Ein schrittweiser Ansatz für die Leistungstechnik in der AWS Cloud

AWS Präskriptive Leitlinien

AWS Präskriptive Leitlinien: Ein schrittweiser Ansatz für die Leistungstechnik in der AWS Cloud

Table of Contents

Ein schrittweiser Ansatz für Performance Engineering in der AWS Cloud

Amazon Web Services (Mitwirkende)

April 2024 (Verlauf der Dokumente)

In diesem Leitfaden werden die bewährten Methoden für die Planung, Erstellung und Aktivierung von Performance Engineering für Anwendungs-Workloads beschrieben, die auf Amazon Web Services ausgeführt werden (AWS). Es legt vier Säulen für Performance Engineering fest und schlägt verschiedene Ansätze vor, um die Leistungsanforderungen von Anwendungen zu erfüllen. Für jede Säule sind in diesem Leitfaden Tools und Lösungen für die Einrichtung von Leistungstests und der Testumgebung aufgeführt.

Was ist Performance Engineering?

Performance Engineering umfasst die Techniken, die während des Entwicklungszyklus eines Systems angewendet werden, um sicherzustellen, dass die nicht funktionalen Leistungsanforderungen (wie Durchsatz, Latenz oder Speichernutzung) erfüllt werden.

Bevor die Leistungstests beginnen, müssen Sie die Leistungsumgebung einrichten. Eine typische Leistungsumgebung basiert auf den folgenden Säulen:

- Generierung von Testdaten
- Beobachtbarkeit testen
- Automatisierung von Tests
- Testberichterstattung

Warum Performance Engineering verwenden?

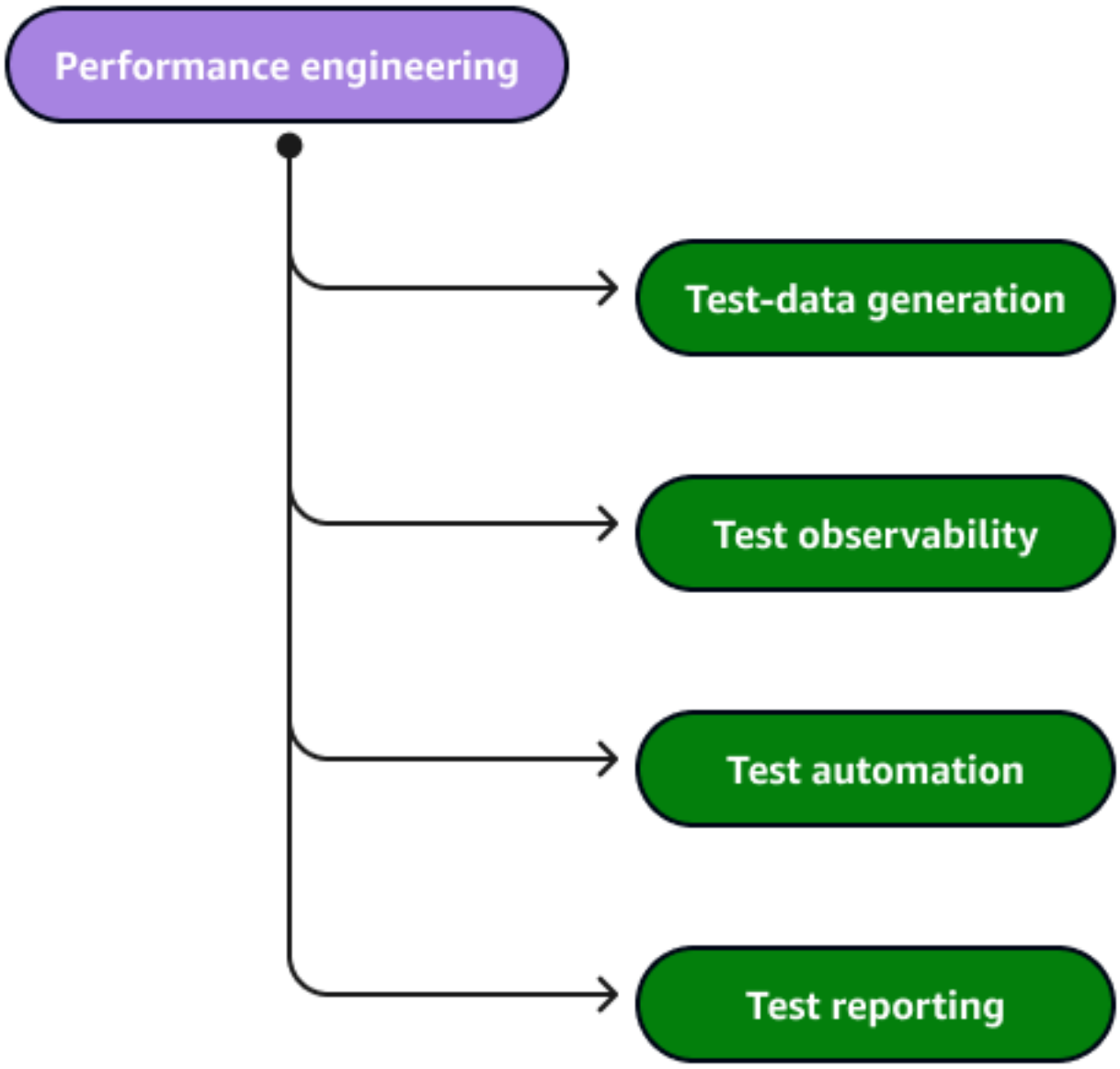
Performance Engineering ist der Prozess der kontinuierlichen Optimierung der Anwendungsleistung von Beginn der Entwurfsphase an. Es bietet einen großen Mehrwert für das Unternehmen, da Nacharbeiten und Refactoring von Code zu einem späteren Zeitpunkt im Entwicklungszyklus vermieden werden. Wenn das Performance-Engineering in der Entwurfsphase beginnt, führt dies zu einer Anwendung, die eine bessere Leistung erbringt, da die Leistung beim Design berücksichtigt

werden kann. Performance Engineering erfordert die aktive Beteiligung von Systemarchitekten DevOps, Entwicklern und der Qualitätssicherung.

Die Säulen des Performance Engineering

Um eine Performance-Engineering-Denkweise zu fördern, ist es wichtig, ein solides Fundament zu schaffen und gleichzeitig Performance Engineering für die Anwendung einzurichten. Performance Engineering erfordert die Einrichtung von vier Hauptpfeilern:

- Generierung von Testdaten — Leistungsingenieure richten Tools zur Generierung der Testdaten ein.
- Testbeobachtbarkeit — Leistungstechniker richten die Beobachtungsumgebung ein, um sicherzustellen, dass der Leistungslauf protokolliert und zurückverfolgt werden kann und dass die Ressourcen, die die Lasten verarbeiten, überwacht werden.
- Testautomatisierung — Performance-Techniker entwickeln mithilfe von Tools wie Apache JMeter oder ghz automatisierte Tests, die den Benutzerverkehr und die Systemlast simulieren.
- Testberichterstattung — Es werden Daten über die Konfiguration jedes Testlaufs zusammen mit den Leistungsergebnissen gesammelt. Die Daten ermöglichen es, Konfigurationsänderungen mit der Leistung zu korrelieren, und liefern wertvolle Erkenntnisse.



Die Einbeziehung dieser Säulen wird die Leistungsorientierung bereits in den Anfangsphasen des Entwurfs fördern. Dadurch können Änderungen an der Anwendung oder Umgebung in späteren Entwicklungs- und Testphasen vermieden werden.

Generierung von Testdaten

Die Generierung von Testdaten beinhaltet die Generierung und Verwaltung einer großen Datenmenge für die Ausführung des Leistungstestfalls. Diese generierten Daten dienen als Eingabe für die Testfälle, sodass die Anwendung an einer Vielzahl von Daten getestet werden kann.

Oft ist das Generieren von Testdaten ein komplexer Prozess. Die Verwendung schlecht erstellter Datensätze kann jedoch zu einem unvorhersehbaren Anwendungsverhalten in der Produktionsumgebung führen. Die Generierung von Testdaten für Leistungstests unterscheidet sich von herkömmlichen Ansätzen zur Generierung von Testdaten. Es erfordert reale Szenarien, und die meisten Kunden möchten ihre Workloads mit Daten testen, die ihren tatsächlichen Produktionsdaten ähneln. Generierte Testdaten müssen in der Regel auch nach jedem Testlauf in ihren ursprünglichen Zustand zurückgesetzt oder aktualisiert werden, was den Zeit- und Arbeitsaufwand erhöht.

Die Generierung von Testdaten beinhaltet die folgenden wichtigen Überlegungen:

- Genauigkeit — Die Genauigkeit der Daten ist in allen Aspekten des Testens wichtig. Ungenaue Daten führen zu ungenauen Ergebnissen. Wenn beispielsweise eine Kreditkartentransaktion generiert wird, sollte sie nicht für ein Datum in der future gelten.
- Gültigkeit — Die Daten sollten für den Anwendungsfall gültig sein. Beim Testen von Kreditkartentransaktionen ist es beispielsweise nicht ratsam, 10.000 Transaktionen pro Benutzer und Tag zu generieren, da dies erheblich vom gültigen Anwendungsszenario abweicht.
- Automatisierung — Die Automatisierung der Testdatengenerierung kann sich positiv auf den Zeitaufwand auswirken. Dies führt auch zu einer effektiven Testautomatisierung. Die manuelle Generierung von Testdaten kann sich auf die Qualitäts- und Zeitanforderungen auswirken.

  Es gibt verschiedene Mechanismen, die je nach Anwendungsfall wie folgt angewendet werden können:

  - API-gesteuert — In diesem Fall stellt der Entwickler eine API zur Generierung von Testdaten bereit, die der Tester zur Generierung von Daten verwenden kann. Mithilfe von Testtools wie JMeter können Tester die Datengenerierung mithilfe einer Business-API skalieren. Wenn Sie beispielsweise über eine API zum Hinzufügen eines Benutzers verfügen, können Sie dieselbe API verwenden, um Hunderte von Benutzern mit unterschiedlichen Profilen zu erstellen. In ähnlicher Weise können Sie die Benutzer löschen, indem Sie den Vorgang zum Löschen der API aufrufen. Für komplexe Workflow-Anwendungen kann der Entwickler eine zusammengesetzte API bereitstellen, mit der Datensätze aus verschiedenen Komponenten generiert werden können. Mithilfe dieses Ansatzes können Tester automatisierte Funktionen schreiben, um die Datensätze auf der Grundlage ihrer Anforderungen zu generieren und zu löschen.

    Wenn das System jedoch komplex ist oder die API-Antwortzeit pro Aufruf hoch ist, kann es lange dauern, die Daten einzurichten und zu entfernen.
  - Anweisungsgesteuert durch SQL-Anweisungen — Ein alternativer Ansatz besteht darin, Backend-SQL-Anweisungen zu verwenden, um eine große Datenmenge zu generieren. Der

Entwickler kann vorlagenbasierte SQL-Anweisungen für die Generierung von Testdaten bereitstellen. Tester können die Anweisungen verwenden, um Daten zu füllen, oder sie können zusätzlich zu diesen Anweisungen Wrapper-Skripten erstellen, um die Testdatengenerierung zu automatisieren. Mit diesem Ansatz können Tester Daten sehr schnell auffüllen und löschen, falls die Daten nach Abschluss des Tests zurückgesetzt werden müssen. Dieser Ansatz erfordert jedoch direkten Zugriff auf die Datenbank der Anwendung, was in einer typischen sicheren Umgebung möglicherweise nicht möglich ist. Darüber hinaus können ungültige Abfragen zu einer falschen Datenfüllung führen, was zu verzerrten Ergebnissen führen kann. Entwickler müssen außerdem die SQL-Anweisungen im Anwendungscode ständig aktualisieren, um Änderungen widerzuspiegeln, die im Laufe der Zeit an der Anwendung vorgenommen wurden.

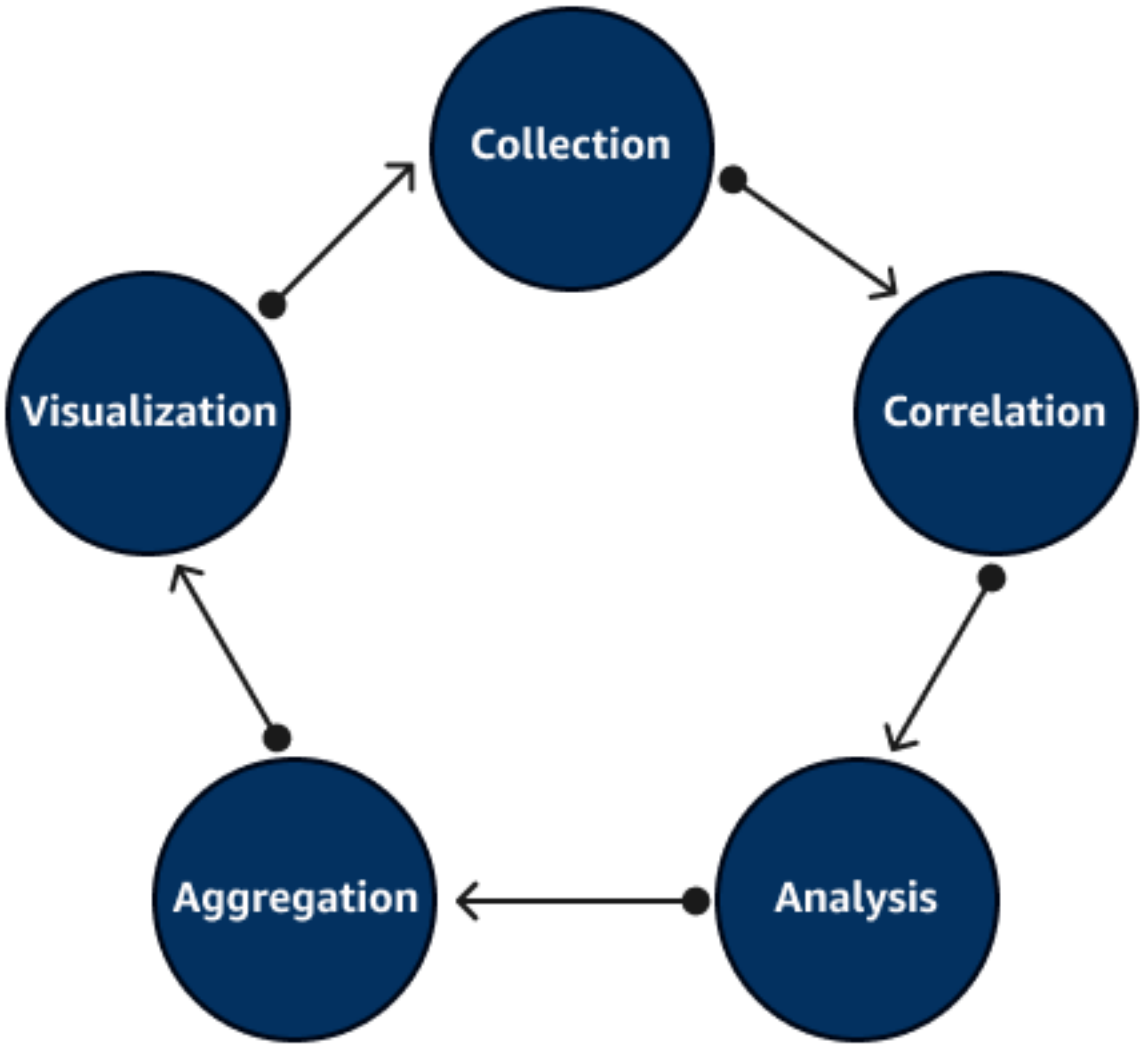
Tools zur Generierung von Testdaten

AWS bietet native benutzerdefinierte Tools, die Sie für die Generierung von Testdaten verwenden können:

- Amazon Kinesis Data Generator — Der Amazon Kinesis Data Generator (KDG) vereinfacht das Generieren von Daten und deren Senden an Amazon Kinesis. Das Tool bietet eine benutzerfreundliche Benutzeroberfläche, die direkt in Ihrem Browser ausgeführt wird. Weitere Informationen und eine Referenzimplementierung finden Sie im Blogbeitrag Testen Sie Ihre Streaming-Datenlösung mit dem neuen Amazon Kinesis Data Generator.
- AWS Glue Test Data Generator — Der AWS Glue Test Data Generator bietet ein konfigurierbares Framework für die Generierung von Testdaten mithilfe AWS Glue PySpark serverloser Jobs. Die erforderliche Beschreibung der Testdaten ist über eine YAML-Konfigurationsdatei vollständig konfigurierbar. Weitere Informationen und eine Referenzimplementierung finden Sie im AWS Glue Test Data Generator-Repository. GitHub

Testen Sie die Beobachtbarkeit

Test Observability unterstützt das Sammeln, Korrelieren, Aggregieren und Analysieren von Telemetriedaten in Ihrem Netzwerk, Ihrer Infrastruktur und Ihren Anwendungen während der Leistungstestläufe. Sie erhalten umfassende Einblicke in das Verhalten, die Leistung und den Zustand Ihres Systems. Diese Erkenntnisse helfen Ihnen, Probleme schneller zu erkennen, zu untersuchen und zu beheben. Durch das Hinzufügen von künstlicher Intelligenz und maschinellem Lernen können Sie proaktiv auf Probleme reagieren, diese vorhersagen und verhindern.



Die Beobachtbarkeit hängt von der Protokollierung, Überwachung und Rückverfolgung ab. Die Verantwortung für die erfolgreiche Umsetzung dieser Aktivitäten liegt bei den Anwendungs- und Infrastrukturteams.

Zu Beginn der Entwurfsphase sollten die Anwendungsteams den aktuellen Status ihres Observability-Stacks, einschließlich Protokollierung, Überwachung und Rückverfolgung, kennen. Sie können dann Tools auswählen, die sich reibungsloser in den Observability-Stack integrieren lassen.

In ähnlicher Weise ist das Infrastrukturteam für die Verwaltung und Skalierung der Observability-Infrastruktur verantwortlich.

Beachten Sie die folgenden Aspekte in Bezug auf die Testbeobachtbarkeit:

- Verfügbarkeit von Anwendungsprotokollen und Ablaufverfolgungen

- Korrelation von Protokollen und Traces
- Verfügbarkeit von Knoten, Containern und Anwendungsmetriken
- Automatisierung zur Einrichtung und Aktualisierung der Observability-Infrastruktur bei Bedarf
- Fähigkeit, die Telemetrie zu visualisieren
- Skalierung der Observability-Infrastruktur

Protokollierung

Bei der Protokollierung werden Daten über Ereignisse gespeichert, die in einem System auftreten. Das Protokoll kann Probleme, Fehler oder Informationen über den aktuellen Vorgang enthalten. Protokolle können in verschiedene Typen eingeteilt werden, z. B. in die folgenden:

- Protokoll der Ereignisse
- Serverprotokoll
- Systemprotokoll
- Autorisierungs- und Zugriffsprotokolle
- Prüfungsprotokolle

Ein Entwickler kann die Protokolle nach bestimmten Fehlercodes oder Mustern durchsuchen, sie nach bestimmten Feldern filtern oder sie für future Analysen sicher archivieren. Protokolle helfen dem Entwickler bei der Ursachenanalyse für Leistungsprobleme und bei der Korrelation zwischen Systemkomponenten.

Der Aufbau einer effektiven Logging-Lösung erfordert eine enge Abstimmung zwischen den Anwendungs- und Infrastrukturteams. Anwendungsprotokolle sind nur dann nützlich, wenn es eine skalierbare Protokollierungsinfrastruktur gibt, die Anwendungsfälle wie Parsen, Filtern, Puffern und Korrelation von Protokollen unterstützt. Gängige Anwendungsfälle, wie das Generieren einer Korrelations-ID, das Protokollieren der Laufzeit für geschäftskritische Methoden und das Definieren von Protokollmustern, können vereinfacht werden.

