Membangun pagar pembatas dan pemantauan untuk presigned URLs

AWS Bimbingan Preskriptif

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

AWS Bimbingan Preskriptif: Membangun pagar pembatas dan pemantauan untuk presigned URLs

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

Membangun pagar pembatas dan pemantauan untuk URL yang telah ditetapkan sebelumnya

Ryan Baker, Amazon Web Services ()AWS

Juli 2024 (sejarah dokumen)

Keamanan adalah perhatian penting bagi semua perusahaan dan pilar utama dalam AWS Well-Architected Framework. Sebagai insinyur keamanan, Anda akan ingin menerapkan pagar pembatas administratif yang selaras dengan persyaratan kontrol organisasi. Dalam AWS Well-Architected Framework, pagar pembatas menentukan batas-batas yang membatasi aktivitas.

Panduan ini memberikan informasi latar belakang dan praktik terbaik untuk menggunakan URL yang telah ditetapkan sebelumnya, yang digunakan dengan objek Amazon Simple Storage Service (Amazon S3). URL yang telah ditetapkan sebelumnya memungkinkan pengguna atau aplikasi yang memiliki akses ke kredensyal yang valid untuk menghasilkan permintaan yang ditandatangani sebelumnya dan diterima hingga waktu kedaluwarsa yang ditentukan. Kasus penggunaan umum untuk URL yang telah ditetapkan sebelumnya adalah memperluas akses ke objek atau sumber daya dengan membagikan permintaan ini. Permintaan presigned bersama dihasilkan oleh sistem atau pengguna yang memiliki hak untuk melakukan permintaan tertentu, dan kemudian dapat dikirim ke sistem atau pengguna lain untuk memperluas kemampuan untuk melakukan permintaan yang sama.

Dalam panduan ini, Anda akan belajar:

- Konsep URL yang telah ditetapkan sebelumnya
- Kasus penggunaan untuk URL yang telah ditetapkan sebelumnya
- Pagar pembatas yang direkomendasikan dan opsional
- Opsi pemantauan
- Contoh cara Layanan AWS menggunakan URL yang telah ditentukan sebelumnya

Audiens yang dituju

Panduan ini ditujukan untuk arsitek dan insinyur keamanan yang bertanggung jawab untuk menerapkan kontrol keamanan di AWS Cloud.

Tujuan

Sebagai insinyur keamanan, Anda ingin mengetahui bagaimana pembuat solusi menerapkan keamanan dan jenis akses yang dimiliki pengguna akhir Anda. Panduan ini mencakup satu jenis akses, URL yang telah ditetapkan sebelumnya, yang sering digunakan dengan Amazon S3. URL yang telah ditetapkan sebelumnya memberi pembangun opsi untuk menjembatani mekanisme otentikasi secara efisien.

Di Amazon S3, URL yang telah ditetapkan sebelumnya mewakili kategori permintaan yang unik. Insinyur keamanan dapat memantau dan mengelola permintaan ini untuk memastikan bahwa mereka hanya digunakan jika sesuai dan perlu. Tujuan dari panduan ini adalah untuk membantu insinyur keamanan memberikan jenis pengawasan tingkat tinggi ini.

Setelah membaca panduan ini, Anda harus memahami apa itu URL yang telah ditentukan sebelumnya, kapan biasanya digunakan, dan motivasi penggunaannya.

Prasyarat

Jika perusahaan Anda belum menetapkan kebijakan keamanan, tujuan kontrol, atau standar, seperti yang dijelaskan dalam panduan Menerapkan kontrol keamanan di AWS, sebaiknya Anda menyelesaikan tugas tata kelola tersebut sebelum melanjutkan dengan panduan ini.

Sebelum memulai, Anda juga harus terbiasa dengan praktik terbaik yang direkomendasikan dan opsional untuk kontrol dan pemantauan. Untuk informasi selengkapnya, lihat:

- Kebijakan kontrol layanan (AWS Organizations dokumentasi)
- Kebijakan bucket untuk Amazon S3 (dokumentasi Amazon S3)
- Permintaan logging dengan pencatatan akses server (dokumentasi Amazon S3)
- Pencatatan panggilan API Amazon S3 menggunakan AWS CloudTrail(dokumentasi Amazon S3)

Ikhtisar URL yang telah ditetapkan sebelumnya

URL presigned adalah jenis permintaan HTTP yang dikenali oleh layanan AWS Identity and Access Management (IAM). Yang membedakan jenis permintaan ini dari semua AWS permintaan lainnya adalah parameter kueri X-Amz-Expires. Seperti permintaan otentikasi lainnya, permintaan URL presigned menyertakan tanda tangan. Untuk permintaan URL yang telah ditetapkan sebelumnya, tanda tangan ini dikirimkan`X-Amz-Signature`. Tanda tangan menggunakan operasi kriptografi Signature Version 4 untuk menyandikan semua parameter permintaan lainnya.

> **Catatan**
>
> - Signature Version 2 saat ini sedang dalam proses tidak digunakan lagi, tetapi masih didukung di beberapa. Wilayah AWS Panduan ini berlaku untuk penandatanganan Signature Version 4.
> - Layanan penerima dapat memproses header yang tidak ditandatangani, tetapi dukungan untuk opsi itu terbatas dan ditargetkan, sejalan dengan praktik terbaik. Kecuali dinyatakan lain, asumsikan bahwa semua header harus ditandatangani agar permintaan diterima.

`X-Amz-Expires`Parameter memungkinkan tanda tangan diproses sebagai valid dengan penyimpangan yang lebih besar dari waktu tanggal yang dikodekan. Aspek lain dari validitas tanda tangan masih dievaluasi. Kredensi penandatanganan, jika sementara, tidak boleh kedaluwarsa pada saat tanda tangan diproses. Kredensi penandatanganan harus dilampirkan pada kepala sekolah IAM yang memiliki otorisasi yang memadai pada saat pemrosesan.

URL yang telah ditetapkan sebelumnya adalah bagian dari permintaan yang telah ditetapkan sebelumnya

URL presigned bukan satu-satunya metode untuk menandatangani permintaan untuk masa depan. Amazon S3 juga mendukung permintaan POST, yang biasanya juga sudah ditentukan sebelumnya. Tanda tangan POST yang telah ditetapkan sebelumnya memungkinkan unggahan yang mematuhi kebijakan yang ditandatangani dan memiliki tanggal kedaluwarsa yang disematkan dalam kebijakan tersebut.

Tanda tangan untuk permintaan dapat diberi tanggal di masa mendatang, meskipun ini jarang terjadi. Selama kredensyal yang mendasarinya valid, algoritma tanda tangan tidak melarang penanggalan

masa depan. Namun, permintaan ini tidak dapat berhasil diproses sampai jendela waktu yang valid, yang membuat kencan future tidak praktis untuk sebagian besar kasus penggunaan.

Apa yang diizinkan oleh permintaan yang telah ditetapkan sebelumnya?

Permintaan presigned hanya dapat mengizinkan tindakan yang diizinkan oleh kredensyal yang digunakan untuk menandatangani permintaan. Jika kredensyal secara implisit atau eksplisit menolak tindakan yang ditentukan oleh permintaan yang telah ditetapkan sebelumnya, permintaan yang telah ditetapkan sebelumnya akan ditolak saat dikirim. Ini berlaku untuk yang berikut:

- Kebijakan sesi yang terkait dengan kredensialnya
- Kebijakan yang terkait dengan kepala sekolah yang terkait dengan kredensialnya
- Kebijakan sumber daya yang memengaruhi sesi atau prinsipal
- Kebijakan pengendalian layanan yang memengaruhi sesi atau prinsipal

Motivasi untuk menggunakan permintaan yang telah ditentukan sebelumnya

Sebagai insinyur keamanan, Anda harus menyadari apa yang memotivasi pembuat solusi untuk menggunakan URL yang telah ditetapkan sebelumnya. Memahami apa yang diperlukan dan apa yang opsional akan membantu Anda berkomunikasi dengan pembuat solusi. Motivasi mungkin termasuk yang berikut:

- Untuk mendukung mekanisme otentikasi non-IAM sambil memanfaatkan skalabilitas di Amazon S3. Motivasi inti adalah berkomunikasi langsung dengan Amazon S3 untuk mendapatkan manfaat dari skalabilitas bawaan yang disediakan oleh layanan ini. Tanpa komunikasi langsung ini, solusi perlu mendukung beban dari transmisi ulang byte yang dikirim dan panggilan. PutObjectGetObject Bergantung pada beban total, persyaratan ini menambahkan tantangan penskalaan yang mungkin ingin dihindari oleh pembuat solusi.

  Cara lain untuk berkomunikasi langsung dengan Amazon S3, seperti menggunakan kredensyal sementara AWS Security Token Service di AWS STS() atau tanda tangan Versi Tanda Tangan 4 di luar URL, mungkin tidak sesuai untuk kasus penggunaan Anda. Amazon S3 mengidentifikasi pengguna melalui AWS kredensyal, sedangkan permintaan presigned mengandaikan identifikasi melalui mekanisme selain kredensyal. AWS Menjembatani perbedaan ini sambil mempertahankan komunikasi langsung untuk data dapat dicapai melalui permintaan yang telah ditentukan sebelumnya.

- Untuk mendapatkan manfaat dari pemahaman asli browser tentang URL. URL dipahami oleh browser, sedangkan AWS STS kredensyal dan tanda tangan Versi Tanda Tangan 4 tidak. Ini bermanfaat saat berintegrasi dengan solusi berbasis browser. Solusi alternatif memerlukan lebih banyak kode, akan menggunakan lebih banyak memori untuk file besar, dan mungkin diperlakukan berbeda dengan ekstensi seperti malware dan pemindai virus.

Perbandingan dengan AWS STS kredensyal sementara

Kredensyal sementara mirip dengan permintaan yang telah ditetapkan sebelumnya. Keduanya kedaluwarsa, memungkinkan pelingkupan akses, dan biasanya digunakan untuk menjembatani kredensyal non-IAM dengan penggunaan yang memerlukan kredensyal AWS.

