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 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 pada 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, meningkatkan kinerja, dan meningkatkan keamanan.

Sumber daya

Dokumen terkait:

- [Mengakses Rekomendasi Instans Terpesan](#)
- [Harga Amazon EC2](#)
- [Opsi pembelian instans](#)
- [Tabel Wilayah](#)

Video terkait:

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

COST07-BP03 Pilih perjanjian pihak ketiga dengan persyaratan 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 diterapkan: 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. Skema harga harus diskalakan sesuai dengan hasil dan nilai yang disediakan oleh solusi atau layanan tersebut. 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 memberikan rekomendasi untuk Amazon EC2 dan membebankan persentase dari seluruh tagihan Anda dapat menjadi lebih 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-langkah implementasi

- Menganalisis perjanjian dan persyaratan 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 Instans Spot](#)

COST07-BP04 Menerapkan model penetapan 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 diterapkan: Rendah

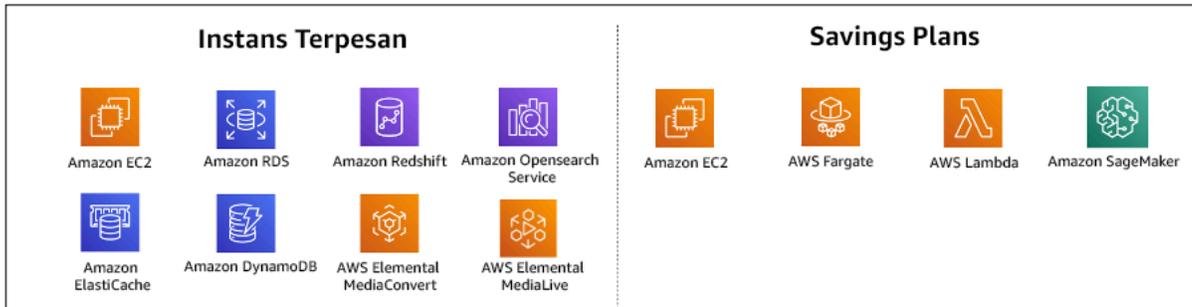
Panduan implementasi

Untuk meningkatkan efisiensi biaya, AWS berikan beberapa rekomendasi komitmen berdasarkan penggunaan Anda sebelumnya. Anda dapat menggunakan rekomendasi ini untuk memahami apa yang dapat Anda hemat, dan bagaimana penggunaan komitmen tersebut. Anda dapat menggunakan layanan ini sebagai On-Demand, Spot, atau membuat komitmen untuk jangka waktu tertentu dan mengurangi biaya sesuai permintaan Anda dengan Instans Cadangan () dan Savings Plans (RIs). SPs Anda perlu memahami tidak hanya setiap komponen beban kerja dan beberapa AWS layanan, tetapi juga diskon komitmen, opsi pembelian, dan Instans Spot untuk layanan ini guna 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. Contoh beban kerja ini terdiri dari beberapa AWS layanan, seperti Amazon Route 53, Amazon AWS WAF, instans Amazon CloudFront,

EC2 instans Amazon, Load RDS Balancer, 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 dari mereka mungkin memenuhi syarat untuk RIs atau SPs, sementara beberapa dari mereka mungkin hanya tersedia berdasarkan permintaan. Seperti yang ditunjukkan gambar berikut, beberapa AWS layanan dapat dilakukan menggunakan RIs atau SPs.



AWS layanan yang dilakukan menggunakan Instans Cadangan dan Savings Plans

Langkah-langkah implementasi

- Menerapkan model harga: Dengan menggunakan hasil analisis Anda, lakukan pembelian paket Savings Plans, Instans Terpesan, atau terapkan Instans Spot. Jika ini adalah pembelian komitmen pertama Anda, pilih lima atau sepuluh rekomendasi teratas dalam daftar, lalu pantau dan analisis hasilnya selama satu atau dua bulan ke depan. AWS Cost Management Console memandu Anda melalui proses. Tinjau rekomendasi RI atau SP dari konsol, sesuaikan rekomendasi (jenis, pembayaran, serta jangka waktu), dan tinjau komitmen per jam (misalnya 20 USD 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 EC2 instans Amazon sesuai permintaan dan alokasikan sumber daya untuk persyaratan yang tersisa.
- Siklus peninjauan beban kerja: Menerapkan siklus peninjauan untuk beban kerja yang secara khusus menganalisis cakupan model penetapan 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 AWS layanan lain](#)
- [Layanan yang Didukung Savings Plans](#)

Video terkait:

- [Hemat hingga 90% dan jalankan beban kerja produksi di Instans 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 diterapkan: 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 serta komponennya yang terus berubah.

Gunakan alat-alat rekomendasi [AWS Cost Explorer](#) untuk menemukan peluang-peluang mendapat 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. Savings Plans Instans
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 Terkonsolidasi, 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 ke penggunaan pertama dengan tingkat 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 AWS Cost Management 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, lakukan analisis model harga AWS dan selaraskan dengan persyaratan bisnis Anda untuk menemukan total biaya kepemilikan dan peluang [optimalisasi biaya](#).

Langkah-langkah implementasi

Lakukan analisis discount komitmen: Gunakan Cost Explorer di akun Anda, tinjau rekomendasi Savings Plans dan Instans Terpesan. Pastikan Anda memahami rekomendasi Savings Plans, dan perkirakan estimasi 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 penetapan harga AWS?](#)
- [Opsi pembelian instans](#)
- [Ikhtisar Savings Plan](#)
- [Rekomendasi Savings Plans](#)
- [Mengakses Rekomendasi Instans Terpesan](#)
- [Memahami rekomendasi Savings Plans Anda](#)
- [Bagaimana Savings Plans diterapkan ke penggunaan AWS Anda](#)
- [Savings Plans dengan Penagihan Terkonsolidasi](#)
- [Mengaktifkan diskon instans terpesan dan Savings Plans bersama](#)

Video terkait:

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

Contoh terkait:

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

- [Bagaimana cara menggunakan paket Savings Plans bergulir untuk mengurangi risiko komitmen?](#)
- [Kapan Harus Menggunakan Instans Spot](#)

Merencanakan transfer data

Salah satu keunggulan cloud adalah bahwa cloud merupakan layanan jaringan terkelola. Anda tidak perlu lagi 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 harus membayar sesuai dengan penggunaan Anda. Penggunaan sumber daya jaringan yang efisien harus dilakukan untuk optimasi biaya di cloud.

Praktik terbaik

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

COST08-BP01 Lakukan pemodelan transfer data

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

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 AWS ditentukan oleh sumber, tujuan, dan volume lalu lintas. Memperhitungkan biaya-biaya ini dalam fase desain dapat menghasilkan penghematan biaya. Memahami di mana transfer data terjadi dalam beban kerja Anda, biaya transfer, dan manfaat terkait sangat penting untuk secara akurat memperkirakan total biaya kepemilikan (TCO). 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.

