Unable to locate subtitle

AWS Well-Architected Framework

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

AWS Well-Architected Framework: ***Unable to locate subtitle***

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

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Überblick und Einführung ... 1
- Einführung ... 1
- Definitionen ... 2
- Architektur-Überlegungen ... 5
- Allgemeine Designprinzipien ... 7

Die Säulen des Framework ... 9
- Operative Exzellenz ... 9
  - Designprinzipien ... 10
  - Definition ... 11
  - Bewährte Methoden ... 12
  - Ressourcen ... 23
- Sicherheit ... 23
  - Designprinzipien ... 24
  - Definition ... 25
  - Bewährte Methoden ... 25
  - Ressourcen ... 33
- Zuverlässigkeit ... 33
  - Designprinzipien ... 34
  - Definition ... 35
  - Bewährte Methoden ... 35
  - Ressourcen ... 41
- Leistungseffizienz ... 42
  - Designprinzipien ... 42
  - Definition ... 43
  - Bewährte Methoden ... 44
  - Ressourcen ... 49
- Kostenoptimierung ... 50
  - Designprinzipien ... 50
  - Definition ... 51
  - Bewährte Methoden ... 52
  - Ressourcen ... 59
- Nachhaltigkeit ... 59
  - Designprinzipien ... 59
  - Definition ... 61

Bewährte Methoden ... 61
Die Überprüfung ... 70
Fazit ... 73
Mitwirkende ... 74
Weitere Informationen ... 75
Dokumentversionen ... 76
Anhang: Fragen und bewährte Methoden ... 80
Operational Excellence ... 80
Organisation ... 80
Vorbereitung ... 143
Betrieb ... 219
Weiterentwicklung ... 264
Sicherheit ... 285
Sicherheitsgrundlagen ... 285
Identity and Access Management ... 313
Erkennung ... 374
Schutz der Infrastruktur ... 390
Datenschutz ... 419
Vorfallsreaktion ... 456
Anwendungssicherheit ... 481
Zuverlässigkeit ... 503
Grundlagen ... 504
Workload-Architektur ... 547
Änderungsverwaltung ... 599
Fehlerverwaltung ... 644
Leistungseffizienz ... 755
Auswahl der Architektur ... 756
Computer und Hardware ... 772
Datenverwaltung ... 791
Networking und Bereitstellung von Inhalten ... 817
Prozess und Kultur ... 850
Kostenoptimierung ... 868
Praxis für Cloud-Finanzmanagement ... 868
Ausgabenerkennung und Nutzungsbewusstsein ... 895
Kostengünstige Ressourcen ... 944
Verwaltung von Nachfrage und Bereitstellung von Ressourcen ... 989

Optimierung im Laufe der Zeit 1003
Nachhaltigkeit 1013
Auswahl von Regionen erläutert 1013
Ausrichtung am Bedarf 1015
Software und Architektur 1032
Daten 1045
Hardware und Services 1066
Prozess und Kultur 1077
Hinweise 1086

AWS Well-Architected Framework

Veröffentlichungsdatum: 27. Juni 2024 (Dokumentversionen)

Das AWS-Well-Architected-Framework unterstützt Sie dabei, die Vor- und Nachteile der Entscheidungen nachzuvollziehen, die Sie beim Aufbau von Systemen in AWS treffen. Das Framework hilft Ihnen, bewährte Architekturmethoden für die Entwicklung und den Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Systeme in der Cloud zu ermitteln.

Einführung

Das AWS-Well-Architected-Framework unterstützt Sie dabei, die Vor- und Nachteile der Entscheidungen nachzuvollziehen, die Sie beim Aufbau von Systemen in AWS treffen. Das Framework hilft Ihnen, bewährte Architekturmethoden für den Entwurf und Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Workloads in der AWS Cloud zu ermitteln. Es bietet Ihnen die Möglichkeit, Ihre Architekturen konsistent auf die Einhaltung bewährter Methoden zu prüfen und Verbesserungspotenzial zu identifizieren. Die Überprüfung einer Architektur ist kein Audit. Vielmehr ist es eine konstruktive Konversation, in der es um architektonische Entscheidungen geht. Wir sind davon überzeugt, dass eine durchdachte Systemarchitektur maßgeblich zu Ihrem künftigen geschäftlichen Erfolg beiträgt.

AWS Solutions Architects entwerfen seit vielen Jahren Lösungen für unterschiedlichste Branchen und Anwendungsfälle. Wir waren am Design und an der Überprüfung Tausender Kundenarchitekturen auf AWS beteiligt. Daher kennen wir die bewährten Methoden und Kernstrategien für erfolgreiche Systemarchitekturen in der Cloud.

Das AWS-Well-Architected-Framework dokumentiert grundlegende Fragen, die Ihnen dabei helfen zu klären, ob eine Architektur einwandfrei mit bewährten Methoden für die Cloud vereinbar ist. Über das Framework erhalten Sie eine einheitliche Herangehensweise zur Bewertung der Eigenschaften, die Sie von modernen Cloud-basierten Systemen erwarten, sowie Vorschläge zur Realisierung dieser Eigenschaften. AWS entwickelt sich ständig weiter, und auch wir lernen durch die Arbeit mit unseren Kunden ständig dazu. Mit diesem wachsenden Wissen können wir immer noch genauer definieren, wodurch sich eine gute architektonische Struktur auszeichnet.

Dieses Framework richtet sich an Technologiefachleute, z. B. Chief Technology Officers (CTO), Architekten, Entwickler und Operations-Mitarbeiter. Die darin enthaltenen bewährten Methoden und Strategien für AWS kommen bei der Gestaltung und Nutzung von Cloud-Workloads zum Einsatz. Die

Links verweisen auf weitere Implementierungsdetails und Architekturmodelle. Weitere Informationen finden Sie auf der Homepage von AWS-Well-Architected-Homepage.

AWS bietet auch eine kostenfreie Prüfung Ihrer Workloads an. Das AWS Well-Architected Tool (AWS WA Tool) ist ein Service in der Cloud, der einen einheitlichen Prozess zum Überprüfen und Messen Ihrer Architektur mit AWS Well-Architected Framework bietet. Vom AWS WA Tool erhalten Sie Empfehlungen, wie Sie Ihre Workloads zuverlässiger, sicherer, effizienter und kostengünstiger machen können.

Um Ihnen das Arbeiten nach bewährten Methoden zu erleichtern, haben wir AWS Well-Architected Labsentwickelt. Der Code und die Dokumentation von Labs erlauben Ihnen, eigene Erfahrungen mit der Implementierung bewährter Methoden zu sammeln. Außerdem haben wir uns mit ausgewählten Partnern aus dem AWS Partner Network (APN) zusammengetan, die Mitglieder des AWS Well-Architected-Partnerprogramms sind.. Diese AWS-Partner sind bestens mit AWS vertraut und können Sie beim Überprüfen und Verbessern Ihrer Workloads unterstützen.

Definitionen

Die Experten von AWS unterstützen Kunden tagtäglich beim Entwerfen von Systemarchitekturen, die ihnen eine optimale Nutzung bewährter Methoden in der Cloud ermöglichen. Während wir zusammen mit Ihnen die Architektur entwerfen, wägen wir die Anforderungen ab und treffen die richtigen Kompromisse. Wenn Sie die Systeme dann in Live-Umgebungen bereitstellen, beobachten wir, wie gut diese Systeme laufen und welche Auswirkungen die Kompromisse haben.

Unsere bisherigen Erkenntnisse sind die Grundlage von AWS Well-Architected Framework. Das Framework enthält einheitlich zusammengestellte bewährte Methoden, mit denen Kunden und Partner Architekturen bewerten. Anhand verschiedener Fragen können sie beurteilen, wie gut eine Architektur auf die bewährten Methoden von AWS ausgerichtet ist.

Das AWS-Well-Architected-Framework basiert auf sechs Säulen: operative Exzellenz, Sicherheit, Zuverlässigkeit, Leistungseffizienz, Kostenoptimierung und Nachhaltigkeit.

Tabelle 1. Die Säulen des AWS Well-Architected Framework

Name	Beschreibung
Operative Exzellenz	Die Fähigkeit, die Entwicklung zu unterstützen und Workloads effektiv auszuführen, Einblicke in die Betriebsabläufe zu erhalten

Name	Beschreibung
	und für geschäftlichen Mehrwert unterstüt zende Prozesse und Verfahren fortlaufend zu verbessern.
Sicherheit	In der Säule der Sicherheit wird beschrieben, wie Sie Cloud-Technologien nutzen, um Daten, Systeme und Komponenten so zu schützen, dass sich Ihre Sicherheitslage verbessert.
Zuverlässigkeit	Die Säule „Zuverlässigkeit“ umfasst die Fähigkeit eines Workloads, die beabsicht igte Funktion erwartungsgemäß korrekt und konsistent auszuführen. Dies umfasst die Möglichkeit, den Workload während des gesamten Lebenszyklus zu betreiben und zu testen. Dieses Dokument bietet umfassend e Informationen mit Best Practices für die Implementierung zuverlässiger Workloads in AWS.
Leistungseffizienz	Die Fähigkeit, Rechenressourcen effizient entsprechend den Systemanforderungen zu nutzen und diese Effizienz aufrechtzuerhalten , während sich die Nachfrage ändert und die Technologie weiterentwickelt.
Kostenoptimierung	Die Fähigkeit, Systeme so auszuführen, dass sie geschäftlichen Wert bei geringstmöglichen Kosten liefern.

Name	Beschreibung
Nachhaltigkeit	Die Fähigkeit, die Auswirkungen auf die Nachhaltigkeit kontinuierlich zu verbesser n, indem der Energieverbrauch reduziert und die Effizienz aller Komponenten eines Workloads erhöht wird, indem der Nutzen der bereitgestellten Ressourcen maximiert und die insgesamt erforderlichen Ressourcen minimiert werden.

In Zusammenhang mit dem AWS-Well-Architected-Framework verwenden wir diese Bezeichnungen:

- A Komponente besteht aus dem Code, der Konfiguration und den AWS-Ressourcen, die für eine Anforderung bereitgestellt werden. Eine Komponente ist häufig die Einheit technischen Eigentums und von anderen Komponenten losgelöst.
- Der Begriff Workload wird für zusammengehörige Komponenten, die geschäftlichen Mehrwert darstellen, verwendet. Ein Workload ist in vielen Fällen der Detaillierungsgrad, von dem Führungskräfte aus Wirtschaft und Technik häufig sprechen.
- Wir betrachten Architektur als das Zusammenwirken von Komponenten in einem Workload. Wie Komponenten kommunizieren und interagieren, ist häufig der Schwerpunkt von Architekturdiagrammen.
- Meilensteine markieren wichtige Änderungen im Laufe der Entwicklung einer Architektur im Produktlebenszyklus (Entwurf, Implementierung, Tests, Inbetriebnahme und Produktionsbetrieb).
- Innerhalb einer Organisation ist das Technologieportfolio die für den Geschäftsbetrieb erforderliche Sammlung an Workloads.
- Das Grad des Aufwands bezeichnet die Zeitspanne, den Aufwand und die Komplexität, die für die Implementierung einer Aufgabe benötigt werden. Jede Organisation muss die Größe und das Fachwissen des Teams sowie die Komplexität des Workloads berücksichtigen, um den Grad des Aufwands für die Organisation richtig einzuordnen.
  - Hoch: Die Arbeit dauert möglicherweise mehrere Wochen oder Monate. Sie könnte in mehrere Abschnitte, Veröffentlichungen und Aufgaben aufgeteilt werden.
  - Mittel: Die Arbeit dauert möglicherweise mehrere Tage oder Wochen. Sie könnte in mehrere Veröffentlichungen und Aufgaben aufgeteilt werden.

- Niedrig: Die Arbeit dauert möglicherweise mehrere Stunden oder Tage. Sie könnte in mehrere Aufgaben aufgeteilt werden.

Beim Entwerfen von Workloads stellen Sie eine Kosten-Nutzen-Abwägung zwischen Säulen abhängig von Ihrem Geschäftskontext an. Diese Geschäftsentscheidungen können Ihre technischen Prioritäten beeinflussen. In Entwicklungsumgebungen könnten Sie im Hinblick auf eine Verbesserung der Nachhaltigkeitswirkung und eine Verringerung der Kosten zulasten der Zuverlässigkeit optimieren. Bei unternehmenskritischen Lösungen könnten Sie dagegen die Zuverlässigkeit optimieren und dafür höhere Kosten und stärkere Auswirkungen auf die Nachhaltigkeit in Kauf nehmen. Bei E-Commerce-Lösungen kann sich die Leistung auf die Einnahmen und die Kauflust der Kunden auswirken. Sicherheit und operative Exzellenz haben in der Regel keine Wechselwirkung mit den anderen Säulen.

Architektur-Überlegungen

In On-Premises-Umgebungen setzen Kunden oft ein zentrales Technologiearchitektur-Team ein. Dies dient als Überlagerung für Produkt- oder Feature-Teams, um zu verifizieren, dass diese nach bewährten Methoden arbeiten. Technologiearchitektur-Teams setzen sich üblicherweise aus Fachleuten mit unterschiedlichen Aufgabengebieten zusammen, z. B.: Technical Architect (Infrastruktur), Solutions Architect (Software), Data Architect, Networking Architect und Security Architect. Diese Teams arbeiten oft nach dem TOGAF-Modell oder dem Zachman Framework – als Teil eines Kompetenzbereichs für Enterprise-Architektur.

Bei AWS werden die Fähigkeiten lieber auf einzelne Teams verteilt, als sie in einem Zentralteam zu konzentrieren. Wenn die Entscheidungsbefugnis auf mehrere Teams verteilt wird, geht das mit Risiken einher. So muss beispielsweise bestätigt sein, dass die Teams internen Standards gerecht werden. Um diese Risiken aufzufangen, verwenden wir zwei Methoden. Zum einen arbeiten wir mit Praktiken (Vorgehensweisen, Prozessen, Standards und gemeinhin anerkannte Normen), die darauf abzielen, jedes Team mit dieser Fähigkeit auszustatten. Dazu setzen wir Experten ein, die dafür sorgen, dass die Teams die vorgegebenen Standards übertreffen. Zweitens implementieren wir Mechanismen, die automatisch kontrollieren, ob Standards eingehalten werden.

> „Gut gemeinte Absichten funktionieren nicht. Wer etwas erreichen will, braucht gute Mechanismen“ – Jeff Bezos.

Das bedeutet konkret, dass wir das Bestmögliche, das Menschen leisten können, durch (oftmals automatisierte) Mechanismen ersetzen, die kontrollieren, ob Regeln oder Prozesse eingehalten werden. Hinter diesem breit aufgestellten Ansatz stehen die Führungsprinzipien von Amazon. Diese stellen sicher, dass in allen Aufgabenbereichen eine Kultur verankert wird, die vom Kunden aus denkt . Vom Kunden aus zu denken, ist ein grundlegender Bestandteil unseres Innovationsprozesses. Unsere Arbeit richtet sich ganz nach dem Kunden und dessen Wünschen. Kundenfixierte Teams richten die Produktentwicklung auf Kundenwünsche aus.

In Zusammenhang mit Architekturen bedeutet das: Wir erwarten von jedem Team, dass es Architekturen erstellen und nach bewährten Methoden arbeiten kann. Um neuen Teams zu diesen Fähigkeiten zu verhelfen bzw. um bestehende Teams leistungsfähiger zu machen, nehmen wir sie in eine virtuelle Community auf, in der Principal Engineers ihre Entwürfe begutachten und sie an die bewährten Methoden von AWS heranführen. Die Community der Principal Engineers hat die Aufgabe, bewährte Methoden sichtbar und verständlich zu machen. Dies geschieht beispielsweise durch Mittagsvorträge, in denen es um die Anwendung bewährter Methoden an praktischen Beispielen geht. Die Vorträge werden aufgezeichnet und können für das Onboarding neuer Teammitglieder eingesetzt werden.

Wir haben bislang mehrere Tausende internetähnliche Systeme eingerichtet und dabei einen Erfahrungsschatz aufgebaut, aus dem sich die bewährten Methoden von AWS herauskristallisiert haben. Wir bevorzugen, bewährte Methoden mit Hilfe von Daten zu definieren. Wir setzen dafür aber auch Fachexperten (z. B. Principal Engineers) ein. Principal Engineers sind direkt dabei, wenn sich neue bewährte Methoden abzeichnen. Als Community können sie bestätigen, dass die Teams danach arbeiten. Im Laufe der Zeit werden diese bewährten Methoden in unsere internen Prüfprozesse sowie in Compliance-Mechanismen aufgenommen. Das Well-Architected Framework ist die kundenseitige Implementierung unseres internen Prüfprozesses. Darin ist die Denkweise der Principal Engineers für Zuständigkeitsbereiche vor Ort (z. B. Solutions Architecture, interne Engineering-Teams) festgeschrieben. Das Well-Architected Framework ist ein skalierbarer Mechanismus, mit dem Sie von diesen Erkenntnissen profitieren können.

Wenn so vorgegangen wird wie in einer Community aus Principal Engineers (mit verteilten Architekturzuständigkeiten), kann unserer Ansicht nach eine Well-Architected Enterprise-Architektur zustande kommen, die auf die Kundenwünsche ausgerichtet ist. Technologievordenker (z. B. CTO oder Entwicklungsleiter), die all Ihre Workloads nach den Prinzipien des Well-Architected-Ansatzes prüfen, können die Risiken Ihres Technologieportfolios aufzeigen. Sie identifizieren teamübergreifende Themen, die Ihre Organisation mit Hilfe von Mechanismen, Training oder Mittagsvorträgen angehen könnte. Allesamt Gelegenheiten für Ihre Principal Engineers, ihr Wissen zu bestimmten Themen an mehrere Teams weiterzugeben.

Allgemeine Designprinzipien

Das Well-Architected Framework fasst allgemeine konzeptionelle Grundsätze zusammen, die gutes Design in der Cloud fördern:

- Keine Ungewissheit mehr über die Kapazität: Wenn Sie bei der Bereitstellung eines Workloads eine schlechte Entscheidung zur Kapazität treffen, sitzen Sie anschließend möglicherweise auf nicht genutzten Ressourcen oder haben zu wenig Kapazität und müssen sich mit mangelnder Performance herumschlagen. Beim Cloud-Computing gibt es diese Probleme nicht. Sie arbeiten mit so viel oder so wenig Kapazität wie nötig. Das System wird automatisch hoch- oder herunterskaliert.
- Systeme auf Produktionsbetrieb testen: Sie können in der Cloud bei Bedarf eine Testumgebung in Produktionsgröße einrichten, Ihre Tests abschließen und die Ressourcen dann wieder außer Betrieb nehmen. Weil Sie für die Testumgebung nur dann zahlen, wenn sie genutzt wird, können Sie Ihre Live-Umgebung zu einem Bruchteil der Kosten testen, die Sie an einem On-Premises-Standort hätten.
- Automatisieren unter Berücksichtigung architektonischer Experimente: Wenn Sie automatisieren, können Sie Ihre Workloads kostengünstig erstellen und replizieren und vermeiden manuellen Aufwand. Sie können an der Automatisierung vorgenommene Änderungen nachverfolgen, die Auswirkungen nachprüfen und ggf. auf die vorherigen Parameter zurücksetzen.
- Evolutionäre Architekturen berücksichtigen: In herkömmlichen Umgebungen sind architekturrelevante Entscheidungen oft als statische, einmalig auftretende Ereignisse implementiert. Dementsprechend gibt es während der Lebensdauer des Systems einige wenige große Versionen. Geschäftsvoraussetzungen und ihr Kontext entwickeln sich stetig weiter. Diese anfangs getroffenen Entscheidungen könnten die Fähigkeit des Systems beeinträchtigen, sich auf neue Geschäftsvoraussetzungen einzustellen. In der Cloud können Sie jederzeit automatisieren und testen. Dadurch wird weniger wahrscheinlich, dass sich Änderungen am Design negativ auswirken. Systeme können sich somit im Laufe der Zeit weiterentwickeln. Unternehmen können dann wie selbstverständlich Innovationen für sich nutzen.
- Mit Daten Architekturen weiterentwickeln: Sie können in der Cloud Daten zu der Frage sammeln, wie Ihre architekturrelevanten Entscheidungen auf das Verhalten Ihres Workloads durchschlagen. Sie können also mit faktenbasierten Entscheidungen Ihren Workload verbessern. Ihre Cloud-Infrastruktur ist Code. Das bedeutet, dass Sie diese Daten im Laufe der Zeit in architekturrelevante Entscheidungen und Verbesserungsmaßnahmen einfließen lassen können.
- Verbesserung mit Hilfe von Ernstfallübungen: Simulieren Sie an regelmäßigen Gamedays Vorfälle in der Produktion, um das Verhalten Ihrer Architektur und Prozesse zu simulieren. So können Sie

nachvollziehen, wo nachgebessert werden kann. Zudem üben Sie dabei ein, wie Ihre Organisation mit Ereignissen umgeht.

Die Säulen des Framework

Wenn Sie ein Softwaresystem bauen, gehen Sie ähnlich vor wie beim Hausbau. Wenn das Fundament nicht trägt, können Risse auftreten und das Gebäude unbrauchbar machen. Wenn Sie die Architektur einer Technologielösung planen und die sechs Säulen Operative Exzellenz, Sicherheit, Zuverlässigkeit, Leistungseffizienz, Kostenoptimierung und Nachhaltigkeit vernachlässigen, kann es schwer werden, ein System zu schaffen, das Ihre Erwartungen und Anforderungen erfüllt. Berücksichtigen Sie aber diese Säulen in Ihrer Architektur, steht am Ende ein stabiles, effizientes System. Und das gibt Ihnen Freiraum, um sich auf andere Designaspekte (z. B. funktionale Anforderungen) zu konzentrieren.

Säulen

- Operational Excellence
- Sicherheit
- Zuverlässigkeit
- Leistungseffizienz
- Kostenoptimierung
- Nachhaltigkeit

Operational Excellence

Die Säule für die betriebliche Exzellenz beinhaltet die Fähigkeit, die Entwicklung zu unterstützen und Workloads effektiv auszuführen, Einblicke in die Betriebsabläufe zu erhalten und unterstützende Prozesse und Verfahren fortlaufend zu verbessern, um geschäftlichen Mehrwert zu schaffen.

Die Säule „Operative Exzellenz“ gibt einen Überblick über konzeptionelle Grundsätze, bewährte Methoden und Fragen. Obligatorische Anleitungen zur Implementierung finden Sie im Whitepaper „Säule der operativen Exzellenz“.

Themen

- Designprinzipien
- Definition
- Bewährte Methoden
- Ressourcen

Designprinzipien

Nachfolgend finden Sie die konzeptionellen Grundsätze für Operational Excellence in der Cloud:

- Organisieren der Teams nach Geschäftsergebnissen: Die Fähigkeit eines Teams, Geschäftsergebnisse zu erzielen, hängt von der Vision der Führung, effektiven Abläufen und einem geschäftsorientierten Betriebsmodell ab. Die Führungskräfte sollten sich voll und ganz für eine CloudOps-Transformation mit einem geeigneten Cloud-Betriebsmodell einsetzen, das die Teams dazu anregt, möglichst effizient zu arbeiten und Geschäftsergebnisse zu erzielen. Ein geeignetes Betriebsmodell nutzt Personal-, Prozess- und Technologiekapazitäten, um zu skalieren, die Produktivität zu optimieren und und durch Agilität, Reaktionsfähigkeit und Anpassung einen Wettbewerbsvorteil zu erlangen. Die langfristige Vision der Organisation wird in Ziele umgesetzt, die Stakeholdern und Verbrauchern Ihrer Cloud-Services unternehmensweit vermittelt werden. Ziele und operative KPIs sind auf allen Ebenen aufeinander abgestimmt. Diese Vorgehensweise sorgt dafür, dass der langfristige Mehrwert, der sich aus der Umsetzung der folgenden Gestaltungsprinzipien ergibt, dauerhaft gewährleistet ist.
- Implementieren von Beobachtbarkeit für umsetzbare Erkenntnisse: Gewinnen Sie ein umfassendes Verständnis hinsichtlich Workload-Verhalten, -Leistung, -Zuverlässigkeit, -Kosten und -Zustand. Legen Sie wichtige Key Performance Indicators (KPIs, Leistungskennzahlen) fest und nutzen Sie die Telemetrie zur Beobachtung, um fundierte Entscheidungen zu treffen und sofort einzugreifen, wenn die Geschäftsergebnisse gefährdet sind. Verbessern Sie proaktiv Leistung, Zuverlässigkeit und Kosten auf der Grundlage von verwertbaren Daten zur Beobachtbarkeit.
- Sicher automatisieren wenn möglich: In der Cloud können Sie die gleichen technischen Vorgehensweisen wie beim Anwendungscode in Ihrer gesamten Umgebung anwenden. Sie können Ihren gesamten Workload und seinen Betrieb (Anwendungen, Infrastruktur, Konfiguration und Verfahren) als Code definieren und aktualisieren. Anschließend können Sie den Betrieb Ihrer Workloads automatisieren, indem Sie sie als Reaktion auf Ereignisse initiieren. In der Cloud können Sie Automatisierungssicherheit einsetzen, indem Sie einen Integritätsschutz wie Ratenkontrolle, Fehlerschwellenwerte und Genehmigungen einrichten. Durch eine effektive Automatisierung können Sie konsistente Reaktionen auf Ereignisse durchsetzen, menschliche Fehler begrenzen und den Arbeitsaufwand der Mitarbeiter reduzieren.
- Durchführen häufiger, kleiner, umkehrbarer Änderungen: Entwerfen Sie Workloads, die skalierbar und lose gekoppelt sind, damit die Komponenten regelmäßig aktualisiert werden können. Automatisierte Bereitstellungstechniken in Verbindung mit kleineren, inkrementellen Änderungen verringern den „Blast Radius“ und ermöglichen eine schnellere Umkehrung bei Fehlern. Dadurch erhöht sich das Vertrauen, vorteilhafte Änderungen an Ihrem Workload vornehmen zu können,

während die Qualität erhalten bleibt und Sie sich schnell an veränderte Marktbedingungen anpassen können.