Anda dapat secara ketat mencakup AWS STS kredensi sementara ke satu objek dan tindakan S3, tetapi ini dapat menghasilkan tantangan penskalaan karena AWS STS API memiliki batasan. (Untuk informasi selengkapnya, lihat artikel Bagaimana cara mengatasi pelambatan API atau kesalahan “Nilai terlampaui” untuk IAM dan di situs web AWS AWS STS re:Post.) Selain itu, setiap kredensi yang dihasilkan memerlukan panggilan AWS STS API, yang menambahkan latensi dan ketergantungan baru yang dapat memengaruhi ketahanan. AWS STS Kredensyal sementara juga memiliki waktu kedaluwarsa minimum 15 menit, sedangkan permintaan yang telah ditentukan sebelumnya dapat mendukung durasi yang lebih pendek. (60 detik praktis mengingat kondisi yang tepat.)

Perbandingan dengan solusi khusus tanda tangan

Satu-satunya komponen rahasia inheren dari permintaan yang telah ditetapkan sebelumnya adalah tanda tangan Versi Tanda Tangan 4. Jika klien mengetahui detail lain dari permintaan dan diberikan tanda tangan yang valid yang cocok dengan detail tersebut, ia dapat mengirim permintaan yang valid. Tanpa tanda tangan yang valid, tidak bisa.

URL yang telah ditetapkan sebelumnya dan solusi khusus tanda tangan serupa secara kriptografis. Namun, solusi khusus tanda tangan memiliki keuntungan praktis seperti kemampuan untuk menggunakan header HTTP alih-alih parameter string kueri untuk mengirimkan tanda tangan (lihat bagian Interaksi dan mitigasi Logging). Administrator juga harus mempertimbangkan bahwa string kueri lebih sering diperlakukan sebagai metadata, sedangkan header lebih jarang diperlakukan seperti itu.

Di sisi lain, AWS SDK memberikan lebih sedikit dukungan untuk menghasilkan dan menggunakan tanda tangan secara langsung. Membangun solusi khusus tanda tangan memerlukan lebih banyak

kode khusus. Dari perspektif praktis, menggunakan pustaka alih-alih kode khusus untuk keamanan adalah praktik terbaik umum, sehingga kode untuk solusi khusus tanda tangan memerlukan pengawasan ekstra.

Solusi khusus tanda tangan tidak menggunakan string `X-Amz-Expires` kueri dan tidak memberikan periode validitas eksplisit. IAM mengelola periode validitas implisit tanda tangan yang tidak memiliki waktu kedaluwarsa eksplisit. Periode implisit tersebut tidak dipublikasikan. Mereka biasanya tidak berubah, tetapi dikelola dengan mempertimbangkan keamanan, jadi Anda tidak boleh bergantung pada periode validitas. Ada tradeoff antara memiliki kontrol eksplisit atas tanggal kedaluwarsa dan meminta IAM mengelola kedaluwarsa.

Sebagai administrator, Anda mungkin lebih memilih solusi khusus tanda tangan. Namun, dalam arti praktis, Anda harus mendukung solusi seperti yang dibangun.

Mengidentifikasi permintaan yang telah ditetapkan sebelumnya

Mengidentifikasi permintaan yang menggunakan URL presigned

Amazon S3 menyediakan dua mekanisme bawaan untuk memantau penggunaan pada tingkat permintaan: log AWS CloudTrail akses server Amazon S3 dan peristiwa data. Kedua mekanisme tersebut dapat mengidentifikasi penggunaan URL yang telah ditentukan sebelumnya.

Untuk memfilter log untuk penggunaan URL yang telah ditetapkan sebelumnya, Anda dapat menggunakan jenis otentikasi. Untuk log akses server, periksa bidang Jenis Otentikasi, yang biasanya diberi nama authtype saat didefinisikan dalam tabel Amazon Athena. Untuk CloudTrail, periksa AuthenticationMethoddi `additionalEventData` lapangan. Dalam kedua kasus, nilai bidang untuk permintaan yang menggunakan URL presigned adalah`QueryString`, sedangkan `AuthHeader` nilai untuk sebagian besar permintaan lainnya.

`QueryString`penggunaan tidak selalu dikaitkan dengan URL yang telah ditetapkan sebelumnya. Untuk membatasi penelusuran Anda hanya menggunakan URL yang telah ditentukan sebelumnya, temukan permintaan yang berisi parameter string kueri. `X-Amz-Expires` Untuk log akses server, periksa request-URI dan cari permintaan yang memiliki `X-Amz-Expires` parameter dalam string kueri. Untuk CloudTrail, periksa `requestParameters` elemen untuk `X-Amz-Expires` elemen.

```
{"Records": [{…, "requestParameters": {…, "X-Amz-Expires": "300"}}, …]}
```

Kueri Athena berikut menerapkan filter ini:

```
SELECT * FROM {athena-table} WHERE
  authtype = 'QueryString' AND
  request_uri LIKE '%X-Amz-Expires=%';
```

Untuk AWS CloudTrail Lake, kueri berikut menerapkan filter ini:

```
SELECT * FROM {data-store-event-id} WHERE
  additionalEventData['AuthenticationMethod'] = 'QueryString' AND
  requestParameters['X-Amz-Expires'] IS NOT NULL
```

Mengidentifikasi jenis permintaan presigned lainnya

Permintaan POST juga memiliki jenis otentikasi unik,`HtmlForm`, di log akses server Amazon S3 dan. CloudTrail Jenis otentikasi ini kurang umum, jadi Anda mungkin tidak menemukan permintaan ini di lingkungan Anda.

Kueri Athena berikut menerapkan filter untuk: `HtmlForm`

```
SELECT * FROM {athena-table} WHERE
  authtype = 'HtmlForm';
```

Untuk CloudTrail Lake, kueri berikut menerapkan filter:

```
SELECT * FROM {data-store-event-id} WHERE
  additionalEventData['AuthenticationMethod'] = 'HtmlForm'
```

Mengidentifikasi pola permintaan

Anda dapat menemukan permintaan yang telah ditentukan sebelumnya dengan menggunakan teknik yang dibahas di bagian sebelumnya. Namun, untuk membuat data itu berguna, Anda akan ingin menemukan pola. `TOP 10`Hasil sederhana untuk kueri Anda mungkin memberikan wawasan, tetapi jika itu tidak cukup, gunakan opsi pengelompokan dalam tabel berikut.

Opsi pengelompokan	Log akses server	CloudTrailDanau	Deskripsi
Agen pengguna	`GROUP BY useragent`	`GROUP BY userAgent`	Opsi pengelompokan ini membantu Anda menemukan sumber dan tujuan permintaan. Agen pengguna disediakan oleh pengguna dan tidak dapat diandalkan sebagai mekanisme otentikasi atau otorisasi. Namun, ini dapat

Opsi pengelompokan	Log akses server	CloudTrailDanau	Deskripsi
			mengungkapkan banyak hal jika Anda mencari pola, karena sebagian besar klien menggunakan string unik yang setidakny a sebagian dapat dibaca manusia.
Pemohon	`GROUP BY requester`	`GROUP BY userIdent ity['arn']`	Opsi pengelomp okan ini membantu menemukan kepala sekolah IAM yang menandatangani permintaan. Jika tujuan Anda adalah untuk memblokir permintaan ini atau membuat pengecual ian untuk permintaa n yang ada, kueri ini memberikan informasi yang cukup untuk tujuan itu. Ketika Anda menggunak an peran sesuai dengan praktik terbaik IAM, peran tersebut memiliki pemilik yang diidentifikasi dengan jelas, dan Anda dapat menggunak an informasi tersebut untuk mengetahui lebih lanjut.

Opsi pengelompokan	Log akses server	CloudTrailDanau	Deskripsi
Alamat IP sumber	`GROUP BY remoteip`	`GROUP BY sourceIPAddress`	Opsi ini dikelompokkan berdasarkan lompatan terjemahan jaringan terakhir sebelum mencapai Amazon S3. • Jika lalu lintas melewati gateway NAT, ini akan menjadi alamat NAT Gateway. • Jika lalu lintas melewati gateway internet, ini akan menjadi alamat IP publik yang mengirim lalu lintas ke gateway internet. • Jika lalu lintas berasal dari luar AWS, ini akan menjadi alamat internet publik yang terkait dengan asal. • Jika melewati titik akhir gateway virtual private cloud (VPC), ini akan menjadi alamat IP instance di VPC.

Opsi pengelompokan	Log akses server	CloudTrailDanau	Deskripsi
			• Jika melewati antarmuka virtual publik (VIF), ini akan menjadi IP lokal pemohon atau perantara apa pun seperti server proxy atau firewall yang hanya mengekspos alamat IP-nya. • Jika melewati titik akhir VPC antarmuka, ini bisa menjadi alamat IP dari sebuah instance di VPC. Ini juga bisa berupa alamat IP dari VPC lain atau jaringan lokal. Seperti halnya VIF publik, ini bisa berupa alamat IP perantara apa pun. Data ini berguna jika tujuan Anda adalah memaksaka n kontrol jaringan. Anda mungkin harus menggabun gkan opsi ini dengan data seperti

Opsi pengelompokan	Log akses server	CloudTrailDanau	Deskripsi
			`endpoint` (untuk log akses server) atau `vpcEndpointId` (untuk CloudTrail Lake) untuk memperjelas sumbernya, karena jaringan yang berbeda mungkin menduplikasi alamat IP pribadi.
Nama ember S3	`GROUP BY bucket_name`	`GROUP BY requestParameters['bucketName']`	Opsi pengelompokan ini membantu menemukan bucket yang menerima permintaan. Ini membantu Anda mengidentifikasi kebutuhan akan pengecualian.

Praktik terbaik untuk menggunakan permintaan yang telah ditetapkan sebelumnya

Bagian ini membahas praktik terbaik untuk menggunakan permintaan yang telah ditetapkan sebelumnya yang harus dipertimbangkan oleh insinyur keamanan. Pedoman tersebut meliputi:

- Praktik terbaik dasar, yang merupakan praktik yang harus diikuti oleh setiap organisasi.
- Pagar pembatas tambahan, yang merupakan praktik yang harus Anda pertimbangkan, tetapi mungkin memutuskan untuk menerapkan sebagian atau dengan pengecualian. Ini dimaksudkan untuk memberikan kontrol dan pertahanan tambahan secara mendalam, tetapi harus diimbangi dengan kompleksitas keseluruhan.
- Interaksi logging, yang mungkin dihasilkan dari perangkat atau layanan yang merupakan bagian dari tanggung jawab Anda atau pelanggan Anda dalam model tanggung jawab bersama. Bagian ini mencakup tindakan pencegahan untuk membatasi informasi yang dapat diakses melalui log.