Ini AWS Kalkulator Harga dapat membantu Anda melihat perkiraan biaya untuk AWS layanan tertentu dan transfer data yang diharapkan. Jika Anda memiliki beban kerja yang sudah berjalan (untuk tujuan pengujian atau dalam lingkungan pra-produksi), gunakan [AWS Cost Explorer](#) atau [AWS Cost and Usage Report](#) (CUR) untuk memahami dan memodelkan 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-langkah implementasi

- Identifikasi persyaratan: Apa tujuan utama dan persyaratan bisnis untuk transfer data terencana antara sumber dan tujuan? Apa hasil bisnis yang diharapkan di akhir nanti? Kumpulkan persyaratan bisnis dan tentukan hasil akhir yang diharapkan.
- Identifikasi sumber dan tujuan: Apa sumber data dan tujuan transfer data, seperti di dalam Wilayah AWS, ke AWS layanan, atau keluar ke internet?
 - [Transfer data dalam 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 AWS layanan atau alat yang akan digunakan: AWS Layanan apa yang digunakan untuk transfer data ini? Apakah memungkinkan untuk menggunakan layanan yang sudah tersedia untuk beban kerja yang lain?
- Hitung biaya transfer data: Gunakan [Penetapan Harga AWS](#) yang pemodelan transfer datanya sudah Anda buat sebelumnya untuk menghitung biaya transfer data untuk beban kerja tersebut. 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.

- Menghubungkan pengeluaran biaya dengan hasil yang diperoleh: Untuk setiap biaya transfer data yang dikenakan, tentukan hasil yang akan diperoleh dari transfer data tersebut untuk beban kerja tertentu. 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, buatlah pemodelan transfer data dasar konseptual untuk beberapa kasus penggunaan dan beban kerja yang berbeda.

Sumber daya

Dokumen terkait:

- [AWS solusi caching](#)
- [AWS Penetapan Harga](#)
- [EC2Harga Amazon](#)
- [VPCHarga Amazon](#)
- [Memahami biaya transfer data](#)

Video terkait:

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

Contoh terkait:

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

COST08-BP02 Pilih komponen untuk mengoptimalkan biaya transfer data

Semua komponen diseleksi, dan arsitektur didesain untuk mengurangi biaya transfer data. Ini termasuk menggunakan komponen seperti wide-area-network (WAN) optimasi dan konfigurasi Multi-Availability Zone (AZ)

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 WAN optimasi dan optimasi 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 AWS layanan yang tepat untuk 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 VPC titik akhir dalam AWS: VPC titik akhir memungkinkan koneksi pribadi antara layanan Anda VPC dan layanan yang didukung AWS . Hal ini memungkinkan Anda untuk menghindari penggunaan internet publik, yang dapat menimbulkan biaya transfer data.
- Gunakan NAT gateway: Gunakan [NATgateway](#) sehingga instance di subnet pribadi dapat terhubung ke internet atau ke layanan di luar Anda. VPC Periksa apakah sumber daya di belakang NAT gateway yang mengirim lalu lintas terbanyak berada di Availability Zone yang sama dengan NAT gateway. Jika tidak, buat NAT gateway baru di Availability Zone yang sama dengan sumber daya untuk mengurangi biaya transfer data lintas-AZ.
- Gunakan AWS Direct Connect AWS Direct Connect bypass internet publik dan membuat koneksi pribadi langsung antara jaringan lokal Anda dan AWS Hal ini bisa lebih hemat biaya dan konsisten daripada mentransfer data dalam jumlah besar melalui internet.
- Hindari mentransfer data melintasi batas Regional: Transfer data antara Wilayah AWS (dari satu Wilayah ke Wilayah lain) biasanya dikenakan biaya. Sebaiknya putus dengan cermat jika ingin memilih jalur multi-Wilayah. Untuk detail selengkapnya, lihat [skenario Multi-Wilayah](#).
- Pantau transfer data: Gunakan Amazon CloudWatch dan [VPCflow log](#) untuk menangkap detail tentang transfer data dan penggunaan jaringan Anda. Analisis informasi lalu lintas jaringan yang ditangkap di AndaVPCs, seperti alamat IP atau jangkauan ke dan dari antarmuka jaringan.
- Analisis penggunaan jaringan Anda: Gunakan alat pengukuran dan pelaporan seperti AWS Cost Explorer, CUDOS Dasbor, atau CloudWatch untuk memahami biaya transfer data beban kerja Anda.

Langkah-langkah implementasi

- Pilih komponen untuk transfer data: Gunakan pemodelan transfer data yang dijelaskan di [COST08-BP01 Lakukan pemodelan transfer data](#), fokus pada area dengan biaya transfer data terbesar atau area lain 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-praktik terbaik terkait:

- [COST08-BP01 Lakukan pemodelan transfer data](#)
- [COST08-BP03 Mengimplementasikan layanan untuk mengurangi biaya transfer data](#)

Dokumen terkait:

- [Migrasi Data Cloud](#)
- [solusi penerapan cache AWS](#)
- [Mengirimkan konten lebih cepat dengan Amazon CloudFront](#)

Contoh terkait:

- [Ikhtisar Biaya Transfer Data untuk Arsitektur Umum](#)
- [AWS Tips Optimasi Jaringan](#)
- [Optimalkan kinerja dan kurangi biaya untuk analitik jaringan dengan VPC Flow Logs 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 diterapkan: 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 sebuah jaringan penyampaian konten yang menyampaikan data dengan latensi rendah dan kecepatan transfer data 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 bundling penghematan keamanan](#) dapat membantu Anda menghemat hingga 30% atas penggunaan CloudFront Anda jika Anda berencana meningkatkan penggunaan Anda dari waktu ke waktu.
- [AWS Direct Connect](#) memungkinkan Anda membangun sebuah 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 membangun koneksi yang aman dan privat antara jaringan privat Anda dan jaringan AWS global. 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 antar layanan AWS lewat jaringan pribadi dan dapat digunakan untuk mengurangi transfer data publik serta biaya [gateway NAT](#). [Titik akhir VPC Gateway](#) tidak mengenakan biaya per jam, dan mendukung Amazon S3 dan Amazon DynamoDB. [Titik akhir VPC antarmuka](#) disediakan oleh [AWS PrivateLink](#) dan memiliki biaya per jam dan biaya penggunaan per GB.
- [Gateway NAT](#) menyediakan penskalaan dan pengelolaan bawaan untuk mengurangi biaya dibandingkan dengan instans NAT yang berdiri sendiri. Tempatkan gateway NAT di dalam Zona Ketersediaan yang sama karena instans-instans memiliki lalu lintas yang tinggi dan pertimbangkan menggunakan titik akhir VPC untuk instans-instans yang perlu mengakses Amazon DynamoDB atau Amazon S3 untuk mengurangi biaya transfer data dan pemrosesan data.
- Gunakan perangkat [AWS Snow Family](#) yang memiliki sumber daya komputasi untuk mengumpulkan dan memproses data di edge. Perangkat AWS Snow Family ([Snowball Edge](#), [Snowball Edge](#) and [Snowmobile](#)) memungkinkan Anda memindahkan petabyte data ke biaya AWS Cloud secara efektif dan offline.

