Alat pemantauan dan peringatan serta praktik terbaik untuk Amazon RDS for MySQL dan MariaDB

AWS Bimbingan Preskriptif

AWS Bimbingan Preskriptif: Alat pemantauan dan peringatan serta praktik terbaik untuk Amazon RDS for MySQL dan MariaDB

Table of Contents

Alat pemantauan dan peringatan serta praktik terbaik untuk Amazon RDS untuk MySQL dan MariaDB

Igor Obradovic, Layanan Web Amazon (AWS)

Juni 2023(riwayat dokumen)

Pemantauan database adalah proses pengukuran, pelacakan, dan menilai ketersediaan, kinerja, dan fungsionalitas database. Solusi pemantauan dan peringatan membantu organisasi memastikan bahwa layanan database mereka, dan oleh karena itu aplikasi dan beban kerja terkait mereka, aman, berkinerja tinggi, tangguh, dan efisien. Di AWS, Anda dapat mengumpulkan dan menganalisis log, metrik, peristiwa, dan jejak beban kerja Anda untuk memahami kesehatan beban kerja Anda dan untuk mendapatkan wawasan dari operasi dari waktu ke waktu.

Anda dapat memantau sumber daya Anda untuk memastikan bahwa mereka berkinerja seperti yang diharapkan, dan untuk mendeteksi dan memperbaiki masalah apa pun sebelum memengaruhi pelanggan Anda. Anda harus menggunakan metrik, log, peristiwa, dan pelacakan yang Anda pantau untuk meningkatkan alarm saat ambang batas dilanggar.

Panduan ini menjelaskan observabilitas database dan alat pemantauan serta praktik terbaik untuk database Amazon Relational Database Service (Amazon RDS). Panduan ini berfokus pada database MySQL dan MariaDB, meskipun sebagian besar informasi juga berlaku untuk mesin database Amazon RDS lainnya.

Panduan ini untuk arsitek solusi, arsitek database, DBA, seniorDevOpsteknisi, dan anggota tim lainnya yang terlibat dalam merancang, mengimplementasikan, dan mengelola solusi pemantauan dan observabilitas untuk beban kerja database mereka yang berjalan di AWS Cloud.

Isi

- Memberi tahu
- Langkah dan sumber daya selanjutnya

Gambaran Umum

Pemantauan dan peringatan termasuk dalam empat pilar Kerangka AWS Well-Architected.

- Pilar keunggulan operasional menetapkan bahwa beban kerja Anda harus dirancang untuk mencakup telemetri dan pemantauan. AWS Layanan seperti Amazon Relational Database Service (Amazon RDS) menyediakan informasi yang diperlukan bagi Anda untuk memahami status internal beban kerja Anda (misalnya, metrik, log, peristiwa, dan jejak). Ketika Anda mengoperasikan database Amazon RDS Anda, Anda akan ingin memahami kesehatan instans database Anda, mendeteksi peristiwa operasional, dan dapat menanggapi peristiwa yang direncanakan dan tidak direncanakan. AWS menyediakan alat pemantauan yang membantu Anda menentukan kapan hasil organisasi dan bisnis berisiko, atau berpotensi berisiko, sehingga Anda dapat mengambil tindakan yang tepat pada waktu yang tepat.
- Pilar efisiensi kinerja menetapkan bahwa Anda harus memantau kinerja sumber daya Anda seperti instans Amazon RDS DB dengan mengumpulkan, menggabungkan, dan memproses metrik terkait kinerja secara real time. Anda dapat mengidentifikasi penurunan kinerja dan memulihkan faktor–misalnya, kueri SQL yang tidak dioptimalkan atau parameter konfigurasi yang tidak memadai–yang menyebabkannya. Anda dapat menaikkan alarm secara otomatis saat pengukuran berada di luar batas yang diharapkan. Kami menyarankan Anda menggunakan alarm tidak hanya untuk notifikasi, tetapi juga untuk memulai tindakan otomatis sebagai respons terhadap peristiwa yang terdeteksi. Anda dapat mengevaluasi metrik yang Anda kumpulkan terhadap ambang batas yang telah ditentukan atau menggunakan algoritme pembelajaran mesin untuk mengidentifikasi perilaku anomali. Misalnya, untuk mendeteksi tren peningkatan pemanfaatan CPU, Anda dapat mengumpulkan dan menganalisis `cpuUtilization.total` metrik selama periode waktu tertentu. Memperingatkan anomali itu secara proaktif, sebelum pemanfaatan CPU mencapai batas sulit, dapat membantu Anda memperbaiki masalah sebelum berdampak pada pelanggan Anda.
- Pilar keandalan mendefinisikan pemantauan dan peringatan sebagai hal penting untuk memastikan bahwa Anda memenuhi persyaratan ketersediaan Anda. Solusi pemantauan Anda harus dapat mendeteksi kegagalan secara efektif. Ketika mendeteksi masalah atau kegagalan, tujuan utamanya adalah untuk memperingatkan masalah tersebut. Menerapkan praktik observabilitas dan pemantauan berkelanjutan sangat penting untuk arsitektur tangguh di cloud. Untuk meningkatkan beban kerja Anda, Anda harus dapat mengukurnya dan memahami keadaan dan kesehatannya. Prinsip desain untuk pemulihan otomatis dari kegagalan, skalabilitas horizontal, dan penyediaan kapasitas bergantung pada layanan pemantauan dan peringatan yang akurat.

- Pilar keamanan membahas deteksi dan pencegahan perubahan konfigurasi yang tidak terduga atau tidak diinginkan, dan perilaku yang tidak terduga. Anda dapat mengonfigurasi instans Amazon RDS for MySQL dan MariaDB DB dengan Plugin Audit MariaDB untuk merekam aktivitas database seperti login pengguna dan operasi tertentu yang dijalankan terhadap database. Plugin menyimpan catatan aktivitas database dalam file log, yang dapat diintegrasikan dan diimpor ke alat pemantauan dan peringatan. File log dianalisis secara real time untuk perilaku yang tidak terduga atau mencurigakan dalam database Anda. Perilaku tak terduga atau mencurigakan seperti itu dapat menunjukkan bahwa instans Amazon RDS DB Anda telah dikompromikan, yang menandakan potensi risiko bagi bisnis Anda. Jika alat pemantauan mendeteksi peristiwa semacam itu, alat ini mengaktifkan alarm untuk memulai respons terhadap insiden keamanan, yang membantu mengatasi aktivitas yang mencurigakan dan berbahaya.

Hasil bisnis yang ditargetkan

Menerapkan praktik terbaik dalam mekanisme pemantauan dan peringatan membantu Anda memastikan infrastruktur yang berkinerja tinggi, tangguh, efisien, aman, dan dioptimalkan biaya untuk aplikasi dan beban kerja Anda. Anda dapat menggunakan alat observabilitas yang mengumpulkan, menyimpan, dan memvisualisasikan metrik, peristiwa, jejak, dan log secara real time untuk mengamati dan menganalisis gambaran yang lebih besar tentang kesehatan dan kinerja database Anda, dan dengan demikian mencegah degradasi atau gangguan layanan TI terkait Anda. Jika degradasi yang tidak direncanakan atau gangguan layanan masih terjadi, alat pemantauan dan peringatan membantu Anda mendeteksi masalah, eskalasi, reaksi, serta penyelidikan dan penyelesaian yang cepat secara tepat waktu. Solusi pemantauan dan peringatan komprehensif untuk beban kerja database cloud Anda membantu Anda mencapai hasil bisnis berikut:

- Tingkatkan pengalaman pelanggan. Layanan yang andal meningkatkan pengalaman pelanggan Anda. Database sering menjadi komponen kunci dari layanan digital seperti aplikasi web dan seluler, streaming media, pembayaran, business-to-business (B2B) API, dan layanan integrasi. Jika Anda dapat memantau dan mengatur peringatan di database Anda untuk mendeteksi masalah dengan cepat, menyelidikinya secara efisien, dan memperbaikinya sesegera mungkin untuk meminimalkan waktu henti dan gangguan lainnya, Anda dapat meningkatkan ketersediaan, keamanan, dan kinerja layanan digital untuk pelanggan Anda.
- Membangun kepercayaan pelanggan.Kinerja yang lebih baik dan pengalaman pengguna yang lebih lancar membantu Anda memenangkan kepercayaan pelanggan Anda, yang dapat menghasilkan lebih banyak bisnis di platform Anda. Misalnya, penyedia layanan pemrosesan pembayaran yang menawarkan layanan online yang andal dapat mengharapkan kepercayaan dan

loyalitas pelanggan yang tinggi, yang menghasilkan lebih banyak pelanggan dan retensi yang lebih baik, peningkatan transaksi yang dapat ditagih, dan layanan inovatif baru yang menghasilkan lebih banyak pendapatan.

- Hindari kerugian finansial.Setiap downtime yang tidak terduga dalam infrastruktur database Anda dapat memengaruhi transaksi bisnis yang dilakukan pelanggan Anda dengan menggunakan aplikasi Anda. Hal ini dapat menyebabkan kerugian finansial yang besar dalam beberapa kasus. Melanggar perjanjian tingkat layanan (SLA) dapat mengakibatkan hilangnya kepercayaan pelanggan, dan, akibatnya, hilangnya pendapatan. Ini juga bisa menjadi dasar hukum untuk uji coba mahal, di mana pelanggan mungkin menuntut kompensasi berdasarkan kewajiban dan kontrak garansi Anda. Menurut sebuah studi oleh Atlassian Corporation, sebuah perusahaan perangkat lunak, biaya rata-rata pemadaman layanan berada di kisaran $140K - $540K per jam, tergantung pada jenis dan ukuran bisnis. Lingkungan database yang stabil adalah kunci untuk mencegah pemadaman yang lama dan hilangnya bisnis.
- Perluas nilai. Mekanisme pemantauan dan peringatan dapat membantu Anda merancang, mengembangkan, dan mengoperasikan layanan digital yang sangat tersedia, tangguh, andal, berkinerja, hemat biaya, dan aman, tetapi ini baru permulaan. Anda akan ingin organisasi Anda untuk skala dan berkembang dari waktu ke waktu, meningkatkan beban kerja cloud yang ada, dan memperkenalkan layanan baru. Layanan baru memberikan nilai tambahan bagi pelanggan Anda dan lebih banyak pendapatan untuk bisnis Anda, menciptakan efek flywheel pada pertumbuhan bisnis Anda.
- Meningkatkan produktivitas pengembang.Pengembang yang produktif dan efisien, dan yang tidak mengalami masalah dan kemacetan dalam tugas pengembangan mereka, dapat memberikan produk berkualitas tinggi dalam waktu yang lebih singkat. Namun, rekayasa perangkat lunak dan operasi TI sering memiliki tantangan yang kompleks, dan kompleksitas ini meningkat dengan skala beban kerja dan arsitekturnya. Untuk menganalisis kinerja dan konsistensi di seluruh aplikasi terdistribusi, pengembang memerlukan alat yang dapat menyediakan metrik dan jejak yang berkorelasi. Ini membantu mengidentifikasi artefak kode yang rusak dan komponen infrastruktur secepat mungkin, dan membantu menentukan dampak pada pengguna akhir. Rangkaian alat pemantauan dan peringatan yang tepat dapat membantu pengembang membuat kode dan menguji dengan lebih baik dan lebih cepat.
- Meningkatkan efektivitas dan efisiensi operasional.Saat Anda mengoperasikan beban kerja cloud dalam skala besar, bahkan sebagian kecil peningkatan kinerja dapat menghasilkan penghematan jutaan dolar. Dengan memantau database Anda dan menganalisis metrik, peristiwa, log, dan jejak, Anda dapat memahami dan memprediksi kebutuhan kapasitas masa depan Anda, dan dapat memanfaatkan penghematan biaya yang tersedia di AWS Cloud. Memahami beban kerja Amazon

RDS dan kesehatan operasional dapat membantu Anda menanggapi peristiwa, memperbaiki masalah, dan merencanakan peningkatan.

Praktik terbaik umum

Praktik terbaik berikut membantu Anda mendapatkan visibilitas yang cukup terhadap kesehatan beban kerja Amazon RDS Anda dan mengambil tindakan yang sesuai sebagai respons terhadap peristiwa operasional dan data pemantauan.

- Identifikasi KPI.Identifikasi indikator kinerja utama (KPI) berdasarkan hasil bisnis yang diinginkan. Evaluasi KPI untuk menentukan keberhasilan beban kerja. Misalnya, jika bisnis inti Anda adalah e-commerce, salah satu hasil bisnis yang Anda inginkan adalah bahwa e-shop Anda tersedia 24/7 bagi pelanggan Anda untuk berbelanja. Untuk mencapai hasil bisnis tersebut, Anda menentukan KPI ketersediaan untuk database Amazon RDS backend yang digunakan aplikasi e-shop Anda, dan menetapkan KPI dasar menjadi 99,99% setiap minggu. Mengevaluasi ketersediaan aktual KPI terhadap nilai dasar membantu Anda menentukan apakah Anda memenuhi ketersediaan database yang diinginkan 99,99% dan dengan demikian mencapai hasil bisnis memiliki layanan 24/7.
- Tentukan metrik beban kerja.Tentukan metrik beban kerja untuk mengukur jumlah dan kualitas beban kerja Amazon RDS Anda. Evaluasi metrik untuk menentukan apakah beban kerja mencapai hasil yang diinginkan, dan untuk memahami kesehatan beban kerja. Misalnya, untuk mengevaluasi KPI ketersediaan untuk instans DB Amazon RDS Anda, Anda harus mengukur metrik seperti waktu aktif dan waktu henti untuk instans DB. Anda kemudian dapat menggunakan metrik tersebut untuk menghitung ketersediaan KPI sebagai berikut:

  ```
  availability = uptime / (uptime + downtime)
  ```

  Metrik mewakili kumpulan titik data yang dipesan waktu. Metrik juga dapat mencakup dimensi, yang berguna dalam kategorisasi dan analisis.
- Kumpulkan dan analisis metrik beban kerja.Amazon RDS menghasilkan metrik dan log yang berbeda, tergantung pada konfigurasi Anda. Beberapa ini mewakili DB misalnya peristiwa, counter, atau statistik sepertidb.Cache.innoDB_buffer_pool_hits. Metrik lain berasal dari sistem operasi, sepertimemory.Total, yang mengukur jumlah total memori instans Amazon Elastic Compute Cloud (Amazon EC2) host. Alat pemantauan harus melakukan analisis reguler dan proaktif terhadap metrik yang dikumpulkan untuk mengidentifikasi tren dan menentukan apakah ada respons yang sesuai diperlukan.
- Tetapkan baseline metrik beban kerja.Tetapkan garis dasar untuk metrik untuk menentukan nilai yang diharapkan dan untuk mengidentifikasi ambang batas yang baik atau buruk. Misalnya, Anda mungkin menentukan baseline untukReadIOPShingga 1.000 di bawah operasi database normal.

Anda kemudian dapat menggunakan baseline ini untuk perbandingan dan untuk mengidentifikasi over-utilization. Jika metrik baru Anda secara konsisten menunjukkan bahwa IOPS baca berada dalam kisaran 2.000-3.000, Anda telah mengidentifikasi penyimpangan yang dapat memicu respons untuk penyelidikan, intervensi, dan peningkatan.

- Waspada saat hasil beban kerja berisiko.Ketika Anda menentukan bahwa hasil bisnis berisiko, naikkan peringatan. Anda kemudian dapat mengatasi masalah secara proaktif, sebelum memengaruhi pelanggan Anda, atau mengurangi dampak insiden secara tepat waktu.
- Identifikasi pola aktivitas yang diharapkan untuk beban kerja Anda.Berdasarkan garis dasar metrik Anda, tetapkan pola aktivitas beban kerja untuk mengidentifikasi perilaku tak terduga dan merespons dengan tindakan yang sesuai jika perlu.AWSmemberialat pemantauanyang menerapkan algoritma statistik dan pembelajaran mesin untuk menganalisis metrik dan mendeteksi anomali.
- Peringatan saat anomali beban kerja terdeteksi.Ketika anomali terdeteksi dalam operasi beban kerja Amazon RDS, naikkan peringatan sehingga Anda dapat merespons dengan tindakan yang sesuai jika perlu.
- Tinjau dan revisi KPI dan metrik. Konfirmasikan bahwa basis data Amazon RDS Anda memenuhi persyaratan yang ditentukan dan identifikasi area potensi peningkatan untuk mencapai tujuan bisnis Anda. Validasi efektivitas metrik yang diukur dan KPI yang dievaluasi, dan revisi jika perlu. Misalnya, Anda menetapkan KPI untuk jumlah optimal koneksi database bersamaan, dan Anda memantau metrik mengenai koneksi yang dicoba dan gagal serta utas pengguna yang dibuat dan sedang berjalan. Anda mungkin memiliki lebih banyak koneksi database daripada yang ditentukan oleh baseline KPI Anda. Dengan menganalisis metrik Anda saat ini, Anda dapat mendeteksi hasilnya tetapi Anda mungkin tidak dapat menentukan akar penyebabnya. Jika demikian, Anda harus merevisi metrik Anda dan menyertakan langkah-langkah pemantauan tambahan, seperti penghitung untuk kunci meja. Metrik baru akan membantu menentukan apakah peningkatan jumlah koneksi database disebabkan oleh kunci tabel yang tidak terduga.

Alat pemantauan

Kami menyarankan Anda menggunakan alat observabilitas, pemantauan, dan peringatan untuk:

- Dapatkan wawasan tentang kinerja lingkungan Amazon RDS Anda
- Mendeteksi perilaku yang tidak terduga dan mencurigakan
- Rencanakan kapasitas dan buat keputusan terdidik tentang mengalokasikan instans Amazon RDS
- Menganalisis metrik dan log untuk memprediksi potensi masalah secara proaktif
- Hasilkan peringatan saat ambang batas dilanggar untuk memecahkan masalah dan menyelesaikan masalah sebelum pengguna Anda terpengaruh

Anda memiliki opsi dan solusi berbeda untuk dipilih, termasuk alat dan layanan observabilitas dan pemantauan cloud-native yang disediakan AWS; solusi perangkat lunak sumber terbuka gratis; dan solusi pihak ketiga komersial untuk memantau instans Amazon RDS DB. Beberapa alat ini dibahas di bagian berikut.

Untuk menentukan alat mana yang paling sesuai dengan kebutuhan Anda, bandingkan fitur dan kemampuan masing-masing alat dengan persyaratan organisasi Anda. Kami juga menyarankan Anda mengevaluasi alat untuk kemudahan penyebaran, konfigurasi dan integrasi, pembaruan dan pemeliharaan perangkat lunak, metode penyebaran (misalnya, perangkat keras atau tanpa server), lisensi, harga, dan faktor lain yang spesifik untuk organisasi Anda.

Bagian

Alat yang disertakan dalam Amazon RDS

Amazon Relational Database Service (Amazon RDS) adalah layanan database terkelola di AWS Cloud. Karena Amazon RDS adalah layanan terkelola, ia membebaskan Anda dari sebagian besar tugas manajemen, seperti pencadangan basis data, sistem operasi (OS) dan instalasi perangkat lunak basis data, penambalan OS dan perangkat lunak, penyiapan ketersediaan tinggi, siklus hidup perangkat keras, dan operasi pusat data. AWS juga menyediakan seperangkat alat lengkap yang memungkinkan Anda membuat solusi observabilitas lengkap untuk instans Amazon RDS DB Anda.

Beberapa alat pemantauan disertakan, dikonfigurasi sebelumnya, dan diaktifkan secara otomatis di layanan Amazon RDS. Dua alat otomatis tersedia untuk Anda segera setelah Anda memulai instans Amazon RDS baru Anda:

- Status instans Amazon RDS memberikan detail tentang kesehatan instans DB Anda saat ini. Misalnya, kode status termasuk Available, Stopped, Creating, Backing-up, dan Failed. Anda dapat menggunakan konsol Amazon RDS, AWS Command Line Interface (AWS CLI), atau Amazon RDS API untuk melihat status instance. Untuk informasi selengkapnya, lihat Melihat status instans Amazon RDS DB di dokumentasi Amazon RDS.
- Rekomendasi Amazon RDS memberikan rekomendasi otomatis untuk instans DB, replika baca, dan grup parameter DB. Rekomendasi ini disediakan dengan menganalisis penggunaan instans DB, data kinerja, dan konfigurasi, dan disampaikan sebagai panduan. Misalnya, rekomendasi usang versi Engine menunjukkan bahwa instans DB Anda tidak menjalankan versi terbaru dari perangkat lunak basis data dan Anda harus memutakhirkan instans DB Anda untuk mendapatkan manfaat dari perbaikan keamanan terbaru dan peningkatan lainnya. Untuk informasi selengkapnya, lihat Melihat rekomendasi Amazon RDS di dokumentasi Amazon RDS.