Praktik terbaik dasar

Praktik terbaik umum yang merupakan kontrol efektif untuk permintaan AWS API lainnya juga berlaku untuk permintaan yang telah ditetapkan sebelumnya. Bagian ini mengulas dua praktik yang paling relevan: hak istimewa terkecil dan batas data. Praktik-praktik ini menciptakan kedalaman kontrol yang diperluas oleh praktik lain.

Menerapkan prinsip hak istimewa paling sedikit

Langkah pertama dalam membatasi penggunaan permintaan presigned adalah membatasi akses ke Amazon S3 secara umum. URL presigned tidak dapat menyediakan akses ke sumber daya yang tidak diberikan kepada prinsipal yang menghasilkan tanda tangan untuk URL yang telah ditetapkan sebelumnya. Juga tidak dapat menyediakan akses ke sumber daya dengan cara yang tidak diberikan kepada kepala sekolah itu. Dengan demikian, menerapkan praktik terbaik untuk memberikan hak istimewa paling sedikit kepada para prinsipal tersebut adalah pagar pembatas yang efektif.

Proses pembuatan URL presigned adalah operasi algoritmik yang didasarkan pada standar yang diterbitkan (Signature Version 4) untuk pembuatan tanda tangan. Oleh karena itu, tidak mungkin untuk menempatkan batasan pada generasi presigned URLs. Namun, agar relevan, URL yang telah ditetapkan sebelumnya harus valid dan menyediakan akses ke sumber daya, sehingga validitas URL yang telah ditetapkan sebelumnya juga merupakan pagar pembatas yang efektif.

Untuk informasi selengkapnya tentang hak istimewa terkecil, lihat Berikan akses hak istimewa paling sedikit di Pilar AWS Well-Architected Framework, Security.

Menerapkan perimeter data

Perpanjangan hak istimewa terkecil adalah mempertahankan perimeter data yang konsisten dengan kebutuhan organisasi Anda. Presigned URLs kompatibel dengan perimeter data. Seperti permintaan lainnya, validitas permintaan URL yang telah ditetapkan sebelumnya dievaluasi pada waktu permintaan. Jika properti jaringan, sumber daya, sesi peran, dan prinsipal berubah, mereka dievaluasi pada saat itu dan dengan menggunakan metode dimana permintaan diterima.

Misalnya, katakanlah layanan yang berjalan di container Amazon Elastic Kubernetes Service (Amazon EKS) menandatangani permintaan. Permintaan tersebut kemudian dikirim dari sistem komputer pribadi pengguna yang terhubung ke internet. Dalam hal ini, SourceIp kondisi aws: mengevaluasi alamat IP publik yang terlihat dari permintaan dari sistem pribadi pengguna, bukan alamat IP layanan di wadah Amazon EKS.

Demikian pula, jika tag prinsipal atau sumber daya berubah sebelum permintaan dikirim, nilai yang diperbarui, bukan asli, akan berlaku untuk permintaan melalui aws: PrincipalTag /tag-key dan aws: ResourceTag /tag-key conditions.

Pagar pembatas tambahan

Ketika permintaan presigned digunakan dengan tepat oleh pembuat solusi dan pengguna, mereka menyediakan mekanisme yang aman untuk memberikan pengguna akses ke data. Selain itu, kemampuan untuk menghasilkan permintaan yang telah ditetapkan sebelumnya tidak memberikan akses kepada prinsipal yang belum mereka miliki.

Dalam konteks itu, apakah kontrol tambahan diperlukan? Pembenaran untuk kontrol tambahan tidak didasarkan pada kebutuhan untuk menolak akses tetapi untuk menyediakan kemampuan untuk memantau, untuk menyetujui penggunaan dan menetapkan batasan, dan untuk mengurangi risiko dari kesalahan pengguna. Dengan cara ini Anda dapat membantu memastikan bahwa penggunaan sesuai dan perlu.

Pagar pembatas berikut membantu Anda dalam tujuan ini. Sebelum mengaktifkan kontrol ini, Anda mungkin ingin menentukan penggunaan yang ada dengan mengidentifikasi permintaan yang telah ditetapkan sebelumnya. Identifikasi ini membantu Anda mempersiapkan dampak pagar pembatas terhadap penggunaan yang ada atau merencanakan pengecualian jika diperlukan.

Pagar pembatas untuk S3: SignatureAge

Salah satu karakteristik yang menentukan dari permintaan yang telah ditetapkan sebelumnya adalah bahwa mereka menggambarkan waktu kedaluwarsa. Tanda tangan untuk permintaan berisi tanggal. Tanggal ini ditransmisikan sebagai parameter string `X-Amz-Date` kueri untuk presigned URLs, dan sebagai Tanggal atau x-amz-date header untuk POST presigned.

Amazon S3 menyediakan kunci kondisi, S3:SignatureAge, yang dapat Anda gunakan untuk membatasi waktu maksimum antara tanggal yang ditandatangani dan kedaluwarsa efektif permintaan. Kondisi ini tidak pernah dapat meningkatkan masa berlaku, tetapi dapat menguranginya.

Dalam kebijakan berikut, kunci `s3:signatureAge` kondisi membatasi permintaan yang telah ditetapkan sebelumnya hingga 15 menit validitas. Contoh berikut semua menggunakan 15 menit untuk membatasi validitas untuk jangka waktu yang sama sebagai dukungan penandatanganan standar.

Pernyataan kedua dari kebijakan tersebut menolak akses Signature Version 2. Versi protokol penandatanganan ini tidak digunakan lagi, tetapi masih didukung di beberapa. Wilayah AWS Kami menyarankan Anda memblokirnya secara eksplisit sebelum sepenuhnya tidak digunakan lagi.

Anda dapat menerapkan kebijakan berikut sebagai kebijakan kontrol AWS Organizations layanan (SCP). Pengguna masih dapat menggunakan permintaan yang telah ditetapkan sebelumnya dan menerapkan solusi yang bergantung pada permintaan tersebut, selama waktu antara pembuatan tanda tangan dan penggunaan kurang dari 15 menit. Bergantung pada implementasinya, batasan ini mungkin tidak berdampak, dapat menyebabkan solusi menjadi tidak dapat digunakan, atau mungkin menyebabkan kegagalan sesekali yang dapat dicoba ulang.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyPresignedOver15Minutes",
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": "*",
      "Condition": {
        "NumericGreaterThan": {
          "s3:signatureAge": "900000"
        }
      }
```

```
    },
    {
      "Sid": "DenySignatureVersion2",
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
           "s3:signatureversion": "AWS"
        }
      }
    }
  ]
}
```

Pengecualian

Jika solusi memerlukan waktu yang lebih lama sebelum kedaluwarsa dan oleh karena itu dipengaruhi oleh kebijakan sebelumnya, kami sarankan Anda menyediakan metode untuk menyetujui pengecualian. Untuk menghindari penghitungan pengecualian dalam SCP, gunakan aws:, seperti dalam kebijakan berikutPrincipalTag, untuk mengelola pengecualian dengan cara yang dapat diskalakan. AWS Contoh lain, seperti contoh kebijakan perimeter data AWS, menggunakan strategi ini.

Jika Anda menerapkan kebijakan pengecualian dengan menggunakan`aws:PrincipalTag`, Anda harus mengontrol akses untuk menyetel tag pada prinsipal. Tag jenis ini dapat datang langsung dari prinsipal dan dapat dikontrol oleh SCP, seperti dalam contoh pengendalian nilai tag mana yang dapat diatur. Tag jenis ini juga dapat berasal dari tag sesi, yang ditetapkan oleh penyedia identitas (iDP) atau saat menggunakan. AWS STS Mengontrol akses ke `aws:PrincipalTag` adalah topik yang kompleks. Namun, organisasi yang memiliki pengalaman dalam menggunakan kontrol akses berbasis atribut (ABAC) akan memiliki pengalaman dan kontrol untuk memungkinkan penggunaan yang tepat `aws:PrincipalTag` untuk kasus penggunaan ini.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyPresignedOver15Minutes",
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": "*",
```

```
      "Condition": {
        "NumericGreaterThan": {
          "s3:signatureAge": "900000"
        },
--- Example exception ---
        "StringNotEquals": {
          "aws:PrincipalTag/long-presigned-allowed": "true"
        }
--- Example exception end ---
      }
    }
  ]
}
```

Kebijakan bucket

Anda dapat menerapkan kebijakan bucket ke semua atau bucket yang dipilih dengan menggunakan kebijakan seperti pada contoh berikut. Tidak seperti SCP, kebijakan bucket juga menargetkan penggunaan utama layanan. Lampiran A tidak mendokumentasikan penggunaan utama layanan yang diharapkan dari permintaan yang telah ditetapkan sebelumnya, tetapi jika Anda ingin menerapkan kontrol untuk membuktikan batas tersebut, kebijakan berikut akan memberikan kontrol tersebut. Selain itu, tidak seperti SCP, kebijakan bucket dapat berlaku untuk prinsipal di akun manajemen Anda. Pengecualian berbasis ABAC bekerja dalam kebijakan bucket dengan cara yang sama seperti SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyPresignedOver15Minutes",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::{bucket-name}/*",
      "Condition": {
        "NumericGreaterThan": {
          "s3:signatureAge": "900000"
        },
--- Example exception ---
        "StringNotEquals": {
          "aws:PrincipalTag/long-presigned-allowed": "true"
        }
```

```
--- Example exception end ---
      }
    }
  ]
}
```

Guardrail untuk S3:AuthType

Presigned URLs menggunakan otentikasi string kueri, dan presigned POSTs selalu menggunakan otentikasi POST. Amazon S3 mendukung penolakan permintaan berdasarkan jenis otentikasi melalui kunci kondisi S3:authType. `REST-QUERY-STRING`adalah `s3:authType` nilai untuk string kueri, dan `POST` merupakan `s3:authType` nilai untuk POST.

Anda dapat menerapkan kebijakan berikut sebagai SCP. Kebijakan ini digunakan `s3:authType` untuk mengizinkan hanya autentikasi berbasis header. Ini juga mengkonfigurasi metode untuk memberikan pengecualian kepada pengguna individu atau peran.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyNonHeaderAuth",
      "Effect": "Deny",
      "Action": "s3:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "s3:authType": "REST-HEADER",
          "aws:PrincipalTag/non-header-auth-allowed": "true"
        }
      }
    }
  ]
}
```

Menolak permintaan berdasarkan jenis otentikasi memengaruhi solusi atau fitur apa pun yang menggunakan jenis otentikasi yang ditolak. Misalnya, menyangkal `REST-QUERY-STRING` mencegah pengguna melakukan unggahan atau unduhan dari konsol Amazon S3. Jika Anda ingin pengguna menggunakan konsol Amazon S3, jangan gunakan pagar pembatas ini, atau buat pengecualian untuk pengguna. Di sisi lain, jika Anda tidak ingin pengguna menggunakan konsol Amazon S3, Anda dapat menolak `REST-QUERY-STRING` untuk pengguna.