Langkah-langkah implementasi

- Terapkan layanan: Pilih layanan jaringan AWS yang berlaku berdasarkan jenis beban kerja layanan Anda dengan menggunakan pemodelan transfer data dan peninjauan 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](#)
- [Jelajahi Produk Kami AWS](#)
- [solusi penerapan cache AWS](#)
- [Amazon CloudFront](#)
- [AWS Snow Family](#)
- [Paket Penghematan Keamanan CloudFront dari Amazon](#)

Video terkait:

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

Contoh terkait:

- [Cara melakukan chargeback layanan bersama: Sebuah contoh AWS Transit Gateway](#)
- [Memahami rincian transfer data AWS secara mendalam dari laporan biaya dan penggunaan dengan 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 saya dapat 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 hal itu akan menghilangkan pemborosan biaya dan penyediaan berlebih yang tidak terpakai. Anda juga dapat mengubah permintaan dengan menggunakan throttle, buffer, atau antrian, untuk memperlancar permintaan serta memenuhinya dengan sumber daya yang lebih sedikit.

Manfaat ekonomi dari pemasokan yang dilakukan secara 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 akan membantu Anda meminimalkan manajemen lingkungan Anda – bahkan saat Anda menskalakan. Saat melakukan modifikasi terhadap permintaan, Anda harus mengetahui penundaan maksimum yang dapat diterima dan dapat ditangani oleh beban kerja.

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

Praktik terbaik

- [COST09-BP01 Melakukan analisis pada permintaan beban kerja](#)
- [COST09-BP02 Menerapkan 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. Pemanfaatan sumber daya dan metrik kinerja: Menganalisis 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: Memahami perilaku pengguna dan bagaimana hal itu 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-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 identifikasi 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 pada sumber daya AWS Anda. Anda juga dapat menggunakan Amazon CloudWatch untuk mengetahui visibilitas di seluruh sistem ke dalam pemanfaatan sumber daya, performa 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 (penskalaan otomatis) agar sesuai dengan peningkatan atau penurunan permintaan.

Terakhir, Anda dapat menggunakan [AWS Cost Explorer](#) atau [QuickSight](#) dengan file (CUR) AWS Cost and Usage Report 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-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.
- Perkirakan pengaruh dari 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](#)
- [Mulai menggunakan Amazon SQS](#)
- [AWS Cost Explorer](#)
- [QuickSight](#)

Contoh terkait:

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

COST09-BP02 Menerapkan buffer atau throttle untuk mengelola permintaan

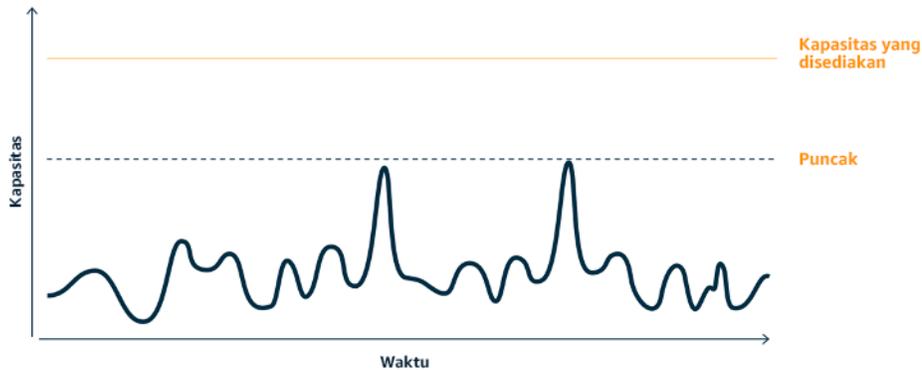
Buffering dan throttling memodifikasi permintaan di beban kerja Anda, sehingga akan menstabilkan fluktuasi. Implementasikan throttling ketika klien Anda mencoba ulang. Implementasikan buffering untuk menyimpan permintaan dan menunda pemrosesan ke lain waktu. Pastikan throttle dan buffer Anda didesain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

Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: 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 berbeda yang membutuhkan kapasitas penyediaan tinggi

Asumsikan beban kerja dengan kurva permintaan yang ditunjukkan pada gambar berikut. Beban kerja ini memiliki dua puncak, dan untuk menangani puncak-puncak ini, disediakan kapasitas sumber daya sebagaimana yang ditunjukkan oleh garis berwarna oranye. 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 terhadap 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, maka 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 menerapkan throttling.

Berbasis buffer: Sebuah pendekatan berbasis buffer menggunakan produsen (komponen yang mengirimkan pesan ke antrean), konsumen (komponen yang menerima pesan dari antrean), dan antrean (yang menyimpan pesan) untuk menyimpan pesan. Pesan dibaca oleh konsumen dan diproses, sehingga pesan dapat dijalankan dengan tingkat yang memenuhi persyaratan-persyaratan

bisnis konsumen. Dengan menggunakan metodologi buffer-sentris, pesan dari produsen disimpan dalam antrean atau aliran, siap diakses oleh konsumen dalam kecepatan yang selaras dengan tuntutan operasional mereka.

Di AWS, Anda dapat memilih dari beberapa layanan untuk menerapkan pendekatan buffering. [Amazon Simple Queue Service \(AmazonSQS\)](#) adalah layanan terkelola yang menyediakan antrian yang memungkinkan satu konsumen membaca pesan individual. [Amazon Kinesis](#) memberikan aliran yang memungkinkan banyak konsumen untuk 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. Lakukan analisis terhadap permintaan secara keseluruhan, tingkat perubahan, dan waktu respons yang diperlukan untuk ukuran throttling atau buffering yang tepat.

Langkah-langkah implementasi

- Analisis kebutuhan klien: Analisis permintaan klien untuk menentukan apakah klien ini mampu 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.
- Menerapkan buffer atau throttling: Menerapkan buffer atau throttle di beban kerja. Antrian seperti Amazon Simple Queue Service SQS (Amazon) dapat menyediakan buffer ke komponen beban kerja Anda. Amazon API Gateway dapat menyediakan pembatasan untuk komponen beban kerja Anda.