- Betriebliche Verfahren regelmäßig nachbessern: Wenn Sie Ihre Workloads weiterentwickeln, müssen Sie auch Ihre Abläufe entsprechend anpassen. Suchen Sie beim Einsatz betrieblicher Verfahren nach Möglichkeiten, diese zu verbessern. Führen Sie regelmäßige Überprüfungen durch und vergewissern Sie sich, dass alle Verfahren effektiv sind und dass die Teams mit ihnen vertraut sind. Wenn Lücken festgestellt werden, aktualisieren Sie die Verfahren entsprechend. Informieren Sie alle Beteiligten und Teams über Aktualisierungen der Verfahren. Gamifizieren Sie Ihren Betrieb zum Weitergeben von bewährten Methoden und zur Schulung von Teams.
- Antizipieren von Ausfällen: Maximieren Sie den betrieblichen Erfolg, indem Sie Fehlerszenarien erstellen, um das Risikoprofil des Workloads und seine Auswirkungen auf Ihre Geschäftsergebnisse zu verstehen. Testen Sie die Wirksamkeit Ihrer Verfahren und die Reaktion Ihres Teams auf diese simulierten Fehler. Treffen Sie fundierte Entscheidungen, um offene Risiken zu auszuräumen, die anhand Ihrer Tests identifiziert wurden.
- Lernen aus allen betrieblichen Ereignissen und Metriken: Steigern Sie die Verbesserung durch die Erkenntnisse, die aus allen betrieblichen Ereignissen und Fehlern gewonnen werden. Geben Sie Ihre Erkenntnisse an alle Teams in Ihrer gesamten Organisation weiter. Die Erkenntnisse sollten Daten und Anekdoten enthalten, wie die Betriebsabläufe zu den Geschäftsergebnissen beitragen.
- Nutzen verwalteter Services: Verringern Sie den betrieblichen Aufwand, indem Sie verwaltete AWS-Services nutzen, wo immer dies möglich ist. Erstellen Sie operative Verfahren für die Interaktion mit diesen Services.

Definition

Die bewährten Methoden für operative Exzellenz in der Cloud lassen sich in vier Bereiche einteilen:

- Organisation
- Vorbereitung
- Betrieb
- Weiterentwicklung

Die Geschäftsleitung Ihres Unternehmens definiert Geschäftsziele. Anforderungen und Prioritäten müssen in Ihrem Unternehmen bekannt sein, damit Aufgaben entsprechend organisiert und durchgeführt und die Geschäftsergebnisse erreicht werden können. Ihr Workload muss die Informationen ausgeben, die für dessen Unterstützung erforderlich sind. Die Implementierung von

Services zur Integration, Bereitstellung und Lieferung Ihres Workloads schafft einen erhöhten Fluss nützlicher Änderungen in die Produktion, indem wiederkehrende Prozesse automatisiert werden.

Es kann Risiken im Zusammenhang mit dem Betrieb Ihres Workloads geben. Verstehen Sie diese Risiken und treffen Sie eine fundierte Entscheidung dazu, ob der Übergang in die Produktion vollzogen werden sollte. Ihre Teams müssen in der Lage sein, den Workload zu unterstützen. Geschäfts- und Betriebsmetriken, die von den gewünschten Geschäftsergebnissen abgeleitet werden, erlauben es Ihnen, den Zustand Ihres Workloads und Ihrer Betriebsaktivitäten nachzuvollziehen und auf Vorfälle zu reagieren. Ihre Prioritäten ändern sich, wenn sich Ihre geschäftlichen Anforderungen und die geschäftliche Umgebung ändern. Verwenden Sie diese als Feedback-Schleife, um Ihr Unternehmen und den Betrieb Ihres Workloads kontinuierlich zu verbessern.

Bewährte Methoden

> **Note**
>
> Alle Fragen zur „Operational Excellence“ haben das OPS-Präfix als Abkürzung für die Säule.

Themen

- Organisation
- Vorbereitung
- Betrieb
- Weiterentwicklung

Organisation

Um die Prioritäten festlegen zu können, die den geschäftlichen Erfolg ermöglichen, müssen Ihre Teams gemeinsam in Erfahrung bringen, wie sämtliche Workloads aussehen, welche Rolle die einzelnen Teams dabei spielen und was für geschäftliche Ziele damit erreicht werden sollen. Mit gut definierten Prioritäten erzielen Ihre Bemühungen den größtmöglichen Nutzen. Bewerten Sie die Bedürfnisse interner und externer Kunden. Binden Sie dabei alle wichtigen Beteiligten ein, einschließlich der Geschäfts-, Entwicklungs- und Betriebsteams, um zu bestimmen, auf welche Bereiche die Anstrengungen konzentriert werden sollten. Durch das Bewerten von Kundenbedürfnissen wird sichergestellt, dass Sie den Support, der für die Erzielung der gewünschten geschäftlichen Ergebnisse erforderlich ist, genau kennen und verstehen. Vergewissern Sie sich, dass

Sie sich der Richtlinien oder Verpflichtungen bewusst sind, die von der Führung Ihres Unternehmens definiert wurden. Bewerten Sie externe Faktoren, z. B. gesetzliche Compliance-Anforderungen und Branchenstandards, die einen bestimmten Fokus erfordern oder verstärken können. Überprüfen Sie, ob Sie Mechanismen haben, um Änderungen an internen Governance- und externen Compliance-Anforderungen zu identifizieren. Wenn keine Anforderungen festgestellt werden, stellen Sie sicher, dass diese Prüfung sorgfältig durchgeführt wurde. Überprüfen Sie Ihre Prioritäten regelmäßig, damit sie bei Bedarf aktualisiert werden können.

Bewerten Sie Bedrohungen für das Unternehmen (z. B. Geschäftsrisiken und -verpflichtungen und Bedrohungen der Informationssicherheit) und pflegen Sie diese Informationen in einem Risikoregister. Bewerten Sie die Auswirkungen von Risiken und Kompromissen zwischen konkurrierenden Interessen oder alternativen Ansätzen. Beispielsweise kann eine beschleunigte Markteinführung neuer Funktionen vor der Kostenoptimierung Vorrang haben, oder Sie können eine relationale Datenbank für nicht relationale Daten wählen, um die Migration eines Systems ohne Refactoring zu vereinfachen. Wägen Sie die Vorteile und Risiken ab, um fundierte Entscheidungen zu treffen, wenn es darum geht, auf welche Bereiche die Anstrengungen konzentriert werden sollen. Einige Risiken oder Entscheidungen können eine bestimmte Zeit lang akzeptabel sein. Es gibt ggf. die Möglichkeit, die damit verbundenen Risiken zu minimieren, oder es ist zu einem bestimmten Zeitpunkt nicht mehr akzeptabel, dass ein Risiko weiterhin bestehen bleibt. In diesem Fall ergreifen Sie Maßnahmen, um das Risiko zu beheben.

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams müssen ihre Rolle für den Erfolg anderer Teams und die Rolle anderer Teams für ihren Erfolg verstehen und gemeinsame Ziele haben. Indem sie die Konzepte Verantwortlichkeit und Zuständigkeit verstehen und wissen, wie Entscheidung getroffen werden und wer dazu berechtigt ist, können ihre Anstrengungen fokussiert und der Nutzen Ihrer Teams maximiert werden. Die Anforderungen eines Teams werden durch den unterstützten Kunden, das Unternehmen, die Zusammensetzung des Teams und die Merkmale der jeweiligen Workloads beeinflusst. Es ist nicht sinnvoll, davon auszugehen, dass ein einziges Betriebsmodell alle Teams und Workloads in Ihrem Unternehmen unterstützen kann.

Stellen Sie sicher, dass für jede Anwendung, jeden Workload, jede Plattform und jede Infrastrukturkomponente zuständige Besitzer vorhanden sind und dass jeder Prozess und jedes Verfahren einen festen Besitzer hat, der für die Definition verantwortlich ist, und Besitzer, die für die Leistung verantwortlich sind.

Durch das Verständnis für den geschäftlichen Nutzen der einzelnen Komponenten, Prozesse und Verfahren sowie dafür, weshalb diese Ressourcen vorhanden sind oder Aktivitäten ausgeführt

werden und warum diese Zuständigkeit besteht, basieren die Aktionen Ihrer Teammitglieder auf fundierten Informationen. Definieren Sie eindeutig die Verantwortlichkeiten der Teammitglieder, damit sie entsprechend handeln und Mechanismen zur Identifizierung von Verantwortlichkeit und Zuständigkeit besitzen. Nutzen Sie entsprechende Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen, damit Sie die Innovation nicht einschränken. Definieren Sie Vereinbarungen zwischen Teams, die beschreiben, wie sie für die gegenseitige und die Unterstützung der Geschäftsergebnisse zusammenarbeiten.

Unterstützen Sie Ihre Teammitglieder, damit sie effektiver handeln und positiv zu Ihrem Geschäftsergebnis beitragen können. Die beteiligten Führungskräfte sollten Erwartungen festlegen und den Erfolg messen. Die Geschäftsführung sollte Sponsor, Fürsprecher und treibende Kraft für die Übernahme bewährter Methoden und die Weiterentwicklung des Unternehmens sein. Lassen Sie die Teammitglieder Maßnahmen ergreifen, wenn Ergebnisse gefährdet sind, um Auswirkungen zu minimieren. Sie müssen dazu ermutigt werden, Entscheidungsträger und Interessenvertreter über ermittelte Risiken zu informieren, damit diese angegangen und Vorfälle vermieden werden können. Kommunizieren Sie bekannte Risiken und geplante Ereignisse zeitnah, klar und umsetzbar, damit Teammitglieder rechtzeitig entsprechende Maßnahmen ergreifen können.

Ermutigen Sie das Ausprobieren neuer Ansätze, damit schneller Erkenntnisse erreicht werden und sorgen Sie dafür, dass Teammitglieder interessiert und motiviert bleiben. Teams müssen ihre Fähigkeiten erweitern, um neue Technologien einzuführen und Änderungen bei Bedarf und Zuständigkeiten zu unterstützen. Dies sollten sie durch spezielle, strukturierte Lernzeiten unterstützen und ermutigen. Stellen Sie sicher, dass Ihre Teams über die nötigen Ressourcen verfügen (Tools und Teammitglieder), um positiv zu Ihren Geschäftsergebnissen beitragen zu können. Profitieren Sie von der Diversität im gesamten Unternehmen, um verschiedene einzigartige Standpunkte zu erfahren. Nutzen Sie diese Perspektive, um Innovation zu fördern, Ihre Annahmen in Frage zu stellen und das Risiko einer Verzerrung durch automatische Bestätigung zu reduzieren. Stärken Sie die Inklusion, Diversität und Zugänglichkeit innerhalb Ihrer Teams, um nützliche Perspektiven zu gewinnen.

Wenn es externe gesetzliche Vorschriften oder Compliance-Anforderungen gibt, die für Ihre Organisation gelten, sollten Sie Ihre Teams mithilfe der von AWS-Cloud-Compliance bereitgestellten Ressourcen darin schulen, welche Auswirkungen es bei Ihren Prioritäten zu berücksichtigen gilt. Das Well-Architected Framework legt den Schwerpunkt auf Lernen, Messen und Verbessern. Es bietet einen konsistenten Ansatz, mit dem Sie Architekturen bewerten und Designs implementieren können, die sich im Laufe der Zeit skalieren lassen. AWS stellt das AWS Well-Architected Tool bereit, mit dem Sie Ihren Ansatz vor der Entwicklung, den Status Ihrer Workloads vor der Produktion und den Status Ihrer Workloads in der Produktion überprüfen können. Sie können Workloads mit den neuesten bewährten Methoden für die AWS-Architektur vergleichen, ihren Gesamtstatus überwachen

und Einblicke in potenzielle Risiken erhalten. AWS Trusted Advisor bietet als Tool Zugriff auf verschiedene wichtige Prüfungen, die Optimierungsempfehlungen ausgeben. Diese Informationen können Ihnen beim Festlegen Ihrer Prioritäten helfen. Kunden mit Business und Enterprise Support erhalten Zugriff auf weitere Prüfungen in den Bereichen Sicherheit, Zuverlässigkeit, Leistung, Kostenoptimierung und Nachhaltigkeit, die beim Festlegen von Prioritäten noch hilfreicher sind.

AWS kann Ihnen helfen, Ihre Teams über AWS und die verfügbaren Services zu schulen, sodass alle Mitarbeiter wissen, welche Auswirkungen ihre Entscheidungen auf Ihren Workload haben können. Nutzen Sie bei der Schulung Ihrer Teams die vom AWS -Support (AWS Knowledge Center, AWS Discussion Forums und AWS -Support Center) bereitgestellten Ressourcen und AWS-Dokumente. Wenn Sie eine Frage zu AWS haben, können Sie sich über das AWS -Support Center an den AWS -Support wenden. AWS stellt in der Amazon Builders' Library auch bewährte Methoden und Muster vor, die wir durch den Betrieb von AWS gelernt haben. Eine Vielzahl weiterer nützlicher Informationen finden Sie im AWS-Blog und im offiziellen AWS-Podcast. AWS Training and Certification bietet einige Schulungen durch digitale Kurse im Selbststudium zu den Grundlagen von AWS. Sie können sich auch für eine Schulung registrieren, die von Dozenten geleitet wird, um die AWS-Fähigkeiten Ihres Teams auszubauen.

Verwenden Sie die Tools oder Services, mit denen Sie Ihre Umgebungen kontenübergreifend verwalten können, z. B. AWS Organizations. Das unterstützt Sie bei der Verwaltung Ihrer Betriebsmodelle. Services wie AWS Control Tower erweitern diese Verwaltungsfunktion, sodass Sie Vorlagen (die Ihre Betriebsmodelle unterstützen) für die Einrichtung von Konten definieren, laufende Governance mit AWS Organizations anwenden und die Bereitstellung neuer Konten automatisieren können. Anbieter von verwalteten Services wie AWS Managed Services, AWS Managed Services-Partner oder Anbieter von verwalteten Services im AWS-Partnernetzwerk stellen Fachwissen zur Implementierung von Cloud-Umgebungen bereit und unterstützen Ihre Sicherheits- und Compliance-Anforderungen und Geschäftsziele. Durch die Erweiterung Ihres Betriebsmodells um Managed Services können Sie Zeit und Ressourcen sparen, Ihre internen Teams klein halten und sich auf strategische Ergebnisse konzentrieren, die Ihr Unternehmen auszeichnen, anstatt neue Fähigkeiten und Kompetenzen zu entwickeln.

In den folgenden Fragen geht es um Überlegungen zur Operational Excellence. (Eine Liste der Fragen und bewährten Methoden zur operativen Exzellenz finden Sie im Anhang.)

OPS 1: Wie können Sie Ihre Prioritäten bestimmen?

Jeder muss verstehen, welchen Beitrag er zum Geschäftserfolg leistet. Setzen Sie sich gemeinsame Ziele, damit Sie die Prioritäten für Ressourcen festlegen können. Dadurch erzielen Ihre Bemühungen den größtmöglichen Nutzen.

OPS 2: Wie strukturieren Sie Ihr Unternehmen, um die gewünschten Geschäftsergebnisse zu erzielen?

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams müssen ihre Rolle für den Erfolg anderer Teams und die Rolle anderer Teams für ihren Erfolg verstehen und gemeinsame Ziele haben. Indem sie die Konzepte Verantwortlichkeit und Zuständig keit verstehen und wissen, wie Entscheidung getroffen werden und wer dazu berechtigt ist, können ihre Anstrengungen fokussiert und der Nutzen Ihrer Teams maximiert werden.

OPS 3: Wie unterstützt Ihre Unternehmenskultur Ihre Geschäftsergebnisse?

Lassen Sie Ihren Teammitgliedern Unterstützung zukommen, damit sie effektiver handeln und Ihr Geschäftsergebnis unterstützen können.

Manchmal kann es vorkommen, dass das Augenmerk zu stark auf eine kleine Auswahl von operativen Prioritäten gerichtet wird. Gehen Sie langfristig gut ausgewogen vor, um sicherzustellen, dass erforderliche Fähigkeiten entwickelt und Risiken verwaltet werden. Überprüfen Sie die Prioritäten regelmäßig und passen Sie sie an geänderte Anforderungen an. Wenn Verantwortlichkeit und Zuständigkeit undefiniert oder unbekannt sind, besteht das Risiko, dass erforderliche Aktionen nicht rechtzeitig ausgeführt werden und redundante und potenziell widersprüchliche Anstrengungen unternommen werden, um diese Anforderungen zu erfüllen. Die Unternehmenskultur wirkt sich direkt auf die Zufriedenheit und Bindung der Teammitglieder aus. Ermöglichen Sie die Interaktion und aktivieren Sie die Fähigkeiten Ihrer Teammitglieder für den Erfolg Ihres Unternehmens. Durch Experimente werden Innovationen möglich und Ideen zu Ergebnissen. Sie sollten anerkennen, dass unerwünschte Ergebnisse erfolgreiche Experimente sein können, durch die ein Pfad aufgezeigt wurde, der nicht zum Erfolg führt.

Vorbereitung

Zur Vorbereitung auf operative Exzellenz müssen Sie in Erfahrung bringen, mit welchen Workloads zu rechnen ist und wie diese wahrscheinlich ausfallen werden. Dann können Sie diese so gestalten, dass Sie Einblick in deren Status erhalten und entsprechende Verfahren zu deren Unterstützung entwerfen.

Gestalten Sie Ihren Workload so, dass er die Informationen bereitstellt, die Sie benötigen, um den internen Status (z. B. Metriken, Protokolle, Ereignisse und Ablaufverfolgungen) über alle Komponenten hinweg zu verstehen. Dies erhöht die Transparenz und erleichtert die Untersuchung von Problemen. Beobachtbarkeit geht über die einfache Überwachung hinaus und bietet ein umfassendes Verständnis der internen Funktionsweise eines Systems auf der Grundlage seiner externen Ergebnisse. Beobachtbarkeit basiert auf Metriken, Protokollen und Traces und liefert tiefgreifende Erkenntnisse zum Verhalten und zur Dynamik von Systemen. Mit effektiver Beobachtbarkeit können Teams Muster, Anomalien und Trends erkennen, sodass sie potenzielle Probleme proaktiv angehen und einen optimalen Systemzustand aufrechterhalten können. Die Identifizierung von wichtigen Leistungskennzahlen (KPIs) ist entscheidend, um sicherzustellen, dass die Überwachungsaktivitäten und die Geschäftsziele aufeinander abgestimmt sind. Diese Abstimmung stellt sicher, dass Teams datengestützte Entscheidungen anhand von Metriken treffen, die wirklich wichtig sind, wodurch sowohl die Systemleistung als auch die Geschäftsergebnisse optimiert werden. Darüber hinaus ermöglicht Beobachtbarkeit Unternehmen, proaktiv statt reaktiv zu handeln. Teams können die Ursache-Wirkung-Beziehungen innerhalb ihrer Systeme verstehen und Probleme vorhersagen und verhindern, anstatt nur auf sie zu reagieren. Da sich Workloads weiterentwickeln, ist es wichtig, die Beobachtbarkeitsstrategie immer wieder neu aufzugreifen und zu verfeinern, um sicherzustellen, dass sie relevant und effektiv bleibt.

Verwenden Sie Strategien, die die Übertragung von Änderungen auf die Produktionsumgebung verbessern und einen Faktorwechsel, schnelles Feedback zur Qualität sowie eine schnelle Fehlerbehebung erreichen. Dadurch fließen nützliche Änderungen schneller in die Produktion ein und es treten bei der Bereitstellung weniger Probleme auf. Zudem können Probleme, die durch Bereitstellungsaktivitäten verursacht oder in Ihren Umgebungen erkannt werden, schnell aufgespürt und gelöst werden.

Verwenden Sie Ansätze, die schnelles Feedback zur Qualität liefern und eine schnelle Wiederherstellung bei Änderungen ermöglichen, die nicht zu den gewünschten Ergebnissen führen. Mit diesen Verfahren können Sie die Auswirkung von Problemen eindämmen, die durch Änderungen entstehen. Kalkulieren Sie nicht erfolgreiche Änderungen ein, damit Sie bei Bedarf schneller reagieren und die vorgenommenen Änderungen testen und validieren können. Achten

Sie auf geplante Aktivitäten in Ihren Umgebungen, damit Sie mit dem Risiko von Änderungen umgehen können, die sich auf geplante Aktivitäten auswirken. Nehmen Sie häufige, kleine und umkehrbare Änderungen vor, um den Umfang der Änderungen einzuschränken. Dies beschleunigt die Fehlersuche und ermöglicht eine schnellere Korrektur, da die Möglichkeit besteht, eine Änderung zurückzusetzen. Dies bedeutet auch, dass Sie häufiger von den Vorteilen wertvoller Änderungen profitieren.

Bewerten Sie die operative Bereitschaft Ihres Workloads, der Prozesse und Verfahren sowie Ihrer Mitarbeiter, damit Sie die operativen Risiken im Zusammenhang mit Ihrem Workload genau kennen. Wenden Sie einen konsistenten Prozess (inklusive manueller und automatisierter Checklisten) an, damit Sie wissen, wann Sie bereit sind, Ihren Workload oder eine Änderung live zu schalten. Auf diese Weise können Sie auch alle Bereiche finden, um die Sie sich kümmern müssen. Ihre routinemäßigen Aktivitäten sollten in Runbooks notiert werden, und Playbooks helfen Ihnen bei der Lösung von Problemen. Machen Sie sich mit den Vorteilen und Risiken vertraut, um fundierte Entscheidungen treffen und Änderungen für die Produktion ermöglichen zu können.

Mit AWS können Sie sämtliche Workloads (Anwendungen, Infrastruktur, Richtlinien, Governance und Betrieb) als Code aufrufen. Das bedeutet, dass Sie für jedes Element Ihres Stacks dieselbe technische Vorgehensweise anwenden können, die Sie für Anwendungscode nutzen. Diese können Sie über Teams oder Organisationen hinweg teilen und damit die Auswirkung der Entwicklungsbemühungen verstärken. Verwenden Sie Operations-as-Code in der Cloud und nutzen Sie die Möglichkeit, sicher zu experimentieren, Ihren Workload und betriebliche Verfahren zu entwickeln und Ausfälle zu üben. Durch den Einsatz von AWS CloudFormation verfügen Sie über konsistente, auf Vorlagen basierende und in einer Sandbox befindliche Entwicklungs-, Test- und Produktionsumgebungen mit steigender betrieblicher Kontrolle.

In den folgenden Fragen geht es um Überlegungen zur Operational Excellence.

OPS 4: Wie implementieren Sie die Überwachbarkeit in Ihrem Workload?

Implementieren Sie die Beobachtbarkeit in Ihrem Workload, damit Sie dessen Zustand verstehen und datengesteuerte Entscheidungen auf der Grundlage von Geschäftsanforderungen treffen können.

OPS 5: Wie können Sie Fehler reduzieren, die Fehlerbehebung erleichtern und den Ablauf bis zur Produktion verbessern?

