AWSWhitepaper

Praktik Terbaik untuk Menandai Sumber Daya AWS

Praktik Terbaik untuk Menandai Sumber Daya AWS: AWSWhitepaper

Table of Contents

Praktik Terbaik untuk Menandai Sumber Daya AWS

Tanggal publikasi: 30 Maret 2023 () Revisi dokumen

Amazon Web Services (AWS) memungkinkan Anda untuk menetapkan metadata ke banyak AWS sumber daya Anda dalam bentuk tag. Setiap tag adalah label sederhana yang terdiri dari kunci dan nilai opsional untuk menyimpan informasi tentang sumber daya atau data yang disimpan pada sumber daya tersebut. Whitepaper ini berfokus pada penandaan kasus penggunaan, strategi, teknik, dan alat yang dapat membantu Anda mengkategorikan sumber daya berdasarkan tujuan, tim, lingkungan, atau kriteria lain yang relevan dengan bisnis Anda. Menerapkan strategi penandaan yang konsisten dapat mempermudah penyaringan dan pencarian sumber daya, memantau biaya dan penggunaan, dan mengelola AWS lingkungan Anda.

Paper ini didasarkan pada praktik dan panduan yang disediakan dalam whitepaper Mengatur AWS Lingkungan Anda Menggunakan Beberapa Akun. Disarankan agar Anda membaca whitepaper sebelum yang satu ini. AWSmerekomendasikan agar Anda membangun fondasi cloud Anda secara holistik. Untuk informasi tambahan, lihat Membangun Cloud Foundation Anda di AWS.

Apakah Anda Well-Architected?

AWSWell-Architected Framework membantu Anda memahami pro dan kontra dari keputusan yang Anda buat saat membangun sistem di cloud. Enam pilar Kerangka memungkinkan Anda mempelajari praktik terbaik arsitektur untuk merancang dan mengoperasikan sistem yang andal, aman, efisien, hemat biaya, dan berkelanjutan. Dengan menggunakan AWS Well-Architected Tool, tersedia tanpa biaya di AWS Management Console, Anda dapat meninjau beban kerja Anda terhadap praktik terbaik ini dengan menjawab serangkaian pertanyaan untuk setiap pilar.

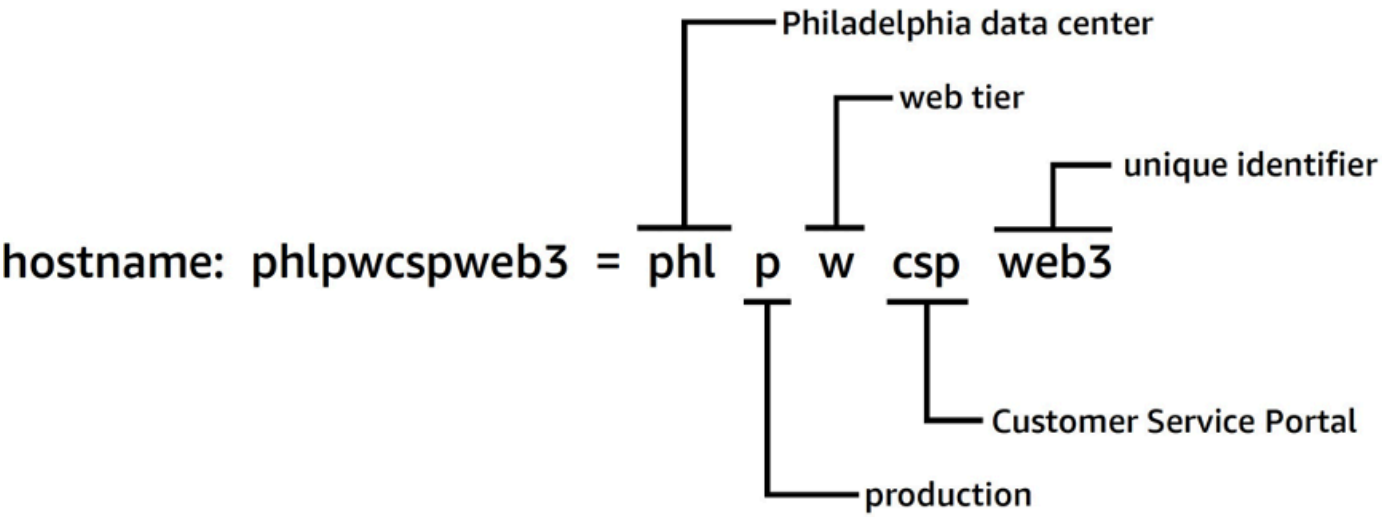
Untuk panduan ahli dan praktik terbaik lainnya untuk arsitektur cloud Anda—penerapan arsitektur referensi, diagram, dan whitepaperer—lihat Pusat Arsitektur. AWS

Pengantar

AWSmemudahkan penerapan beban kerja Anda AWS dengan membuat sumber daya, seperti instans Amazon EC2, volume Amazon EBS, grup keamanan, dan fungsi. AWS Lambda Anda juga dapat menskalakan dan menumbuhkan aliran AWS sumber daya yang meng-host aplikasi Anda, menyimpan data Anda, dan memperluas AWS infrastruktur Anda dari waktu ke waktu. Ketika AWS penggunaan Anda tumbuh ke banyak jenis sumber daya yang mencakup beberapa aplikasi, Anda

akan memerlukan mekanisme untuk melacak sumber daya mana yang ditugaskan ke aplikasi mana. Gunakan mekanisme ini untuk mendukung aktivitas operasional Anda, seperti pemantauan biaya, manajemen insiden, penambalan, pencadangan, dan kontrol akses.

Dalam lingkungan lokal, pengetahuan ini sering ditangkap dalam sistem manajemen pengetahuan, sistem manajemen dokumen, dan pada halaman wiki internal. Dengan database manajemen konfigurasi (CMDB), Anda dapat menyimpan dan mengelola metadata terperinci yang relevan menggunakan proses kontrol perubahan standar. Pendekatan ini menyediakan tata kelola, tetapi membutuhkan upaya tambahan untuk mengembangkan dan memelihara. Anda dapat mengambil pendekatan terstruktur untuk penamaan sumber daya, tetapi nama sumber daya hanya dapat menyimpan informasi dalam jumlah terbatas.



Pendekatan terstruktur untuk penamaan sumber daya

Misalnya, instans EC2 memiliki tag yang telah ditentukan sebelumnya bernama Name yang menyediakan fungsionalitas serupa dan memungkinkan Anda memberi nama beban kerja saat dipindahkan. AWS

Pada tahun 2010, AWS meluncurkan tag sumber daya untuk menyediakan mekanisme fleksibel dan terukur untuk melampirkan metadata ke sumber daya Anda. Whitepaper ini memandu Anda melalui proses pengembangan dan penerapan strategi penandaan yang kuat di seluruh lingkungan Anda. AWS Panduan ini akan membantu Anda memastikan konsistensi dan cakupan penandaan yang mendukung pengambilan keputusan dan kegiatan operasional Anda

Apa itu tag?

Tag adalah pasangan kunci-nilai yang diterapkan ke sumber daya untuk menyimpan metadata tentang sumber daya tersebut. Setiap tag adalah label yang terdiri dari kunci dan nilai opsional. Tidak semua layanan dan tipe sumber daya saat ini mendukung tag (lihat Layanan yang mendukung API Penandaan Resource Groups). Layanan lain dapat mendukung tag melalui API mereka sendiri. Perlu dicatat bahwa tag tidak dienkripsi dan tidak boleh digunakan untuk menyimpan data sensitif, seperti informasi pengidentifikasi jatuh diri (PII).

Tag yang dibuat dan diterapkan pengguna ke AWS sumber daya menggunakanAWS CLI, API, atau AWS Management Console yang dikenal sebagai tag yang ditentukan pengguna. Beberapa AWS layanan, sepertiAWS CloudFormation, Elastic Beanstalk, dan Auto Scaling, secara otomatis menetapkan tag ke sumber daya yang mereka buat dan kelola. Kunci ini dikenal sebagai tag AWSyang dihasilkan dan biasanya diawali dengan`aws`. Awalan ini tidak dapat digunakan dalam kunci tag yang ditentukan pengguna.

Ada persyaratan penggunaan dan batasan jumlah tag yang ditentukan pengguna yang dapat ditambahkan ke sumber daya. AWS Untuk informasi selengkapnya, lihat batas dan persyaratan penamaan Tag dalam panduan Referensi AWS Umum. AWStag yang dihasilkan tidak dihitung terhadap batas tag yang ditentukan pengguna ini.

Tabel 1 - Contoh kunci dan nilai tag yang ditentukan pengguna

ID Instans	Kunci Tag	Nilai Tag
`i-01234567abcdef89a`	`CostCenter`	`98765`
	`Stack`	`Test`
i-12345678abcdef90b	`CostCenter`	`98765`
	`Stack`	`Production`

Tabel 2 - Contoh tag AWS yang dihasilkan

AWSTombol Tag yang Dihasilkan	Dasar Pemikiran
`aws:ec2spot:fleet-request-id`	Mengidentifikasi permintaan Instans Spot Amazon EC2 yang meluncurkan instans
`aws:cloudformation:stack-name`	Mengidentifikasi AWS CloudFormation tumpukan yang menciptakan sumber daya
`lambda-console:blueprint`	Mengidentifikasi cetak biru yang digunakan sebagai template untuk suatu fungsi AWS Lambda
`elasticbeanstalk:environment-name`	Mengidentifikasi aplikasi yang menciptakan sumber daya
`aws:servicecatalog: provisionedProductArn`	Produk yang disediakan Amazon Resource Name (ARN)
`aws:servicecatalog:productArn`	ARN dari produk dari mana produk yang disediakan diluncurkan

AWStag yang dihasilkan membentuk namespace. Misalnya, dalam AWS CloudFormation template, Anda menentukan satu set sumber daya yang akan digunakan bersama dalam`stack`, di mana `stack-name` adalah nama deskriptif yang Anda tetapkan untuk mengidentifikasinya. Jika Anda memeriksa kunci seperti***`aws:cloudformation:stack-name`***, Anda dapat melihat namespace yang digunakan untuk cakupan parameter menggunakan tiga elemen: aws the organization, cloudformation the service, dan stack-name the parameter.

Tag yang ditentukan pengguna juga dapat menggunakan ruang nama dan menggunakan pengenal organisasi sebagai awalan disarankan. Ini membantu Anda mengidentifikasi dengan cepat apakah tag adalah sesuatu dari skema terkelola Anda, atau sesuatu yang ditentukan oleh layanan atau alat yang Anda gunakan di lingkungan Anda.

Dalam Establishing Your Cloud Foundation di AWS whitepaper, kami merekomendasikan satu set tag yang harus diterapkan. Sangat mungkin bahwa bisnis yang berbeda akan memiliki pola yang diizinkan berbeda dan daftar yang berbeda untuk tag yang diberikan. Melihat contoh pada Tabel 3:

Tabel 3 - Kunci tag yang sama, aturan validasi nilai yang berbeda

Organisasi	Kunci Tag	Validasi Nilai Tag	Contoh Nilai Tag
Perusahaan A	`CostCenter`	`5432, 5422, 5499`	`5432`
Perusahaan B	`CostCenter`	`ABC*`	`ABC123`

Jika kedua skema ini berada dalam organisasi yang terpisah, maka tidak ada masalah dengan konflik tag. Namun, jika kedua lingkungan ini bergabung, maka ruang nama dapat bertentangan dan validasi menjadi lebih kompleks. Skenario ini mungkin tampak tidak mungkin, tetapi bisnis diperoleh atau digabungkan, dan ada skenario lain, seperti klien yang bekerja dengan penyedia layanan terkelola, penerbit game, atau bisnis modal ventura, di mana akun dari organisasi yang berbeda merupakan bagian dari Organisasi bersamaAWS. Dengan menggunakan nama bisnis sebagai awalan untuk mendefinisikan namespace yang unik, tantangan ini dapat dihindari, seperti yang ditunjukkan pada Tabel 4:

Tabel 4 - Penggunaan ruang nama dalam kunci tag

Organisasi	Kunci Tag	Validasi Nilai Tag	Contoh Nilai Tag
Perusahaan A	`company-a :CostCenter`	`5432, 5422, 5499`	`5432`
Perusahaan B	`company-b :CostCenter`	`ABC*`	`ABC123`

Dalam organisasi besar dan kompleks di mana bisnis diperoleh dan divestasi secara teratur, situasi ini akan lebih sering terjadi. Karena proses dan praktik akuisisi baru diselaraskan di seluruh kelompok yang lebih luas, situasinya teratasi. Memiliki ruang nama yang berbeda membantu karena penggunaan tag lama dapat dilaporkan dan tim yang relevan dihubungi untuk mengadopsi skema baru. Namespace juga dapat digunakan untuk menunjukkan ruang lingkup atau mewakili kasus penggunaan atau area tanggung jawab yang selaras dengan pemilik organisasi.