Sumber daya

Praktik-praktik terbaik terkait:

- [SUS02-BP06 Menerapkan buffering atau throttling untuk meratakan kurva permintaan](#)
- [REL05-BP02 Permintaan Throttle](#)

Dokumen terkait:

- [AWS Auto Scaling](#)

- [Penjadwal Instans AWS](#)
- [API Gerbang Amazon](#)
- [Amazon Simple Queue Service](#)
- [Memulai dengan Amazon SQS](#)
- [Amazon Kinesis](#)

Video terkait:

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

Contoh terkait:

- [Mengelola dan memantau API pembatasan dalam beban kerja Anda](#)
- [Melambat skala REST API multi-tenant berjenjang menggunakan Gateway API](#)
- [Mengaktifkan Tiering dan Throttling dalam Solusi Amazon SaaS EKS Multi-Tenant Menggunakan Amazon Gateway API](#)
- [Integrasi Aplikasi Menggunakan Antrian 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.

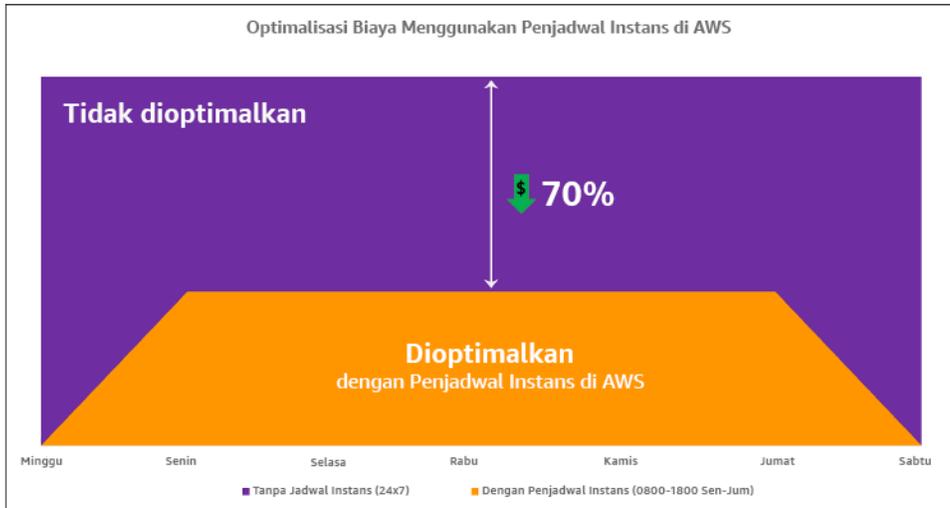
Tingkat risiko yang terjadi jika praktik terbaik ini tidak diterapkan: Rendah

Panduan implementasi

Ada beberapa cara bagi AWS pelanggan untuk meningkatkan sumber daya yang tersedia untuk aplikasi mereka dan menyediakan sumber daya untuk memenuhi permintaan. Salah satu opsi ini adalah menggunakan AWS Instance Scheduler, yang mengotomatiskan memulai dan menghentikan instans Amazon Elastic Compute Cloud (AmazonEC2) dan Amazon Relational Database Service (Amazon). RDS Opsi lainnya adalah menggunakan AWS Auto Scaling, yang memungkinkan Anda untuk secara otomatis menskalakan sumber daya komputasi Anda 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.

[AWS Penjadwal Instance](#) memungkinkan Anda mengonfigurasi penghentian dan awal RDS instans Amazon EC2 dan Amazon pada waktu yang ditentukan sehingga Anda dapat memenuhi permintaan sumber daya yang sama dalam pola waktu yang konsisten seperti setiap hari pengguna mengakses EC2 instans Amazon pada pukul delapan pagi yang tidak mereka butuhkan setelah jam enam malam. Solusi ini membantu mengurangi biaya operasional dengan menghentikan sumber daya yang tidak digunakan dan memulainya saat diperlukan.



Optimalisasi biaya dengan AWS Instance Scheduler.

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

[AWS Auto Scaling](#) akan membantu Anda menyesuaikan kapasitas Anda untuk menjaga kinerja yang stabil dan terprediksi dengan biaya serendah mungkin untuk memenuhi permintaan yang berubah. Ini adalah layanan yang dikelola sepenuhnya dan gratis untuk meningkatkan kapasitas aplikasi Anda yang terintegrasi dengan EC2 instans Amazon dan Spot Fleets, Amazon, Amazon ECS DynamoDB, dan Amazon Aurora. Penskalaan otomatis 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 kapasitas grup Auto Scaling Anda:

- Menjaga tingkat instance saat ini setiap saat
- Menskalakan secara manual
- Menskalakan berdasarkan jadwal
- Menskalakan berdasarkan permintaan
- Gunakan penskalaan prediktif

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

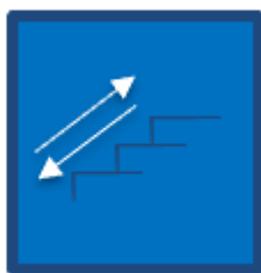
Anda dapat menggunakan AWS Auto Scaling atau menggabungkan penskalaan dalam kode Anda dengan [AWS APIs atau SDKs](#). Hal ini akan menghemat biaya beban kerja secara keseluruhan dengan menghilangkan biaya operasional yang diperlukan untuk membuat perubahan secara manual di lingkungan Anda, dan perubahan dapat dilakukan dengan lebih cepat. Hal ini juga menyesuaikan pengadaan sumber daya beban kerja Anda dengan permintaan Anda kapan saja. Untuk mengikuti praktik terbaik ini dan menyediakan sumber daya secara dinamis untuk organisasi Anda, Anda harus memahami penskalaan horizontal dan vertikal dalam AWS Cloud, serta sifat aplikasi yang berjalan di instans AmazonEC2. Sebaiknya tim Manajemen Keuangan Cloud Anda bekerja dengan tim teknis untuk mengikuti praktik terbaik ini.

[Penyeimbang Beban Elastis \(Penyeimbang Beban Elastis\)](#) membantu Anda melakukan penskalaan dengan mendistribusikan permintaan ke berbagai 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 dalam grup Auto Scaling. Permintaan akan didistribusikan di antara semua target dari grup target secara round-robin tanpa mempertimbangkan kapasitas atau pemanfaatan.

Metrik tipikal dapat berupa EC2 metrik Amazon standar, seperti CPU pemanfaatan, throughput jaringan, dan permintaan yang diamati Elastic Load Balancing dan latensi respons. 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 Penskalaan Otomatis ke dalam dua kategori yakni model penyediaan berdasarkan permintaan dan berdasarkan waktu, dan kami akan menjelaskan masing-masing.