Mungkin Anda sudah menolak akses langsung pengguna ke sumber daya Amazon S3. Dalam hal ini, pagar pembatas untuk jenis otentikasi berlebihan. Namun, pernyataan `s3:authType` penolakan menyediakan defense-in-depth utilitas karena implementasi untuk menolak akses langsung biasanya mencakup banyak pernyataan kontrol, beberapa dengan pengecualian.

Peran yang digunakan untuk beban kerja biasanya tidak memerlukan akses ke string kueri atau `POST` otentikasi. Pengecualian adalah peran yang mendukung layanan yang dirancang untuk menggunakan permintaan yang telah ditetapkan sebelumnya. Anda dapat membuat pengecualian khusus untuk peran tersebut.

Anda juga dapat menerapkan kebijakan bucket ke semua atau bucket yang dipilih dengan menggunakan kebijakan seperti berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyNonHeaderAuth",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::{bucket-name}/*",
      "Condition": {
        "StringNotEquals": {
          "s3:authType": "REST-HEADER",
          "aws:PrincipalTag/non-header-auth-allowed": "true"
        }
      }
    }
  ]
}
```

Kebijakan bucket ini memiliki efek menolak penggunaan CopyObjectdan UploadPartCopy APIs untuk membuat salinan lintas wilayah. Replikasi Amazon S3 tidak terpengaruh karena tidak bergantung pada ini. APIs

Jika Anda ingin menggunakan kebijakan bucket seperti kebijakan sebelumnya dan masih mendukung Cross-region CopyObjectatau UploadPartCopyAPI, tambahkan kondisi yang `aws:ViaAWSService` serupa dengan berikut ini:

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyNonHeaderAuth",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::{bucket-name}/*",
      "Condition": {
        "StringNotEquals": {
          "s3:authType": "REST-HEADER",
          "aws:PrincipalTag/non-header-auth-allowed": "true"
        },
        "Bool": {
          "aws:ViaAWSService": "false"
        },
      }
    }
  ]
}
```

Menggabungkan pagar pembatas dan pengecualian untuk pagar pembatas lainnya

Jika Anda tidak berencana untuk menerapkan pagar pembatas secara umum kepada pengguna dan peran Anda, Anda mungkin ingin menerapkannya pada pengecualian pagar pembatas umum lainnya, sehingga pengecualian tersebut tidak mendukung permintaan yang telah ditetapkan sebelumnya.

Jika Anda memiliki batasan jaringan tetapi mengizinkan pengecualian untuk mitra eksternal atau kasus penggunaan khusus, Anda harus memblokir string kueri atau `POST` otentikasi saat pengecualian tersebut diterapkan, kecuali jika secara khusus diidentifikasi sebagai diperlukan.

Batasan untuk S3: SignatureAge

Administrator akan merasa berguna untuk memahami implikasi `s3:signatureAge` lebih lengkap. Setiap permintaan yang ditandatangani termasuk`X-Amz-Date`, yang harus menunjukkan waktu saat ini. Nilai ini diisi oleh klien dan penandatangan permintaan. AWS menolak permintaan yang dianggap memiliki waktu yang tidak valid. Namun, seorang penandatangan dapat menghasilkan tanda tangan terlebih dahulu dengan waktu yang akan datang. Amazon S3 menolak permintaan yang menentukan masa depan jika dikirim terlalu jauh sebelumnya. Namun, jika permintaan tidak dikirim sampai waktu yang ditandatangani ke tanda tangan, tanda tangan dapat dibuat lebih awal dan dikirim kemudian.

`s3:signatureAge`membatasi usia maksimum `X-Amz-Date` dalam tanda tangan hanya untuk permintaan yang telah ditetapkan sebelumnya. Permintaan yang lebih tua dari usia yang ditentukan ditolak, bahkan jika kedaluwarsa `X-Amz-Expires` atau `POST` kebijakan akan menyatakannya valid. `s3:signatureAge`tidak mengubah periode valid untuk permintaan yang tidak menyertakan kedaluwarsa eksplisit. Itu juga tidak mengontrol nilai `X-Amz-Date` yang digunakan klien untuk tanda tangan.

Jika jam sistem salah atau jika klien sengaja melakukan permintaan tanggal di masa depan, waktu yang ditandatangani mungkin bukan waktu tanda tangan dibuat. Ini membatasi seberapa banyak yang `s3:signatureAge` dapat mengontrol solusi. Solusi yang menggunakan waktu saat ini ketika menghasilkan tanda tangan dibatasi dengan cara yang diharapkan: Tanda tangan tetap valid untuk jumlah milidetik yang ditentukan dalam. `s3:signatureAge` Solusi yang tidak menggunakan waktu saat ini akan memiliki batas yang berbeda. Salah satu batasan adalah bahwa kredensil yang digunakan untuk menandatangani tanda tangan harus tetap valid. Sebagai administrator, Anda dapat mengontrol validitas maksimum kredenal sementara yang dikeluarkan. Anda dapat mengizinkan kredensialnya valid hingga 36 jam atau membatasi validitas hingga serendah 15 menit. Berakhirnya kredensi sementara tidak tergantung pada nilai. `X-Amz-Date`

Kredensi permanen tidak memiliki batasan ini. Hanya menggunakan kredensil sementara adalah praktik terbaik, dan Anda dapat secara eksplisit mencabut kredensi permanen apa pun, yang juga akan membatalkan tanda tangan apa pun berdasarkan kredensi tersebut.

Meskipun `s3:signatureAge` diukur dalam milidetik, tidak praktis untuk mengaturnya menjadi kurang dari 60 detik, bahkan jika Anda memiliki jam yang disinkronkan dengan baik dan penggunaan latensi rendah. Pengaturan yang lebih rendah dari 60 detik berisiko menolak permintaan yang valid. Jika Anda mengharapkan penundaan antara pembuatan tanda tangan dan pengiriman permintaan, atau masalah dengan sinkronisasi jam, Anda harus memperhitungkannya dalam pengelolaan. `s3:signatureAge`

Menargetkan ember dalam skala

SCPs dapat digunakan `aws:PrincipalTag` untuk membuat pengecualian bagi pengguna. Anda tidak dapat menggunakan tag pada bucket untuk mengontrol akses melalui `aws:ResourceTag` – hanya tag objek yang digunakan untuk kontrol akses. Umumnya tidak dapat diskalakan untuk menambahkan tag ke setiap objek yang ingin Anda terapkan kontrol ini.

Solusi yang sesuai dengan banyak kasus penggunaan adalah dengan menerapkan kebijakan dan pengecualian di tingkat akun, baik dengan mengubah akun yang diterapkan SCP atau dengan

menggunakan aws:ResourceAccount, aws: ResourceOrgPaths, atau aws: ResourceOrg ID. Misalnya, SCP dapat diterapkan ke satu set akun produksi.

Solusi lain adalah dengan menggunakan AWS Config aturan khusus untuk menerapkan kontrol detektif atau kontrol responsif. Tujuannya adalah agar setiap ember berisi kebijakan ember dengan pagar pembatas yang sesuai. Selain menguji konten kebijakan bucket, AWS Config aturan kustom dapat mengambil tag dari bucket dan mengecualikan bucket dari aturan jika bucket diberi tag dengan nilai tertentu. Jika aturan itu gagal dalam pemeriksaan kepatuhannya, aturan tersebut dapat menandai bucket sebagai tidak sesuai atau meminta perbaikan untuk menambahkan pagar pembatas ke kebijakan bucket.

> **Note**
>
> Anda tidak dapat membatasi konten tag permintaan. PutBucketTagging Untuk mempertahankan kontrol atas bagaimana bucket ditandai, Anda harus membatasi akses ke `PutBucketTagging` dan DeleteBucketTagging.

Interaksi dan mitigasi logging

URL presigned berisi tanda tangan dan dapat digunakan, selama periode sebelum kedaluwarsa, untuk melakukan operasi API tertentu yang ditandatangani. Ini harus diperlakukan sebagai kredensi akses sementara. Tanda tangan harus tetap pribadi hanya untuk pihak yang perlu mengetahuinya. Di sebagian besar lingkungan, ini adalah klien yang mengirim permintaan dan server yang menerimanya. Mengirim tanda tangan sebagai bagian dari sesi HTTPS langsung mempertahankan sifat pribadinya, karena hanya peserta sesi HTTPS yang memiliki visibilitas ke URI yang mentransmisikan tanda tangan.