Tabel 5 - Contoh lingkup atau lingkup kasus penggunaan dalam kunci tag

Kasus Penggunaan	Kunci Tag	Dasar Pemikiran	Nilai yang Diizinkan
Klasifikasi Data	`example-inc:info-sec:data-classification`	Sekelompok klasifikasi data yang ditentukan keamanan informasi	`sensitive`, `company-confidential`, `customer-identifiable`
Operasi	`example-inc:dev-ops:environment`	Menerapkan penjadwalan lingkungan pengujian dan pengembangan	`development`, `staging`, `quality-assurance`, `production`
Pemulihan Bencana	`example-inc:disaster-recovery:rpo`	Tentukan tujuan titik pemulihan (RPO) untuk sumber daya	6h, 24h
Alokasi Biaya	`example-inc:cost-allocation:business-unit`	Tim keuangan membutuhkan pelaporan biaya untuk penggunaan dan pengeluaran masing-masing tim	`corporate`, `recruitment`, `support`, `engineering`

Tag sederhana dan fleksibel. Baik kunci dan nilai tag adalah string panjang variabel dan dapat mendukung kumpulan karakter yang lebar. Untuk informasi selengkapnya tentang panjang dan kumpulan karakter, lihat Menandai AWS sumber daya di Referensi AWS Umum. Tag bersifat case sensitive, artinya `costCenter` dan `costcenter` merupakan kunci tag yang berbeda. Di berbagai negara, ejaan kata mungkin berbeda, yang mungkin memengaruhi kunci Anda. Misalnya, di Amerika Serikat, seseorang mungkin mendefinisikan kunci sebagai`costcenter`, tetapi di Inggris, `costcentre` mungkin lebih disukai. Ini adalah kunci yang berbeda dari perspektif penandaan sumber daya. Tentukan ejaan, kasus, dan tanda baca sebagai bagian dari strategi penandaan Anda. Gunakan definisi ini sebagai referensi bagi siapa saja yang membuat atau mengelola sumber daya. Topik ini dibahas secara lebih rinci di bagian selanjutnya,Membangun strategi penandaan Anda.

Membangun strategi penandaan Anda

Seperti banyak praktik dalam operasi, menerapkan strategi penandaan adalah proses iterasi dan peningkatan. Mulailah dari yang kecil dengan prioritas langsung Anda dan kembangkan skema penandaan sesuai kebutuhan.

Menandai iterasi strategi dan siklus peningkatan

Sepanjang proses ini, kepemilikan adalah kunci akuntabilitas dan kemajuan. Karena tag dapat digunakan untuk berbagai tujuan, strategi penandaan keseluruhan dapat dibagi menjadi area tanggung jawab dalam suatu organisasi. Penandaan memungkinkan pendekatan terprogram untuk kegiatan yang bergantung pada karakterisasi sumber daya. Kisaran pemangku kepentingan yang dapat mengambil manfaat dari penandaan akan tergantung pada ukuran organisasi dan praktik operasional. Organisasi yang lebih besar dapat memperoleh manfaat dari mendefinisikan secara jelas tanggung jawab tim yang terlibat dalam membangun dan menerapkan strategi penandaan. Beberapa pemangku kepentingan dapat bertanggung jawab untuk mengidentifikasi kebutuhan

(mendefinisikan kasus penggunaan) untuk penandaan; yang lain dapat bertanggung jawab untuk memelihara, menerapkan, dan meningkatkan strategi penandaan.

Dengan menetapkan kepemilikan, Anda berada dalam posisi yang baik untuk menerapkan aspek individual dari strategi. Jika perlu, kepemilikan ini dapat diformalkan sebagai kebijakan dan didokumentasikan dalam matriks tanggung jawab (misalnya, RACI: Bertanggung Jawab, Bertanggung Jawab, Dikonsultasikan, dan Diinformasikan), atau dalam model tanggung jawab bersama. Dalam organisasi yang lebih kecil, tim mungkin memainkan banyak peran dalam strategi penandaan, mulai dari definisi persyaratan hingga implementasi dan penegakan hukum.

Mendefinisikan kebutuhan dan kasus penggunaan

Mulailah membangun strategi Anda dengan terlibat dengan pemangku kepentingan yang memiliki kebutuhan mendasar mendasar untuk mengkonsumsi metadata. Tim-tim ini menentukan metadata yang perlu diberi tag sumber daya untuk mendukung aktivitas mereka, seperti pelaporan, otomatisasi, dan klasifikasi data. Mereka menguraikan bagaimana sumber daya perlu diatur dan kebijakan mana yang perlu dipetakan. Contoh peran dan fungsi yang dapat dimiliki oleh para pemangku kepentingan ini dalam organisasi meliputi:

- Keuangan dan Lini Bisnis perlu memahami nilai investasi dengan memetakannya ke biaya untuk memprioritaskan tindakan yang perlu diambil ketika menangani ketidakseimbangan. Memahami biaya vs nilai yang dihasilkan membantu mengidentifikasi lini bisnis atau penawaran produk yang gagal. Ini mengarah pada keputusan berdasarkan informasi tentang dukungan berkelanjutan, mengadopsi alternatif (misalnya, menggunakan penawaran SaaS atau layanan terkelola), atau menghentikan penawaran bisnis yang tidak menguntungkan.
- Tata Kelola dan Kepatuhan perlu memahami kategorisasi data (misalnya, publik, sensitif, atau rahasia), apakah beban kerja tertentu berada di dalam atau di luar ruang lingkup untuk audit terhadap standar atau peraturan tertentu, dan kekritisan layanan (apakah layanan atau aplikasi bersifat bisnis penting) untuk menerapkan kontrol dan pengawasan yang sesuai, seperti izin, kebijakan, dan pemantauan.
- Operasi dan Pengembangan perlu memahami siklus hidup beban kerja, tahapan implementasi produk yang didukung, dan pengelolaan tahapan rilis (misalnya, Pengembangan, Pengujian, Pembagian produksi) dan prioritas dukungan terkait dan persyaratan manajemen pemangku kepentingan. Tugas seperti Backup, Patching, Observability dan Deprecation juga perlu didefinisikan dan dipahami.

- Keamanan Informasi (InfoSec) dan Operasi Keamanan (SecOps) menguraikan kontrol apa yang harus diterapkan dan mana yang direkomendasikan. InfoSec biasanya mendefinisikan implementasi kontrol, dan umumnya SecOps bertanggung jawab untuk mengelola kontrol tersebut.

Bergantung pada kasus penggunaan, prioritas, ukuran organisasi, dan praktik operasional, Anda mungkin memerlukan perwakilan dari berbagai tim dalam organisasi, seperti Keuangan (termasuk Pengadaan), Keamanan Informasi, Pengaktifan Cloud, dan Operasi Cloud. Anda juga memerlukan representasi dari pemilik aplikasi dan proses untuk fungsi-fungsi seperti menambal, mencadangkan dan memulihkan, memantau, penjadwalan pekerjaan, dan pemulihan bencana. Perwakilan ini membantu mendorong definisi, implementasi, dan mengukur efektivitas strategi penandaan. Mereka harus bekerja mundur dari pemangku kepentingan dan kasus penggunaannya, dan melakukan lokakarya lintas fungsi. Dalam lokakarya, mereka mendapatkan kesempatan untuk berbagi perspektif dan kebutuhan mereka, dan membantu mendorong strategi secara keseluruhan. Contoh peserta dan keterlibatan mereka dalam berbagai kasus penggunaan dijelaskan kemudian dalam whitepaper ini.

Para pemangku kepentingan juga mendefinisikan dan memvalidasi kunci untuk tag wajib, dan dapat merekomendasikan ruang lingkup untuk tag opsional. Misalnya, Tim Keuangan mungkin perlu menghubungkan sumber daya ke pusat biaya internal, unit bisnis, atau keduanya. Dengan demikian, mereka mungkin mengharuskan kunci tag tertentu, seperti `CostCenter` dan`BusinessUnit`, dibuat wajib. Tim pengembangan individu mungkin memutuskan untuk menggunakan tag tambahan untuk tujuan otomatisasi, seperti`EnvironmentName`,`OptIn`, atau`OptOut`.

Pemangku kepentingan utama perlu menyetujui pendekatan strategi penandaan, dan mendokumentasikan jawaban untuk pertanyaan terkait kepatuhan dan tata kelola, seperti:

- Kasus penggunaan apa yang perlu ditangani?
- Siapa yang bertanggung jawab untuk menandai sumber daya (implementasi)?
- Bagaimana tag diberlakukan dan metode dan otomatisasi apa yang akan digunakan (proaktif atau reaktif)?
- Bagaimana efektivitas dan tujuan penandaan diukur?
- Seberapa sering strategi penandaan harus ditinjau?
- Siapa yang mendorong perbaikan? Bagaimana ini dilakukan?

Fungsi bisnis, seperti Cloud Enablement, Cloud Business Oce, dan Cloud Platform Engineering, kemudian dapat berperan sebagai fasilitator untuk proses membangun strategi penandaan,

membantu mendorong adopsi, dan memastikan konsistensi aplikasinya dengan mengukur kemajuan, menghilangkan hambatan, dan mengurangi upaya duplikat.

Mendefinisikan dan menerbitkan skema penandaan

Gunakan pendekatan yang konsisten dalam menandai AWS sumber daya Anda, baik untuk tag wajib maupun opsional. Skema penandaan yang komprehensif membantu Anda mencapai konsistensi ini. Contoh berikut dapat membantu Anda memulai:

- Setuju pada kunci tag wajib
- Tentukan nilai yang dapat diterima dan konvensi penamaan tag (huruf besar atau kecil, tanda hubung atau garis bawah, hierarki, dan sebagainya)
- Konfirmasi nilai bukan merupakan informasi identitas pribadi (PII)
- Tentukan siapa yang dapat menentukan dan membuat kunci tag baru
- Setuju tentang cara menambahkan nilai tag wajib baru dan cara mengelola tag opsional

Tinjau tabel kategori penandaan berikut, yang dapat digunakan sebagai dasar dari apa yang mungkin Anda sertakan dalam skema penandaan Anda. Anda masih perlu menentukan konvensi yang akan Anda gunakan untuk kunci tag dan nilai apa yang diizinkan untuk masing-masing. Skema penandaan adalah dokumen di mana Anda mendefinisikan ini untuk lingkungan Anda.