Pasokan berdasarkan permintaan: Manfaatkan elastisitas cloud untuk memasok sumber daya guna memenuhi permintaan yang berubah-ubah dengan mengandalkan kondisi permintaan yang mendekati waktu nyata. Untuk penawaran, penggunaan, APIs atau fitur layanan berbasis permintaan untuk memvariasikan jumlah sumber daya cloud dalam arsitektur Anda secara terprogram. 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)



**Penskalaan Sederhana/
Langkah**



Pelacakan Target

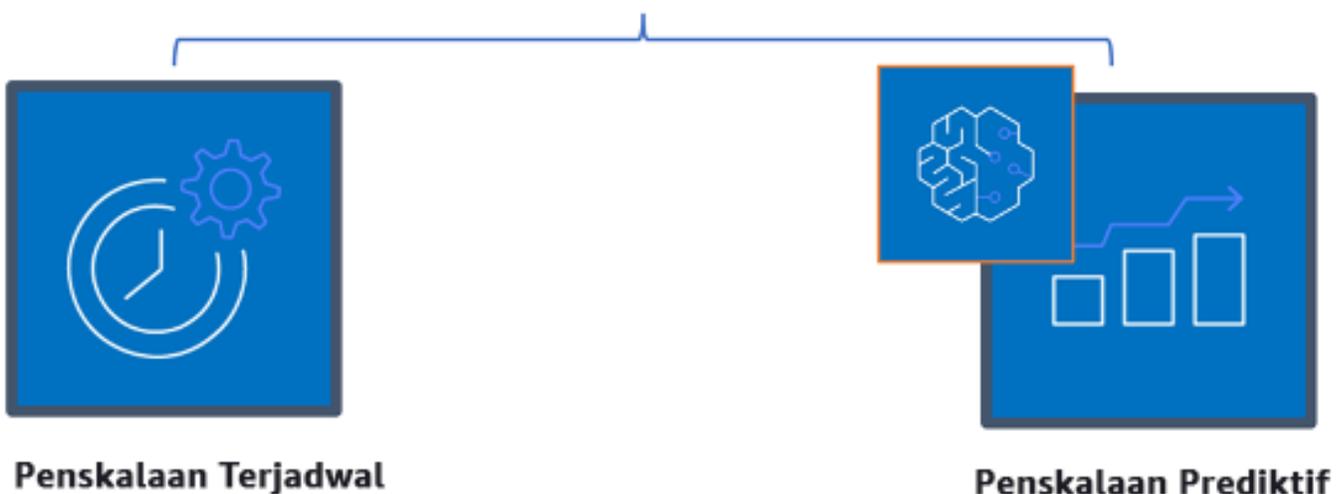
Kebijakan penskalaan dinamis berdasarkan permintaan

- **Penskalaan Sederhana per 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.

Pasokan berdasarkan waktu: Pendekatan berdasarkan waktu yang 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 menambahkan skala (scale out) atau menurunkan skala (scale in) pada waktu yang ditentukan (misalnya, awal jam kerja), sehingga sumber daya tersedia saat pengguna datang atau permintaan meningkat. Penskalaan prediktif menggunakan pola untuk menambahkan skala (scale out) sedangkan penskalaan terjadwal menggunakan waktu yang ditetapkan di awal untuk menambahkan skala (scale out). Anda juga dapat menggunakan [strategi pemilihan tipe instance \(ABS\) berbasis atribut](#) dalam grup Auto Scaling, yang memungkinkan Anda mengekspresikan persyaratan instance sebagai sekumpulan atribut, seperti CPU v, memori, dan penyimpanan. Ini juga memungkinkan Anda untuk secara otomatis menggunakan jenis instans generasi yang lebih baru saat dirilis dan mengakses jangkauan kapasitas yang lebih luas dengan Instans EC2 Spot Amazon. Amazon EC2 Fleet dan Amazon EC2 Auto Scaling memilih dan meluncurkan instance yang sesuai dengan atribut yang ditentukan, sehingga tidak perlu memilih jenis instance secara manual.

Anda juga dapat memanfaatkan [AWS APIs dan SDKs dan AWS CloudFormation](#) secara otomatis menyediakan dan menonaktifkan seluruh lingkungan saat Anda membutuhkannya. Pendekatan ini ideal untuk lingkungan pengujian atau pengembangan yang hanya berjalan pada jam kerja atau periode waktu yang ditentukan. Anda dapat menggunakan APIs untuk menskalakan ukuran sumber daya 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 Amazon EBS Elastic Volume, yang dapat dimodifikasi untuk meningkatkan ukuran, menyesuaikan kinerja (IOPS) atau mengubah jenis volume saat 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-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. Penskalaan ini juga bermanfaat jika pembuatan dan konfigurasi sumber daya tidak cukup cepat untuk merespons perubahan permintaan. Menggunakan analisis beban kerja untuk mengonfigurasi penskalaan terjadwal menggunakan AWS Auto Scaling. Untuk mengonfigurasi

penjadwalan berbasis waktu, Anda dapat menggunakan penskalaan prediktif penskalaan terjadwal untuk meningkatkan jumlah instans Amazon EC2 di grup Auto Scaling Anda terlebih dahulu sesuai dengan perubahan beban yang diharapkan atau yang dapat diprediksi.

- Konfigurasi penskalaan prediktif: Penskalaan prediktif memungkinkan Anda meningkatkan jumlah EC2 instans Amazon di grup Auto Scaling sebelum pola harian dan 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 (penskalaan otomatis) untuk melakukan peluncuran pada tingkat sumber daya yang benar, dan pastikan beban kerja menskalakan pada waktu yang diinginkan. Anda dapat meluncurkan dan secara otomatis menyesuaikan armada Instans Sesuai Permintaan dan Instans Spot dalam satu grup Auto Scaling. Selain menerima diskon untuk menggunakan Spot Instance, Anda dapat menggunakan Instans Pesanan atau Savings Plan untuk menerima tarif diskon dengan harga biasa Instans Sesuai Permintaan. Semua faktor ini digabungkan membantu Anda mengoptimalkan penghematan biaya untuk EC2 instans Amazon dan membantu Anda mendapatkan skala dan kinerja yang diinginkan untuk aplikasi Anda.

Sumber daya

Dokumen terkait:

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

Video terkait:

- [Kebijakan Penskalaan Pelacakan Target untuk Penskalaan Otomatis](#)
- [AWS Penjadwal Instance](#)

Contoh terkait:

- [Pemilihan Jenis Instance berbasis atribut untuk Auto Scaling untuk Amazon Fleet EC2](#)
- [Mengoptimalkan biaya Layanan Kontainer Elastis Amazon dengan menggunakan penskalaan terjadwal](#)
- [Penskalaan Prediktif dengan Amazon EC2 Auto Scaling](#)
- [Bagaimana cara menggunakan Penjadwal Instance untuk AWS CloudFormation menjadwalkan EC2 instans Amazon?](#)

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.