CloudWatch ruang nama

Amazon RDS terintegrasi dengan Amazon CloudWatch, yang merupakan layanan pemantauan dan peringatan untuk sumber daya cloud dan aplikasi yang berjalan di AWS. Amazon RDS secara otomatis mengumpulkan metrik, file log, jejak, dan peristiwa tentang operasi, pemanfaatan, kinerja, dan kesehatan instans DB, dan mengirimkannya ke penyimpanan, analisis, dan CloudWatch peringatan jangka panjang.

Amazon RDS untuk MySQL dan Amazon RDS untuk MariaDB secara otomatis menerbitkan satu set metrik default dalam interval satu menit tanpa biaya tambahan. CloudWatch Metrik tersebut dikumpulkan ke dalam dua ruang nama, yang merupakan wadah untuk metrik:

- Namespace AWS/RDS menyertakan metrik tingkat instans DB. Contohnya termasuk `BinLogDiskUsage` (jumlah ruang disk yang ditempati oleh log biner), `CPUUtilization` (persentase pemanfaatan CPU), `DatabaseConnections` (jumlah koneksi jaringan klien ke instans DB), dan banyak lagi.
- Ruang nama AWS/penggunaan mencakup metrik penggunaan tingkat akun, yang digunakan untuk menentukan apakah Anda beroperasi dalam kuota layanan Amazon RDS Anda. Contohnya termasuk `DBInstances` (jumlah instans DB di akun AWS atau Wilayah Anda), `DBSubnetGroups` (jumlah grup subnet DB di AWS akun atau Wilayah Anda), dan `ManualSnapshots` (jumlah snapshot database yang dibuat secara manual di AWS akun atau Wilayah Anda).

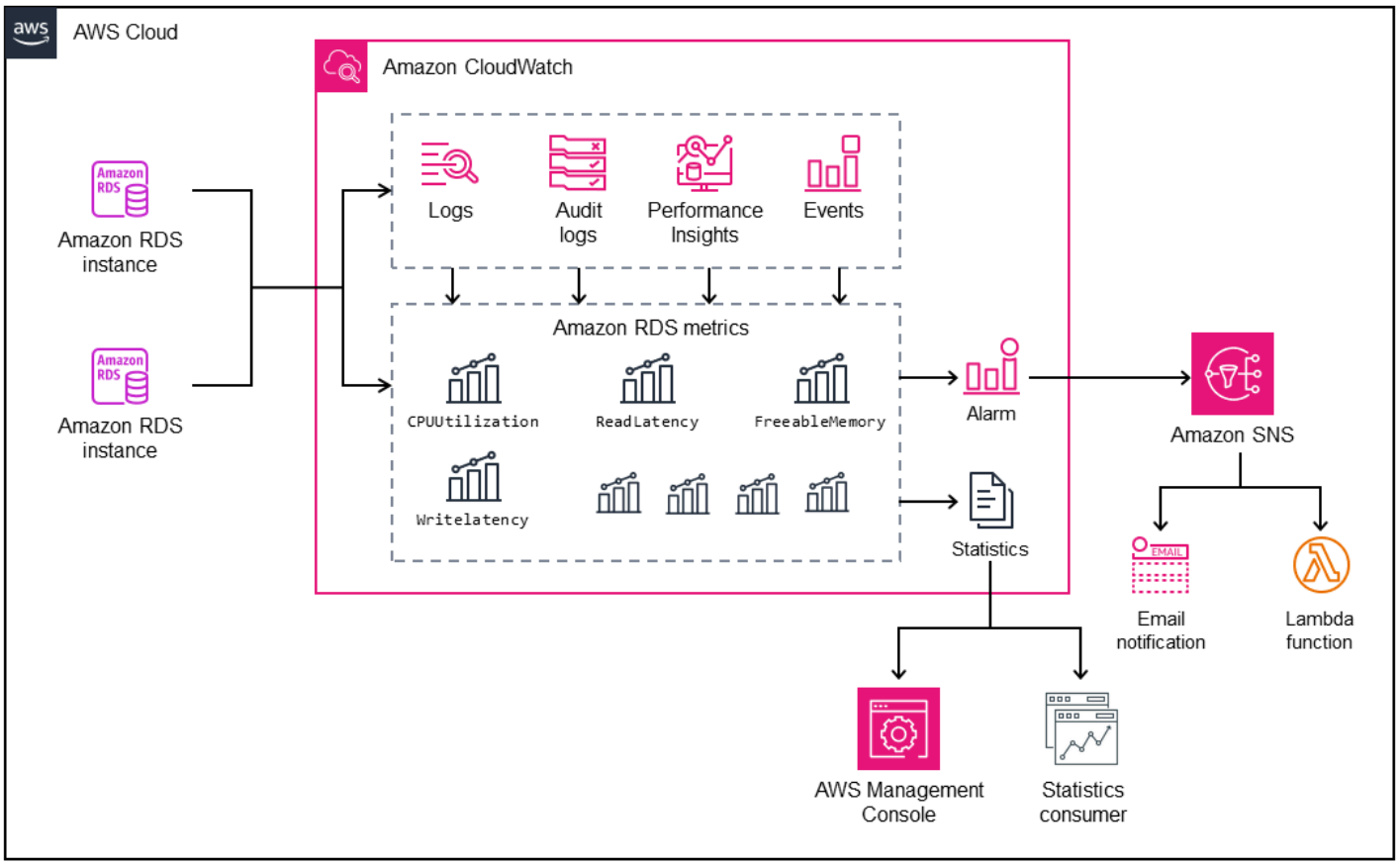
CloudWatch mempertahankan data metrik sebagai berikut:

- 3 jam: Metrik kustom resolusi tinggi dengan periode kurang dari 60 detik dipertahankan selama 3 jam. Setelah 3 jam, titik data dikumpulkan menjadi metrik periode 1 menit dan disimpan selama 15 hari.
- 15 hari: Poin data dengan jangka waktu 60 detik (1 menit) dipertahankan selama 15 hari. Setelah 15 hari, titik data dikumpulkan menjadi metrik periode 5 menit dan disimpan selama 63 hari.
- 63 hari: Titik data dengan jangka waktu 300 detik (5 menit) dipertahankan selama 63 hari. Setelah 63 hari, titik data dikumpulkan menjadi metrik periode 1 jam dan disimpan selama 15 bulan.
- 15 bulan: Titik data dengan jangka waktu 3.600 detik (1 jam) tersedia selama 15 bulan (455 hari).

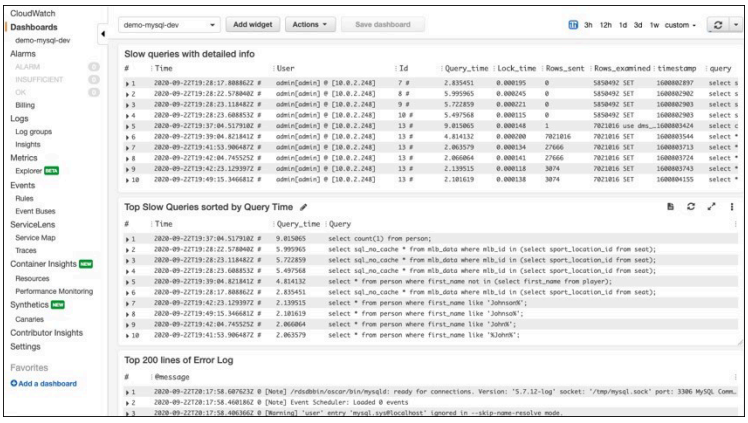
Untuk informasi selengkapnya, lihat Metrik dalam CloudWatch dokumentasi.

CloudWatch alarm dan dasbor

Anda dapat menggunakan CloudWatch alarm Amazon untuk menonton metrik Amazon RDS tertentu selama periode waktu tertentu. Misalnya, Anda dapat memantau`FreeStorageSpace`, dan kemudian melakukan satu atau beberapa tindakan jika nilai metrik melanggar ambang batas yang Anda tetapkan. Jika Anda menetapkan ambang batas ke 250 MB dan ruang penyimpanan kosong adalah 200 MB (kurang dari ambang batas), alarm akan diaktifkan dan dapat memicu tindakan untuk secara otomatis menyediakan penyimpanan tambahan untuk instans Amazon RDS DB. Alarm juga dapat mengirim SMS notifikasi ke DBA dengan menggunakan Amazon Simple Notification Service (Amazon SNS). Diagram berikut menggambarkan proses.

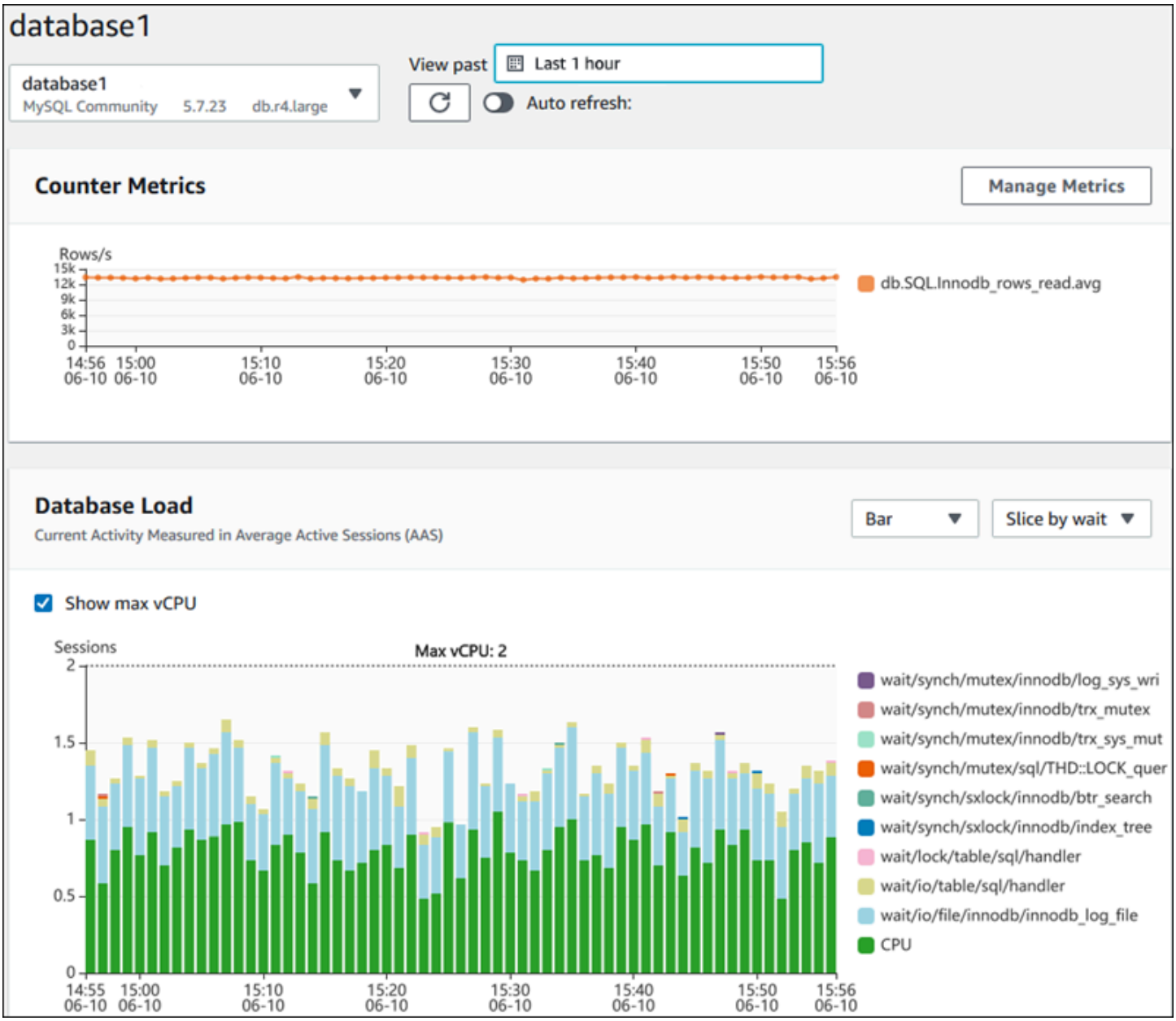


CloudWatch juga menyediakan dasbor, yang dapat Anda gunakan untuk membuat, menyesuaikan, berinteraksi, dan menyimpan tampilan (grafik) metrik yang disesuaikan. Anda juga dapat menggunakan Wawasan CloudWatch Log untuk membuat dasbor untuk memantau log kueri lambat dan log kesalahan, dan untuk menerima peringatan jika pola tertentu telah terdeteksi di log tersebut. Layar berikut menunjukkan contoh CloudWatch dasbor.



Wawasan Performa Amazon RDS

Amazon RDS Performance Insights adalah alat penyetelan dan pemantauan kinerja database yang memperluas fitur pemantauan Amazon RDS. Ini membantu Anda menganalisis kinerja database Anda dengan memvisualisasikan beban instans DB dan memfilter beban dengan menunggu, pernyataan SQL, host, atau pengguna. Alat ini menggabungkan beberapa metrik ke dalam satu grafik interaktif yang membantu Anda mengidentifikasi jenis kemacetan yang mungkin dimiliki instans DB Anda, seperti menunggu kunci, konsumsi CPU tinggi, atau latensi I/O, dan menentukan pernyataan SQL mana yang membuat hambatan. Layar berikut menunjukkan contoh visualisasi.



Anda harus mengaktifkan Performance Insights selama proses pembuatan instans DB untuk mengumpulkan metrik instans Amazon RDS DB di akun Anda. Tingkat gratis mencakup tujuh hari riwayat data kinerja dan satu juta permintaan API per bulan. Secara opsional, Anda dapat membeli periode retensi yang lebih lama. Untuk informasi harga selengkapnya, lihat Harga Wawasan Performa.

Untuk informasi tentang cara menggunakan Performance Insights untuk memantau instans DB, lihat bagian pemantauan instans DB nanti di panduan ini.

Performance Insights secara otomatis menerbitkan metrik ke. CloudWatch Selain menggunakan alat Performance Insights, Anda dapat memanfaatkan fitur tambahan yang CloudWatch disediakan. Anda dapat memeriksa metrik Performance Insights menggunakan CloudWatch konsol, the AWS CLI, atau API. CloudWatch Anda juga dapat menambahkan CloudWatch alarm, seperti halnya metrik lainnya. Misalnya, Anda mungkin ingin memicu pemberitahuan SMS ke DBA atau mengambil tindakan korektif jika `DBLoad` metrik melanggar nilai ambang batas yang Anda tetapkan. Anda juga dapat menambahkan metrik Performance Insights ke dasbor yang ada. CloudWatch

Pemantauan yang Ditingkatkan

Enhanced Monitoring adalah alat yang menangkap metrik secara real time untuk sistem operasi (OS) yang dijalankan instans Amazon RDS DB Anda. Metrik ini memberikan granularitas hingga satu detik untuk CPU, memori, Amazon RDS dan proses OS, sistem file, dan data I/O disk, antara lain. Anda dapat mengakses dan menganalisis metrik ini di konsol Amazon RDS. Seperti halnya Performance Insights, metrik Enhanced Monitoring dikirimkan dari Amazon RDS ke CloudWatch, di mana Anda dapat memanfaatkan fitur tambahan seperti pelestarian metrik jangka panjang untuk analisis, membuat filter metrik, menampilkan grafik di dasbor, dan menyiapkan alarm. CloudWatch Secara default, Enhanced Monitoring dinonaktifkan saat Anda membuat instans Amazon RDS DB baru. Anda dapat mengaktifkan fitur saat Anda membuat atau memodifikasi instans DB. Harga didasarkan pada jumlah data yang ditransfer dari Amazon RDS ke CloudWatch Log, dan tarif penyimpanan. Bergantung pada perincian dan jumlah instans DB di mana Enhanced Monitoring diaktifkan, beberapa bagian data pemantauan dapat dimasukkan dalam tingkat bebas CloudWatch Log. Untuk detail harga lengkap, lihat CloudWatch Harga Amazon. Untuk informasi selengkapnya tentang alat ini, lihat dokumentasi Amazon RDS dan FAQ Pemantauan yang Ditingkatkan.

AWS Layanan tambahan

AWS menyediakan beberapa layanan pendukung, yang juga terintegrasi dengan Amazon RDS dan CloudWatch, untuk lebih meningkatkan pengamatan database Anda. Ini termasuk Amazon EventBridge, Amazon CloudWatch Logs, dan AWS CloudTrail.

- Amazon EventBridge adalah bus acara tanpa server yang dapat menerima, memfilter, mengubah, merutekan, dan mengirimkan acara dari aplikasi dan AWS sumber daya Anda, termasuk instans Amazon RDS DB Anda. Acara Amazon RDS menunjukkan perubahan di lingkungan Amazon RDS. Misalnya, ketika instans DB mengubah statusnya dari Tersedia menjadi Berhenti, Amazon RDS menghasilkan acara `RDS-EVENT-0087 / The DB instance has been stopped` tersebut. Amazon RDS mengirimkan acara ke CloudWatch Acara dan EventBridge dalam waktu dekat. Menggunakan EventBridge dan CloudWatch Acara, Anda dapat menentukan aturan untuk mengirim peringatan pada peristiwa Amazon RDS tertentu yang menarik dan mengotomatiskan tindakan yang akan diambil saat acara cocok dengan aturan. Berbagai target tersedia sebagai respons terhadap suatu peristiwa, seperti AWS Lambda fungsi yang dapat melakukan tindakan korektif, atau topik Amazon SNS yang dapat mengirim email atau SMS untuk memberi tahu DBA DevOps atau insinyur tentang acara tersebut.
- Amazon CloudWatch Logs adalah layanan yang memusatkan penyimpanan file log dari semua aplikasi, sistem, dan AWS layanan Anda, termasuk Amazon RDS for MySQL dan MariaDB instans dan. AWS CloudTrail Jika Anda mengaktifkan fitur untuk instans DB Anda, Amazon RDS secara otomatis menerbitkan log berikut ke Log: CloudWatch
  - Log kesalahan
  - Log kueri lambat
  - Log umum
  - Log audit

  Anda dapat menggunakan Wawasan CloudWatch Log untuk menanyakan dan menganalisis data log. Fitur ini mencakup bahasa kueri yang dibuat khusus yang membantu Anda mencari peristiwa log yang cocok dengan pola, yang Anda tentukan. Misalnya, Anda dapat melacak korupsi tabel di instance MySQL DB Anda dengan memantau file log kesalahan untuk pola berikut:. `"ERROR 1034 (HY000): Incorrect key file for table '*'; try to repair it OR Table * is marked as crashed"` Data log yang difilter dapat diubah menjadi CloudWatch metrik. Anda kemudian dapat menggunakan metrik untuk membuat dasbor dengan grafik atau data tabular, atau menyetel alarm jika nilai ambang batas yang ditentukan dilanggar. Ini sangat berguna saat menggunakan log audit, karena Anda dapat secara otomatis memantau, mengirim peringatan, dan

mengambil tindakan korektif jika ada perilaku yang tidak terduga atau mencurigakan terdeteksi. Anda dapat mengakses dan mengelola log database menggunakan AWS Management Console, Amazon RDS API, atau AWS SDK for CloudWatch Logs. AWS CLI

- AWS CloudTrailmencatat dan terus memantau aktivitas pengguna dan API di akun AWS Anda. Ini membantu Anda dengan audit, pemantauan keamanan, dan pemecahan masalah operasional Amazon RDS for MySQL atau instans DB MariaDB Anda. CloudTrail terintegrasi dengan Amazon RDS. Semua tindakan dapat dicatat, dan CloudTrail menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Amazon RDS. Misalnya, saat pengguna membuat instans Amazon RDS DB baru, peristiwa terdeteksi, dan log menyertakan informasi tentang tindakan yang diminta (`"eventName": "CreateDBInstance"`), tanggal dan waktu tindakan (`"eventTime": "2022-07-30T22:14:06Z"`), parameter permintaan (`"requestParameters": {"dBInstanceIdentifier": "test-instance", "engine": "mysql", "dBInstanceClass": "db.m6g.large"}`), dan sebagainya. Peristiwa yang dicatat oleh CloudTrail menyertakan panggilan dari konsol Amazon RDS dan panggilan dari kode yang menggunakan Amazon RDS API.