Verwenden Sie Ansätze, die den Fluss von Änderungen in die Produktion verbessern, die einen Faktorwechsel ermöglichen, schnelles Feedback zur Qualität geben und Fehler beheben. Dadurch fließen nützliche Änderungen schneller in die Produktion ein und es treten bei der Bereitstellung weniger Probleme auf. Zudem können Probleme, die durch Bereitstellungsaktivitäten verursacht werden, schnell aufgespürt und gelöst werden.

OPS 6: Wie können Sie Bereitstellungsrisiken eindämmen?

Verwenden Sie Ansätze, die schnelles Feedback zur Qualität liefern und eine schnelle Wiederher stellung bei Änderungen ermöglichen, die nicht zu den gewünschten Ergebnissen führen. Mit diesen Verfahren können Sie die Auswirkung von Problemen eindämmen, die durch Änderungen entstehen.

OPS 7: Wie bringen Sie in Erfahrung, ob Sie für die Unterstützung eines Workloads bereit sind?

Bewerten Sie die Betriebsbereitschaft Ihres Workloads, von Prozessen und Verfahren sowie Ihrer Mitarbeiter, damit Sie die betrieblichen Risiken im Zusammenhang mit Ihrem Workload genau kennen.

Investieren Sie in die Implementierung von Betriebsabläufen als Code, um die Produktivität von Betriebsmitarbeitern zu maximieren, Fehlerraten zu minimieren und automatisierte Reaktionen zu erreichen. Beugen Sie Fehlern nach Möglichkeit vor und stellen Sie entsprechende Abläufe auf. Wenden Sie Metadaten mithilfe von Ressourcen-Tags und AWS Resource Groups nach einer konsistenten Markierungsstrategie an, um die Identifizierung Ihrer Ressourcen zu erreichen. Versehen Sie Ihre Ressourcen mit Tags für Organisation, Kostenkalkulation, Zugriffssteuerung und Zielrichtung der Ausführung von automatisierten Betriebsaktivitäten. Übernehmen Sie Bereitstellungsmethoden, die die Elastizität der Cloud ausnutzen, um Entwicklungsaktivitäten, die Vorabbereitstellung von Systemen und damit schnellere Implementierungen zu ermöglichen. Wenn Sie an Checklisten, mit denen Sie Ihre Workloads beurteilen, Änderungen vornehmen, bedenken Sie auch, was mit live geschalteten Systemen geschehen soll, die mit den Änderungen nicht mehr kompatibel sind.

Betrieb

Beobachtbarkeit ermöglicht es Ihnen, sich auf aussagekräftige Daten zu konzentrieren und die Interaktionen und Ergebnisse Ihrer Workloads zu verstehen. Indem Sie sich auf wichtige Erkenntnisse konzentrieren und unnötige Daten eliminieren, behalten Sie einen einfachen Ansatz zum Verständnis der Workload-Leistung bei. Es ist wichtig, Daten nicht nur zu erfassen, sondern sie auch richtig zu interpretieren. Definieren Sie klare Ausgangswerte, legen Sie geeignete Alarmschwellenwerte fest und überwachen Sie aktiv, ob Abweichungen vorliegen. Wenn eine wichtige Metrik abweicht, insbesondere wenn sie mit anderen Daten korreliert, kann dies spezifische Problembereiche aufzeigen. Mit Beobachtbarkeit sind Sie besser in der Lage, potenzielle Herausforderungen vorherzusehen und zu bewältigen sowie sicherzustellen, dass Ihr Workload reibungslos funktioniert und den Geschäftsanforderungen entspricht.

Der erfolgreiche Betrieb eines Workloads wird daran gemessen, ob geschäftliche Ergebnisse erreicht und Kundenanforderungen erfüllt werden. Definieren Sie zu erwartende Ergebnisse, legen Sie fest, wie der Erfolg gemessen wird, und geben Sie an, welche Metriken in Berechnungen verwendet werden sollen, mit denen festgestellt wird, ob Workload und Betrieb erfolgreich sind. Der betriebliche Status beinhaltet sowohl den Status des Workloads als auch den Status und Erfolg der betrieblichen Vorgänge, die zur Unterstützung des Workloads ausgeführt werden (z. B. Bereitstellung und Vorfallreaktion). Legen Sie Metrikausgangswerte für die Verbesserung, Untersuchung und Intervention fest. Erfassen und analysieren Sie Ihre Metriken und prüfen Sie dann nach, wie weit diese mit ihrem Verständnis von betrieblichen Erfolgen übereinstimmen und welche Änderungen es im zeitlichen Verlauf gibt. Finden Sie anhand gesammelter Metriken heraus, ob kundenseitige und geschäftliche Anforderungen erfüllt werden, und stellen Sie fest, wo noch etwas verbessert werden kann.

Um betriebliche Exzellenz zu erreichen, ist eine effiziente und effektive Verwaltung betrieblicher Ereignisse erforderlich. Dies gilt sowohl für geplante als auch für ungeplante betriebliche Ereignisse. Greifen Sie bei bekannten Ereignissen auf vorab aufgestellte Runbooks zurück. Lassen Sie sich bei der Untersuchung und Behebung von Problemen von Playbooks helfen. Priorisieren Sie Ihre Reaktionen auf Ereignisse anhand der Beeinträchtigungen, die das jeweilige Ereignis für den Geschäftsbetrieb und die Kunden mit sich bringt. Stellen Sie sicher, dass für einen Alarm, der bei einem bestimmten Ereignis ausgelöst werden soll, auch ein auszuführendes Verfahren inklusive eines zuständigen Besitzers festgelegt ist. Legen Sie vorab fest, welche Mitarbeiter für die Behebung eines Ereignisses zuständig sein sollen. Dazu gehören auch Prozesse für einen Eskalationsprozess, über den im Notfall auf der Grundlage der Dringlichkeit und Auswirkungen weitere Mitarbeiter herangezogen werden sollen. Für den Fall, dass eine nicht vorab festgelegte Vorfallreaktion

erforderlich ist, die möglicherweise den geschäftlichen Betrieb beeinträchtigen kann, legen Sie Personen fest, die über die nötige Autorität für Entscheidungen verfügen.

Geben Sie Informationen zum betrieblichen Status von Workloads über Dashboards und Mitteilungen weiter, die auf die Zielgruppe (z. B. Kunde, Unternehmen, Entwickler, Betriebsteam) zugeschnitten sind, damit die jeweiligen Personen geeignete Maßnahmen durchführen können und wissen, wann der normale Betrieb wieder weitergeht.

In AWS können Sie Dashboard-Ansichten Ihrer Metriken generieren, die aus Workloads erfasst wurden oder nativ aus AWS stammen. Sie können CloudWatch oder Anwendungen von Drittanbietern verwenden, um Ansichten von betrieblichen Aktivitäten auf geschäftlicher, Workload-bezogener und betrieblicher Ebene zusammenzustellen und anzuzeigen. AWS stellt über seine Protokollierungsfähigkeiten (wie AWS X-Ray, CloudWatch, CloudTrail und VPC Flow Logs) Einblicke in Workloads bereit. So können Workload-Probleme identifiziert werden, was bei der Ursachenanalyse und Behebung von Fehlern hilft.

In den folgenden Fragen geht es um Überlegungen zur Operational Excellence.

OPS 8: Wie nutzen Sie die Überwachbarkeit von Workloads in Ihrer Organisation?

Sorgen Sie für einen optimalen Zustand des Workloads, indem Sie die Beobachtbarkeit nutzen. Nutzen Sie relevante Metriken, Protokolle und Traces, um sich einen umfassenden Überblick über die Leistung Ihres Workloads zu verschaffen und Probleme effizient zu beheben.

OPS 9: Wie können Sie den Zustand Ihrer Operationen beurteilen?

Definieren, erfassen und analysieren Sie Metriken für Operationen, um einen Einblick in Ereigniss e rund um Ihre Betriebsabläufe zu erhalten. Dies ist wichtig, damit Sie bei Bedarf entsprechende Maßnahmen ergreifen können.

OPS 10: Wie bewältigen Sie Workload- und operationsspezifische Ereignisse?

Erarbeiten und prüfen Sie Verfahren für die Reaktion auf Ereignisse, um Beeinträchtigungen für Ihren Workload zu minimieren.

Alle von Ihnen erfassten Metriken sollten an die geschäftlichen Anforderungen und Ergebnisse angepasst werden, die sie unterstützen. Entwickeln Sie skriptbasierte Antworten auf bekannte Ereignisse und automatisieren Sie deren Leistung als Reaktion auf die Ereigniserkennung.

Weiterentwicklung

Lernen Sie dazu und streben Sie kontinuierliche Verbesserungen an, um anhaltende operative Exzellenz zu erreichen. Planen Sie Arbeitszyklen ein, um nahezu kontinuierlich kleinere Verbesserungen vorzunehmen. Analysieren Sie nach einem Vorfall alle Ereignisse, die sich auf den Kunden auswirken. Identifizieren Sie die beitragenden Faktoren und Präventivmaßnahmen, um Wiederholungen zu begrenzen oder zu verhindern. Teilen Sie den betroffenen Communitys die beitragenden Faktoren nach Bedarf mit. Beurteilen und priorisieren Sie in regelmäßigen Abständen Möglichkeiten für Verbesserungen (z. B. Anfragen nach Features, Behebung von Problemen, Compliance-Anforderungen), inklusive Workload- und Betriebsverfahren.

Nehmen Sie Feedback-Schleifen in Ihre Verfahren auf, um Verbesserungsmöglichkeiten schnell zu erfahren und Rückmeldungen aus dem Praxisbetrieb zu dokumentieren.

Geben Sie die Dinge, die Sie erfahren, an andere Teams weiter, damit alle davon profitieren. Untersuchen Sie, ob Ihre neuen Erkenntnisse vielleicht Trends aufzeigen, und führen Sie nachträglich teamübergreifende Analysen von operativen Metriken durch, um Verbesserungsmöglichkeiten und -methoden festzustellen. Implementieren Sie Änderungen, die zu Verbesserungen führen sollen, und beurteilen Sie deren Ergebnisse.

In AWS können Sie Ihre Protokolldaten nach Amazon S3 exportieren oder Protokolle zur langfristigen Speicherung direkt an Amazon S3 senden. Mit AWS Glue können Sie Ihre Protokolldaten in Amazon S3 für Analysen erkunden und vorbereiten und zugehörige Metadaten in der AWS Glue Data Catalog speichern. Amazon Athena kann durch seine native Integration mit AWS Glue dann zum Analysieren Ihrer Protokolldaten durch Abfragen mit Standard-SQL verwendet werden. Mit einem Business Intelligence-Tool wie Amazon QuickSight können Sie Ihre Daten visualisieren, untersuchen und analysieren. Erkennen von Trends und Ereignissen, die zu einer Verbesserung führen können.

In den folgenden Fragen geht es um Überlegungen zur Operational Excellence.

OPS 11. Wie können Sie Arbeitsvorgänge weiterentwickeln?

Widmen Sie nahezu kontinuierlichen inkrementellen Verbesserungen Zeit und Ressourcen, um die Effektivität und Effizienz Ihrer Betriebsabläufe weiterzuentwickeln.

Die Voraussetzung für eine erfolgreiche Weiterentwicklung des Betriebs sind kontinuierliche kleinere Verbesserungen, das Bereitstellen sicherer Umgebungen und Zeitfenster zum Experimentieren, das Entwickeln und Testen von Verbesserungen sowie die Schaffung eines Umfeldes, in dem alle ermutigt werden, aus Fehlern zu lernen. Die operative Unterstützung für Sandbox-, Entwicklungs-, Test- und Produktionsumgebungen, mit steigenden Leveln von operativer Kontrolle erleichtert die Entwicklung und steigert die Kalkulierbarkeit, dass Änderungen zu erfolgreichen Ergebnissen führen.

Ressourcen

Weitere Informationen zu bewährten Methoden für operative Exzellenz finden Sie in den folgenden Ressourcen.

Dokumentation

- DevOps und AWS

Whitepaper

- Säule „Operative Exzellenz“

Video

- DevOps bei Amazon

Sicherheit

In der Säule der Sicherheit wird beschrieben, wie Sie Daten, Systeme und Komponenten so schützen, dass Sie Cloud-Technologien nutzen können, um Ihre Sicherheitslage zu verbessern.

Die Säule für Sicherheit bietet einen Überblick über konzeptionelle Grundsätze, Best Practices und Fragen. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper der Säule für Sicherheit.

Themen

- Designprinzipien
- Definition
- Bewährte Methoden

- Ressourcen

Designprinzipien

Es gibt sieben Designprinzipien für die Sicherheit in der Cloud:

- Implementieren einer starken IdentitätsgrundlageImplementieren Sie das Prinzip der geringsten Berechtigung und erzwingen Sie die Trennung von Pflichten durch eine entsprechende Autorisierung für jede Interaktion mit Ihren AWS-Ressourcen. Zentralisieren Sie die Identitätsverwaltung und vermeiden Sie die Abhängigkeit von langfristigen statischen Anmeldeinformationen.
- Nachverfolgbarkeit: Überwachen, melden und prüfen Sie Aktionen und Änderungen in Ihrer Umgebung in Echtzeit. Integrieren Sie die Protokoll- und Metrikerfassung in Systeme, um automatisch zu untersuchen und Maßnahmen zu ergreifen.
- Sicherheit auf allen Ebenen: Wenden Sie einen umfassenden Verteidigungsansatz mit mehreren Sicherheitskontrollen an. Wenden Sie diesen auf allen Ebenen an (z. B. Netzwerkgrenzen, VPC, Lastverteilung, alle Instances und Datenverarbeitungsservices, Betriebssystem, Anwendung und Code).
- Automatisieren bewährter Sicherheitsverfahren: Mithilfe automatisierter softwarebasierter Sicherheitsmechanismen können Sie Ihr System sicher, schnell und kosteneffektiv skalieren. Erstellen Sie sichere Architekturen, einschließlich implementierter Kontrollen, die als Code in versionsgesteuerten Vorlagen definiert und verwaltet werden.
- Schutz von Daten während der Übertragung und im Ruhezustand: Klassifizieren Sie Daten nach Sensibilität und Nutzungsmechanismen wie Verschlüsselung, Tokenisierung und Zugriff, sofern zutreffend.
- Trennen von Benutzern und Daten: Verwenden Sie Mechanismen und Tools, um den direkten Zugriff oder die manuelle Verarbeitung von Daten zu reduzieren oder gänzlich zu eliminieren. Sie reduzieren dadurch das Risiko, dass sensible Daten verloren gehen, geändert werden oder anderweitigen Benutzerfehlern unterliegen.
- Vorbereitung auf Sicherheitsereignisse: Seien Sie auf Vorfälle vorbereitet. Richten Sie entsprechend Ihren organisatorischen Anforderungen ein Verfahren zur Vorfallverwaltung sowie Richtlinien für die Überprüfung ein. Simulieren Sie Vorfallreaktionen und nutzen Sie automatisierbare Tools, um die Erkennung, Untersuchung und Wiederherstellung zu beschleunigen.

Definition

Es gibt sechs bewährte Methoden für die Sicherheit in der Cloud:

- Sicherheit
- Identity and Access Management
- Erkennung
- Schutz der Infrastruktur
- Datenschutz
- Vorfallbehandlung

Vor der Entwicklung von Workloads ist es wichtig, geeignete Sicherheitsverfahren festzulegen. Sie müssen die einzelnen Prozesse steuern können. Wichtig ist auch, dass Sie Sicherheitsvorfälle erkennen, Ihre Systeme und Services schützen und die Vertraulichkeit und Integrität von Daten durch entsprechende Schutzmaßnahmen wahren können. Richten Sie ein gut definiertes und geübtes Verfahren ein, das es Ihnen ermöglicht, auf Sicherheitsvorfälle zu reagieren. Derartige Tools und Techniken sind unabdinglich, um Ihr Unternehmen vor finanziellen Verlusten zu schützen und gesetzliche Vorgaben zu erfüllen.

Das AWS-Modell der geteilten Verantwortung ermöglicht Unternehmen, durch die Migration zur Cloud ihre Sicherheits- und Compliance-Ziele zu erfüllen. Dadurch, dass sich AWS um den physischen Schutz der Infrastruktur unserer Cloud-Services kümmert, können Sie sich als AWS-Kunde darauf konzentrieren, mithilfe unserer Services Ihre Ziele zu erreichen. Sie haben in der AWS Cloud auch einen verbesserten Zugriff auf Sicherheitsdaten und können automatisch auf Sicherheitsereignisse reagieren.

Bewährte Methoden

Themen

- Sicherheit
- Identity and Access Management
- Erkennung
- Schutz der Infrastruktur
- Datenschutz
- Vorfallsreaktion

Sicherheit

Um Ihre Workload sicher zu betreiben, müssen Sie auf jeden Sicherheitsbereich übergreifende bewährte Methoden anwenden. Nutzen Sie Anforderungen und Prozesse, die Sie in Operational Excellence definiert haben, auf Organisations- und Workload-Ebene, und wenden Sie sie auf alle Bereiche an.

Bleiben Sie auf dem Laufenden mit AWS- und Branchenempfehlungen sowie Bedrohungsinformationen, um Ihr Bedrohungsmodell und Ihre Kontrollziele weiterzuentwickeln. Durch die Automatisierung von Sicherheitsprozessen, Tests und Validierung können Sie Ihre Sicherheitsvorgänge skalieren.

In der folgenden Frage geht es um Überlegungen zur Sicherheit. (Eine Liste der Fragen und bewährten Methoden zur Sicherheit finden Sie im Anhang).

SICH 1: Wie können Sie Ihre Workload sicher betreiben?
Um Ihre Workload sicher zu betreiben, müssen Sie auf jeden Sicherheitsbereich übergreifende bewährte Methoden anwenden. Nutzen Sie Anforderungen und Prozesse, die Sie in Operational Excellence definiert haben, auf Organisations- und Workload-Ebene, und wenden Sie sie auf alle Bereiche an. Bleiben Sie auf dem Laufenden mit Empfehlungen von AWS, branchenspezifischen Quellen sowie Informationsquellen zu Bedrohungen, um Ihr Bedrohungsmodell und Ihre Kontrollz iele weiterzuentwickeln. Durch die Automatisierung von Sicherheitsprozessen, Tests und Validieru ng können Sie Ihre Sicherheitsvorgänge skalieren.

In AWS empfehlen wir die Trennung verschiedener Workloads nach Konto, basierend auf ihrer Funktion und den Anforderungen an die Compliance oder Datensensibilität.

Identity and Access Management

Das Identity and Access Management ist ein wichtiger Bestandteil eines Informationssicherheitsprogramms. Es stellt sicher, dass nur autorisierte und authentifizierte Benutzer in dem von Ihnen gewünschten Umfang auf Ihre Ressourcen zugreifen können. Definieren Sie beispielsweise Prinzipien (d. h. Konten, Benutzer, Rollen und Services, die Aktionen in Ihrem Konto durchführen), erstellen Sie entsprechende Richtlinien, und implementieren Sie eine strenge Verwaltung von Anmeldeinformationen. Diese Elemente der Rechteverwaltung bilden die Grundlage der Authentifizierung und Autorisierung.

In AWS erfolgt die Rechteverwaltung primär durch den AWS Identity and Access Management (IAM)-Service. Damit können Sie sowohl den Benutzerzugriff als auch den programmgesteuerten Zugriff auf AWS-Services und -Ressourcen steuern. Wenden Sie detaillierte Richtlinien an, um Benutzern, Gruppen, Rollen oder Ressourcen Berechtigungen zuzuweisen. Darüber hinaus können Sie die Verwendung starker Kennwörter erzwingen. Sie können deren Komplexität vorgeben, Wiederverwendungen vermeiden und Multi-Factor Authentication (MFA) nutzen. Sie haben die Möglichkeit, die Rechteverwaltung mit Ihrem Verzeichnisdienst zu verbinden. Wenn Sie Workloads haben, die Zugriff auf AWS erfordern, ermöglicht IAM diesen auf sichere Weise durch Rollen, Instance-Profile, Identitätsverbund und temporäre Anmeldeinformationen.

In den folgenden Fragen geht es um Überlegungen zur Sicherheit.

SICH 2: Wie verwalten Sie Identitäten für Personen und Maschinen?

Es gibt zwei Arten von Identitäten, die Sie beim Betrieb sicherer AWS-Workloads verwalten müssen. Wenn Sie wissen, welche Art von Identität Sie verwalten und wie Sie Zugriff gewähren müssen, können Sie sicherstellen, dass die richtigen Identitäten unter den richtigen Bedingungen Zugriff auf die richtigen Ressourcen haben.

Menschliche Identitäten: Ihre Administratoren, Entwickler, Bediener und Endbenutzer benötigen eine Identität für den Zugriff auf Ihre AWS-Umgebungen und -Anwendungen. Dies sind Mitgliede r Ihrer Organisation oder externe Benutzer, mit denen Sie zusammenarbeiten, und die mit Ihren AWS-Ressourcen über einen Webbrowser, eine Client-Anwendung oder interaktive Befehlszeilen-Tools interagieren.

Maschinenidentitäten: Ihre Service-Anwendungen, betrieblichen Tools und Workloads benötigen eine Identität, um Anforderungen an AWS-Services zu stellen, z. B. um Daten zu lesen. Zu diesen Identitäten gehören Maschinen, die in Ihrer AWS-Umgebung ausgeführt werden, z. B. Amazon EC2-Instances oder AWS Lambda-Funktionen. Sie können auch Maschinenidentitäten für externe Parteien verwalten, die Zugriff benötigen. Darüber hinaus verfügen Sie möglicherweise auch über Maschinen außerhalb von AWS, die Zugriff auf Ihre AWS-Umgebung benötigen.

SICH 3: Wie verwalten Sie Berechtigungen für Personen und Maschinen?

Verwalten Sie Berechtigungen zum Steuern des Zugriffs auf Personen- und Maschinenidentitäten, die Zugriff auf AWS und Ihren Workload benötigen. Berechtigungen steuern, wer worauf und unter welchen Bedingungen zugreifen kann.

Anmeldeinformationen dürfen nicht zwischen Benutzern oder Systemen weitergegeben werden. Der Benutzerzugriff sollten nach dem Prinzip der geringsten Rechte erfolgen, passwortgeschützt sein und nur mittels MFA möglich sein. Der programmgesteuerte Zugriff etwa durch API-Aufrufe von AWS-Services sollte mit eingeschränkten Berechtigungen und temporären Anmeldeinformationen erfolgen, die beispielsweise durch den AWS Security Token Service ausgegeben werden.

AWS bietet Ressourcen, die Ihnen das Identity and Access Management erleichtern. Mehr zu den Best Practices erfahren Sie in unseren praktischen Übungen zu den Themen Verwaltung von Anmeldeinformationen und Authentifizierung, Steuerung des Benutzerzugriffs, und Steuerung des programmgesteuerten Zugriffs.

Erkennung

Aufdeckende Kontrollen bieten Ihnen die Möglichkeit, potenzielle Sicherheitsbedrohungen oder -vorfälle zu erkennen. Die Kontrollmechanismen sind ein wesentlicher Bestandteil von Governance-Frameworks. Sie können zur Unterstützung von Qualitätssicherungsverfahren, zur Einhaltung gesetzlicher Vorgaben und Pflichten sowie zur Erkennung und Abwehr von Bedrohungen genutzt werden. Es gibt unterschiedliche Arten aufdeckender Kontrollen. Eine Bestandserfassung der Ressourcen und ihrer detaillierten Attribute trägt beispielsweise zu einer effektiveren Entscheidungsfindung (und Lebenszyklussteuerung) bei, wenn es darum geht, operative Ausgangswerte festzulegen. Sie können auch durch eine interne Prüfung der mit Informationssystemen verbundenen Steuerelemente sicherstellen, dass Ihre Verfahren den Richtlinien und Anforderungen entsprechen. Basierend auf definierten Bedingungen sind passende automatisierte Benachrichtigungen möglich. Diese Steuerelemente sind wichtige reaktive Faktoren, die es Ihrem Unternehmen ermöglichen, den Umfang anomaler Aktivitäten zu ermitteln und zu verstehen.

In AWS können Sie aufdeckende Kontrollen durch Verarbeitungsprotokolle, Ereignisse und Überwachungsfunktionen implementieren, die eine Prüfung, automatisierte Analyse und Benachrichtigung ermöglichen. Mit CloudTrail-Protokollen, AWS API-Aufrufen und CloudWatch können Sie Kennzahlen überwachen und Benachrichtigungen senden. Der

Konfigurationsverlauf ist mit AWS Config einsehbar. Amazon GuardDuty ist ein verwalteter Service zur Bedrohungserkennung, der Ihre AWS-Konten und -Workloads zu deren Schutz fortlaufend auf böswillige oder unbefugte Verhaltensweisen überwacht. Mit Serviceprotokollen etwa von Amazon Simple Storage Service (Amazon S3) können Sie Zugriffsanfragen protokollieren.