Tabel 6 - Contoh skema penandaan definitif (bagian 1)

Kasus Penggunaan	Tag kunci	Dasar Pemikiran	Nilai yang Diizinkan (Terdaftar atau awalan nilai/sux)	Digunakan untuk Alokasi Biaya	Jenis Sumber Daya	Cakupan	Diperlukan
Alokasi Biaya	`example-inc:cost-allocation: ApplicationId`	Lacak biaya vs nilai yang dihasilkan oleh setiap lini bisnis	`DataLakeX` , `RetailSiteX`	Y	Semua	Semua	Wajib
Alokasi Biaya	`example-inc:cost-allocation: BusinessUnitId`	Pantau biaya berdasarkan unit bisnis	`Architecture` , `DevOps`, `Finance`	Y	Semua	Semua	Wajib
Alokasi Biaya	`example-inc:cost-allocation: CostCenter`	Pantau biaya berdasarkan pusat biaya	`123-*`	Y	Semua	Semua	Wajib
Alokasi Biaya	`example-inc:cost-allocation :Owner`	Pemegang anggaran mana yang bertanggung jawab atas beban kerja ini	`Marketing` , `RetailSupport`	Y	Semua	Semua	Wajib
Kontrol Akses	`example-inc:access-control:`	Identifikasi SubComponent /Layer untuk	`DB_Layer`, `Web_Layer` , `App_Layer`	N	Semua	Semua	Opsional

Tabel 6 - Contoh skema penandaan definitif (bagian 2)

Kasus Penggunaan	Tag kunci	Dasar Pemikiran	Nilai yang Diizinkan (Terdaftar atau awalan nilai/sux)	Digunakan untuk Alokasi Biaya	Jenis Sumber Daya	Cakupan	Diperlukan
DevOps	`example-inc:operations: Owner`	Tim/regu mana yang bertanggung jawab atas pembuatan dan pemeliharaan sumber daya	`Squad01`	N	Semua	Semua	Wajib
Pemulihan Bencana	`example-inc:disaster-recovery:rpo`	Tentukan tujuan titik pemulihan (RPO) untuk sumber daya	`6h`, `24h`	N	S3, EBS	Prod	Wajib
Klasifikasi Data	`example-inc:data:classification`	Klasifikasi data untuk kepatuhan dan tata kelola	`Public`, `Private`, `Confidential` , `Restricted`	N	S3, EBS	Semua	Wajib
Kepatuhan	`example-inc:compliance:framework`	Mengidentifikasi kerangka kerja kepatuhan yang dikenakan beban kerja	`PCI-DSS`, `HIPAA`	N	Semua	Prod	Wajib

Setelah skema penandaan didefinisikan, kelola skema dalam repositori yang dikendalikan versi yang dibuat dapat diakses oleh semua pemangku kepentingan yang relevan untuk referensi yang mudah dan pembaruan yang dapat dilacak. Pendekatan ini meningkatkan efisiensi dan memungkinkan kelincahan.

AWS Organizations— Kebijakan tag

Kebijakan AWS Organizations mengizinkan Anda menerapkan jenis tata kelola tambahan Akun AWS di organisasi Anda. Kebijakan tag adalah bagaimana Anda dapat mengekspresikan skema penandaan Anda dalam bentuk JSON sehingga platform dapat melaporkan dan secara opsional menegakkan skema dalam lingkungan Anda. AWS Kebijakan tag mendefinisikan nilai yang dapat diterima untuk kunci tag pada jenis sumber daya tertentu. Kebijakan ini dapat berupa daftar nilai, atau awalan yang diikuti oleh karakter wildcard ()*. Pendekatan awalan sederhana kurang ketat daripada daftar nilai diskrit tetapi membutuhkan lebih sedikit perawatan.

Contoh berikut menunjukkan cara mendefinisikan kebijakan penandaan untuk memvalidasi nilai yang dapat diterima untuk kunci yang diberikan. Bekerja dari definisi tabel skema yang ramah manusia, Anda dapat menyalin informasi ini ke dalam satu atau beberapa kebijakan tag. Kebijakan terpisah dapat digunakan untuk mendukung kepemilikan yang didelegasikan atau beberapa kebijakan mungkin hanya berlaku dalam skenario tertentu.

ExampleInc- CostAllocation .json

Berikut ini adalah contoh kebijakan tag yang melaporkan tag Alokasi Biaya:

```
{
  "tags": {
    "example-inc:cost-allocation:ApplicationId": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:ApplicationId"
      },
      "tag_value": {
        "@@assign": [
          "DataLakeX",
          "RetailSiteX"
        ]
      }
    },
    "example-inc:cost-allocation:BusinessUnitId": {
```

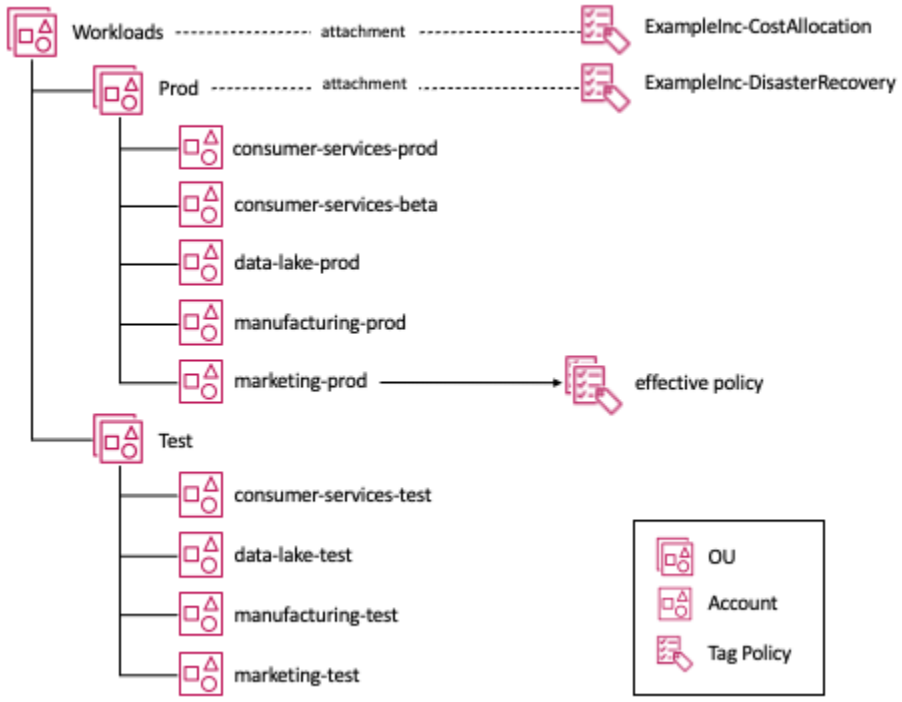
```
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:BusinessUnitId"
      },
      "tag_value": {
        "@@assign": [
          "Architecture",
          "DevOps",
          "FinanceDataLakeX"
        ]
      }
    },
    "example-inc:cost-allocation:CostCenter": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:CostCenter"
      },
      "tag_value": {
        "@@assign": [
          "123-*"
        ]
      }
    }
  }
}
```

ExampleInc- DisasterRecovery .json

Berikut ini adalah contoh kebijakan tag yang melaporkan tag Disaster Recovery:

```
{
    "tags": {
        "example-inc:disaster-recovery:rpo": {
            "tag_key": {
                "@@assign": "example-inc:disaster-recovery:rpo"
            },
            "tag_value": {
                "@@assign": [
                    "6h",
                    "24h"
                ]
            }
        }
    }
}
```

Dalam contoh ini, kebijakan `ExampleInc-CostAllocation` tag dilampirkan ke `Workloads` OU, dan oleh karena itu berlaku untuk semua akun di OU `Prod` dan `Test` anak. Demikian pula, kebijakan `ExampleInc-DisasterRecovery` tag dilampirkan ke `Prod` OU dan oleh karena itu hanya berlaku untuk akun di bawah OU ini. Whitepaper Mengatur Lingkungan Anda Menggunakan Beberapa Akun mengeksplorasi struktur OU yang direkomendasikan secara lebih rinci.



Lampiran kebijakan tag ke struktur OU

Melihat `marketing-prod` akun dalam diagram, kedua kebijakan tag berlaku untuk akun ini, jadi kami memiliki konsep kebijakan yang efektif, yaitu konvolusi kebijakan dari jenis tertentu yang berlaku untuk akun. Jika Anda mengelola sumber daya secara manual, Anda dapat meninjau kebijakan efektif dengan mengunjungi Resource Groups & Tag Editor:Tag policy di konsol. Jika Anda menggunakan infrastruktur sebagai kode (IAc) atau skrip untuk mengelola sumber daya, Anda dapat menggunakan panggilan `AWS::Organizations::DescribeEffectivePolicy`API.

Menerapkan dan menegakkan penandaan

Di bagian ini, kami akan memperkenalkan Anda pada alat yang tersedia untuk strategi manajemen sumber daya berikut: manual, infrastruktur sebagai kode (IAc), dan integrasi berkelanjutan/pengiriman berkelanjutan (CI/CD). Dimensi kunci untuk pendekatan ini adalah tingkat penyebaran yang semakin sering.

Sumber daya yang dikelola secara manual

Ini biasanya beban kerja yang termasuk dalam dasar atau tahap migrasi adopsi. Seringkali, ini adalah beban kerja statis sederhana yang telah dibangun menggunakan prosedur tertulis tradisional atau yang dimigrasi seperti menggunakan alat seperti CloudEndure dari lingkungan lokal. Alat migrasi, seperti Migration Hub dan CloudEndure, dapat menerapkan tag sebagai bagian dari proses migrasi. Namun, jika tag tidak diterapkan selama migrasi asli atau skema penandaan telah berubah sejak saat itu, Editor Tag (fitur dariAWS Management Console) memungkinkan Anda untuk mencari sumber daya menggunakan berbagai kriteria pencarian dan menambah, memodifikasi, atau menghapus tag secara massal. Kriteria pencarian dapat mencakup sumber daya dengan atau tanpa kehadiran tag atau nilai tertentu. AWSResource Tagging API memungkinkan Anda menjalankan fungsi-fungsi ini secara terprogram.

Karena beban kerja ini dimodernisasi, jenis sumber daya seperti grup Auto Scaling diperkenalkan. Jenis sumber daya ini memungkinkan elastisitas yang lebih besar dan ketahanan yang lebih baik. Grup penskalaan otomatis mengelola instans Amazon EC2 atas nama Anda, namun, Anda mungkin masih ingin instans EC2 diberi tag secara konsisten dengan sumber daya yang dibuat secara manual. Template peluncuran Amazon EC2 menyediakan sarana untuk menentukan tag yang harus diterapkan Auto Scaling ke instance yang dibuatnya.

Ketika sumber daya beban kerja dikelola secara manual, akan sangat membantu untuk mengotomatiskan penandaan sumber daya. Ada berbagai solusi yang tersedia. Salah satu pendekatannya adalah dengan menggunakanAturan AWS Config, yang dapat memeriksa `required_tags` dan kemudian memulai fungsi Lambda untuk menerapkannya. Aturan AWS Configdijelaskan lebih detail nanti di whitepaper ini.

Sumber daya yang dikelola Infrastruktur sebagai kode (IAc)

AWS CloudFormationmenyediakan bahasa umum untuk menyediakan semua sumber daya infrastruktur di lingkungan AndaAWS. CloudFormation template adalah file JSON atau YAMAL yang membuat AWS sumber daya secara otomatis. Saat membuat AWS sumber daya menggunakan CloudFormation templat, Anda dapat menggunakan properti Tag CloudFormation Sumber Daya untuk menerapkan tag ke jenis sumber daya yang didukung saat pembuatan. Mengelola tag serta sumber daya dengan IAc membantu memastikan konsistensi.

Ketika sumber daya dibuat olehAWS CloudFormation, layanan secara otomatis menerapkan satu set tag yang AWS ditentukan ke sumber daya yang dibuat oleh AWS CloudFormation template. Ini adalah:

```
aws:cloudformation:stack-name
aws:cloudformation:stack-id
aws:cloudformation:logical-id
```

Anda dapat dengan mudah menentukan grup sumber daya berdasarkan AWS CloudFormation tumpukan. Tag AWS yang ditentukan ini diwarisi oleh sumber daya yang dibuat oleh tumpukan. Namun, untuk instans Amazon EC2 dalam grup Auto Scaling, `AWS::AutoScaling::AutoScalingGroup TagProperty`perlu diatur dalam definisi grup Auto Scaling di template Anda. AWS CloudFormation Atau, jika Anda menggunakan Template Peluncuran EC2 dengan grup Auto Scaling maka Anda dapat menentukan tag dalam definisinya. Dianjurkan untuk menggunakan Template Peluncuran EC2 dengan grup Auto Scaling atau dengan AWS layanan kontainer. Layanan ini dapat membantu memastikan penandaan instans Amazon EC2 Anda secara konsisten dan juga mendukung Auto Scaling di Beberapa Jenis Instans & Opsi Pembelian, yang dapat meningkatkan ketahanan dan mengoptimalkan biaya komputasi Anda.

AWS CloudFormationHooks menyediakan pengembang dengan sarana untuk menjaga aspek-aspek kunci dari aplikasi mereka konsisten dengan standar organisasi mereka. Hooks dapat dikonfigurasi untuk memberikan peringatan atau mencegah penyebaran. Fitur ini paling cocok untuk memeriksa elemen konfigurasi utama dalam template Anda, seperti apakah grup Auto Scaling dikonfigurasi untuk menerapkan tag yang ditentukan pelanggan ke semua instans Amazon EC2 yang akan diluncurkan, atau untuk memastikan bahwa semua bucket Amazon S3 dibuat dengan pengaturan enkripsi yang diperlukan. Dalam kedua kasus tersebut, evaluasi kepatuhan ini didorong ke proses penerapan sebelumnya dengan AWS CloudFormation kait sebelum penerapan.