Anwendungsteam

Ein Anwendungsentwickler muss sicherstellen, dass die generierten Protokolle den bewährten Protokollierungsmethoden entsprechen. Zu den bewährten Methoden gehören die folgenden:

- Generierung von Korrelations-IDs zur Nachverfolgung eindeutiger Anfragen

- Protokollierung des Zeitaufwands geschäftskritischer Methoden
- Protokollierung auf einer geeigneten Protokollebene
- Gemeinsame Nutzung einer gemeinsamen Logging-Bibliothek

Wenn Sie Anwendungen entwerfen, die mit verschiedenen Microservices interagieren, verwenden Sie diese Prinzipien des Logging-Designs, um das Filtern und Extrahieren von Protokollen im Backend zu vereinfachen.

Generierung von Korrelations-IDs zur Nachverfolgung eindeutiger Anfragen

Wenn die Anwendung die Anfrage erhält, kann sie überprüfen, ob bereits eine Korrelations-ID im Header vorhanden ist. Wenn keine ID vorhanden ist, sollte die Anwendung eine ID generieren. Ein Application Load Balancer fügt beispielsweise einen Header namens `X-Amzn-Trace-Id` hinzu. Die Anwendung kann den Header verwenden, um die Anfrage vom Load Balancer mit der Anwendung zu korrelieren. In ähnlicher Weise sollte die Anwendung beim Aufrufen abhängiger Microservices eine Eingabe vornehmen, sodass die von verschiedenen Komponenten in einem Anforderungsablauf generierten Protokolle `traceId` korreliert werden.

Protokollierung der Zeit, die unternehmenskritische Methoden in Anspruch nehmen

Wenn die Anwendung eine Anfrage erhält, interagiert sie mit einer anderen Komponente. Die Anwendung sollte die Zeit, die für geschäftskritische Methoden benötigt wird, in einem definierten Muster protokollieren. Dies kann das Analysieren der Protokolle im Backend erleichtern. Es kann Ihnen auch helfen, nützliche Erkenntnisse aus den Protokollen zu gewinnen. Sie können Ansätze wie aspektorientierte Programmierung (AOP) verwenden, um solche Protokolle zu generieren, sodass Sie die Probleme mit der Protokollierung von Ihrer Geschäftslogik trennen können.

Protokollierung auf einer geeigneten Protokollebene

Die Anwendung sollte Protokolle schreiben, die eine hilfreiche Menge an Informationen enthalten. Verwenden Sie Protokollebenen, um Ereignisse nach ihrem Schweregrad zu kategorisieren. Verwenden Sie beispielsweise die `ERROR` Werte `WARNING` und Stufen für wichtige Ereignisse, die untersucht werden müssen. Verwenden Sie `INFO` und `DEBUG` für eine detaillierte Nachverfolgung und für Ereignisse mit hohem Datenvolumen. Richten Sie Log-Handler so ein, dass nur die Ebenen erfasst werden, die für die Produktion erforderlich sind. Es ist nicht hilfreich, zu viel Logging `INFO` auf dieser Ebene zu generieren, und das erhöht den Druck auf die Backend-Infrastruktur. `DEBUG`Protokollierung kann nützlich sein, sollte aber mit Vorsicht verwendet werden. Die Verwendung

von DEBUG Protokollen kann eine große Datenmenge erzeugen, weshalb sie in Umgebungen mit Leistungstests nicht empfohlen wird.

Gemeinsame Nutzung einer gemeinsamen Logging-Bibliothek

Die Anwendungsteams sollten eine gemeinsame Protokollierungsbibliothek verwenden, z. B. AWS SDK für Javamit einem vordefinierten gemeinsamen Protokollierungsmuster, das Entwickler als Abhängigkeiten in ihrem Projekt verwenden können.

Das Infrastrukturteam

DevOps Techniker können den Aufwand reduzieren, indem sie beim Filtern und Extrahieren von Protokollen im Backend die folgenden Entwurfsprinzipien für die Protokollierung verwenden. Das Infrastrukturteam muss die folgenden Ressourcen einrichten und unterstützen.

Agent protokollieren

Ein Log-Agent (Log Shipper) ist ein Programm, das Logs von einem Standort liest und an einen anderen Ort sendet. Protokollagenten werden verwendet, um auf einem Computer gespeicherte Protokolldateien zu lesen und Protokollereignisse zur Zentralisierung in das Backend hochzuladen.

Protokolle sind unstrukturierte Daten, die strukturiert werden müssen, bevor Sie aussagekräftige Erkenntnisse aus ihnen gewinnen können. Protokollagenten verwenden Parser, um Protokollanweisungen zu lesen und relevante Felder wie Zeitstempel, Protokollebene und Dienstname zu extrahieren, und strukturieren diese Daten in ein JSON-Format. Ein einfacher Log-Agent am Edge ist nützlich, da er zu einer geringeren Ressourcenauslastung führt. Der Log-Agent kann direkt an das Backend weiterleiten, oder er kann einen zwischengeschalteten Log-Forwarder verwenden, der die Daten an das Backend weiterleitet. Durch die Verwendung eines Log-Forwarders wird die Arbeit von den Log-Agenten an der Quelle entlastet.

Log-Parser

Ein Log-Parser konvertiert die unstrukturierten Logs in strukturierte Logs. Log-Agent-Parser bereichern die Protokolle auch durch Hinzufügen von Metadaten. Die Datenanalyse der Daten kann an der Quelle (Anwendungsseite) oder zentral erfolgen. Das Schema zum Speichern der Protokolle sollte erweiterbar sein, sodass Sie neue Felder hinzufügen können. Wir empfehlen die Verwendung von Standardprotokollformaten wie JSON. In einigen Fällen müssen die Protokolle jedoch zur besseren Suche in JSON-Formate umgewandelt werden. Das Schreiben des richtigen Parser-Ausdrucks ermöglicht eine effiziente Transformation.

Protokolliert das Backend

Ein Logs-Backend-Dienst sammelt, erfasst und visualisiert Protokolldaten aus verschiedenen Quellen. Der Log-Agent kann direkt in das Backend schreiben oder einen zwischengeschalteten Log-Forwarder verwenden. Achten Sie beim Leistungstest darauf, die Protokolle zu speichern, damit sie zu einem späteren Zeitpunkt durchsucht werden können. Speichern Sie die Protokolle für jede Anwendung separat im Backend. Verwenden Sie beispielsweise einen speziellen Index für eine Anwendung und verwenden Sie das Indexmuster, um nach Protokollen zu suchen, die über verschiedene verwandte Anwendungen verteilt sind. Wir empfehlen, Daten für die Protokollsuche mindestens 7 Tage zu speichern. Eine längere Speicherung der Daten kann jedoch zu unnötigen Speicherkosten führen. Da während des Leistungstests eine große Menge an Protokollen generiert wird, ist es wichtig, dass die Protokollierungsinfrastruktur das Logging-Backend skaliert und die richtige Größe hat.

Visualisierung von Protokollen

Um aussagekräftige und umsetzbare Erkenntnisse aus Anwendungsprotokollen zu gewinnen, verwenden Sie spezielle Visualisierungstools, um die Protokollrohdaten zu verarbeiten und in grafische Darstellungen umzuwandeln. Visualisierungen wie Diagramme, Grafiken und Dashboards können dabei helfen, Trends, Muster und Anomalien aufzudecken, die bei der Betrachtung der Rohprotokolle möglicherweise nicht ohne weiteres erkennbar sind.

Zu den wichtigsten Vorteilen der Verwendung von Visualisierungstools gehört die Fähigkeit, Daten über mehrere Systeme und Anwendungen hinweg zu korrelieren, um Abhängigkeiten und Engpässe zu identifizieren. Interaktive Dashboards unterstützen die detaillierte Untersuchung der Daten auf unterschiedlichen Granularitätsebenen, um Probleme zu beheben oder Nutzungstrends zu erkennen. Spezialisierte Datenvisualisierungsplattformen bieten Funktionen wie Analysen, Warnmeldungen und Datenaustausch, die die Überwachung und Analyse verbessern können.

Durch die Nutzung der Leistungsfähigkeit der Datenvisualisierung in Anwendungsprotokollen können sich Entwicklungs- und Betriebsteams einen Überblick über die System- und Anwendungsleistung verschaffen. Die gewonnenen Erkenntnisse können für eine Vielzahl von Zwecken genutzt werden, darunter zur Optimierung der Effizienz, zur Verbesserung der Benutzererfahrung, zur Verbesserung der Sicherheit und zur Kapazitätsplanung. Das Endergebnis sind Dashboards, die auf verschiedene Stakeholder zugeschnitten sind und at-a-glance Ansichten bieten, in denen Protokolldaten zu umsetzbaren und aufschlussreichen Informationen zusammengefasst werden.

Automatisierung der Logging-Infrastruktur

Da verschiedene Anwendungen unterschiedliche Anforderungen haben, ist es wichtig, die Installation und den Betrieb der Protokollierungsinfrastruktur zu automatisieren. Verwenden Sie Infrastructure-as-Code-Tools (IaC), um das Backend der Protokollierungsinfrastruktur bereitzustellen. Anschließend können Sie die Protokollierungsinfrastruktur entweder als gemeinsam genutzten Dienst oder als unabhängige, maßgeschneiderte Bereitstellung für eine bestimmte Anwendung bereitstellen.

Wir empfehlen Entwicklern, Continuous Delivery (CD) -Pipelines zu verwenden, um Folgendes zu automatisieren:

- Stellen Sie die Protokollierungsinfrastruktur bei Bedarf bereit und bauen Sie sie ab, wenn sie nicht benötigt wird.
- Stellen Sie Log-Agenten für verschiedene Ziele bereit.
- Stellen Sie Log-Parser- und Forwarder-Konfigurationen bereit.
- Stellen Sie Anwendungs-Dashboards bereit.

Tools für die Protokollierung

AWS bietet native Protokollierungs-, Alarm- und Dashboard-Dienste. Im Folgenden finden Sie beliebte Ressourcen AWS-Services und Ressourcen für die Protokollierung:

- Amazon OpenSearch Service unterstützt Unternehmen dabei, Protokolldaten aus verschiedenen Quellen zu sammeln, aufzunehmen und zu visualisieren. Weitere Informationen finden Sie unter Zentralisierte Protokollierung mit OpenSearch.
- Amazon CloudWatch Agent und AWS for Fluent Bit sind die beliebtesten Log-Agenten auf AWS. Informationen zur Verwendung des CloudWatch Agenten mit Amazon CloudWatch Logs Insights finden Sie im Blogbeitrag Simplifying Apache server logs with Amazon CloudWatch Logs Insights. Informationen AWS zur Fluent Bit-Referenzimplementierung finden Sie im Blogbeitrag Centralized Container Logging with Fluent Bit.

Überwachen

Überwachung ist der Prozess, bei dem verschiedene Messwerte wie CPU und Arbeitsspeicher gesammelt und in einer Zeitreihendatenbank wie Amazon Managed Service for Prometheus gespeichert werden. Das Überwachungssystem kann Push- oder Pull-basiert sein. In Push-

basierten Systemen überträgt die Quelle regelmäßig Metriken in die Zeitreihendatenbank. In Pull-basierten Systemen erfasst der Scraper Metriken aus verschiedenen Quellen und speichert sie in der Zeitreihendatenbank. Entwickler können die Metriken analysieren, filtern und sie im Zeitverlauf grafisch darstellen, um die Leistung zu visualisieren. Die erfolgreiche Implementierung von Monitoring kann in zwei große Bereiche unterteilt werden: Anwendung und Infrastruktur.

Für Anwendungsentwickler sind die folgenden Kennzahlen von entscheidender Bedeutung:

- Latenz — Die Zeit, die benötigt wird, um eine Antwort zu erhalten
- Anforderungsdurchsatz — Die Gesamtzahl der Anfragen, die pro Sekunde bearbeitet wurden
- Fehlerrate bei Anfragen — Die Gesamtzahl der Fehler

Erfassen Sie die Ressourcenauslastung, die Auslastung und die Anzahl der Fehler für jede Ressource (z. B. den Anwendungscontainer, die Datenbank), die an der Geschäftstransaktion beteiligt ist. Bei der Überwachung der CPU-Auslastung können Sie beispielsweise die durchschnittliche CPU-Auslastung, die durchschnittliche Last und die Spitzenlast während des Leistungstestlaufs verfolgen. Wenn eine Ressource während eines Stresstests die Sättigung erreicht, während eines Leistungslaufs für einen kürzeren Zeitraum jedoch möglicherweise nicht die Sättigung erreicht.

Metriken

Anwendungen können zur Überwachung ihrer Anwendungen unterschiedliche Aktuatoren verwenden, z. B. Spring-Boot-Aktuatoren. Diese produktionstauglichen Bibliotheken stellen in der Regel einen REST-Endpunkt zur Überwachung von Informationen über die laufenden Anwendungen bereit. Die Bibliotheken können die zugrunde liegende Infrastruktur, Anwendungsplattformen und andere Ressourcen überwachen. Wenn eine der Standardmetriken die Anforderungen nicht erfüllt, muss der Entwickler benutzerdefinierte Metriken implementieren. Benutzerdefinierte Metriken können dabei helfen, wichtige Leistungsindikatoren (KPIs) für Unternehmen nachzuverfolgen, die nicht anhand von Daten aus Standardimplementierungen verfolgt werden können. Möglicherweise möchten Sie beispielsweise einen Geschäftsvorgang nachverfolgen, z. B. die Latenz bei der API-Integration eines Drittanbieters oder die Gesamtzahl der abgeschlossenen Transaktionen.

Kardinalität

Die Kardinalität bezieht sich auf die Anzahl der eindeutigen Zeitreihen einer Metrik. Metriken sind beschriftet, um zusätzliche Informationen bereitzustellen. Beispielsweise gibt eine REST-basierte Anwendung, die die Anzahl der Anfragen für eine bestimmte API verfolgt, eine Kardinalität von 1 an.

Wenn Sie ein Benutzerlabel hinzufügen, um die Anzahl der Anfragen pro Benutzer zu identifizieren, steigt die Kardinalität proportional zur Anzahl der Benutzer. Durch Hinzufügen von Labels, die für Kardinalität sorgen, können Sie Metriken nach verschiedenen Gruppen aufteilen. Es ist wichtig, die richtigen Labels für den richtigen Anwendungsfall zu verwenden, da die Kardinalität die Anzahl der Metrikreihen in der Backend-Monitoring-Zeitreihendatenbank erhöht.

Auflösung

In einem typischen Monitoring-Setup ist die Überwachungsanwendung so konfiguriert, dass sie die Metriken regelmäßig aus der Anwendung entfernt. Die Periodizität des Scrapings definiert die Granularität der Überwachungsdaten. In kürzeren Intervallen erfasste Messwerte bieten tendenziell einen genaueren Überblick über die Leistung, da mehr Datenpunkte verfügbar sind. Die Auslastung der Zeitreihendatenbank nimmt jedoch zu, je mehr Einträge gespeichert werden. In der Regel entspricht eine Granularität von 60 Sekunden der Standardauflösung und 1 Sekunde der hohen Auflösung.

DevOps Team

Anwendungsentwickler bitten DevOps Ingenieure häufig, eine Überwachungsumgebung zur Visualisierung von Kennzahlen der Infrastruktur und der Anwendungen einzurichten. Der DevOps Techniker muss eine Umgebung einrichten, die skalierbar ist und die vom Anwendungsentwickler verwendeten Datenvisualisierungstools unterstützt. Dabei werden Überwachungsdaten aus verschiedenen Quellen gesammelt und an eine zentrale Zeitreihendatenbank wie Amazon Managed Service for Prometheus gesendet.

Überwachung des Backends

Ein Monitoring-Backend-Service unterstützt die Erfassung, Speicherung, Abfrage und Visualisierung von Metrikdaten. In der Regel handelt es sich um eine Zeitreihendatenbank wie Amazon Managed Service for Prometheus oder InfluxDB. InfluxData Mithilfe eines Service-Discovery-Mechanismus kann der Monitoring Collector Metriken aus verschiedenen Quellen sammeln und speichern. Bei Leistungstests ist es wichtig, die Metrikdaten zu speichern, damit sie zu einem späteren Zeitpunkt durchsucht werden können. Wir empfehlen, Daten für Metriken mindestens 15 Tage zu speichern. Das Speichern der Metriken über einen längeren Zeitraum bietet jedoch keine wesentlichen Vorteile und führt zu unnötigen Speicherkosten. Da der Leistungstest eine große Menge an Metriken generieren kann, ist es wichtig, dass die Metrikinfrastruktur skaliert und gleichzeitig eine schnelle Abfrageleistung bietet. Der Monitoring-Backend-Service bietet eine Abfragesprache, mit der die Metrikdaten angezeigt werden können.

Visualisierung

Stellen Sie Visualisierungstools bereit, mit denen die Anwendungsdaten angezeigt werden können, um aussagekräftige Einblicke zu erhalten. Der DevOps Techniker und der Anwendungsentwickler sollten die Abfragesprache für das Monitoring-Backend erlernen und eng zusammenarbeiten, um eine Dashboard-Vorlage zu erstellen, die wiederverwendet werden kann. Geben Sie in den Dashboards Latenz und Fehler an und zeigen Sie gleichzeitig die Ressourcennutzung und -auslastung in der gesamten Infrastruktur und den Anwendungsressourcen an.

Automatisierung der Überwachungsinfrastruktur

Ähnlich wie bei der Protokollierung ist es wichtig, die Installation und den Betrieb der Überwachungsinfrastruktur zu automatisieren, damit Sie den unterschiedlichen Anforderungen verschiedener Anwendungen gerecht werden können. Verwenden Sie IaC-Tools, um das Backend der Überwachungsinfrastruktur bereitzustellen. Anschließend können Sie die Überwachungsinfrastruktur entweder als gemeinsam genutzten Dienst oder als unabhängige, maßgeschneiderte Bereitstellung für eine bestimmte Anwendung bereitstellen.

Verwenden Sie CD-Pipelines, um Folgendes zu automatisieren:

- Stellen Sie die Überwachungsinfrastruktur bei Bedarf bereit und bauen Sie sie ab, wenn sie nicht benötigt wird.
- Aktualisieren Sie die Überwachungskonfiguration, um Metriken zu filtern oder zu aggregieren.
- Stellen Sie Anwendungs-Dashboards bereit.

Überwachungstools

Amazon Managed Service for Prometheus ist ein Prometheus-kompatibler Überwachungsservice für Container-Infrastruktur und Anwendungsmetriken für Container, mit dem Sie Container-Umgebungen in großem Maßstab sicher überwachen können. Weitere Informationen finden Sie im Blogbeitrag Erste Schritte mit Amazon Managed Service for Prometheus.

Amazon CloudWatch bietet Full-Stack-Überwachung für AWS. CloudWatch unterstützt sowohl AWS native als auch Open-Source-Lösungen, sodass Sie jederzeit nachvollziehen können, was in Ihrem Technologie-Stack passiert.

Zu den nativen AWS Tools gehören die folgenden:

- CloudWatch Amazon-Dashboards

- CloudWatch Container Insights
- CloudWatch Metriken
- CloudWatch Alarme

Amazon CloudWatch bietet speziell entwickelte Funktionen für spezifische Anwendungsfälle wie die Containerüberwachung über CloudWatch Container Insights. Diese Funktionen sind integriert, CloudWatch sodass Sie Protokolle, die Erfassung und Überwachung von Metriken einrichten können.

Verwenden Sie Container Insights für Ihre containerisierten Anwendungen und Microservices, um Metriken und Protokolle zu sammeln, zu aggregieren und zusammenzufassen. Container Insights ist für Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) und Kubernetes-Plattformen auf Amazon Elastic Compute Cloud (Amazon EC2) verfügbar. Container Insights sammelt Daten als Leistungsprotokollereignisse im eingebetteten metrischen Format. Diese Leistungsprotokollereigniseinträge verwenden ein strukturiertes JSON-Schema, das die Erfassung und Speicherung von Daten mit hoher Kardinalität in großem Maßstab unterstützt.

Informationen zur Implementierung von Container Insights mit Amazon EKS finden Sie im Blogbeitrag Introducing Amazon CloudWatch Container Insights for Amazon EKS Fargate using AWS Distro for. OpenTelemetry

Nachverfolgung

Die Ablaufverfolgung beinhaltet die spezielle Verwendung von Protokollierungsinformationen über die Prozesse eines Programms. Die Erkenntnisse aus den Protokollen können Technikern helfen, einzelne Transaktionen zu debuggen und Engpässe zu identifizieren. Die Ablaufverfolgung kann automatisch oder mithilfe manueller Instrumente aktiviert werden.