Sambil mengoptimalkan beban kerja Anda dari waktu ke waktu dan meningkatkan budaya [CFM](#) di organisasi Anda, lakukan evaluasi biaya atas upaya-upaya Anda untuk operasi di cloud, tinjau operasi cloud yang memakan banyak waktu, dan lakukan otomatisasi atas operasi itu untuk mengurangi upaya dan biaya manusia dengan mengadopsi layanan-layanan AWS terkait, produk pihak ketiga, atau alat khusus (seperti [AWS CLI](#) atau [SDK 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 Tinjau dan analisis 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 tersebut 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 diterapkan: 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. Sama seperti proses [peninjauan pilar kinerja](#) dan proses [peninjauan pilar keandalan](#), identifikasi, validasi, dan prioritaskan kegiatan optimalisasi dan peningkatan serta perbaikan masalah dan masukkan ini ke dalam backlog Anda.

Langkah-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 keseluruhan 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:

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

Contoh terkait:

- [AWS Support Layanan Proaktif](#)
- [Ulasan beban kerja reguler untuk SAP beban kerja](#)

COST10-BP02 Tinjau dan analisis 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 diterapkan: Sedang

Panduan implementasi

AWS terus menambahkan fitur baru sehingga Anda dapat bereksperimen dan berinovasi lebih cepat dengan teknologi terbaru. [AWS Apa yang baru](#) merinci bagaimana melakukan AWS ini dan memberikan gambaran singkat tentang AWS layanan, fitur, dan pengumuman ekspansi Regional saat dirilis. Anda dapat mendalami peluncuran yang telah diumumkan dan menggunakannya untuk meninjau dan menganalisis beban kerja Anda yang sudah ada. Untuk mewujudkan manfaat AWS layanan dan fitur baru, Anda meninjau beban kerja Anda dan menerapkan layanan dan fitur baru sesuai kebutuhan. Ini berarti Anda mungkin perlu mengganti layanan yang ada yang Anda gunakan untuk beban kerja Anda, atau memodernisasi beban kerja Anda untuk mengadopsi layanan baru ini. AWS Misalnya, Anda mungkin meninjau beban kerja Anda dan mengganti komponen perpesanan dengan Layanan Email Sederhana Amazon. 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 [This is My Architecture](#) AWS untuk mempelajari tentang desain arsitektur pelanggan lain, tantangan mereka, dan solusi mereka. Periksa [seri All-In](#) untuk mengetahui aplikasi AWS layanan dan kisah pelanggan dunia nyata. Anda juga dapat menonton seri video [Kembali ke Dasar](#) yang menjelaskan, memeriksa, dan memecah praktik terbaik pola arsitektur cloud dasar. Sumber lain adalah [How to Build This](#) video, yang dirancang untuk membantu orang-orang dengan ide-ide besar tentang cara menghidupkan produk minimum mereka (MVP) menggunakan AWS layanan. Ini adalah cara bagi pembangun dari seluruh dunia yang memiliki ide kuat untuk mendapatkan bimbingan arsitektur dari Arsitek AWS Solusi berpengalaman. Terakhir, Anda dapat meninjau materi sumber daya [Memulai](#), yang di dalamnya terdapat tutorial langkah demi langkah.

Sebelum menjalankan proses peninjauan Anda, ikuti persyaratan bisnis Anda seputar persyaratan beban kerja, keamanan, dan persyaratan privasi data untuk menggunakan layanan tertentu atau persyaratan kinerja dan Wilayah sembari menjalani proses peninjauan yang Anda disetujui.

Langkah-langkah implementasi

- Secara rutin tunjau beban kerja: Menggunakan proses yang telah Anda tetapkan, 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.
- Terapkan layanan baru: Jika hasil analisis mengatakan perubahan harus diterapkan, pertama lakukan hal terdasar dari beban kerja untuk mengetahui biaya saat ini untuk setiap output. Implementasikan perubahan, kemudian lakukan analisis untuk mengonfirmasi biaya baru untuk setiap output.

Sumber daya

Dokumen terkait:

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

Video terkait:

- [AWS - Ini Arsitektur Saya](#)
- [AWS - Kembali ke Dasar-dasar](#)
- [AWS - Seri All-In](#)
- [Bagaimana Membangun Ini](#)

Otomatisasi operasi

Praktik terbaik

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

COST11-BP01 Lakukan otomatisasi untuk operasi

Evaluasi biaya operasional di cloud, dengan berfokus pada penghitungan penghematan waktu dan upaya dalam tugas-tugas administrasi, deployment, mitigasi risiko kesalahan manusia, kepatuhan, dan operasi lainnya melalui otomatisasi. Nilai waktu serta biaya terkait yang diperlukan untuk upaya operasional dan implementasikan otomatisasi untuk tugas-tugas administrasi guna meminimalkan upaya manual apabila memungkinkan.

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

Panduan implementasi

Otomatisasi operasi mengurangi frekuensi tugas manual, meningkatkan efisiensi, dan menguntungkan pelanggan 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 nilai bisnis. Korporasi 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.

Mulailah dengan memprioritaskan aktivitas operasional Anda berdasarkan upaya yang diperlukan dengan melihat biaya operasi secara keseluruhan. Contohnya, berapa lama waktu yang diperlukan untuk melakukan deployment sumber daya baru di cloud, membuat perubahan optimalisasi pada sumber daya yang sudah ada, atau mengimplementasikan konfigurasi-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, periksa waktu yang diluangkan untuk melakukan tugas secara manual dibandingkan tugas otomatis. Prioritaskan otomatisasi aktivitas repetitif, bernilai tinggi, memakan waktu, dan kompleks. Aktivitas yang menimbulkan nilai tinggi atau risiko 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).

Gunakan alat otomatisasi seperti AWS Systems Manager atau AWS Config untuk merampingkan operasi, kepatuhan, pemantauan, siklus hidup, dan proses penghentian. Dengan AWS layanan, alat, dan produk pihak ketiga, Anda dapat menyesuaikan otomatisasi yang Anda terapkan untuk memenuhi kebutuhan spesifik Anda. Tabel berikut menunjukkan beberapa dari kemampuan dan fungsi operasi inti yang dapat Anda capai dengan layanan AWS untuk mengotomatiskan administrasi dan operasi:

- [AWS Audit Manager](#): Terus audit AWS penggunaan Anda untuk menyederhanakan penilaian risiko dan kepatuhan
- [AWS Backup](#): Secara terpusat mengelola dan mengotomatiskan perlindungan data.
- [AWS Config](#): Mengonfigurasi sumber daya komputasi, penilaian, audit, mengevaluasi konfigurasi, dan melakukan inventarisasi sumber daya.
- [AWS CloudFormation](#): Meluncurkan sumber daya yang sangat tersedia dengan Infrastruktur sebagai Kode.
- [AWS CloudTrail](#): Manajemen, kepatuhan, dan kontrol perubahan TI.
- [Amazon EventBridge](#) Jadwalkan acara dan pemicu AWS Lambda untuk mengambil tindakan.
- [AWS Lambda](#): Otomatiskan proses berulang dengan memicunya dengan peristiwa atau dengan menjalankannya pada jadwal tetap dengan AWS EventBridge
- [AWS Systems Manager](#): Memulai dan menghentikan beban kerja, menambal sistem operasi, mengotomatiskan konfigurasi, dan manajemen berkelanjutan.
- [AWS Step Functions](#): Jadwalkan pekerjaan dan otomatiskan alur kerja.
- [AWS Service Catalog](#): Konsumsi templat, infrastruktur sebagai kode dengan kepatuhan dan kontrol.