Alat pemantauan pihak ketiga

Dalam beberapa skenario, selain rangkaian lengkap alat observabilitas dan pemantauan cloud-native yang AWS menyediakan Amazon RDS, Anda mungkin ingin menggunakan alat pemantauan dari vendor perangkat lunak lain. Skenario tersebut mencakup penerapan hibrid, di mana Anda mungkin memiliki sejumlah database yang berjalan di pusat data lokal dan kumpulan database lain yang berjalan di pusat data lokal. AWS Cloud Jika Anda telah membuat solusi observabilitas perusahaan, Anda mungkin ingin terus menggunakan alat yang ada dan memperluasnya ke penerapan AWS Cloud Anda. Tantangan dalam menyiapkan solusi pemantauan pihak ketiga seringkali terletak pada perlindungan yang diberlakukan oleh Amazon RDS sebagai layanan yang dikelola cloud. Misalnya, Anda tidak dapat menginstal perangkat lunak agen pada sistem operasi host yang menjalankan instans DB, karena akses ke mesin host database ditolak. Namun, Anda dapat mengintegrasikan banyak solusi pemantauan pihak ketiga dengan Amazon RDS dengan membangun di atas CloudWatch dan AWS Cloud layanan lainnya. Misalnya, metrik, log, peristiwa, dan jejak Amazon RDS dapat diekspor dan kemudian diimpor ke alat pemantauan pihak ketiga untuk analisis, visualisasi, dan peringatan lebih lanjut. Beberapa solusi pihak ketiga ini termasuk Prometheus, Grafana, dan Percona.

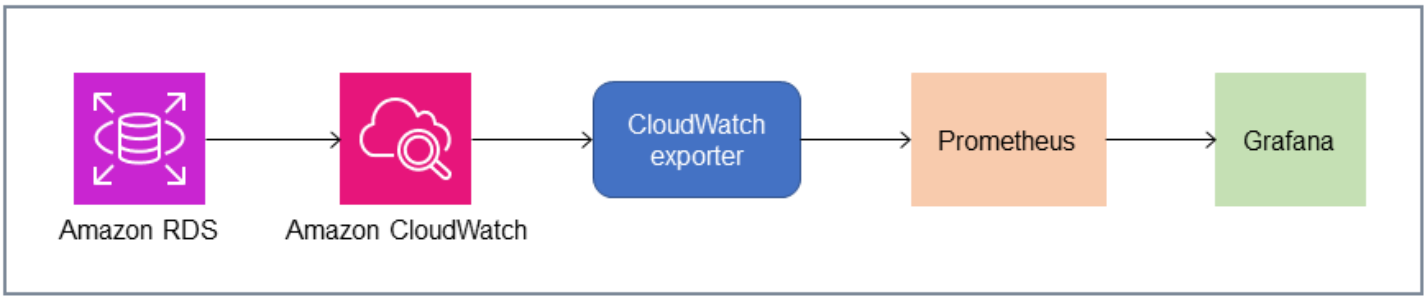
Prometheus dan Grafana

Prometheus adalah solusi pemantauan sumber terbuka yang mengumpulkan metrik dari target yang dikonfigurasi pada interval tertentu. Ini adalah solusi pemantauan tujuan umum yang dapat memantau aplikasi atau layanan apa pun. Saat Anda memantau instans Amazon RDS DB, CloudWatch kumpulkan metrik dari Amazon RDS. Metrik kemudian diekspor ke server Prometheus dengan menggunakan eksportir open-source seperti eksportir YACE atau Eksportir. CloudWatch

- Eksportir YACE mengoptimalkan tugas ekspor data dengan mengambil beberapa metrik dalam satu permintaan ke API. CloudWatch Setelah metrik disimpan di server Prometheus, server mengevaluasi ekspresi aturan dan dapat menghasilkan peringatan ketika kondisi tertentu diamati.
- CloudWatch Eksportir secara resmi dikelola oleh Prometheus. Ini mengambil CloudWatch metrik melalui CloudWatch API dan menyimpannya di server Prometheus dalam format yang kompatibel dengan Prometheus, dengan menggunakan permintaan REST API ke titik akhir HTTP.

Saat Anda memilih eksportir, rancang model penerapan Anda, dan konfigurasikan instance eksportir, pertimbangkan CloudWatchdan layanan CloudWatch Log serta kuota API, karena ekspor metrik CloudWatch ke server Prometheus diimplementasikan di atas API. CloudWatch Misalnya, menerapkan beberapa instance CloudWatch Eksportir dalam satu Akun AWS dan Wilayah untuk memantau ratusan instans Amazon RDS DB dapat mengakibatkan kesalahan pelambatan () dan kesalahan kode 400. ThrottlingException Untuk mengatasi keterbatasan tersebut, pertimbangkan untuk menggunakan eksportir YACE, yang dioptimalkan untuk mengumpulkan hingga 500 metrik berbeda dalam satu permintaan. Selain itu, untuk menerapkan sejumlah besar instans Amazon RDS DB, Anda harus mempertimbangkan untuk menggunakan beberapa Akun AWS, alih-alih memusatkan beban kerja menjadi satu Akun AWS, dan membatasi jumlah instans eksportir di masing-masing instans. Akun AWS

Peringatan dihasilkan oleh server Prometheus dan ditangani oleh Alertmanager. Alat ini menangani deduplikasi, pengelompokan, dan perutean peringatan ke penerima yang benar seperti email, SMS, atau Slack, atau memulai tindakan respons otomatis. Alat open-source lain yang disebut Grafana menampilkan visualisasi untuk metrik ini. Grafana menyediakan widget visualisasi yang kaya, seperti grafik canggih, dasbor dinamis, dan fitur analitik seperti kueri ad-hoc dan penelusuran dinamis. Hal ini juga dapat mencari dan menganalisis log, dan termasuk fitur peringatan untuk terus mengevaluasi metrik dan log, dan mengirim pemberitahuan ketika data cocok dengan aturan peringatan.



Percona

Percona Monitoring and Management (PMM) adalah solusi pemantauan, manajemen, dan observabilitas database open-source gratis untuk MySQL dan MariaDB. PMM mengumpulkan ribuan metrik kinerja dari instans DB dan hostnya. Ini menyediakan UI web untuk memvisualisasikan data di dasbor dan fitur tambahan seperti penasihat otomatis untuk penilaian kesehatan database. Anda dapat menggunakan PMM untuk memantau Amazon RDS. Namun, klien PMM (agen) tidak diinstal pada host yang mendasari instans Amazon RDS DB, karena tidak memiliki akses ke host. Sebagai gantinya, alat ini terhubung ke instans Amazon RDS DB, kueri statistik server, skema sys`INFORMATION_SCHEMA`, dan Skema Kinerja, dan menggunakan CloudWatch API untuk memperoleh metrik, log, peristiwa, dan jejak. PMM memerlukan kunci akses pengguna AWS Identity and Access Management (IAM) (peran IAM) dan secara otomatis menemukan instans Amazon RDS DB yang tersedia untuk pemantauan. Alat PMM diprofilkan untuk pemantauan basis data dan mengumpulkan lebih banyak metrik khusus database daripada Prometheus. Untuk menggunakan dasbor PMM Query Analytics, Anda harus mengonfigurasi Skema Kinerja sebagai sumber kueri, karena agen Query Analytics tidak diinstal untuk Amazon RDS dan tidak dapat membaca log kueri lambat. Sebagai gantinya, ia menanyakan `performance_schema` dari instance MySQL dan MariaDB DB secara langsung untuk mendapatkan metrik. Salah satu fitur yang menonjol dari PMM adalah kemampuannya untuk memperingatkan dan menyarankan DBA tentang masalah yang diidentifikasi alat dalam database mereka. PMM menawarkan serangkaian pemeriksaan yang dapat mendeteksi ancaman keamanan umum, penurunan kinerja, kehilangan data, dan korupsi data.

Selain alat-alat ini, ada beberapa solusi observabilitas dan pemantauan komersial yang tersedia di pasar yang dapat diintegrasikan dengan Amazon RDS. Contohnya termasuk Datadog Database Monitoring, Dynatrace Amazon RDS monitoring, dan Database Monitoring. AppDynamics

Pemantauan instans DB

SEBUAHDB misalnyaadalah blok bangunan dasar Amazon RDS. Ini adalah lingkungan database terisolasi yang berjalan di cloud. Untuk database MySQL dan MariaDB, instance DB adalahmysqldprogram, juga dikenal sebagai server MySQL, yang mencakup beberapa benang dan komponen seperti parser SQL, optimizer query, thread/koneksi handler, sistem dan status variabel, dan satu atau lebih mesin penyimpanan pluggable. Setiap mesin penyimpanan dirancang untuk mendukung kasus penggunaan khusus. Mesin penyimpanan default dan yang direkomendasikan adalahInnoDB, yang merupakan transaksional, tujuan umum, mesin database relasional yang sesuai dengan atomicity, konsistensi, isolasi, daya tahan (ACID) model. Fitur InnoDBstruktur dalam memori(buffer pool, ubah buffer, indeks hash adaptif, buffer log) sertastruktur on-disk(tablespace, tabel, indeks, undo log, redo log, file buffer doublewrite). Untuk memastikan bahwa database Anda melekat erat dengan model ACID,Mesin penyimpanan InnoDB mengimplementasikan berbagai kemampuanuntuk melindungi data Anda, termasuk transaksi, komit, rollback, pemulihan kerusakan, penguncian tingkat baris, dan kontrol konkurensi multiversi (MVCC).

Semua komponen internal instans DB ini bekerja bersama untuk membantu menjaga ketersediaan, integritas, dan keamanan data Anda pada tingkat kinerja yang diharapkan dan memuaskan. Bergantung pada beban kerja Anda, setiap komponen dan fitur mungkin memaksakan permintaan sumber daya pada subsistem CPU, memori, jaringan, dan penyimpanan. Ketika lonjakan permintaan untuk sumber daya tertentu melebihi kapasitas yang disediakan atau batas perangkat lunak untuk sumber daya tersebut (diberlakukan baik oleh parameter konfigurasi atau oleh desain perangkat lunak), instans DB dapat mengalami penurunan kinerja atau ketidaktersediaan dan korupsi lengkap. Oleh karena itu, sangat penting untuk mengukur dan memantau komponen internal ini, membandingkannya dengan nilai dasar yang ditentukan, dan menghasilkan peringatan jika nilai yang dipantau menyimpang dari nilai yang diharapkan.

Seperti dijelaskan sebelumnya, Anda dapat menggunakan yang berbedaalatuntuk memantau instance MySQL dan MariaDB Anda. Kami menyarankan Anda menggunakan Amazon RDS Performance Insights danCloudWatchalat untuk pemantauan dan peringatan, karena alat ini terintegrasi dengan Amazon RDS, mengumpulkan metrik resolusi tinggi, menyajikan informasi kinerja terbaru dalam waktu dekat, dan menghasilkan alarm.

Terlepas dari alat pemantauan pilihan Anda, kami sarankan Andanyalakan Skema Kinerjadi instans MySQL dan MariaDB DB Anda. YangSkema Kinerjaadalah fitur opsional untuk memantau pengoperasian server MySQL (instans DB) pada tingkat rendah, dan dirancang untuk memiliki dampak minimal pada kinerja database secara keseluruhan. Anda dapat mengelola fitur ini dengan

menggunakan`performance_schema`parameter. Meskipun parameter ini opsional, Anda harus menggunakannya untuk mengumpulkan metrik per SQL resolusi tinggi (satu detik), metrik sesi aktif, peristiwa tunggu, dan informasi pemantauan tingkat rendah lainnya yang terperinci, yang dikumpulkan oleh Amazon RDS Performance Insights.

Bagian

- Metrik Wawasan Performa untuk instans DB
- CloudWatchmetrik untuk instans DB
- Mempublikasikan metrik Wawasan Kinerja keCloudWatch

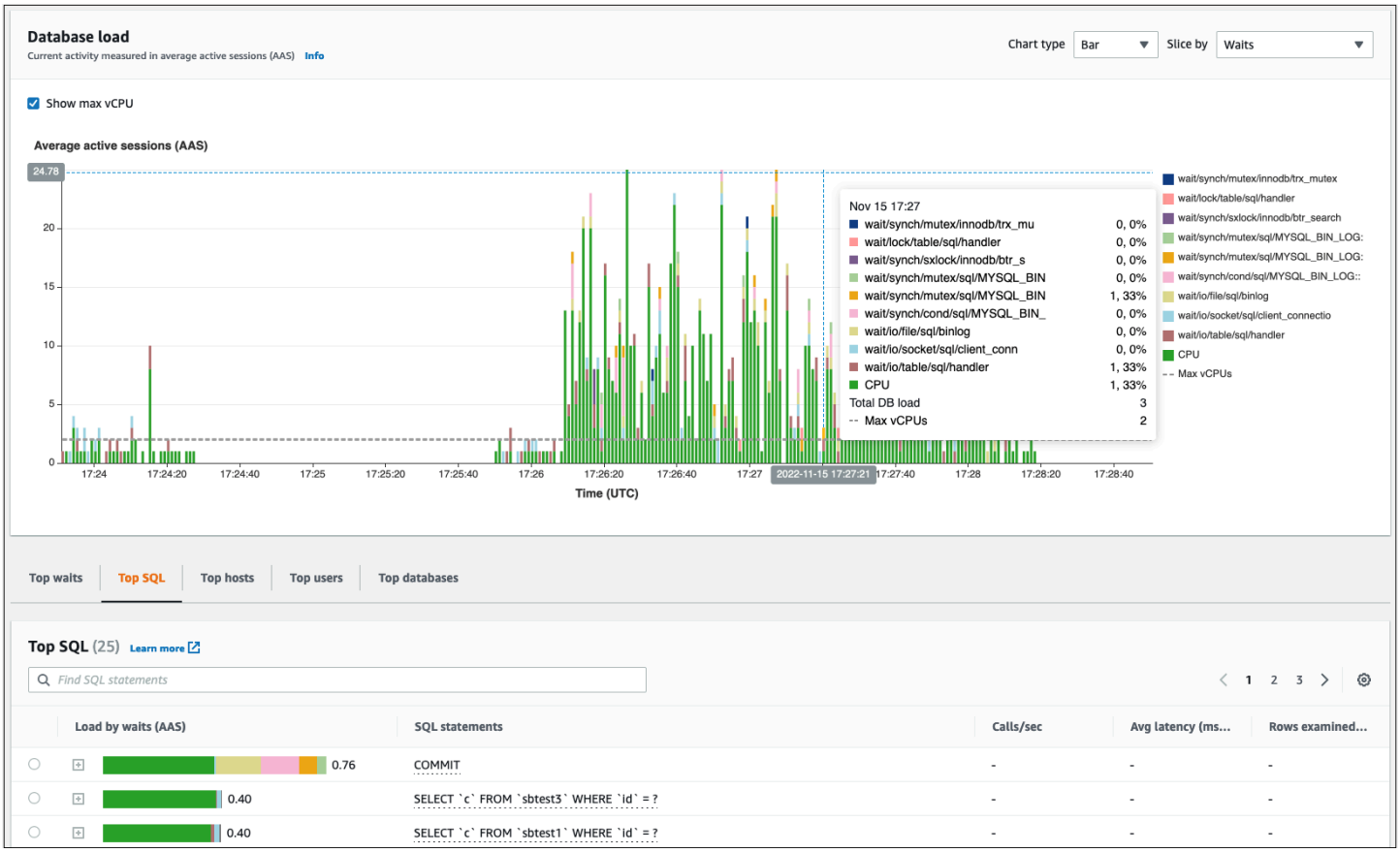
Metrik Wawasan Performa untuk instans DB

Performance Insights memantau berbagai jenis metrik, seperti yang dibahas di bagian berikut.

Muatan database

Beban database (`DBLoad`) adalah metrik kunci dalam Wawasan Kinerja yang mengukur tingkat aktivitas dalam database Anda. Itu dikumpulkan setiap detik dan secara otomatis dipublikasikan ke AmazonCloudWatch. Ini mewakili aktivitas instans DB dalam sesi aktif rata-rata (AAS), yang merupakan jumlah sesi yang secara bersamaan menjalankan kueri SQL. Yang`DBLoad`metrik berbeda dari metrik deret waktu lainnya, karena dapat ditafsirkan dengan menggunakan salah satu dari lima dimensi berikut: waits, SQL, host, users, dan database. Dimensi ini adalah subkategori dari`DBLoad`metrik. Anda dapat menggunakannya sebagaiiris olehkategori untuk mewakili karakteristik yang berbeda dari beban database. Untuk penjelasan rinci tentang bagaimana kita menghitung beban database, lihatBeban databasedalam dokumentasi Amazon RDS.

Ilustrasi layar berikut menunjukkan alat Performance Insights.



Dimensi

- Tunggu acaraadalah kondisi bahwa sesi database menunggu sumber daya atau operasi lain untuk menyelesaikan untuk melanjutkan pemrosesannya. Jika Anda menjalankan pernyataan SQL seperti`SELECT * FROM big_table`dan jika tabel ini jauh lebih besar dari pangkalan buffer InnoDB yang dialokasikan, sesi Anda kemungkinan besar akan menunggu`wait/io/file/innodb/innodb_data_file`menunggu peristiwa, yang disebabkan oleh fisik I/O operasi pada file data. Menunggu peristiwa adalah dimensi penting untuk pemantauan database, karena mereka menunjukkan kemungkinan hambatan kinerja. Tunggu peristiwa menunjukkan sumber daya dan operasi yang pernyataan SQL yang Anda jalankan dalam sesi menghabiskan waktu paling lama menunggu. Misalnya,`wait/synch/mutex/innodb/trx_sys_mutex`peristiwa terjadi ketika ada aktivitas database yang tinggi dengan sejumlah besar transaksi, dan`wait/synch/mutex/innodb/buf_pool_mutex`acara terjadi ketika thread telah mengakuisisi kunci pada kolam penyangga InnoDB untuk mengakses halaman dalam memori. Untuk informasi tentang semua acara tunggu MySQL dan MariaDB, lihatTunggu Tabel Ringkasan Acaradalam dokumentasi MySQL. Untuk memahami cara menafsirkan nama instrumen, lihatKonvensi Penamaan Instrumen Skema Kinerjadalam dokumentasi MySQL.

- SQLmenunjukkan pernyataan SQL mana yang berkontribusi paling besar terhadap total beban database. YangDimensi teratas, yang terletak di bawahBeban databasegrafik di Amazon RDS Performance Insights, bersifat interaktif. Anda dapat memperoleh daftar rinci dari peristiwa menunggu yang terkait dengan pernyataan SQL dengan mengklik bar diMuat dengan menunggu (AAS)kolom. Bila Anda memilih pernyataan SQL dalam daftar, Performance Insights menampilkan peristiwa tunggu terkait diBeban databasegrafik dan teks pernyataan SQL diTeks SQLbagian. Statistik SQL ditampilkan di sisi kananDimensi teratasmeja.
- Hostmenunjukkan nama host dari klien yang terhubung. Dimensi ini membantu Anda mengidentifikasi host klien mana yang mengirim sebagian besar beban ke database.
- Penggunakelompokkan beban DB oleh pengguna yang masuk ke database.
- Databasemengelompokkan beban DB dengan nama database klien terhubung ke.

Metrik penghitung

Metrik penghitung adalah metrik kumulatif yang nilainya hanya dapat meningkat atau diatur ulang ke nol saat instans DB dimulai ulang. Nilai metrik penghitung tidak dapat dikurangi ke nilai sebelumnya. Metrik ini mewakili satu penghitung yang meningkat secara monoton.

- Penghitung asliadalah metrik yang ditentukan oleh mesin database dan bukan oleh Amazon RDS. Misalnya:
  - `SQL.Innodb_rows_inserted`merupakan jumlah baris dimasukkan ke dalam tabel InnoDB.
  - `SQL.Select_scan`mewakili jumlah bergabung yang menyelesaikan scan penuh dari tabel pertama.
  - `Cache.Innodb_buffer_pool_reads`mewakili jumlah pembacaan logis bahwa mesin InnoDB tidak bisa mengambil dari buffer pool dan harus membaca langsung dari disk.
  - `Cache.Innodb_buffer_pool_read_requests`mewakili jumlah permintaan baca logis.