Da eine Anwendung in verschiedene Dienste integriert ist, ist es wichtig, die Leistung der Anwendung und der zugrunde liegenden Dienste zu ermitteln. Tracing funktioniert mit Traces und Spans. Ein Trace ist der komplette Anforderungsprozess, und jeder Trace besteht aus Spans. Eine Spanne ist ein markiertes Zeitintervall und bezeichnet die Aktivität innerhalb der einzelnen Komponenten oder Dienste eines Systems. Ablaufverfolgungen geben einen Überblick darüber, was passiert, wenn eine Anfrage an eine Anwendung gestellt wird.

Bewerbungsteam

Anwendungsentwickler instrumentieren ihre Anwendungen, indem sie Trace-Daten für eingehende und ausgehende Anfragen und andere Ereignisse innerhalb der Anwendung zusammen mit

Metadaten zu jeder Anfrage senden. Um Traces zu generieren, muss eine Anwendung so instrumentiert sein, dass sie Traces generiert. Die Instrumentierung kann automatisch oder manuell erfolgen.

Automatische Instrumentierung

Mithilfe der automatischen Instrumentierung können Sie Telemetriedaten aus einer Anwendung erfassen, ohne den Quellcode ändern zu müssen. Agenten für automatische Instrumentierung können Anwendungsspuren einer Anwendung oder eines Dienstes generieren. In der Regel verwenden Sie Konfigurationsänderungen, um den Agenten oder einen anderen Mechanismus hinzuzufügen.

Bei der Bibliotheksinstrumentierung werden nur minimale Änderungen am Anwendungscode vorgenommen, um vorgefertigte Instrumentierung hinzuzufügen. Die Instrumentierung zielt auf bestimmte Bibliotheken oder Frameworks ab, z. B. das AWS SDK, Apache HTTP-Clients oder SQL-Clients.

Manuelle Instrumentierung

Bei diesem Ansatz fügen Anwendungsentwickler der Anwendung an jedem Ort, an dem sie Trace-Informationen sammeln möchten, Instrumentierungscode hinzu. Verwenden Sie beispielsweise aspektorientierte Programmierung (AOP), um Ablaufverfolgungsdaten zu sammeln. AWS X-Ray Entwickler können SDKs zur Instrumentierung ihrer Anwendungen verwenden.

Sampling

Trace-Daten werden häufig in großen Mengen generiert. Es ist wichtig, über einen Mechanismus zu verfügen, mit dem festgelegt werden kann, ob die Trace-Daten exportiert werden sollen oder nicht. Bei der Probenahme wird bestimmt, welche Daten exportiert werden sollen. Dies geschieht in der Regel, um Kosten zu sparen. Durch die Anpassung der Stichprobenregeln können Sie die Menge der aufgenommenen Daten steuern. Sie können auch das Sampling-Verhalten ändern, ohne Ihren Code zu ändern und erneut bereitzustellen. Es ist wichtig, die Abtastrate zu kontrollieren, um die richtige Anzahl an Traces zu erzeugen.

Anwendungsentwickler können die Traces mit Anmerkungen versehen, indem sie Metadaten als Schlüssel-Wert-Paare hinzufügen. Die Anmerkungen bereichern die Traces und helfen, die Filterung im Backend zu verfeinern.

DevOps Team

DevOps Techniker werden häufig gebeten, eine Ablaufverfolgungsumgebung für den Anwendungsentwickler einzurichten, um Traces für Infrastruktur und Anwendungen zu visualisieren. Bei der Einrichtung der Ablaufverfolgungsumgebung werden Trace-Daten aus verschiedenen Quellen gesammelt und zur Visualisierung an einen zentralen Speicher gesendet.

Das Backend für die Nachverfolgung

Ein Tracing-Backend ist ein Dienst, der Daten über Anfragen sammelt AWS X-Ray , die Ihre Anwendung bedient. Es bietet Tools, mit denen Sie diese Daten anzeigen, filtern und Einblicke in sie gewinnen können, um Probleme und Optimierungsmöglichkeiten zu identifizieren. Für jede verfolgte Anfrage an Ihre Anwendung können Sie detaillierte Informationen zu der Anfrage und Antwort sowie zu anderen Aufrufen einsehen, die Ihre Anwendung an nachgelagerte AWS Ressourcen, Microservices, Datenbanken und Web-APIs sendet.

Automatisieren der Ablaufverfolgung

Da verschiedene Anwendungen unterschiedliche Tracing-Anforderungen haben, ist es wichtig, die Konfiguration und den Betrieb der Tracing-Infrastruktur zu automatisieren. Verwenden Sie IaC-Tools, um das Backend der Tracing-Infrastruktur bereitzustellen.

Verwenden Sie CD-Pipelines, um Folgendes zu automatisieren:

- Stellen Sie die Tracing-Infrastruktur bei Bedarf bereit und bauen Sie sie ab, wenn sie nicht benötigt wird.
- Stellen Sie die Ablaufverfolgungskonfiguration anwendungsübergreifend bereit.

Tools zur Nachverfolgung

AWS stellt die folgenden Dienste für die Ablaufverfolgung und die zugehörige Visualisierung bereit:

- AWS X-Ray empfängt Traces von Ihrer Anwendung, zusätzlich zu Traces von AWS Diensten, die Ihre Anwendung verwendet und die bereits in X-Ray integriert sind. Es gibt mehrere SDKs, Agenten und Tools, mit denen Sie Ihre Anwendung für die Röntgenverfolgung instrumentieren können. Weitere Informationen finden Sie in der AWS X-Ray -Dokumentation.

Entwickler können AWS X-Ray SDKs auch verwenden, um Traces an X-Ray zu senden. AWS X-Ray stellt SDKs fürGo,Java, Node.jsPython, .NET und bereit. Ruby Jedes X-Ray-SDK bietet Folgendes:

- Interceptors, die Sie Ihrem Code hinzufügen können, um eingehende HTTP-Anforderungen nachzuverfolgen.
- Client-Handler zur Instrumentierung von AWS SDK-Clients, die Ihre Anwendung verwendet, um andere AWS Dienste aufzurufen
- Ein HTTP-Client zur Instrumentierung von Aufrufen an andere interne und externe HTTP-Webdienste

X-Ray-SDKs unterstützen auch die Instrumentierung von Aufrufen von SQL-Datenbanken, automatische AWS SDK-Client-Instrumentierung und andere Funktionen. Anstatt Trace-Daten direkt an X-Ray zu senden, sendet das SDK JSON-Segmentdokumente an einen Daemon-Prozess, der auf UDP-Verkehr wartet. Der X-Ray-Daemon puffert Segmente in einer Warteschlange und lädt sie stapelweise auf X-Ray hoch. Weitere Informationen zur Instrumentierung Ihrer Anwendung mithilfe eines X-Ray-SDK finden Sie in der X-Ray-Dokumentation.

- Amazon OpenSearch Service ist ein AWS verwalteter Service zum Ausführen und Skalieren von OpenSearch Clustern, der zum zentralen Speichern von Protokollen, Metriken und Traces verwendet werden kann. Das Beobachtbarkeits-Plug-In bietet ein einheitliche Erlebnis zum Erfassen und Überwachen von Metriken, Protokollen und Traces aus gängigen Datenquellen. Die Datenerfassung und -überwachung an einem zentralen Ort ermöglicht die vollständige Überwachung end-to-end Ihrer gesamten Infrastruktur. Informationen zur Implementierung finden Sie in der OpenSearch Servicedokumentation.
- AWS Distro for OpenTelemetry (ADOT) ist eine AWS Distribution, die auf dem Projekt Cloud Native Computing Foundation (CNCF) basiert. OpenTelemetry ADOT bietet derzeit Unterstützung für automatische Instrumentierung für Java und Python. Darüber hinaus unterstützt ADOT die automatische Instrumentierung von AWS Lambda Funktionen und ihren Downstream-Anfragen mithilfe Java von Node.js und Python Laufzeiten über ADOT Managed Lambda Layers. Entwickler können den ADOT-Collector verwenden, um Traces an verschiedene Backends zu senden, einschließlich Amazon AWS X-Ray OpenSearch Service.

  Ein Referenzbeispiel für die Instrumentierung Ihrer Anwendung mithilfe des ADOT SDK finden Sie in der Dokumentation. Ein Referenzbeispiel für die Verwendung des ADOT SDK zum Senden von Daten an Amazon OpenSearch Service finden Sie in der OpenSearch Servicedokumentation.

Ein Referenzbeispiel dafür, wie Sie Ihre auf Amazon EKS ausgeführte Anwendung instrumentieren können, finden Sie im Blogbeitrag Erfassung von Metriken und Traces mithilfe von Amazon EKS-Add-Ons für AWS Distro for OpenTelemetry.

Automatisierung von Tests

Automatisierte Tests mit einem speziellen Framework und speziellen Tools können menschliche Eingriffe reduzieren und die Qualität maximieren. Automatisierte Leistungstests unterscheiden sich nicht von Automatisierungstests wie Komponententests und Integrationstests.

Verwenden Sie DevOps Pipelines in den verschiedenen Phasen für Leistungstests.

Die fünf Phasen der Testautomatisierungspipeline sind:

1. Einrichtung — Verwenden Sie für diese Phase die im Abschnitt Testdatengenerierung beschriebenen Testdatenansätze. Die Generierung realistischer Testdaten ist entscheidend, um valide Testergebnisse zu erhalten. Sie müssen sorgfältig verschiedene Testdaten erstellen, die eine Vielzahl von Anwendungsfällen abdecken und den Live-Produktionsdaten weitgehend entsprechen. Bevor Sie umfassende Leistungstests durchführen, müssen Sie möglicherweise erste Testtests durchführen, um die Testskripte, Umgebungen und Überwachungstools zu validieren.
2. Testtool — Um die Leistungstests durchzuführen, wählen Sie ein geeignetes Lasttest-Tool aus, z. B. JMeter oder ghz. Überlegen Sie, welches Modell am besten zu Ihren Geschäftsanforderungen passt, was die Simulation realer Benutzerlasten angeht.
3. Testlauf — Nachdem die Testtools und Umgebungen eingerichtet sind, können Sie end-to-end Leistungstests für eine Reihe erwarteter Benutzerlasten und -dauern durchführen. Überwachen Sie während des gesamten Tests den Zustand des getesteten Systems genau. Dies ist in der Regel eine lang andauernde Phase. Überwachen Sie die Fehlerquoten für die automatische Invalidierung von Tests und beenden Sie den Test, wenn zu viele Fehler vorliegen.

   Das Tool für Lasttests bietet Einblicke in die Ressourcennutzung, Reaktionszeiten und potenzielle Engpässe.

4. Testberichte — Sammeln Sie die Testergebnisse zusammen mit der Anwendungs- und Testkonfiguration. Automatisieren Sie die Erfassung der Anwendungskonfiguration, der Testkonfiguration und der Ergebnisse, was die Aufzeichnung und zentrale Speicherung der Daten im Zusammenhang mit Leistungstests erleichtert. Durch die zentrale Verwaltung von Leistungsdaten erhalten Sie gute Einblicke und können Erfolgskriterien für Ihr Unternehmen programmatisch definieren.
5. Aufräumen — Nachdem Sie einen Leistungstestlauf abgeschlossen haben, setzen Sie die Testumgebung und die Daten zurück, um sich auf nachfolgende Testläufe vorzubereiten. Zunächst machen Sie alle während des Testlaufs an den Testdaten vorgenommenen Änderungen rückgängig. Sie müssen die Datenbanken und anderen Datenspeicher in ihren ursprünglichen Zustand zurückversetzen und alle während des Tests generierten neuen, aktualisierten oder gelöschten Datensätze wiederherstellen.

Sie können die Pipeline wiederverwenden, um den Test mehrmals zu wiederholen, bis die Ergebnisse die gewünschte Leistung wiedergeben. Sie können die Pipeline auch verwenden, um zu überprüfen, ob Codeänderungen die Leistung nicht beeinträchtigen. Sie können Tests zur Codevalidierung außerhalb der Geschäftszeiten durchführen und die verfügbaren Test- und Beobachtbarkeitsdaten zur Fehlerbehebung verwenden.

Zu den bewährten Methoden gehören die folgenden:

- Erfassen Sie die Start- und Endzeit und generieren Sie automatisch URLs für die Protokollierung. Auf diese Weise können Sie Observability-Daten in den entsprechenden Zeitfenstern filtern. Überwachungs- und Ablaufverfolgungssysteme.
- Fügen Sie beim Aufrufen der Tests Testkennungen in den Header ein. Anwendungsentwickler können ihre Logging-, Überwachungs- und Tracing-Daten erweitern, indem sie den Identifier als Filter im Backend verwenden.
- Beschränken Sie die Pipeline auf jeweils nur einen Lauf. Das gleichzeitige Ausführen von Tests erzeugt Geräusche, die bei der Fehlerbehebung zu Verwirrung führen können. Es ist auch wichtig, den Test in einer speziellen Leistungsumgebung auszuführen.

Tools zur Testautomatisierung

Testtools spielen bei jeder Testautomatisierung eine wichtige Rolle. Zu den beliebtesten Optionen für Open-Source-Testtools gehören:

- Apache Meter ist das erfahrene Kraftpferd. Im Laufe der Jahre ist Apache JMeter zuverlässiger geworden und hat zusätzliche Features hinzugefügt. Mit der grafischen Oberfläche können Sie komplexe Tests erstellen, ohne eine Programmiersprache zu kennen. Unternehmen wie BlazeMeter unterstützen Apache JMeter.
- K6 ist ein kostenloses Tool, das Support, Hosting der Lastquelle und eine integrierte Weboberfläche zur Organisation, Ausführung und Analyse von Lasttests bietet.
- Der Vegeta-Lasttest folgt einem anderen Konzept. Anstatt Parallelität zu definieren oder Ihr System zu belasten, definieren Sie eine bestimmte Rate. Das Tool erstellt diese Last dann unabhängig von den Reaktionszeiten Ihres Systems.
- Hey and ab, das Apache HTTP Server-Benchmarking-Tool, sind grundlegende Tools, die Sie von der Befehlszeile aus verwenden können, um die angegebene Last auf einem einzelnen Endpunkt auszuführen. Dies ist der schnellste Weg, um Last zu erzeugen, wenn Sie über einen Server verfügen, auf dem die Tools ausgeführt werden können. Selbst ein lokaler Laptop ist leistungsfähig, obwohl er möglicherweise nicht leistungsfähig genug ist, um eine hohe Last zu erzeugen.
- ghz ist ein Befehlszeilenprogramm und ein Go-Paket für Lasttests und Benchmarking-gRPC-Dienste.

AWS stellt die AWS Lösung für Distributed Load Testing bereit. Die Lösung erstellt und simuliert Tausende von verbundenen Benutzern, die in konstantem Tempo Transaktionsdatensätze generieren, ohne dass Server bereitgestellt werden müssen. Weitere Informationen finden Sie in der AWS Lösungsbibliothek.

Sie können sie verwenden AWS CodePipeline , um die Pipeline für Leistungstests zu automatisieren. Weitere Informationen zur Automatisierung Ihrer API-Tests mithilfe CodePipeline von Using finden Sie im AWS DevOps Blog und in der AWS Dokumentation.

Testberichterstattung

Testberichte beziehen sich auf die Erfassung, Analyse und Präsentation von Daten, die sich auf die Leistung von Systemen, Anwendungen, Diensten oder Prozessen beziehen. Dabei werden verschiedene Metriken und Indikatoren gemessen, um die Effizienz, Reaktionsfähigkeit, Zuverlässigkeit und Gesamteffektivität eines bestimmten Systems oder einer Komponente zu bewerten.

Bei der Berichterstattung über Leistungstests werden relevante Kennzahlen auf der Grundlage des Kontextes und der Ziele der Analyse ausgewählt. Zu den gängigen Leistungskennzahlen gehören

Reaktionszeiten, Durchsatz, Fehlerraten, Ressourcenauslastung (CPU, Arbeitsspeicher, Festplatte) und Netzwerklatenz.

Nachdem die leistungsbezogenen Daten erfasst wurden, müssen sie in einem zentralen Repository gespeichert werden. Diese Testergebnisse können aus verschiedenen Umgebungen, Anwendungen und Testtools stammen. Wenn Sie mehrere Workloads in unterschiedlichen Umgebungen ausführen, ist es schwierig, leistungsbezogene Daten zu sammeln und diese Datenpunkte miteinander zu korrelieren, um fundierte Schlüsse zu ziehen. Wir empfehlen, eine Standardmethode für die Erfassung von Leistungskennzahldaten mithilfe eines zentralen Repositorys für die Datenspeicherung und -visualisierung zu definieren.

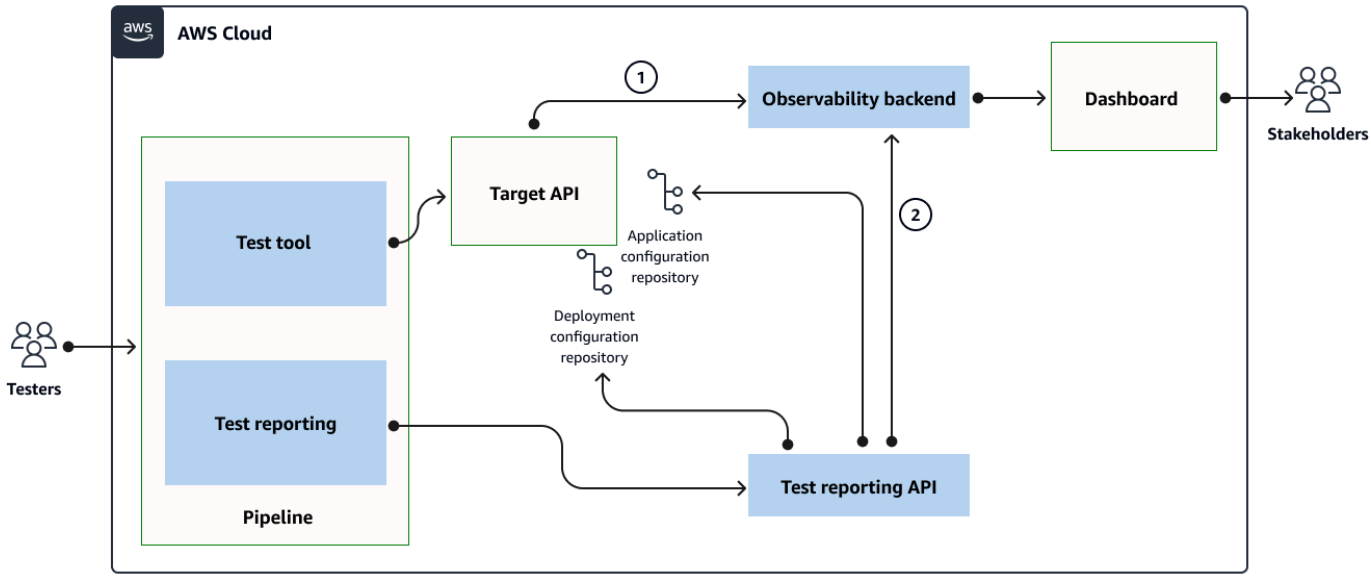
Standardisierte Aufzeichnung

Wir empfehlen, die Art und Weise, wie verschiedene Stakeholder die Leistungstests durchführen, zu standardisieren und die resultierenden Daten in ein zentrales Repository zu schreiben. Dies könnte beispielsweise in Form einer API geschehen, die die Ergebnisse akzeptiert und in einer persistenten Speicherlösung speichert. In Situationen, in denen Daten aus Quellen wie GitOps Amazon Managed Service for Prometheus abgerufen werden müssen, kann die API diese Details direkt aus den angegebenen Quellen abrufen, basierend auf Schemadateien, die beschreiben, wie die Felder aus Bereitstellungsspezifikationen und Kubernetes-Spezifikationen extrahiert werden. Die Schemadateien können `JSONPath` Ausdrücke oder Prometheus Query Language (PromQL) verwenden. Wie bereits erwähnt, sollten die gesammelten Kennzahlen für den Kontext und die Ziele der Leistungsanalyse relevant sein.