In der folgenden Frage geht es um Überlegungen zur Sicherheit.

SICH 4: Wie erkennen und untersuchen Sie Sicherheitsereignisse?
Erfassen und analysieren Sie Ereignisse mithilfe von Protokollen und Kennzahlen, um Einblick zu erhalten. Ergreifen Sie Maßnahmen bei Sicherheitsereignissen und potenziellen Bedrohungen, um Ihren Workload zu schützen.

Die Protokollverwaltung ist für eine Well-Architected-Workload wichtig, um so vielfältige Bereiche wie Sicherheit, Forensik sowie die Einhaltung gesetzlicher Vorgaben abzudecken. Zur Ermittlung potenzieller Sicherheitsvorfälle müssen diese Protokolle analysiert und bei Bedarf entsprechende Maßnahmen ergriffen werden. AWS bietet Funktionen, die die Protokollverwaltung erleichtern. Sie können damit einen Lebenszyklus für die Datenaufbewahrung festlegen oder angeben, wo Daten gespeichert, archiviert oder schließlich gelöscht werden. Dies vereinfacht die vorhersehbare und zuverlässige Datenverarbeitung und senkt die Kosten.

Schutz der Infrastruktur

Zum Schutz der Infrastruktur sind Steuermethoden wie etwa eine tiefgreifende Abwehr erforderlich, um Best Practices sowie organisatorische und gesetzliche Verpflichtungen zu erfüllen. Die Nutzung dieser Methoden ist für erfolgreiche, kontinuierliche Betriebsabläufe sowohl in der Cloud als auch lokal ausschlaggebend.

AWS ermöglicht die Überprüfung zustandsbehafteter und zustandsloser Pakete. Sie können dafür wahlweise AWS-native Technologien oder im AWS Marketplace angebotene Partnerprodukte und -services nutzen. Amazon Virtual Private Cloud (Amazon VPC) wird empfohlen, um eine private, sichere und skalierbare Umgebung zu erstellen, in der Sie Ihre Topologie, einschließlich Gateways, Routing-Tabellen sowie öffentlichen und privaten Subnetzen definieren können.

In den folgenden Fragen geht es um Überlegungen zur Sicherheit.

SEC 5: Wie schützen Sie Ihre Netzwerkressourcen?
Alle Workloads, die über eine Art Netzwerkverbindung verfügen, unabhängig davon, ob es sich um das Internet oder ein privates Netzwerk handelt, erfordern mehrere Abwehrebenen, um Schutz vor externen und internen Netzwerkbedrohungen sicherzustellen.

SICH 6: Wie schützen Sie Ihre Datenverarbeitungsressourcen?
Datenverarbeitungsressourcen in Ihrem Workload erfordern mehrere Ebenen der Abwehr zum Schutz vor externen und internen Bedrohungen. Zu den Datenverarbeitungsressourcen zählen EC2-Instances, Container, AWS Lambda-Funktionen, Datenbankservices, IoT-Geräte und mehr.

Ungeachtet der Umgebung sollten mehrere Abwehrebenen vorhanden sein. Was den Schutz der Infrastruktur anbelangt, gelten viele der Konzepte und Methoden für Cloud- und lokale Modelle gleichermaßen. Das Erzwingen des Grenzschutzes, die Überwachung von Ein- und Ausgangspunkten sowie die umfassende Protokollierung, Überwachung und Benachrichtigung sind für einen effektiven Informationssicherheitsplan wichtig.

AWS-Kunden können die Konfiguration der Amazon Elastic Compute Cloud (Amazon EC2) sowie von Amazon Elastic Container Service-Containern (Amazon ECS) und AWS Elastic Beanstalk-Instances anpassen oder härten und in einem unveränderlichen Amazon Machine Image (AMI) speichern. Dadurch erhalten alle neuen virtuellen Server (Instances), die mit diesem AMI gestartet werden, diese gehärtete Konfiguration. Dabei spielt es keine Rolle, ob sie durch Auto Scaling oder manuell ausgelöst wurden.

Datenschutz

Vor der Entwicklung eines Systems sollten grundlegende Sicherheitspraktiken implementiert werden. Mittels Datenklassifizierung lassen sich beispielsweise organisatorische Daten nach Sensitivität kategorisieren. Die Verschlüsselung macht sie zudem für unbefugte Benutzer unleserlich. Derartige Tools und Techniken sind unabdinglich, um Ihr Unternehmen vor finanziellen Verlusten zu schützen und gesetzliche Vorgaben zu erfüllen.

In AWS können Sie Daten mit folgenden Maßnahmen schützen:

- Als AWS-Kunde behalten Sie die vollständige Kontrolle über Ihre Daten.

- AWS erleichtert Ihnen die Datenverschlüsselung und die Schlüsselverwaltung, einschließlich einer regulären Schlüsselrotation. Sie können diese auf einfache Weise selbst verwalten oder von AWS automatisieren lassen.
- Sie haben Zugriff auf detaillierte Protokolle mit wichtigen Angaben etwa zu Dateizugriffen und -änderungen.
- Die Speichersysteme von AWS zeichnen sich durch eine exzeptionelle Ausfallsicherheit aus. Amazon S3 Standard, S3 Standard-IA, S3 One Zone-IA und Amazon Glacier bieten beispielsweise eine einjährige Objektlanglebigkeit von 99,999999999 %. Dies entspricht einem jährlichen erwarteten Verlust von 0,000000001 % der Objekte.
- Die Versionierung, die in ein umfassenderes Verfahren zur Datenlebenszyklusverwaltung eingebunden sein kann, bietet Schutz vor versehentlichen Überschreibungen, Löschungen und ähnlichen Gefahren.
- AWS veranlasst niemals eine Verschiebung von Daten zwischen Regionen. Die in einer Region platzierten Inhalte bleiben in dieser Region, sofern Sie dies nicht ausdrücklich mithilfe einer Funktion oder eines Services veranlassen.

In den folgenden Fragen geht es um Überlegungen zur Sicherheit.

SICH 7: Wie klassifizieren Sie Ihre Daten?

Die Datenklassifizierung bietet eine Möglichkeit, Daten basierend auf Wichtigkeit und Sensibilität zu kategorisieren, um Ihnen dabei zu helfen, angemessene Schutz- und Aufbewahrungskontrollen zu bestimmen.

SICH 8: Wie schützen Sie Ihre Daten im Ruhezustand?

Schützen Sie Ihre Daten im Ruhezustand, indem Sie mehrere Kontrollen implementieren, um das Risiko eines unbefugten Zugriffs oder eines Missbrauchs zu reduzieren.

SICH 9: Wie schützen Sie Ihre Daten bei der Übertragung?

Schützen Sie Ihre Daten während der Übertragung, indem Sie mehrere Kontrollen implementieren, um das Risiko eines unbefugten Zugriffs oder Verlusts zu reduzieren.

AWS bietet mehrere Möglichkeiten zur Verschlüsselung von Daten im Ruhezustand und während der Übertragung. Unsere Services enthalten Funktionen, die die Verschlüsselung Ihrer Daten erleichtern. Wir haben beispielsweise in Amazon S3 eine serverseitige Verschlüsselung (Server-Side Encryption, SSE) implementiert, die die Speicherung Ihrer Daten in verschlüsselter Form vereinfacht. Sie können auch das komplette Ver- und -Entschlüsselungsverfahren mit HTTPS (generell als SSL-Terminierung bekannt) mit Elastic Load Balancing (ELB) arrangieren.

Vorfallsreaktion

Obwohl die präventiven und aufdeckenden Kontrollen mittlerweile extrem ausgereift sind, sollte Ihr Unternehmen dennoch Verfahren etablieren, um auf Sicherheitsvorfälle reagieren und mögliche Auswirkungen mindern zu können. Wie effektiv Ihre Teams bei einem Vorfall reagieren können, um Systeme zu isolieren oder zu bergen und Betriebsabläufe in einem bekanntermaßen funktionierenden Zustand wiederherzustellen, hängt stark von der Architektur des Workloads ab. Indem Sie sich mit entsprechenden Tools und Zugriffsmöglichkeiten auf Sicherheitsvorfälle vorbereiten und die Vorfallreaktion regelmäßig im Rahmen von Gamedays üben, stellen Sie eine zeitnahe Untersuchung und Wiederherstellung sicher.

In AWS ermöglichen die folgenden Praktiken eine effektive Vorfallreaktion:

- Eine detaillierte Protokollierung wichtiger Informationen etwa zu Dateizugriffen und -änderungen.
- Ereignisse können automatisch verarbeitet werden und Tools auslösen, die Reaktionen über AWS APIs automatisieren.
- Sie können vorab mit AWS CloudFormation entsprechende Tools und einen "Reinraum" bereitstellen. Sie erhalten dadurch eine sichere, isolierte Umgebung für forensische Untersuchungen.

In der folgenden Frage geht es um Überlegungen zur Sicherheit.

SICH 10: Wie können Sie Vorfälle voraussagen, darauf reagieren und diese beheben?
Die Vorbereitung ist entscheidend für eine rechtzeitige und effektive Untersuchung, Reaktion auf und Wiederherstellung nach Sicherheitsvorfällen, um Unterbrechungen der Geschäftsabläufe zu minimieren.

Wichtig ist, dass Sie eine Möglichkeit haben, Ihrem Sicherheitsteam für forensische Zwecke schnell Zugriff gewähren zu können. Automatisieren Sie sowohl die Isolation von Instances als auch die Erfassung von Daten und Zuständen.

Ressourcen

Werfen Sie einen Blick auf die folgenden Ressourcen, um mehr über unsere bewährten Methoden für die Sicherheit zu erfahren.

Dokumentation

- AWS Cloud-Sicherheit
- AWS-Compliance
- AWS-Sicherheitsblog

Whitepaper

- Säule „Sicherheit“
- Übersicht über AWS-Sicherheit
- AWS – Risiko und Compliance

Video

- AWS-Sicherheitsstatus der Union
- Übersicht über die gemeinsame Verantwortlichkeit

Zuverlässigkeit

Die Säule „Zuverlässigkeit“ umfasst die Fähigkeit eines Workloads, die beabsichtigte Funktion erwartungsgemäß korrekt und konsistent auszuführen. Dies umfasst die Möglichkeit, den Workload während des gesamten Lebenszyklus zu betreiben und zu testen. Dieses Dokument bietet umfassende Informationen mit Best Practices für die Implementierung zuverlässiger Workloads in AWS.

Die Säule der Zuverlässigkeit bietet einen Überblick über Designprinzipien, bewährte Methoden und Fragen. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule der Zuverlässigkeit.

Themen

- Designprinzipien
- Definition
- Bewährte Methoden
- Ressourcen

Designprinzipien

Es gibt fünf Designprinzipien für die Zuverlässigkeit in der Cloud:

- Automatische Wiederherstellung nach einem Fehler: Durch die Überwachung wichtiger Leistungskennzahlen (KPIs, Key Performance Indicators) eines Workloads können Sie die Automatisierung auslösen, sobald ein Schwellenwert überschritten wurde. Diese KPIs sollten als Kennzahlen für den Geschäftswert und nicht als technische Aspekte für den Betrieb des Service betrachtet werden. Dies ermöglicht eine automatische Benachrichtigung bei und Verfolgung von Fehlern sowie die Einleitung einer automatisierten Wiederherstellung, die eine Fehlerumgehung bietet oder den Fehler behebt. Bei einer ausgefeilteren Automatisierung ist es möglich, Fehler vor ihrem eigentlichen Auftreten zu antizipieren und zu beheben.
- Testen von Wiederherstellungsverfahren: In einer lokalen Umgebung werden Tests häufig durchgeführt, um nachzuweisen, dass der Workload in einem bestimmten Szenario funktioniert. Mit den Tests werden in der Regel keine Wiederherstellungsstrategien validiert. In der Cloud können Sie testen, in welchen Situationen die Workload Fehler produziert, und Sie können die Wiederherstellungsverfahren validieren. Mit der Automatisierung können Sie verschiedene Fehler simulieren oder Szenarios reproduzieren, die zuvor zu Fehlern geführt haben. Diese Vorgehensweise legt Fehlerpfade offen, die Sie testen und beheben können, bevor ein echtes Fehlerszenario auftritt. Dadurch werden die Risiken verringert.
- Horizontales Skalieren zur Erhöhung der aggregierten Workload-Verfügbarkeit: Ersetzen Sie eine große Ressource durch mehrere kleine Ressourcen, um die Auswirkung eines einzelnen Fehlers auf den Gesamt-Workload zu reduzieren. Verteilen Sie Anfragen auf mehrere kleinere Ressourcen, damit sie keine gemeinsame Fehlerquelle aufweisen.
- Genaue Analyse der verfügbaren Kapazität: Eine häufige Fehlerursache bei lokalen Workloads ist die Ressourcensättigung. Ein solches Szenario liegt vor, wenn die Anforderungen an den

Workload dessen Kapazität überschreiten (dies ist häufig das Ziel von Denial-of-Service-Angriffen). In der Cloud können Sie die Nachfrage und die Workload-Auslastung überwachen und das Hinzufügen oder Entfernen von Ressourcen automatisieren, um den Bedarf ohne Über- oder Unterbereitstellung stets optimal zu erfüllen. Es gibt weiterhin Grenzen, aber einige Kontingente können gesteuert und andere verwaltet werden (siehe "Service Quotas und Einschränkungen verwalten").

- Verwalten von Änderungen an der Automatisierung: Änderungen an Ihrer Infrastruktur sollten über die Automatisierung vorgenommen werden. Zu den Änderungen, die verwaltet werden müssen, gehören Änderungen an der Automatisierung, die anschließend nachverfolgt und überprüft werden können.

Definition

Die bewährten Methoden für Zuverlässigkeit in der Cloud lassen sich in vier Bereiche einteilen:

- Grundlagen
- Workload-Architektur
- Änderungsmanagement
- Fehlerverwaltung

Um Zuverlässigkeit zu erreichen, müssen Sie mit den Grundlagen beginnen – einer Umgebung, in der Servicekontingente und die Netzwerktopologie für die Workload angemessen sind. Die Workload-Architektur des verteilten Systems muss so ausgelegt sein, dass Ausfälle verhindert und abgemildert werden. Die Workload muss Änderungen in Bezug auf den Bedarf oder die Anforderungen verarbeiten und so konzipiert sein, dass sie Fehler erkennt und sie automatisch selbst behebt.

Bewährte Methoden

Themen

- Grundlagen
- Workload-Architektur
- Änderungsverwaltung
- Fehlerverwaltung

Grundlagen

Grundlegende Anforderungen sind diejenigen, deren Umfang über einen einzelnen Workload oder ein einzelnes Projekt hinausgeht. Vor dem Aufbau der Architektur eines System sollten grundlegende Anforderungen, die sich auf die Zuverlässigkeit auswirken, implementiert werden. So müssen Sie beispielsweise Ihre Rechenzentren mit einer ausreichenden Netzwerkbandbreite versorgen.

In AWS sind die meisten dieser grundlegenden Anforderungen bereits berücksichtigt oder können nach Bedarf erfüllt werden. Die Cloud bietet nahezu unbegrenzte Möglichkeiten. Daher liegt es in der Verantwortung von AWS, die Anforderungen in Bezug auf ausreichende Netzwerk- und Rechenkapazität zu erfüllen. Sie können die Ressourcengröße und die Zuweisungen nach Bedarf ändern.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit. (Eine Liste der Fragen und bewährten Methoden zur Zuverlässigkeit finden Sie im Anhang).

ZUV 1: Was ist bei der Verwaltung von Servicekontingenten und Einschränkungen zu beachten?
Für cloudbasierte Workload-Architekturen gibt es Servicekontingente (die auch als Service Limits bezeichnet werden). Diese Kontingente dienen dazu, nicht versehentlich mehr Ressource n bereitzustellen als nötig und Anfrageraten für API-Vorgänge zu begrenzen, um Services vor Missbrauch zu schützen. Darüber hinaus gibt es Ressourceneinschränkungen, z. B. die Rate, mit der Bits durch ein Glasfaserkabel geschleust werden können, oder die Speichermenge auf einer physischen Festplatte.

ZUV 2: Was ist bei der Planung der Netzwerktopologie zu beachten?
Workloads existieren häufig in mehreren Umgebungen. Dazu gehören mehrere Cloud-Umg ebungen (öffentlich zugängliche und private) und möglicherweise die vorhandene Infrastruktur Ihres Rechenzentrums. Die Pläne müssen Netzwerkaspekte umfassen, wie z. B. die Konnektivität innerhalb und zwischen Systemen, die Verwaltung öffentlicher und privater IP-Adressen und die Auflösung von Domänennamen.

Für cloudbasierte Workload-Architekturen gibt es Servicekontingente (die auch als Service Limits bezeichnet werden). Diese Kontingente sollen verhindern, dass versehentlich mehr Ressourcen bereitgestellt werden als nötig. Zudem begrenzen sie die Anfrageraten für API-Vorgänge, um

Services vor Missbrauch zu schützen. Workloads existieren häufig in mehreren Umgebungen. Diese Kontingente müssen Sie für alle Workload-Umgebungen überwachen und verwalten. Dazu gehören mehrere Cloud-Umgebungen (öffentlich zugängliche und private) und möglicherweise die vorhandene Infrastruktur Ihres Rechenzentrums. Die Pläne müssen Netzwerkaspekte umfassen, wie z. B. Konnektivität innerhalb und zwischen Systemen, Verwaltung öffentlicher und privater IP-Adressen und Auflösung von Domänennamen.

Workload-Architektur

Ausgangspunkt für einen zuverlässigen Workload sind vorab getroffene Designentscheidungen für Software und Infrastruktur. Ihre Auswahl in puncto Architektur wirkt sich in allen fünf Well-Architected-Säulen auf das Verhalten der Workload aus. Zur Gewährleistung von Zuverlässigkeit sind bestimmte Muster zu befolgen.

Bei AWS haben Entwickler von Workloads die Wahl zwischen verschiedenen Sprachen und Technologien. AWS SDKs vereinfachen die Codierung durch die Bereitstellung sprachspezifischer APIs für AWS-Services. Diese SDKs und die Auswahl an Sprachen ermöglichen es Entwicklern, die hier aufgeführten bewährten Methoden zur Gewährleistung von Zuverlässigkeit zu implementieren. Entwickler können sich auch darüber informieren, wie Software von Amazon erstellt und betrieben wird. Die Amazon Builders' Library.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit.

ZUV 3: Wie entwerfen Sie Ihre Workload-Service-Architektur?

Erstellen Sie hoch skalierbare und zuverlässige Workloads mithilfe einer serviceorientierten Architektur (SOA) oder einer Microservices-Architektur. Eine serviceorientierte Architektur (SOA) hat zum Ziel, Softwarekomponenten über Service-Schnittstellen wiederverwendbar zu machen. Die Microservices-Architektur geht noch weiter, um Komponenten kleiner und einfacher zu machen.

ZUV 4: Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?

Verteilte Systeme nutzen Kommunikationsnetzwerke, um Komponenten wie Server oder Services miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder höherer Latenz in diesen Netzwerken zuverlässig ausgeführt werden. Komponenten des verteilten Systems müssen

ZUV 4: Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?

so funktionieren, dass sie keine negativen Auswirkungen auf andere Komponenten oder die Workload haben. Diese bewährten Methoden verhindern Ausfälle und verbessern die mittlere Zeit zwischen Ausfällen (MTBF).

ZUV 5: Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle abgemildert oder bewältigt werden?

Verteilte Systeme nutzen Kommunikationsnetzwerke, um Komponenten (wie Server oder Services) miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder höherer Latenz in diesen Netzwerken zuverlässig ausgeführt werden. Komponenten des verteilten Systems müssen so funktionieren, dass sie keine negativen Auswirkungen auf andere Komponenten oder die Workload haben. Mit den folgenden bewährten Methoden können Workloads Belastungen oder Ausfällen standhalten, schneller wiederhergestellt werden und die Auswirkungen solcher Beeinträc htigungen verringern. Das Ergebnis ist eine verbesserte mittlere Reparaturzeit (MTTR).

Änderungsverwaltung

Änderungen an Ihrem Workload oder der Umgebung müssen vorausgesehen und berücksichtigt werden, um einen zuverlässigen Betrieb der Workload zu erreichen. Zu diesen Änderungen gehören durch äußere Faktoren hervorgerufene Änderungen (z. B. Bedarfsspitzen) sowie interne Änderungen wie Funktionsbereitstellungen und Sicherheitspatches.

Mit AWS können Sie das Verhalten eines Workloads überwachen und die Reaktion auf KPIs automatisieren. Beispielsweise kann die Workload bei einer zunehmenden Zahl von Benutzern zusätzliche Server hinzufügen. Sie können kontrollieren und steuern, welche Benutzer Änderungen an der Workload vornehmen dürfen, und die Historie dieser Änderungen überprüfen.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit.

ZUV 6: Was ist bei der Überwachung von Workload-Ressourcen zu beachten?

Protokolle und Metriken sind wertvolle Tools, um einen Einblick in den Zustand Ihrer Workloads zu gewinnen. Sie können Ihre Workload so konfigurieren, dass Protokolle und Metriken überwacht

ZUV 6: Was ist bei der Überwachung von Workload-Ressourcen zu beachten?

und bei Über- oder Unterschreiten von Schwellenwerten oder wichtigen Ereignissen Benachrichtigungen gesendet werden. Dank der Überwachung kann die Workload erkennen, wenn Schwellenwerte für eine niedrige Leistung unterschritten werden oder Ausfälle auftreten, sodass als Reaktion drauf eine automatische Wiederherstellung erfolgen kann.

ZUV 7: Wie lässt sich der Workload so gestalten, dass er sich an Bedarfsänderungen anpasst?

Eine skalierbare Workload bietet die Elastizität, Ressourcen automatisch entsprechend dem aktuellen Bedarf hinzuzufügen oder zu entfernen.

ZUV 8: Wie implementieren Sie Änderungen?

Kontrollierte Änderungen sind erforderlich, um neue Funktionen bereitzustellen und um sicherzustellen, dass die Workloads und die Betriebsumgebung bekannte Software ausführen und auf vorhersagbare Weise durch Patches aktualisiert oder ersetzt werden können. Wenn diese Änderungen nicht kontrolliert stattfinden, ist es schwierig, ihre Auswirkungen vorherzusagen oder daraus entstehende Probleme zu beheben.

Wenn Sie eine Workload so gestalten, dass Ressourcen als Reaktion auf Bedarfsänderungen automatisch hinzugefügt und entfernt werden, erhöht das nicht nur die Zuverlässigkeit. Vielmehr sorgt diese Vorgehensweise auch dafür, dass geschäftlicher Erfolg nicht zu einer Belastung wird. Bei einer vorhandenen Überwachung wird Ihr Team automatisch benachrichtigt, wenn KPIs von erwarteten Normen abweichen. Mit dem automatischen Protokollieren von Änderungen an Ihrer Umgebung können Sie auf Aktionen prüfen, die sich möglicherweise auf die Zuverlässigkeit ausgewirkt haben, und diese schnell identifizieren. Mit der Kontrolle und Steuerung des Änderungsmanagements können Sie die Regeln durchsetzen, die für die benötigte Zuverlässigkeit sorgen.

Fehlerverwaltung

In Systemen mit großer Komplexität ist es wahrscheinlich, dass Fehler auftreten. Zur Gewährleistung von Zuverlässigkeit muss Ihr Workload auftretende Fehler erkennen und Maßnahmen ergreifen, um Auswirkungen auf die Verfügbarkeit zu vermeiden. Workloads müssen Ausfälle verkraften sowie Probleme automatisch beheben können.

Mit AWS können Sie automatisch auf überwachte Daten reagieren. Wenn eine bestimmte Kennzahl beispielsweise einen Schwellenwert überschreitet, können Sie eine automatische Maßnahme zur Behebung dieses Problems auslösen. Statt also zu versuchen, eine fehlerhafte Ressource, die Teil Ihrer Produktionsumgebung ist, zu diagnostizieren und zu reparieren, können Sie sie durch eine neue Ressource ersetzen und die Analyse der fehlerhaften Ressource extern vornehmen. Da Sie in der Cloud temporäre Versionen eines gesamten Systems zu geringen Kosten aufstellen können, können Sie automatisiertes Testen verwenden, um vollständige Wiederherstellungsprozesse zu überprüfen.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit.