AWS CloudFormationmenyediakan kemampuan untuk mendeteksi ketika sumber daya (lihat Sumber daya yang mendukung deteksi drift) yang disediakan dari templat telah dimodifikasi dan sumber daya tidak lagi cocok dengan konfigurasi templat yang diharapkan. Ini dikenal sebagai drift. Jika Anda menggunakan otomatisasi untuk menerapkan tag ke sumber daya yang dikelola melalui IAc, maka Anda memodifikasinya, memperkenalkan drift. Saat menggunakan IAc, saat ini disarankan untuk mengelola persyaratan penandaan apa pun sebagai bagian dari templat IAc, menerapkan AWS CloudFormation kait, dan menerbitkan kumpulan aturan AWS CloudFormation Guard yang dapat digunakan oleh otomatisasi.

Sumber daya terkelola saluran CI/CD

Ketika kematangan beban kerja meningkat, kemungkinan teknik seperti integrasi berkelanjutan dan penerapan berkelanjutan (CI/CD) diadopsi. Teknik-teknik ini membantu mengurangi risiko penyebaran dengan membuatnya lebih mudah untuk menerapkan perubahan kecil lebih sering

dengan peningkatan otomatisasi pengujian. Strategi observabilitas yang mendeteksi perilaku tak terduga yang diperkenalkan oleh penerapan dapat secara otomatis memutar kembali penerapan dengan dampak pengguna minimal. Ketika Anda mencapai tahap penyebaran puluhan kali sehari, menerapkan tag secara surut tidak lagi praktis. Semuanya perlu dinyatakan sebagai kode atau konfigurasi, dikontrol versi, dan, sedapat mungkin, diuji dan dievaluasi sebelum penerapan ke dalam produksi. Dalam model pengembangan dan operasi (DevOps) gabungan, banyak praktik membahas pertimbangan operasional sebagai kode dan memvalidasinya di awal siklus hidup penerapan.

Idealnya, Anda ingin mendorong pemeriksaan ini sedini mungkin dalam proses (seperti yang ditunjukkan dengan AWS CloudFormation kait), sehingga Anda dapat yakin bahwa AWS CloudFormation template Anda memenuhi kebijakan Anda sebelum mereka meninggalkan mesin pengembang.

AWS CloudFormationGuard 2.0 menyediakan sarana untuk menulis aturan kepatuhan preventif untuk apa pun yang dapat Anda definisikan. CloudFormation Template divalidasi terhadap aturan di lingkungan pengembangan. Jelas, fitur ini memiliki berbagai aplikasi, tetapi dalam whitepaper ini, kita hanya akan melihat beberapa contoh yang akan memastikan selalu `AWS::AutoScaling::AutoScalingGroup TagProperty`digunakan.

Berikut ini adalah contoh aturan CloudFormation Guard:

```
let all_asgs = Resources.*[ Type == 'AWS::AutoScaling::AutoScalingGroup' ]

rule tags_asg_automation_EnvironmentId when %all_asgs !empty {
    let required_tags = %all_asgs.Properties.Tags.*[
        Key == 'example-inc:automation:EnvironmentId' ]
    %required_tags[*] {
        PropagateAtLaunch == 'true'
        Value IN ['Prod', 'Dev', 'Test', 'Sandbox']
        <<Tag must have a permitted value
          Tag must have PropagateAtLaunch set to 'true'>>
    }
}

rule tags_asg_costAllocation_CostCenter when %all_asgs !empty {
    let required_tags = %all_asgs.Properties.Tags.*[
        Key == 'example-inc:cost-allocation:CostCenter' ]
    %required_tags[*] {
        PropagateAtLaunch == 'true'
        Value == /^123-/
        <<Tag must have a permitted value
```

```
            Tag must have PropagateAtLaunch set to 'true'>>
    }
}
```

Dalam contoh kode, kami memfilter template untuk semua sumber daya yang berjenis`AutoScalingGroup`, dan kemudian memiliki dua aturan:

- **`tags_asg_automation_EnvironmentId`**- Memeriksa bahwa tag dengan kunci ini ada, memiliki nilai dalam daftar nilai yang diizinkan, dan `PropagateAtLaunch` itu diatur ke `true`
- **`tags_asg_costAllocation_CostCenter`**- Memeriksa bahwa tag ada dengan kunci ini, memiliki nilai yang dimulai dengan nilai awalan yang ditentukan, dan yang `PropagateAtLaunch` diatur ke `true`

Penegakan

Seperti dijelaskan sebelumnya, Resource Groups & Tag Editor menyediakan sarana untuk mengidentifikasi di mana sumber daya Anda gagal memenuhi persyaratan penandaan yang ditentukan dalam kebijakan tag yang diterapkan pada OU organisasi. Mengakses alat konsol Resource Groups & Tag Editor dari dalam akun anggota Organisasi menunjukkan kepada Anda kebijakan yang berlaku untuk akun tersebut dan sumber daya dalam akun yang gagal memenuhi persyaratan kebijakan tag. Jika diakses dari akun manajemen (dan jika kebijakan Tag mengaktifkan Akses di layanan di bawahAWS Organizations), Anda dapat melihat kepatuhan kebijakan tag untuk semua akun tertaut di organisasi.

Dalam Kebijakan Tag itu sendiri, Anda dapat mengaktifkan penegakan untuk jenis sumber daya tertentu. Dalam contoh kebijakan berikut, kami telah menambahkan penegakan hukum sehingga semua jenis sumber daya `ec2:instance` dan `ec2:volume` harus mematuhi kebijakan. Ada beberapa batasan yang diketahui, seperti harus ada tag pada sumber daya agar dapat dievaluasi oleh kebijakan tag. Lihat Sumber daya yang mendukung penegakan dengan kebijakan tag untuk daftar.

ExampleInc-Alokasi biaya.json

Berikut ini adalah contoh kebijakan tag yang melaporkan dan/atau memberlakukan tag Alokasi Biaya:

```
{
  "tags": {
    "example-inc:cost-allocation:ApplicationId": {
      "tag_key": {
```

```
        "@@assign": "example-inc:cost-allocation:ApplicationId"
      },
      "tag_value": {
        "@@assign": [
          "DataLakeX",
          "RetailSiteX"
        ]
      },
      "enforced_for": {
        "@@assign": [
          "ec2:instance",
          "ec2:volume"
        ]
      }
    },
    "example-inc:cost-allocation:BusinessUnitId": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:BusinessUnitId"
      },
      "tag_value": {
        "@@assign": [
          "Architecture",
          "DevOps",
          "FinanceDataLakeX"
        ]
      },
      "enforced_for": {
        "@@assign": [
          "ec2:instance",
          "ec2:volume"
        ]
      }
    },
    "example-inc:cost-allocation:CostCenter": {
      "tag_key": {
        "@@assign": "example-inc:cost-allocation:CostCenter"
      },
      "tag_value": {
        "@@assign": [
          "123-*"
        ]
      },
      "enforced_for": {
        "@@assign": [
```

```
          "ec2:instance",
          "ec2:volume"
        ]
      }
    }
  }
}
```

AWS Config (**`required_tag`**)

AWS ConfigAdalah layanan yang mengizinkan Anda menilai, mengaudit, dan mengevaluasi konfigurasi AWS sumber daya (lihat Jenis sumber daya yang didukung oleh AWS Config). Dalam kasus penandaan, kita dapat menggunakannya untuk mengidentifikasi sumber daya yang tidak memiliki tag dengan kunci tertentu, menggunakan `required_tags` aturan (lihat tipe Sumber daya yang didukung oleh required_tags). Dari contoh sebelumnya, kami mungkin menguji keberadaan kunci pada semua instans Amazon EC2. Dalam kasus di mana kunci tidak ada, instance akan terdaftar sebagai tidak sesuai. AWS CloudFormationTemplate ini menjelaskan AWS Config Aturan untuk menguji keberadaan kunci wajib yang dijelaskan dalam tabel, di bucket Amazon S3, instans Amazon EC2, dan volume Amazon EBS.

```
Resources:
 MandatoryTags:
   Type: AWS::Config::ConfigRule
   Properties:
     ConfigRuleName: ExampleIncMandatoryTags
     Description: These tags should be in place
     InputParameters:
       tag1Key: example-inc:cost-allocation:ApplicationId
       tag2Key: example-inc:cost-allocation:BusinessUnitId
       tag3Key: example-inc:cost-allocation:CostCenter
       tag4Key: example-inc:automation:EnvironmentId
     Scope:
       ComplianceResourceTypes:
       - "AWS::S3::Bucket"
       - "AWS::EC2::Instance"
       - "AWS::EC2::Volume"
     Source:
       Owner: AWS
       SourceIdentifier: REQUIRED_TAGS
```

Untuk lingkungan di mana sumber daya dikelola secara manual, AWS Config aturan dapat ditingkatkan untuk secara otomatis menambahkan kunci tag yang hilang ke sumber daya menggunakan remediasi otomatis melalui AWS Lambda fungsi. Meskipun ini berfungsi dengan baik untuk beban kerja statis, ini semakin kurang efektif saat Anda mulai mengelola sumber daya Anda melalui IAc dan pipeline penerapan.

AWS OrganizationsKebijakan kontrol layanan (SCP) adalah jenis kebijakan organisasi yang dapat Anda gunakan untuk mengelola izin di organisasi Anda. SCP menawarkan kontrol pusat atas izin maksimum yang tersedia untuk semua akun di organisasi atau unit organisasi (OU) Anda. SCP hanya memengaruhi pengguna dan peran yang dikelola oleh akun yang merupakan bagian dari organisasi. Meskipun mereka tidak memengaruhi sumber daya secara langsung, mereka membatasi izin pengguna dan peran yang mencakup izin untuk tindakan penandaan. Sehubungan dengan penandaan, SCP dapat memberikan perincian tambahan untuk penegakan tag serta kebijakan tag apa yang dapat disediakan.

Dalam contoh berikut, kebijakan akan menolak `ec2:RunInstances` permintaan di mana `example-inc:cost-allocation:CostCenter` tag tidak ada.