Die an die API übergebenen Daten können Details und Tags enthalten, die sich auf die Anwendung und die Umgebung beziehen, für die der Test durchgeführt wurde. Dies hilft bei der Durchführung von Analysen der Leistungstestdaten.

Säulen der Leistungstechnik in Aktion

Die folgende Referenzarchitektur zeigt die Säulen der Performance-Entwicklung beim Testen einer bestimmten API.



1. Daten zum Protokollieren, Überwachen und Nachverfolgen werden von der Ziel-API an das Backend gesendet.
2. Wenn die Testberichts-API aufgerufen wird, sendet sie Ergebnisse und Konfigurationsinformationen an das Backend.

Die Kernkomponente ist die Ziel-API oder Anwendung, die getestet wird. Die Ziel-API wird auf die gleiche GitOps Weise mit dem Repository für die Anwendungskonfiguration und dem Repository für die Bereitstellungskonfiguration synchronisiert, um die neuesten Anwendungs- und Infrastrukturkonfigurationen zu erhalten. Durch diese Synchronisation können die automatisierten Tests mit dem aktuellen gewünschten Status der Anwendung und ihrer unterstützenden Infrastruktur, wie in den Git-Repositorys definiert, ausgeführt werden.

Die Testautomatisierungspipeline automatisiert die Generierung der Testdaten, die Ausführung des Tests und die Berichterstattung über die Testergebnisse für die Ziel-API.

Die Ziel-API generiert Performance-Erkenntnisse (Metriken, Logs und Traces) unter Verwendung von Best Practices im Bereich Observability und streamt Metrikdaten an das Observability-Backend.

Die Testberichts-API sammelt alle testbezogenen Berichtsdaten (Konfiguration und Testergebnisse) und speichert sie im Observability-Backend.

Die Aggregation von Leistungseinblicken und Berichtsdaten (Konfiguration, Testergebnisse) hilft Ihnen dabei, leistungsbezogene Daten für die Ziel-API abzufragen. Sie könnten beispielsweise Folgendes fragen:

- Was sind die zehn langsamsten Transaktionen?
- Was ist die durchschnittliche Anzahl der Tests auf P99, P90?
- Wie lassen sich die Konfigurationen der beiden Testläufe vergleichen?

Die Korrelation von Testfällen mit Ergebnissen, Konfigurationen und Kennzahlen über einen bestimmten Zeitraum hilft bei der Identifizierung der besten Konfiguration und der Leistungsergebnisse.

Mithilfe dieser Testergebnisse können Sie genauere, datengestützte Entscheidungen für die API treffen und sich darauf verlassen, dass die API in Produktion geht.

Ressourcen

AWS-Services

- Amazon CloudWatch
- AWS CodePipeline
- AWS Distribution für OpenTelemetry
- OpenSearch Amazon-Dienst
- AWS X-Ray

Implementierungen

- amazon-kinesis-data-generator
- AWS Glue Testen Sie den Datengenerator
- Testen verteilter Lasten auf AWS

Blog-Posts

- Zentralisierte Container-Protokollierung mit Fluent Bit
- Testen Sie Ihre Streaming-Datenlösung mit dem neuen Amazon Kinesis Data Generator
- Wir stellen vor: Amazon CloudWatch Container Insights für Amazon EKS Fargate mit AWS Distro für OpenTelemetry
- Anwendungsverfolgung auf Kubernetes mit AWS X-Ray
- Erfassung von Metriken und Traces mit Amazon EKS-Add-Ons für AWS Distro für OpenTelemetry
- Erste Schritte mit Amazon Managed Service für Prometheus

Werkstatt

- Einführung in die AWS Observability

AWS Präskriptive Leitlinien

- Anwendungen zum Testen von Lasten (Leitfaden)

Anwendungen von Drittanbietern

- Apache JMeter
- K6
- Vegeta
- Hey und ab
- ghz

Mitwirkende

Zu den Mitwirkenden an diesem Dokument gehören:

- Varun Sharma, leitender Berater, AWS
- Akash Kumar, leitender Berater, AWS
- Archana Bhatnagar, Praxismanagerin, AWS
- Pratik Sharma, Professionelle Dienstleistungen II, AWS

Dokumentverlauf

In der folgenden Tabelle werden wichtige Änderungen in diesem Leitfaden beschrieben. Um Benachrichtigungen über zukünftige Aktualisierungen zu erhalten, können Sie einen RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Erste Veröffentlichung	—	24. April 2024

AWS Glossar zu präskriptiven Leitlinien

Die folgenden Begriffe werden häufig in Strategien, Leitfäden und Mustern von AWS Prescriptive Guidance verwendet. Um Einträge vorzuschlagen, verwenden Sie bitte den Link Feedback geben am Ende des Glossars.

Zahlen

7 Rs

Sieben gängige Migrationsstrategien für die Verlagerung von Anwendungen in die Cloud. Diese Strategien bauen auf den 5 Rs auf, die Gartner 2011 identifiziert hat, und bestehen aus folgenden Elementen:

- Faktorwechsel/Architekturwechsel – Verschieben Sie eine Anwendung und ändern Sie ihre Architektur, indem Sie alle Vorteile cloudnativer Feature nutzen, um Agilität, Leistung und Skalierbarkeit zu verbessern. Dies beinhaltet in der Regel die Portierung des Betriebssystems und der Datenbank. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank auf die Amazon Aurora PostgreSQL-kompatible Edition.
- Plattformwechsel (Lift and Reshape) – Verschieben Sie eine Anwendung in die Cloud und führen Sie ein gewisses Maß an Optimierung ein, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Amazon Relational Database Service (Amazon RDS) für Oracle in der. AWS Cloud
- Neukauf (Drop and Shop) – Wechseln Sie zu einem anderen Produkt, indem Sie typischerweise von einer herkömmlichen Lizenz zu einem SaaS-Modell wechseln. Beispiel: Migrieren Sie Ihr CRM-System (Customer Relationship Management) zu Salesforce.com.
- Hostwechsel (Lift and Shift) – Verschieben Sie eine Anwendung in die Cloud, ohne Änderungen vorzunehmen, um die Cloud-Funktionen zu nutzen. Beispiel: Migrieren Sie Ihre lokale Oracle-Datenbank zu Oracle auf einer EC2 Instanz in der. AWS Cloud
- Verschieben (Lift and Shift auf Hypervisor-Ebene) – Verlagern Sie die Infrastruktur in die Cloud, ohne neue Hardware kaufen, Anwendungen umschreiben oder Ihre bestehenden Abläufe ändern zu müssen. Sie migrieren Server von einer lokalen Plattform zu einem Cloud-Dienst für dieselbe Plattform. Beispiel: Migrieren Sie eine Microsoft Hyper-V Anwendung zu AWS.
- Beibehaltung (Wiederaufgreifen) – Bewahren Sie Anwendungen in Ihrer Quellumgebung auf. Dazu können Anwendungen gehören, die einen umfangreichen Faktorwechsel erfordern und

die Sie auf einen späteren Zeitpunkt verschieben möchten, sowie ältere Anwendungen, die Sie beibehalten möchten, da es keine geschäftliche Rechtfertigung für ihre Migration gibt.

- Außerbetriebnahme – Dekommissionierung oder Entfernung von Anwendungen, die in Ihrer Quellumgebung nicht mehr benötigt werden.

A

ABAC

Siehe attributbasierte Zugriffskontrolle.

abstrahierte Dienste

Weitere Informationen finden Sie unter Managed Services.

ACID

Siehe Atomarität, Konsistenz, Isolierung und Haltbarkeit.

Aktiv-Aktiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden (mithilfe eines bidirektionalen Replikationstools oder dualer Schreibvorgänge) und beide Datenbanken Transaktionen von miteinander verbundenen Anwendungen während der Migration verarbeiten. Diese Methode unterstützt die Migration in kleinen, kontrollierten Batches, anstatt einen einmaligen Cutover zu erfordern. Es ist flexibler, erfordert aber mehr Arbeit als eine aktiv-passive Migration.

Aktiv-Passiv-Migration

Eine Datenbankmigrationsmethode, bei der die Quell- und Zieldatenbanken synchron gehalten werden, aber nur die Quelldatenbank Transaktionen von verbindenden Anwendungen verarbeitet, während Daten in die Zieldatenbank repliziert werden. Die Zieldatenbank akzeptiert während der Migration keine Transaktionen.

Aggregatfunktion

Eine SQL-Funktion, die mit einer Gruppe von Zeilen arbeitet und einen einzelnen Rückgabewert für die Gruppe berechnet. Beispiele für Aggregatfunktionen sind `SUM` und`MAX`.

AI

Siehe künstliche Intelligenz.

AIOps

Siehe Operationen im Bereich künstliche Intelligenz.

Anonymisierung

Der Prozess des dauerhaften Löschens personenbezogener Daten in einem Datensatz. Anonymisierung kann zum Schutz der Privatsphäre beitragen. Anonymisierte Daten gelten nicht mehr als personenbezogene Daten.

Anti-Muster

Eine häufig verwendete Lösung für ein wiederkehrendes Problem, bei dem die Lösung kontraproduktiv, ineffektiv oder weniger wirksam als eine Alternative ist.

Anwendungssteuerung

Ein Sicherheitsansatz, bei dem nur zugelassene Anwendungen verwendet werden können, um ein System vor Schadsoftware zu schützen.

Anwendungsportfolio

Eine Sammlung detaillierter Informationen zu jeder Anwendung, die von einer Organisation verwendet wird, einschließlich der Kosten für die Erstellung und Wartung der Anwendung und ihres Geschäftswerts. Diese Informationen sind entscheidend für den Prozess der Portfoliofindung und -analyse und hilft bei der Identifizierung und Priorisierung der Anwendungen, die migriert, modernisiert und optimiert werden sollen.

künstliche Intelligenz (KI)

Das Gebiet der Datenverarbeitungswissenschaft, das sich der Nutzung von Computertechnologien zur Ausführung kognitiver Funktionen widmet, die typischerweise mit Menschen in Verbindung gebracht werden, wie Lernen, Problemlösen und Erkennen von Mustern. Weitere Informationen finden Sie unter Was ist künstliche Intelligenz?

Operationen mit künstlicher Intelligenz (AIOps)

Der Prozess des Einsatzes von Techniken des Machine Learning zur Lösung betrieblicher Probleme, zur Reduzierung betrieblicher Zwischenfälle und menschlicher Eingriffe sowie zur Steigerung der Servicequalität. Weitere Informationen zur Verwendung in der AWS Migrationsstrategie finden Sie im Operations Integration Guide. AIOps

Asymmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der ein Schlüsselpaar, einen öffentlichen Schlüssel für die Verschlüsselung und einen privaten Schlüssel für die Entschlüsselung verwendet. Sie können den öffentlichen Schlüssel teilen, da er nicht für die Entschlüsselung verwendet wird. Der Zugriff auf den privaten Schlüssel sollte jedoch stark eingeschränkt sein.

Atomizität, Konsistenz, Isolierung, Haltbarkeit (ACID)

Eine Reihe von Softwareeigenschaften, die die Datenvalidität und betriebliche Zuverlässigkeit einer Datenbank auch bei Fehlern, Stromausfällen oder anderen Problemen gewährleisten.

Attributbasierte Zugriffskontrolle (ABAC)

Die Praxis, detaillierte Berechtigungen auf der Grundlage von Benutzerattributen wie Abteilung, Aufgabenrolle und Teamname zu erstellen. Weitere Informationen finden Sie unter ABAC AWS in der AWS Identity and Access Management (IAM-) Dokumentation.

autoritative Datenquelle

Ein Ort, an dem Sie die primäre Version der Daten speichern, die als die zuverlässigste Informationsquelle angesehen wird. Sie können Daten aus der maßgeblichen Datenquelle an andere Speicherorte kopieren, um die Daten zu verarbeiten oder zu ändern, z. B. zu anonymisieren, zu redigieren oder zu pseudonymisieren.

Availability Zone

Ein bestimmter Standort innerhalb einer AWS-Region , der vor Ausfällen in anderen Availability Zones geschützt ist und kostengünstige Netzwerkkonnektivität mit niedriger Latenz zu anderen Availability Zones in derselben Region bietet.

AWS Framework für die Cloud-Einführung (AWS CAF)

Ein Framework mit Richtlinien und bewährten Verfahren, das Unternehmen bei der Entwicklung eines effizienten und effektiven Plans für den erfolgreichen Umstieg auf die Cloud unterstützt. AWS AWS CAF unterteilt die Leitlinien in sechs Schwerpunktbereiche, die als Perspektiven bezeichnet werden: Unternehmen, Mitarbeiter, Unternehmensführung, Plattform, Sicherheit und Betrieb. Die Perspektiven Geschäft, Mitarbeiter und Unternehmensführung konzentrieren sich auf Geschäftskompetenzen und -prozesse, während sich die Perspektiven Plattform, Sicherheit und Betriebsabläufe auf technische Fähigkeiten und Prozesse konzentrieren. Die Personalperspektive zielt beispielsweise auf Stakeholder ab, die sich mit Personalwesen (HR), Personalfunktionen und Personalmanagement befassen. Aus dieser Perspektive bietet AWS CAF Leitlinien für Personalentwicklung, Schulung und Kommunikation, um das Unternehmen auf eine erfolgreiche

Cloud-Einführung vorzubereiten. Weitere Informationen finden Sie auf der AWS -CAF-Webseite und dem AWS -CAF-Whitepaper.

AWS Workload-Qualifizierungsrahmen (AWS WQF)

Ein Tool, das Workloads bei der Datenbankmigration bewertet, Migrationsstrategien empfiehlt und Arbeitsschätzungen bereitstellt. AWS WQF ist in () enthalten. AWS Schema Conversion Tool AWS SCT Es analysiert Datenbankschemas und Codeobjekte, Anwendungscode, Abhängigkeiten und Leistungsmerkmale und stellt Bewertungsberichte bereit.

B

schlechter Bot

Ein Bot, der Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen soll.

BCP

Siehe Planung der Geschäftskontinuität.

Verhaltensdiagramm

Eine einheitliche, interaktive Ansicht des Ressourcenverhaltens und der Interaktionen im Laufe der Zeit. Sie können ein Verhaltensdiagramm mit Amazon Detective verwenden, um fehlgeschlagene Anmeldeversuche, verdächtige API-Aufrufe und ähnliche Vorgänge zu untersuchen. Weitere Informationen finden Sie unter Daten in einem Verhaltensdiagramm in der Detective-Dokumentation.

Big-Endian-System

Ein System, welches das höchstwertige Byte zuerst speichert. Siehe auch Endianness.

Binäre Klassifikation

Ein Prozess, der ein binäres Ergebnis vorhersagt (eine von zwei möglichen Klassen). Beispielsweise könnte Ihr ML-Modell möglicherweise Probleme wie „Handelt es sich bei dieser E-Mail um Spam oder nicht?“ vorhersagen müssen oder „Ist dieses Produkt ein Buch oder ein Auto?“

Bloom-Filter

Eine probabilistische, speichereffiziente Datenstruktur, mit der getestet wird, ob ein Element Teil einer Menge ist.

Blau/Grün-Bereitstellung

Eine Bereitstellungsstrategie, bei der Sie zwei separate, aber identische Umgebungen erstellen. Sie führen die aktuelle Anwendungsversion in einer Umgebung (blau) und die neue Anwendungsversion in der anderen Umgebung (grün) aus. Mit dieser Strategie können Sie schnell und mit minimalen Auswirkungen ein Rollback durchführen.

Bot

Eine Softwareanwendung, die automatisierte Aufgaben über das Internet ausführt und menschliche Aktivitäten oder Interaktionen simuliert. Manche Bots sind nützlich oder nützlich, wie z. B. Webcrawler, die Informationen im Internet indexieren. Einige andere Bots, sogenannte bösartige Bots, sollen Einzelpersonen oder Organisationen stören oder ihnen Schaden zufügen.

Botnetz

Netzwerke von Bots, die mit Malware infiziert sind und unter der Kontrolle einer einzigen Partei stehen, die als Bot-Herder oder Bot-Operator bezeichnet wird. Botnetze sind der bekannteste Mechanismus zur Skalierung von Bots und ihrer Wirkung.

branch

Ein containerisierter Bereich eines Code-Repositorys. Der erste Zweig, der in einem Repository erstellt wurde, ist der Hauptzweig. Sie können einen neuen Zweig aus einem vorhandenen Zweig erstellen und dann Feature entwickeln oder Fehler in dem neuen Zweig beheben. Ein Zweig, den Sie erstellen, um ein Feature zu erstellen, wird allgemein als Feature-Zweig bezeichnet. Wenn das Feature zur Veröffentlichung bereit ist, führen Sie den Feature-Zweig wieder mit dem Hauptzweig zusammen. Weitere Informationen finden Sie unter Über Branches (GitHub Dokumentation).

Zugang durch Glasbruch

Unter außergewöhnlichen Umständen und im Rahmen eines genehmigten Verfahrens ist dies eine schnelle Methode für einen Benutzer, auf einen Bereich zuzugreifen AWS-Konto , für den er in der Regel keine Zugriffsrechte besitzt. Weitere Informationen finden Sie unter dem Indikator Implementation break-glass procedures in den AWS Well-Architected-Leitlinien.

Brownfield-Strategie

Die bestehende Infrastruktur in Ihrer Umgebung. Wenn Sie eine Brownfield-Strategie für eine Systemarchitektur anwenden, richten Sie sich bei der Gestaltung der Architektur nach den Einschränkungen der aktuellen Systeme und Infrastruktur. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

Puffer-Cache

Der Speicherbereich, in dem die am häufigsten abgerufenen Daten gespeichert werden.

Geschäftsfähigkeit

Was ein Unternehmen tut, um Wert zu generieren (z. B. Vertrieb, Kundenservice oder Marketing). Microservices-Architekturen und Entwicklungsentscheidungen können von den Geschäftskapazitäten beeinflusst werden. Weitere Informationen finden Sie im Abschnitt Organisiert nach Geschäftskapazitäten des Whitepapers Ausführen von containerisierten Microservices in AWS.

Planung der Geschäftskontinuität (BCP)

Ein Plan, der die potenziellen Auswirkungen eines störenden Ereignisses, wie z. B. einer groß angelegten Migration, auf den Betrieb berücksichtigt und es einem Unternehmen ermöglicht, den Betrieb schnell wieder aufzunehmen.

C

CAF

Weitere Informationen finden Sie unter Framework für die AWS Cloud-Einführung.

Bereitstellung auf Kanaren

Die langsame und schrittweise Veröffentlichung einer Version für Endbenutzer. Wenn Sie sich sicher sind, stellen Sie die neue Version bereit und ersetzen die aktuelle Version vollständig.

CCoE

Weitere Informationen finden Sie im Cloud Center of Excellence.

CDC

Siehe Erfassung von Änderungsdaten.

Erfassung von Datenänderungen (CDC)

Der Prozess der Nachverfolgung von Änderungen an einer Datenquelle, z. B. einer Datenbanktabelle, und der Aufzeichnung von Metadaten zu der Änderung. Sie können CDC für verschiedene Zwecke verwenden, z. B. für die Prüfung oder Replikation von Änderungen in einem Zielsystem, um die Synchronisation aufrechtzuerhalten.

Chaos-Technik

Absichtliches Einführen von Ausfällen oder Störungsereignissen, um die Widerstandsfähigkeit eines Systems zu testen. Sie können AWS Fault Injection Service (AWS FIS) verwenden, um Experimente durchzuführen, die Ihre AWS Workloads stress, und deren Reaktion zu bewerten.

CI/CD

Siehe Continuous Integration und Continuous Delivery.

Klassifizierung

Ein Kategorisierungsprozess, der bei der Erstellung von Vorhersagen hilft. ML-Modelle für Klassifikationsprobleme sagen einen diskreten Wert voraus. Diskrete Werte unterscheiden sich immer voneinander. Beispielsweise muss ein Modell möglicherweise auswerten, ob auf einem Bild ein Auto zu sehen ist oder nicht.

clientseitige Verschlüsselung

Lokale Verschlüsselung von Daten, bevor das Ziel sie AWS-Service empfängt.