Jika Anda ingin segera mengadopsi otomatisasi dengan menggunakan AWS produk dan layanan dan jika tidak memiliki keterampilan dalam organisasi Anda, hubungi [AWS Managed Services \(AMS\)](#), [Layanan AWS Profesional](#), atau [AWS Mitra](#) untuk meningkatkan adopsi otomatisasi dan meningkatkan keunggulan operasional Anda di cloud.

AWS Managed Services (AMS) adalah layanan yang mengoperasikan AWS infrastruktur atas nama pelanggan dan mitra perusahaan. Layanan ini menyediakan lingkungan yang aman dan patuh untuk melakukan deployment beban kerja Anda. AMS menggunakan model operasi cloud perusahaan dengan otomatisasi untuk memungkinkan Anda memenuhi persyaratan organisasi Anda, pindah ke cloud lebih cepat, dan mengurangi biaya manajemen yang sedang berlangsung.

AWS Layanan Profesional juga dapat membantu Anda mencapai hasil bisnis yang Anda inginkan dan mengotomatiskan operasi dengan AWS. Praktik ini membantu pelanggan melakukan deployment operasi IT otomatis yang andal dan tangkas, dan kemampuan tata kelola yang dioptimalkan untuk cloud. Untuk contoh pemantauan mendetail dan praktik terbaik yang direkomendasikan, lihat laporan resmi Pilar Keunggulan Operasional.

Langkah-langkah implementasi

- Bangun sekali dan terapkan banyak: Gunakan infrastructure-as-code seperti CloudFormation, AWS SDK, atau AWS CLI gunakan sekali dan gunakan berkali-kali untuk lingkungan serupa atau untuk skenario pemulihan bencana. Terapkan tag saat melakukan deployment untuk melacak pemakaian Anda sebagaimana ditetapkan dalam praktik terbaik lainnya. Gunakan [AWS Launch Wizard](#) untuk mengurangi waktu untuk menyebarkan banyak beban kerja perusahaan populer. AWS Launch Wizard memandu Anda melalui ukuran, konfigurasi, dan penyebaran beban kerja perusahaan mengikuti AWS praktik terbaik. Anda juga dapat menggunakan [Service Catalog](#), yang membantu Anda membuat dan mengelola templat yang infrastructure-as-code disetujui untuk digunakan AWS sehingga siapa pun dapat menemukan sumber daya cloud swalayan yang disetujui.
- Otomatiskan kepatuhan berkelanjutan: Pertimbangkan untuk melakukan otomatisasi terhadap pelaksanaan evaluasi dan remediasi atas konfigurasi terekam berdasarkan standar yang telah ditentukan sebelumnya. Ketika Anda menggabungkan AWS Organizations dengan kemampuan AWS Config dan [AWS CloudFormation](#), Anda dapat mengelola dan mengotomatiskan kepatuhan konfigurasi secara efisien dalam skala besar untuk ratusan akun anggota. Anda dapat meninjau perubahan konfigurasi dan hubungan antar AWS sumber daya dan menyelami riwayat konfigurasi sumber daya.
- Otomatiskan tugas pemantauan AWS menyediakan berbagai alat yang dapat Anda gunakan untuk memantau layanan. Anda dapat mengonfigurasi alat-alat tersebut untuk mengotomatiskan tugas pemantauan. Buat dan implementasikan rencana pemantauan yang mengumpulkan data pemantauan dari semua bagian dalam beban kerja Anda, sehingga Anda dapat lebih mudah melakukan debugging kegagalan multitik apabila terjadi. Misalnya, Anda dapat menggunakan alat pemantauan otomatis untuk mengamati Amazon EC2 dan melaporkan kembali kepada Anda ketika ada sesuatu yang salah untuk pemeriksaan status sistem, pemeriksaan status instans, dan CloudWatch alarm Amazon.
- Otomatiskan pemeliharaan dan operasi: Jalankan operasi rutin secara otomatis tanpa campur tangan manusia. Dengan menggunakan AWS layanan dan alat, Anda dapat memilih AWS otomatisasi mana yang akan diterapkan dan disesuaikan untuk kebutuhan spesifik Anda. Misalnya, gunakan [EC2Image Builder](#) untuk membangun, menguji, dan menyebarkan mesin virtual dan gambar kontainer untuk digunakan di AWS atau di tempat atau menambal instance

Anda EC2. AWS SSM Jika tindakan yang Anda inginkan tidak dapat dilakukan dengan AWS layanan atau Anda memerlukan tindakan yang lebih kompleks dengan memfilter sumber daya, maka otomatisasi operasi Anda dengan menggunakan [AWS Command Line Interface](#) (AWS CLI) atau AWS SDK alat. AWS CLI menyediakan kemampuan untuk mengotomatiskan seluruh proses pengendalian dan pengelolaan AWS layanan dengan skrip tanpa menggunakan AWS Management Console. Pilih pilihan Anda AWS SDKs untuk berinteraksi dengan AWS layanan. Untuk contoh kode lainnya, lihat [Repositori contoh AWS SDK](#) kode.

- Buat siklus hidup berkelanjutan dengan otomatisasi: Penting bagi Anda untuk membuat dan mempertahankan kebijakan siklus hidup yang matang tidak hanya untuk peraturan atau redundansi tetapi juga untuk pengoptimalan biaya. Anda dapat menggunakannya AWS Backup untuk mengelola dan mengotomatiskan perlindungan data penyimpanan data secara terpusat, seperti bucket, volume, database, dan sistem file Anda. Anda juga dapat menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan pembuatan, penyimpanan, dan penghapusan snapshot dan -backed. EBS EBS AMIs
- Hapus sumber daya yang tidak perlu: Sangat umum untuk mengakumulasi sumber daya yang tidak digunakan di kotak pasir atau pengembangan. Akun AWS Developer membuat dan bereksperimen dengan berbagai layanan dan sumber daya sebagai bagian dari siklus pengembangan normal, kemudian mereka tidak menghapus sumber daya tersebut ketika sudah tidak diperlukan. Sumber daya yang tidak digunakan dapat menimbulkan biaya yang tidak perlu dan terkadang mahal bagi organisasi. Dengan menghapus sumber daya tersebut, biaya pengoperasian lingkungan ini dapat berkurang. Pastikan data Anda tidak diperlukan atau sudah dicadangkan jika Anda tidak yakin. Anda dapat menggunakan AWS CloudFormation untuk membersihkan stack yang telah di-deploy, yang secara otomatis menghapus sebagian besar sumber daya yang ditentukan dalam templat. [Atau, Anda dapat membuat otomatisasi untuk penghapusan AWS sumber daya menggunakan alat seperti aws-nuke.](#)