Untuk presigned URLs, tanda tangan ditransmisikan sebagai parameter string `X-Amz-Signature` kueri. Parameter string kueri adalah komponen dari URI. Risikonya adalah klien dapat mencatat URI dan tanda tangan dengannya. Klien memiliki akses ke seluruh permintaan HTTP dan dapat mencatat bagian mana pun dari permintaan, data, dan header (termasuk header otentikasi). Namun, ini menurut konvensi kurang umum. Pencatatan URI lebih umum dan diperlukan dalam kasus seperti pencatatan akses. Klien harus menggunakan redaksi atau masking untuk menghapus tanda tangan sebelum login. URIs

Di beberapa lingkungan, pengguna mengizinkan perantara (proxy) untuk mendapatkan visibilitas ke sesi HTTPS mereka. Mengaktifkan proxy memerlukan tingkat akses istimewa yang tinggi ke sistem

klien, karena memerlukan konfigurasi dan sertifikat tepercaya. Pemasangan konfigurasi proxy dan sertifikat tepercaya, dalam konteks lokal lingkungan perantara klien, memungkinkan tingkat hak istimewa yang sangat tinggi. Untuk alasan ini, akses ke perantara tersebut harus dikontrol dengan ketat.

Tujuan dari perantara biasanya untuk memblokir jalan keluar yang tidak diinginkan dan untuk melacak jalan keluar lainnya. Dengan demikian, biasanya perantara tersebut mencatat permintaan. Meskipun perantara dapat, seperti klien, mencatat konten, header, dan data apa pun (yang semuanya akan sangat sensitif), lebih umum bagi mereka untuk mencatat URIs, seperti yang menyertakan `X-Amz-Signature` parameter string kueri.

Mitigasi

Kami menyarankan agar pencatatan URI menyunting parameter string `X-Amz-Signature` kueri, menyunting seluruh string kueri, atau memperlakukan informasi sebagai sangat rahasia, seperti dengan akses langsung ke server perantara. Meskipun perlindungan ini sangat direkomendasikan, fakta bahwa URLs kedaluwarsa yang telah ditentukan sebelumnya mengurangi risiko paparan log, selama eksposur tertunda cukup lama agar tanda tangan kedaluwarsa.

Amazon S3 juga melihat tanda tangan dan harus menanganinya dengan tepat. Log akses server Amazon S3 menyertakan URI permintaan tetapi menyuntingX`-Amz-Signature`, seperti yang disarankan. Hal yang sama berlaku ketika peristiwa CloudTrail data dicatat untuk Amazon S3. Anda dapat mengonfigurasi CloudWatch Log Amazon untuk menutupi data dengan menggunakan pengidentifikasi data khusus.

Ekspresi reguler berikut `X-Amz-Signature` cocok dengan yang muncul di URI:

```
X-Amz-Signature=[a-f0-9]{64}
```

Ekspresi reguler berikut ini menambahkan pola pengelompokan untuk mengidentifikasi teks yang akan diganti secara lebih spesifik:

```
(?:X-Amz-Signature=)([a-f0-9]{64})
```

Jika Anda memiliki entri log akses seperti berikut ini:

```
X-Amz-Signature=733255ef022bec3f2a8701cd61d4b371f3f28c9f193a1f02279211d48d5193d7
```

Ekspresi reguler pertama menerjemahkan entri log akses ke:

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Ekspresi reguler kedua, pada sistem yang mendukung grup non-penangkapan, menerjemahkan entri log akses ke:

```
X-Amz-Signature=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Pertanyaan yang Sering Diajukan

Bisakah permintaan yang telah ditetapkan sebelumnya digunakan beberapa kali? Apakah itu risiko keamanan?

Ya, tanda tangan dalam permintaan yang telah ditetapkan sebelumnya dapat digunakan lebih dari satu kali. Apakah ini risiko keamanan adalah pertanyaan kontekstual. Metode lain untuk mengakses layanan AWS memungkinkan pengulangan juga. Pengguna atau beban kerja dengan AWS kredensyal dapat mengirim banyak permintaan ke Layanan AWS, dan salah satu permintaan tersebut dapat berupa duplikat.

Jika kasus penggunaan Anda memerlukan eksekusi sekali dan hanya sekali, Anda harus menerapkan mekanisme lain untuk menerapkan penggunaan tunggal. Penggunaan tunggal bukanlah fitur permintaan yang telah ditetapkan sebelumnya. Sebagai insinyur keamanan, Anda harus meninjau kasus penggunaan dan implementasi, tetapi dalam banyak kasus, beberapa penggunaan akan sesuai dengan penggunaan yang dapat diterima.

Dapatkah seseorang selain pengguna yang dituju menggunakan permintaan yang telah ditentukan sebelumnya?

Tanda tangan dalam permintaan yang telah ditentukan sebelumnya dapat dikirim oleh siapa saja yang memilikinya. Ini akan diterima hanya jika melewati bentuk validasi lain, seperti kontrol perimeter data. Jika tanda tangan telah kedaluwarsa, kredensyal penandatanganan telah kedaluwarsa, atau kredensyal penandatanganan tidak memiliki akses ke sumber daya yang diminta, permintaan akan ditolak.

Hal yang sama berlaku untuk metode otentikasi lainnya dengan Layanan AWS. Kredensyal yang dibagikan secara tidak tepat memungkinkan akses yang tidak pantas. Praktik terbaik inti adalah berbagi kredensyal dan tanda tangan hanya dengan audiens yang dituju. Jika Anda tidak dapat mempercayai audiens yang dituju untuk menjaga keamanan data pribadi dan tidak membagikannya dengan orang lain, ini akan merusak segala bentuk otentikasi.

Dapatkah pengguna yang berwenang menggunakan permintaan yang telah ditetapkan sebelumnya untuk mengekstrasi data?

Mengamankan data membutuhkan tindakan yang kuat. Mengaktifkan akses untuk tujuan yang dimaksudkan sambil mempertahankan perimeter data memerlukan pendekatan yang komprehensif. Akses dengan hak istimewa paling sedikit, kontrol perimeter data, dan hanya menggunakan kredensyal akses sementara adalah praktik terbaik umum yang berlaku untuk mengamankan data. Penggunaan yang tepat dari kontrol ini juga membatasi kemampuan pengguna untuk melakukan tindakan melalui permintaan yang telah ditetapkan sebelumnya yang mereka hasilkan.

Hal ini karena akses yang disediakan oleh permintaan presigned adalah bagian dari akses yang diberikan ke kredensi yang digunakan untuk menandatangani permintaan. Dalam konteks ini, praktik terbaik yang berlaku untuk mengakses data umumnya berlaku untuk permintaan yang telah ditetapkan sebelumnya, tetapi permintaan yang telah ditetapkan sebelumnya tidak membuat akses baru ke data.

- Kedaluwarsa maksimum terbatas pada berakhirnya kredensyal penandatanganan. Jika kredensyal penandatanganan dicabut, tanda tangan berdasarkan kredensyal tidak lagi valid.
- Jika izin untuk prinsipal IAM yang terkait dengan kredensyal penandatanganan tidak menyertakan eksekusi tindakan yang terkait dengan permintaan yang telah ditetapkan sebelumnya, pemanggilan permintaan yang telah ditetapkan sebelumnya akan menghasilkan respons “akses ditolak”. Respons bergantung pada status izin saat ini pada saat pemanggilan, yang tidak memiliki hubungan dengan waktu tanda tangan permintaan yang telah ditetapkan sebelumnya dibuat.
- Properti prinsipal dievaluasi berdasarkan prinsip yang terkait dengan kredensi penandatanganan.
- Properti sesi peran dievaluasi berdasarkan sesi peran yang terkait dengan kredensyal penandatanganan.
- Properti jaringan dievaluasi berdasarkan bagaimana permintaan diterima, seperti permintaan normal.

Dalam konteks ini, pemeriksaan risiko yang terkait dengan permintaan yang telah ditetapkan sebelumnya dibatasi pada area di mana mereka ditandatangani dengan kredensyal yang berbeda dari kredensyal pengguna dan menyediakan akses yang bukan bagian dari prinsipal pengguna. Pemeriksaan ini harus diterapkan pada desain layanan, beban kerja, atau solusi yang menghasilkan tanda tangan atas nama pengguna, bukan kemampuan permintaan yang telah ditentukan sebelumnya itu sendiri.

Dapatkah saya menolak akses dari URL yang telah ditetapkan sebelumnya jika saya curiga telah dibagikan dengan cara yang tidak sah?

Ya. Ini memerlukan pembatalan kredensyal yang ditandatangani URL. Ada beberapa cara untuk mencapai ini:

- Hapus izin dari prinsipal IAM yang menjadi milik kredensialnya. Jika prinsipal IAM tersebut tidak lagi memiliki akses ke sumber daya dan operasi tempat URL ditandatangani, URL tidak dapat menjalankan operasi itu. Ini mempengaruhi semua penggunaan yang cocok dari prinsipal IAM tersebut.
- Jika kredensyal yang digunakan untuk menandatangani URL adalah AWS STS kredensyal sementara, Anda dapat mencabut izin sesi untuk kredensyal sementara yang dikeluarkan sebelum waktu tertentu untuk prinsipal IAM. Bergantung pada kasus penggunaan, mungkin ada sesi valid lainnya yang menjadi tidak valid sebelum waktu kedaluwarsa normal, tetapi sesi baru tidak akan terpengaruh. Mencabut izin sesi juga membatalkan URL apa pun yang ditandatangani dengan menggunakan kredensyal yang terkait dengan sesi tersebut, tetapi URL baru yang terkait dengan sesi baru tidak akan terpengaruh.
- Jika kredensyal yang digunakan untuk menandatangani URL adalah kredensyal permanen, nonaktifkan kunci akses. Ini memengaruhi semua penggunaan yang terkait dengan kredensyal tersebut.

Sumber daya

Dokumentasi Amazon S3

- Permintaan Otentikasi (Versi AWS Tanda Tangan 4)
- Permintaan Otentikasi: Menggunakan Parameter Kueri (Versi AWS Tanda Tangan 4)
- Permintaan Otentikasi: Unggahan Berbasis Browser Menggunakan POST (AWS Versi Tanda Tangan 4)
- Amazon S3 Signature Versi 4 Kunci Kebijakan Khusus Otentikasi
- Bekerja dengan URL yang telah ditetapkan sebelumnya

Referensi lainnya

- Membangun Perimeter Data di AWS (AWS whitepaper)
- SEC03-BP02 Berikan akses hak istimewa paling sedikit (Kerangka yang Dirancang AWS dengan Baik, pilar Keamanan)
- SEC03-BP05 Tentukan pagar pembatas izin untuk organisasi Anda (Kerangka Kerja yang Dirancang dengan Baik, Pilar Keamanan)AWS

Lampiran A: Bagaimana penggunaan presigned Layanan AWS URLs

Lampiran ini memberikan informasi tentang Layanan AWS dan fitur yang menggunakan presigned. URLs Informasi ini melayani dua tujuan:

- Untuk menyediakan insinyur keamanan yang menerapkan kontrol dengan informasi tentang kemungkinan dampak dari kontrol tersebut.
- Untuk menciptakan kesadaran akan situasi di mana risiko ini mungkin relevan untuk interaksi URL logging.

> **Important**
>
> Lampiran ini tidak menyediakan daftar lengkap Layanan AWS atau penggunaan presigned. URLs Ini juga tidak mencakup solusi kustom atau pihak ketiga.

Konsol Amazon S3

Prinsipal: Pengguna konsol

Kedaluwarsa default: 5 menit

> **Sanggahan**
>
> Bagian ini mendokumentasikan perilaku konsol Amazon S3 saat ini. AWS perilaku konsol dapat berubah tanpa pemberitahuan.

Konsol Amazon S3 mendukung pengunduhan dan pengunggahan objek. Unduhan menggunakan presigned URL yang memiliki waktu kedaluwarsa 300 detik (5 menit). URLItu dihasilkan oleh permintaan untuk`https://<bucket-region>.console.aws.amazon.com/s3/batch0psServlet-proxy`.

Permintaan itu dimulai ketika pengguna mengklik tombol unduh, sehingga URL tidak dibuat terlebih dahulu atau dikirim ke klien sampai permintaan eksplisit untuk mengunduh terjadi.

Unggahan serupa, kecuali konsol mengirimkan dua permintaan: `OPTIONS` sebagai CORS pemeriksaan pra-penerbangan, dan. `PUT` Kedua permintaan menggunakan tanda tangan yang sama.

Kredensil yang digunakan untuk penandatanganan adalah kredensil sementara yang terkait dengan pengguna yang saat ini masuk. Rincian tentang metode untuk mendapatkan kredensil sementara tersebut berada di luar cakupan panduan ini.

Amazon S3 Objek Lambda

Prinsipal: Penelepon titik akses

Kedaluwarsa default: 61 detik

Amazon S3 Object Lambda menggunakan AWS Lambda fungsi untuk memproses dan mengubah data secara otomatis saat diambil dari Amazon S3. Ketika S3 Object Lambda memanggil fungsi, fungsi tersebut diberikan URL presigned `inputS3Url` () yang dapat digunakan untuk mengunduh objek asli dari titik akses pendukung.

Presigned ini ditandatangani URLs untuk jalur akses Amazon S3 pendukung, yang disediakan saat Anda mengonfigurasi S3 Object Lambda. (Ini tidak sama dengan titik akses Object Lambda.) Alih-alih menggunakan peran yang terikat pada fungsi Lambda, peran URL tersebut ditandatangani dengan menggunakan identitas pemanggil asli, dan izin pengguna tersebut akan berlaku saat digunakan. URL Jika ada header yang ditandatangani diURL, fungsi Lambda harus menyertakan header ini dalam panggilan ke Amazon S3.

Presigned URL yang dikembalikan memiliki waktu kedaluwarsa 61 detik (satu detik lebih dari durasi maksimum untuk fungsi Lambda Objek S3). Yang dihasilkan hanya URL dapat digunakan dengan titik akses pendukung. Penelepon titik akses Lambda Objek S3 perlu memiliki akses ke titik akses ini. Anda dapat membatasi akses itu ke konteks S3 Object Lambda dengan menggunakan kondisi. `"aws:CalledVia": ["s3-object-lambda.amazonaws.com"]` Ketika kondisi tersebut dilampirkan ke titik akses atau bucket pendukung, pengguna tidak dapat mengakses jalur akses atau bucket pendukung secara langsung.

Nilai dari pendekatan ini adalah tidak perlu memberikan akses fungsi Lambda ke bucket atau titik akses S3 Anda. Peran yang terkait dengan fungsi Lambda akan memerlukan izin untuk WriteGetObjectResponse, tetapi tidak memerlukan izin untuk. GetObject

Ketika S3 Object Lambda menghasilkan URLs presigned, itu tidak menambahkan batasan jaringan, sehingga URL dapat digunakan di luar fungsi Lambda. Namun, batasan apa pun yang ditempatkan

pada penelepon S3 Object Lambda masih berlaku. Misalnya, jika fungsi Lambda Anda berjalan di a VPC dan Anda membatasi pemanggil untuk menggunakan VPC titik akhir, siapa pun yang memiliki presigned URL akan memerlukan kemampuan untuk mengirimnya melalui titik akhir tersebut. VPC Pembatasan ini juga berlaku untuk SourceIpdan VpcSourceIp.

> **Note**
>
> Untuk menggunakan fungsi Lambda Objek S3 di VPC a, VPC harus memiliki rute ke titik akhir S3 publik untuk dipanggil. WriteGetObjectResponse Ini tidak menunjukkan bahwa persyaratan untuk menggunakan VPC titik akhir tidak akan berlaku untuk permintaan untuk mengambil data dari bucket.

AWS Lambda Lintas Wilayah CopyObject

Prinsipal:AWS internal

Kedaluwarsa default: 3600 detik

Saat Anda menggunakan CopyObjectatau UploadPartCopyAPIuntuk menyalin Wilayah AWS, Amazon S3 menggunakan URLs presigned secara internal. Ini APIs dapat dipanggil langsung dari SDKs atau dari AWS CLI perintah `aws s3api copy-object` dan`aws s3api upload-part`. Ini APIs tidak digunakan untuk Replikasi Amazon S3, tetapi digunakan oleh `aws s3 sync` perintah AWS CLI `aws s3 cp` dan saat sumber dan tujuan adalah bucket S3. Mereka juga didukung oleh `TransferManager` implementasi di berbagai AWS SDKs.

AWS Lambda GetFunction

Prinsipal: AWS internal

Kedaluwarsa default: 10 menit

AWS Lambda menyimpan versi pengguna dalam bucket S3 yang dimiliki tim Lambda, sebelum membuat aset yang digunakan ke kontainer Lambda. Ketika Anda ingin mengakses kode untuk fungsi Anda, Anda memanggil GetFunctionAPI. Ini API merespons dengan`Code.Location`, yang berisi presigned URL yang valid selama 10 menit (waktu kedaluwarsa ini adalah perilaku saat ini dan bukan kontrak yang dipublikasikan). Jika Anda tidak ingin kode, Anda dapat menggunakan kombinasi

GetFunctionConfiguration, GetFunctionConcurrency, dan ListTagsuntuk mengambil data lain yang dikembalikan olehGetFunction.

Pengembalian URL tidak ditandatangani dengan kredensil pengguna yang saat ini masuk, tetapi atas nama pengguna oleh Lambda. Untuk alasan ini, kunci kondisi (sepertiaws:SourceIP) yang diterapkan ke pengguna yang saat ini masuk atau kredensi sesi sementara pengguna tidak berlaku untuk yang dihasilkan. URL Ini benar apakah kunci kondisi diterapkan GetFunctionhanya, atau diterapkan ke semua AWS API penggunaan untuk pengguna atau sesi.

Konsol Lambda juga menggunakan GetFunctiondan presigned itu kembaliURL. Konsol menggunakan kredensil sementara yang terkait dengan pengguna yang saat ini masuk untuk menelepon. GetFunction Rincian tentang mendapatkan kredensil sementara tersebut berada di luar cakupan dokumen ini.

Amazon ECR

Prinsipal: AWS internal

Kedaluwarsa default: 1 jam

Amazon Elastic Container Registry (AmazonECR) menyediakan GetDownloadUrlForLayerAPI, URL yang mengembalikan presigned yang valid selama satu jam dan mendukung pengunduhan satu lapisan dari ECR gambar Amazon. Namun, operasi ini digunakan oleh ECR proxy Amazon dan umumnya tidak digunakan oleh pengguna untuk menarik dan mendorong gambar.

Amazon Redshift Spectrum

Prinsipal: Peran diteruskan CREATEEXTERNALSCHEMAIAM_ROLE

Kedaluwarsa default: 1 jam

Amazon Redshift Spectrum menggunakan URLs presigned internal dan melarang pembatasan kombinasi bucket dan peran Amazon Redshift yang akan membatasi presigned. URLs Anda dapat menggunakan s3:signatureAge nilai 16 menit, tetapi nilai yang sangat rendah tidak dapat diandalkan. Nilai minimum yang dapat Anda gunakan tergantung pada waktu dan ukuran kueri Anda. Meskipun nilai yang lebih rendah dari 16 menit berfungsi untuk banyak skenario, itu membutuhkan pengujian. Peran tersebut dapat dan harus dibatasi untuk digunakan hanya oleh Redshift Spectrum, yang tidak mengungkapkan yang URLs dihasilkannya, sehingga mengurangi pembenaran khas untuk nilai kedaluwarsa yang lebih rendah.

Studio SageMaker AI Amazon

Amazon SageMaker AI Studio mendukung dua API tindakan: CreatePresignedDomainUrldan CreatePresignedNotebookInstanceUrl. Namun, ini APIs tidak terkait dengan URL fitur yang telah ditetapkan sebelumnya Versi Tanda Tangan 4. Ini APIs membuat URL yang menggunakan `authToken` parameter, tetapi tidak mendukung parameter kueri Signature Version 4 standar apa pun.

`authToken`adalah mekanisme yang berbeda tetapi memiliki kesamaan dengan URLs presigned. Ini dikirim sebagai parameter string kueri dan mendukung waktu kedaluwarsa 5 menit.