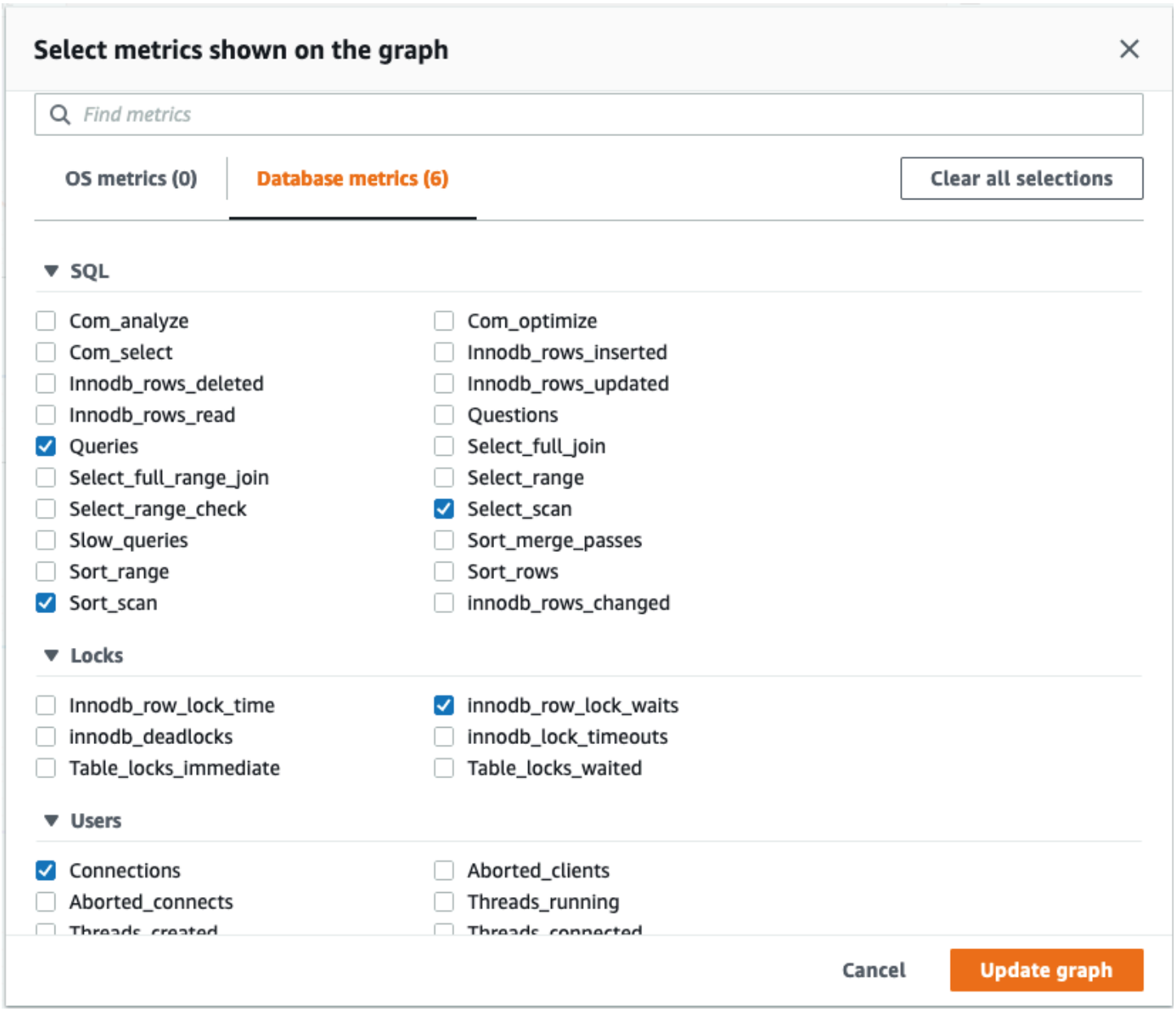
  Untuk definisi semua metrik native, lihatVariabel Status Serverdalam dokumentasi MySQL.
- Penghitung non-pribumididefinisikan oleh Amazon RDS. Anda dapat memperoleh metrik ini baik dengan menggunakan kueri tertentu atau mendapatkannya dengan menggunakan dua atau lebih metrik asli dalam perhitungan. Metrik penghitung non-native dapat mewakili latensi, rasio, atau hit rate. Misalnya:
  - `Cache.innoDB_buffer_pool_hits`mewakili jumlah operasi baca yang InnoDB bisa mengambil dari buffer pool tanpa memanfaatkan disk. Hal ini dihitung dari metrik counter asli sebagai berikut:

```
db.Cache.Innodb_buffer_pool_read_requests - db.Cache.Innodb_buffer_pool_reads
```

- `IO.innoDB_datafile_writes_to_disk`merupakan jumlah file data InnoDB menulis operasi ke disk. Ini menangkap hanya operasi pada file data-tidak doublewrite atau mengulang log menulis operasi. Hal ini dihitung sebagai berikut:

  ```
  db.IO.Innodb_data_writes - db.IO.Innodb_log_writes - db.IO.Innodb_dblwr_writes
  ```

Anda dapat memvisualisasikan metrik instans DB secara langsung di dasbor Performance Insights. PilihKelola Metrik, pilihMetrik databasetab, dan kemudian pilih metrik yang menarik, seperti yang ditunjukkan pada ilustrasi berikut.



PilihPerbarui grafiktombol untuk menampilkan metrik yang Anda pilih, seperti yang ditunjukkan pada ilustrasi berikut.

Statistik SQL

Performance Insights mengumpulkan metrik terkait kinerja tentang kueri SQL untuk setiap detik yang dijalankan kueri dan untuk setiap panggilan SQL. Secara umum, Performance Insights mengumpulkanStatistik SQLpada tingkat pernyataan dan mencerna. Namun, untuk instans MariaDB dan MySQL DB, statistik dikumpulkan hanya pada tingkat intisari.

- Statistik digest adalah metrik komposit dari semua kueri yang memiliki pola yang sama tetapi akhirnya memiliki nilai literal yang berbeda. Intisari menggantikan nilai literal tertentu dengan variabel; misalnya:

```
SELECT department_id, department_name FROM departments WHERE location_id = ?
```

- Ada metrik yang mewakili statistikper detikuntuk setiap pernyataan SQL dicerna. Sebagai contoh,`sql_tokenized.stats.count_star_per_sec`mewakili panggilan per detik (yaitu, berapa kali per detik pernyataan SQL telah dijalankan).

- Wawasan Kinerja juga mencakup metrik yang menyediakanper panggilanstatistik untuk pernyataan SQL. Sebagai contoh,`sql_tokenized.stats.sum_timer_wait_per_call`menunjukkan latensi rata-rata pernyataan SQL per panggilan, dalam milidetik.

Statistik SQL tersedia di dasbor Performance Insights, diSQL Teratastab dariDimensi teratasmeja.

CloudWatchmetrik untuk instans DB

AmazonCloudWatchjuga berisi metrik yang diterbitkan Amazon RDS secara otomatis. Metrik yang berada di`AWS/RDS`namespace adalahmetrik tingkat instans, yang mengacu pada instans Amazon RDS (layanan) (yaitu, lingkungan database terisolasi yang berjalan di cloud) daripada instans DB dalam arti ketatmysqldproses. Oleh karena itu, kebanyakan dari merekametrik defaulttermasuk dalam kategori metrik OS, dalam definisi istilah yang ketat. Contohnya meliputi:`CPUUtilization`,`WriteIOPS`,`SwapUsage`, dan lainnya. Namun demikian, ada beberapa metrik instans DB yang berlaku untuk MariaDB dan MySQL:

- `BinLogDiskUsage`- Jumlah ruang disk yang ditempati oleh log biner.
- `DatabaseConnections`- Jumlah koneksi jaringan klien ke instans DB.
- `ReplicaLag`- Jumlah waktu instance DB replika baca tertinggal dari instans DB sumber.

Mempublikasikan metrik Wawasan Kinerja keCloudWatch

Amazon RDS Performance Insights memantau sebagian besar metrik dan dimensi instans DB dan membuatnya tersedia melalui dasbor Performance Insights diAWSKonsol Manajemen. Dasbor ini sangat cocok untuk pemecahan masalah database dan analisis akar penyebab. Namun, tidak mungkin membuat alarm di Wawasan Kinerja untuk metrik terkait kinerja. Untuk membuat alarm berdasarkan metrik Wawasan Kinerja, Anda harus memindahkan metrik tersebutCloudWatch.

Memiliki metrik diCloudWatchjuga memberi Anda akses ke fitur pemantauan tingkat lanjut seperti CloudWatch deteksi anomali, matematika metrik, dan statistik, dan Anda dapat mengekspor metrik ke alat pemantauan eksternal seperti Prometheus dan Grafana.

Metrik Wawasan Performa tidak dipublikasikan secara otomatis CloudWatch (kecuali untuk metrik dbLoad). Untuk mempublikasikan metrik instans DB dari Performance Insights ke CloudWatch, Anda dapat menggunakan API Wawasan Kinerja untuk mengambil metrik, dan CloudWatch API untuk mempublikasikan metrik ke CloudWatch. Untuk mengotomatiskan proses, Anda dapat membuat fungsi Lambda dan menjadwalkannya di Amazon EventBridge untuk berjalan pada periode waktu yang ditentukan–misalnya, setiap dua menit. Anda dapat menentukan metrik Wawasan Kinerja yang ingin Anda publikasikan CloudWatch. Fungsi Lambda mendapatkan metrik tersebut dari semua instans Amazon RDS yang mengaktifkan Performance Insights, dan menyimpan metrik CloudWatch. Untuk informasi lebih lanjut tentang proses ini, lihat posting blog tentang memberikan metrik penghitung Wawasan Kinerja ke CloudWatch.

Pemantauan OS

Instans DB di Amazon RDS untuk MySQL atau MariaDB berjalan pada sistem operasi Linux, yang menggunakan sumber daya sistem yang mendasarinya: CPU, memori, jaringan, dan penyimpanan.

```
MySQL [(none)]> SHOW variables LIKE 'version%';
+-------------------------+---------------------+
| Variable_name           | Value               |
+-------------------------+---------------------+
| version                 | 8.0.28              |
| version_comment         | Source distribution |
| version_compile_machine | aarch64             |
| version_compile_os      | Linux               |
| version_compile_zlib    | 1.2.11              |
+-------------------------+---------------------+
5 rows in set (0.00 sec)
```

Kinerja keseluruhan database Anda dan sistem operasi yang mendasarinya sangat bergantung pada pemanfaatan sumber daya sistem. Misalnya, CPU adalah komponen kunci untuk kinerja sistem Anda, karena ia menjalankan instruksi perangkat lunak database dan mengelola sumber daya sistem lainnya. Jika CPU terlalu banyak digunakan (yaitu, jika beban membutuhkan lebih banyak daya CPU daripada yang disediakan untuk instans DB Anda), masalah ini akan memengaruhi kinerja dan stabilitas database Anda dan akibatnya aplikasi Anda.

Mesin database secara dinamis mengalokasikan dan membebaskan memori. Ketika tidak ada cukup memori dalam RAM untuk melakukan pekerjaan saat ini, sistem menulis halaman memori ke memori swap, yang berada pada disk. Karena disk jauh lebih lambat daripada memori, bahkan jika disk didasarkan pada teknologi SSD NVMe, alokasi memori yang berlebihan menyebabkan degradasi kinerja. Pemanfaatan memori yang tinggi menyebabkan peningkatan latensi respons database, karena ukuran file halaman tumbuh untuk mendukung memori tambahan. Jika alokasi memori begitu tinggi sehingga menghabiskan RAM dan ruang memori swap, layanan database mungkin menjadi tidak tersedia dan pengguna dapat mengamati kesalahan seperti`[ERROR] mysqld: Out of memory (Needed xyz bytes)`.

Sistem manajemen database MySQL dan MariaDB memanfaatkan subsistem penyimpanan, yang terdiri dari disk yang menyimpanstruktur on-diskseperti tabel, indeks, log biner, log ulang, undo log, dan file buffer doublewrite. Oleh karena itu, database, berbeda dengan jenis perangkat lunak lainnya, harus melakukan banyak aktivitas disk. Untuk pengoperasian database Anda yang optimal, penting

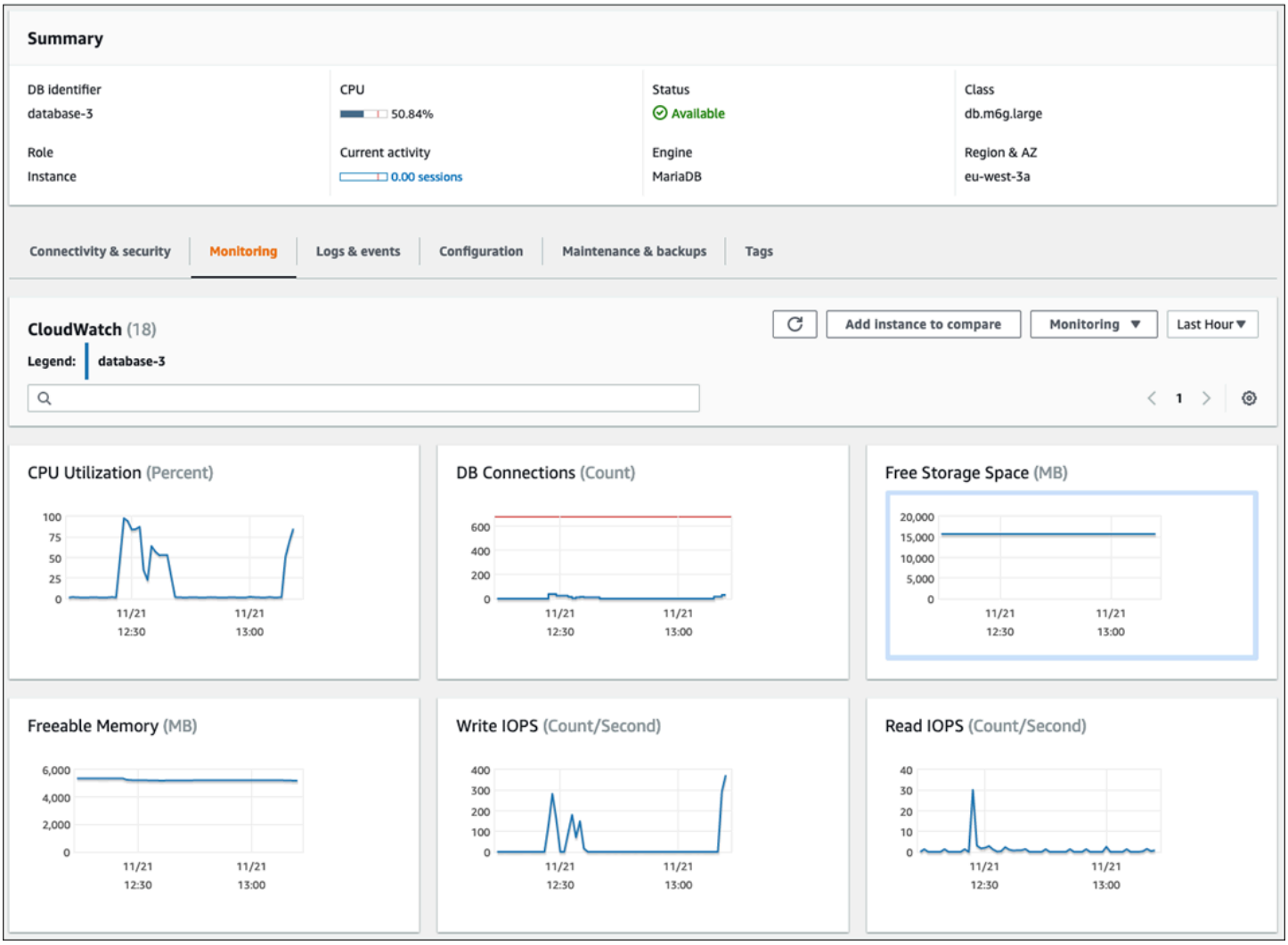
bagi Anda untuk memantau dan menyetel pemanfaatan I/O disk dan alokasi ruang disk. Kinerja database dapat terpengaruh ketika database mencapai batasan IOPS maksimum atau throughput yang didukung oleh disk. Misalnya, semburan akses acak yang disebabkan oleh pemindaian indeks dapat menyebabkan sejumlah besar operasi I/O per detik, yang pada akhirnya mungkin mencapai batasan penyimpanan yang mendasarinya. Pemindaian tabel penuh mungkin tidak mencapai batas IOPS, tetapi dapat menyebabkan throughput tinggi yang diukur dalam megabyte per detik. Sangat penting untuk memantau dan menghasilkan peringatan pada alokasi ruang disk, karena kesalahan seperti`OS error code 28: No space left on device`dapat menyebabkan tidak tersedianya dan korupsi database.

Amazon RDS menyediakan metrik secara real time untuk sistem operasi yang dijalankan instans DB Anda. Amazon RDS secara otomatis menerbitkan satu set metrik OS keCloudWatch. Metrik tersebut tersedia untuk Anda tampilkan dan dianalisis di konsol Amazon RDS danCloudWatchdasbor, dan Anda dapat mengatur alarm pada metrik yang dipilih diCloudWatch. Contohnya termasuk:

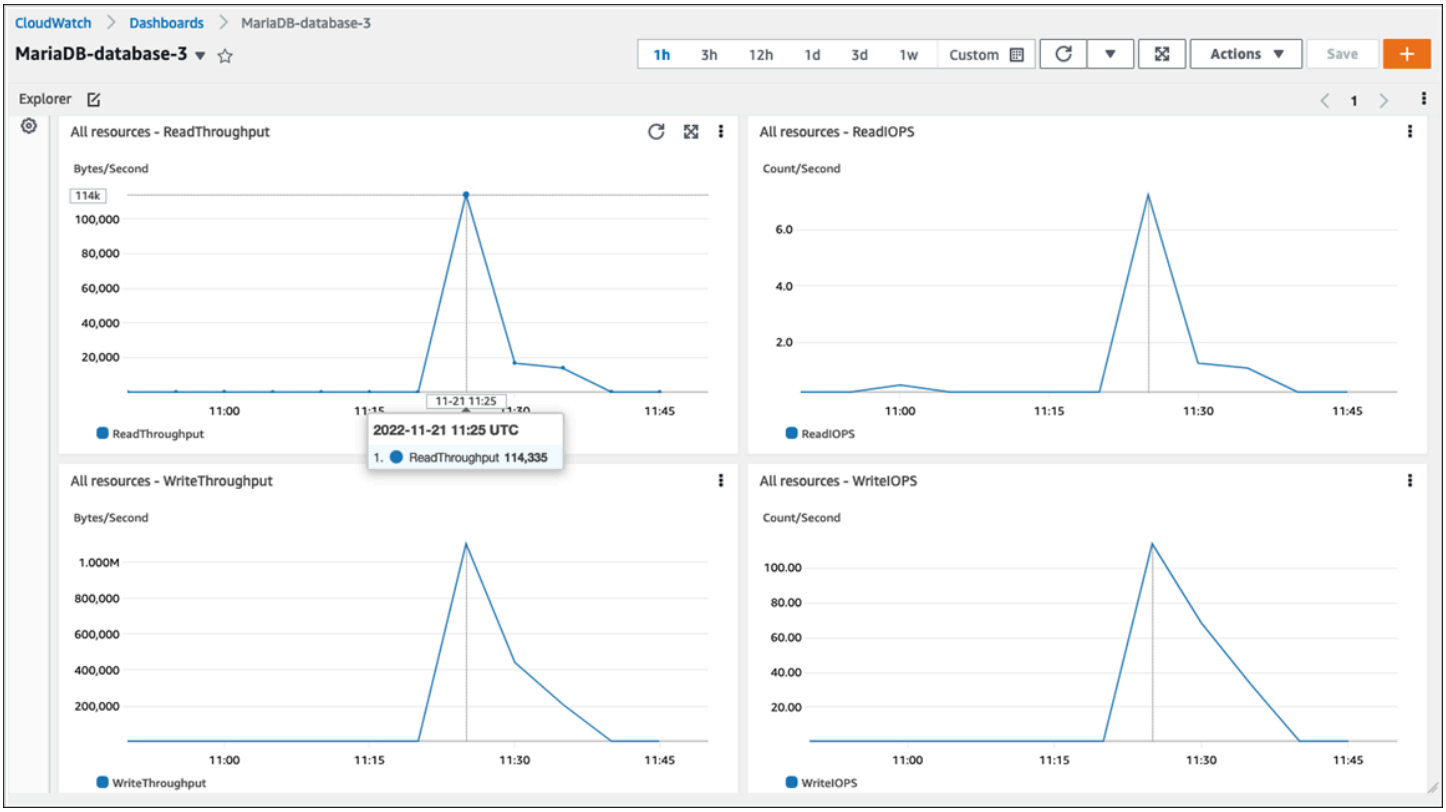
- `CPUUtilization`- Persentase pemanfaatan CPU.
- `BinLogDiskUsage`- Jumlah ruang disk yang ditempati oleh log biner.
- `FreeableMemory`- Jumlah memori akses acak yang tersedia. Ini mewakili nilai`MemAvailable`bidang`/proc/meminfo`.
- `ReadIOPS`- Jumlah rata-rata disk membaca I/O operasi per detik.
- `WriteThroughput`- Jumlah rata-rata byte yang ditulis ke disk per detik untuk penyimpanan lokal.
- `NetworkTransmitThroughput`— Lalu lintas jaringan keluar pada node DB, yang menggabungkan lalu lintas database dan lalu lintas Amazon RDS yang digunakan untuk pemantauan dan replikasi.