Berikut ini adalah penolakan SCP:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyRunInstanceWithNoCostCenterTag",
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:*:*:instance/"
      ],
      "Condition": {
        "Null": {
          "aws:RequestTag/example-inc:cost-allocation:CostCenter": "true"
        }
      }
    }
  ]
}
```

Tidak mungkin untuk mengambil kebijakan kontrol layanan efektif yang berlaku untuk akun tertaut berdasarkan desain. Jika Anda menerapkan penandaan dengan SCP, dokumentasi harus tersedia

bagi pengembang sehingga mereka dapat memastikan sumber daya mereka memenuhi kebijakan yang telah diterapkan ke akun mereka. Memberikan akses hanya baca ke CloudTrail acara dalam akun mereka dapat mendukung pengembang dalam tugas debugging ketika sumber daya mereka gagal mematuhi.

Mengukur efektivitas penandaan dan mendorong peningkatan

Setelah Anda menerapkan strategi penandaan, penting untuk mengukur efektivitasnya terhadap kasus penggunaan target. Ukuran efektivitas akan bervariasi menurut kasus penggunaan. Misalnya:

- Atribusi biaya - Anda dapat mengukur cakupan penandaan sumber daya berdasarkan pengeluaran menggunakan alat seperti AWS Cost Exploreratau Laporan AWSBiaya dan Penggunaan. Misalnya, Anda dapat melacak persentase sumber daya yang ditandai atau tidak ditandai yang menghasilkan biaya, terutama memantau kunci tag tertentu.
- Otomatisasi - Anda mungkin ingin mengaudit jika hasil yang diinginkan telah tercapai. Misalnya, apakah instans Amazon EC2 non-produksi ditangguhkan di luar jam kerja, waktu mulai dan berhenti instans audit.

Resource Groups & Tag Editor dalam akun manajemen menyediakan kemampuan tambahan untuk menganalisis kepatuhan kebijakan tag untuk semua akun tertaut di organisasi Anda.

Berdasarkan hasil pengukuran efektivitas penandaan Anda, identifikasi apakah ada perbaikan atau perubahan yang diperlukan dalam salah satu langkah seperti definisi kasus penggunaan, penerapan skema penandaan atau penegakan hukum. Buat perubahan yang diperlukan dan ulangi siklus sampai efektivitas yang diinginkan tercapai. Dalam contoh dengan atribusi biaya, Anda dapat melihat peningkatan persentase.

Karena pengembang dan operatorlah yang perlu melakukan penandaan sumber daya yang sebenarnya, sangat penting untuk meminta mereka mengambil kepemilikan. Ini bukan satu-satunya tanggung jawab tambahan yang biasanya diasumsikan tim dalam perjalanan AWS adopsi mereka. Peningkatan tanggung jawab untuk keamanan dan biaya pengembangan dan pengoperasian aplikasi mereka juga penting. Organizations sering menggunakan tujuan dan target sebagai sarana untuk memotivasi penerapan praktik baru, jadi ini juga dapat diterapkan di sini.

Mengamankan kasus penggunaan yang digunakan

Topik

Tag untuk alokasi biaya dan manajemen keuangan

Salah satu kasus penggunaan penandaan pertama yang sering ditangani organisasi adalah visibilitas dan pengelolaan biaya dan penggunaan. Biasanya ada beberapa alasan untuk ini:

- Ini biasanya skenario yang dipahami dengan baik dan persyaratan sudah diketahui. Misalnya, tim keuangan ingin melihat total biaya beban kerja dan infrastruktur yang menjangkau berbagai layanan, fitur, akun, atau tim. Salah satu cara untuk mencapai visibilitas biaya ini adalah melalui penandaan sumber daya yang konsisten.
- Tag dan nilainya didefinisikan dengan jelas. Biasanya, mekanisme alokasi biaya sudah ada dalam sistem keuangan organisasi, misalnya, melacak berdasarkan pusat biaya, unit bisnis, tim, atau fungsi organisasi.
- Pengembalian investasi yang cepat dan dapat dibuktikan. Dimungkinkan untuk melacak tren pengoptimalan biaya dari waktu ke waktu ketika sumber daya ditandai secara konsisten, misalnya, untuk sumber daya yang berukuran benar, diskalakan otomatis, atau diletakkan pada jadwal.

Memahami bagaimana Anda mengeluarkan biaya AWS memungkinkan Anda membuat keputusan keuangan yang tepat. Mengetahui di mana Anda telah mengeluarkan biaya di tingkat sumber daya, beban kerja, tim, atau organisasi meningkatkan pemahaman Anda tentang nilai yang diberikan pada tingkat yang berlaku jika dibandingkan dengan hasil bisnis yang dicapai.

Tim teknik mungkin tidak memiliki pengalaman dengan manajemen keuangan sumber daya mereka. Melampirkan seseorang dengan keterampilan khusus dalam manajemen AWS keuangan yang dapat melatih tim teknik dan pengembangan pada dasar-dasar manajemen AWS keuangan dan menciptakan hubungan antara keuangan dan teknik untuk menumbuhkan budaya FinOps akan membantu mencapai hasil yang terukur untuk bisnis dan mendorong tim untuk membangun dengan biaya dalam pikiran. Menetapkan praktik keuangan yang baik tercakup secara mendalam oleh Pilar

Optimalisasi Biaya dari Kerangka Well-Architected, tetapi kami akan menyentuh beberapa prinsip dasar dalam whitepaper ini.

Tag alokasi biaya

Alokasi biaya mengacu pada penugasan atau distribusi biaya yang dikeluarkan kepada pengguna atau penerima manfaat dari biaya tersebut setelah proses yang ditentukan. Dalam konteks whitepaper ini, kami membagi alokasi biaya menjadi dua jenis: showback dan chargeback.

Alat dan mekanisme showback membantu meningkatkan kesadaran biaya. Chargeback membantu pemulihan biaya dan mendorong pemberdayaan kesadaran biaya. Showback adalah tentang presentasi, perhitungan, dan pelaporan biaya yang dikeluarkan oleh entitas tertentu, seperti unit bisnis, aplikasi, pengguna, atau pusat biaya. Misalnya: “tim teknik infrastruktur bertanggung jawab atas $X AWS pengeluaran bulan lalu”. Chargeback adalah tentang pengisian aktual biaya yang dikeluarkan kepada entitas tersebut melalui proses akuntansi internal organisasi, seperti sistem keuangan atau voucher jurnal. Misalnya: “$ X dikurangkan dari AWS anggaran tim teknik infrastruktur.” Dalam kedua kasus tersebut, menandai sumber daya dengan tepat dapat membantu mengalokasikan biaya ke entitas, satu-satunya perbedaan adalah apakah seseorang diharapkan melakukan pembayaran atau tidak.

Tata kelola keuangan organisasi Anda mungkin memerlukan akuntansi transparan biaya yang dikeluarkan di aplikasi, unit bisnis, pusat biaya, dan tingkat tim. Melakukan atribusi biaya yang didukung oleh Tag Alokasi Biaya memberi Anda data yang diperlukan untuk secara akurat mengaitkan biaya yang dikeluarkan oleh entitas dari sumber daya yang ditandai dengan tepat.

- Akuntabilitas — Memastikan bahwa biaya dialokasikan kepada mereka yang bertanggung jawab atas penggunaan sumber daya. Satu titik layanan atau kelompok dapat bertanggung jawab atas ulasan dan pelaporan pengeluaran.
- Transparansi keuangan — Tunjukkan pandangan yang jelas tentang alokasi uang tunai terhadap TI dengan membuat dasbor yang efektif dan analisis biaya yang berarti untuk kepemimpinan.
- Investasi TI yang diinformasikan — Lacak ROI berdasarkan proyek, aplikasi, atau lini bisnis, dan memberdayakan tim untuk membuat keputusan bisnis yang lebih baik, misalnya, mengalokasikan lebih banyak dana untuk aplikasi yang menghasilkan pendapatan.

Singkatnya, tag alokasi biaya dapat membantu memberi tahu Anda:

- Siapa yang memiliki pengeluaran dan bertanggung jawab untuk mengoptimalkannya?

- Beban kerja, aplikasi, atau produk apa yang mengeluarkan pengeluaran? Lingkungan atau panggung yang mana?
- Area pengeluaran apa yang tumbuh paling cepat?
- Berapa banyak pengeluaran yang dapat dikurangkan dari AWS anggaran berdasarkan tren masa lalu?
- Apa dampak dari upaya optimalisasi biaya dalam beban kerja, aplikasi, atau produk tertentu?

Mengaktifkan tag sumber daya untuk alokasi biaya membantu dengan definisi praktik pengukuran dalam organisasi yang dapat digunakan untuk memberikan visibilitas AWS penggunaan yang meningkatkan transparansi ke dalam akuntabilitas untuk pengeluaran. Ini juga berfokus pada menciptakan tingkat granularitas yang sesuai sehubungan dengan visibilitas biaya dan penggunaan dan mempengaruhi perilaku konsumsi cloud melalui pelaporan alokasi biaya dan pelacakan KPI.

Membangun strategi alokasi biaya

Mendefinisikan dan menerapkan model alokasi biaya

Buat struktur akun dan biaya untuk sumber daya yang digunakan. AWS Menetapkan hubungan antara biaya dari AWS pengeluaran, bagaimana biaya itu dikeluarkan, dan siapa atau apa yang mengeluarkan biaya tersebut. Struktur biaya umum didasarkan padaAWS Organizations,Akun AWS, lingkungan, dan entitas dalam organisasi Anda, seperti lini bisnis atau beban kerja. Struktur biaya dapat didasarkan pada beberapa atribut untuk memungkinkan pemeriksaan biaya dengan cara yang berbeda atau pada tingkat granularitas yang berbeda seperti menggulung biaya beban kerja individu ke lini bisnis yang mereka layani.

Ketika memilih struktur biaya yang selaras dengan hasil yang diinginkan, evaluasi mekanisme alokasi biaya pada kemudahan implementasi versus akurasi yang diinginkan. Ini mungkin termasuk pertimbangan dalam hal akuntabilitas, ketersediaan perkakas, dan perubahan budaya. Tiga model alokasi biaya populer yang biasanya dimulai oleh AWS pelanggan adalah:

- Berbasis akun — Model ini membutuhkan upaya paling sedikit dan memberikan akurasi tinggi untuk showback dan chargeback, dan cocok untuk organisasi yang memiliki struktur akun yang ditentukan (dan konsisten dengan rekomendasi dari whitepaper Pengorganisasian AWS Lingkungan Anda Menggunakan Beberapa Akun). Ini memberikan visibilitas biaya yang jelas berdasarkan per akun. Untuk visibilitas dan alokasi biaya, Anda dapat menggunakan AWS Cost Explorer, Laporan Biaya dan Penggunaan, serta AWSAnggaran untuk pemantauan dan pelacakan biaya. Alat-alat ini menyediakan opsi penyaringan dan pengelompokan berdasarkan. Akun AWS

Dari perspektif alokasi biaya, model ini tidak harus bergantung pada penandaan yang akurat dari sumber daya individu.

- Unit Bisnis atau Berbasis Tim — Biaya yang dialokasikan untuk tim, unit bisnis, atau organisasi dalam suatu perusahaan. Model ini membutuhkan upaya yang moderat, memberikan akurasi tinggi untuk showback dan chargeback, dan cocok untuk organisasi yang memiliki struktur akun yang ditentukan (biasanya menggunakanAWS Organizations), dengan pemisahan antara berbagai tim, aplikasi, dan jenis beban kerja. Ini memberikan visibilitas biaya yang jelas di seluruh tim dan aplikasi, dan sebagai manfaat tambahan mengurangi risiko mencapai kuota AWS layanan dalam satu. Akun AWS Misalnya, setiap tim mungkin memiliki lima akun (`prod`,,`staging`,`test`,`sandbox`)`dev`, dan tidak ada dua tim dan aplikasi yang akan berbagi akun yang sama. Dengan struktur tersebut AWSCost Categories kemudian akan menyediakan fungsionalitas untuk mengelompokkan akun atau tag lain (“meta-tagging”) ke dalam kategori, yang dapat dilacak dalam alat yang disebutkan dalam contoh sebelumnya. Penting untuk dicatat bahwa AWS Organizations memungkinkan penandaan akun dan unit organisasi (OU), namun tag ini tidak akan berlaku untuk alokasi biaya dan pelaporan penagihan (yaitu, Anda tidak dapat mengelompokkan atau memfilter biaya Anda AWS Cost Explorer berdasarkan OU). AWS Cost Categories harus digunakan untuk tujuan ini.
- Berbasis Tag — Model ini membutuhkan lebih banyak usaha dibandingkan dengan dua sebelumnya dan akan memberikan akurasi tinggi untuk showback dan chargeback tergantung pada persyaratan dan tujuan akhir. Meskipun kami sangat menyarankan agar Anda mengadopsi praktik yang diuraikan dalam Mengatur AWS Lingkungan Anda Menggunakan beberapa akun whitepaper, secara realistis pelanggan sering menemukan diri mereka dengan struktur akun campuran dan kompleks yang membutuhkan waktu untuk bermigrasi. Menerapkan strategi penandaan yang ketat dan efektif adalah kunci dalam skenario ini, diikuti dengan mengaktifkan tag yang relevan untuk alokasi biaya di konsol Billing and Cost Management (AWS Organizationsdalam, tag dapat diaktifkan untuk alokasi biaya hanya dari akun Management Payer). Setelah tag diaktifkan untuk alokasi biaya, maka alat untuk visibilitas biaya dan alokasi yang disebutkan dalam metode sebelumnya dapat digunakan untuk showback dan chargeback. Perhatikan bahwa tag alokasi biaya tidak retrospektif, dan hanya akan muncul di pelaporan penagihan dan alat pelacak biaya setelah diaktifkan untuk alokasi biaya.