Cloud-Exzellenzzentrum (CCoE)

Ein multidisziplinäres Team, das die Cloud-Einführung in der gesamten Organisation vorantreibt, einschließlich der Entwicklung bewährter Cloud-Methoden, der Mobilisierung von Ressourcen, der Festlegung von Migrationszeitplänen und der Begleitung der Organisation durch groß angelegte Transformationen. Weitere Informationen finden Sie in den CCoE-Beiträgen im AWS Cloud Enterprise Strategy Blog.

Cloud Computing

Die Cloud-Technologie, die typischerweise für die Ferndatenspeicherung und das IoT-Gerätemanagement verwendet wird. Cloud Computing ist häufig mit Edge-Computing-Technologie verbunden.

Cloud-Betriebsmodell

In einer IT-Organisation das Betriebsmodell, das zum Aufbau, zur Weiterentwicklung und Optimierung einer oder mehrerer Cloud-Umgebungen verwendet wird. Weitere Informationen finden Sie unter Aufbau Ihres Cloud-Betriebsmodells.

Phasen der Einführung der Cloud

Die vier Phasen, die Unternehmen bei der Migration in der Regel durchlaufen AWS Cloud:

- Projekt – Durchführung einiger Cloud-bezogener Projekte zu Machbarkeitsnachweisen und zu Lernzwecken
- Fundament — Tätigen Sie grundlegende Investitionen, um Ihre Cloud-Einführung zu skalieren (z. B. Einrichtung einer landing zone, Definition eines CCo E, Einrichtung eines Betriebsmodells)
- Migration – Migrieren einzelner Anwendungen
- Neuentwicklung – Optimierung von Produkten und Services und Innovation in der Cloud

Diese Phasen wurden von Stephen Orban im Blogbeitrag The Journey Toward Cloud-First & the Stages of Adoption im AWS Cloud Enterprise Strategy-Blog definiert. Informationen darüber, wie sie mit der AWS Migrationsstrategie zusammenhängen, finden Sie im Leitfaden zur Vorbereitung der Migration.

CMDB

Siehe Datenbank für das Konfigurationsmanagement.

Code-Repository

Ein Ort, an dem Quellcode und andere Komponenten wie Dokumentation, Beispiele und Skripts gespeichert und im Rahmen von Versionskontrollprozessen aktualisiert werden. Zu den gängigen Cloud-Repositorys gehören GitHub oderBitbucket Cloud. Jede Version des Codes wird Zweig genannt. In einer Microservice-Struktur ist jedes Repository einer einzelnen Funktionalität gewidmet. Eine einzelne CI/CD-Pipeline kann mehrere Repositorien verwenden.

Kalter Cache

Ein Puffer-Cache, der leer oder nicht gut gefüllt ist oder veraltete oder irrelevante Daten enthält. Dies beeinträchtigt die Leistung, da die Datenbank-Instance aus dem Hauptspeicher oder der Festplatte lesen muss, was langsamer ist als das Lesen aus dem Puffercache.

Kalte Daten

Daten, auf die selten zugegriffen wird und die in der Regel historisch sind. Bei der Abfrage dieser Art von Daten sind langsame Abfragen in der Regel akzeptabel. Durch die Verlagerung dieser Daten auf leistungsschwächere und kostengünstigere Speicherstufen oder -klassen können Kosten gesenkt werden.

Computer Vision (CV)

Ein Bereich der KI, der maschinelles Lernen nutzt, um Informationen aus visuellen Formaten wie digitalen Bildern und Videos zu analysieren und zu extrahieren. Amazon SageMaker AI bietet beispielsweise Bildverarbeitungsalgorithmen für CV.

Drift in der Konfiguration

Bei einer Arbeitslast eine Änderung der Konfiguration gegenüber dem erwarteten Zustand. Dies kann dazu führen, dass der Workload nicht mehr richtlinienkonform wird, und zwar in der Regel schrittweise und unbeabsichtigt.

Verwaltung der Datenbankkonfiguration (CMDB)

Ein Repository, das Informationen über eine Datenbank und ihre IT-Umgebung speichert und verwaltet, inklusive Hardware- und Softwarekomponenten und deren Konfigurationen. In der Regel verwenden Sie Daten aus einer CMDB in der Phase der Portfolioerkennung und -analyse der Migration.

Konformitätspaket

Eine Sammlung von AWS Config Regeln und Abhilfemaßnahmen, die Sie zusammenstellen können, um Ihre Konformitäts- und Sicherheitsprüfungen individuell anzupassen. Mithilfe einer YAML-Vorlage können Sie ein Conformance Pack als einzelne Entität in einer AWS-Konto AND-Region oder unternehmensweit bereitstellen. Weitere Informationen finden Sie in der Dokumentation unter Conformance Packs. AWS Config

Kontinuierliche Bereitstellung und kontinuierliche Integration (CI/CD)

Der Prozess der Automatisierung der Quell-, Build-, Test-, Staging- und Produktionsphasen des Softwareveröffentlichungsprozesses. CI/CD is commonly described as a pipeline. CI/CDkann Ihnen helfen, Prozesse zu automatisieren, die Produktivität zu steigern, die Codequalität zu verbessern und schneller zu liefern. Weitere Informationen finden Sie unter Vorteile der kontinuierlichen Auslieferung. CD kann auch für kontinuierliche Bereitstellung stehen. Weitere Informationen finden Sie unterKontinuierliche Auslieferung im Vergleich zu kontinuierlicher Bereitstellung.

CV

Siehe Computer Vision.

D

Daten im Ruhezustand

Daten, die in Ihrem Netzwerk stationär sind, z. B. Daten, die sich im Speicher befinden.

Datenklassifizierung

Ein Prozess zur Identifizierung und Kategorisierung der Daten in Ihrem Netzwerk auf der Grundlage ihrer Kritikalität und Sensitivität. Sie ist eine wichtige Komponente jeder Strategie für das Management von Cybersecurity-Risiken, da sie Ihnen hilft, die geeigneten Schutz- und Aufbewahrungskontrollen für die Daten zu bestimmen. Die Datenklassifizierung ist ein Bestandteil der Sicherheitssäule im AWS Well-Architected Framework. Weitere Informationen finden Sie unter Datenklassifizierung.

Datendrift

Eine signifikante Variation zwischen den Produktionsdaten und den Daten, die zum Trainieren eines ML-Modells verwendet wurden, oder eine signifikante Änderung der Eingabedaten im Laufe der Zeit. Datendrift kann die Gesamtqualität, Genauigkeit und Fairness von ML-Modellvorhersagen beeinträchtigen.

Daten während der Übertragung

Daten, die sich aktiv durch Ihr Netzwerk bewegen, z. B. zwischen Netzwerkressourcen.

Datennetz

Ein architektonisches Framework, das verteilte, dezentrale Dateneigentum mit zentraler Verwaltung und Steuerung ermöglicht.

Datenminimierung

Das Prinzip, nur die Daten zu sammeln und zu verarbeiten, die unbedingt erforderlich sind. Durch Datenminimierung im AWS Cloud können Datenschutzrisiken, Kosten und der CO2-Fußabdruck Ihrer Analysen reduziert werden.

Datenperimeter

Eine Reihe präventiver Schutzmaßnahmen in Ihrer AWS Umgebung, die sicherstellen, dass nur vertrauenswürdige Identitäten auf vertrauenswürdige Ressourcen von erwarteten Netzwerken zugreifen. Weitere Informationen finden Sie unter Aufbau eines Datenperimeters auf. AWS

Vorverarbeitung der Daten

Rohdaten in ein Format umzuwandeln, das von Ihrem ML-Modell problemlos verarbeitet werden kann. Die Vorverarbeitung von Daten kann bedeuten, dass bestimmte Spalten oder Zeilen entfernt und fehlende, inkonsistente oder doppelte Werte behoben werden.

Herkunft der Daten

Der Prozess der Nachverfolgung des Ursprungs und der Geschichte von Daten während ihres gesamten Lebenszyklus, z. B. wie die Daten generiert, übertragen und gespeichert wurden.

betroffene Person

Eine Person, deren Daten gesammelt und verarbeitet werden.

Data Warehouse

Ein Datenverwaltungssystem, das Business Intelligence wie Analysen unterstützt. Data Warehouses enthalten in der Regel große Mengen historischer Daten und werden in der Regel für Abfragen und Analysen verwendet.

Datenbankdefinitionssprache (DDL)

Anweisungen oder Befehle zum Erstellen oder Ändern der Struktur von Tabellen und Objekten in einer Datenbank.

Datenbankmanipulationssprache (DML)

Anweisungen oder Befehle zum Ändern (Einfügen, Aktualisieren und Löschen) von Informationen in einer Datenbank.

DDL

Siehe Datenbankdefinitionssprache.

Deep-Ensemble

Mehrere Deep-Learning-Modelle zur Vorhersage kombinieren. Sie können Deep-Ensembles verwenden, um eine genauere Vorhersage zu erhalten oder um die Unsicherheit von Vorhersagen abzuschätzen.

Deep Learning

Ein ML-Teilbereich, der mehrere Schichten künstlicher neuronaler Netzwerke verwendet, um die Zuordnung zwischen Eingabedaten und Zielvariablen von Interesse zu ermitteln.

defense-in-depth

Ein Ansatz zur Informationssicherheit, bei dem eine Reihe von Sicherheitsmechanismen und -kontrollen sorgfältig in einem Computernetzwerk verteilt werden, um die Vertraulichkeit, Integrität und Verfügbarkeit des Netzwerks und der darin enthaltenen Daten zu schützen. Wenn Sie diese Strategie anwenden AWS, fügen Sie mehrere Steuerelemente auf verschiedenen Ebenen der AWS Organizations Struktur hinzu, um die Ressourcen zu schützen. Ein defense-in-depth Ansatz könnte beispielsweise Multi-Faktor-Authentifizierung, Netzwerksegmentierung und Verschlüsselung kombinieren.

delegierter Administrator

In AWS Organizations kann ein kompatibler Dienst ein AWS Mitgliedskonto registrieren, um die Konten der Organisation und die Berechtigungen für diesen Dienst zu verwalten. Dieses Konto wird als delegierter Administrator für diesen Service bezeichnet. Weitere Informationen und eine Liste kompatibler Services finden Sie unter Services, die mit AWS Organizations funktionieren in der AWS Organizations -Dokumentation.

Bereitstellung

Der Prozess, bei dem eine Anwendung, neue Feature oder Codekorrekturen in der Zielumgebung verfügbar gemacht werden. Die Bereitstellung umfasst das Implementieren von Änderungen an einer Codebasis und das anschließende Erstellen und Ausführen dieser Codebasis in den Anwendungsumgebungen.

Entwicklungsumgebung

Siehe Umgebung.

Detektivische Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, ein Ereignis zu erkennen, zu protokollieren und zu warnen, nachdem ein Ereignis eingetreten ist. Diese Kontrollen stellen eine zweite Verteidigungslinie dar und warnen Sie vor Sicherheitsereignissen, bei denen die vorhandenen präventiven Kontrollen umgangen wurden. Weitere Informationen finden Sie unter Detektivische Kontrolle in Implementierung von Sicherheitskontrollen in AWS.

Abbildung des Wertstroms in der Entwicklung (DVSM)

Ein Prozess zur Identifizierung und Priorisierung von Einschränkungen, die sich negativ auf Geschwindigkeit und Qualität im Lebenszyklus der Softwareentwicklung auswirken. DVSM erweitert den Prozess der Wertstromanalyse, der ursprünglich für Lean-Manufacturing-Praktiken

konzipiert wurde. Es konzentriert sich auf die Schritte und Teams, die erforderlich sind, um durch den Softwareentwicklungsprozess Mehrwert zu schaffen und zu steigern.

digitaler Zwilling

Eine virtuelle Darstellung eines realen Systems, z. B. eines Gebäudes, einer Fabrik, einer Industrieanlage oder einer Produktionslinie. Digitale Zwillinge unterstützen vorausschauende Wartung, Fernüberwachung und Produktionsoptimierung.

Maßtabelle

In einem Sternschema eine kleinere Tabelle, die Datenattribute zu quantitativen Daten in einer Faktentabelle enthält. Bei Attributen von Dimensionstabellen handelt es sich in der Regel um Textfelder oder diskrete Zahlen, die sich wie Text verhalten. Diese Attribute werden häufig zum Einschränken von Abfragen, zum Filtern und zur Kennzeichnung von Ergebnismengen verwendet.

Katastrophe

Ein Ereignis, das verhindert, dass ein Workload oder ein System seine Geschäftsziele an seinem primären Einsatzort erfüllt. Diese Ereignisse können Naturkatastrophen, technische Ausfälle oder das Ergebnis menschlichen Handelns sein, z. B. unbeabsichtigte Fehlkonfigurationen oder ein Malware-Angriff.

Disaster Recovery (DR)

Die Strategie und der Prozess, die Sie verwenden, um Ausfallzeiten und Datenverluste aufgrund einer Katastrophe zu minimieren. Weitere Informationen finden Sie unter Disaster Recovery von Workloads unter AWS: Wiederherstellung in der Cloud im AWS Well-Architected Framework.

DML

Siehe Sprache zur Datenbankmanipulation.

Domainorientiertes Design

Ein Ansatz zur Entwicklung eines komplexen Softwaresystems, bei dem seine Komponenten mit sich entwickelnden Domains oder Kerngeschäftszielen verknüpft werden, denen jede Komponente dient. Dieses Konzept wurde von Eric Evans in seinem Buch Domaingesteuertes Design: Bewältigen der Komplexität im Herzen der Software (Boston: Addison-Wesley Professional, 2003) vorgestellt. Informationen darüber, wie Sie domaingesteuertes Design mit dem Strangler-Fig-Muster verwenden können, finden Sie unter Schrittweises Modernisieren älterer Microsoft ASP.NET (ASMX)-Webservices mithilfe von Containern und Amazon API Gateway.

DR

Siehe Disaster Recovery.

Erkennung von Driften

Verfolgung von Abweichungen von einer Basiskonfiguration. Sie können es beispielsweise verwenden, AWS CloudFormation um Abweichungen bei den Systemressourcen zu erkennen, oder Sie können AWS Control Tower damit Änderungen in Ihrer landing zone erkennen, die sich auf die Einhaltung von Governance-Anforderungen auswirken könnten.

DVSM

Siehe Abbildung des Wertstroms in der Entwicklung.

E

EDA

Siehe explorative Datenanalyse.

EDI

Siehe elektronischer Datenaustausch.

Edge-Computing

Die Technologie, die die Rechenleistung für intelligente Geräte an den Rändern eines IoT-Netzwerks erhöht. Im Vergleich zu Cloud Computing kann Edge Computing die Kommunikationslatenz reduzieren und die Reaktionszeit verbessern.

elektronischer Datenaustausch (EDI)

Der automatisierte Austausch von Geschäftsdokumenten zwischen Organisationen. Weitere Informationen finden Sie unter Was ist elektronischer Datenaustausch.

Verschlüsselung

Ein Rechenprozess, der Klartextdaten, die für Menschen lesbar sind, in Chiffretext umwandelt.

Verschlüsselungsschlüssel

Eine kryptografische Zeichenfolge aus zufälligen Bits, die von einem Verschlüsselungsalgorithmus generiert wird. Schlüssel können unterschiedlich lang sein, und jeder Schlüssel ist so konzipiert, dass er unvorhersehbar und einzigartig ist.

Endianismus

Die Reihenfolge, in der Bytes im Computerspeicher gespeichert werden. Big-Endian-Systeme speichern das höchstwertige Byte zuerst. Little-Endian-Systeme speichern das niedrigwertigste Byte zuerst.

Endpunkt

Siehe Service-Endpunkt.

Endpunkt-Services

Ein Service, den Sie in einer Virtual Private Cloud (VPC) hosten können, um ihn mit anderen Benutzern zu teilen. Sie können einen Endpunktdienst mit anderen AWS-Konten oder AWS Identity and Access Management (IAM AWS PrivateLink -) Prinzipalen erstellen und diesen Berechtigungen gewähren. Diese Konten oder Prinzipale können sich privat mit Ihrem Endpunktservice verbinden, indem sie Schnittstellen-VPC-Endpunkte erstellen. Weitere Informationen finden Sie unter Einen Endpunkt-Service erstellen in der Amazon Virtual Private Cloud (Amazon VPC)-Dokumentation.

Unternehmensressourcenplanung (ERP)

Ein System, das wichtige Geschäftsprozesse (wie Buchhaltung, MES und Projektmanagement) für ein Unternehmen automatisiert und verwaltet.

Envelope-Verschlüsselung

Der Prozess der Verschlüsselung eines Verschlüsselungsschlüssels mit einem anderen Verschlüsselungsschlüssel. Weitere Informationen finden Sie unter Envelope-Verschlüsselung in der AWS Key Management Service (AWS KMS) -Dokumentation.

Umgebung

Eine Instance einer laufenden Anwendung. Die folgenden Arten von Umgebungen sind beim Cloud-Computing üblich:

- Entwicklungsumgebung – Eine Instance einer laufenden Anwendung, die nur dem Kernteam zur Verfügung steht, das für die Wartung der Anwendung verantwortlich ist. Entwicklungsumgebungen werden verwendet, um Änderungen zu testen, bevor sie in höhere Umgebungen übertragen werden. Diese Art von Umgebung wird manchmal als Testumgebung bezeichnet.
- Niedrigere Umgebungen – Alle Entwicklungsumgebungen für eine Anwendung, z. B. solche, die für erste Builds und Tests verwendet wurden.

- Produktionsumgebung – Eine Instance einer laufenden Anwendung, auf die Endbenutzer zugreifen können. In einer CI/CD-Pipeline ist die Produktionsumgebung die letzte Bereitstellungsumgebung.
- Höhere Umgebungen – Alle Umgebungen, auf die auch andere Benutzer als das Kernentwicklungsteam zugreifen können. Dies kann eine Produktionsumgebung, Vorproduktionsumgebungen und Umgebungen für Benutzerakzeptanztests umfassen.

Epics

In der agilen Methodik sind dies funktionale Kategorien, die Ihnen helfen, Ihre Arbeit zu organisieren und zu priorisieren. Epics bieten eine allgemeine Beschreibung der Anforderungen und Implementierungsaufgaben. Zu den Sicherheitsthemen AWS von CAF gehören beispielsweise Identitäts- und Zugriffsmanagement, Detektivkontrollen, Infrastruktursicherheit, Datenschutz und Reaktion auf Vorfälle. Weitere Informationen zu Epics in der AWS - Migrationsstrategie finden Sie im Leitfaden zur Programm-Implementierung.

ERP

Siehe Enterprise Resource Planning.

Explorative Datenanalyse (EDA)

Der Prozess der Analyse eines Datensatzes, um seine Hauptmerkmale zu verstehen. Sie sammeln oder aggregieren Daten und führen dann erste Untersuchungen durch, um Muster zu finden, Anomalien zu erkennen und Annahmen zu überprüfen. EDA wird durchgeführt, indem zusammenfassende Statistiken berechnet und Datenvisualisierungen erstellt werden.

F

Faktentabelle

Die zentrale Tabelle in einem Sternschema. Sie speichert quantitative Daten über den Geschäftsbetrieb. In der Regel enthält eine Faktentabelle zwei Arten von Spalten: Spalten, die Kennzahlen enthalten, und Spalten, die einen Fremdschlüssel für eine Dimensionstabelle enthalten.

schnell scheitern

Eine Philosophie, die häufige und inkrementelle Tests verwendet, um den Entwicklungslebenszyklus zu verkürzen. Dies ist ein wichtiger Bestandteil eines agilen Ansatzes.

Grenze zur Fehlerisolierung

Dabei handelt es sich um eine Grenze AWS Cloud, z. B. eine Availability Zone AWS-Region, eine Steuerungsebene oder eine Datenebene, die die Auswirkungen eines Fehlers begrenzt und die Widerstandsfähigkeit von Workloads verbessert. Weitere Informationen finden Sie unter Grenzen zur AWS Fehlerisolierung.

Feature-Zweig

Siehe Zweig.

Features

Die Eingabedaten, die Sie verwenden, um eine Vorhersage zu treffen. In einem Fertigungskontext könnten Feature beispielsweise Bilder sein, die regelmäßig von der Fertigungslinie aus aufgenommen werden.