ZUV 9: Was ist bei der Sicherung von Daten zu beachten?

Sichern Sie Daten, Anwendungen und Konfigurationen, um die Anforderungen im Hinblick auf das Recovery Time Objective (RTO, Wiederherstellungsdauer) und das Recovery Point Objective (RPO, Wiederherstellungszeitpunkt) zu erfüllen.

ZUV 10: Wie schützen Sie Ihren Workload mithilfe der Fehlerisolierung?

Fehlerisolierte Grenzen beschränken die Auswirkungen eines Ausfalls innerhalb eines Workloads auf eine begrenzte Anzahl von Komponenten. Komponenten außerhalb der Grenze sind vom Ausfall nicht betroffen. Wenn Sie mehrere fehlerisolierte Grenzen verwenden, können Sie die Auswirkungen auf Ihren Workload einschränken.

ZUV 11: Wie lassen sich Workloads so gestalten, dass sie Komponentenausfälle verkraften?

Workloads, für die eine hohe Verfügbarkeit und eine niedrige mittlere Reparaturzeit erforderlich sind, müssen auf Ausfallsicherheit ausgelegt sein.

ZUV 12: Wie lässt sich die Zuverlässigkeit testen?

Nachdem Sie Ihre Workload so konzipiert haben, dass sie den Belastungen der Produktion standhält, sind Tests die einzige Möglichkeit, sie auf die erwartete Funktionalität und Ausfallsi cherheit hin zu testen.

ZUV 13: Was ist bei der Planung der Notfallwiederherstellung zu beachten?

Backups und redundante Workload-Komponenten sind der Ausgangspunkt Ihrer Strategie für die Notfallwiederherstellung. RTO und RPO sind Ihre Ziele für die Wiederherstellung Ihrer Workload. Legen Sie diese Ziele entsprechend den geschäftlichen Anforderungen fest. Implementieren Sie eine Strategie, um diese Ziele zu erreichen. Berücksichtigen Sie dabei Standorte und Funktionen von Workload-Ressourcen und -Daten. Die Wahrscheinlichkeit von Disruptionen und die Kosten von Wiederherstellungen sind ebenfalls wichtige Faktoren bei der Ermittlung des Unternehm enswerts, den Notfallwiederherstellungen von Workloads bieten.

Sichern Sie Ihre Daten regelmäßig und stellen Sie anhand von Tests der Sicherungsdateien sicher, dass Sie Wiederherstellungen nach logischen und physischen Fehlern durchführen können. Ein Schlüssel zur Verwaltung von Fehlern ist das regelmäßige und automatisierte Testen von Workloads, um Ausfälle hervorzurufen, und das anschließende Beobachten des Wiederherstellungsverhaltens. Führen Sie diese Tests regelmäßig durch, auch nach größeren Workload-Änderungen. Verfolgen Sie KPIs aktiv wie auch das Recovery Time Objective (RTO, Wiederherstellungsdauer) und das Recovery Point Objective (RPO, Wiederherstellungszeitpunkt), um die Ausfallsicherheit einer Workload (insbesondere unter Fehlertestszenarios) zu bewerten. Die Verfolgung von KPIs unterstützt Sie bei der Identifizierung und Milderung einzelner Fehlerquellen. Hierbei geht es darum, Ihre Prozesse zur Wiederherstellung von Workloads gründlich zu testen, damit Sie darauf vertrauen können, dass Sie alle Daten wiederherstellen und Ihre Kunden unterbrechungsfrei bedienen können. Und zwar selbst dann, wenn länger anhaltende Probleme auftreten. Mit Ihren Wiederherstellungsprozessen sollten Sie sich genauso vertraut machen wie mit Ihren normalen Produktionsprozessen.

Ressourcen

Werfen Sie einen Blick auf die folgenden Ressourcen, um mehr über unsere bewährten Methoden für die Zuverlässigkeit zu erfahren.

Dokumentation

- AWS-Dokumentation
- Globale AWS-Infrastruktur
- AWS Auto Scaling: Funktionsweise von Skalierungsplänen
- Was ist AWS Backup?

Whitepaper

- Säule „Zuverlässigkeit“: AWS Well-Architected
- Implementieren von Microservices in AWS

Leistungseffizienz

Die Säule ""Leistungseffizienz"" umfasst die Fähigkeit, Rechenressourcen effizient entsprechend den Systemanforderungen zu nutzen und diese Effizienz aufrechtzuerhalten, während sich die Nachfrage ändert und die Technologie weiterentwickelt.

Die Säule der Leistungseffizienz bietet einen Überblick über Designprinzipien, bewährte Methoden und Fragen. Verbindliche Anleitungen zur Implementierung finden Sie im Whitepaper „Säule der Leistungseffizienz“.

Themen

- Designprinzipien
- Definition
- Bewährte Methoden
- Ressourcen

Designprinzipien

Es gibt fünf Designprinzipien für die Leistungseffizienz in der Cloud:

- Demokratisieren fortschrittlicher Technologien: Gestalten Sie die Implementierung fortschrittlicher Technologien für Ihr Team reibungsloser, indem Sie komplexe Aufgaben an Ihren Cloud-Anbieter delegieren. Statt Ihr IT-Team aufzufordern, sich näher über das Hosten und Ausführen einer neuen Technologie zu informieren, sollten Sie die Technologie als Service nutzen. Es gibt Technologien, wie etwa die NoSQL-Datenbanken, das Transcodieren von Medien sowie Machine Learning, die spezielles Fachwissen erfordern. In der Cloud kann Ihr Team diese Technologien als Service nutzen und sich auf die Produktentwicklung konzentrieren, ohne sich um die Bereitstellung und Verwaltung von Ressourcen kümmern zu müssen.
- Globale Reichweite innerhalb von Minuten: Durch die Bereitstellung Ihrer Workload in mehreren AWS-Regionen auf der ganzen Welt können Sie Ihren Kunden geringere Latenz und eine bessere Erfahrung bei minimalen Kosten bieten.

- Nutzung von Serverless-Architekturen: Aufgrund der in der Cloud verwendeten Serverless-Architekturen brauchen Sie für herkömmliche Datenverarbeitungsaktivitäten keine physischen Server mehr auszuführen und zu verwalten. Serverless- Speicherservices können beispielsweise als statische Websites genutzt werden, wodurch sich Webserver erübrigen. Ihren Code können Sie von Ereignisservices hosten lassen. Auf diese Weise entfällt nicht nur die Verwaltung physischer Server, sondern auch die Transaktionskosten sinken, da verwaltete Services in der Cloud-Umgebung ausgeführt werden.
- Vermehrtes Experimentieren: Mit virtuellen und automatisierbaren Ressourcen können Sie schnell unterschiedliche Konfigurationen, Instance- oder Speichertypen miteinander vergleichen.
- Aufbringen von technischem Verständnis: Befassen Sie sich mit der Verwendungsweise von Cloud-Services und nutzen Sie stets den Technologieansatz, der für Ihre Workload-Ziele geeignet ist. Berücksichtigen Sie bei der Auswahl des passenden Datenbank- oder Speicherkonzepts beispielsweise die Datenzugriffsmuster.

Definition

Es gibt fünf bewährte Methoden für die Leistungseffizienz in der Cloud:

- Auswahl der Architektur
- Computer und Hardware
- Datenverwaltung
- Netzwerk und Bereitstellung von Inhalten
- Prozess und Kultur

Um eine leistungsstarke Architektur sicherzustellen, empfiehlt sich für deren Entwicklung ein datenbasierter Ansatz. Sammeln Sie zu allen Aspekten der Architektur Daten, angefangen vom allgemeinen Design bis hin zur Auswahl und Konfiguration der Ressourcentypen.

Durch regelmäßiges Überprüfen Ihrer Auswahl stellen Sie die bestmögliche Nutzung der sich fortlaufend weiterentwickelnden AWS Cloud sicher. Durch Überwachung erkennen Sie Abweichungen von der erwarteten Leistung. Zur Leistungssteigerung der Architektur können Sie auch Kompromisse eingehen, beispielsweise durch Komprimierung oder Caching, oder indem Sie hinsichtlich der Konsistenz mehr Toleranz einräumen.

Bewährte Methoden

Themen

- Auswahl der Architektur
- Computer und Hardware
- Datenverwaltung
- Netzwerk und Bereitstellung von Inhalten
- Prozess und Kultur

Auswahl der Architektur

Die optimale Lösung für einen bestimmten Workload variiert und Lösungen bestehen häufig aus einer Kombination mehrerer Ansätze. Well-Architected-Workloads nutzen mehrere Lösungen und ermöglichen verschiedene Funktionen zur Verbesserung der Leistung.

AWS-Ressourcen sind in vielen Typen und Konfigurationen verfügbar, wodurch es einfacher ist, einen Ansatz zu finden, der Ihren Anforderungen weitgehend entspricht. Sie können zudem Optionen nutzen, die sich in Ihrer On-Premises-Infrastruktur nicht ohne Weiteres umsetzen ließen. Nehmen wir beispielsweise den verwalteten Service Amazon DynamoDB. Dieser bietet eine vollständig verwaltete NoSQL-Datenbank mit einer Latenz im einstelligen Millisekundenbereich ungeachtet des Volumens.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz. (Eine Liste der Fragen und bewährten Methoden zur Leistungseffizienz finden Sie im Appendix.)

PERF 1: How do you select appropriate cloud resources and architecture patterns for your workload?
Often, multiple approaches are required for more effective performance across a workload. Well-Architected systems use multiple solutions and features to improve performance.

Computer und Hardware

Die optimale Datenverarbeitungsoption für einen bestimmten Workload kann sich je nach Anwendungsdesign, Nutzungsmustern und Konfigurationseinstellungen unterscheiden. Architekturen können verschiedene Computing-Optionen für verschiedene Komponenten verwenden

und verschiedene Funktionen zur Verbesserung der Leistung bieten. Die Wahl der falschen Datenverarbeitungslösung für eine Architektur kann die Leistungseffizienz schmälern.

In AWS gibt es drei Arten der Datenverarbeitung: Instances, Container und Funktionen.

- Instances sind virtualisierte Server, deren Funktionen mit einer Schaltfläche oder einem API-Aufruf geändert werden können. Da Ressourcenentscheidungen in der Cloud flexibel sind, können Sie mit verschiedenen Servertypen experimentieren. AWS bietet diese virtuellen Server-Instances in unterschiedlichen Varianten und Größen mit einer umfassenden Auswahl an Optionen, einschließlich Solid-State-Laufwerken (SSDs) und Grafikprozessoren (GPUs).
- Container dienen zur Virtualisierung des Betriebssystems. Sie können damit eine Anwendung und deren Abhängigkeiten in von der Ressource isolierten Prozessen ausführen. AWS Fargate bietet eine Serverless-Datenverarbeitung für Container. Amazon EC2 kann verwendet werden, wenn Sie Kontrolle über die Installation, Konfiguration und Verwaltung Ihrer Datenverarbeitungsumgebung benötigen. Zudem haben Sie die Auswahl unter mehreren Plattformen zur Container-Orchestrierung: Amazon Elastic Container Service (ECS) oder Amazon Elastic Kubernetes Service (EKS).
- Funktionen abstrahieren die Ausführungsumgebung vom Code, den Sie anwenden möchten. AWS Lambda ermöglicht es Ihnen beispielsweise, Code auszuführen, ohne eine Instance auszuführen.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF 2: How do you select and use compute resources in your workload?

The more efficient compute solution for a workload varies based on application design, usage patterns, and configuration settings. Architectures can use different compute solutions for various components and turn on different features to improve performance. Selecting the wrong compute solution for an architecture can lead to lower performance efficiency.

Datenverwaltung

Die optimale Datenverwaltungslösung für ein bestimmtes System hängt vom Datentyp (Block, Datei oder Objekt), den Zugriffsmustern (zufällig oder sequenziell), dem erforderlichen Durchsatz, der Zugriffshäufigkeit (online, offline, Archiv), der Aktualisierungshäufigkeit (WORM, dynamisch) sowie den Verfügbarkeits- und Lebensdaueranforderungen ab. Well-Architected-Workloads verwenden

zweckgebundene Datenspeicher, die verschiedene Funktionen zur Verbesserung der Leistung ermöglichen.

In AWS ist Speicher in drei Formen verfügbar: Objekt-, Block- und Dateispeicher:

- Objektspeicher bietet eine skalierbare, robuste Plattform, damit Daten überall im Internet zugänglich sind. Das gilt für benutzergenerierte Inhalte, aktive Archive, Serverless-Datenverarbeitung, Big-Data-Speicher oder die Sicherung und Wiederherstellung. Bei Amazon Simple Storage Service (Amazon S3) handelt es sich um einen Objektspeicherservice mit branchenführender Skalierbarkeit, Datenverfügbarkeit, Sicherheit und Leistung. Amazon S3 ist auf eine Verfügbarkeit von 99,999999999 % (elf Neunen) ausgelegt und speichert Daten für Millionen von Anwendungen für Unternehmen weltweit.
- Blockspeicher bietet hochverfügbaren, konsistenten Blockspeicher mit geringer Latenz für virtuelle Hosts. Er ist vergleichbar mit Direct Attached Storage (DAS) oder einem Storage Area Network (SAN). Amazon Elastic Block Store (Amazon EBS) ist auf Workloads ausgelegt, die einen persistenten, für EC2-Instances zugänglichen Speicher benötigen. So können Sie Anwendungen in Sachen Speicherkapazität, Leistung und Kosten optimieren.
- Dateispeicher bietet auf mehreren Systemen Zugriff auf ein gemeinsam genutztes Dateisystem. Dateispeicherlösungen wie Amazon Elastic File System (Amazon EFS) eignen sich ideal für Anwendungsfälle wie große Inhalts-Repositorys, Entwicklungsumgebungen, Medienspeicher oder Hauptverzeichnisse von Benutzern. Amazon FSx macht das Starten und Ausführen beliebter Dateisysteme einfach und kostengünstig. Somit können Sie die umfangreichen Funktionen und die hohe Leistung weit verbreiteter Open-Source- und kommerzieller Dateisysteme nutzen.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF 3: How do you store, manage, and access data in your workload?

The more efficient storage solution for a system varies based on the kind of access operation (block, file, or object), patterns of access (random or sequential), required throughput, frequency of access (online, offline, archival), frequency of update (WORM, dynamic), and availability and durability constraints. Well-architected systems use multiple storage solutions and turn on different features to improve performance and use resources efficiently.

Netzwerk und Bereitstellung von Inhalten

Welche Networking-Lösung für einen Workload optimal ist, richtet sich nach der Latenz, dem erforderlichen Durchsatz, dem Jitter und der Bandbreite. Die Standortoptionen sind von den physischen Einschränkungen abhängig, z. B. von Benutzer- oder On-Premises-Ressourcen. Diese Einschränkungen können durch Edge-Standorte oder die Ressourcenplatzierung wettgemacht werden.

In AWS wird das Netzwerk virtualisiert und es sind unterschiedliche Typen und Konfigurationen verfügbar. Das erleichtert Ihnen die Anpassung Ihrer Networking-Anforderungen. AWS bietet zur Optimierung des Netzwerkdatenverkehrs Produktfunktionen wie Enhanced Networking, für Amazon EC2-Networking optimierte Instances, Amazon S3 Transfer Acceleration sowie den dynamischen Amazon CloudFront-Service. Zur Verbesserung der Latenz und der Stabilität des Netzwerks finden Sie in AWS zudem Networking-Funktionen wie die latenzbasierte Weiterleitung mit Amazon Route 53, Amazon VPC-Endpunkte, AWS Direct Connect und AWS Global Accelerator.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF 4: How do you select and configure networking resources in your workload?
This question includes guidance and best practices to design, configure, and operate efficient networking and content delivery solutions in the cloud.

Prozess und Kultur

Bei der Architektur von Workloads gibt es Prinzipien und Praktiken, die Sie übernehmen können, um effiziente und leistungsstarke Cloud-Workloads besser zu betreiben. Um eine Kultur zu schaffen, die die Leistungseffizienz von Cloud-Workloads fördert, sollten Sie diese Schlüsselprinzipien und -praktiken berücksichtigen:

Beachten Sie beim Aufbau dieser Kultur die folgenden Schlüsselprinzipien:

- Infrastruktur als Code: Definieren Sie Ihre Infrastruktur beispielsweise mithilfe von AWS CloudFormation-Vorlagen als Code. Mit Vorlagen können Sie Ihre Infrastruktur zusammen mit Ihrem Anwendungscode und Ihren Konfigurationen per Quellcodeüberwachung verwalten. Dies ermöglicht es Ihnen, dieselben Verfahren wie bei der Softwareentwicklung auch auf Ihre Infrastruktur anzuwenden, um von einer schnellen Iteration zu profitieren.

- Bereitstellungspipeline: Nutzen Sie zur Bereitstellung der Infrastruktur eine CI/CD-Pipeline (Continuous Integration/Continuous Deployment) wie etwa ein Quellcode-Repository, Build-Systeme sowie automatisierte Bereitstellungs- und Testverfahren. Dies lässt eine reproduzierbare, konsistente und kostengünstige Iteration zu.
- Gut definierte Metriken: Richten Sie Metriken so ein, dass wichtige Leistungskennzahlen (KPIs) erfasst werden, und überwachen Sie sie entsprechend. Wir empfehlen die Verwendung technischer und geschäftlicher Metriken. Anhand von wichtigen Metriken für Websites oder mobile Apps wird die Zeit bis zum ersten Byte oder Rendering erfasst. Zu den weiteren allgemein anwendbaren Metriken zählen die Thread-Anzahl, die Garbage Collection-Rate sowie Wartezustände. Anhand von geschäftlichen Kennzahlen wie den aggregierten kumulativen Kosten pro Anfrage können Sie Möglichkeiten zur Kostensenkung ermitteln. Erwägen Sie sorgfältig, wie Kennzahlen interpretiert werden sollen. Sie können beispielsweise anstelle von Durchschnittswerten Maximalwerte oder das 99. Perzentil wählen.
- Automatische Leistungstests: Sorgen Sie im Rahmen der Bereitstellung dafür, dass nach dem erfolgreichen Absolvieren der schnelleren Ausführungstests automatisch Leistungstests gestartet werden. Durch die Automatisierung sollte eine neue Umgebung mit entsprechenden Anfangsbedingungen, z. B. Testdaten, entstehen, in der anschließend einige Benchmark- und Lasttests ausgeführt werden. Die Ergebnisse dieser Tests sollten mit dem Build in Verbindung gebracht werden, um Leistungsänderungen verfolgen zu können. Für langwierige Tests können Sie diesen Teil der Pipeline gegenüber dem restlichen Build asynchron ausführen. Sie haben auch die Möglichkeit, Leistungstests über Nacht mit Amazon EC2 Spot Instances auszuführen.
- Lastgenerierung: Erstellen Sie eine Reihe von Testskripts zum Replizieren synthetischer oder vorab aufgezeichneter Benutzerreisen. Diese Skripts sollten idempotent und nicht gekoppelt sein. Um gültige Ergebnisse zu erzielen, sind möglicherweise zusätzliche vorbereitende Skripts erforderlich. Die Testskripts sollten das Nutzungsverhalten in der Produktion möglichst authentisch replizieren. Zur Lastgenerierung können Sie Software- oder Software-as-a-Service-Lösungen (SaaS) verwenden. Erwägen Sie die Verwendung von AWS Marketplace-Lösungen und Spot Instances. Dies können kostengünstige Ansätze zum Generieren der Last sein.
- Leistungstransparenz: Wichtige Metriken sollten für das ganze Team sichtbar sein. Dies gilt insbesondere für die Metriken der einzelnen Build-Versionen. Damit lassen sich wichtige positive oder negative Trends erkennen. Wichtig sind auch Metriken zur Anzahl der Fehler oder Ausnahmen, um sicherzustellen, dass das System funktioniert.
- Visualisierung: Nutzen Sie Visualisierungstechniken, mit denen Leistungsprobleme, Hotspots, Wartezustände oder niedrige Auslastungen klar aufgezeigt werden. Zeigen Sie Leistungsmetriken

in Architekturdiagrammen an. Aufrufgrafiken oder Code können die Problemerkennung beschleunigen.

- Regelmäßiger Überprüfungsvorgang: Wenn Architekturen eine schlechte Leistung aufweisen, liegt dies normalerweise daran, dass ein Prozess zu Überprüfung der Leistung fehlt oder fehlerhaft ist. Falls Sie derartige Probleme mit Ihrer Architektur haben, können Sie jederzeit ein Leistungsprüfverfahren implementieren und somit iterative Verbesserungen fördern.
- Kontinuierliche Optimierung: Schaffen Sie eine Kultur fortlaufender Optimierung der Leistungseffizienz Ihrer Cloud-Workloads.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF 5: What process do you use to support more performance efficiency for your workload?

When architecting workloads, there are principles and practices that you can adopt to help you better run efficient high-performing cloud workloads. To adopt a culture that fosters performance efficiency of cloud workloads, consider these key principles and practices.

Ressourcen

Weitere Informationen zu bewährten Methoden für die Leistungseffizienz finden Sie in den folgenden Ressourcen.

Dokumentation

- Amazon S3-Leistungsoptimierung
- Amazon EBS-Volume-Leistung

Whitepaper

- Säule „Leistungseffizienz“

Video

- AWS re:Invent 2019: Amazon EC2-Grundlagen (CMP211-R2)
- AWS re:Invent 2019: Leadership Session: Der aktuelle Speicherstatus (STG201-L)

- AWS re:Invent 2019: Leadership Session: Speziell entwickelte Datenbanken von AWS (DAT209-L)
- AWS re:Invent 2019: Konnektivität mit AWS und Hybrid-AWS-Netzwerkarchitekturen (NET317-R1)
- AWS re:Invent 2019: Amazon EC2 der neuesten Generation: Ausführliche Beschreibung des Nitro-Systems (CMP303-R2)
- AWS re:Invent 2019: Erweitern Sie den Umfang auf Ihre ersten 10 Millionen Benutzer (ARC211-R)

Kostenoptimierung

Die Säule Kostenoptimierung umfasst die Fähigkeit, Systeme so auszuführen, dass sie geschäftlichen Wert bei geringstmöglichen Kosten liefern.

Die Säule der Kostenoptimierung bietet einen Überblick über Designprinzipien, bewährte Methoden und Fragen. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule der Kostenoptimierung.

Themen

- Designprinzipien
- Definition
- Bewährte Methoden
- Ressourcen

Designprinzipien

Es gibt fünf Designprinzipien für die Kostenoptimierung in der Cloud:

- Implementieren des Cloud-Finanzmanagements: Um finanziellen Erfolg zu haben und die Wertschöpfung in der Cloud zu beschleunigen, müssen Sie in Cloud-Finanzmanagement/Kostenoptimierung investieren. Ihr Unternehmen muss Zeit und Ressourcen aufwenden, um Know-how in diesem neuen Bereich des Technologie- und Nutzungsmanagements aufzubauen. Wie bei Ihren Funktionen zur Sicherheit oder betriebliche Exzellenz müssen Sie Fähigkeiten durch Wissensaufbau, Programme, Ressourcen und Prozesse aufbauen, damit Sie zu einer kosteneffizienten Organisation werden können.
- Verbrauchsmodell einführen: Zahlen Sie nur für die benötigten Computing-Ressourcen, und erhöhen oder verringern Sie die Nutzung auf Basis Ihrer Geschäftsanforderungen und nicht durch aufwändige Prognosen. Entwicklungs- und Testumgebungen werden in einer normalen

Arbeitswoche beispielsweise nur acht Stunden pro Tag benötigt. Sie können diese Ressourcen anhalten, wenn sie nicht verwendet werden und damit potenzielle Einsparungen von 75 % (40 Stunden vs. 168 Stunden) erzielen.

- Gesamteffizienz messen: Messen Sie die geschäftliche Leistung des Workloads und die mit der Bereitstellung verknüpften Kosten. Verwenden Sie diese Kennzahlen, um die Gewinne zu ermitteln, die Sie durch die Erhöhung der Leistung und die Reduzierung der Kosten erzielen.
- Kein Geld mehr für undifferenzierte, aufwendige Arbeiten ausgeben: AWS erledigt die aufwendigsten Arbeiten im Rechenzentrum bezüglich Server-Racks, -Stacks und -Stromversorgung. Außerdem entfällt der betriebliche Aufwand für die Verwaltung von Betriebssystemen und Anwendungen mit verwalteten Services. So können Sie sich auf Ihre Kunden und Geschäftsprojekte anstatt auf die IT-Infrastruktur konzentrieren.
- Ausgaben analysieren und zuordnen: Mit der Cloud ist es einfacher, die Nutzung und die Kosten von Systemen genau zu ermitteln und auf Basis dieser Daten eine transparente Zuordnung der IT-Kosten auf einzelne Workload-Besitzer durchzuführen. Auf diese Weise erhalten Sie Unterstützung bei der Messung der Umsatzrendite (ROI) und Workload-Eigentümer erhalten die Möglichkeit, ihre Ressourcen zu optimieren und die Kosten zu reduzieren.