Untuk meringkas, jika Anda perlu melacak biaya berdasarkan unit bisnis, Anda dapat menggunakan AWSCost Categories untuk mengelompokkan akun tertaut dalam AWS Organisasi yang sesuai dan melihat pengelompokan ini dalam laporan penagihan. Saat membuat akun terpisah untuk lingkungan produksi dan non-produksi, Anda juga dapat memfilter biaya yang terkait dengan lingkungan di alat seperti AWS Cost Explorer, atau melacak biaya tersebut menggunakan AWSAnggaran. Terakhir, jika

kasus penggunaan Anda memerlukan pelacakan biaya yang lebih terperinci, misalnya, berdasarkan beban kerja atau aplikasi individual, Anda dapat menandai sumber daya dalam akun tersebut, mengaktifkan kunci tag tersebut untuk alokasi biaya pada akun manajemen, dan kemudian memfilter biaya tersebut berdasarkan kunci tag di alat pelaporan penagihan.

Mengamankan proses pelaporan dan pemantauan biaya

Mulailah dengan mengidentifikasi jenis-jenis biaya yang penting bagi pemangku kepentingan internal (misalnya, pengeluaran harian, biaya berdasarkan akun, biaya oleh X, biaya diamortisasi). Dengan demikian, Anda dapat mengurangi risiko anggaran yang terkait dengan pengeluaran tak terduga atau anomali lebih cepat daripada menunggu faktur yang diselesaikan. AWS Tag menyediakan atribusi yang memungkinkan skenario pelaporan ini. Wawasan yang diperoleh dari pelaporan dapat menginformasikan tindakan Anda untuk mengurangi dampak dari pengeluaran anomali dan tak terduga pada anggaran keuangan. Ketika ada lonjakan biaya yang tidak terduga, penting untuk mengevaluasi apakah ada lonjakan tak terduga dalam nilai yang disampaikan sehingga Anda dapat menentukan apakah dan tindakan apa yang diperlukan.

Saat mengembangkan strategi penandaan untuk mendukung alokasi biaya, ingatlah elemen-elemen berikut:

- AWS Organizations- Alokasi biaya dalam beberapa akun dapat dilakukan oleh akun, grup akun, atau grup tag yang dibuat untuk sumber daya pada akun tersebut. Tag yang dibuat untuk sumber daya yang berada di akun individu AWS Organizations dapat digunakan untuk alokasi biaya hanya dari akun manajemen.
- AWSAkun - Alokasi biaya dalam satu Akun AWS dapat dilakukan dengan dimensi tambahan seperti layanan atau wilayah. Dimungkinkan untuk menandai sumber daya lebih lanjut dalam akun dan bekerja dengan grup tag sumber daya tersebut.
- Tag Alokasi Biaya - Tag buatan pengguna dan tag yang AWS dihasilkan dapat diaktifkan untuk alokasi biaya, jika perlu. Mengaktifkan tag untuk alokasi biaya di konsol penagihan (akun manajemen diAWS Organizations) membantu dengan showback dan chargeback.
- Cost Categories - AWS Cost Categories memungkinkan pengelompokan akun dan pengelompokan tag (“meta-tagging”) dalam suatu AWS Organisasi, yang selanjutnya menyediakan kemampuan untuk menganalisis biaya yang terkait dengan kategori ini melalui alat sepertiAWS Cost Explorer, AWS Anggaran dan Laporan Biaya dan Penggunaan. AWS

Melakukan showback dan chargeback untuk unit bisnis, tim, atau organisasi dalam perusahaan

Atribut biaya menggunakan proses alokasi biaya Anda didukung oleh struktur biaya dan tag alokasi biaya Anda. Tag dapat digunakan untuk memberikan showback kepada tim yang tidak secara langsung bertanggung jawab untuk membayar biaya tetapi bertanggung jawab atas biaya tersebut. Pendekatan ini memberikan kesadaran akan kontribusi mereka untuk dibelanjakan dan bagaimana biaya tersebut dikeluarkan. Lakukan tolak bayar kepada tim yang secara langsung bertanggung jawab atas biaya untuk memulihkan biaya sumber daya yang telah mereka konsumsi, dan untuk memberi mereka kesadaran akan biaya-biaya tersebut dan bagaimana biaya-biaya tersebut dikeluarkan.

Mengukur dan mengedarkan efisiensi atau nilai KPI

Setujui serangkaian biaya unit atau metrik KPI untuk mengukur dampak investasi manajemen keuangan cloud Anda. Latihan ini menciptakan bahasa umum di seluruh pemangku kepentingan teknologi dan bisnis, dan menceritakan kisah berbasis sains, daripada cerita yang hanya berfokus pada pengeluaran agregat absolut. Untuk informasi tambahan, periksa blog ini yang membahas bagaimana metrik unit dapat membantu menciptakan keselarasan antara fungsi bisnis.

Mengamankan pengeluaran yang tidak dapat dialokasikan

Tergantung pada praktik akuntansi organisasi, jenis biaya yang berbeda mungkin memerlukan perlakuan yang berbeda. Identifikasi sumber daya atau kategori biaya yang tidak dapat ditandai. Bergantung pada layanan yang digunakan dan yang direncanakan untuk digunakan, sepakati mekanisme tentang cara memperlakukan dan mengukur pengeluaran yang tidak dapat dialokasikan tersebut. Misalnya, periksa daftar sumber daya yang didukung oleh AWS Resource Groupsdan Editor Tag di Panduan Pengguna AWS Resource Groups dan Tag.

Contoh umum dari kategori biaya yang tidak dapat ditandai adalah beberapa biaya untuk diskon berbasis komitmen seperti Instans Cadangan (RI) dan Savings Plans (SP). Meskipun biaya berlangganan dan biaya SP dan RI yang tidak digunakan tidak dapat ditandai sebelum muncul di alat pelaporan penagihan, Anda dapat melacak bagaimana diskon RI dan SP berlaku untuk akun, sumber daya, dan tag mereka AWS Organizations setelah fakta. Misalnya, Anda dapat melihat biaya yang diamortisasi, kelompokkan yang dibelanjakan berdasarkan kunci tag yang relevan, dan terapkan filter yang relevan dengan kasus penggunaan Anda. AWS Cost Explorer Dalam Laporan AWS Biaya dan Penggunaan (CUR), Anda dapat memfilter baris yang sesuai dengan penggunaan yang dicakup oleh diskon RI dan SP (baca lebih lanjut di bagian kasus penggunaan dokumentasi CUR) dan pilih kolom

yang hanya relevan bagi Anda. Setiap kunci tag yang diaktifkan untuk alokasi biaya akan disajikan dalam kolom terpisah di akhir laporan CUR, mirip dengan bagaimana itu disajikan dalam laporan penagihan lama lainnya, seperti laporan alokasi biaya bulanan. Untuk referensi tambahan, periksa AWSWell-Architected Labs untuk contoh mendapatkan wawasan biaya dan penggunaan dari data CUR.

Pelaporan

Selain AWS alat yang tersedia untuk membantu showback dan chargeback, ada berbagai solusi AWS buatan dan pihak ketiga lainnya yang dapat membantu memantau biaya sumber daya yang ditandai, dan mengukur efektivitas strategi penandaan. Bergantung pada persyaratan dan tujuan akhir organisasi, seseorang dapat menginvestasikan waktu dan sumber daya untuk membangun solusi khusus atau membeli alat yang disediakan oleh salah satu Mitra Kompetensi Alat AWS Cloud Manajemen. Jika Anda memutuskan untuk membuat alat alokasi biaya kebenaran sumber tunggal Anda sendiri dengan parameter terkontrol yang relevan untuk bisnis, Laporan AWS Biaya dan Penggunaan (CUR) menyediakan data biaya dan penggunaan yang paling rinci dan memungkinkan pembuatan dasbor pengoptimalan yang disesuaikan, memungkinkan pemfilteran dan pengelompokan berdasarkan akun, layanan, kategori biaya, tag alokasi biaya, dan beberapa dimensi lainnya. Di antara solusi berbasis CURE yang dikembangkan oleh AWS yang dapat digunakan sebagai salah satu alat ini, periksa Cloud Intelligence Dashboards di situs web Well-Architected AWS Labs.

Tag untuk operasi dan dukungan

AWSLingkungan akan memiliki banyak akun, sumber daya, dan beban kerja dengan persyaratan operasional yang berbeda. Tag dapat digunakan untuk memberikan konteks dan panduan untuk mendukung tim operasi untuk meningkatkan manajemen layanan Anda. Tag juga dapat digunakan untuk memberikan transparansi tata kelola operasional dari sumber daya yang dikelola.

Beberapa faktor utama yang mendorong definisi tag operasional yang konsisten adalah:

- Untuk memfilter sumber daya selama aktivitas infrastruktur otomatis. Misalnya, saat menerapkan, memperbarui, atau menghapus sumber daya. Lain adalah penskalaan sumber daya untuk optimalisasi biaya dan pengurangan penggunaan di luar jam. Lihat solusi Penjadwal AWS Instance untuk contoh kerja.
- Mengidentifikasi sumber daya yang terisolasi atau mencela. Sumber daya yang telah melampaui umur yang ditentukan atau telah ditandai untuk diisolasi oleh mekanisme internal harus ditandai

dengan tepat untuk membantu personel pendukung dalam penyelidikan mereka. Sumber daya yang tidak digunakan lagi harus ditandai sebelum isolasi, pengarsipan, dan penghapusan.

- Persyaratan Support untuk sekelompok sumber daya. Sumber daya seringkali memiliki persyaratan dukungan yang berbeda, misalnya, persyaratan ini dapat dinegosiasikan antar tim atau ditetapkan sebagai bagian dari kekritisan aplikasi. Panduan lebih lanjut tentang model operasi dapat ditemukan di Pilar Keunggulan Operasional.
- Meningkatkan proses manajemen insiden. Dengan menandai sumber daya dengan tag yang menawarkan transparansi yang lebih besar dalam proses manajemen insiden, tim pendukung dan insinyur serta tim Manajemen Insiden Utama (MIM) dapat mengelola acara secara lebih efektif.
- Cadangan. Tag juga dapat digunakan untuk mengidentifikasi frekuensi sumber daya Anda perlu dicadangkan, dan ke mana salinan cadangan harus pergi atau ke mana harus memulihkan cadangan. Panduan preskriptif untuk pendekatan Backup dan pemulihan pada. AWS
- Menambal. Menambal instance yang dapat berubah berjalan sangat AWS penting dalam strategi penambalan menyeluruh Anda dan untuk menambal kerentanan zero-day. Panduan yang lebih dalam tentang strategi penambalan yang lebih luas dapat ditemukan dalam panduan preskriptif. Penambalan kerentanan zero-day dibahas di blog ini.
- Observabilitas operasional. Memiliki strategi KPI operasional yang diterjemahkan ke tag sumber daya akan membantu tim operasi untuk melacak dengan lebih baik apakah target terpenuhi untuk meningkatkan persyaratan bisnis. Mengembangkan strategi KPI adalah topik yang terpisah, tetapi cenderung difokuskan pada bisnis yang beroperasi dalam keadaan mapan atau di mana mengukur dampak dan hasil perubahan. Dashboard KPI (AWSWell-Architected labs) dan Operations KPI Workshop (layanan proaktif Dukungan AWS Perusahaan) keduanya mengukur kinerja dalam kondisi mapan. Artikel blog strategi AWS perusahaan Mengukur Keberhasilan Transformasi Anda, mengeksplorasi pengukuran KPI untuk program transformasi, seperti modernisasi TI atau migrasi dari tempat ke tempat. AWS

Kegiatan infrastruktur otomatis

Tag dapat digunakan dalam berbagai aktivitas otomatisasi saat mengelola infrastruktur. Penggunaan AWSSystems Manager, misalnya, akan memungkinkan Anda mengelola otomatisasi dan runbook pada sumber daya yang ditentukan oleh pasangan nilai kunci yang ditentukan yang Anda buat. Untuk node yang dikelola, Anda dapat menentukan set tag untuk melacak atau menargetkan node dengan menggunakan sistem operasi dan lingkungan. Anda kemudian dapat menjalankan skrip pembaruan untuk semua node dalam grup atau meninjau status node tersebut. Sumber Daya Systems Manager juga dapat ditandai untuk lebih menyempurnakan dan melacak aktivitas otomatis Anda.

Mengotomatiskan siklus hidup awal dan berhenti sumber daya lingkungan dapat memberikan pengurangan biaya yang signifikan bagi organisasi mana pun. Penjadwal instans AWS aktif adalah contoh solusi yang dapat memulai dan menghentikan instans Amazon EC2 dan Amazon RDS saat tidak diperlukan. Misalnya, lingkungan pengembang yang menggunakan Amazon EC2 atau Amazon RDS instans yang tidak harus berjalan pada akhir pekan tidak menggunakan potensi penghematan biaya yang dapat ditawarkan oleh penutupan instans tersebut. Dengan menganalisis kebutuhan tim dan lingkungan mereka, dan menandai sumber daya ini dengan benar untuk mengotomatiskan manajemen mereka, Anda dapat memanfaatkan anggaran Anda secara efektif.

Contoh tag jadwal yang digunakan oleh penjadwal instans pada instans Amazon EC2:

```
{
    "Tags": [
        {
            "Key": "Schedule",
            "ResourceId": "i-1234567890abcdef8",
            "ResourceType": "instance",
            "Value": "mon-9am-fri-5pm"
        }
    ]
}
```

Siklus hidup beban kerja

Meninjau keakuratan data operasional pendukung. Pastikan bahwa ada tinjauan berkala dari tag yang terkait dengan siklus hidup beban kerja Anda, dan bahwa pemangku kepentingan yang sesuai terlibat dalam ulasan ini.

Tabel 7 — Tinjau tag operasional sebagai bagian dari siklus hidup beban kerja

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
Pemilik Akun	`example-inc:account-owner:owner`	Pemilik akun dan itu berisi sumber daya.	`ops-center` , `dev-ops`, `app-team`
Ulasan Pemilik Akun	`example-inc:account-owner:review`	Tinjau detail kepemilikan akun yang mutakhir dan benar.	<review date in the correct format defined

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
			in your tagging library>
Pemilik data	`example-inc:data-owner:owner`	Pemilik data dari akun yang berada di data.	`bi-team`, `logistics` , `security`
Ulasan Pemilik Data	`example-inc:data-owner:review`	Tinjauan detail kepemilikan data yang mutakhir dan benar.	<review date in the correct format defined in your tagging library>

Menetapkan tag untuk menangguhkan akun sebelum bermigrasi ke OU yang ditangguhkan

Sebelum menangguhkan akun dan pindah ke OU yang ditangguhkan seperti yang dijelaskan dalam whitepaper Mengatur AWS Lingkungan Anda Menggunakan Beberapa Akun, tag harus ditambahkan ke akun untuk membantu penelusuran internal dan audit siklus hidup akun Anda. Misalnya, URL relatif atau referensi tiket pada sistem tiket ITSM organisasi, yang menunjukkan jejak audit untuk aplikasi yang ditangguhkan.

Tabel 8 - Tambahkan tag operasional saat siklus hidup beban kerja memasuki tahap baru

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
Pemilik Akun	`example-inc:account-owner:owner`	Pemilik akun dan itu berisi sumber daya.	`ops-center` , `dev-ops`, `app-team`
Pemilik data	`example-inc:data-owner:owner`	Pemilik data dari akun yang berada di data.	`bi-team`, `logistics` , `security`
Tanggal Ditangguhkan	`example-inc:suspension:date`	Tanggal akun ditangguhkan	<suspended date in the correct format

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
			defined in your tagging library>
Persetujuan untuk penangguhan	`example-inc:suspension:approval`	Tautan ke persetujuan penangguhan akun	`workload/deprecation`

Manajemen insiden

Tag dapat memainkan peran penting dalam semua fase manajemen insiden mulai dari pencatatan insiden, prioritas, investigasi, komunikasi, resolusi hingga penutupan.

Tag dapat merinci di mana insiden harus dicatat, tim atau tim yang harus diberitahu tentang insiden tersebut, dan prioritas eskalasi yang ditentukan. Penting untuk diingat bahwa tag tidak dienkripsi, jadi pertimbangkan informasi apa yang Anda simpan di dalamnya. Juga, dalam organisasi, tim, dan jalur pelaporan, tanggung jawab berubah, jadi pertimbangkan untuk menyimpan tautan ke portal aman di mana informasi ini dapat dikelola dengan lebih efektif. Tag ini tidak harus eksklusif. Misalnya, ID aplikasi dapat digunakan untuk mencari jalur eskalasi di portal manajemen layanan TI. Pastikan jelas dalam definisi operasional Anda bahwa tag ini digunakan untuk berbagai tujuan.

Tag persyaratan operasional dapat dirinci juga, untuk membantu manajer insiden dan personel operasi lebih menyempurnakan tujuan mereka dalam menanggapi insiden atau peristiwa.

Tautan relatif (ke URL basis sistem pengetahuan) untuk runbook dan buku pedoman dapat dimasukkan sebagai tag untuk membantu tim yang merespons dalam mengidentifikasi proses, prosedur, dan dokumentasi yang sesuai.

Tabel 9 - Gunakan tag operasional untuk menginformasikan manajemen insiden

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
Manajemen Insiden	`example-inc:incident-management:escalationlog`	Sistem yang digunakan oleh tim pendukung untuk mencatat insiden	`jira`, `servicenow` , `zendesk`

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
Manajemen Insiden	`example-inc:incident-management:escalationpath`	Jalur eskalasi	`ops-center` , `dev-ops`, `app-team`
Alokasi Biaya dan Manajemen Insiden	`example-inc:cost-allocation:CostCenter`	Pantau biaya berdasarkan pusat biaya. Ini adalah contoh tag penggunaan ganda di mana pusat biaya digunakan sebagai kode aplikasi untuk pencatatan insiden	`123-*`
Jadwal Backup	`example-inc:backup:schedule`	Jadwal Backup Sumber Daya	`Daily`
Playbook/Manajemen Insiden	`example-inc:incident-management:playbook`	Buku pedoman yang didokumentasikan	`webapp/incident/playbook`

Menambal

Organizations dapat mengotomatiskan strategi patching mereka untuk lingkungan komputasi yang dapat berubah dan menjaga instance yang dapat berubah sejalan dengan baseline patch yang ditentukan dari lingkungan aplikasi tersebut dengan menggunakan Systems Manager Patch Manager dan. AWS AWS Lambda Strategi penandaan untuk instance yang dapat berubah dalam lingkungan ini dapat dikelola dengan menetapkan instance tersebut ke Grup Patch dan Windows Pemeliharaan. Lihat contoh berikut untuk pemisahan Dev → Test → Prod. AWSpanduan preskriptif tersedia untuk manajemen tambalan instance yang bisa berubah.

Tabel 10 - Tag operasional dapat spesifik lingkungan

Pengembangan	Pementasan	Produksi

{
"Tags": [
{
"Key": "Maintenance
Window",
"ResourceId":
"i-012345678ab9ab1
11",
"ResourceType":
"instance",
"Value": "cron(30 23 ?
* TUE#1 *)"
},
{
"Key": "Name",
"ResourceId":
"i-012345678ab9ab2
22",
"ResourceType":
"instance",
"Value": "WEBAPP"
},
{
"Key": "Patch Group",
"ResourceId":
"i-012345678ab9ab3
33",
"ResourceType":
"instance",
"Value": "WEBAPP-DEV-
AL2"
}
]
}
	{
	"Tags": [
	{
	"Key": "Maintenance
	Window",
	"ResourceId":
	"i-012345678ab9ab4
	44",
	"ResourceType":
	"instance",
	"Value": "cron(30 23 ?
	* TUE#2 *)"
	},
	{
	"Key": "Name",
	"ResourceId":
	"i-012345678ab9ab5
	55",
	"ResourceType":
	"instance",
	"Value": "WEBAPP"
	},
	{
	"Key": "Patch Group",
	"ResourceId":
	"i-012345678ab9ab6
	66",
	"ResourceType":
	"instance",
	"Value": "WEBAPP-TEST-
	AL2"
	}
	]
	}
		{
		"Tags": [
		{
		"Key": "Maintenance
		Window",
		"ResourceId":
		"i-012345678ab9ab7
		77",
		"ResourceType":
		"instance",
		"Value": "cron(30 23 ?
		* TUE#3 *)"
		},
		{
		"Key": "Name",
		"ResourceId":
		"i-012345678ab9ab8
		88",
		"ResourceType":
		"instance",
		"Value": "WEBAPP"
		},
		{
		"Key": "Patch Group",
		"ResourceId":
		"i-012345678ab9ab9
		99",
		"ResourceType":
		"instance",
		"Value": "WEBAPP-PROD-
		AL2"
		}
		]
		}

Kerentanan zero-day juga dapat dikelola dengan memiliki tag yang ditentukan untuk melengkapi strategi patching Anda. Lihat Hindari kerentanan zero-day dengan patch keamanan pada hari yang sama menggunakan Systems Manager untuk panduan terperinci. AWS

Observabilitas operasional

Observabilitas diperlukan untuk mendapatkan wawasan yang dapat ditindaklanjuti tentang kinerja lingkungan Anda dan membantu Anda mendeteksi dan menyelidiki masalah. Ini juga memiliki tujuan sekunder yang memungkinkan Anda untuk menentukan dan mengukur indikator kinerja utama (KPI) dan tujuan tingkat layanan (SLO) seperti uptime. Bagi sebagian besar organisasi, KPI operasi penting adalah mean time to detect (MTTD) dan mean time to recover (MTTR) dari suatu insiden.

Sepanjang observabilitas, konteks penting, karena data dikumpulkan dan kemudian tag terkait dikumpulkan. Terlepas dari tingkat layanan, aplikasi, atau aplikasi yang Anda fokuskan, Anda dapat memfilter dan menganalisis untuk kumpulan data tertentu. Tag dapat digunakan untuk mengotomatiskan orientasi ke CloudWatch Alarm sehingga tim yang tepat dapat diperingatkan ketika ambang batas metrik tertentu dilanggar. Misalnya, kunci tag `example-inc:ops:alarm-tag` dan nilai di atasnya dapat menunjukkan pembuatan CloudWatch Alarm. Solusi yang menunjukkan hal ini dijelaskan dalam Gunakan tag untuk membuat dan memelihara CloudWatch alarm Amazon untuk instans Amazon EC2.

Memiliki terlalu banyak alarm yang dikonfigurasi dapat dengan mudah membuat badai peringatan — ketika sejumlah besar alarm atau notifikasi dengan cepat membanjiri operator dan mengurangi efektivitas keseluruhan mereka sementara operator secara manual memprioritaskan dan memprioritaskan alarm individu. Konteks tambahan untuk alarm dapat disediakan dalam bentuk tag, yang berarti bahwa aturan dapat didefinisikan dalam Amazon EventBridge untuk membantu memastikan bahwa fokus diberikan pada masalah hulu daripada dependensi hilir.

Peran operasi di samping DevOps sering diabaikan, tetapi bagi banyak organisasi, tim operasi pusat masih memberikan respons pertama yang kritis di luar jam kerja normal. (Rincian lebih lanjut dapat ditemukan tentang model ini di whitepaper Operational Excellence.) Tidak seperti DevOps tim yang memiliki beban kerja, mereka biasanya tidak memiliki kedalaman pengetahuan yang sama, sehingga konteks yang disediakan tag dalam dasbor dan peringatan, dapat mengarahkannya ke runbook yang benar untuk masalah ini, atau memulai runbook otomatis (lihat posting blog Mengotomatiskan Alarm Amazon dengan). CloudWatch AWS Systems Manager

Tag untuk keamanan data, manajemen risiko, dan kontrol akses

Organizations memiliki berbagai kebutuhan dan kewajiban yang harus dipenuhi mengenai penanganan penyimpanan dan pemrosesan data yang tepat. Klasifikasi data merupakan prekursor penting untuk beberapa kasus penggunaan, seperti kontrol akses, retensi data, analisis data, dan kepatuhan.

Keamanan data dan manajemen risiko

Dalam suatu AWS lingkungan, Anda mungkin akan memiliki akun dengan kepatuhan dan persyaratan keamanan yang berbeda. Misalnya, Anda mungkin memiliki kotak pasir pengembang, dan akun yang menampung lingkungan produksi untuk beban kerja yang sangat diatur, seperti memproses pembayaran. Dengan mengisolasinya ke akun yang berbeda, Anda dapat menerapkan kontrol keamanan yang berbeda, membatasi akses ke data sensitif, dan mengurangi ruang lingkup audit untuk beban kerja yang diatur.

Mengadopsi standar tunggal untuk semua beban kerja dapat menimbulkan tantangan. Meskipun banyak kontrol berlaku sama di seluruh lingkungan, beberapa kontrol berlebihan atau tidak relevan untuk akun yang tidak perlu memenuhi kerangka peraturan tertentu, dan akun di mana tidak ada data pribadi yang dapat diidentifikasi (misalnya, kotak pasir pengembang, atau akun pengembangan beban kerja). Hal ini biasanya mengarah pada temuan keamanan positif palsu yang harus diprioritaskan dan ditutup tanpa tindakan, yang menghilangkan upaya dari temuan yang harus diselidiki.