Bedeutung der Feature

Wie wichtig ein Feature für die Vorhersagen eines Modells ist. Dies wird in der Regel als numerischer Wert ausgedrückt, der mit verschiedenen Techniken wie Shapley Additive Explanations (SHAP) und integrierten Gradienten berechnet werden kann. Weitere Informationen finden Sie unter Interpretierbarkeit von Modellen für maschinelles Lernen mit AWS.

Featuretransformation

Daten für den ML-Prozess optimieren, einschließlich der Anreicherung von Daten mit zusätzlichen Quellen, der Skalierung von Werten oder der Extraktion mehrerer Informationssätze aus einem einzigen Datenfeld. Das ermöglicht dem ML-Modell, von den Daten profitieren. Wenn Sie beispielsweise das Datum „27.05.2021 00:15:37“ in „2021“, „Mai“, „Donnerstag“ und „15“ aufschlüsseln, können Sie dem Lernalgorithmus helfen, nuancierte Muster zu erlernen, die mit verschiedenen Datenkomponenten verknüpft sind.

Eingabeaufforderung mit wenigen Klicks

Bereitstellung einer kleinen Anzahl von Beispielen, die die Aufgabe und das gewünschte Ergebnis veranschaulichen, bevor das LLM aufgefordert wird, eine ähnliche Aufgabe auszuführen. Bei dieser Technik handelt es sich um eine Anwendung des kontextbezogenen Lernens, bei der Modelle anhand von Beispielen (Aufnahmen) lernen, die in Eingabeaufforderungen eingebettet sind. Bei Aufgaben, die spezifische Formatierungs-, Argumentations- oder Fachkenntnisse erfordern, kann die Eingabeaufforderung mit wenigen Handgriffen effektiv sein. Siehe auch Zero-Shot Prompting.

FGAC

Siehe detaillierte Zugriffskontrolle.

Feinkörnige Zugriffskontrolle (FGAC)

Die Verwendung mehrerer Bedingungen, um eine Zugriffsanfrage zuzulassen oder abzulehnen.

Flash-Cut-Migration

Eine Datenbankmigrationsmethode, bei der eine kontinuierliche Datenreplikation durch Erfassung von Änderungsdaten verwendet wird, um Daten in kürzester Zeit zu migrieren, anstatt einen schrittweisen Ansatz zu verwenden. Ziel ist es, Ausfallzeiten auf ein Minimum zu beschränken.

FM

Siehe Fundamentmodell.

Fundamentmodell (FM)

Ein großes neuronales Deep-Learning-Netzwerk, das mit riesigen Datensätzen generalisierter und unbeschrifteter Daten trainiert wurde. FMs sind in der Lage, eine Vielzahl allgemeiner Aufgaben zu erfüllen, z. B. Sprache zu verstehen, Text und Bilder zu generieren und Konversationen in natürlicher Sprache zu führen. Weitere Informationen finden Sie unter Was sind Foundation-Modelle.

G

generative KI

Eine Untergruppe von KI-Modellen, die mit großen Datenmengen trainiert wurden und mit einer einfachen Textaufforderung neue Inhalte und Artefakte wie Bilder, Videos, Text und Audio erstellen können. Weitere Informationen finden Sie unter Was ist Generative KI.

Geoblocking

Siehe geografische Einschränkungen.

Geografische Einschränkungen (Geoblocking)

Bei Amazon eine Option CloudFront, um zu verhindern, dass Benutzer in bestimmten Ländern auf Inhaltsverteilungen zugreifen. Sie können eine Zulassungsliste oder eine Sperrliste verwenden,

um zugelassene und gesperrte Länder anzugeben. Weitere Informationen finden Sie in der Dokumentation unter Beschränkung der geografischen Verteilung Ihrer Inhalte. CloudFront

Gitflow-Workflow

Ein Ansatz, bei dem niedrigere und höhere Umgebungen unterschiedliche Zweige in einem Quellcode-Repository verwenden. Der Gitflow-Workflow gilt als veraltet, und der Trunk-basierte Workflow ist der moderne, bevorzugte Ansatz.

goldenes Bild

Ein Snapshot eines Systems oder einer Software, der als Vorlage für die Bereitstellung neuer Instanzen dieses Systems oder dieser Software verwendet wird. In der Fertigung kann ein Golden Image beispielsweise zur Bereitstellung von Software auf mehreren Geräten verwendet werden und trägt zur Verbesserung der Geschwindigkeit, Skalierbarkeit und Produktivität bei der Geräteherstellung bei.

Greenfield-Strategie

Das Fehlen vorhandener Infrastruktur in einer neuen Umgebung. Bei der Einführung einer Neuausrichtung einer Systemarchitektur können Sie alle neuen Technologien ohne Einschränkung der Kompatibilität mit der vorhandenen Infrastruktur auswählen, auch bekannt als Brownfield. Wenn Sie die bestehende Infrastruktur erweitern, könnten Sie Brownfield- und Greenfield-Strategien mischen.

Integritätsschutz

Eine allgemeine Regel, die dazu beiträgt, Ressourcen, Richtlinien und die Einhaltung von Vorschriften in allen Unternehmenseinheiten zu regeln (OUs). Präventiver Integritätsschutz setzt Richtlinien durch, um die Einhaltung von Standards zu gewährleisten. Sie werden mithilfe von Service-Kontrollrichtlinien und IAM-Berechtigungsgrenzen implementiert. Detektivischer Integritätsschutz erkennt Richtlinienverstöße und Compliance-Probleme und generiert Warnmeldungen zur Abhilfe. Sie werden mithilfe von AWS Config, AWS Security Hub, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector und benutzerdefinierten AWS Lambda Prüfungen implementiert.

H

HEKTAR

Siehe Hochverfügbarkeit.

Heterogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank in eine Zieldatenbank, die eine andere Datenbank-Engine verwendet (z. B. Oracle zu Amazon Aurora). Eine heterogene Migration ist in der Regel Teil einer Neuarchitektur, und die Konvertierung des Schemas kann eine komplexe Aufgabe sein. AWS bietet AWS SCT, welches bei Schemakonvertierungen hilft.

hohe Verfügbarkeit (HA)

Die Fähigkeit eines Workloads, im Falle von Herausforderungen oder Katastrophen kontinuierlich und ohne Eingreifen zu arbeiten. HA-Systeme sind so konzipiert, dass sie automatisch ein Failover durchführen, gleichbleibend hohe Leistung bieten und unterschiedliche Lasten und Ausfälle mit minimalen Leistungseinbußen bewältigen.

historische Modernisierung

Ein Ansatz zur Modernisierung und Aufrüstung von Betriebstechnologiesystemen (OT), um den Bedürfnissen der Fertigungsindustrie besser gerecht zu werden. Ein Historian ist eine Art von Datenbank, die verwendet wird, um Daten aus verschiedenen Quellen in einer Fabrik zu sammeln und zu speichern.

Holdout-Daten

Ein Teil historischer, beschrifteter Daten, der aus einem Datensatz zurückgehalten wird, der zum Trainieren eines Modells für maschinelles Lernen verwendet wird. Sie können Holdout-Daten verwenden, um die Modellleistung zu bewerten, indem Sie die Modellvorhersagen mit den Holdout-Daten vergleichen.

Homogene Datenbankmigration

Migrieren Sie Ihre Quelldatenbank zu einer Zieldatenbank, die dieselbe Datenbank-Engine verwendet (z. B. Microsoft SQL Server zu Amazon RDS für SQL Server). Eine homogene Migration ist in der Regel Teil eines Hostwechsels oder eines Plattformwechsels. Sie können native Datenbankserviceprogramme verwenden, um das Schema zu migrieren.

heiße Daten

Daten, auf die häufig zugegriffen wird, z. B. Echtzeitdaten oder aktuelle Translationsdaten. Für diese Daten ist in der Regel eine leistungsstarke Speicherebene oder -klasse erforderlich, um schnelle Abfrageantworten zu ermöglichen.

Hotfix

Eine dringende Lösung für ein kritisches Problem in einer Produktionsumgebung. Aufgrund seiner Dringlichkeit wird ein Hotfix normalerweise außerhalb des typischen DevOps Release-Workflows erstellt.

Hypercare-Phase

Unmittelbar nach dem Cutover, der Zeitraum, in dem ein Migrationsteam die migrierten Anwendungen in der Cloud verwaltet und überwacht, um etwaige Probleme zu beheben. In der Regel dauert dieser Zeitraum 1–4 Tage. Am Ende der Hypercare-Phase überträgt das Migrationsteam in der Regel die Verantwortung für die Anwendungen an das Cloud-Betriebsteam.

I

IaC

Sehen Sie sich Infrastruktur als Code an.

Identitätsbasierte Richtlinie

Eine Richtlinie, die einem oder mehreren IAM-Prinzipalen zugeordnet ist und deren Berechtigungen innerhalb der AWS Cloud Umgebung definiert.

Leerlaufanwendung

Eine Anwendung mit einer durchschnittlichen CPU- und Arbeitsspeicherauslastung zwischen 5 und 20 Prozent über einen Zeitraum von 90 Tagen. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen oder sie On-Premises beizubehalten.

IIoT

Siehe Industrielles Internet der Dinge.

unveränderliche Infrastruktur

Ein Modell, das eine neue Infrastruktur für Produktionsworkloads bereitstellt, anstatt die bestehende Infrastruktur zu aktualisieren, zu patchen oder zu modifizieren. Unveränderliche Infrastrukturen sind von Natur aus konsistenter, zuverlässiger und vorhersehbarer als veränderliche Infrastrukturen. Weitere Informationen finden Sie in der Best Practice Deploy using immutable infrastructure im AWS Well-Architected Framework.

Eingehende (ingress) VPC

In einer Architektur AWS mit mehreren Konten ist dies eine VPC, die Netzwerkverbindungen von außerhalb einer Anwendung akzeptiert, überprüft und weiterleitet. Die AWS Security Reference Architecture empfiehlt, Ihr Netzwerkkonto mit eingehendem und ausgehendem Datenverkehr und Inspektion einzurichten, VPCs um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet im weiteren Sinne zu schützen.

Inkrementelle Migration

Eine Cutover-Strategie, bei der Sie Ihre Anwendung in kleinen Teilen migrieren, anstatt eine einziges vollständiges Cutover durchzuführen. Beispielsweise könnten Sie zunächst nur einige Microservices oder Benutzer auf das neue System umstellen. Nachdem Sie sich vergewissert haben, dass alles ordnungsgemäß funktioniert, können Sie weitere Microservices oder Benutzer schrittweise verschieben, bis Sie Ihr Legacy-System außer Betrieb nehmen können. Diese Strategie reduziert die mit großen Migrationen verbundenen Risiken.

Industrie 4.0

Ein Begriff, der 2016 von Klaus Schwab eingeführt wurde und sich auf die Modernisierung von Fertigungsprozessen durch Fortschritte in den Bereichen Konnektivität, Echtzeitdaten, Automatisierung, Analytik und KI/ML bezieht.

Infrastruktur

Alle Ressourcen und Komponenten, die in der Umgebung einer Anwendung enthalten sind.

Infrastructure as Code (IaC)

Der Prozess der Bereitstellung und Verwaltung der Infrastruktur einer Anwendung mithilfe einer Reihe von Konfigurationsdateien. IaC soll Ihnen helfen, das Infrastrukturmanagement zu zentralisieren, Ressourcen zu standardisieren und schnell zu skalieren, sodass neue Umgebungen wiederholbar, zuverlässig und konsistent sind.

industrielles Internet der Dinge (T) IIo

Einsatz von mit dem Internet verbundenen Sensoren und Geräten in Industriesektoren wie Fertigung, Energie, Automobilindustrie, Gesundheitswesen, Biowissenschaften und Landwirtschaft. Weitere Informationen finden Sie unter Aufbau einer digitalen Transformationsstrategie für das industrielle Internet der Dinge (IIoT).

Inspektions-VPC

In einer Architektur AWS mit mehreren Konten eine zentralisierte VPC, die Inspektionen des Netzwerkverkehrs zwischen VPCs (in demselben oder unterschiedlichen AWS-Regionen), dem Internet und lokalen Netzwerken verwaltet. In der AWS Security Reference Architecture wird empfohlen, Ihr Netzwerkkonto mit eingehendem und ausgehendem Datenverkehr sowie Inspektionen einzurichten, VPCs um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet im weiteren Sinne zu schützen.

Internet of Things (IoT)

Das Netzwerk verbundener physischer Objekte mit eingebetteten Sensoren oder Prozessoren, das über das Internet oder über ein lokales Kommunikationsnetzwerk mit anderen Geräten und Systemen kommuniziert. Weitere Informationen finden Sie unter Was ist IoT?

Interpretierbarkeit

Ein Merkmal eines Modells für Machine Learning, das beschreibt, inwieweit ein Mensch verstehen kann, wie die Vorhersagen des Modells von seinen Eingaben abhängen. Weitere Informationen finden Sie unter Interpretierbarkeit von Modellen für maschinelles Lernen mit. AWS

IoT

Siehe Internet der Dinge.

IT information library (ITIL, IT-Informationsbibliothek)

Eine Reihe von bewährten Methoden für die Bereitstellung von IT-Services und die Abstimmung dieser Services auf die Geschäftsanforderungen. ITIL bietet die Grundlage für ITSM.

T service management (ITSM, IT-Servicemanagement)

Aktivitäten im Zusammenhang mit der Gestaltung, Implementierung, Verwaltung und Unterstützung von IT-Services für eine Organisation. Informationen zur Integration von Cloud-Vorgängen mit ITSM-Tools finden Sie im Leitfaden zur Betriebsintegration.

BIS

Weitere Informationen finden Sie in der IT-Informationsbibliothek.

ITSM

Siehe IT-Servicemanagement.

L

Labelbasierte Zugangskontrolle (LBAC)

: Eine Implementierung der Mandatory Access Control (MAC), bei der den Benutzern und den Daten selbst jeweils explizit ein Sicherheitslabelwert zugewiesen wird. Die Schnittmenge zwischen der Benutzersicherheitsbeschriftung und der Datensicherheitsbeschriftung bestimmt, welche Zeilen und Spalten für den Benutzer sichtbar sind.

Landing Zone

: Eine landing zone ist eine gut strukturierte AWS Umgebung mit mehreren Konten, die skalierbar und sicher ist. Dies ist ein Ausgangspunkt, von dem aus Ihre Organisationen Workloads und Anwendungen schnell und mit Vertrauen in ihre Sicherheits- und Infrastrukturumgebung starten und bereitstellen können. Weitere Informationen zu Landing Zones finden Sie unter Einrichtung einer sicheren und skalierbaren AWS -Umgebung mit mehreren Konten..

großes Sprachmodell (LLM)

: Ein Deep-Learning-KI-Modell, das anhand einer riesigen Datenmenge vorab trainiert wurde. Ein LLM kann mehrere Aufgaben ausführen, z. B. Fragen beantworten, Dokumente zusammenfassen, Text in andere Sprachen übersetzen und Sätze vervollständigen. Weitere Informationen finden Sie unter Was sind. LLMs

Große Migration

: Eine Migration von 300 oder mehr Servern.

SCHWARZ

: Weitere Informationen finden Sie unter Label-basierte Zugriffskontrolle.

Geringste Berechtigung

: Die bewährte Sicherheitsmethode, bei der nur die für die Durchführung einer Aufgabe erforderlichen Mindestberechtigungen erteilt werden. Weitere Informationen finden Sie unter Geringste Berechtigungen anwenden in der IAM-Dokumentation.

Lift and Shift

: Siehe 7 Rs.

Little-Endian-System

: Ein System, welches das niedrigwertigste Byte zuerst speichert. Siehe auch Endianness.

LLM

Siehe großes Sprachmodell.

Niedrigere Umgebungen

Siehe Umgebung.

M

Machine Learning (ML)

Eine Art künstlicher Intelligenz, die Algorithmen und Techniken zur Mustererkennung und zum Lernen verwendet. ML analysiert aufgezeichnete Daten, wie z. B. Daten aus dem Internet der Dinge (IoT), und lernt daraus, um ein statistisches Modell auf der Grundlage von Mustern zu erstellen. Weitere Informationen finden Sie unter Machine Learning.

Hauptzweig

Siehe Filiale.

Malware

Software, die entwickelt wurde, um die Computersicherheit oder den Datenschutz zu gefährden. Malware kann Computersysteme stören, vertrauliche Informationen durchsickern lassen oder sich unbefugten Zugriff verschaffen. Beispiele für Malware sind Viren, Würmer, Ransomware, Trojaner, Spyware und Keylogger.

verwaltete Dienste

AWS-Services für die die Infrastrukturebene, das Betriebssystem und die Plattformen AWS betrieben werden, und Sie greifen auf die Endgeräte zu, um Daten zu speichern und abzurufen. Amazon Simple Storage Service (Amazon S3) und Amazon DynamoDB sind Beispiele für Managed Services. Diese werden auch als abstrakte Dienste bezeichnet.

Manufacturing Execution System (MES)

Ein Softwaresystem zur Nachverfolgung, Überwachung, Dokumentation und Steuerung von Produktionsprozessen, bei denen Rohstoffe in der Fertigung zu fertigen Produkten umgewandelt werden.

MAP

Siehe Migration Acceleration Program.

Mechanismus

Ein vollständiger Prozess, bei dem Sie ein Tool erstellen, die Akzeptanz des Tools vorantreiben und anschließend die Ergebnisse überprüfen, um Anpassungen vorzunehmen. Ein Mechanismus ist ein Zyklus, der sich im Laufe seiner Tätigkeit selbst verstärkt und verbessert. Weitere Informationen finden Sie unter Aufbau von Mechanismen im AWS Well-Architected Framework.

Mitgliedskonto

Alle AWS-Konten außer dem Verwaltungskonto, die Teil einer Organisation in sind. AWS Organizations Ein Konto kann jeweils nur einer Organisation angehören.

DURCHEINANDER

Siehe Manufacturing Execution System.

Message Queuing-Telemetrietransport (MQTT)

Ein leichtes machine-to-machine (M2M) -Kommunikationsprotokoll, das auf dem Publish/Subscribe-Muster für IoT-Geräte mit beschränkten Ressourcen basiert.

Microservice

Ein kleiner, unabhängiger Dienst, der über genau definierte Kanäle kommuniziert APIs und in der Regel kleinen, eigenständigen Teams gehört. Ein Versicherungssystem kann beispielsweise Microservices beinhalten, die Geschäftsfunktionen wie Vertrieb oder Marketing oder Subdomains wie Einkauf, Schadenersatz oder Analytik zugeordnet sind. Zu den Vorteilen von Microservices gehören Agilität, flexible Skalierung, einfache Bereitstellung, wiederverwendbarer Code und Ausfallsicherheit. Weitere Informationen finden Sie unter Integration von Microservices mithilfe serverloser Dienste. AWS

Microservices-Architekturen

Ein Ansatz zur Erstellung einer Anwendung mit unabhängigen Komponenten, die jeden Anwendungsprozess als Microservice ausführen. Diese Microservices kommunizieren mithilfe von Lightweight über eine klar definierte Schnittstelle. APIs Jeder Microservice in dieser Architektur kann aktualisiert, bereitgestellt und skaliert werden, um den Bedarf an bestimmten Funktionen einer Anwendung zu decken. Weitere Informationen finden Sie unter Implementierung von Microservices auf. AWS

Migration Acceleration Program (MAP)

Ein AWS Programm, das Beratung, Unterstützung, Schulungen und Services bietet, um Unternehmen dabei zu unterstützen, eine solide betriebliche Grundlage für die Umstellung auf

die Cloud zu schaffen und die anfänglichen Kosten von Migrationen auszugleichen. MAP umfasst eine Migrationsmethode für die methodische Durchführung von Legacy-Migrationen sowie eine Reihe von Tools zur Automatisierung und Beschleunigung gängiger Migrationsszenarien.

Migration in großem Maßstab

Der Prozess, bei dem der Großteil des Anwendungsportfolios in Wellen in die Cloud verlagert wird, wobei in jeder Welle mehr Anwendungen schneller migriert werden. In dieser Phase werden die bewährten Verfahren und Erkenntnisse aus den früheren Phasen zur Implementierung einer Migrationsfabrik von Teams, Tools und Prozessen zur Optimierung der Migration von Workloads durch Automatisierung und agile Bereitstellung verwendet. Dies ist die dritte Phase der AWS - Migrationsstrategie.