SageMaker AI mendukung pembatasan jaringan. Jika Anda membatasi `sagemaker:CreatePresignedDomainUrl` tindakan, tindakan tersebut berlaku baik untuk panggilan CreatePresignedDomainUrlmaupun penggunaan yang dihasilkanURL. Jika a URL dihasilkan dari jaringan yang valid dan kemudian dikirim oleh jaringan yang tidak valid, API panggilan untuk menghasilkan URL berhasil, tetapi permintaan yang mengirim gagal. URL Hal yang sama berlaku CreatePresignedNotebookInstanceUrldan `sagemaker:CreatePresignedNotebookInstanceUrl` tindakannya.

Untuk informasi selengkapnya, lihat dokumentasi SageMaker AI.

Lampiran B: Bagaimana kontrol untuk URL yang telah ditetapkan sebelumnya memengaruhi Layanan AWS

Lampiran ini menjelaskan interaksi antara Layanan AWS yang menggunakan URL yang telah ditetapkan sebelumnya, seperti yang dijelaskan dalam Lampiran A, dan kontrol yang dijelaskan sebelumnya dalam panduan ini.

Pagar pembatas untuk S3: SignatureAge

Konsol Amazon S3 tidak terganggu oleh kedaluwarsa maksimum 5 menit yang ditetapkan oleh tombol kondisi. `s3:signatureAge` Konsol Amazon S3 menghasilkan URL yang telah ditentukan sebelumnya saat Anda memilih tombol Unduh dan menerapkan waktu kedaluwarsa 5 menitnya sendiri. Durasi maksimum yang lebih pendek dari 2 menit dapat membuat kegagalan acak berdasarkan sinkronisasi jam dan latensi.

Amazon S3 Object Lambda menggunakan waktu kedaluwarsa 61 detik, jadi pengaturan kondisi pada `s3:signatureAge` nilai 61 detik atau lebih tidak akan menyebabkan gangguan apa pun. Durasi yang lebih pendek mungkin kurang dapat diandalkan dan dapat menyebabkan kegagalan intermiten.

Amazon S3 Cross-region CopyObjecttidak terganggu oleh kedaluwarsa maksimum 5 menit. Namun, durasi yang lebih pendek dapat membuat kegagalan acak berdasarkan sinkronisasi jam dan latensi.

Di AWS Lambda, GetFunctionberikan URL ke objek di luar akun pelanggan, sehingga kebijakan pelanggan tidak memengaruhi URL yang dihasilkan.

Amazon Redshift Spectrum telah diuji `s3:signatureAge` dengan kondisi 16 menit. Namun, durasi yang lebih pendek dapat menyebabkan gangguan.

Guardrail untuk S3:authType saat tidak menggunakan batasan jaringan

Konsol Amazon S3 biasanya dipengaruhi oleh pagar `s3:authType` pembatas. Konsol merutekan ke Amazon S3 berdasarkan konfigurasi jaringan lokal. Jika jaringan lokal merutekan ke Amazon S3 dengan cara yang memungkinkan pembatasan jaringan, konsol Amazon S3 akan tetap berfungsi. Namun, jika dialihkan melalui proxy atau internet publik dengan cara yang tidak diizinkan,

penggunaan akan diblokir. Namun, memblokir penggunaan mungkin merupakan maksud dari kebijakan ini.

Lambda Objek Amazon S3 terpengaruh jika fungsi Lambda tidak terhubung ke VPC yang sesuai. Dalam konfigurasi ini, VPC harus memiliki gateway NAT, bukan untuk mengakses bucket S3, tetapi untuk menelepon. WriteGetObjectResponse

Amazon S3 Lintas wilayah CopyObjectterganggu jika pagar pembatas ini diterapkan pada kebijakan bucket tanpa pengecualian yang disarankan untuk kapan benar. **`aws:viaAWSService`**

Amazon Redshift Spectrum dipengaruhi oleh pagar pembatas kecuali `s3:authType` perutean VPC yang ditingkatkan digunakan. Saat ini, Redshift Spectrum mendukung peningkatan perutean VPC hanya dengan cluster tanpa server, bukan dengan cluster yang disediakan.

Riwayat dokumen

Tabel berikut menjelaskan perubahan signifikan pada panduan ini. Jika Anda ingin diberi tahu tentang pembaruan masa depan, Anda dapat berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Publikasi awal	—	Juli 23, 2024

AWS Glosarium Panduan Preskriptif

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

Nomor

7 Rs

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

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

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

A

ABAC

Lihat kontrol akses berbasis atribut.

layanan abstrak

Lihat layanan terkelola.

ASAM

Lihat atomisitas, konsistensi, isolasi, daya tahan.

migrasi aktif-aktif

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

migrasi aktif-pasif

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

fungsi agregat

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

AI

Lihat kecerdasan buatan.

AIOps

Lihat operasi kecerdasan buatan.

anonimisasi

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

anti-pola

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

kontrol aplikasi

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

portofolio aplikasi

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

kecerdasan buatan (AI)

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

operasi kecerdasan buatan (AIOps)

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

enkripsi asimetris

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

atomisitas, konsistensi, isolasi, daya tahan (ACID)

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

kontrol akses berbasis atribut (ABAC)

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

sumber data otoritatif

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

Zona Ketersediaan

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

AWS Kerangka Adopsi Cloud (AWS CAF)

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

AWS Kerangka Kualifikasi Beban Kerja (AWS WQF)

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

B

bot buruk

Bot yang dimaksudkan untuk mengganggu atau membahayakan individu atau organisasi.

BCP

Lihat perencanaan kontinuitas bisnis.

grafik perilaku

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

sistem big-endian

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

klasifikasi biner

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

filter mekar

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

deployment biru/hijau

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

bot

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

botnet

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

cabang

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

akses break-glass

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

strategi brownfield

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

cache penyangga

Area memori tempat data yang paling sering diakses disimpan.

kemampuan bisnis

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

perencanaan kelangsungan bisnis (BCP)

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

C

KAFE

Lihat Kerangka Adopsi AWS Cloud.

penyebaran kenari

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

CCoE

Lihat Cloud Center of Excellence.

CDC

Lihat mengubah pengambilan data.

ubah pengambilan data (CDC)

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

rekayasa kekacauan

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

CI/CD

Lihat integrasi berkelanjutan dan pengiriman berkelanjutan.

klasifikasi

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

Enkripsi sisi klien

Enkripsi data secara lokal, sebelum target Layanan AWS menerimanya.

Pusat Keunggulan Cloud (CCoE)

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

komputasi cloud

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

model operasi cloud

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

tahap adopsi cloud

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

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

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

CMDB

Lihat database manajemen konfigurasi.

repositori kode

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

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

cache dingin

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

data dingin

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

visi komputer (CV)

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

konfigurasi drift

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

database manajemen konfigurasi (CMDB)

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

paket kesesuaian

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

integrasi berkelanjutan dan pengiriman berkelanjutan (CI/CD)

Proses mengotomatiskan sumber, membangun, menguji, pementasan, dan tahap produksi dari proses rilis perangkat lunak. CI/CD is commonly described as a pipeline. CI/CDdapat membantu

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

CV

Lihat visi komputer.

D

data saat istirahat

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

klasifikasi data

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

penyimpangan data

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

data dalam transit

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

jala data

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

minimalisasi data

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

perimeter data

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

prapemrosesan data

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

asal data

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

subjek data

Individu yang datanya dikumpulkan dan diproses.

gudang data

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

bahasa definisi database (DDL)

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

bahasa manipulasi basis data (DHTML)

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

DDL

Lihat bahasa definisi database.

ansambel yang dalam

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

pembelajaran mendalam

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

defense-in-depth

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

administrator yang didelegasikan

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

deployment

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

lingkungan pengembangan

Lihat lingkungan.

kontrol detektif

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

pemetaan aliran nilai pengembangan (DVSM)

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

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

kembar digital

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

tabel dimensi

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

musibah

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

pemulihan bencana (DR)

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

DML~

Lihat bahasa manipulasi basis data.

desain berbasis domain

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

DR

Lihat pemulihan bencana.

deteksi drift

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

DVSM

Lihat pemetaan aliran nilai pengembangan.

E

EDA

Lihat analisis data eksplorasi.

EDI

Lihat pertukaran data elektronik.

komputasi tepi

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

pertukaran data elektronik (EDI)

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

enkripsi

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

kunci enkripsi

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

endianness

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

titik akhir

Lihat titik akhir layanan.

layanan endpoint

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

perencanaan sumber daya perusahaan (ERP)

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

enkripsi amplop

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

lingkungan

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

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

epik

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

ERP

Lihat perencanaan sumber daya perusahaan.

analisis data eksplorasi (EDA)

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

F

tabel fakta

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

gagal cepat

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

batas isolasi kesalahan

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

cabang fitur

Lihat cabang.

fitur

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

pentingnya fitur

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

transformasi fitur

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

beberapa tembakan mendorong

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

FGAC

Lihat kontrol akses berbutir halus.

kontrol akses berbutir halus (FGAC)

Penggunaan beberapa kondisi untuk mengizinkan atau menolak permintaan akses.

migrasi flash-cut

Metode migrasi database yang menggunakan replikasi data berkelanjutan melalui pengambilan data perubahan untuk memigrasikan data dalam waktu sesingkat mungkin, alih-alih menggunakan pendekatan bertahap. Tujuannya adalah untuk menjaga downtime seminimal mungkin.

FM

Lihat model pondasi.

model pondasi (FM)

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

G

AI generatif

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

pemblokiran geografis

Lihat pembatasan geografis.

pembatasan geografis (pemblokiran geografis)

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

Alur kerja Gitflow

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

gambar emas

Sebuah snapshot dari sistem atau perangkat lunak yang digunakan sebagai template untuk menyebarkan instance baru dari sistem atau perangkat lunak itu. Misalnya, di bidang manufaktur, gambar emas dapat digunakan untuk menyediakan perangkat lunak pada beberapa perangkat dan membantu meningkatkan kecepatan, skalabilitas, dan produktivitas dalam operasi manufaktur perangkat.

strategi greenfield

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

pagar pembatas

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

H

HA

Lihat ketersediaan tinggi.

migrasi database heterogen

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

ketersediaan tinggi (HA)

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

modernisasi sejarawan

Pendekatan yang digunakan untuk memodernisasi dan meningkatkan sistem teknologi operasional (OT) untuk melayani kebutuhan industri manufaktur dengan lebih baik. Sejarawan

adalah jenis database yang digunakan untuk mengumpulkan dan menyimpan data dari berbagai sumber di pabrik.

data penahanan

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

migrasi database homogen

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

data panas

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

perbaikan terbaru

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

periode hypercare

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

I

IAc

Lihat infrastruktur sebagai kode.

kebijakan berbasis identitas

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

aplikasi idle

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

IIoT

Lihat Internet of Things industri.

infrastruktur yang tidak dapat diubah