Definition

Es gibt fünf bewährte Methoden für die Kostenoptimierung in der Cloud:

- Praxis für Cloud-Finanzmanagement
- Ausgabenerkennung und Nutzungsbewusstsein
- Kostengünstige Ressourcen
- Verwaltung von Nachfrage und Bereitstellung von Ressourcen
- Optimierung im Laufe der Zeit

Wie bei den anderen Säulen innerhalb des Well-Architected Framework sind Kompromisse unvermeidbar, so müssen Sie beispielsweise entscheiden, ob Sie die Markteinführungsgeschwindigkeit oder die Kosten optimieren möchten. In manchen Fällen ist es sinnvoll, die Priorität auf Geschwindigkeit zu legen, z. B. verbunden mit einer raschen Markteinführung, der Bereitstellung neuer Funktionen oder einer simplen Fristerfüllung, statt im Vorfeld in Kostenoptimierung zu investieren. Konzeptionelle Entscheidungen werden gelegentlich durch Eile statt auf Basis von Daten getroffen, und man ist immer der Versuchung ausgesetzt, einem potenziellen Szenario zu viel Bedeutung beizumessen, statt Zeit in die Bestimmung der

kostengünstigsten Bereitstellung zu investieren. Dies führt häufig übermäßigen und mangelhaft optimierten Bereitstellungen. Es ist jedoch die richtige Wahl, wenn Sie Ressourcen aus Ihrer lokalen Umgebung in die Cloud verlagern und die Optimierung anschließend durchführen möchten. Wenn Sie vorab genügend Arbeit in eine Strategie zur Kostenoptimierung investieren, können Sie die wirtschaftlichen Vorteile der Cloud schneller nutzen, indem Sie eine konsistente Einhaltung bewährter Methoden sicherstellen und Überbereitstellungen vermeiden. In den folgenden Abschnitten finden Sie Techniken und bewährte Methoden sowohl für die erste als auch die fortlaufende Implementierung von Cloud-Finanzmanagement und Kostenoptimierung für Ihre Workloads.

Bewährte Methoden

Themen

- Praxis für Cloud-Finanzmanagement
- Ausgabenerkennung und Nutzungsbewusstsein
- Kostengünstige Ressourcen
- Verwaltung von Nachfrage und Bereitstellung von Ressourcen
- Optimierung im Laufe der Zeit

Praxis für Cloud-Finanzmanagement

Mit der Einführung der Cloud können Technologieteams dank verkürzter Genehmigungs-, Beschaffungs- und Infrastrukturbereitstellungszyklen schneller innovieren. Ein neuer Ansatz für das Finanzmanagement in der Cloud ist erforderlich, um geschäftlichen Nutzen und finanziellen Erfolg zu erzielen. Dieser Ansatz ist das Cloud-Finanzmanagement. Es baut Funktionen in Ihrer gesamten Organisation auf, indem organisationsweit Wissensaufbau, Programme, Ressourcen und Prozesse implementiert werden.

Viele Organisationen bestehen aus vielen verschiedenen Einheiten mit unterschiedlichen Prioritäten. Durch die Fähigkeit, Ihre Organisation an mehreren vereinbarten Finanzzielen auszurichten und ihr die Mechanismen zur Erreichung der Ziele bereitzustellen, wird die Effizienz der Organisation gesteigert. Ein leistungsfähiges Unternehmen innoviert und entwickelt schneller, ist agiler und passt sich einfacher an beliebige interne oder externe Faktoren an.

In AWS können Sie Cost Explorer und optional Amazon Athena und Amazon QuickSight mit dem Kosten- und Nutzungsbericht (Cost and Usage Report, CUR) verwenden. So können Sie in Ihrer gesamten Organisation ein Kosten- und Nutzungsbewusstsein schaffen. AWS-Budgets bietet

proaktive Benachrichtigungen zu Kosten und Nutzung. Die AWS-Blogs bieten Informationen zu neuen Services und Funktionen, damit Sie immer über neue Serviceversionen auf dem Laufenden sind.

In der folgenden Frage geht es um Überlegungen zur Kostenoptimierung. (Eine Liste der Fragen und bewährten Methoden zur Kostenoptimierung finden Sie im Anhang).

KOSTEN 1: Wie implementieren Sie das Cloud Financial Management?

Die Implementierung von Cloud Financial Management (CFM) ermöglicht es Unternehmen, geschäftlichen Nutzen und finanziellen Erfolg zu erzielen, wenn sie ihre Kosten und Nutzung optimieren und auf AWS skalieren.

Beim Aufbau einer Kostenoptimierungsfunktion sollten Sie Teammitglieder einsetzen und das Team um Experten für CFM und Kostenoptimierung ergänzen. Bestehende Teammitglieder wissen, wie die Organisation derzeit funktioniert und Verbesserungen schnell implementiert werden können. Erwägen Sie auch, Personen mit ergänzenden oder speziellen Kenntnissen, wie im Bereich Analyse oder Projektmanagement, mit einzubinden.

Wenn Sie in Ihrer Organisation ein Kostenbewusstsein implementieren, verbessern Sie vorhandene Programme oder bauen auf diesen auf. Es geht viel schneller, bestehende Prozesse und Programme zu ergänzen, als sie neu zu erstellen. So werden die Ergebnisse viel schneller erreicht.

Ausgabenerkennung und Nutzungsbewusstsein

Die erhöhte Flexibilität und Agilität der Cloud fördert Innovationen und schnelle Entwicklungen und Bereitstellungen. Diese Merkmale eliminieren die manuellen Prozesse und den Zeitaufwand für die Bereitstellung einer lokalen Infrastruktur, einschließlich der Identifizierung von Hardware-Spezifikationen, dem Verhandeln von Preisen, der Verwaltung von Bestellungen, der Planung von Lieferungen und schließlich der Bereitstellung der Ressourcen. Die einfache Nutzung und die nahezu unbegrenzte On-Demand-Verfügbarkeit macht neue Wege erforderlich, über Ausgaben nachzudenken.

Viele Unternehmen bestehen aus einer Vielzahl von Systemen, die von unterschiedlichen Teams betrieben werden. Die Möglichkeit, die Ressourcenkosten der jeweiligen Organisation oder den jeweiligen Produkteigentümer zuzuordnen, fördert ein effizientes Nutzungsverhalten und hilft, Verschwendung von Ressourcen einzudämmen. Mit einer präzisen Kostenzuordnung wissen Sie,

welche Produkte wirklich profitabel sind, und können fundiertere Entscheidungen in Bezug auf die Budgetaufteilung treffen.

In AWS erstellen Sie mit AWS Organizations oder AWS Control Tower eine Kontostruktur, die eine Trennung ermöglicht und Sie bei der Zuordnung Ihrer Kosten und Nutzung unterstützt. Sie können auch das Ressourcen-Tagging verwenden, um Geschäfts- und Organisationsinformationen auf Ihre Nutzung und Kosten anzuwenden. Verwenden Sie AWS Cost Explorer, um Einblicke in Ihre Kosten und Nutzung zu erhalten, oder erstellen Sie benutzerdefinierte Dashboards und Analysen mit Amazon Athena und Amazon QuickSight. Die Kontrolle Ihrer Kosten und Nutzung erfolgt durch Benachrichtigungen über AWS-Budgets sowie Kontrollen mithilfe von AWS Identity and Access Management (IAM) und Service Quotas.

In den folgenden Fragen geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 2: Wie können Sie die Nutzung steuern?
Definieren Sie Richtlinien und Verfahren, um sicherzustellen, dass sich die Kosten auf dem Weg zur Erreichung Ihrer Ziele in einem angemessenen Rahmen bewegen. Durch den Einsatz eines Kontrollsystems können Sie Innovationen vorantreiben, ohne das Budget zu überschreiten.

KOSTEN 3: Wie können Sie die Nutzung und Kosten überwachen?
Definieren Sie Richtlinien und Verfahren, um Ihre Kosten überwachen und richtig zuordnen zu können. Dadurch können Sie die Kosteneffizienz des Workloads bewerten und verbessern.

KOSTEN 4: Wie können Sie Ressourcen außer Betrieb nehmen?
Implementieren Sie vom Beginn bis zum Abschluss eines Projekts eine Änderungskontrolle und Ressourcenverwaltung. Auf diese Weise können Sie ungenutzte Ressourcen herunterfahren oder beenden, um Verschwendungen zu minimieren.

Sie können Tags für die Kostenzuordnung verwenden, um Ihre Nutzung und Kosten in AWS zu kategorisieren und zu verfolgen. Wenn Sie Tags auf Ihre AWS-Ressourcen anwenden (z. B. EC2-Instances oder S3-Buckets), generiert AWS einen Kosten- und Nutzungsbericht mit Ihrer

Nutzung und Ihren Tags. Sie können Tags anwenden, die für Unternehmenskategorien stehen (z. B. Kostenstellen, Workload-Namen oder Besitzer), um Ihre Kosten verschiedenen Services zuzuordnen.

Achten Sie darauf, dass Sie den richtigen Detail- und Granularitätsgrad für die Kosten- und Nutzungsberichterstattung und -überwachung verwenden. Um allgemeine Erkenntnisse zu gewinnen und Trends zu erkennen, verwenden Sie die tägliche Granularität mit AWS Cost Explorer. Für tiefgehendere Analysen und Prüfungen verwenden Sie die stündliche Granularität in AWS Cost Explorer oder Amazon Athena und Amazon QuickSight sowie den Kosten- und Nutzungsbericht (CUR) mit stündlicher Granularität.

Durch die Kombination von mit Tags gekennzeichneten Ressourcen und Entitätslebenszyklus-Tracking (Mitarbeiter, Projekte) können Sie verwaiste Ressourcen oder Projekte identifizieren, die für das Unternehmen keinen Wert mehr generieren und außer Betrieb genommen werden sollten. Sie können Abrechnungsbenachrichtigungen einrichten, um Sie über prognostizierte Budgetüberschreitungen zu informieren.

Kostengünstige Ressourcen

Die Verwendung geeigneter Instances und Ressourcen für Ihren Workload ist für Kosteneinsparungen von entscheidender Bedeutung. Die Ausführung eines Berichtsprozesses kann auf kleineren Servern beispielsweise bis zu fünf Stunden dauern, auf einem doppelt so teuren großen Server jedoch lediglich eine Stunde. Auf beiden Servern erhalten Sie dasselbe Ergebnis, der kleinere Server generiert über den Ausführungszeitraum jedoch höhere Kosten.

Architektonisch gute Workloads verwenden die kostengünstigsten Ressourcen; dieses Verhalten kann eine signifikante und positive wirtschaftliche Auswirkung haben. Sie haben außerdem die Möglichkeit, verwaltete Services für die Kostenreduzierung zu verwenden. So können Sie für die E-Mail-Zustellung beispielsweise einen Service nutzen, bei dem die Kosten nach der Anzahl der versendeten Nachrichten berechnet werden, statt Server für diese Aufgabe bereithalten zu müssen.

AWS bietet eine Vielzahl flexibler und kosteneffektiver Preisoptionen für den Erwerb von Instances von Amazon EC2 und anderen Services auf eine Weise, die Ihre Anforderungen ideal erfüllt. On demand Instances zahlen Sie auf Stundenbasis für die genutzte Rechenkapazität und gehen keine Mindestverpflichtungen ein. Savings Plans und Reserved Instances bieten Einsparungen von bis zu 75 % gegenüber On-Demand-Preisen. Mit Spot-Instances können Sie ungenutzte Amazon EC2-Kapazität nutzen und von Einsparungen von bis zu 90 % im Vergleich zum On-Demand-Preis profitieren. Spot Instances eignen sich, wenn das System eine Flotte von Servern toleriert, bei der einzelne Server dynamisch aktiviert und deaktiviert werden können, wie z. B. bei zustandslosen Webservern, bei der Stapelverarbeitung oder bei der Nutzung von HPC und Big Data.

Auch mit der Auswahl geeigneter Services ist es möglich, Nutzung und Kosten zu reduzieren. So können Sie beispielsweise CloudFront nutzen, um das Datenübertragungsvolumen zu reduzieren, oder Kosten vollständig eliminieren, z. B. mit Amazon Aurora on RDS, mit dem Sie kostspielige Datenbanklizensierungskosten vermeiden können.

In den folgenden Fragen geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 5: Wie können Sie die Kosten bei der Auswahl von Services einschätzen?

Bei Amazon EC2, Amazon EBS und Amazon S3 handelt es sich um AWS-Services, die als einzelne Bausteine angeboten werden. Verwaltete Services, etwa Amazon RDS und Amazon DynamoDB, sind AWS-Services auf einer höheren Ebene oder Anwendungsebene. Wenn Sie sich für die richtigen Bausteine und verwalteten Services entscheiden, können Sie die Kosten dieses Workloads optimieren. Durch die Nutzung von verwalteten Services können Sie einen Großteil Ihres administrativen und betrieblichen Overheads reduzieren oder beseitigen und damit Kapazität en für anwendungs- und geschäftsbezogene Aktivitäten gewinnen.

KOSTEN 6: Wie können Sie bei der Auswahl des Ressourcentyps, -umfangs und der Anzahl der Ressourcen Kostenziele erfüllen?

Stellen Sie sicher, dass Sie den geeigneten Ressourcenumfang und die Anzahl der Ressourcen für die jeweilige Aufgabe auswählen. Durch die Auswahl des kostengünstigsten Typs, Umfangs und der kostengünstigsten Anzahl minimieren Sie die Verschwendung von Ressourcen.

KOSTEN 7: Wie können Sie Kosten mithilfe von Preismodellen senken?

Verwenden Sie das Preismodell, das sich für Ihre Ressourcen am besten eignet. So halten Sie die Ausgaben möglichst niedrig.

KOSTEN 8: Wie können Sie die Kosten für Datenübertragungen planen?

Damit Sie architekturbezogene Entscheidungen zur Kostenminimierung treffen können, müssen Sie unbedingt die Datenübertragungskosten einplanen und überwachen. Eine geringfügige, aber

KOSTEN 8: Wie können Sie die Kosten für Datenübertragungen planen?

effektive Änderung an der Architektur kann Ihre Betriebskosten über einen längeren Zeitraum hinweg erheblich senken.

Durch das Einkalkulieren der Kosten während der Serviceauswahl und die Verwendung von Tools wie Cost Explorer und AWS Trusted Advisor zur regelmäßigen Überprüfung Ihrer AWS-Nutzung können Sie Ihre Nutzung aktiv überwachen und Ihre Bereitstellungen entsprechend anpassen.

Verwaltung von Nachfrage und Bereitstellung von Ressourcen

Wenn Sie in die Cloud wechseln, zahlen Sie nur für die genutzten Ressourcen. Sie können Ressourcen so bereitstellen, dass sie dem Workload-Bedarf zum jeweiligen Zeitpunkt entsprechen. Dadurch werden kostspielige Überbereitstellungen überflüssig. Sie können den Bedarf auch anpassen, indem Sie eine Drosselung, einen Puffer oder eine Warteschlange verwenden, um den Bedarf zu glätten und ihn mit weniger Ressourcen zu erfüllen, was zu niedrigeren Kosten führt. Außerdem können Sie ihn mit einem Batch-Service zu einem späteren Zeitpunkt verarbeiten.

In AWS können Sie Ressourcen automatisch so bereitstellen, dass sie den Workload-Bedarf erfüllen. Durch Auto Scaling mit bedarfs- oder zeitbasiertem Ansatz können Sie Ressourcen nach Bedarf hinzufügen und entfernen. Wenn Sie in der Lage sind, Bedarfsänderungen zu antizipieren, können Sie mehr Kosten einsparen und zugleich sicherstellen, dass Ihre Ressourcen Ihren Workload-Anforderungen entsprechen. Sie können Amazon API Gateway verwenden, um eine Drosselung zu implementieren, oder Amazon SQS einsetzen, um eine Warteschlange für Ihren Workload zu implementieren. Mit beiden können Sie den Bedarf für Ihre Workload-Komponenten anpassen.

In der folgenden Frage geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 9: Wie verwalten Sie die Nachfrage und stellen Ressourcen bereit?

Stellen Sie bei einem Workload mit ausgewogenen Ausgaben und Leistungen sicher, dass alles, wofür Sie bezahlen, genutzt wird, und vermeiden Sie eine erhebliche Unterauslastung der Instances. Eine verschobene Auslastungsmetrik in einer der Richtungen wirkt sich nachteilig auf Ihr Unternehmen aus, entweder im Hinblick auf die Betriebskosten (verschlechterte Leistung aufgrund von Überbelegung) oder auf die verschwendeten AWS-Ausgaben (aufgrund von Überversorgung).

Wenn Sie planen, dass Ressourcen für Bedarf und Bereitstellung geändert werden können, denken Sie auch an die Nutzungsmuster, die Zeit für die Bereitstellung neuer Ressourcen und die Vorhersehbarkeit des Bedarfsmusters. Stellen Sie beim Verwalten des Bedarfs sicher, dass Ihre Warteschlange oder Ihr Puffer korrekt dimensioniert ist und Sie in der erforderlichen Zeit auf den Workload-Bedarf reagieren.

Optimierung im Laufe der Zeit

Im Zuge der Veröffentlichung neuer Services und Funktionen durch AWS empfiehlt es sich, dass Sie Ihre bestehenden Entscheidungen zur Architektur überdenken, um sicherzustellen, dass diese weiterhin so kostengünstig wie möglich sind. Wenn sich Ihre Anforderungen ändern, zögern Sie nicht, und nehmen Sie Ressourcen, ganze Services und Systeme, die Sie nicht mehr benötigen, außer Betrieb.

Durch die Implementierung neuer Funktionen oder Ressourcentypen können Sie Ihren Workload inkrementell optimieren und gleichzeitig den Aufwand für die Implementierung der Änderung minimieren. Dadurch wird die Effizienz im Laufe der Zeit kontinuierlich verbessert und sichergestellt, dass Sie stets die aktuellste Technologie nutzen, um die Betriebskosten zu senken. Sie können mit neuen Services auch Komponenten des Workloads ersetzen oder ihm neue Komponenten hinzufügen. Dies kann zu erheblichen Effizienzsteigerungen führen. Daher ist es wichtig, Ihren Workload regelmäßig zu überprüfen und neue Services und Funktionen zu implementieren.

In der folgenden Frage geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 10: Wie können Sie neue Services bewerten?
Im Zuge der Veröffentlichung neuer Services und Funktionen durch AWS empfiehlt es sich, dass Sie Ihre bestehenden Entscheidungen zur Architektur überdenken, um sicherzustellen, dass diese weiterhin so kostengünstig wie möglich sind.

Wenn Sie Ihre Bereitstellungen regelmäßig überprüfen, sollten Sie auch bewerten, wie Sie mit neueren Services möglicherweise Geld sparen können. Mit Amazon Aurora on RDS können Sie beispielsweise die Kosten für relationale Datenbanken reduzieren. Wenn Sie serverlose Technologie wie Lambda verwenden, müssen Sie Instances nicht mehr betreiben und verwalten, um Code auszuführen.

Ressourcen

Weitere Informationen zu bewährten Methoden für die Kostenoptimierung finden Sie in den folgenden Ressourcen.

Dokumentation

- AWS-Dokumentation

Whitepaper

- Säule „Kostenoptimierung“

Nachhaltigkeit

Bei der Säule „Nachhaltigkeit“ geht es um Auswirkungen auf die Umwelt, insbesondere um Energieverbrauch und -effizienz, da diese wichtige Faktoren für Architekten sind, die ihre direkten Aktionen zur Reduzierung des Ressourcenverbrauchs beeinflussen. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule der Nachhaltigkeit.

Themen

- Designprinzipien
- Definition
- Bewährte Methoden

Designprinzipien

Es gibt sechs Designprinzipien für die Nachhaltigkeit in der Cloud:

- Verstehen Sie Ihre Auswirkungen: Messen Sie die Auswirkungen Ihrer Cloud-Workloads und modellieren Sie diese Auswirkungen für die Zukunft. berücksichtigen Sie dabei alle relevanten Faktoren, darunter Auswirkungen durch die Verwendung Ihrer Produkte durch Kunden sowie solche durch deren Außerbetriebnahme und Entsorgung. Vergleichen Sie den produktiven Output mit den Gesamtauswirkungen Ihrer Cloud-Workloads, indem Sie die für jede Arbeitseinheit erforderlichen Ressourcen und die damit verbundenen Emissionen ermitteln. Anhand dieser Daten können Sie Leistungskennzahlen (KPIs) einrichten, Möglichkeiten zur Verbesserung der

Produktivität bei gleichzeitiger Reduzierung der Auswirkungen finden und berechnen, wie sich vorgeschlagene Änderungen im Zeitverlauf auswirken werden.

- Legen Sie Nachhaltigkeitsziele fest: Formulieren Sie für alle Cloud-Workloads langfristige Nachhaltigkeitsziele wie etwa die Reduzierung der pro Transaktion erforderlichen Computing- und Speicherressourcen. Modellieren Sie den ROI von Verbesserungen in Bezug auf die Nachhaltigkeit vorhandener Workloads. Stellen Sie den Besitzern die nötigen Ressourcen zur Verfügung, um in Nachhaltigkeitsziele investieren zu können. Planen Sie wachstumsorientiert und gestalten Sie Ihre Workloads so, dass das Wachstum mit geringeren Auswirkungen einhergeht – gemessen in einer sinnvollen Einheit, etwa pro Benutzer oder pro Transaktion. Ziele helfen Ihnen, die allgemeinen Nachhaltigkeitsziele Ihres Unternehmens oder Ihrer Organisation zu erreichen, Rückschritte zu identifizieren und Bereiche mit Verbesserungsmöglichkeiten zu priorisieren.
- Maximieren Sie Ihre Auslastung: Sorgen Sie für Workloads angemessenen Umfangs und nutzen Sie effiziente Designprinzipien, um hohe Auslastung zu gewährleisten und die Energieeffizienz der zugrunde liegenden Hardware so zu maximieren. Zwei Hosts mit 30 % Auslastung sind aufgrund des grundlegenden Energieverbrauchs pro Host weniger effizient als ein Host mit 60 % Auslastung. Gleichzeitig sollten Sie nicht genutzte Ressourcen, Verarbeitungsvorgänge und Speicher beseitigen oder minimieren, um den Gesamtenergieverbrauch für Ihren Workload zu senken.
- Antizipieren und nutzen Sie neue und effizientere Hardware- und Software-Angebote: Unterstützen Sie die Verbesserungen, die Ihre Partner und Lieferanten in früheren Prozessphasen vornehmen, um die Auswirkungen Ihrer Cloud-Workloads zu reduzieren. Achten Sie stets auf neue und effizientere Hardware- und Software-Angebote. Planen Sie für Flexibilität, damit neue effiziente Technologien schnell eingeführt werden können.
- Verwenden Sie verwaltete Services: Die gemeinsame Nutzung von Services über eine breite Kundenbasis hinweg hilft dabei, die Ressourcennutzung zu maximieren und dadurch den Umfang der Infrastruktur zu verringern, der für die Unterstützung Ihrer Cloud-Workloads erforderlich ist. So können Kunden die Auswirkungen allgemeiner Rechenzentrumskomponenten wie Energieversorgung und Netzwerk teilen, indem sie Workloads zur AWS Cloud migrieren und verwaltete Services einführen, z. B. AWS Fargate für Serverless-Container. Dabei kann AWS skalierbar ausgeführt werden und ist für einen effizienten Betrieb verantwortlich. Verwenden Sie verwaltete Services, die dabei helfen können, Ihre Auswirkungen zu verringern, wie etwa die automatische Verschiebung selten genutzter Daten in „kalte“ Speicher mit Amazon S3 Lifecycle-Konfigurationen oder Amazon EC2 Auto Scaling, um Ihre Kapazitäten an die jeweiligen Anforderungen anzupassen.