Migrationsfabrik

Funktionsübergreifende Teams, die die Migration von Workloads durch automatisierte, agile Ansätze optimieren. Zu den Teams in der Migrationsabteilung gehören in der Regel Betriebsabläufe, Geschäftsanalysten und Eigentümer, Migrationsingenieure, Entwickler und DevOps Experten, die in Sprints arbeiten. Zwischen 20 und 50 Prozent eines Unternehmensanwendungsportfolios bestehen aus sich wiederholenden Mustern, die durch einen Fabrik-Ansatz optimiert werden können. Weitere Informationen finden Sie in Diskussion über Migrationsfabriken und den Leitfaden zur Cloud-Migration-Fabrik in diesem Inhaltssatz.

Migrationsmetadaten

Die Informationen über die Anwendung und den Server, die für den Abschluss der Migration benötigt werden. Für jedes Migrationsmuster ist ein anderer Satz von Migrationsmetadaten erforderlich. Beispiele für Migrationsmetadaten sind das Zielsubnetz, die Sicherheitsgruppe und AWS das Konto.

Migrationsmuster

Eine wiederholbare Migrationsaufgabe, in der die Migrationsstrategie, das Migrationsziel und die verwendete Migrationsanwendung oder der verwendete Migrationsservice detailliert beschrieben werden. Beispiel: Rehost-Migration zu Amazon EC2 mit AWS Application Migration Service.

Migration Portfolio Assessment (MPA)

Ein Online-Tool, das Informationen zur Validierung des Geschäftsszenarios für die Migration auf das bereitstellt. AWS Cloud MPA bietet eine detaillierte Portfoliobewertung (richtige Servergröße, Preisgestaltung, Gesamtbetriebskostenanalyse, Migrationskostenanalyse) sowie Migrationsplanung (Anwendungsdatenanalyse und Datenerfassung, Anwendungsgruppierung,

Migrationspriorisierung und Wellenplanung). Das MPA-Tool (Anmeldung erforderlich) steht allen AWS Beratern und APN-Partnerberatern kostenlos zur Verfügung.

Migration Readiness Assessment (MRA)

Der Prozess, bei dem mithilfe des AWS CAF Erkenntnisse über den Cloud-Bereitschaftsstatus eines Unternehmens gewonnen, Stärken und Schwächen identifiziert und ein Aktionsplan zur Schließung festgestellter Lücken erstellt wird. Weitere Informationen finden Sie im Benutzerhandbuch für Migration Readiness. MRA ist die erste Phase der AWS - Migrationsstrategie.

Migrationsstrategie

Der Ansatz, der verwendet wurde, um einen Workload auf den AWS Cloud zu migrieren. Weitere Informationen finden Sie im Eintrag 7 Rs in diesem Glossar und unter Mobilisieren Sie Ihr Unternehmen, um umfangreiche Migrationen zu beschleunigen.

ML

Siehe maschinelles Lernen.

Modernisierung

Umwandlung einer veralteten (veralteten oder monolithischen) Anwendung und ihrer Infrastruktur in ein agiles, elastisches und hochverfügbares System in der Cloud, um Kosten zu senken, die Effizienz zu steigern und Innovationen zu nutzen. Weitere Informationen finden Sie unter Strategie zur Modernisierung von Anwendungen in der AWS Cloud.

Bewertung der Modernisierungsfähigkeit

Eine Bewertung, anhand derer festgestellt werden kann, ob die Anwendungen einer Organisation für die Modernisierung bereit sind, Vorteile, Risiken und Abhängigkeiten identifiziert und ermittelt wird, wie gut die Organisation den zukünftigen Status dieser Anwendungen unterstützen kann. Das Ergebnis der Bewertung ist eine Vorlage der Zielarchitektur, eine Roadmap, in der die Entwicklungsphasen und Meilensteine des Modernisierungsprozesses detailliert beschrieben werden, sowie ein Aktionsplan zur Behebung festgestellter Lücken. Weitere Informationen finden Sie unter Evaluierung der Modernisierungsbereitschaft von Anwendungen in der AWS Cloud.

Monolithische Anwendungen (Monolithen)

Anwendungen, die als ein einziger Service mit eng gekoppelten Prozessen ausgeführt werden. Monolithische Anwendungen haben verschiedene Nachteile. Wenn ein Anwendungs-Feature stark nachgefragt wird, muss die gesamte Architektur skaliert werden. Das Hinzufügen oder

Verbessern der Feature einer monolithischen Anwendung wird ebenfalls komplexer, wenn die Codebasis wächst. Um diese Probleme zu beheben, können Sie eine Microservices-Architektur verwenden. Weitere Informationen finden Sie unter Zerlegen von Monolithen in Microservices.

MPA

Siehe Bewertung des Migrationsportfolios.

MQTT

Siehe Message Queuing-Telemetrietransport.

Mehrklassen-Klassifizierung

Ein Prozess, der dabei hilft, Vorhersagen für mehrere Klassen zu generieren (wobei eines von mehr als zwei Ergebnissen vorhergesagt wird). Ein ML-Modell könnte beispielsweise fragen: „Ist dieses Produkt ein Buch, ein Auto oder ein Telefon?“ oder „Welche Kategorie von Produkten ist für diesen Kunden am interessantesten?“

veränderbare Infrastruktur

Ein Modell, das die bestehende Infrastruktur für Produktionsworkloads aktualisiert und modifiziert. Für eine verbesserte Konsistenz, Zuverlässigkeit und Vorhersagbarkeit empfiehlt das AWS Well-Architected Framework die Verwendung einer unveränderlichen Infrastruktur als bewährte Methode.

O

OAC

Siehe Origin Access Control.

OAI

Siehe Zugriffsidentität von Origin.

COM

Siehe organisatorisches Change-Management.

Offline-Migration

Eine Migrationsmethode, bei der der Quell-Workload während des Migrationsprozesses heruntergefahren wird. Diese Methode ist mit längeren Ausfallzeiten verbunden und wird in der Regel für kleine, unkritische Workloads verwendet.

OI

Siehe Betriebsintegration.

OLA

Siehe Vereinbarung auf operativer Ebene.

Online-Migration

Eine Migrationsmethode, bei der der Quell-Workload auf das Zielsystem kopiert wird, ohne offline genommen zu werden. Anwendungen, die mit dem Workload verbunden sind, können während der Migration weiterhin funktionieren. Diese Methode beinhaltet keine bis minimale Ausfallzeit und wird in der Regel für kritische Produktionsworkloads verwendet.

OPC-UA

Siehe Open Process Communications — Unified Architecture.

Offene Prozesskommunikation — Einheitliche Architektur (OPC-UA)

Ein machine-to-machine (M2M) -Kommunikationsprotokoll für die industrielle Automatisierung. OPC-UA bietet einen Interoperabilitätsstandard mit Datenverschlüsselungs-, Authentifizierungs- und Autorisierungsschemata.

Vereinbarung auf Betriebsebene (OLA)

Eine Vereinbarung, in der klargestellt wird, welche funktionalen IT-Gruppen sich gegenseitig versprechen zu liefern, um ein Service Level Agreement (SLA) zu unterstützen.

Überprüfung der Betriebsbereitschaft (ORR)

Eine Checkliste mit Fragen und zugehörigen bewährten Methoden, die Ihnen helfen, Vorfälle und mögliche Ausfälle zu verstehen, zu bewerten, zu verhindern oder deren Umfang zu reduzieren. Weitere Informationen finden Sie unter Operational Readiness Reviews (ORR) im AWS Well-Architected Framework.

Betriebstechnologie (OT)

Hardware- und Softwaresysteme, die mit der physischen Umgebung zusammenarbeiten, um industrielle Abläufe, Ausrüstung und Infrastruktur zu steuern. In der Fertigung ist die Integration von OT- und Informationstechnologie (IT) -Systemen ein zentraler Schwerpunkt der Industrie 4.0-Transformationen.

Betriebsintegration (OI)

Der Prozess der Modernisierung von Abläufen in der Cloud, der Bereitschaftsplanung, Automatisierung und Integration umfasst. Weitere Informationen finden Sie im Leitfaden zur Betriebsintegration.

Organisationspfad

Ein Pfad, der von erstellt wird und in AWS CloudTrail dem alle Ereignisse für alle AWS-Konten in einer Organisation protokolliert werden. AWS Organizations Diese Spur wird in jedem AWS-Konto , der Teil der Organisation ist, erstellt und verfolgt die Aktivität in jedem Konto. Weitere Informationen finden Sie in der CloudTrail Dokumentation unter Einen Trail für eine Organisation erstellen.

Organisatorisches Veränderungsmanagement (OCM)

Ein Framework für das Management wichtiger, disruptiver Geschäftstransformationen aus Sicht der Mitarbeiter, der Kultur und der Führung. OCM hilft Organisationen dabei, sich auf neue Systeme und Strategien vorzubereiten und auf diese umzustellen, indem es die Akzeptanz von Veränderungen beschleunigt, Übergangsprobleme angeht und kulturelle und organisatorische Veränderungen vorantreibt. In der AWS Migrationsstrategie wird dieses Framework aufgrund der Geschwindigkeit des Wandels, der bei Projekten zur Cloud-Einführung erforderlich ist, als Mitarbeiterbeschleunigung bezeichnet. Weitere Informationen finden Sie im OCM-Handbuch.

Ursprungszugriffskontrolle (OAC)

In CloudFront, eine erweiterte Option zur Zugriffsbeschränkung, um Ihre Amazon Simple Storage Service (Amazon S3) -Inhalte zu sichern. OAC unterstützt alle S3-Buckets insgesamt AWS-Regionen, serverseitige Verschlüsselung mit AWS KMS (SSE-KMS) sowie dynamische `PUT` und `DELETE` Anfragen an den S3-Bucket.

Ursprungszugriffsidentität (OAI)

In CloudFront, eine Option zur Zugriffsbeschränkung, um Ihre Amazon S3 S3-Inhalte zu sichern. Wenn Sie OAI verwenden, CloudFront erstellt es einen Principal, mit dem sich Amazon S3 authentifizieren kann. Authentifizierte Principals können nur über eine bestimmte Distribution auf Inhalte in einem S3-Bucket zugreifen. CloudFront Siehe auch OAC, das eine detailliertere und verbesserte Zugriffskontrolle bietet.

ORR

Weitere Informationen finden Sie unter Überprüfung der Betriebsbereitschaft.

NICHT

Siehe Betriebstechnologie.

Ausgehende (egress) VPC

In einer Architektur AWS mit mehreren Konten eine VPC, die Netzwerkverbindungen verarbeitet, die von einer Anwendung aus initiiert werden. Die AWS Security Reference Architecture empfiehlt die Einrichtung Ihres Netzwerkkontos mit eingehendem und ausgehendem Datenverkehr sowie Inspektion, VPCs um die bidirektionale Schnittstelle zwischen Ihrer Anwendung und dem Internet im weiteren Sinne zu schützen.

P

Berechtigungsgrenze

Eine IAM-Verwaltungsrichtlinie, die den IAM-Prinzipalen zugeordnet ist, um die maximalen Berechtigungen festzulegen, die der Benutzer oder die Rolle haben kann. Weitere Informationen finden Sie unter Berechtigungsgrenzen für IAM-Entitys in der IAM-Dokumentation.

persönlich identifizierbare Informationen (PII)

Informationen, die, wenn sie direkt betrachtet oder mit anderen verwandten Daten kombiniert werden, verwendet werden können, um vernünftige Rückschlüsse auf die Identität einer Person zu ziehen. Beispiele für personenbezogene Daten sind Namen, Adressen und Kontaktinformationen.

Personenbezogene Daten

Siehe persönlich identifizierbare Informationen.

Playbook

Eine Reihe vordefinierter Schritte, die die mit Migrationen verbundenen Aufgaben erfassen, z. B. die Bereitstellung zentraler Betriebsfunktionen in der Cloud. Ein Playbook kann die Form von Skripten, automatisierten Runbooks oder einer Zusammenfassung der Prozesse oder Schritte annehmen, die für den Betrieb Ihrer modernisierten Umgebung erforderlich sind.

PLC

Siehe programmierbare Logiksteuerung.

PLM

Siehe Produktlebenszyklusmanagement.

policy

Ein Objekt, das Berechtigungen definieren (siehe identitätsbasierte Richtlinie), Zugriffsbedingungen spezifizieren (siehe ressourcenbasierte Richtlinie) oder die maximalen Berechtigungen für alle Konten in einer Organisation definieren kann AWS Organizations (siehe Dienststeuerungsrichtlinie).

Polyglotte Beharrlichkeit

Unabhängige Auswahl der Datenspeichertechnologie eines Microservices auf der Grundlage von Datenzugriffsmustern und anderen Anforderungen. Wenn Ihre Microservices über dieselbe Datenspeichertechnologie verfügen, kann dies zu Implementierungsproblemen oder zu Leistungseinbußen führen. Microservices lassen sich leichter implementieren und erzielen eine bessere Leistung und Skalierbarkeit, wenn sie den Datenspeicher verwenden, der ihren Anforderungen am besten entspricht. Weitere Informationen finden Sie unter Datenpersistenz in Microservices aktivieren.

Portfoliobewertung

Ein Prozess, bei dem das Anwendungsportfolio ermittelt, analysiert und priorisiert wird, um die Migration zu planen. Weitere Informationen finden Sie in Bewerten der Migrationsbereitschaft.

predicate

Eine Abfragebedingung, die `true` oder zurückgibt`false`, was üblicherweise in einer Klausel vorkommt. `WHERE`

Prädikat Pushdown

Eine Technik zur Optimierung von Datenbankabfragen, bei der die Daten in der Abfrage vor der Übertragung gefiltert werden. Dadurch wird die Datenmenge reduziert, die aus der relationalen Datenbank abgerufen und verarbeitet werden muss, und die Abfrageleistung wird verbessert.

Präventive Kontrolle

Eine Sicherheitskontrolle, die verhindern soll, dass ein Ereignis eintritt. Diese Kontrollen stellen eine erste Verteidigungslinie dar, um unbefugten Zugriff oder unerwünschte Änderungen an Ihrem Netzwerk zu verhindern. Weitere Informationen finden Sie unter Präventive Kontrolle in Implementierung von Sicherheitskontrollen in AWS.

Prinzipal

Eine Entität AWS , die Aktionen ausführen und auf Ressourcen zugreifen kann. Bei dieser Entität handelt es sich in der Regel um einen Root-Benutzer für eine AWS-Konto, eine IAM-Rolle oder einen Benutzer. Weitere Informationen finden Sie unter Prinzipal in Rollenbegriffe und -konzepte in der IAM-Dokumentation.

Datenschutz von Natur aus

Ein systemtechnischer Ansatz, der den Datenschutz während des gesamten Entwicklungsprozesses berücksichtigt.

Privat gehostete Zonen

Ein Container, der Informationen darüber enthält, wie Amazon Route 53 auf DNS-Abfragen für eine Domain und deren Subdomains innerhalb einer oder mehrerer VPCs Domains antworten soll. Weitere Informationen finden Sie unter Arbeiten mit privat gehosteten Zonen in der Route-53-Dokumentation.

proaktive Steuerung

Eine Sicherheitskontrolle, die den Einsatz nicht richtlinienkonformer Ressourcen verhindern soll. Diese Steuerelemente scannen Ressourcen, bevor sie bereitgestellt werden. Wenn die Ressource nicht mit der Steuerung konform ist, wird sie nicht bereitgestellt. Weitere Informationen finden Sie im Referenzhandbuch zu Kontrollen in der AWS Control Tower Dokumentation und unter Proaktive Kontrollen unter Implementierung von Sicherheitskontrollen am AWS.

Produktlebenszyklusmanagement (PLM)

Das Management von Daten und Prozessen für ein Produkt während seines gesamten Lebenszyklus, vom Design, der Entwicklung und Markteinführung über Wachstum und Reife bis hin zur Markteinführung und Markteinführung.

Produktionsumgebung

Siehe Umgebung.

Speicherprogrammierbare Steuerung (SPS)

In der Fertigung ein äußerst zuverlässiger, anpassungsfähiger Computer, der Maschinen überwacht und Fertigungsprozesse automatisiert.

schnelle Verkettung

Verwendung der Ausgabe einer LLM-Eingabeaufforderung als Eingabe für die nächste Aufforderung, um bessere Antworten zu generieren. Diese Technik wird verwendet, um eine komplexe Aufgabe in Unteraufgaben zu unterteilen oder um eine vorläufige Antwort iterativ zu verfeinern oder zu erweitern. Sie trägt dazu bei, die Genauigkeit und Relevanz der Antworten eines Modells zu verbessern und ermöglicht detailliertere, personalisierte Ergebnisse.

Pseudonymisierung

Der Prozess, bei dem persönliche Identifikatoren in einem Datensatz durch Platzhalterwerte ersetzt werden. Pseudonymisierung kann zum Schutz der Privatsphäre beitragen. Pseudonymisierte Daten gelten weiterhin als personenbezogene Daten.

publish/subscribe (pub/sub)

Ein Muster, das asynchrone Kommunikation zwischen Microservices ermöglicht, um die Skalierbarkeit und Reaktionsfähigkeit zu verbessern. In einem auf Microservices basierenden MES kann ein Microservice beispielsweise Ereignismeldungen in einem Kanal veröffentlichen, den andere Microservices abonnieren können. Das System kann neue Microservices hinzufügen, ohne den Veröffentlichungsservice zu ändern.

Q

Abfrageplan

Eine Reihe von Schritten, wie Anweisungen, die für den Zugriff auf die Daten in einem relationalen SQL-Datenbanksystem verwendet werden.

Abfrageplanregression

Wenn ein Datenbankserviceoptimierer einen weniger optimalen Plan wählt als vor einer bestimmten Änderung der Datenbankumgebung. Dies kann durch Änderungen an Statistiken, Beschränkungen, Umgebungseinstellungen, Abfrageparameter-Bindungen und Aktualisierungen der Datenbank-Engine verursacht werden.

R

RACI-Matrix

Siehe verantwortlich, rechenschaftspflichtig, konsultiert, informiert (RACI).

LAPPEN

Siehe Erweiterte Generierung beim Abrufen.

Ransomware

Eine bösartige Software, die entwickelt wurde, um den Zugriff auf ein Computersystem oder Daten zu blockieren, bis eine Zahlung erfolgt ist.

RASCI-Matrix

Siehe verantwortlich, rechenschaftspflichtig, konsultiert, informiert (RACI).

RCAC

Siehe Zugriffskontrolle für Zeilen und Spalten.

Read Replica

Eine Kopie einer Datenbank, die nur für Lesezwecke verwendet wird. Sie können Abfragen an das Lesereplikat weiterleiten, um die Belastung auf Ihrer Primärdatenbank zu reduzieren.

neu strukturieren

Siehe 7 Rs.

Recovery Point Objective (RPO)

Die maximal zulässige Zeitspanne seit dem letzten Datenwiederherstellungspunkt. Damit wird festgelegt, was als akzeptabler Datenverlust zwischen dem letzten Wiederherstellungspunkt und der Serviceunterbrechung gilt.

Wiederherstellungszeitziel (RTO)

Die maximal zulässige Verzögerung zwischen der Betriebsunterbrechung und der Wiederherstellung des Dienstes.

Refaktorierung

Siehe 7 Rs.

Region

Eine Sammlung von AWS Ressourcen in einem geografischen Gebiet. Jeder AWS-Region ist isoliert und unabhängig von den anderen, um Fehlertoleranz, Stabilität und Belastbarkeit zu gewährleisten. Weitere Informationen finden Sie unter Geben Sie an, was AWS-Regionen Ihr Konto verwenden kann.

Regression

Eine ML-Technik, die einen numerischen Wert vorhersagt. Zum Beispiel, um das Problem „Zu welchem Preis wird dieses Haus verkauft werden?“ zu lösen Ein ML-Modell könnte ein lineares Regressionsmodell verwenden, um den Verkaufspreis eines Hauses auf der Grundlage bekannter Fakten über das Haus (z. B. die Quadratmeterzahl) vorherzusagen.

rehosten

Siehe 7 Rs.