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

masuk (masuknya) VPC

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

migrasi inkremental

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

Industri 4.0

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

infrastruktur

Semua sumber daya dan aset yang terkandung dalam lingkungan aplikasi.

infrastruktur sebagai kode (IAc)

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

Internet of Things industri (IIoT)

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

inspeksi VPC

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

Internet of Things (IoT)

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

interpretabilitas

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

IoT

Lihat Internet of Things.

Perpustakaan informasi TI (ITIL)

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

Manajemen layanan TI (ITSM)

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

ITIL

Lihat perpustakaan informasi TI.

ITSM

Lihat manajemen layanan TI.

L

kontrol akses berbasis label (LBAC)

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

landing zone

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

model bahasa besar (LLM)

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

migrasi besar

Migrasi 300 atau lebih server.

LBAC

Lihat kontrol akses berbasis label.

hak istimewa paling sedikit

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

angkat dan geser

Lihat 7 Rs.

sistem endian kecil

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

LLM

Lihat model bahasa besar.

lingkungan yang lebih rendah

Lihat lingkungan.

M

pembelajaran mesin (ML)

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

cabang utama

Lihat cabang.

malware

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

layanan terkelola

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

sistem eksekusi manufaktur (MES)

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

PETA

Lihat Program Percepatan Migrasi.

mekanisme

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

akun anggota

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

MES

Lihat sistem eksekusi manufaktur.

Transportasi Telemetri Antrian Pesan (MQTT)

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

layanan mikro

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

informasi selengkapnya, lihat Mengintegrasikan layanan mikro dengan menggunakan layanan tanpa AWS server.

arsitektur microservices

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

Program Percepatan Migrasi (MAP)

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

migrasi dalam skala

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

pabrik migrasi

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

metadata migrasi

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

pola migrasi

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

Penilaian Portofolio Migrasi (MPA)

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

Penilaian Kesiapan Migrasi (MRA)

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

strategi migrasi

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

ML

Lihat pembelajaran mesin.

modernisasi

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

penilaian kesiapan modernisasi

Evaluasi yang membantu menentukan kesiapan modernisasi aplikasi organisasi; mengidentifikasi manfaat, risiko, dan dependensi; dan menentukan seberapa baik organisasi dapat mendukung keadaan masa depan aplikasi tersebut. Hasil penilaian adalah cetak biru arsitektur target, peta

jalan yang merinci fase pengembangan dan tonggak untuk proses modernisasi, dan rencana aksi untuk mengatasi kesenjangan yang diidentifikasi. Untuk informasi lebih lanjut, lihat Mengevaluasi kesiapan modernisasi untuk aplikasi di. AWS Cloud

aplikasi monolitik (monolit)

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

MPA

Lihat Penilaian Portofolio Migrasi.

MQTT

Lihat Transportasi Telemetri Antrian Pesan.

klasifikasi multiclass

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

infrastruktur yang bisa berubah

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

O

OAC

Lihat kontrol akses asal.

OAI

Lihat identitas akses asal.

OCM

Lihat manajemen perubahan organisasi.

migrasi offline

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

OI

Lihat integrasi operasi.

OLA

Lihat perjanjian tingkat operasional.

migrasi online

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

OPC-UA

Lihat Komunikasi Proses Terbuka - Arsitektur Terpadu.

Komunikasi Proses Terbuka - Arsitektur Terpadu (OPC-UA)

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

perjanjian tingkat operasional (OLA)

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

Tinjauan Kesiapan Operasional (ORR)

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

teknologi operasional (OT)

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

integrasi operasi (OI)

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

jejak organisasi

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

manajemen perubahan organisasi (OCM)

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

kontrol akses asal (OAC)

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

identitas akses asal (OAI)

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

ORR

Lihat tinjauan kesiapan operasional.

OT

Lihat teknologi operasional.

keluar (jalan keluar) VPC

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

P

batas izin

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

Informasi Identifikasi Pribadi (PII)

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

PII

Lihat informasi yang dapat diidentifikasi secara pribadi.

buku pedoman

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

PLC

Lihat pengontrol logika yang dapat diprogram.

PLM

Lihat manajemen siklus hidup produk.

kebijakan

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

ketekunan poliglot

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

penilaian portofolio

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

predikat

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

predikat pushdown

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

kontrol preventif

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

principal

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

privasi berdasarkan desain

Pendekatan rekayasa sistem yang memperhitungkan privasi melalui seluruh proses pengembangan.

zona yang dihosting pribadi

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

kontrol proaktif

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

manajemen siklus hidup produk (PLM)

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

lingkungan produksi

Lihat lingkungan.

pengontrol logika yang dapat diprogram (PLC)

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

rantai cepat

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

pseudonimisasi

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

publish/subscribe (pub/sub)

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

Q

rencana kueri

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

regresi rencana kueri

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

R

Matriks RACI

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

LAP

Lihat Retrieval Augmented Generation.

ransomware

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

Matriks RASCI

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

RCAC

Lihat kontrol akses baris dan kolom.

replika baca

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

arsitek ulang

Lihat 7 Rs.

tujuan titik pemulihan (RPO)

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

tujuan waktu pemulihan (RTO)

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

refactor

Lihat 7 Rs.

Wilayah

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

regresi

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

rehost

Lihat 7 Rs.

melepaskan

Dalam proses penyebaran, tindakan mempromosikan perubahan pada lingkungan produksi.

memindahkan

Lihat 7 Rs.

memplatform ulang

Lihat 7 Rs.

pembelian kembali

Lihat 7 Rs.

ketahanan

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

kebijakan berbasis sumber daya

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

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

Matriks yang mendefinisikan peran dan tanggung jawab untuk semua pihak yang terlibat dalam kegiatan migrasi dan operasi cloud. Nama matriks berasal dari jenis tanggung jawab yang didefinisikan dalam matriks: bertanggung jawab (R), akuntabel (A), dikonsultasikan (C), dan diinformasikan (I). Tipe dukungan (S) adalah opsional. Jika Anda menyertakan dukungan, matriks disebut matriks RASCI, dan jika Anda mengecualikannya, itu disebut matriks RACI.

kontrol responsif

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

melestarikan

Lihat 7 Rs.

pensiun

Lihat 7 Rs.

Retrieval Augmented Generation (RAG)

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

rotasi

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

kontrol akses baris dan kolom (RCAC)

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

RPO

Lihat tujuan titik pemulihan.

RTO

Lihat tujuan waktu pemulihan.

buku runbook

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

D

SAML 2.0

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

SCADA

Lihat kontrol pengawasan dan akuisisi data.

SCP

Lihat kebijakan kontrol layanan.

Rahasia

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

keamanan dengan desain

Pendekatan rekayasa sistem yang memperhitungkan keamanan melalui seluruh proses pengembangan.

kontrol keamanan

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

pengerasan keamanan

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

sistem informasi keamanan dan manajemen acara (SIEM)

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

otomatisasi respons keamanan

Tindakan yang telah ditentukan dan diprogram yang dirancang untuk secara otomatis merespons atau memulihkan peristiwa keamanan. Otomatisasi ini berfungsi sebagai kontrol keamanan

detektif atau responsif yang membantu Anda menerapkan praktik terbaik AWS keamanan. Contoh tindakan respons otomatis termasuk memodifikasi grup keamanan VPC, menambal instans EC2 Amazon, atau memutar kredensil.

enkripsi sisi server

Enkripsi data di tujuannya, oleh Layanan AWS yang menerimanya.

kebijakan kontrol layanan (SCP)

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

titik akhir layanan

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

perjanjian tingkat layanan (SLA)

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

indikator tingkat layanan (SLI)

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

tujuan tingkat layanan (SLO)

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

model tanggung jawab bersama

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

SIEM

Lihat informasi keamanan dan sistem manajemen acara.

titik kegagalan tunggal (SPOF)

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

SLA

Lihat perjanjian tingkat layanan.

SLI

Lihat indikator tingkat layanan.

SLO

Lihat tujuan tingkat layanan.

split-and-seed model

Pola untuk menskalakan dan mempercepat proyek modernisasi. Ketika fitur baru dan rilis produk didefinisikan, tim inti berpisah untuk membuat tim produk baru. Ini membantu meningkatkan kemampuan dan layanan organisasi Anda, meningkatkan produktivitas pengembang, dan mendukung inovasi yang cepat. Untuk informasi lebih lanjut, lihat Pendekatan bertahap untuk memodernisasi aplikasi di. AWS Cloud

SPOF

Lihat satu titik kegagalan.

skema bintang

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

pola ara pencekik

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

subnet

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

kontrol pengawasan dan akuisisi data (SCADA)

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

enkripsi simetris

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

pengujian sintetis

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

sistem prompt

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

T

tag

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

variabel target

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

daftar tugas

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

lingkungan uji

Lihat lingkungan.

pelatihan

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

gerbang transit

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

alur kerja berbasis batang

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

akses tepercaya

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

penyetelan

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

tim dua pizza

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

U

waswas

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

tugas yang tidak terdiferensiasi

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

lingkungan atas

Lihat lingkungan.

V

menyedot debu

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

kendali versi

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

Peering VPC

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

kerentanan

Kelemahan perangkat lunak atau perangkat keras yang membahayakan keamanan sistem.

W

cache hangat

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

data hangat

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

fungsi jendela

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

beban kerja

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

aliran kerja

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

CACING

Lihat menulis sekali, baca banyak.

WQF

Lihat AWS Kerangka Kualifikasi Beban Kerja.

tulis sekali, baca banyak (WORM)

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

Z

eksploitasi zero-day

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

kerentanan zero-day

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

bisikan zero-shot

Memberikan LLM dengan instruksi untuk melakukan tugas tetapi tidak ada contoh (tembakan) yang dapat membantu membimbingnya. LLM harus menggunakan pengetahuan pra-terlatih untuk menangani tugas. Efektivitas bidikan nol tergantung pada kompleksitas tugas dan kualitas prompt. Lihat juga beberapa bidikan yang diminta.

aplikasi zombie

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

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