- Reduzieren Sie die nachgelagerten Auswirkungen Ihrer Cloud-Workloads: Senken Sie den Energie- oder Ressourcenverbrauch für die Nutzung Ihrer Services. Reduzieren oder beseitigen Sie die Erfordernis einer Geräteaktualisierung auf Kundenseite, wenn sie Ihre Services nutzen möchten. Verwenden Sie in Ihren Tests Gerätefarmen, um die zu erwartenden Auswirkungen zu verstehen, und führen Sie Tests mit Kunden durch, um die tatsächlichen Auswirkungen der Nutzung Ihrer Services zu erkennen.

Definition

Es gibt sechs bewährte Methoden für die Nachhaltigkeit in der Cloud:

- Auswahl von Regionen
- Verhaltensmuster von Benutzern
- Software- und Architekturmuster
- Datenmuster
- Hardwaremuster
- Entwicklungs- und Bereitstellungsprozess

Nachhaltigkeit in der Cloud ist ein kontinuierliches Bestreben, das sich in erster Linie auf die Reduzierung des Energieverbrauchs und die Effizienz aller Komponenten eines Workloads konzentriert. Dazu muss der maximale Nutzen aus den bereitgestellten Ressourcen gezogen und die insgesamt erforderlichen Ressourcen müssen minimiert werden. Diese Bemühung kann von der anfänglichen Auswahl einer effizienten Programmiersprache, der Einführung moderner Algorithmen, der Nutzung effizienter Datenspeichertechniken, der Bereitstellung einer korrekt dimensionierten und effizienten Recheninfrastruktur bis hin zur Minimierung der Anforderungen an leistungsstarke Endbenutzerhardware reichen.

Bewährte Methoden

Themen

- Auswahl von Regionen
- Verhaltensmuster von Benutzern
- Software- und Architekturmuster
- Datenmuster
- Hardwaremuster

- Entwicklungs- und Bereitstellungsmuster
- Ressourcen

Auswahl von Regionen

Wählen Sie die Regionen, in denen Sie Ihre Workloads implementieren, anhand Ihrer geschäftlichen Anforderungen und Ihrer Nachhaltigkeitsziele aus.

In der folgenden Frage geht es um Überlegungen zur Nachhaltigkeit. (Eine Liste der Fragen und bewährten Methoden zur Nachhaltigkeit finden Sie im Anhang.)

SUS 1: Wie wählen Sie Regionen aus, um Ihre Nachhaltigkeitsziele zu unterstützen?
Wählen Sie Regionen in der Nähe von Amazon-Projekten für erneuerbare Energien aus. Es sollte sich um Regionen handeln, in denen das Stromnetz nachweislich geringere Kohlendioxidemissi onen generiert als andere Standorte (oder Regionen).

Verhaltensmuster von Benutzern

Die Art und Weise, wie Benutzer Ihre Workloads und andere Ressourcen nutzen, kann Sie bei der Identifizierung von Verbesserungen unterstützen, um Nachhaltigkeitsziele zu erreichen. Skalieren Sie Ihre Infrastruktur, um die Benutzerlast kontinuierlich anzupassen. Sorgen Sie dafür, dass zur Unterstützung der Benutzer stets nur die mindestens erforderlichen Ressourcen bereitgestellt werden. Richten Sie Service-Levels an den Kundenanforderungen aus. Positionieren Sie Ressourcen so, dass die für ihre Nutzung erforderlichen Netzwerkkapazitäten begrenzt werden. Entfernen Sie vorhandene, nicht verwendete Komponenten. Identifizieren Sie erstellte, aber nicht verwendete Komponenten und beenden Sie ihre Generierung. Stellen Sie Teammitgliedern Geräte zur Verfügung, die ihre Anforderungen bei geringstmöglichen Auswirkungen auf die Nachhaltigkeit erfüllen.

In der folgenden Frage geht es um diese Überlegungen zur Nachhaltigkeit:

SUS 2: Wie können Sie Verhaltensmuster von Benutzern zur Unterstützung Ihrer Nachhalti gkeitsziele nutzen?
Die Art und Weise, wie Benutzer Ihre Workloads und andere Ressourcen nutzen, kann Sie bei der Identifizierung von Verbesserungen unterstützen, um Nachhaltigkeitsziele zu erreichen. Skalieren

SUS 2: Wie können Sie Verhaltensmuster von Benutzern zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Sie Ihre Infrastruktur, um die Benutzerlast kontinuierlich anzupassen. Sorgen Sie dafür, dass zur Unterstützung der Benutzer stets nur die mindestens erforderlichen Ressourcen bereitgestellt werden. Richten Sie Service-Levels an den Kundenanforderungen aus. Positionieren Sie Ressourcen so, dass die für ihre Nutzung erforderlichen Netzwerkkapazitäten begrenzt werden. Entfernen Sie vorhandene, nicht verwendete Komponenten. Identifizieren Sie erstellte, aber nicht verwendete Komponenten und beenden Sie ihre Generierung. Stellen Sie Teammitgliedern Geräte zur Verfügung, die ihre Anforderungen bei geringstmöglichen Auswirkungen auf die Nachhaltigkeit erfüllen.

Skalieren der Infrastruktur anhand der Benutzerlast: Identifizieren Sie Zeiträume mit geringer oder gar keiner Nutzung und skalieren Sie Ressourcen, um überschüssige Kapazitäten zu entfernen und die Effizienz zu verbessern.

Ausrichten von SLAs an Nachhaltigkeitszielen: Definieren und aktualisieren Sie Service Level Agreements (SLAs), darunter die Zeiträume für Verfügbarkeit und Datenaufbewahrung, um den Ressourcenaufwand für Ihre Workloads zu minimieren und gleichzeitig geschäftliche Anforderungen weiter erfüllen zu können.

Beenden der Erstellung und Wartung nicht verwendeter Komponenten: Analysieren Sie Anwendungskomponenten (wie vorab kompilierte Berichte, Datensätze und statische Bilder) sowie Zugriffsmuster für Komponenten, um Redundanzen, eine zu geringe Auslastung und mögliche Kandidaten für die Außerbetriebnahme zu identifizieren. Konsolidieren Sie generierte Komponenten mit redundanten Inhalten (z. B. monatliche Berichte mit sich überschneidenden oder gemeinsam genutzten Datensätzen und Ausgaben), um für duplizierte Ausgaben genutzte Ressourcen zu eliminieren. Deaktivieren Sie nicht verwendete Komponenten (z. B. Bilder von Produkten, die nicht mehr verkauft werden), um genutzte Ressourcen freizugeben und die Zahl der Ressourcen zu reduzieren, die zur Unterstützung von Workloads verwendet werden.

Optimieren der geografischen Platzierung von Workloads für Benutzerstandorte: Analysieren Sie Netzwerkzugriffsmuster, um zu erkennen, aus welchen geographischen Regionen Ihre Kunden Verbindungen herstellen. Wählen Sie Regionen und Services im Hinblick auf die Reduzierung der Distanz für den Netzwerkdatenverkehr aus, um die Zahl der Netzwerkressourcen zu verringern, die zur Unterstützung von Workloads benötigt werden.

Optimieren von Ressourcen für Teammitglieder im Hinblick auf die ausgeführten Aktivitäten: Optimieren Sie die Ressourcen, die Teammitgliedern zur Verfügung gestellt werden, um negative Auswirkungen auf die Nachhaltigkeit zu minimieren und gleichzeitig ihre Anforderungen zu erfüllen. Beispielsweise können Sie komplexe Vorgänge wie Rendering und Kompilierung auf intensiv genutzten, geteilten Cloud-Desktops statt auf weniger ausgelasteten Einzelbenutzersystemen mit hohem Energieverbrauch ausführen.

Software- und Architekturmuster

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

In den folgenden Fragen geht es um Überlegungen zur Nachhaltigkeit:

SUS 3: Wie können Sie Software- und Architekturmuster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

Optimieren von Software und Architektur für asynchrone und geplante Aufträge: Verwenden Sie effiziente Softwaredesigns und Architekturen, um die Zahl der für einzelne Arbeitseinheiten im Durchschnitt benötigten Ressourcen zu minimieren. Implementieren Sie Mechanismen für die

gleichmäßige Nutzung von Komponenten, um die Zahl der Ressourcen zu reduzieren, die zwischen Aufgaben nicht genutzt werden, und die Auswirkungen von Lastspitzen zu minimieren.

Entfernen von Workload-Komponenten mit geringer oder keiner Nutzung oder Faktorwechsel: Überwachen Sie die Workload-Aktivität, um Änderungen bei der Nutzung einzelner Komponenten über die Zeit zu erkennen. Entfernen Sie ungenutzte Komponenten, die nicht mehr benötigt werden. Setzen Sie wenig genutzte Ressourcen neu ein, um die Verschwendung von Ressourcen zu begrenzen.

Optimieren von Codebereichen, die die meiste Zeit oder die meisten Ressourcen verbrauchen: Überwachen Sie die Workload-Aktivität, um die Anwendungskomponenten zu identifizieren, die die meisten Ressourcen verbrauchen. Optimieren Sie den Code, der innerhalb dieser Komponenten ausgeführt wird, um die Ressourcennutzung zu minimieren und die Leistung zu maximieren.

Optimieren der Auswirkungen auf Geräte und Ausrüstung von Kunden: Identifizieren Sie die Geräte und Einrichtungen, mit denen Ihre Kunden Ihre Services nutzen, ihren voraussichtlichen Lebenszyklus und die finanziellen und nachhaltigkeitsbezogenen Auswirkungen der Ersetzung dieser Komponenten. Implementieren Sie Softwaremuster und Architekturen, die es für Kunden unnötig machen, Geräte zu ersetzen oder ihre Ausrüstung zu aktualisieren. Implementieren Sie beispielsweise neue Funktionen, die Code verwenden, der mit älterer Hardware und älteren Betriebssystemversionen abwärtskompatibel ist, oder gestalten Sie die Größe von Nutzlasten so, dass sie die Speicherkapazitäten der Zielgeräte nicht überschreiten.

Verwenden von Softwaremustern und Architekturen, die Datenzugriffs- und Speichermuster optimal unterstützen: Identifizieren Sie, wie Daten in Ihrem Workload verwendet, von Benutzern genutzt, übertragen und gespeichert werden. Wählen Sie Technologien aus, die die Anforderungen an Datenverarbeitung und -speicherung minimieren.

Datenmuster

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

In der folgenden Frage geht es um Überlegungen zur Nachhaltigkeit:

SUS 4: Wie können Sie Datenzugriffs- und -nutzungsmuster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Implementieren Sie Verfahren für die Datenverwaltung, die den zur Unterstützung Ihres Workloads bereitgestellten Speicher und die für dessen Nutzung erforderlichen Ressourcen reduzieren. Identifizieren Sie Ihre Daten und verwenden Sie Speichertechnologien und Konfigurationen, die den Unternehmenswert und die Nutzung der Daten optimal unterstützen. Verschieben Sie die Daten während des Lebenszyklus zu effizienteren Speichern mit geringerer Leistung, wenn die Anforderungen abnehmen. Löschen Sie Daten, die nicht mehr benötigt werden.

Implementieren einer Richtlinie für die Klassifizierung von Daten: Klassifizieren Sie Daten, um ihre Bedeutung für geschäftliche Ergebnisse zu verstehen. Nutzen Sie diese Informationen, um festzulegen, wann Daten in einen energieeffizienteren Speicher übertragen oder auf sichere Weise gelöscht werden können.

Verwenden von Technologien, die Datenzugriff und Speichermuster unterstützen: Nutzen Sie einen Speicher, der den Zugriff auf Ihre Daten und ihre Speicherung jeweils optimal unterstützt, um die Zahl der bereitgestellten Ressourcen zu minimieren und gleichzeitig den Workload zu unterstützen. Beispielsweise verbrauchen SSD-Laufwerke mehr Energie als magnetische Laufwerke und sollten nur für aktive Datenanwendungsfälle eingesetzt werden. Verwenden Sie für Daten, auf die nicht häufig zugegriffen wird, einen energieeffizienten Archivierungsspeicher.

Verwenden von Lebenszyklusrichtlinien zum Löschen nicht notwendiger Daten: Verwalten Sie den Lebenszyklus aller Daten und setzen Sie automatisch Löschfristen durch, um die Speicheranforderungen Ihres Workloads insgesamt zu minimieren.

Minimieren übermäßiger Bereitstellungen im Blockspeicher: Erstellen Sie zur Minimierung des insgesamt bereitgestellten Speichers Blockspeicher mit Größenzuweisungen entsprechend dem jeweiligen Workload. Verwenden Sie elastische Volumes, um den Speicher bei wachsenden Datenmengen erweitern zu können, ohne die Größe des an Computing-Ressourcen angefügten Speichers ändern zu müssen. Überprüfen Sie elastische Volumes regelmäßig und verkleinern Sie zu große Volumes, um sie an den aktuellen Datenumfang anzupassen.

Entfernen nicht benötigter oder redundanter Daten: Duplizieren Sie Daten nur wie notwendig, um den insgesamt genutzten Speicher zu minimieren. Verwenden Sie Backup-Technologien, die Daten

auf Datei- und Blockebene deduplizieren. Verwenden Sie Konfigurationen mit Redundant Array of Independent Drives (RAID) nur, wenn dies zur Erfüllung von SLAs notwendig ist.

Verwenden geteilter Dateisysteme oder Objektspeicher für den Zugriff auf allgemeine Daten: Verwenden Sie geteilten Speicher und zentrale Datenquellen, um Datenduplizierungen zu vermeiden und den Gesamtspeicherbedarf des Workloads zu reduzieren. Rufen Sie Daten nur wie notwendig aus dem geteilten Speicher ab. Trennen Sie nicht genutzte Volumes, um Ressourcen freizugeben. Minimieren Sie Datenübertragungen über Netzwerke hinweg. Verwenden Sie stattdessen einen geteilten Speicher und greifen Sie über regionale Datenspeicher auf die Daten zu, um die Zahl der Netzwerkressourcen zu minimieren, die für Datenübertragungen für Ihren Workload benötigt werden.

Sichern von Daten nur in dem Fall, dass ihre erneute Erstellung schwierig ist: Sichern Sie zur Minimierung der Speichernutzung nur Daten, die einen Unternehmenswert besitzen oder zur Erfüllung von Compliance-Anforderungen benötigt werden. Prüfen Sie Backup-Richtlinien und vermeiden Sie einen flüchtigen Speicher, der in einem Wiederherstellungsszenario keinen Wert bietet.

Hardwaremuster

Suchen Sie nach Möglichkeiten, die Auswirkungen auf die Nachhaltigkeit Ihrer Workloads durch Änderungen der Methoden für die Hardwareverwaltung zu reduzieren. Minimieren Sie den Umfang der für die Bereitstellung erforderlichen Hardware und wählen Sie die jeweils effizienteste Hardware für den jeweiligen Workload aus.

In der folgenden Frage geht es um Überlegungen zur Nachhaltigkeit:

SUS 5: Wie können Hardwareverwaltung und Nutzungsverfahren Ihre Nachhaltigkeitsziele unterstützen?
Suchen Sie nach Möglichkeiten, die Auswirkungen auf die Nachhaltigkeit Ihrer Workloads durch Änderungen der Methoden für die Hardwareverwaltung zu reduzieren. Minimieren Sie den Umfang der für die Bereitstellung erforderlichen Hardware und wählen Sie die jeweils effizienteste Hardware für den jeweiligen Workload aus.

Verwenden der geringstmöglichen Menge an Hardware zur Erfüllung Ihrer Anforderungen: Mit den Möglichkeiten der Cloud können Sie häufige Änderungen für Ihre Workload-Implementierungen ausführen. Aktualisieren Sie bereitgestellte Komponenten, wenn sich Ihre Anforderungen ändern.

Überwachen Sie kontinuierlich die Einführung neuer Instance-Typen und nutzen Sie Verbesserungen bei der Energieeffizienz, einschließlich Instance-Typen, die zur Unterstützung spezifischer Workloads bestimmt sind, wie z. B. Machine-Learning-Trainings und -Inferenzen und Videotranskodierung.

Verwenden von Instance-Typen mit den geringsten Auswirkungen: Mit verwalteten Services geht die Verantwortung für die Wahrung einer hohen durchschnittlichen Nutzung und die Optimierung der Nachhaltigkeit der bereitgestellten Hardware auf AWS über. Mit verwalteten Services können Sie die nachhaltigkeitsbezogenen Auswirkungen des Service über alle Mandanten des Service verteilen und so Ihren Beitrag verringern.

Optimieren der GPU-Nutzung: Grafikverarbeitungseinheiten (Graphics Processing Units, GPUs) können sehr viel Energie verbrauchen. Zahlreiche GPU-Workloads sind hoch variabel, z. B. Rendern, Transkodieren sowie Machine-Learning-Trainings und -Modellierungen. Führen Sie GPU-Instances nur für die benötigte Zeit aus und automatisieren Sie ihre Außerbetriebnahme, wenn sie nicht benötigt werden, um den Ressourcenverbrauch zu minimieren.

Entwicklungs- und Bereitstellungsmuster

Reduzieren Sie nachhaltigkeitsbezogene Auswirkungen, indem Sie Ihre Entwicklungs-, Test- und Bereitstellungsmethoden ändern.

In der folgenden Frage geht es um Überlegungen zur Nachhaltigkeit:

SUS 6: Wie können Ihre Entwicklungs- und Bereitstellungsprozesse Ihre Nachhaltigkeitsziele unterstützen?
Reduzieren Sie nachhaltigkeitsbezogene Auswirkungen, indem Sie Ihre Entwicklungs-, Test- und Bereitstellungsmethoden ändern.

Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen: Testen und validieren Sie potenzielle Verbesserungen, bevor Sie sie für die Produktion bereitstellen. Berücksichtigen Sie die Testkosten bei der Berechnung des potenziellen zukünftigen Nutzens einer Verbesserung. Entwickeln Sie kostengünstige Testmethoden, um kleine Verbesserungen einzuführen.

Konstantes Aktualisieren Ihres Workloads: Aktuelle Betriebssysteme, Bibliotheken und Anwendungen können die Workload-Effizienz verbessern und die Nutzung effizienterer Technologien unterstützen. Eine aktuelle Software kann darüber hinaus Funktionen für eine genauere Messung

der Auswirkungen Ihres Workloads bereitstellen, da die Anbieter mit ihrer Software ebenfalls Nachhaltigkeitsziele erfüllen müssen.

Höhere Auslastung von Entwicklungsumgebungen: Verwenden Sie Automatisierung und Infrastructure-as-Code, um Vorproduktionsumgebungen bei Bedarf in Betrieb und bei Nichtverwendung wieder außer Betrieb zu nehmen. Eine typische Vorgehensweise besteht in der Planung von Verfügbarkeitszeiten, die mit den Arbeitszeiten der Entwicklungsteams übereinstimmen. Der Ruhezustand ist ein nützliches Tool, um den aktuellen Status beizubehalten und Instances nur zu aktivieren, wenn sie benötigt werden. Verwenden Sie Instance-Typen mit Burst-Kapazität, Spot-Instances, Elastic Database-Services, Containern und anderen Technologien, um Entwicklungs- und Testkapazität an die Nutzung anzupassen.

Verwenden verwalteter Gerätefarmen für Tests verwenden: Verwaltete Gerätefarmen verteilen die nachhaltigkeitsbezogenen Auswirkungen der Hardwarefertigung und der Ressourcennutzung über zahlreiche Beteiligte. Verwaltete Gerätefarmen stellen verschiedene Gerätetypen bereit, unterstützen auch ältere und weniger verbreitete Hardware und vermeiden nachhaltigkeitsbezogene Auswirkungen durch unnötige Geräte-Upgrades seitens Kunden.

Ressourcen

Werfen Sie einen Blick auf die folgenden Ressourcen, um mehr über unsere bewährten Methoden für die Nachhaltigkeit zu erfahren.

Whitepaper

- Säule „Nachhaltigkeit“

Video

- The Climate Pledge

Die Überprüfung

Architekturen müssen nach einheitlichen Gesichtspunkten überprüft werden. Wenn dabei niemand an den Pranger gestellt wird, ist eine Voraussetzung für tief schürfende Analysen gegeben. Der Prozess sollte nicht schwerfällig sein (Stunden, nicht Tage) und als Konversation angelegt sein, nicht als Prüfung. Architekturen werden überprüft, um festzustellen, ob kritische Mängel vorliegen, gegen die etwas unternommen werden muss – oder um festzustellen, ob bestimmte Bereiche nachgebessert werden können. Am Ende der Überprüfung stehen Maßnahmen, die dem Kunden, der mit dem Workload arbeitet, ein angenehmeres Erlebnis ermöglichen.

Wie bereits im Abschnitt "Architektur-Überlegungen" angesprochen, ist es in Ihrem Interesse, dass jedes Teammitglied Verantwortung für die Qualität der Architektur übernimmt. Wir empfehlen, dass die Teammitglieder, die die Architektur entwerfen, mit Hilfe des Well-Architected Framework ihre Architektur fortlaufend überprüfen, anstatt eine formelle Überprüfungsbesprechung anzusetzen. Findet die Überprüfung fortlaufend statt, können Ihre Teammitglieder parallel mit der Entwicklung der Architektur Antworten aktualisieren und mit jeder neuen Funktion die Architektur verbessern.

Das AWS Well-Architected Framework ist ähnlich aufgebaut wie der interne AWS-Prozess zur Überprüfung von Systemen und Services. Der architektonische Ansatz wird beeinflusst von konzeptionellen Grundsätzen und Fragen, die sicherstellen, dass Bereiche nicht vernachlässigt werden, die häufig in der Ursachenanalyse auftauchen. Tritt an einem internen System, AWS-Service oder bei einem Kunden ein schwerwiegendes Problem auf, untersuchen wir die Ursachenanalyse auf Verbesserungsmöglichkeiten für unsere Überprüfungsprozesse.

Die Überprüfungen müssen an wichtigen Meilensteinen des Produktzyklus erfolgen – früh in der Entwurfsphase, um einseitige Türen zu vermeiden, an denen schwer nachzubessern ist. Und zuletzt schließlich kurz vor dem Go-Live. Viele Entscheidungen können rückgängig gemacht werden; es gibt zwei Möglichkeiten. Für diese Entscheidungen reicht ein schlanker Prozess. Gibt es nur eine Möglichkeit, kann diese nur schwer oder gar nicht rückgängig gemacht werden und muss genauer inspiziert werden, bevor sie gewählt wird. Nachdem Sie in Produktion gehen, verändert sich Ihr Workload weiter, da neue Funktionen hinzukommen und Sie Technologieimplementierungen anpassen. Die Architektur eines Workloads verändert sich mit der Zeit. Treffen Sie durchdachte Hygienemaßnahmen, um zu verhindern, dass die Qualität seiner architektonischen Merkmale im Zuge der Weiterentwicklung nachlässt. Wenn Sie an der Architektur signifikante Änderungen vornehmen, müssen Sie bestimmte Hygieneprozesse befolgen, z. B. eine Überprüfung nach dem Well-Architected-Prinzip.

Wenn die Überprüfung als einmalige Momentaufnahme oder unabhängige Messung vorgesehen ist, müssen alle wichtigen Beteiligten in die Konversation eingebunden sein. Häufig ist die Überprüfung der Punkt, an dem einem Team das erste Mal richtig klar wird, was es implementiert hat. Wird der Workload eines anderen Teams überprüft, ist es sinnvoll, mehrere informelle Konversationen über seine Architektur einzuplanen. In diesen Gesprächen erhalten Sie Antworten auf die meisten Fragen. Im Anschluss daran können Sie in ein oder zwei Besprechungen Punkte abklären und ausführlich auf Unklarheiten oder eventuelle Risiken eingehen.

Damit Ihre Besprechungen erfolgreich verlaufen, empfehlen wir folgende Ausstattung:

- Besprechungszimmer mit Whiteboards
- Diagramme und Entwurfsnotizen ausgedruckt auf Papier
- Liste der Fragen, die sich nicht mit herkömmlichen Mitteln beantworten lassen (z. B. „Werden die Daten verschlüsselt?“)

Nach der Überprüfung sollten Sie eine Liste mit Problemen vorliegen haben. Welche Sie priorisieren, hängt vom geschäftlichen Kontext ab. Berücksichtigen Sie auch, wie sich diese Probleme auf die tägliche Arbeit Ihres Teams auswirken. Wenn Sie die Probleme frühzeitig angehen, gewinnen Sie vielleicht Zeit. Zeit, in der Sie geschäftlichen Mehrwert schaffen können, anstatt sich um wiederkehrende Probleme zu kümmern. Während Sie die Probleme aus der Welt schaffen, können Sie Ihre Überprüfung aktualisieren und so verfolgen, wie sich die Architektur verbessert.