Untuk referensi lengkap dari semua metrik yang dipublikasikan oleh Amazon RDS keCloudWatch, lihatAmazonCloudWatchmetrik untuk Amazon RDSdalam dokumentasi Amazon RDS.

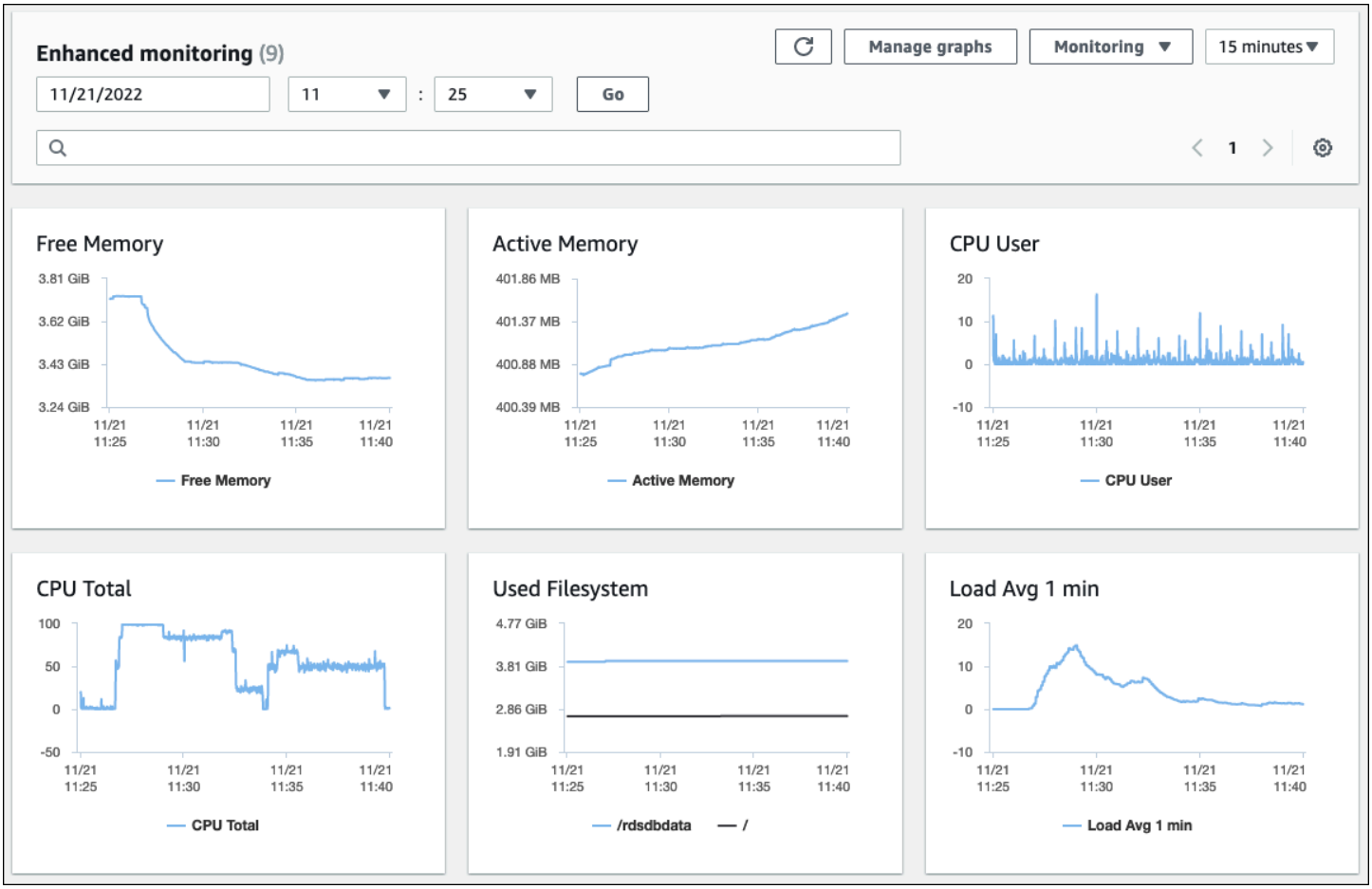
Bagan berikut menunjukkan contohCloudWatchmetrik untuk Amazon RDS yang ditampilkan di konsol Amazon RDS.



Bagan berikut menunjukkan metrik serupa yang ditampilkan diCloudWatchdasbor.



Kumpulan metrik OS lainnya dikumpulkan olehPemantauan yang Ditingkatkanuntuk Amazon RDS. Alat ini memberi Anda visibilitas yang lebih dalam ke dalam kesehatan Amazon RDS untuk MariaDB dan Amazon RDS untuk instans MySQL DB, dengan menyediakan metrik sistem real-time dan informasi proses OS. Ketika Andaaktifkan Pemantauan yang Ditingkatkanpada instans DB Anda dan mengatur perincian yang diinginkan, alat ini mengumpulkan metrik sistem operasi dan informasi proses, yang dapat Anda tampilkan dan analisis diKonsol Amazon RDS, seperti yang ditunjukkan pada layar berikut.



Beberapa metrik utama yang disediakan oleh Enhanced Monitoring adalah:

- `cpuUtilization.total`- Persentase total CPU yang digunakan.
- `cpuUtilization.user`- Persentase CPU yang digunakan oleh program pengguna.
- `memory.active`— Jumlah memori yang ditugaskan, dalam kilobyte.
- `memory.cached`- Jumlah memori yang digunakan untuk caching file berbasis sistem I/O.
- `loadAverageMinute.one`- Jumlah proses yang meminta waktu CPU selama menit terakhir.

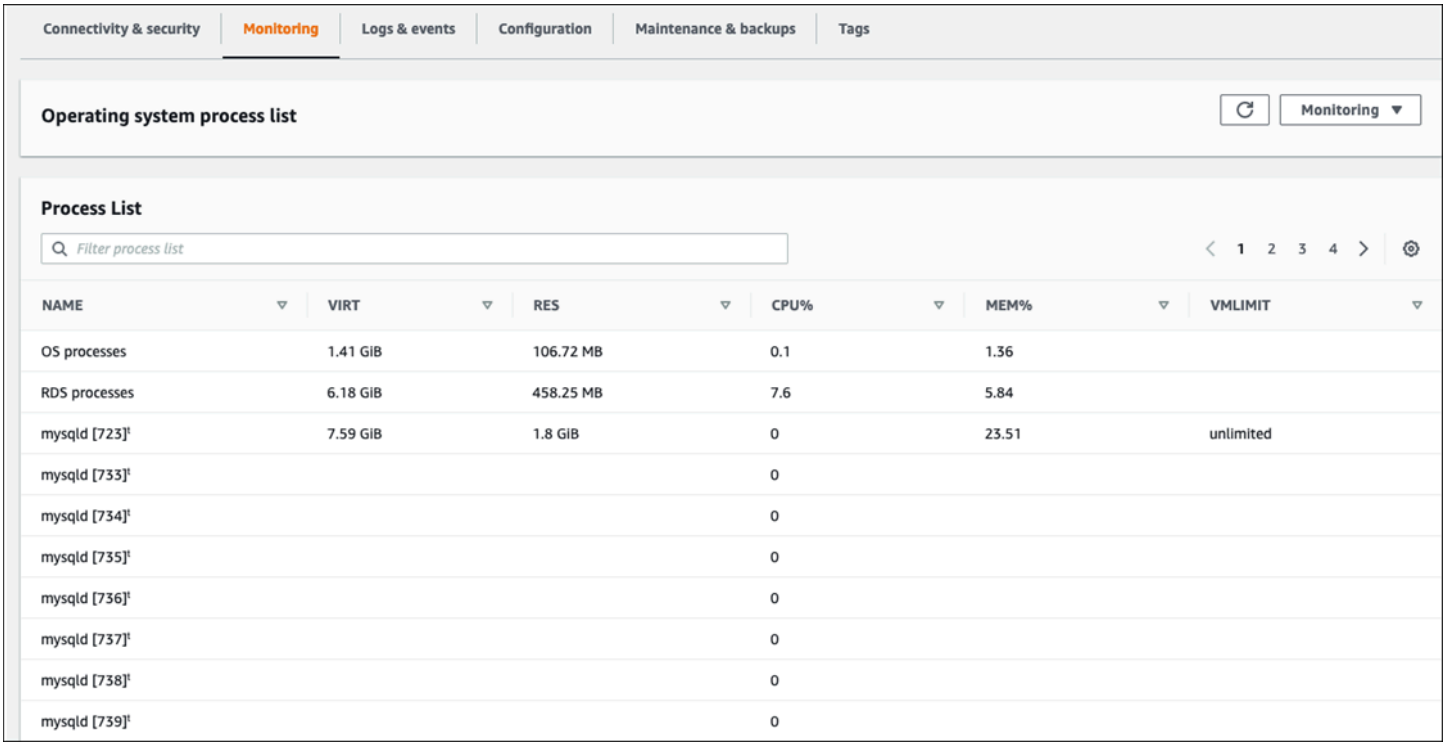
Untuk daftar lengkap metrik, lihatMetrik OS dalam Pemantauan yang Ditingkatkandalam dokumentasi Amazon RDS.

Di konsol Amazon RDS, daftar proses OS memberikan detail untuk setiap proses yang berjalan di instans DB Anda. Daftar ini disusun menjadi tiga bagian:

- Proses OS- Bagian ini merupakan ringkasan gabungan dari semua proses kernel dan sistem. Proses ini umumnya memiliki dampak minimal pada kinerja database.

- Proses RDS- Bagian ini merupakan ringkasan dariAWSproses yang diperlukan untuk mendukung instans DB Amazon RDS. Misalnya, ini mencakup agen manajemen Amazon RDS, proses pemantauan dan diagnostik, dan proses serupa.
- Proses anak RDS— Bagian ini mewakili ringkasan proses Amazon RDS yang mendukung instans DB-dalam hal ini,`mysqld`proses dan benang nya. Yang`mysqld`thread muncul bersarang di bawah induk`mysqld`proses.

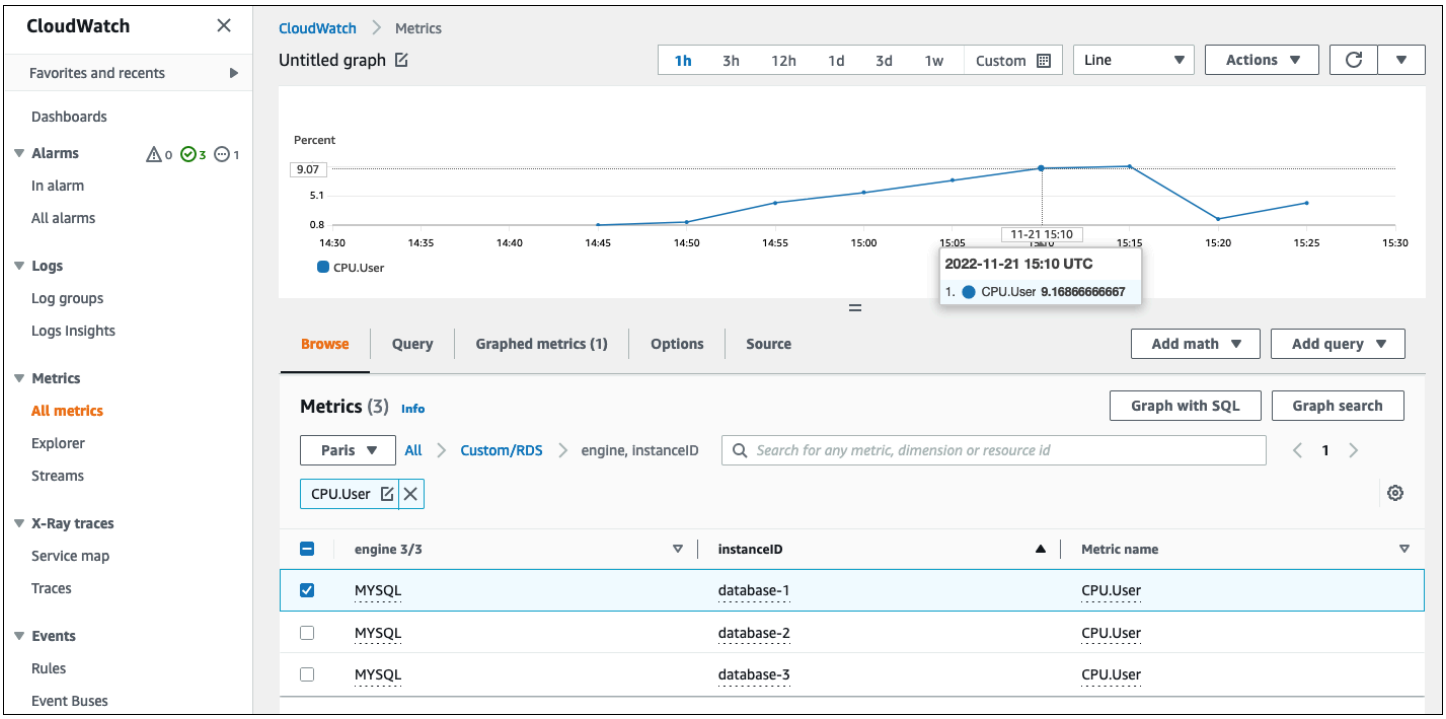
Ilustrasi layar berikut menunjukkan daftar proses OS di konsol Amazon RDS.



Amazon RDS memberikan metrik dari Enhanced Monitoring ke dalamCloudWatchAkun log. Data pemantauan yang ditampilkan di konsol Amazon RDS diambil dariCloudWatchLog. Anda juga bisamengambil metrik untuk instans DB sebagai aliran logdariCloudWatchLog. Metrik ini disimpan dalam format JSON. Anda dapat menggunakan output JSON Enhanced Monitoring dariCloudWatchLog dalam sistem pemantauan pilihan Anda.

Untuk menampilkan grafik padaCloudWatchdasbor dan membuat alarm yang akan memulai tindakan jika metrik melanggar ambang batas yang ditentukan, Anda harus membuat filter metrik diCloudWatchdariCloudWatchLog. Untuk petunjuk terperinci, lihatAWS RE: Posting artikeltentang cara menyaring Enhanced MonitoringCloudWatchLog untuk menghasilkan metrik khusus otomatis untuk Amazon RDS.

Contoh berikut menggambarkan metrik khususCPU.Userdi dalamCustom/RDSnamespace. Metrik khusus ini dibuat dengan memfiltercpuUtilization.userMetrik Pemantauan yang Ditingkatkan dariCloudWatchLog.



Bila metrik tersedia diCloudWatchrepositori, Anda dapat menampilkan dan menganalisisnya diCloudWatchdasbor, terapkan operasi matematika dan kueri lebih lanjut, dan atur alarm untuk memantau metrik spesifik ini dan menghasilkan peringatan jika nilai yang diamati tidak sesuai dengan kondisi alarm yang ditentukan.

Acara, log, dan jejak audit

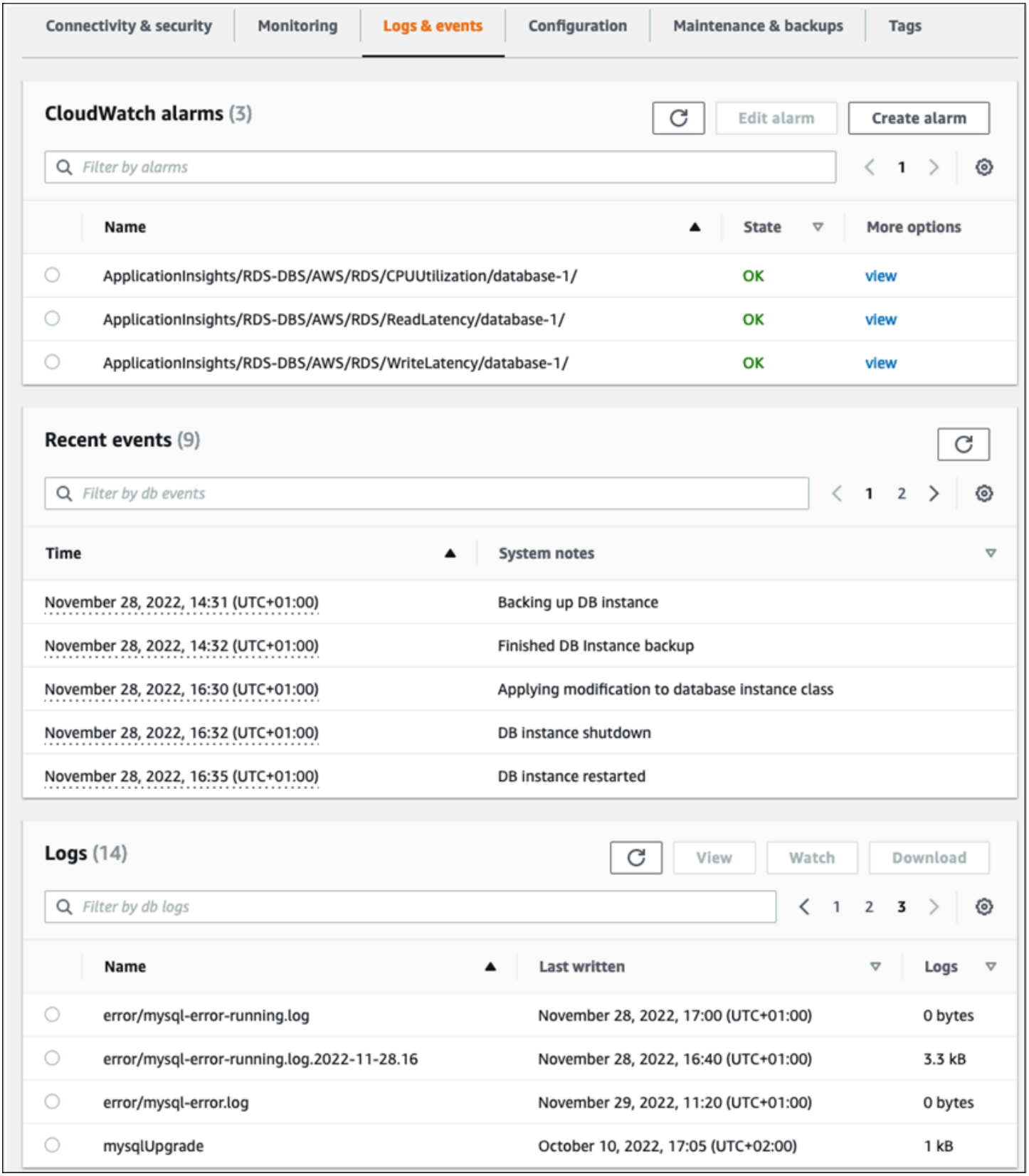
PemantauanMetrik instans DBdanMetrik OS, menganalisis tren dan membandingkan metrik dengan nilai dasar, dan menghasilkan peringatan ketika nilai melanggar ambang batas yang ditentukan semuanya diperlukan dan praktik terbaik yang membantu Anda mencapai dan mempertahankan keandalan, ketersediaan, kinerja, dan keamanan instans DB Amazon RDS Anda. Namun, solusi lengkap juga harus memantau peristiwa database, file log, dan jejak audit database MySQL dan MariaDB.

Bagian

- Acara Amazon RDS
- Log basis data
- Jejak audit

Acara Amazon RDS

SebuahAmazon Acara RDSmenunjukkan perubahan di lingkungan Amazon RDS. Misalnya, ketika status instans DB berubah dariMemulaikepadaTersedia, Amazon RDS menghasilkan acara`RDS-EVENT-0088 The DB instance has been started`. Amazon RDS mengirimkan acara ke AmazonEventBridgedalam waktu dekat real time. Anda dapat mengakses peristiwa melalui konsol Amazon RDS,AWS CLIkomandomenggambarkan-peristiwa, atau operasi API Amazon RDSDescribeEvents. Ilustrasi layar berikut menunjukkan peristiwa dan log yang ditampilkan di konsol Amazon RDS.