Sumber daya

Dokumen terkait:

- [Modernisasi operasi di AWS Cloud](#)
- [Layanan AWS untuk otomatisasi](#)
- [Infrastruktur dan Otomatisasi](#)
- [AWS Otomatisasi Systems Manager](#)
- [Pemantauan otomatis dan manual](#)

- [AWS otomatisasi untuk SAP administrasi dan operasi](#)
- [AWS Managed Services](#)
- [Layanan Profesional AWS](#)

Video terkait:

- [Otomatisasikan Kepatuhan Berkelanjutan pada Skala di AWS](#)
- [AWS Backup Demo: Cadangan Lintas Akun & Lintas Wilayah](#)
- [Menambal untuk Instans Amazon EC2 Anda](#)

Contoh terkait:

- [Menyempurnakan operasi yang diotomatisasi \(Bagian I\)](#)
- [Menyempurnakan operasi yang diotomatisasi \(Bagian II\)](#)
- [Mengotomatiskan penghapusan AWS sumber daya dengan menggunakan aws-nuke](#)
- [Hapus EBS volume Amazon yang tidak digunakan dengan menggunakan AWS Config dan AWS SSM](#)
- [Otomatisasikan kepatuhan berkelanjutan dalam skala besar AWS](#)
- [Otomatisasi TI dengan AWS Lambda](#)

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 Anda, serta memperbarui pilihan komponen Anda.

AWS berusaha untuk membantu Anda meminimalkan biaya saat Anda membangun penerapan yang sangat tangguh, responsif, dan adaptif. Untuk benar-benar menerapkan optimasi biaya deployment Anda, manfaatkan alat, teknik, dan praktik terbaik yang dijelaskan dalam tulisan ini.

Kontributor

Para kontributor untuk dokumen ini antara lain:

- Fatih (Ben) Mergen, Lead pada Pilar Optimalisasi Biaya, Layanan Web Amazon, Well-Architected
- Keith Jarrett, Business Development Lead – Optimalisasi Biaya, Layanan Web Amazon
- Arthur Basbaum, Manajer Pengembang Bisnis, Layanan Web Amazon
- Jarman Hauser, Arsitek Komersil, Layanan Web Amazon

Sumber bacaan lebih lanjut

Untuk mendapatkan informasi tambahan, buka:

- [Kerangka Kerja AWS Well-Architected](#)
- [Pusat Arsitektur AWS](#)

Revisi dokumen

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

| Perubahan | Deskripsi | Tanggal |
|---|---|-----------------|
| Panduan praktik terbaik yang sudah diperbarui | Beberapa pembaruan terhadap praktik terbaik. Praktik terbaik baru COST 06-BP04. | 27 Juni 2024 |
| Panduan praktik terbaik yang sudah diperbarui | Pembaruan praktik terbaik minor di berbagai bagian. | 6 Desember 2023 |
| Panduan praktik terbaik yang sudah diperbarui | Praktik terbaik sudah diperbarui dengan panduan-panduan baru di seluruh pilar. | 3 Oktober 2023 |
| Panduan praktik terbaik yang sudah diperbarui | Praktik terbaik diperbarui dengan panduan baru di bidang-bidang berikut: Tata Kelola , Pemantauan biaya dan penggunaan , Pemilihan model harga terbaik , dan Pengelolaan sumber daya permintaan dan pasokan . | 13 Juli 2023 |
| Pembaruan untuk Kerangka Kerja baru | Praktik terbaik diperbarui dengan panduan preskriptif dan praktik terbaik baru ditambahkan. Pertanyaan COST 11 ditambahkan dengan praktik terbaik baru COST11 -BP01. | 10 April 2023 |

| | | |
|---|--|------------------|
| Pembaruan kecil | Panduan yang hilang akan dikembalikan ke bagian model penetapan harga. | 13 Januari 2023 |
| Laporan resmi diperbarui | Praktik terbaik sudah diperbarui dengan panduan implementasi yang baru. | 15 Desember 2022 |
| Laporan resmi diperbarui | Praktik terbaik diperluas dan rencana pengembangan sudah ditambahkan. | 20 Oktober 2022 |
| Pembaruan kecil | Penambahan Pilar Keberlanjutan ke pengantar. | 2 Desember 2021 |
| Pembaruan kecil | Tautan diperbarui. | 25 April 2021 |
| Pembaruan kecil | Tautan diperbarui. | 10 Maret 2021 |
| Pembaruan untuk Kerangka Kerja baru | Diperbarui untuk menggabungkan CFM, layanan baru, dan integrasi dengan Well-Architected juga. | 8 Juli 2020 |
| Laporan resmi diperbarui | Diperbarui untuk mencerminkan perubahan AWS dan menggabungkan pembelajaran dari ulasan dengan pelanggan. | 1 Juli 2018 |
| Laporan resmi diperbarui | Diperbarui untuk mencerminkan perubahan AWS dan menggabungkan pembelajaran dari ulasan dengan pelanggan. | 1 November 2017 |
| Publikasi awal | Pilar Pengoptimalan Biaya - Kerangka AWS Well-Architected diterbitkan. | 1 November 2016 |

Pemberitahuan

Pelanggan bertanggung jawab untuk membuat penilaian independen mereka sendiri atas informasi dalam dokumen ini. Dokumen ini: (a) hanya untuk tujuan informasi, (b) mewakili penawaran dan praktik AWS produk saat ini, yang dapat berubah tanpa pemberitahuan, dan (c) tidak membuat komitmen atau jaminan apa pun dari AWS dan afiliasinya, pemasok, atau pemberi lisensinya. AWS produk atau layanan disediakan “sebagaimana adanya” tanpa jaminan, representasi, atau kondisi apa pun, baik tersurat maupun tersirat. Tanggung jawab dan kewajiban AWS kepada pelanggannya dikendalikan oleh AWS perjanjian, dan dokumen ini bukan bagian dari, juga tidak mengubah, perjanjian apa pun antara AWS dan pelanggannya.

© 2023 Amazon Web Services, Inc. atau afiliasinya. Semua hak dilindungi undang-undang.

AWS Glosarium

Untuk AWS terminologi terbaru, lihat [AWS glosarium di Referensi](#).Glosarium AWS