Tabel 11 — Contoh keamanan data dan tag manajemen risiko

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
Manajemen insiden	`example-inc:incident-management:escalationlog`	Sistem yang digunakan oleh tim pendukung untuk mencatat insiden	`jira`, `servicenow` , `zendesk`
Manajemen insiden	`example-inc:incident-management:escalationpath`	Jalur eskalasi	`ops-center` , `dev-ops`, `app-team`
Klasifikasi data	`example-inc:data:classification`	Klasifikasi data untuk kepatuhan dan tata kelola	`Public`, `Private`, `Confidential` , `Restricted`
Kepatuhan	`example-inc:compliance:framework`	Mengidentifikasi kerangka kerja kepatuhan yang	`PCI-DSS`, `HIPAA`

Kasus penggunaan	Tag kunci	Dasar Pemikiran	Nilai contoh
		dikenakan beban kerja	

Mengelola kontrol yang berbeda secara manual di seluruh AWS lingkungan memakan waktu dan rawan kesalahan. Langkah selanjutnya adalah mengotomatiskan penerapan kontrol keamanan yang sesuai, dan mengonfigurasi inspeksi sumber daya, berdasarkan klasifikasi akun itu. Dengan menerapkan tag ke akun dan sumber daya di dalamnya, penyebaran kontrol dapat diotomatisasi dan dikonfigurasi dengan tepat untuk beban kerja.

Contoh:

Beban kerja mencakup bucket Amazon S3 dengan `example-inc:data:classification` tag dengan nilainya. `Private` AWS ConfigAturan otomatisasi alat keamanan menerapkan`s3-bucket-public-read-prohibited`, yang memeriksa pengaturan Blokir Akses Publik Amazon S3 bucket, kebijakan bucket, dan daftar kontrol akses bucket (ACL), yang mengonfirmasi konfigurasi bucket sesuai untuk klasifikasi datanya. Untuk memastikan konten bucket konsisten dengan klasifikasi, Amazon Macie dapat dikonfigurasi untuk memeriksa informasi identitas pribadi (PII). Blog Menggunakan Amazon Macie untuk Memvalidasi Klasifikasi Data Bucket S3 mengeksplorasi pola ini secara lebih mendalam.

Lingkungan peraturan tertentu, seperti asuransi dan perawatan kesehatan, mungkin tunduk pada kebijakan penyimpanan data wajib. Penyimpanan data menggunakan tag, dikombinasikan dengan kebijakan Siklus Hidup Amazon S3, dapat menjadi cara yang efektif dan sederhana untuk menjangkau transisi objek ke tingkat penyimpanan yang berbeda. Aturan Siklus Hidup Amazon S3 juga dapat digunakan untuk menghapus objek kedaluwarsa setelah periode penahanan wajib berakhir. Lihat Sederhanakan siklus hidup data Anda dengan menggunakan tag objek dengan Siklus Hidup Amazon S3 untuk panduan mendalam tentang proses ini.

Selain itu, ketika melakukan triaging atau menangani temuan keamanan, tag dapat memberikan penyelidik konteks penting yang membantu memenuhi syarat risiko, dan membantu dalam melibatkan tim yang sesuai untuk menyelidiki atau mengurangi temuan tersebut.

Tag untuk manajemen identitas dan kontrol akses

Saat mengelola kontrol akses di seluruh AWS lingkungan denganAWS IAM Identity Center, tag dapat mengaktifkan beberapa pola untuk penskalaan. Ada beberapa pola delegasi yang dapat diterapkan,

beberapa didasarkan pada penandaan. Kami akan mengatasinya secara individual dan memberikan tautan ke bacaan lebih lanjut tentang masing-masing.

ABAC untuk sumber daya individu

Pengguna Pusat Identitas IAM dan peran IAM mendukung kontrol akses berbasis atribut (ABAC), yang memungkinkan Anda untuk menentukan akses ke operasi dan sumber daya berdasarkan tag. ABAC membantu mengurangi kebutuhan untuk memperbarui kebijakan izin dan membantu Anda mendasarkan akses dari atribut karyawan dari direktori perusahaan Anda. Jika Anda sudah menggunakan strategi multi-akun, ABAC dapat digunakan selain kontrol akses berbasis peran (RBAC) untuk menyediakan beberapa tim yang beroperasi di akun yang sama akses granular ke sumber daya yang berbeda. Misalnya, pengguna IAM Identity Center atau peran IAM dapat menyertakan kondisi untuk membatasi akses ke instans Amazon EC2 tertentu yang jika tidak harus dicantumkan secara eksplisit di setiap kebijakan untuk mengaksesnya.

Karena model otorisasi ABAC bergantung pada tag untuk akses ke operasi dan sumber daya, penting untuk menyediakan pagar pembatas untuk mencegah akses yang tidak diinginkan. SCP dapat digunakan untuk melindungi tag di seluruh organisasi Anda dengan hanya mengizinkan tag dimodifikasi dalam kondisi tertentu. Blog Mengamankan tag sumber daya yang digunakan untuk otorisasi dengan menggunakan kebijakan kontrol layanan di AWS Organizations dan batas izin untuk entitas IAM memberikan informasi tentang cara mengimplementasikannya.

Jika instans Amazon EC2 berumur panjang digunakan untuk mendukung praktik operasi yang lebih tradisional maka pendekatan ini dapat digunakan, blog Konfigurasikan IAM Identity Center ABAC untuk instans Amazon EC2 dan Systems Manager Session Manager membahas bentuk kontrol akses berbasis atribut ini secara lebih rinci. Seperti disebutkan sebelumnya, tidak semua jenis sumber daya mendukung penandaan, dan yang melakukannya, tidak semua mendukung penegakan menggunakan kebijakan tag, jadi sebaiknya evaluasi ini sebelum mulai menerapkan strategi ini pada fileAkun AWS.

Untuk mempelajari tentang layanan yang mendukung ABAC, lihat AWSlayanan yang bekerja dengan IAM.

Kesimpulan

AWSsumber daya dapat ditandai untuk berbagai tujuan, mulai dari menerapkan strategi alokasi biaya hingga mendukung otomatisasi atau otorisasi akses ke sumber daya. AWS Menerapkan strategi penandaan dapat menjadi tantangan bagi beberapa organisasi, karena jumlah kelompok pemangku kepentingan yang terlibat dan pertimbangan seperti sumber data dan tata kelola tag.

Dalam whitepaper ini, kami telah menguraikan rekomendasi mengenai merancang dan menerapkan strategi penandaan dalam organisasi berdasarkan praktik operasional, kasus penggunaan yang ditentukan, pemangku kepentingan yang terlibat dalam proses, dan alat dan layanan yang disediakan oleh. AWS Ketika datang ke strategi penandaan, ini adalah proses iterasi dan peningkatan, di mana Anda memulai dari prioritas langsung Anda, mengidentifikasi kasus penggunaan yang relevan di seluruh organisasi Anda, dan kemudian menerapkan dan menumbuhkan skema penandaan yang Anda butuhkan, sambil terus mengukur dan meningkatkan efektivitas. Kami telah menunjukkan bahwa seperangkat tag yang terdefinisi dengan baik dalam organisasi Anda akan memungkinkan Anda untuk menghubungkan AWS penggunaan dan konsumsi dengan tim yang bertanggung jawab atas sumber daya dan tujuan bisnis di mana mereka ada, untuk menyelaraskan dengan strategi dan nilai organisasi.

Kontributor

Kontributor dokumen ini meliputi:

- Chris Pate, Manajer Akun Teknis Spesialis Sr, Amazon Web Services
- Vijay Shekhar Rao, Pimpinan Dukungan Perusahaan, Amazon Web Services
- Nataliya Godunok, Manajer Akun Teknis Spesialis Sr, Amazon Web Services
- Yogish Kutkunje Pai, Arsitek Solusi Sr, Amazon Internet Services Private Limited
- Jamie Ibbs, Manajer Akun Teknis Spesialis Sr, Amazon Web Services

Bacaan lebih lanjut

Untuk informasi lebih lanjut, lihat

- AWSRe:invent 2020: Bekerja mundur: Pendekatan Amazon terhadap inovasi
- AWSPanduan Preskriptif: Penambalan otomatis untuk instans yang dapat berubah di cloud hybrid menggunakan Systems Manager AWS
- AWSPusat Arsitektur

AWSWell-Architected

- AWSKerangka Well-Architected
- Pilar Keunggulan Operasional - Kerangka AWS Well-Architected
- Rencana Pemulihan Bencana (DR) - Pilar Keandalan AWS yang Dirancang dengan Baik
- Pilar Pengoptimalan Biaya - Kerangka AWS Well-Architected
- AWSWell-Architected Labs: AWS Aktifkan Tag Alokasi Biaya yang Dihasilkan
- AWSWell-Architected Labs: Kebijakan Tag
- AWSWell-Architected LabsAWS: CUR Query Library

AWSblog

- AWS HealthSadar - Sesuaikan AWS Health Peringatan untuk Akun Organisasi dan Pribadi AWS
- Cara Menandai Sumber Daya Amazon EC2 Secara Otomatis dalam Menanggapi Peristiwa API
- AWSTag Alokasi Biaya yang Ditentukan Pengguna
- Penandaan Biaya dan Pelaporan dengan AWS Organizations
- Menambal instans Windows EC2 Anda menggunakan Patch Manager AWS Systems Manager
- Hindari kerentanan zero-day dengan patching keamanan hari yang sama menggunakan AWS Systems Manager

AWSdokumentasi

- Menggunakan Tag Alokasi Biaya - AWS Billing and Cost Management dan Manajemen Biaya dan Manajemen Biaya

- Apa itu AWS Biaya dan Laporan Penggunaan
- AWS Resource GroupsReferensi API
- Bagaimana saya bisa menggunakan tag kebijakan IAM untuk membatasi bagaimana instans EC2 atau volume EBS dapat dibuat?
- Model pembaruan yang dapat diubah vs tidak dapat diubah

Lainnya

- Bryar, C. dan Carr, B. (2021). Bekerja Mundur: Wawasan, Cerita, dan Rahasia dari Dalam Amazon. London Macmillan.
- AWS CloudFormationPenjaga (GitHub)

Revisi dokumen

Untuk mendapatkan notifikasi tentang pembaruan whitepaper ini, berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Pembaruan kecil	Pembaruan untuk manajemen identitas	Maret 30, 2023
Revisi kecil	Referensi yang diperbarui di ABAC untuk sumber daya individu.	Februari 24, 2023
Revisi kecil	Panduan yang diperbarui untuk menyelaraskan dengan praktik terbaik IAM. Untuk informasi selengkapnya, lihat Praktik terbaik keamanan di IAM.	Februari 6, 2023
Revisi besar	Menambahkan referensi yang lebih spesifik untuk jenis sumber daya yang didukung oleh AWS Config aturan`required_tags` .	Januari 18, 2023
Revisi besar	Diperbarui untuk memasukkan praktik dan kemampuan layanan terbaru, terutama di bidang identitas.	September 29, 2022
Pembaruan kecil	Pemformatan tabel tetap dalam versi PDF.	April 25, 2022
Revisi besar	Struktur dokumen yang diperbarui dan bagian Strategi Penandaan dan Kasus Penggunaan yang diperluas.	April 22, 2022

	Menambahkan lebih banyak panduan preskriptif berdasarkan alat, teknik, dan sumber daya terbaru yang tersedia.	
Publikasi awal	Whitepaper pertama kali diterbitkan.	Desember 1, 2018

> **Note**
>
> Untuk berlangganan pembaruan RSS, Anda harus mengaktifkan plugin RSS untuk browser yang Anda gunakan.

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. AWSproduk 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.

AWSGlosarium

Untuk AWS terminologi terbaru, lihat AWSglosarium di Referensi. Glosarium AWS

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