Connectivity & security
Monitoring
Logs & events
Configuration
Maintenance & backups
Tags
CloudWatch alarms (3)
Edit alarm
Create alarm
Filter by alarms
1
Name
State
More options
ApplicationInsights/RDS-DBS/AWS/RDS/CPUUtilization/database-1/
OK
view
ApplicationInsights/RDS-DBS/AWS/RDS/ReadLatency/database-1/
OK
view
ApplicationInsights/RDS-DBS/AWS/RDS/WriteLatency/database-1/
OK
view
Recent events (9)
Filter by db events
1
2
Time
System notes
November 28, 2022, 14:31 (UTC+01:00)
Backing up DB instance
November 28, 2022, 14:32 (UTC+01:00)
Finished DB Instance backup
November 28, 2022, 16:30 (UTC+01:00)
Applying modification to database instance class
November 28, 2022, 16:32 (UTC+01:00)
DB instance shutdown
November 28, 2022, 16:35 (UTC+01:00)
DB instance restarted
Logs (14)
View
Watch
Download
Filter by db logs
1
2
3
Name
Last written
Logs
error/mysql-error-running.log
November 28, 2022, 17:00 (UTC+01:00)
0 bytes
error/mysql-error-running.log.2022-11-28.16
November 28, 2022, 16:40 (UTC+01:00)
3.3 kB
error/mysql-error.log
November 29, 2022, 11:20 (UTC+01:00)
0 bytes
mysqlUpgrade
October 10, 2022, 17:05 (UTC+02:00)
1 kB

Amazon RDS memancarkan berbagai jenis peristiwa, termasuk peristiwa instans DB, peristiwa grup parameter DB, peristiwa grup keamanan DB, peristiwa snapshot DB, peristiwa RDS Proxy, dan peristiwa penyebaran biru/hijau. Informasi tersebut meliputi:

- Nama sumber dan jenis sumber; misalnya:`"SourceIdentifier": "database-1", "SourceType": "db-instance"`
- Tanggal dan waktu acara; misalnya:`"Date": "2022-12-01T09:20:28.595000+00:00"`
- Pesan yang terkait dengan acara; misalnya:`"Message": "Finished updating DB parameter group"`
- Kategori acara; misalnya:`"EventCategories": ["configuration change"]`

Untuk referensi lengkap, lihatKategori acara Amazon RDS dan pesan acaradalam dokumentasi Amazon RDS.

Kami menyarankan Anda memantau peristiwa Amazon RDS, karena peristiwa ini menunjukkan perubahan status dalam ketersediaan instans DB, perubahan konfigurasi, perubahan status replika baca, peristiwa cadangan dan pemulihan, tindakan failover, peristiwa kegagalan, modifikasi pada grup keamanan, dan banyak notifikasi lainnya. Misalnya, jika Anda telah menyiapkan instans DB replika baca untuk memberikan kinerja dan daya tahan yang ditingkatkan untuk database Anda, sebaiknya pantau peristiwa Amazon RDS untukbaca replikakategori acara yang terkait dengan contoh DB. Hal ini karena peristiwa sepertiRDS-EVENT-0057 Replication on the read replica was terminatedmenunjukkan bahwa replika baca Anda tidak lagi disinkronkan dengan instans DB utama. Pemberitahuan kepada tim yang bertanggung jawab bahwa peristiwa semacam itu telah terjadi dapat membantu mitigasi masalah secara tepat waktu. AmazonEventBridgedan layanan AWS tambahan, sepertiAWS Lambda, Amazon Simple Queue Service (Amazon SQS), dan Amazon Simple Notification Service (Amazon SNS), dapat membantu Anda mengotomatiskan respons terhadap peristiwa sistem seperti masalah ketersediaan database atau perubahan sumber daya.

Di konsol Amazon RDS, Anda dapat mengambil peristiwa dari 24 jam terakhir. Jika Anda menggunakanAWS CLIatau Amazon RDS API untuk melihat peristiwa, Anda dapat mengambil peristiwa dari 14 hari terakhir dengan menggunakanmenggambarkan-peristiwaperintah sebagai berikut.

```
$ aws rds describe-events --source-identifier database-1 --source-type db-instance
{
    "Events": [
```

```
        {
            "SourceIdentifier": "database-1",
            "SourceType": "db-instance",
            "Message": "CloudWatch Logs Export enabled for logs [audit, error, general,
 slowquery]",
            "EventCategories": [],
            "Date": "2022-12-01T09:20:28.595000+00:00",
            "SourceArn": "arn:aws:rds:eu-west-3:111122223333:db:database-1"
        },
        {
            "SourceIdentifier": "database-1",
            "SourceType": "db-instance",
            "Message": "Finished updating DB parameter group",
            "EventCategories": [
                "configuration change"
            ],
            "Date": "2022-12-01T09:22:40.413000+00:00",
            "SourceArn": "arn:aws:rds:eu-west-3:111122223333:db:database-1"
        }
    ]
}
```

Jika Anda ingin menyimpan acara untuk jangka panjang, baik sampai periode kedaluwarsa yang ditentukan atau secara permanen, Anda dapat menggunakannyaCloudWatchLoguntuk mencatat informasi tentang peristiwa yang dihasilkan oleh Amazon RDS. Untuk mengimplementasikan solusi ini, Anda dapat menggunakan topik Amazon SNS untuk menerima pemberitahuan peristiwa Amazon RDS dan kemudian memanggil fungsi Lambda untuk mencatat peristiwaCloudWatchLog.

1. Buat fungsi Lambda yang akan dipanggil pada acara tersebut dan mencatat informasi dari acara keCloudWatchLog. CloudWatchLog terintegrasi dengan Lambda dan menyediakan cara mudah untuk mencatat informasi acara, dengan menggunakanmencetakfungsi untuk`stdout`.
2. Buat topik SNS dengan berlangganan fungsi Lambda (setProtokolke Lambda), dan aturTitik akhirke Amazon Resource Name (ARN) dari fungsi Lambda yang Anda buat pada langkah sebelumnya.
3. Konfigurasikan topik SNS Anda untuk menerima pemberitahuan peristiwa Amazon RDS. Untuk petunjuk terperinci, lihatAWSRE: posting artikeltentang cara mendapatkan topik Amazon SNS Anda untuk menerima pemberitahuan Amazon RDS.
4. Di konsol Amazon RDS, buat langganan acara baru. SetTargetke ARN, lalu pilih topik SNS yang sebelumnya Anda buat. SetJenis sumberdanKategori acara untuk menyertakansesuai dengan

kebutuhan Anda. Untuk informasi lebih lanjut, lihatBerlangganan pemberitahuan acara Amazon RDSdalam dokumentasi Amazon RDS.

Mencatat Basis Data

Database MySQL dan MariaDB menghasilkan log yang dapat Anda akses untuk audit dan pemecahan masalah. Log tersebut adalah:

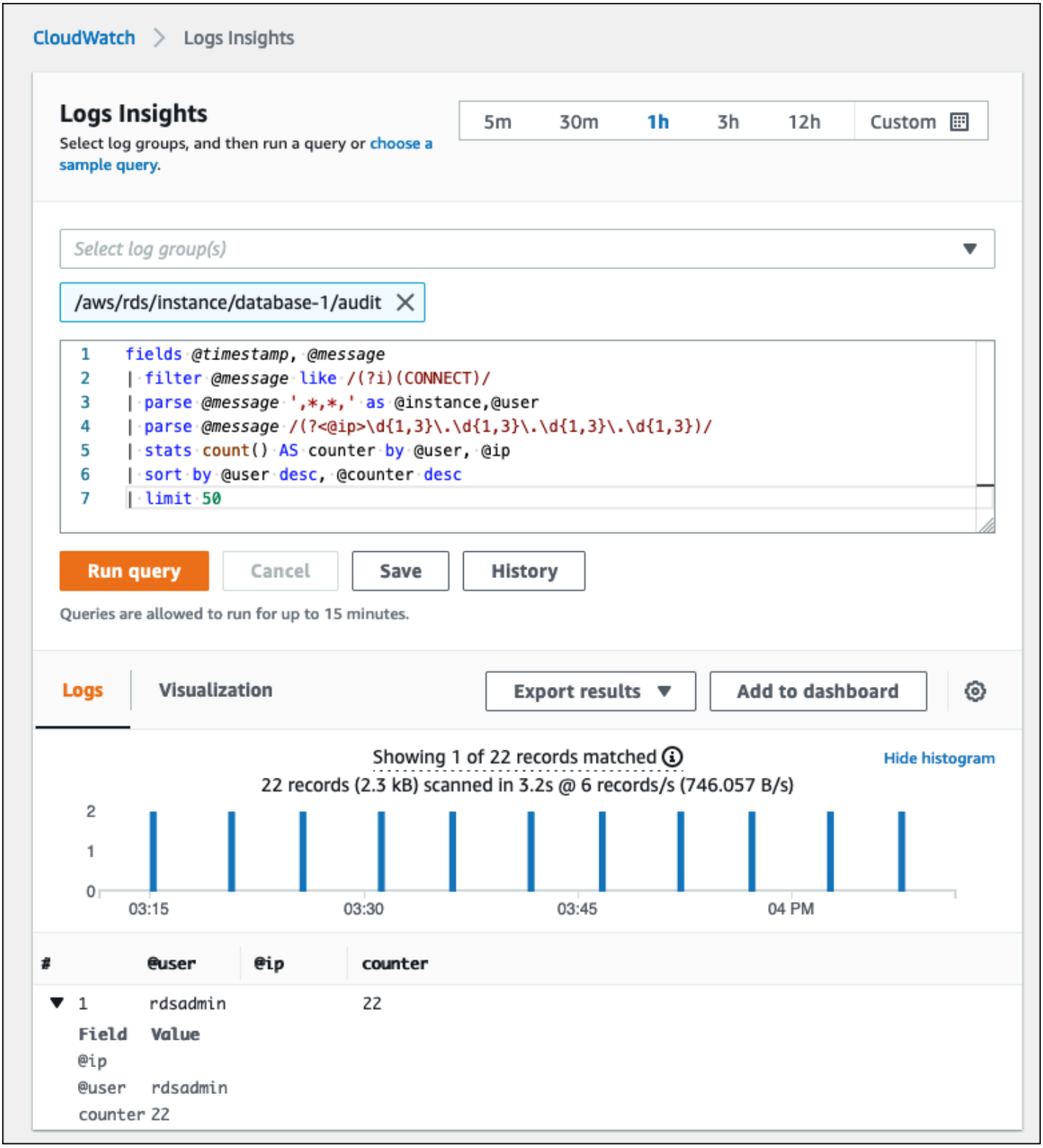
- Audit- Jejak audit adalah seperangkat catatan yang mencatat aktivitas server. Untuk setiap sesi klien, ia mencatat siapa yang terhubung ke server (nama pengguna dan host), yang query dijalankan, tabel mana yang diakses, dan variabel server mana yang diubah.
- Kesalahan- Log ini berisi server (`mysqld`) waktu startup dan shutdown, dan pesan diagnostik seperti kesalahan, peringatan, dan catatan yang terjadi selama startup server dan shutdown, dan saat server sedang berjalan.
- Umum- Log ini mencatat aktivitas`mysqld`, termasuk aktivitas connect dan disconnect untuk setiap klien, dan kueri SQL yang diterima dari klien. Log kueri umum bisa sangat berguna ketika Anda mencurigai adanya kesalahan dan ingin tahu persis apa yang dikirim klien`mysqld`.
- Kueri lambat- Log ini menyediakan catatan query SQL yang membutuhkan waktu lama untuk melakukan.

Sebagai praktik terbaik, Anda harusmempublikasikan log database dari Amazon RDS ke AmazonCloudWatchLog. DenganCloudWatchLog, Anda dapat melakukan analisis real-time dari data log, menyimpan data dalam penyimpanan yang sangat tahan lama, dan mengelola data denganCloudWatchLog agen. Anda bisamengakses dan menonton log database Andadari konsol Amazon RDS. Anda juga dapat menggunakanCloudWatchLog Wawasan untuk secara interaktif mencari dan menganalisis data log Anda diCloudWatchLog. Contoh berikut mengilustrasikan kueri pada log audit yang memeriksa berapa kali`CONNECT`peristiwa muncul di log, yang terhubung, dan klien mana (alamat IP) mereka terhubung dari. Kutipan dari log audit bisa terlihat seperti ini:

```
20221201 14:07:05,ip-10-22-1-51,rdsadmin,localhost,821,0,CONNECT,,,0,SOCKET
20221201 14:07:05,ip-10-22-1-51,rdsadmin,localhost,821,0,DISCONNECT,,,0,SOCKET
20221201 14:12:20,ip-10-22-1-51,rdsadmin,localhost,822,0,CONNECT,,,0,SOCKET
20221201 14:12:20,ip-10-22-1-51,rdsadmin,localhost,822,0,DISCONNECT,,,0,SOCKET
20221201 14:17:35,ip-10-22-1-51,rdsadmin,localhost,823,0,CONNECT,,,0,SOCKET
20221201 14:17:35,ip-10-22-1-51,rdsadmin,localhost,823,0,DISCONNECT,,,0,SOCKET
20221201 14:22:50,ip-10-22-1-51,rdsadmin,localhost,824,0,CONNECT,,,0,SOCKET
```

```
20221201 14:22:50,ip-10-22-1-51,rdsadmin,localhost,824,0,DISCONNECT,,,0,SOCKET
```

Contoh kueri Wawasan Log menunjukkan hal itu`rdsadmin`terhubung ke database dari`localhost`setiap 5 menit, total 22 kali, seperti yang ditunjukkan pada ilustrasi berikut. Hasil ini menunjukkan bahwa aktivitas tersebut berasal dari proses Amazon RDS internal seperti sistem pemantauan itu sendiri.



Peristiwa log sering menyertakan pesan penting yang ingin Anda hitung, seperti peringatan atau kesalahan tentang operasi yang terkait dengan instans MySQL dan MariaDB DB. Misalnya, jika

operasi gagal, kesalahan dapat terjadi dan direkam ke file log kesalahan sebagai berikut:`ERROR 1114 (HY000): The table zip_codes is full`. Anda mungkin ingin memantau entri ini untuk memahami tren kesalahan Anda. Anda bisamembuat kustomCloudWatchmetrik dari log Amazon RDS dengan menggunakan filteruntuk mengaktifkan pemantauan otomatis log database Amazon RDS untuk memantau log tertentu untuk pola tertentu, dan untuk menghasilkan alarm jika ada pelanggaran perilaku yang diharapkan. Sebagai contoh, buat filter metrik untuk grup log`/aws/rds/instance/database-1/error`yang akan memantau log kesalahan dan mencaripola tertentu, sepertiERROR. MengaturPola FilterkepadaERRORdanNilai Metrikkepada1. Filter akan mendeteksi setiap catatan log yang memiliki kata kunciERROR, dan itu akan menambah hitungan sebesar 1 untuk setiap peristiwa log yang berisi “ERROR”. Setelah Anda membuat filter, Anda dapat mengatur alarm untuk memberi tahu Anda jika kesalahan terdeteksi di log kesalahan MySQL atau MariaDB.

Untuk mempelajari lebih lanjut tentang pemantauan log kueri lambat dan log kesalahan dengan membuatCloudWatchdashboard dan menggunakanCloudWatchLog Wawasan, lihat posting blogMembuat AmazonCloudWatchdasbor untuk memantau Amazon RDS dan Amazon Aurora MySQL.

Jejak audit

Jejak audit (atau log audit) menyediakan catatan kronologis peristiwa yang relevan dengan keamanan di akun AWS Anda. Ini mencakup peristiwa untuk Amazon RDS, yang memberikan bukti dokumenter tentang urutan aktivitas yang telah memengaruhi database Anda atau lingkungan cloud Anda. Di Amazon RDS untuk MySQL atau MariaDB, menggunakan jejak audit melibatkan:

- Memantau log audit instans DB
- Memantau panggilan API Amazon RDS diAWS CloudTrail

Untuk instans DB Amazon RDS, tujuan audit biasanya mencakup:

- Mengaktifkan akuntabilitas untuk hal-hal berikut:
  - Modifikasi dilakukan pada parameter atau konfigurasi keamanan
  - Tindakan yang dilakukan dalam skema database, tabel, atau baris, atau tindakan yang memengaruhi konten tertentu
- Deteksi dan investigasi intrusi
- Deteksi dan investigasi aktivitas mencurigakan

- Deteksi masalah otorisasi; misalnya, untuk mengidentifikasi pelanggaran hak akses oleh pengguna reguler atau hak istimewa

Jejak audit database mencoba menjawab pertanyaan-pertanyaan khas ini:Siapa yang melihat atau memodifikasi data sensitif di dalam database Anda? Kapan ini terjadi? Dari mana pengguna tertentu mengakses data? Apakah pengguna istimewa menyalahgunakan hak akses tak terbatas mereka?

Baik MySQL dan MariaDB menerapkan fitur jejak audit instans DB dengan menggunakan Plugin Audit MariaDB. Plugin ini mencatat aktivitas database seperti pengguna yang masuk ke database dan kueri yang berjalan melawan database. Catatan aktivitas basis data disimpan dalam berkas log. Untuk mengakses log audit, DB instance harus menggunakan grup opsi kustom dengan opsi `MARIADB_AUDIT_PLUGIN` . Untuk informasi lebih lanjut, lihatDukungan Plugin Audit MariaDB untuk MySQLdalam dokumentasi Amazon RDS. Catatan dalam log audit disimpan dalam format tertentu, seperti yang didefinisikan oleh plugin. Anda dapat menemukan rincian lebih lanjut tentang format log audit diDokumentasi Server MariaDB.

YangAWS Cloudjejak audit untukAWSakun disediakan olehAWS CloudTraillayanan. CloudTrail menangkap panggilan API untuk Amazon RDS sebagai acara. Semua tindakan Amazon RDS dicatat. CloudTrailmenyediakan catatan tindakan di Amazon RDS yang dilakukan oleh pengguna, peran, atau lainnyaAWSlayanan. Acara termasuk tindakan yang diambil diAWSKonsol Manajemen,AWS CLI, danAWSSDK dan API.

Contoh

Dalam skenario audit umum, Anda mungkin perlu menggabungkannyaAWS CloudTrailjejak dengan log audit database dan pemantauan peristiwa Amazon RDS. Misalnya, Anda mungkin memiliki skenario di mana parameter database instans DB Amazon RDS Anda (misalnya,`database-1`) telah dimodifikasi dan tugas Anda adalah mengidentifikasi siapa yang melakukan modifikasi, apa yang diubah, dan kapan perubahan terjadi.

Untuk menyelesaikan tugas, ikuti langkah-langkah ini:

1. Buat daftar peristiwa Amazon RDS yang terjadi pada instance database`database-1`dan menentukan apakah ada acara dalam kategori`configuration change`yang memiliki pesan`Finished updating DB parameter group`.

```
$ aws rds describe-events --source-identifier database-1 --source-type db-instance
{
    "Events": [
```

```
        {
            "SourceIdentifier": "database-1",
            "SourceType": "db-instance",
            "Message": "Finished updating DB parameter group",
            "EventCategories": [
                "configuration change"
            ],
            "Date": "2022-12-01T09:22:40.413000+00:00",
            "SourceArn": "arn:aws:rds:eu-west-3:111122223333:db:database-1"
        }
    ]
}
```

2. Identifikasi grup parameter DB mana yang digunakan instance DB:

```
$ aws rds describe-db-instances --db-instance-identifier database-1 --query
 'DBInstances[*].[DBInstanceIdentifier,Engine,DBParameterGroups]'
[
    [
        "database-1",
        "mariadb",
        [
            {
                "DBParameterGroupName": "mariadb10-6-test",
                "ParameterApplyStatus": "pending-reboot"
            }
        ]
    ]
]
```

3. GunakanAWS CLIuntuk mencariCloudTrailacaradi Wilayah dimanadatabase-1diterapkan, dalam periode waktu sekitar acara Amazon RDS yang ditemukan pada langkah 1, dan di manaEventName=ModifyDBParameterGroup.

```
$ aws cloudtrail --region eu-west-3 lookup-events --lookup-attributes
 AttributeKey=EventName,AttributeValue=ModifyDBParameterGroup --start-time
 "2022-12-01, 09:00 AM" --end-time "2022-12-01, 09:30 AM"

{
    "eventVersion": "1.08",
    "userIdentity": {
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
```

```
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/Role1",
                "accountId": "111122223333",
                "userName": "User1"
            }
        }
    },
    "eventTime": "2022-12-01T09:18:19Z",
    "eventSource": "rds.amazonaws.com",
    "eventName": "ModifyDBParameterGroup",
    "awsRegion": "eu-west-3",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "parameters": [
            {
                "isModifiable": false,
                "applyMethod": "pending-reboot",
                "parameterName": "innodb_log_buffer_size",
                "parameterValue": "8388612"
            },
            {
                "isModifiable": false,
                "applyMethod": "pending-reboot",
                "parameterName": "innodb_write_io_threads",
                "parameterValue": "8"
            }
        ],
        "dBParameterGroupName": "mariadb10-6-test"
    },
    "responseElements": {
        "dBParameterGroupName": "mariadb10-6-test"
    },
    "requestID": "fdf19353-de72-4d3d-bf29-751f375b6378",
    "eventID": "0bba7484-0e46-4e71-93a8-bd01ca8386fe",
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management",
    "sessionCredentialFromConsole": "true"
```

```
}
```

YangCloudTrailacara mengungkapkan bahwa`User1`dengan peran`Role1`dariAWSakun 111122223333 memodifikasi grup parameter DB`mariadb10-6-test`, yang digunakan oleh instans DB`database-1`di atas`2022-12-01 at 09:18:19 h`. Dua parameter yang dimodifikasi dan diatur ke nilai-nilai berikut:

- `innodb_log_buffer_size = 8388612`
- `innodb_write_io_threads = 8`

TambahanCloudTraildanCloudWatchFitur log

Anda dapat memecahkan masalah insiden operasional dan keamanan selama 90 hari terakhir dengan melihatRiwayat acarapadaCloudTrailkonsol. Untuk memperpanjang periode retensi dan memanfaatkan kemampuan kueri tambahan, Anda dapat menggunakanAWS CloudTrailDanau. DenganAWS CloudTrailSelain itu, Anda dapat menyimpan data acara di toko data acara hingga tujuh tahun. Selain itu, layanan ini mendukung kueri SQL kompleks yang menawarkan tampilan peristiwa yang lebih dalam dan lebih dapat disesuaikan daripada tampilan yang disediakan oleh pencarian kunci-nilai sederhana diRiwayat acara.