Veröffentlichung

In einem Bereitstellungsprozess der Akt der Förderung von Änderungen an einer Produktionsumgebung.

umziehen

Siehe 7 Rs.

neue Plattform

Siehe 7 Rs.

Rückkauf

Siehe 7 Rs.

Ausfallsicherheit

Die Fähigkeit einer Anwendung, Störungen zu widerstehen oder sich von ihnen zu erholen. Hochverfügbarkeit und Notfallwiederherstellung sind häufig Überlegungen bei der Planung der Ausfallsicherheit in der. AWS Cloud Weitere Informationen finden Sie unter AWS Cloud Resilienz.

Ressourcenbasierte Richtlinie

Eine mit einer Ressource verknüpfte Richtlinie, z. B. ein Amazon-S3-Bucket, ein Endpunkt oder ein Verschlüsselungsschlüssel. Diese Art von Richtlinie legt fest, welchen Prinzipalen der Zugriff gewährt wird, welche Aktionen unterstützt werden und welche anderen Bedingungen erfüllt sein müssen.

RACI-Matrix (verantwortlich, rechenschaftspflichtig, konsultiert, informiert)

Eine Matrix, die die Rollen und Verantwortlichkeiten aller an Migrationsaktivitäten und Cloud-Operationen beteiligten Parteien definiert. Der Matrixname leitet sich von den in der Matrix definierten Zuständigkeitstypen ab: verantwortlich (R), rechenschaftspflichtig (A), konsultiert (C) und informiert (I). Der Unterstützungstyp (S) ist optional. Wenn Sie Unterstützung einbeziehen, wird die Matrix als RASCI-Matrix bezeichnet, und wenn Sie sie ausschließen, wird sie als RACI-Matrix bezeichnet.

Reaktive Kontrolle

Eine Sicherheitskontrolle, die darauf ausgelegt ist, die Behebung unerwünschter Ereignisse oder Abweichungen von Ihren Sicherheitsstandards voranzutreiben. Weitere Informationen finden Sie unter Reaktive Kontrolle in Implementieren von Sicherheitskontrollen in AWS.

Beibehaltung

Siehe 7 Rs.

zurückziehen

Siehe 7 Rs.

Retrieval Augmented Generation (RAG)

Eine generative KI-Technologie, bei der ein LLM auf eine maßgebliche Datenquelle verweist, die sich außerhalb seiner Trainingsdatenquellen befindet, bevor eine Antwort generiert wird. Ein RAG-Modell könnte beispielsweise eine semantische Suche in der Wissensdatenbank oder in benutzerdefinierten Daten einer Organisation durchführen. Weitere Informationen finden Sie unter Was ist RAG.

Drehung

Der Vorgang, bei dem ein Geheimnis regelmäßig aktualisiert wird, um es einem Angreifer zu erschweren, auf die Anmeldeinformationen zuzugreifen.

Zugriffskontrolle für Zeilen und Spalten (RCAC)

Die Verwendung einfacher, flexibler SQL-Ausdrücke mit definierten Zugriffsregeln. RCAC besteht aus Zeilenberechtigungen und Spaltenmasken.

RPO

Siehe Recovery Point Objective.

RTO

Siehe Ziel der Wiederherstellungszeit.

Runbook

Eine Reihe manueller oder automatisierter Verfahren, die zur Ausführung einer bestimmten Aufgabe erforderlich sind. Diese sind in der Regel darauf ausgelegt, sich wiederholende Operationen oder Verfahren mit hohen Fehlerquoten zu rationalisieren.

S

SAML 2.0

Ein offener Standard, den viele Identitätsanbieter (IdPs) verwenden. Diese Funktion ermöglicht föderiertes Single Sign-On (SSO), sodass sich Benutzer bei den API-Vorgängen anmelden AWS Management Console oder die AWS API-Operationen aufrufen können, ohne dass Sie einen Benutzer in IAM für alle in Ihrer Organisation erstellen müssen. Weitere Informationen zum SAML-2.0.-basierten Verbund finden Sie unter Über den SAML-2.0-basierten Verbund in der IAM-Dokumentation.

SCADA

Siehe Aufsichtskontrolle und Datenerfassung.

SCP

Siehe Richtlinie zur Dienstkontrolle.

Secret

Interne AWS Secrets Manager, vertrauliche oder eingeschränkte Informationen, wie z. B. ein Passwort oder Benutzeranmeldedaten, die Sie in verschlüsselter Form speichern. Es besteht aus dem geheimen Wert und seinen Metadaten. Der geheime Wert kann binär, eine einzelne Zeichenfolge oder mehrere Zeichenketten sein. Weitere Informationen finden Sie unter Was ist in einem Secrets Manager Manager-Geheimnis? in der Secrets Manager Manager-Dokumentation.

Sicherheit durch Design

Ein systemtechnischer Ansatz, der die Sicherheit während des gesamten Entwicklungsprozesses berücksichtigt.

Sicherheitskontrolle

Ein technischer oder administrativer Integritätsschutz, der die Fähigkeit eines Bedrohungsakteurs, eine Schwachstelle auszunutzen, verhindert, erkennt oder einschränkt. Es gibt vier Haupttypen von Sicherheitskontrollen: präventiv, detektiv, reaktionsschnell und proaktiv.

Härtung der Sicherheit

Der Prozess, bei dem die Angriffsfläche reduziert wird, um sie widerstandsfähiger gegen Angriffe zu machen. Dies kann Aktionen wie das Entfernen von Ressourcen, die nicht mehr benötigt werden, die Implementierung der bewährten Sicherheitsmethode der Gewährung geringster Berechtigungen oder die Deaktivierung unnötiger Feature in Konfigurationsdateien umfassen.

System zur Verwaltung von Sicherheitsinformationen und Ereignissen (security information and event management – SIEM)

Tools und Services, die Systeme für das Sicherheitsinformationsmanagement (SIM) und das Management von Sicherheitsereignissen (SEM) kombinieren. Ein SIEM-System sammelt, überwacht und analysiert Daten von Servern, Netzwerken, Geräten und anderen Quellen, um Bedrohungen und Sicherheitsverletzungen zu erkennen und Warnmeldungen zu generieren.

Automatisierung von Sicherheitsreaktionen

Eine vordefinierte und programmierte Aktion, die darauf ausgelegt ist, automatisch auf ein Sicherheitsereignis zu reagieren oder es zu beheben. Diese Automatisierungen dienen als detektive oder reaktionsschnelle Sicherheitskontrollen, die Sie bei der Implementierung bewährter AWS Sicherheitsmethoden unterstützen. Beispiele für automatisierte Antwortaktionen sind das Ändern einer VPC-Sicherheitsgruppe, das Patchen einer EC2 Amazon-Instance oder das Rotieren von Anmeldeinformationen.

Serverseitige Verschlüsselung

Verschlüsselung von Daten am Zielort durch denjenigen AWS-Service , der sie empfängt.

Service-Kontrollrichtlinie (SCP)

Eine Richtlinie, die eine zentrale Steuerung der Berechtigungen für alle Konten in einer Organisation in ermöglicht AWS Organizations. SCPs Definieren Sie Leitplanken oder legen Sie Grenzwerte für Aktionen fest, die ein Administrator an Benutzer oder Rollen delegieren kann. Sie können sie SCPs als Zulassungs- oder Ablehnungslisten verwenden, um festzulegen, welche Dienste oder Aktionen zulässig oder verboten sind. Weitere Informationen finden Sie in der AWS Organizations Dokumentation unter Richtlinien zur Dienststeuerung.

Service-Endpunkt

Die URL des Einstiegspunkts für einen AWS-Service. Sie können den Endpunkt verwenden, um programmgesteuert eine Verbindung zum Zielservice herzustellen. Weitere Informationen finden Sie unter AWS-Service -Endpunkte in der Allgemeine AWS-Referenz.

Service Level Agreement (SLA)

Eine Vereinbarung, in der klargestellt wird, was ein IT-Team seinen Kunden zu bieten verspricht, z. B. in Bezug auf Verfügbarkeit und Leistung der Services.

Service-Level-Indikator (SLI)

Eine Messung eines Leistungsaspekts eines Dienstes, z. B. seiner Fehlerrate, Verfügbarkeit oder Durchsatz.

Service-Level-Ziel (SLO)

Eine Zielkennzahl, die den Zustand eines Dienstes darstellt, gemessen anhand eines Service-Level-Indikators.

Modell der geteilten Verantwortung

Ein Modell, das die Verantwortung beschreibt, mit der Sie gemeinsam AWS für Cloud-Sicherheit und Compliance verantwortlich sind. AWS ist für die Sicherheit der Cloud verantwortlich, während Sie für die Sicherheit in der Cloud verantwortlich sind. Weitere Informationen finden Sie unter Modell der geteilten Verantwortung.

SIEM

Siehe Sicherheitsinformations- und Event-Management-System.

Single Point of Failure (SPOF)

Ein Fehler in einer einzelnen, kritischen Komponente einer Anwendung, der das System stören kann.

SLA

Siehe Service Level Agreement.

SLI

Siehe Service-Level-Indikator.

ALSO

Siehe Service-Level-Ziel.

split-and-seed Modell

Ein Muster für die Skalierung und Beschleunigung von Modernisierungsprojekten. Sobald neue Features und Produktversionen definiert werden, teilt sich das Kernteam auf, um neue Produktteams zu bilden. Dies trägt zur Skalierung der Fähigkeiten und Services Ihrer Organisation bei, verbessert die Produktivität der Entwickler und unterstützt schnelle Innovationen. Weitere Informationen finden Sie unter Schrittweiser Ansatz zur Modernisierung von Anwendungen in der AWS Cloud.

SPOTTEN

Siehe Single Point of Failure.

Sternschema

Eine Datenbank-Organisationsstruktur, die eine große Faktentabelle zum Speichern von Transaktions- oder Messdaten und eine oder mehrere kleinere dimensionale Tabellen zum Speichern von Datenattributen verwendet. Diese Struktur ist für die Verwendung in einem Data Warehouse oder für Business Intelligence-Zwecke konzipiert.

Strangler-Fig-Muster

Ein Ansatz zur Modernisierung monolithischer Systeme, bei dem die Systemfunktionen schrittweise umgeschrieben und ersetzt werden, bis das Legacy-System außer Betrieb genommen werden kann. Dieses Muster verwendet die Analogie einer Feigenrebe, die zu einem etablierten Baum heranwächst und schließlich ihren Wirt überwindet und ersetzt. Das Muster wurde eingeführt von Martin Fowler als Möglichkeit, Risiken beim Umschreiben monolithischer Systeme zu managen. Ein Beispiel für die Anwendung dieses Musters finden Sie unter Schrittweises Modernisieren älterer Microsoft ASP.NET (ASMX)-Webservices mithilfe von Containern und Amazon API Gateway.

Subnetz

Ein Bereich von IP-Adressen in Ihrer VPC. Ein Subnetz muss sich in einer einzigen Availability Zone befinden.

Aufsichtskontrolle und Datenerfassung (SCADA)

In der Fertigung ein System, das Hardware und Software zur Überwachung von Sachanlagen und Produktionsabläufen verwendet.

Symmetrische Verschlüsselung

Ein Verschlüsselungsalgorithmus, der denselben Schlüssel zum Verschlüsseln und Entschlüsseln der Daten verwendet.

synthetisches Testen

Testen eines Systems auf eine Weise, die Benutzerinteraktionen simuliert, um potenzielle Probleme zu erkennen oder die Leistung zu überwachen. Sie können Amazon CloudWatch Synthetics verwenden, um diese Tests zu erstellen.

Systemaufforderung

Eine Technik, mit der einem LLM Kontext, Anweisungen oder Richtlinien zur Verfügung gestellt werden, um sein Verhalten zu steuern. Systemaufforderungen helfen dabei, den Kontext festzulegen und Regeln für Interaktionen mit Benutzern festzulegen.

T

tags

Schlüssel-Wert-Paare, die als Metadaten für die Organisation Ihrer Ressourcen dienen. AWS Mit Tags können Sie Ressourcen verwalten, identifizieren, organisieren, suchen und filtern. Weitere Informationen finden Sie unter Markieren Ihrer AWS -Ressourcen.

Zielvariable

Der Wert, den Sie in überwachtem ML vorhersagen möchten. Dies wird auch als Ergebnisvariable bezeichnet. In einer Fertigungsumgebung könnte die Zielvariable beispielsweise ein Produktfehler sein.

Aufgabenliste

Ein Tool, das verwendet wird, um den Fortschritt anhand eines Runbooks zu verfolgen. Eine Aufgabenliste enthält eine Übersicht über das Runbook und eine Liste mit allgemeinen Aufgaben, die erledigt werden müssen. Für jede allgemeine Aufgabe werden der geschätzte Zeitaufwand, der Eigentümer und der Fortschritt angegeben.

Testumgebungen

Siehe Umgebung.

Training

Daten für Ihr ML-Modell bereitstellen, aus denen es lernen kann. Die Trainingsdaten müssen die richtige Antwort enthalten. Der Lernalgorithmus findet Muster in den Trainingsdaten, die die Attribute der Input-Daten dem Ziel (die Antwort, die Sie voraussagen möchten) zuordnen. Es gibt ein ML-Modell aus, das diese Muster erfasst. Sie können dann das ML-Modell verwenden, um Voraussagen für neue Daten zu erhalten, bei denen Sie das Ziel nicht kennen.

Transit-Gateway

Ein Netzwerk-Transit-Hub, über den Sie Ihre Netzwerke VPCs und Ihre lokalen Netzwerke miteinander verbinden können. Weitere Informationen finden Sie in der Dokumentation unter Was ist ein Transit-Gateway. AWS Transit Gateway

Stammbasierter Workflow

Ein Ansatz, bei dem Entwickler Feature lokal in einem Feature-Zweig erstellen und testen und diese Änderungen dann im Hauptzweig zusammenführen. Der Hauptzweig wird dann sequentiell für die Entwicklungs-, Vorproduktions- und Produktionsumgebungen erstellt.

Vertrauenswürdiger Zugriff

Gewährung von Berechtigungen für einen Dienst, den Sie angeben, um Aufgaben in Ihrer Organisation AWS Organizations und in deren Konten in Ihrem Namen auszuführen. Der vertrauenswürdige Service erstellt in jedem Konto eine mit dem Service verknüpfte Rolle, wenn diese Rolle benötigt wird, um Verwaltungsaufgaben für Sie auszuführen. Weitere Informationen finden Sie in der AWS Organizations Dokumentation unter Verwendung AWS Organizations mit anderen AWS Diensten.

Optimieren

Aspekte Ihres Trainingsprozesses ändern, um die Genauigkeit des ML-Modells zu verbessern. Sie können das ML-Modell z. B. trainieren, indem Sie einen Beschriftungssatz generieren, Beschriftungen hinzufügen und diese Schritte dann mehrmals unter verschiedenen Einstellungen wiederholen, um das Modell zu optimieren.

Zwei-Pizzen-Team

Ein kleines DevOps Team, das Sie mit zwei Pizzen ernähren können. Eine Teamgröße von zwei Pizzen gewährleistet die bestmögliche Gelegenheit zur Zusammenarbeit bei der Softwareentwicklung.

U

Unsicherheit

Ein Konzept, das sich auf ungenaue, unvollständige oder unbekannte Informationen bezieht, die die Zuverlässigkeit von prädiktiven ML-Modellen untergraben können. Es gibt zwei Arten von Unsicherheit: Epistemische Unsicherheit wird durch begrenzte, unvollständige Daten verursacht, wohingegen aleatorische Unsicherheit durch Rauschen und Randomisierung verursacht wird, die in den Daten liegt. Weitere Informationen finden Sie im Leitfaden Quantifizieren der Unsicherheit in Deep-Learning-Systemen.

undifferenzierte Aufgaben

Diese Arbeit wird auch als Schwerstarbeit bezeichnet. Dabei handelt es sich um Arbeiten, die zwar für die Erstellung und den Betrieb einer Anwendung erforderlich sind, aber dem Endbenutzer keinen direkten Mehrwert bieten oder keinen Wettbewerbsvorteil bieten. Beispiele für undifferenzierte Aufgaben sind Beschaffung, Wartung und Kapazitätsplanung.

höhere Umgebungen

Siehe Umgebung.

V

Vacuuming

Ein Vorgang zur Datenbankwartung, bei dem die Datenbank nach inkrementellen Aktualisierungen bereinigt wird, um Speicherplatz zurückzugewinnen und die Leistung zu verbessern.

Versionskontrolle

Prozesse und Tools zur Nachverfolgung von Änderungen, z. B. Änderungen am Quellcode in einem Repository.

VPC-Peering

Eine Verbindung zwischen zwei VPCs , die es Ihnen ermöglicht, den Verkehr mithilfe privater IP-Adressen weiterzuleiten. Weitere Informationen finden Sie unter Was ist VPC-Peering? in der Amazon-VPC-Dokumentation.

Schwachstelle

Ein Software- oder Hardwarefehler, der die Sicherheit des Systems beeinträchtigt.

W

Warmer Cache

Ein Puffer-Cache, der aktuelle, relevante Daten enthält, auf die häufig zugegriffen wird. Die Datenbank-Instance kann aus dem Puffer-Cache lesen, was schneller ist als das Lesen aus dem Hauptspeicher oder von der Festplatte.

warme Daten

Daten, auf die selten zugegriffen wird. Bei der Abfrage dieser Art von Daten sind mäßig langsame Abfragen in der Regel akzeptabel.

Fensterfunktion

Eine SQL-Funktion, die eine Berechnung für eine Gruppe von Zeilen durchführt, die sich in irgendeiner Weise auf den aktuellen Datensatz beziehen. Fensterfunktionen sind nützlich für die Verarbeitung von Aufgaben wie die Berechnung eines gleitenden Durchschnitts oder für den Zugriff auf den Wert von Zeilen auf der Grundlage der relativen Position der aktuellen Zeile.

Workload

Ein Workload ist eine Sammlung von Ressourcen und Code, die einen Unternehmenswert bietet, wie z. B. eine kundenorientierte Anwendung oder ein Backend-Prozess.

Workstream

Funktionsgruppen in einem Migrationsprojekt, die für eine bestimmte Reihe von Aufgaben verantwortlich sind. Jeder Workstream ist unabhängig, unterstützt aber die anderen Workstreams im Projekt. Der Portfolio-Workstream ist beispielsweise für die Priorisierung von Anwendungen, die Wellenplanung und die Erfassung von Migrationsmetadaten verantwortlich. Der Portfolio-Workstream liefert diese Komponenten an den Migrations-Workstream, der dann die Server und Anwendungen migriert.

WURM

Sehen Sie einmal schreiben, viele lesen.

WQF

Siehe AWS Workload-Qualifizierungsrahmen.

einmal schreiben, viele lesen (WORM)

Ein Speichermodell, das Daten ein einziges Mal schreibt und verhindert, dass die Daten gelöscht oder geändert werden. Autorisierte Benutzer können die Daten so oft wie nötig lesen, aber sie können sie nicht ändern. Diese Datenspeicherinfrastruktur gilt als unveränderlich.

Z

Zero-Day-Exploit

Ein Angriff, in der Regel Malware, der eine Zero-Day-Sicherheitslücke ausnutzt.

Zero-Day-Sicherheitslücke

Ein unfehlbarer Fehler oder eine Sicherheitslücke in einem Produktionssystem. Bedrohungsakteure können diese Art von Sicherheitslücke nutzen, um das System anzugreifen. Entwickler werden aufgrund des Angriffs häufig auf die Sicherheitsanfälligkeit aufmerksam.

Zero-Shot-Aufforderung

Bereitstellung von Anweisungen für die Ausführung einer Aufgabe an einen LLM, jedoch ohne Beispiele (Schnappschüsse), die ihm als Orientierungshilfe dienen könnten. Der LLM muss sein vortrainiertes Wissen einsetzen, um die Aufgabe zu bewältigen. Die Effektivität von Zero-Shot Prompting hängt von der Komplexität der Aufgabe und der Qualität der Aufforderung ab. Siehe auch Few-Shot-Prompting.

Zombie-Anwendung

Eine Anwendung, deren durchschnittliche CPU- und Arbeitsspeichernutzung unter 5 Prozent liegt. In einem Migrationsprojekt ist es üblich, diese Anwendungen außer Betrieb zu nehmen.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.