Untuk memantau jejak audit, menyetel alarm, dan mendapatkan notifikasi saat aktivitas tertentu terjadi, Anda perlumengkonfigurasiCloudTrailuntuk mengirim catatan jejaknya keCloudWatchLog. Setelah catatan jejak disimpan sebagaiCloudWatchLog, Anda dapat menentukan filter metrik untuk mengevaluasi peristiwa log agar sesuai dengan istilah, frasa, atau nilai, dan menetapkan metrik ke filter metrik. Selanjutnya, Anda dapat membuatCloudWatchalarm yang dihasilkan sesuai dengan ambang batas dan periode waktu yang Anda tentukan. Misalnya, Anda dapat mengonfigurasi alarm yang mengirim pemberitahuan ke tim yang bertanggung jawab, sehingga mereka dapat mengambil tindakan yang sesuai. Anda juga dapat mengkonfigurasiCloudWatchuntuk secara otomatis melakukan tindakan dalam menanggapi alarm.

Peringatan

Peringatan adalah salah satu sumber informasi terpenting dalam hal keamanan, ketersediaan, kinerja, dan keandalan infrastruktur TI dan layanan TI Anda. Mereka memberi tahu dan memberi tahu tim TI Anda tentang ancaman keamanan yang sedang berlangsung, pemadaman, masalah kinerja, atau kegagalan sistem.

Perpustakaan Infrastruktur Teknologi Informasi (ITIL), khususnya praktik manajemen layanan TI (ITSM), menetapkan peringatan otomatis pada titik fokus pemantauan dan manajemen acara dan praktik terbaik manajemen insiden.

Peringatan insiden adalah saat alat pemantauan menghasilkan peringatan untuk memberi tahu tim Anda dan alat otomatis (untuk item yang dapat ditindaklanjuti secara otomatis) tentang perubahan, tindakan berisiko tinggi, atau kegagalan di lingkungan TI. Peringatan TI adalah garis pertahanan pertama terhadap pemadaman sistem atau perubahan yang dapat berubah menjadi insiden besar. Dengan memantau sistem secara otomatis dan menghasilkan peringatan untuk pemadaman dan perubahan berisiko, tim TI dapat meminimalkan waktu henti dan mengurangi biaya tinggi yang menyertainya.

Sebagai praktik terbaik,AWSWell-Architected Framework mengatur bahwa Andamenggunakan pemantauan untuk menghasilkan notifikasi berbasis alarm, danmemantau dan alarm secara proaktif. GunakanCloudWatchatau layanan pemantauan pihak ketiga untuk menyetel alarm yang menunjukkan kapan metrik berada di luar batas yang diharapkan.

Tujuan manajemen peringatan adalah untuk menetapkan prosedur yang efisien dan terstandarisasi untuk menangani kejadian dan insiden terkait TI melalui penebangan, klasifikasi, definisi dan implementasi tindakan, penutupan, dan kegiatan tinjauan pasca insiden.

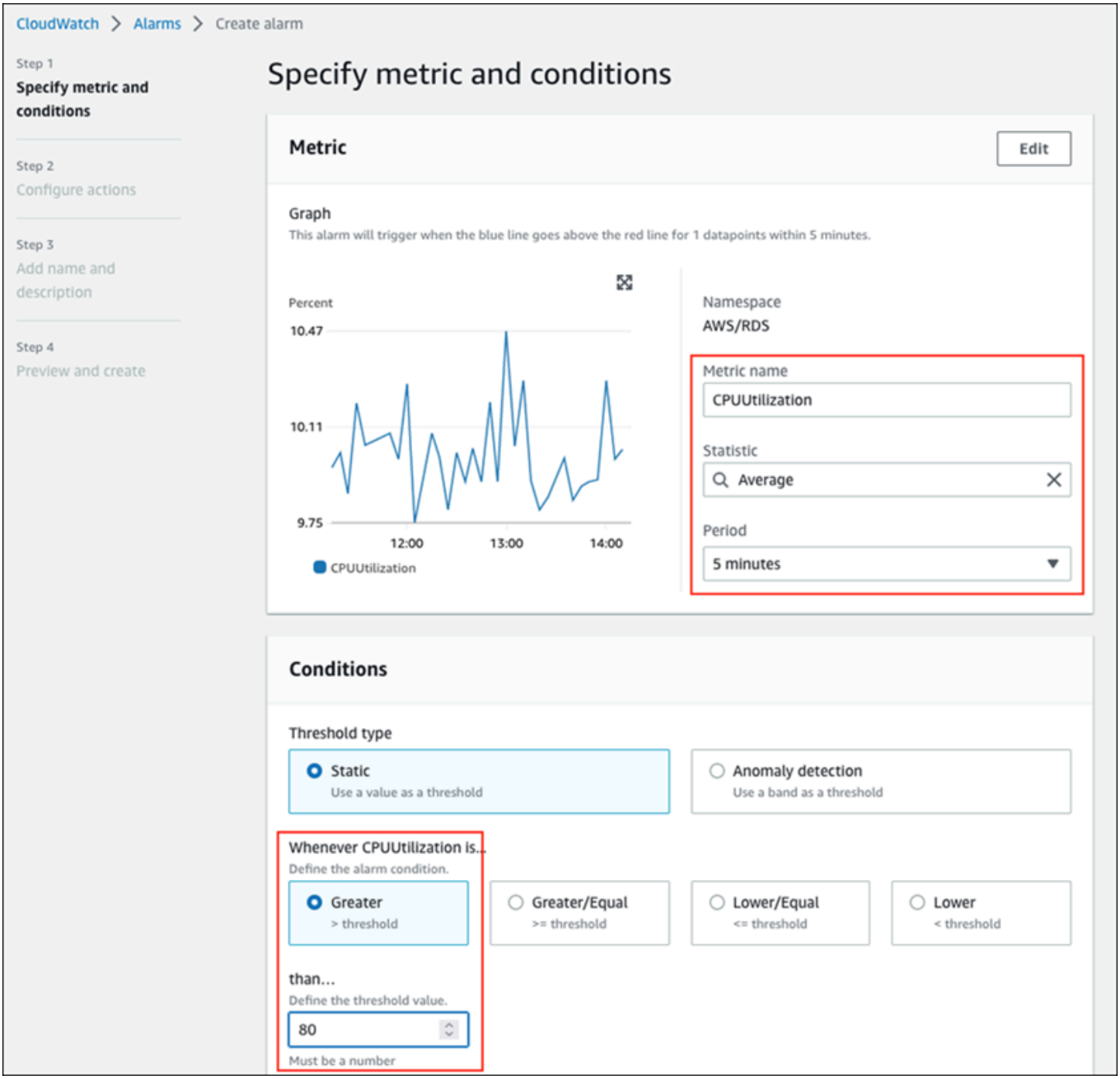
Bagian

- CloudWatchalarm
- EventBridgeaturan
- Menentukan tindakan, mengaktifkan, dan menonaktifkan alarm

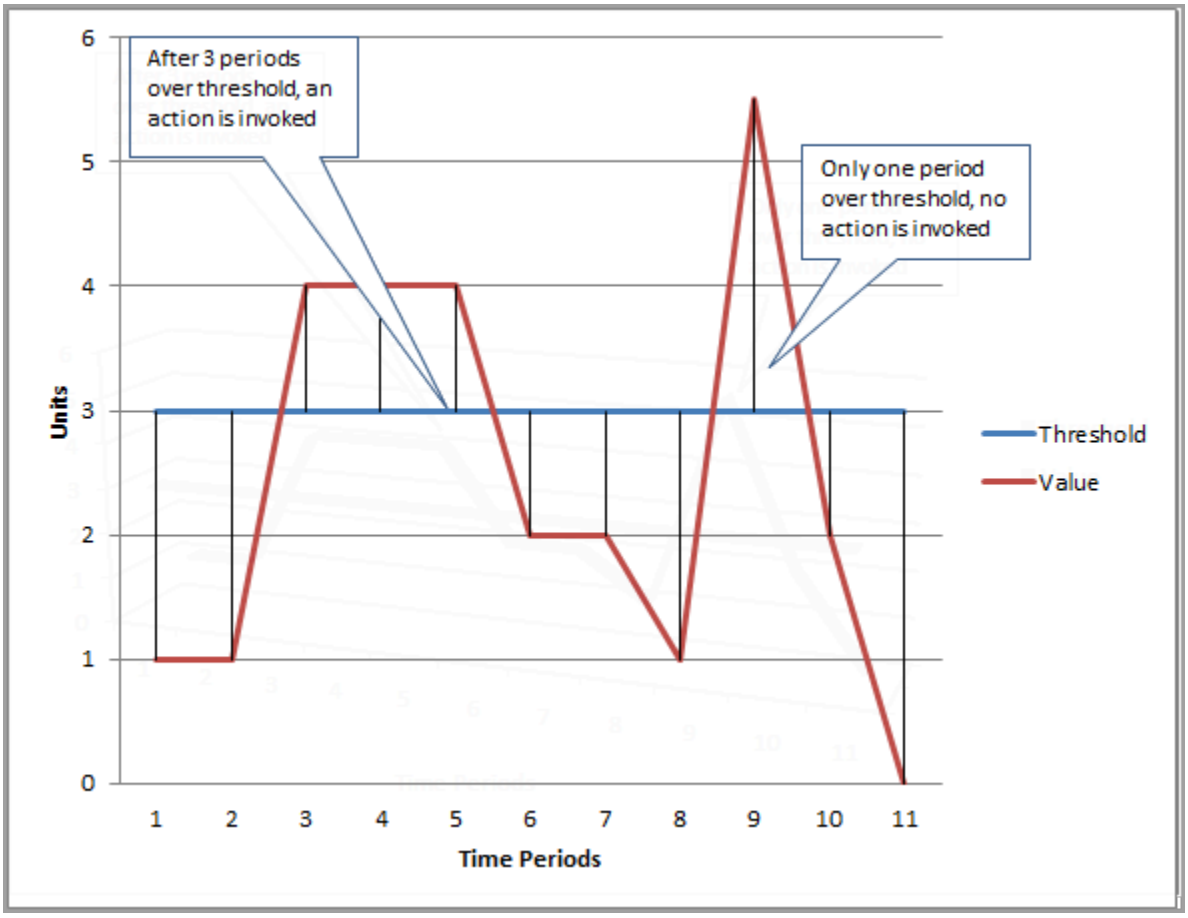
Alarm CloudWatch

Saat Anda mengoperasikan instans DB Amazon RDS, Anda ingin memantau dan menghasilkan peringatan pada berbagai jenis metrik, peristiwa, dan pelacakan. Untuk database MySQL dan MariaDB, sumber informasi penting adalahMetrik instans DB,Metrik OS,peristiwa, log, dan jejak audit. Kami menyarankan Anda menggunakanCloudWatchalarmuntuk menonton metrik tunggal selama periode waktu yang Anda tentukan.

Contoh berikut menggambarkan bagaimana Anda dapat mengatur alarm yang menontonCPUUtilizationmetrik (persentase pemanfaatan CPU) pada semua instans DB Amazon RDS Anda. Anda mengkonfigurasi alarm yang akan dipicu jika pemanfaatan CPU pada instans DB mana pun lebih besar dari 80 persen untuk periode evaluasi 5 menit.



Ini berarti bahwa alarm masuk keALARMmenyatakan jika ada database Anda mengalami pemanfaatan CPU yang tinggi (lebih dari 80 persen) selama 5 menit atau lebih. Alarm tetap ada di0Kmenyatakan jika CPU kadang-kadang meledak ke pemanfaatan lebih dari 80 persen untuk waktu yang singkat, dan kemudian turun lagi di bawah ambang batas. Grafik berikut menggambarkan logika ini.



CloudWatchalarm mendukung alarm metrik dan komposit.

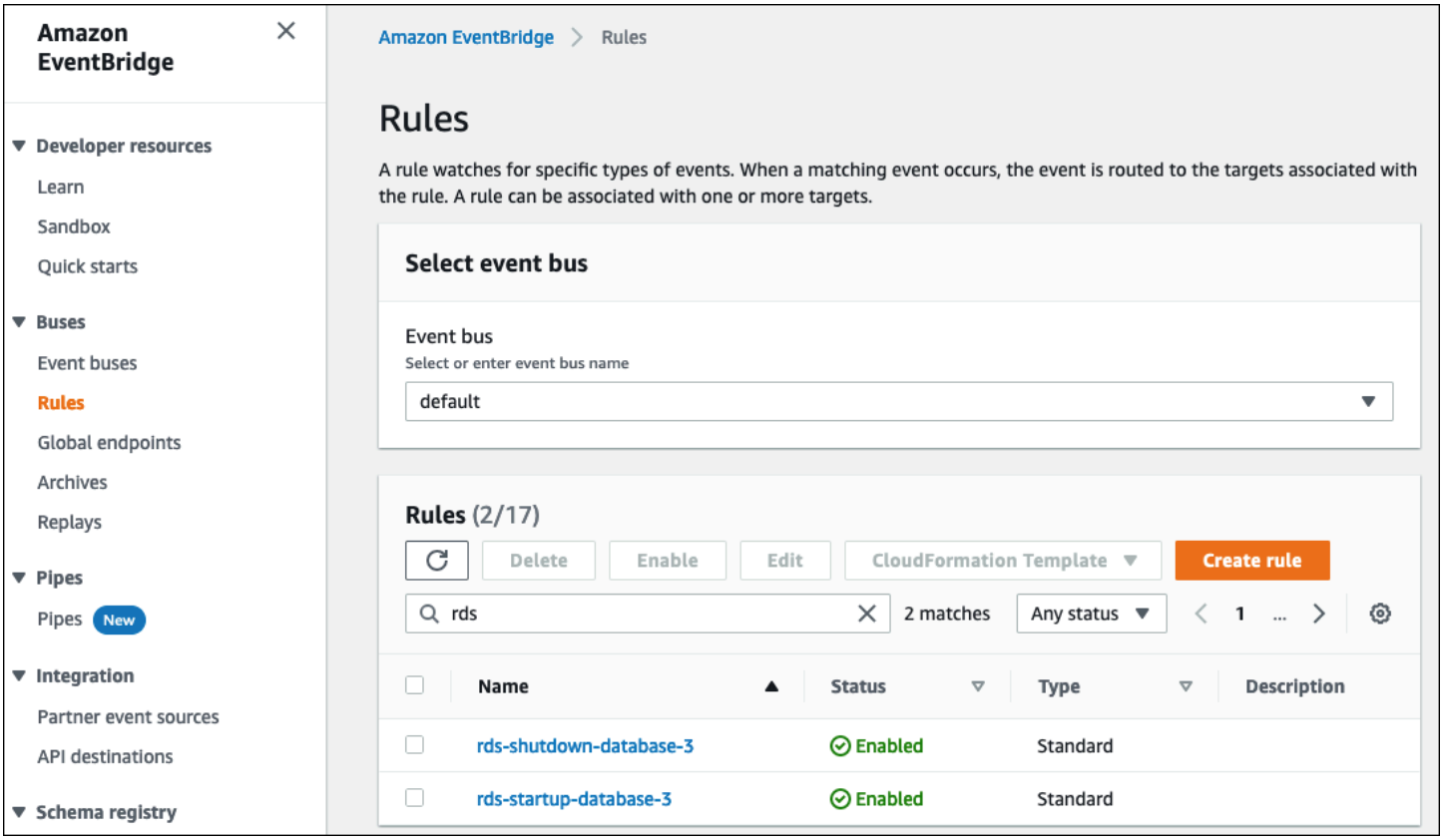
- SEBUAHalarm metrikjam tangan tunggalCloudWatchmetrik dan dapat melakukan ekspresi matematika pada metrik. Alarm metrik dapat mengirim pesan Amazon SNS, yang, pada gilirannya, dapat mengambil satu atau beberapa tindakan berdasarkan nilai metrik relatif terhadap ambang batas tertentu selama beberapa periode waktu.
- SEBUAHalarm kompositdidasarkan pada ekspresi aturan, yang mengevaluasi keadaan beberapa alarm dan masuk keALARMnegara hanya jika semua kondisi aturan terpenuhi. Alarm komposit biasanya digunakan untuk mengurangi jumlah peringatan yang tidak perlu. Misalnya, Anda mungkin memiliki alarm gabungan yang berisi beberapa alarm metrik yang dikonfigurasi untuk tidak mengambil tindakan. Alarm komposit akan mengirim peringatan ketika semua alarm metrik individu dalam komposit sudah ada di`ALARM`

CloudWatchalarm hanya bisa menontonCloudWatchmetrik. Jika Anda ingin membuat alarm berdasarkan kesalahan, kueri lambat, atau log umum, Anda harus membuatCloudWatchmetrik dari log. Anda dapat mencapai itu seperti yang dibahas sebelumnya diPemantauan OSdanAcara,

log, dan jejak auditbagian, dengan menggunakan filter untukmembuat metrik dari peristiwa log. Demikian pula, untuk memperingatkan metrik Enhanced Monitoring, Anda harus membuat filter metrikCloudWatchdariCloudWatchLog.

EventBridgeaturan

Acara Amazon RDSdikirim ke AmazonEventBridge, dan Anda dapat menggunakanEventBridgeaturanuntuk bereaksi terhadap peristiwa-peristiwa itu. Misalnya, Anda dapat membuatEventBridgeaturan yang akan memberi tahu Anda dan mengambil tindakan jika satu instans DB tertentu berhenti atau dimulai, seperti yang ditunjukkan layar berikut.

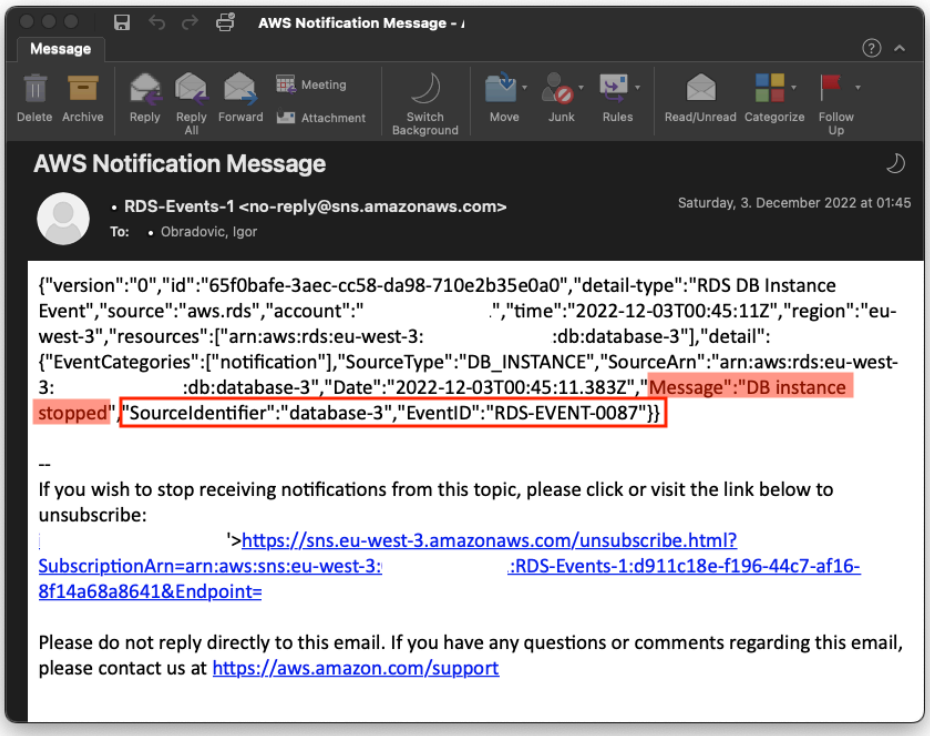


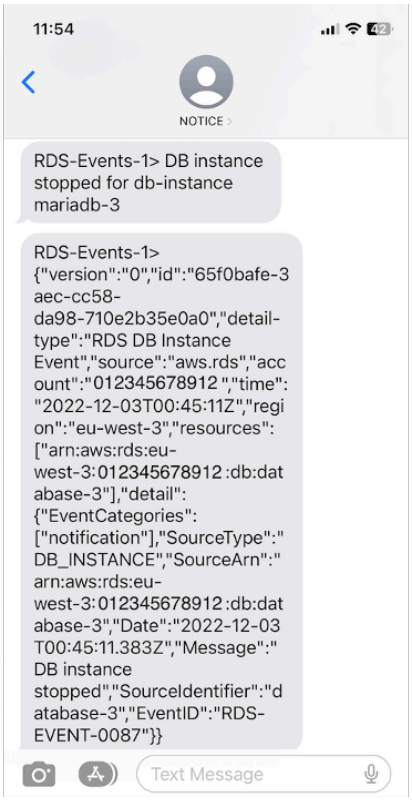
Aturan yang mendeteksiThe `DB instance has been stopped`acara memiliki ID peristiwa Amazon RDS`RDS-EVENT-0087`, sehingga Anda mengatur`Event Pattern`milik aturan untuk:

```
{
  "source": ["aws.rds"],
  "detail-type": ["RDS DB Instance Event"],
  "detail": {
    "SourceArn": ["arn:aws:rds:eu-west-3:111122223333:db:database-3"],
    "EventID": ["RDS-EVENT-0087"]
```

```
  }
}
```

Aturan ini memonitor instans DB`database-3`saja, dan jam tangan untuk`RDS-EVENT-0087`acara. KapanEventBridgemendeteksi acara, ia mengirimkan acara ke sumber daya atau titik akhir, yang dikenal sebagaitarget. Di sinilah Anda dapat menentukan tindakan yang ingin Anda ambil jika instans Amazon RDS dimatikan. Anda dapat mengirim acara ke banyak kemungkinan target, termasuk topik SNS, antrean Amazon Simple Queue Service (Amazon SQS), danAWS Lambdafungsi,AWS Systems ManagerOtomasi, sebuahAWS Batchpekerjaan, Amazon API Gateway, rencana respons di Incident Manager, kemampuanAWS Systems Manager, dan banyak lainnya. Misalnya, Anda dapat membuat topik SNS yang akan mengirim email pemberitahuan dan SMS, dan menetapkan topik SNS sebagai targetEventBridgeaturan. Jika instans DB Amazon RDS`database-3`telah dihentikan, Amazon RDS memberikan acara`RDS-EVENT-0087`kepadaEventBridge, di mana ia akan terdeteksi. EventBridgekemudian memanggil target, yang merupakan topik SNS. Topik SNS dikonfigurasi untuk mengirim email (seperti yang ditunjukkan pada ilustrasi berikut) dan SMS.





Menentukan tindakan, mengaktifkan, dan menonaktifkan alarm

Anda dapat menggunakanCloudWatchalarm untuk menentukan tindakan apa yang harus diambil alarm saat berubah di antara`OK`,`ALARM`, dan`INSUFFICIENT_DATA`menyatakan. CloudWatchmemiliki

integrasi bawaan dengan topik SNS dan beberapa kategori tindakan tambahan yang tidak berlaku untuk metrik Amazon RDS, seperti tindakan Amazon Elastic Compute Cloud (Amazon EC2) atau tindakan grup Amazon EC2 Auto Scaling. EventBridgeumumnya digunakan untuk menulis aturan dan menentukan target yang mengambil tindakan saat alarm dipicu untuk metrik Amazon RDS. CloudWatchmengirimkan acara keEventBridgesetiap kaliCloudWatchalarm mengubah keadaannya. Anda dapat menggunakan peristiwa perubahan status alarm ini untuk memicu target peristiwa diEventBridge. Untuk informasi lebih lanjut, lihatAcara alarm danEventBridgedi dalamCloudWatchdokumentasi.

Anda mungkin juga perlu mengelola alarm; misalnya, untuk menonaktifkan alarm secara otomatis selama perubahan atau pengujian konfigurasi yang direncanakan, lalu mengaktifkan kembali alarm saat tindakan yang direncanakan selesai. Misalnya, jika Anda memiliki pembaruan perangkat lunak database terjadwal yang direncanakan yang memerlukan waktu henti, dan Anda memiliki alarm yang akan diaktifkan jika database tidak tersedia, Anda dapat menonaktifkan dan mengaktifkan alarm dengan menggunakan tindakan APIDisableAlarmActionsdanEnableAlarmActions, ataudisable-alarm-actionsdanenable-alarm-actionsperintah diAWS CLI. Anda juga dapat melihat riwayat alarm diCloudWatchkonsol atau dengan menggunakanDescribeAlarmHistoryAksi API ataudescribe-alarm-historyperintah diAWS CLI. CloudWatch menjaga riwayat alarm selama dua minggu. PadaCloudWatchkonsol, Anda dapat memilihFavorit dan terbarumenu di panel navigasi untuk mengatur dan mengakses alarm favorit dan yang paling baru dikunjungi.

Langkah dan sumber daya selanjutnya

Untuk informasi lebih lanjut tentang migrasi database relasional Anda keAWS Cloud, lihat strategi berikut padaAWSSitus web Prescriptive Guidance:

- Strategi migrasi untuk database relasional

Anda dapat menjelajahipola migrasi databaseuntukstep-by-steppetunjuk mengenai database relasional spesifik Anda yang berjalan diAWS Cloud, termasuk tugas yang terkait dengan pemantauan, migrasi, dan manajemen data.

Gunakan filter pada halaman itu untuk menemukan pola denganAWSlayanan (misalnya, migrasi ke Amazon RDS atau Amazon Aurora), dengan beban kerja (misalnya, open-source, yang mencakup database MySQL dan MariaDB), atau dengan penggunaan yang direncanakan (produksi atau pilot).

Untuk sumber daya tambahan, lihat yang berikut ini:

- Panduan Pengguna Layanan Database Relasional Amazon
- AmazonCloudWatchPanduan Pengguna
- FAQ Amazon RDS
- FAQ Wawasan Kinerja
- Memberikan metrik penghitung Amazon RDS Performance Insights ke penyedia layanan Pemantauan Kinerja Aplikasi pihak ketiga menggunakan AmazonCloudWatchAliran Metrik(AWSposting blog)
- Membuat AmazonCloudWatchdasbor untuk memantau Amazon RDS dan Amazon Aurora MySQL(AWSposting blog)
- Menyetel Amazon RDS untuk MySQL dengan Wawasan Kinerja(AWSposting blog)

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
Informasi yang diperbarui	Memperbarui informasi tentang eksportir dan menambahkan pedoman untuk memilih eksportir.	Juni 13, 2024
Publikasi awal	—	Juni 30, 2023

AWS Glosarium Panduan Preskriptif

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

Nomor

7 Rs

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

- Refactor/Re-Architect — Memindahkan aplikasi dan memodifikasi arsitekturnya dengan memanfaatkan sepenuhnya fitur cloud-native untuk meningkatkan kelincahan, kinerja, dan skalabilitas. Ini biasanya melibatkan porting sistem operasi dan database. Contoh: Migrasikan database Oracle lokal Anda ke Amazon Aurora PostgreSQL Compatible Edition.
- Replatform (angkat dan bentuk ulang) — Pindahkan aplikasi ke cloud, dan perkenalkan beberapa tingkat pengoptimalan untuk memanfaatkan kemampuan cloud. Contoh: Migrasikan database Oracle lokal Anda ke Amazon Relational Database Service (Amazon RDS) untuk Oracle di Cloud. AWS
- Pembelian kembali (drop and shop) - Beralih ke produk yang berbeda, biasanya dengan beralih dari lisensi tradisional ke model SaaS. Contoh: Migrasikan sistem manajemen hubungan pelanggan (CRM) Anda ke Salesforce.com.
- Rehost (lift dan shift) — Pindahkan aplikasi ke cloud tanpa membuat perubahan apa pun untuk memanfaatkan kemampuan cloud. Contoh: Migrasikan database Oracle lokal Anda ke Oracle pada instans EC2 di Cloud. AWS
- Relokasi (hypervisor-level lift and shift) — Pindahkan infrastruktur ke cloud tanpa membeli perangkat keras baru, menulis ulang aplikasi, atau memodifikasi operasi yang ada. Skenario migrasi ini khusus untuk VMware Cloud on AWS, yang mendukung kompatibilitas mesin virtual (VM) dan portabilitas beban kerja antara lingkungan lokal Anda dan. AWS Anda dapat menggunakan teknologi VMware Cloud Foundation dari pusat data lokal saat memigrasikan infrastruktur ke VMware Cloud. AWS Contoh: Pindahkan hypervisor yang menghosting database Oracle Anda ke VMware Cloud on. 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 basis data 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

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.

Cloud Center of Excellence (CCoE)

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

komputasi cloud

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

model operasi cloud

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

tahap adopsi cloud

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

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

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

CMDB

Lihat database manajemen konfigurasi.

repositori kode

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

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

cache dingin

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

data dingin

Data yang jarang diakses dan biasanya historis. Saat menanyakan jenis data ini, kueri lambat biasanya dapat diterima. Memindahkan data ini ke tingkat 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, AWS Panorama menawarkan perangkat yang menambahkan CV ke jaringan kamera lokal, dan Amazon SageMaker menyediakan algoritme pemrosesan gambar untuk CV.

konfigurasi drift

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

database manajemen konfigurasi (CMDB)

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

paket kesesuaian

Kumpulan AWS Config aturan dan tindakan remediasi yang dapat Anda kumpulkan untuk menyesuaikan kepatuhan dan pemeriksaan keamanan Anda. Anda dapat menerapkan paket kesesuaian sebagai entitas tunggal di Akun AWS dan 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 umumnya digambarkan sebagai pipa. CI/CD dapat membantu

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

CV

Lihat visi komputer.

D

data saat istirahat

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

klasifikasi data

Proses untuk mengidentifikasi dan mengkategorikan data dalam jaringan Anda berdasarkan 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 AWS Cloud dapat mengurangi risiko privasi, biaya, dan jejak karbon analitik Anda.

perimeter data

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

prapemrosesan data

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

asal data

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

subjek data

Individu yang datanya dikumpulkan dan diproses.

gudang data

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

bahasa definisi database (DDL)

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

bahasa manipulasi basis data (DHTML)

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

DDL

Lihat bahasa definisi database.

ansambel yang dalam

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

pembelajaran mendalam

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

defense-in-depth

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

administrator yang didelegasikan

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

deployment

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

lingkungan pengembangan

Lihat lingkungan.

kontrol detektif

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

pemetaan aliran nilai pengembangan (DVSM)

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

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

kembar digital

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

tabel dimensi

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

musibah

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

pemulihan bencana (DR)

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

DML~

Lihat bahasa manipulasi database.

desain berbasis domain

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

DR

Lihat pemulihan bencana.

deteksi drift

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

DVSM

Lihat pemetaan aliran nilai pengembangan.

E

EDA

Lihat analisis data eksplorasi.

komputasi tepi

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

enkripsi

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

kunci enkripsi

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

endianness

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

titik akhir

Lihat titik akhir layanan.

layanan endpoint

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

perencanaan sumber daya perusahaan (ERP)

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

enkripsi amplop

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

lingkungan

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

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

epik

Dalam metodologi tangkas, kategori fungsional yang membantu mengatur dan memprioritaskan pekerjaan Anda. Epik memberikan deskripsi tingkat tinggi tentang persyaratan dan tugas

implementasi. Misalnya, epos keamanan AWS CAF mencakup manajemen identitas dan akses, kontrol detektif, keamanan infrastruktur, perlindungan data, dan respons insiden. Untuk informasi selengkapnya tentang epos dalam strategi AWS migrasi, lihat panduan implementasi program.

ERP

Lihat perencanaan sumber daya perusahaan.

analisis data eksplorasi (EDA)

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

F

tabel fakta

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

gagal cepat

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

batas isolasi kesalahan

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

cabang fitur

Lihat cabang.

fitur

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

pentingnya fitur

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

transformasi fitur

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

FGAC

Lihat kontrol akses berbutir halus.

kontrol akses berbutir halus (FGAC)

Penggunaan beberapa kondisi untuk mengizinkan atau menolak permintaan akses.

migrasi flash-cut

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

G

pemblokiran geografis

Lihat pembatasan geografis.

pembatasan geografis (pemblokiran geografis)

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

Alur kerja Gitflow

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

strategi greenfield

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

pagar pembatas

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

H

HA

Lihat ketersediaan tinggi.

migrasi database heterogen

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

ketersediaan tinggi (HA)

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

modernisasi sejarawan

Pendekatan yang digunakan untuk memodernisasi dan meningkatkan sistem teknologi operasional (OT) untuk melayani kebutuhan industri manufaktur dengan lebih baik. Sejarawan adalah jenis database yang digunakan untuk mengumpulkan dan menyimpan data dari berbagai sumber di pabrik.

migrasi database homogen

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

data panas

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

perbaikan terbaru

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

periode hypercare

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

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 VPC masuk, keluar, dan inspeksi untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

migrasi inkremental

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

Industri 4.0

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

infrastruktur

Semua sumber daya dan aset yang terkandung dalam lingkungan aplikasi.

infrastruktur sebagai kode (IAc)

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

Internet of Things industri (IIoT)

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

inspeksi VPC

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

Internet of Things (IoT)

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

interpretabilitas

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

IoT

Lihat Internet of Things.

Perpustakaan informasi TI (ITIL)

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

Manajemen layanan TI (ITSM)

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

ITIL

Lihat perpustakaan informasi TI.

ITSM

Lihat manajemen layanan TI.

L

kontrol akses berbasis label (LBAC)

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

landing zone

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

migrasi besar

Migrasi 300 atau lebih server.

LBAC

Lihat kontrol akses berbasis label.

hak istimewa paling sedikit

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

angkat dan geser

Lihat 7 Rs.

sistem endian kecil

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

lingkungan yang lebih rendah

Lihat lingkungan.

M

pembelajaran mesin (ML)

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

cabang utama

Lihat cabang.

malware

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

layanan terkelola

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 melalui API yang terdefinisi dengan baik dan biasanya dimiliki oleh tim kecil yang mandiri. Misalnya, sistem asuransi mungkin mencakup layanan mikro yang memetakan kemampuan bisnis, seperti penjualan atau pemasaran, atau subdomain, seperti pembelian, klaim, atau analitik. Manfaat layanan mikro termasuk kelincahan, penskalaan yang fleksibel, penyebaran yang mudah, kode yang dapat digunakan kembali, dan ketahanan. Untuk informasi selengkapnya, lihat Mengintegrasikan layanan mikro dengan menggunakan layanan tanpa AWS server.

arsitektur microservices

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

Program Percepatan Migrasi (MAP)

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

migrasi dalam skala

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

pabrik migrasi

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

metadata migrasi

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

pola migrasi

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

Penilaian Portofolio Migrasi (MPA)

Alat online yang menyediakan informasi untuk memvalidasi kasus bisnis untuk bermigrasi ke Cloud. AWS 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 Cloud. AWS 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 selengkapnya, lihat Mengevaluasi kesiapan modernisasi untuk aplikasi di Cloud. AWS

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 VPC masuk, keluar, dan inspeksi untuk melindungi antarmuka dua arah antara aplikasi Anda dan internet yang lebih luas.

P

batas izin

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

Informasi Identifikasi Pribadi (PII)

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

PII

Lihat informasi yang dapat diidentifikasi secara pribadi.

buku pedoman

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

PLC

Lihat pengontrol logika yang dapat diprogram.

PLM

Lihat manajemen siklus hidup produk.

kebijakan

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

persistensi 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 oleh Desain

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

zona host pribadi

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

informasi selengkapnya, lihat Bekerja dengan zona yang dihosting pribadi di dokumentasi Route 53.

kontrol proaktif

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

manajemen siklus hidup produk (PLM)

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

lingkungan produksi

Lihat lingkungan.

pengontrol logika yang dapat diprogram (PLC)

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

pseudonimisasi

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

terbitkan/berlangganan (pub/sub)

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

Q

rencana kueri

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

regresi rencana kueri

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

R

Matriks RACI

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

ransomware

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

Matriks RASCI

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

RCAC

Lihat kontrol akses baris dan kolom.

replika baca

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

arsitek ulang

Lihat 7 Rs.

tujuan titik pemulihan (RPO)

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

tujuan waktu pemulihan (RTO)

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

refactor

Lihat 7 Rs.

Wilayah

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

regresi

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

rehost

Lihat 7 Rs.

melepaskan

Dalam proses penyebaran, tindakan mempromosikan perubahan pada lingkungan produksi.

memindahkan

Lihat 7 Rs.

memplatform ulang

Lihat 7 Rs.

pembelian kembali

Lihat 7 Rs.

ketahanan

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

kebijakan berbasis sumber daya

Kebijakan yang dilampirkan ke sumber daya, seperti bucket Amazon S3, titik akhir, atau kunci enkripsi. Jenis kebijakan ini menentukan 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). Jenis dukungan (S) adalah opsional. Jika Anda menyertakan dukungan, matriks disebut matriks RASCI, dan jika Anda mengecualikannya, itu disebut matriks RACI.

kontrol responsif

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

melestarikan

Lihat 7 Rs.

pensiun

Lihat 7 Rs.

rotasi

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

PENIPUAN

Lihat kontrol pengawasan dan akuisisi data.

SCP

Lihat kebijakan kontrol layanan.

Rahasia

Dalam AWS Secrets Manager, informasi rahasia atau terbatas, seperti kata sandi atau 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 Rahasia dalam dokumentasi Secrets Manager.

kontrol keamanan

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

pengerasan keamanan

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

sistem informasi keamanan dan manajemen acara (SIEM)

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

otomatisasi respons keamanan

Tindakan yang telah ditentukan dan diprogram yang dirancang untuk secara otomatis merespons atau memulihkan peristiwa keamanan. Otomatisasi ini berfungsi sebagai kontrol keamanan detektif atau responsif yang membantu Anda menerapkan praktik terbaik AWS keamanan. Contoh tindakan respons otomatis termasuk memodifikasi grup keamanan VPC, menambal instans Amazon EC2, 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 SCP menentukan pagar pembatas atau menetapkan batasan pada tindakan yang dapat didelegasikan oleh administrator kepada pengguna atau peran. Anda dapat menggunakan SCP sebagai daftar izin atau daftar penolakan, untuk menentukan layanan atau tindakan mana yang diizinkan atau dilarang. Untuk informasi selengkapnya, lihat Kebijakan kontrol layanan dalam AWS Organizations dokumentasi.

titik akhir layanan

URL titik masuk untuk file 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 waktu kerja dan kinerja layanan.

indikator tingkat layanan (SLI)

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

tujuan tingkat layanan (SLO)

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

model tanggung jawab bersama

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

SIEM

Lihat informasi keamanan dan sistem manajemen acara.

titik kegagalan tunggal (SPOF)

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

SLA

Lihat perjanjian tingkat layanan.

SLI

Lihat indikator tingkat layanan.

SLO

Lihat tujuan tingkat layanan.

split-and-seed model

Pola untuk menskalakan dan mempercepat proyek modernisasi. Ketika fitur baru dan rilis produk didefinisikan, tim inti berpisah untuk membuat tim produk baru. Ini membantu meningkatkan kemampuan dan layanan organisasi Anda, meningkatkan produktivitas pengembang, dan

mendukung inovasi yang cepat. Untuk informasi lebih lanjut, lihat Pendekatan bertahap untuk memodernisasi aplikasi di. AWS Cloud

SPOF

Lihat satu titik kegagalan.

skema bintang

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

pola ara pencekik

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

subnet

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

kontrol pengawasan dan akuisisi data (SCADA)

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

enkripsi simetris

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

pengujian sintetis

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

T

tag

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

variabel target

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

daftar tugas

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

lingkungan uji

Lihat lingkungan.

pelatihan

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

gerbang transit

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

alur kerja berbasis batang

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

akses tepercaya

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

penyetelan

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

tim dua pizza

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

U

waswas

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

tugas yang tidak terdiferensiasi

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

lingkungan atas

Lihat lingkungan.

V

menyedot debu

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

kendali versi

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

Peering VPC

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

kerentanan

Kelemahan perangkat lunak atau perangkat keras yang membahayakan keamanan sistem.

W

cache hangat

Cache buffer yang berisi data 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 Kerangka Kualifikasi Beban Kerja AWS.

tulis sekali, baca banyak (WORM)

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

Z

eksploitasi zero-day

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

kerentanan zero-day

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

aplikasi zombie

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

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