Wie hilfreich eine Überprüfung war, zeigt sich erst danach. Neue Teams widersetzen sich möglicherweise zuerst. Sie können Einwänden der Teams entgegnen, indem Sie sie über die Vorteile einer Überprüfung aufklären:

- „Wir sind zu beschäftigt!“ (Häufig im Vorfeld großer Produktstarts zu hören)
  - Wenn ihr euch auf einen großen Launch vorbereitet, sollte der möglichst glatt über die Bühne gehen. Die Überprüfung deckt Schwachstellen auf, die ihr vielleicht übersehen habt.
  - Wir empfehlen, dass ihr früh im Produktzyklus Überprüfungen einbaut, um Risiken aufzudecken und einen Auffangplan auszuarbeiten, der auf die Roadmap für die Feature-Bereitstellung abgestimmt ist.
- „Wir haben nicht die Zeit, um mit den Ergebnissen etwas anzufangen!“ (Oft zu hören, wenn ein unverrückbares Ereignis näher rückt, z. B. eine große Sportveranstaltung, auf das alles ausgerichtet ist)

  - Diese Ereignisse lassen sich nicht verschieben. Wollt ihr da wirklich reingehen, ohne die Risiken eurer Architektur zu kennen? Selbst wenn ihr nicht alle Probleme wegbekommt, könnt ihr euch immer noch mit Playbooks helfen, wenn sie tatsächlich eintreten.
- „Wir möchten nicht, dass andere die Geheimnisse unserer Lösungsimplementierung kennenlernen!“
  - Wenn Sie die Aufmerksamkeit des Teams auf die Fragen im Well-Architected Framework richten, erkennen sie, dass keine der Fragen kommerziell oder technisch sensible Informationen herauszieht.

Wenn Sie mit Teams aus Ihrer Organisation mehrere Überprüfungen durchführen, identifizieren Sie möglicherweise thematische Fragen. So könnte sich beispielsweise herausstellen, dass mehrere Teams in einer bestimmten Säule oder einem bestimmten Themengebiet mehrere zusammenhängende Probleme haben. Werfen Sie einen ganzheitlichen Blick auf all Ihre Überprüfungen und identifizieren Sie Mechanismen, Trainings oder Principal-Engineer-Vorträge, mit deren Hilfe sich diese thematischen Fragen angehen lassen.

Fazit

Das AWS Well-Architected Framework liefert über alle sechs Säulen hinweg bewährte architektonische Methoden für die Entwicklung und den Betrieb zuverlässiger, sicherer, effizienter, kosteneffizienter und nachhaltiger Systeme in der Cloud. Die Fragen aus dem Framework erlauben Ihnen, bestehende und geplante Architekturen zu überprüfen. Außerdem sind darin bewährte AWS-Methoden für die fünf Säulen enthalten. Als fester Bestandteil Ihres Architekturdesigns fördert das Framework stabile und effiziente Systeme. Anschließend können Sie sich auf Ihre funktionalen Anforderungen konzentrieren.

Mitwirkende

Dieses Dokument ist unter der Mitarbeit folgender Personen und Organisationen entstanden:

- Brian Carlson, Operations Lead Well-Architected, Amazon Web Services
- Ben Potter, Security Lead Well-Architected, Amazon Web Services
- Seth Eliot, Reliability Lead Well-Architected, Amazon Web Services
- Eric Pullen, Sr. Solutions Architect, Amazon Web Services
- Rodney Lester, Principal Solutions Architect, Amazon Web Services
- Jon Steele, Sr. Technical Account Manager, Amazon Web Services
- Max Ramsay, Principal Security Solutions Architect, Amazon Web Services
- Callum Hughes, Solutions Architect, Amazon Web Services
- Aden Leirer, Content Program Manager Well-Architected, Amazon Web Services

Weitere Informationen

AWS-Architekturzentrum

AWS Cloud-Compliance

AWS Well-Architected-Partnerprogramm

AWS Well-Architected Tool

AWS Well-Architected-Homepage

Whitepaper zur Säule für die betriebliche Exzellenz

Whitepaper der Säule für Sicherheit

Whitepaper zur Säule der Zuverlässigkeit

Whitepaper zur Säule der Leistungseffizienz

Whitepaper zur Säule der Kostenoptimierung

Whitepaper zur Säule der Nachhaltigkeit

Die Amazon Builders' Library

Dokumentversionen

Abonnieren Sie den RSS-Feed, um über Aktualisierungen des Whitepapers benachrichtigt zu werden.

Änderung	Beschreibung	Datum
Größere Aktualisierung	Bewährte Methoden mit verbindlichen Anleitungen aktualisiert und neue bewährte Methoden hinzugefügt.	June 27, 2024
Größere Aktualisierung	Wichtige Leistungssäule Umstrukturierung, um die Anzahl der Bereiche mit bewährten Methoden auf fünf zu erhöhen. Umfangreiche Aktualisierung der bewährten Methoden und Leitlinien in der Sicherheitssäule in Reaktion auf Vorfälle (SEC 10). Wesentliche inhaltliche Änderungen und Konsolidi erung in den Bereichen Operative Exzellenz OPS 04, 05, 06, 08 und 09. Aktualisi erungen der Anleitungen für die Kostenoptimierung und Zuverlässigkeit Säulen. Geringfügige Aktualisierungen der Nachhaltigkeitskeitssäule Risikostufen.	October 3, 2023
Updates für das neue Framework	Bewährte Methoden mit verbindlichen Anleitungen aktualisiert und neue bewährte Methoden hinzugefügt.	April 10, 2023

	Neue Fragen zu den Säulen Sicherheit und Kostenopt imierung hinzugefügt.	
Kleineres Update	Eine Definition für Grad des Aufwands wurde hinzugefügt und bewährte Methoden im Anhang wurden aktualisiert.	October 20, 2022
Whitepaper aktualisiert	Die Säule „Nachhaltigkeit“ wurde hinzugefügt und Links wurden aktualisiert.	December 2, 2021
Größere Aktualisierung	Die Säule „Nachhaltigkeit“ wurde zum Framework hinzugefügt.	November 20, 2021
Kleineres Update	Nicht inklusive Sprache entfernt.	April 22, 2021
Kleineres Update	Zahlreiche Links wurden repariert.	March 10, 2021
Kleineres Update	Kleinere redaktionelle Änderungen im gesamten Dokument.	July 15, 2020
Updates für das neue Framework	Prüfung und Umformulierung der meisten Fragen und Antworten.	July 8, 2020
Whitepaper aktualisiert	Ergänzung des AWS Well-Architected Tool, Links zu AWS Well-Architected Labs und AWS-Well-Architected-Partnern, kleinere Fehlerbeh ebungen zur Aktivierung mehrerer Sprachversionen des Frameworks.	July 1, 2019

Whitepaper aktualisiert	Die meisten Fragen und Antworten wurden noch einmal durchgelesen und umgeschrieben, damit die Fragen jeweils nur ein Thema behandeln. Dabei wurden einige Fragen in mehrere Einzelfragen aufgeteilt. Häufig verwendete Begriffe (Workload, Komponente usw.) wurden definiert. Darstellung der Fragen im Textkorpus wurde bearbeitet, um Platz zu schaffen für Erläuterungen.	November 1, 2018
Whitepaper aktualisiert	Fragentext ist nach mehreren Updates einfacher formuliert, Antworten sind standardisiert und die Lesbarkeit wurde verbessert.	June 1, 2018
Whitepaper aktualisiert	Operative Exzellenz wurde vor die anderen Säulen gesetzt und umgeschrieben. Umfasst jetzt die anderen Säulen. Die anderen Säulen wurden aktualisiert, um der Weiterentwicklung von AWS Rechnung zu tragen.	November 1, 2017

Whitepaper aktualisiert	Aktualisierung des Framework . Dieses enthält jetzt die Säule „Operative Exzellenz“. Die anderen Säulen wurden überarbeitet und aktualisiert. Dabei wurden Doppelnen nungen ausgeräumt und Erkenntnisse aus Überprüfu ngen bei mehreren Tausend Kunden aufgenommen.	November 1, 2016
Kleinere Updates	Anhang wurde mit aktuellen Amazon CloudWatch Logs Informationen aktualisiert.	November 1, 2015
Erstveröffentlichung	AWS-Well-Architected-Framework wurde veröffent licht.	October 1, 2015

Anhang: Fragen und bewährte Methoden

Dieser Anhang fasst alle Fragen und bewährten Methoden im AWS Well-Architected Framework zusammen.

Säulen

- Operational Excellence
- Sicherheit
- Zuverlässigkeit
- Leistungseffizienz
- Kostenoptimierung
- Nachhaltigkeit

Operational Excellence

Die Säule für die betriebliche Exzellenz umfasst die Unterstützung der Entwicklung und effektive Ausführung von Workloads, Einblicke in Ihre Betriebsabläufe und eine fortlaufende Verbesserung unterstützender Prozesse und Verfahren, damit geschäftlicher Mehrwert geschaffen wird. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule für die betriebliche Exzellenz.

Bereiche für bewährte Methoden

- Organisation
- Vorbereitung
- Betrieb
- Weiterentwicklung

Organisation

Fragen

- OPS 1. Wie können Sie Ihre Prioritäten bestimmen?
- OPS 2. Wie strukturieren Sie Ihr Unternehmen, um die gewünschten Geschäftsergebnisse zu erzielen?

- OPS 3. Wie unterstützt Ihre Unternehmenskultur Ihre Geschäftsergebnisse?

OPS 1. Wie können Sie Ihre Prioritäten bestimmen?

Jeder sollte seinen Teil dazu beitragen, den Geschäftserfolg zu erreichen. Setzen Sie sich gemeinsame Ziele, damit Sie die Prioritäten für Ressourcen festlegen können. Dadurch erzielen Ihre Bemühungen den größtmöglichen Nutzen.

Bewährte Methoden

- OPS01-BP01 Kundenbedürfnisse bewerten
- OPS01-BP02 Bedürfnisse interner Kunden bewerten
- OPS01-BP03 Bewerten der Governance-Anforderungen
- OPS01-BP04 Bewerten der Compliance-Anforderungen
- OPS01-BP05 Bewerten der Bedrohungsszenarien
- OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken

OPS01-BP01 Kundenbedürfnisse bewerten

Binden Sie alle wichtigen Stakeholder ein, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams, um zu bestimmen, welche Bereiche verstärkt auf die Bedürfnisse der externen Kunden ausgerichtet werden müssen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um die gewünschten geschäftlichen Ergebnisse zu erzielen.

Gewünschtes Ergebnis:

- Sie arbeiten rückwärts von den Kundenergebnissen aus.
- Sie wissen, wie Ihre betrieblichen Praktiken Geschäftsergebnisse und -ziele unterstützen.
- Sie binden alle relevanten Parteien ein.
- Sie verfügen über Mechanismen, um Kundenbedürfnisse zu erfassen.

Typische Anti-Muster:

- Sie haben sich entschieden, außerhalb der Kerngeschäftszeiten keinen Kundenservice zu bieten, aber Sie haben dazu keine historischen Supportanfragedaten analysiert. Daher wissen Sie nicht, ob diese Entscheidung Auswirkungen auf Ihre Kunden hat.

- Sie entwickeln ein neues Feature, haben aber Ihre Kunden nicht miteinbezogen, um herauszufinden, ob die Funktion erwünscht ist und wie sie genau aussehen sollte. Außerdem haben Sie keine Tests durchgeführt, um die Nachfrage und die Methode der Bereitstellung zu validieren.

Vorteile der Einführung dieser bewährten Methode: Kunden, deren Anforderungen erfüllt sind, bleiben mit höherer Wahrscheinlichkeit als Kunden erhalten. Die Bewertung und das Verständnis externer Kundenbedürfnisse liefert die Grundlage dafür, wie Sie Ihre Anstrengungen zur Bereitstellung eines geschäftlichen Mehrwerts priorisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Verstehen Sie die geschäftlichen Anforderungen: Der geschäftliche Erfolg basiert auf gemeinsamen Zielen und der Kommunikation zwischen allen Stakeholdern, zu denen auch die Teams aus den Bereichen Geschäft, Entwicklung und Betrieb gehören.

Überprüfen Sie die geschäftlichen Ziele, Anforderungen und Prioritäten externer Kunden: Führen Sie wichtige Stakeholder zusammen, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams, um die Ziele, Anforderungen und Prioritäten externer Kunden zu besprechen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um die gewünschten Geschäfts- und Kundenergebnisse zu erzielen.

Schaffen Sie ein gemeinsames Verständnis: Sorgen Sie dafür, dass alle Beteiligten die Geschäftsfunktionen des Workloads und die Rollen der einzelnen Teams bei den Workload-spezifischen betrieblichen Abläufen kennen. Außerdem sollte bekannt sein, wie diese Faktoren Ihre gemeinsamen Geschäftsziele mit internen und externen Kunden beeinflussen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP03 Implementieren von Feedback-Schleifen

OPS01-BP02 Bedürfnisse interner Kunden bewerten

Binden Sie alle wichtigen Stakeholder ein, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams, um zu bestimmen, welche Bereiche verstärkt auf die Bedürfnisse der internen

Kunden ausgerichtet werden müssen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um geschäftliche Ergebnisse zu erzielen.

Gewünschtes Ergebnis:

- Anhand Ihrer etablierten Prioritäten können Sie erkennen, an welchen Stellen die Verbesserungsbemühungen konzentriert werden sollten (z. B. Teamfähigkeiten entwickeln, die Workload-Leistung verbessern, Kosten senken, Runbooks automatisieren oder die Überwachung ausbauen).
- Wenn sich Anforderungen ändern, aktualisieren Sie Ihre Prioritäten entsprechend.

Typische Anti-Muster:

- Sie haben sich entschieden, die Zuweisung von IP-Adressen für Ihre Produktteams zu ändern, um die Netzwerkverwaltung zu vereinfachen. Dabei haben Sie jedoch nicht mit den Mitarbeitern gesprochen. Sie wissen also nicht, welche Auswirkungen diese Änderung auf Ihre Produktteams haben wird.
- Sie implementieren ein neues Entwicklungstool, haben aber Ihre internen Kunden nicht einbezogen, um herauszufinden, ob das Tool benötigt wird oder mit den Abläufen der Kunden kompatibel ist.
- Sie implementieren ein neues Überwachungssystem, haben aber Ihre internen Kunden nicht kontaktiert, um herauszufinden, ob spezifische Überwachungs- oder Berichtsanforderungen vorliegen, die berücksichtigt werden sollten.

Vorteile der Einführung dieser bewährten Methode: Die Bewertung und das Verständnis interner Kundenbedürfnisse liefert die Grundlage dafür, wie Sie Ihre Anstrengungen zur Bereitstellung eines geschäftlichen Mehrwerts priorisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

- Verstehen Sie die geschäftlichen Anforderungen: Der geschäftliche Erfolg basiert auf gemeinsamen Zielen und der Kommunikation zwischen allen Stakeholdern, zu denen auch die Teams aus den Bereichen Geschäft, Entwicklung und Betrieb gehören.
- Überprüfen Sie die geschäftlichen Ziele, Anforderungen und Prioritäten interner Kunden: Führen Sie wichtige Stakeholder zusammen, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams,

um die Ziele, Anforderungen und Prioritäten interner Kunden zu besprechen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um die gewünschten Geschäfts- und Kundenergebnisse zu erzielen.

- Schaffen Sie ein gemeinsames Verständnis: Sorgen Sie dafür, dass alle Beteiligten die Geschäftsfunktionen des Workloads und die Rollen der einzelnen Teams bei den Workload-spezifischen betrieblichen Abläufen kennen. Außerdem sollte bekannt sein, wie diese Faktoren die gemeinsamen Geschäftsziele mit internen und externen Kunden beeinflussen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP03 Implementieren von Feedback-Schleifen

OPS01-BP03 Bewerten der Governance-Anforderungen

Governance bezeichnet die Reihe von Richtlinien, Regeln oder Rahmen, die ein Unternehmen nutzt, um die geschäftlichen Ziele zu erreichen. Die Governance-Anforderungen werden innerhalb Ihrer Organisation erstellt. Sie können sich darauf auswirken, welche Arten von Technologien Sie nutzen oder wie Sie Ihren Workload betreiben. Integrieren Sie die Governance-Anforderungen Ihrer Organisation in Ihren Workload. Konformität ist die Fähigkeit, nachzuweisen, dass Sie die Governance-Anforderungen implementiert haben.

Gewünschtes Ergebnis:

- Die Governance-Anforderungen werden in das Architekturdesign und den Betrieb Ihres Workloads integriert.
- Sie können nachweisen, dass Sie den Governance-Anforderungen nachkommen.
- Die Governance-Anforderungen werden regelmäßig überprüft und aktualisiert.

Typische Anti-Muster:

- Ihre Organisation verlangt Multi-Faktor-Authentifizierung für das Stammkonto. Sie haben diese Anforderung nicht implementiert und das Stammkonto wurde kompromittiert.
- Während des Entwurfs Ihres Workloads wählen Sie einen Instance-Typ, der nicht von der IT-Abteilung genehmigt wurde. Sie können Ihren Workload nicht starten und müssen ihn überarbeiten.

- Sie sind verpflichtet, über einen Plan für die Notfallwiederherstellung zu verfügen. Sie haben keinen Plan erstellt und Ihr Workload ist von einem längeren Ausfall betroffen.
- Ihr Team möchte neue Instances verwenden, Ihre Governance-Anforderungen wurden jedoch nicht aktualisiert, sodass die Instances nicht zulässig sind.

Vorteile der Nutzung dieser bewährten Methode:

- Durch das Erfüllen der Governance-Anforderungen wird Ihr Workload auf die größeren Organisationsrichtlinien abgestimmt.
- Die Governance-Anforderungen spiegeln Branchenstandards und bewährte Methoden für Ihre Organisation wider.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Ermitteln Sie Governance-Anforderungen, indem Sie mit Stakeholdern und Governance-Organisationen zusammenarbeiten. Integrieren Sie die Governance-Anforderungen in Ihren Workload. Seien Sie in der Lage, nachzuweisen, dass Sie den Governance-Anforderungen nachkommen.

Kundenbeispiel

Das Cloud-Operations-Team bei AnyCompany Retail arbeitet mit Stakeholdern im gesamten Unternehmen zusammen, um Governance-Anforderungen zu entwickeln. Beispielsweise wird SSH-Zugriff auf Amazon EC2-Instances verboten. Wenn Teams Systemzugriff benötigen, müssen Sie AWS Systems Manager Session Manager verwenden. Das Cloud-Operations-Team aktualisiert die Governance-Anforderungen regelmäßig, sobald neue Services verfügbar sind.

Implementierungsschritte

1. Identifizieren Sie die Stakeholder für Ihren Workload, einschließlich zentralisierter Teams.
2. Arbeiten Sie mit den Stakeholdern zusammen, um Governance-Anforderungen zu ermitteln.
3. Nachdem Sie eine Liste erstellt haben, ordnen Sie die Verbesserungspunkte entsprechend der Priorität und beginnen Sie damit, sie in Ihren Workload zu implementieren.
   a. Nutzen Sie Services wie AWS Config, um Governance-as-Code zu erstellen und zu überprüfen, ob die Governance-Anforderungen erfüllt werden.

   b. Wenn Sie AWS Organizations nutzen, können Sie Service-Kontrollrichtlinien verwenden, um die Governance-Anforderungen zu implementieren.

4. Stellen Sie Unterlagen bereit, die die Implementierung bestätigen.

Grad des Aufwands für den Implementierungsplan: mittel. Die Implementierung fehlender Governance-Anforderungen kann dazu führen, dass Sie Ihren Workload überarbeiten müssen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP04 Bewerten der Compliance-Anforderungen – Compliance ist wie Governance, stammt jedoch von außerhalb eines Unternehmens.

Zugehörige Dokumente:

- AWS Management and Governance Cloud Environment Guide (AWS-Leitfaden zur Verwaltung und Governance der Cloud-Umgebung)
- Best Practices for AWS Organizations Service Control Policies in a Multi-Account Environment (Bewährte Methoden für AWS Organizations-Service-Kontrollrichtlinien in einer Umgebung mit mehreren Konten)
- Governance in the AWS Cloud: The Right Balance Between Agility and Safety (Governance in der AWS Cloud: Das richtige Gleichgewicht zwischen Agilität und Sicherheit)
- What is Governance, Risk, And Compliance (GRC)? (Was ist Governance, Risiko und Compliance (GRC)?)

Zugehörige Videos:

- AWS Management and Governance: Configuration, Compliance, and Audit - AWS Online Tech Talks (Verwaltung und Governance in AWS: Konfiguration, Compliance und Audit – AWS Online Tech Talks)
- AWS re:Inforce 2019: Governance for the Cloud Age (DEM12-R1) (AWS re:Inforce 2019: Governance für das Cloud-Zeitalter (DEM12-R1))
- AWS re:Invent 2020: Achieve compliance as code using AWS Config(AWS re:Invent 2020: Mit AWS Config Compliance als Code erzielen)

- AWS re:Invent 2020: Agile governance on AWS GovCloud (US)(AWS re:Invent 2020: Agile Governance in AWS GovCloud (US))

Zugehörige Beispiele:

- AWS Config Conformance Pack Samples (AWS Config-Conformance-Pack-Beispielvorlagen)

Zugehörige Services:

- AWS Config
- AWS Organizations – Service-Kontrollrichtlinien

OPS01-BP04 Bewerten der Compliance-Anforderungen

Regulatorische, branchenspezifische und interne Compliance-Anforderungen sind ein wichtiger Faktor, wenn Sie die Prioritäten Ihrer Organisation definieren. Ihr Compliance-Regelwerk hindert Sie möglicherweise daran, spezifische Technologien oder geografische Standorte zu nutzen. Wenden Sie die erforderliche Sorgfalt an, wenn keine externen Compliance-Regelwerke identifiziert sind. Erstellen Sie Audits oder Berichte, die die Compliance bestätigen.

Wenn Sie damit werben, dass Ihr Produkt bestimmte Compliance-Standards erfüllt, benötigen Sie einen internen Prozess zur kontinuierlichen Gewährleistung der Compliance. Beispiele für Compliance-Standards sind PCI DSS, FedRamp und HIPAA. Die geltenden Compliance-Standards werden durch verschiedene Faktoren bestimmt, beispielsweise dadurch, welche Datentypen von der Lösung gespeichert oder gesendet werden und welche geografischen Regionen die Lösung unterstützt.

Gewünschtes Ergebnis:

- Die regulatorischen, branchenspezifischen und internen Compliance-Anforderungen werden bei der Auswahl der Architektur berücksichtigt.
- Sie können die Compliance bestätigen und Audit-Berichte erstellen.

Typische Anti-Muster:

- Teile Ihres Workloads fallen unter das Regelwerk des Payment Card Industry Data Security Standard (PCI-DSS), Ihr Workload speichert Kreditkartendaten jedoch unverschlüsselt.

- Ihren Software-Entwicklern und -Architekten ist das Compliance-Regelwerk, das Ihre Organisation einhalten muss, nicht bekannt.
- Das jährliche Audit Systems and Organizations Control (SOC2) Type II steht bevor und Sie können nicht nachweisen, dass Kontrollelemente implementiert sind.

Vorteile der Nutzung dieser bewährten Methode:

- Die Bewertung und das Verständnis der Compliance-Anforderungen für Ihren Workload liefern die Grundlage dafür, wie Sie Ihre Anstrengungen zur Bereitstellung eines geschäftlichen Mehrwerts priorisieren.
- Sie wählen die Ihrem Compliance-Regelwerk entsprechenden Standorte und Technologien.
- Indem Sie Ihren Workload so entwerfen, dass Überprüfungen möglich sind, können Sie nachweisen, dass Sie das Compliance-Regelwerk einhalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Wenn Sie diese bewährte Methode implementieren, bedeutet dies, dass Sie Compliance-Anforderungen in den Entwurfsprozess für Ihre Architektur integrieren. Ihren Teammitgliedern ist das erforderliche Compliance-Regelwerk bekannt. Sie bestätigen Ihre Compliance mit diesem Regelwerk.

Kundenbeispiel

AnyCompany Retail speichert Kreditkarteninformationen für Kunden. Die Entwickler im Team für die Kartenspeicherung wissen, dass sie das PCI-DSS-Regelwerk einhalten müssen. Sie haben Schritte unternommen, um nachzuweisen, dass die Kreditkarteninformationen in Übereinstimmung mit dem PCI-DSS-Regelwerk sicher gespeichert und aufgerufen werden. Jedes Jahr arbeiten sie mit dem Sicherheitsteam zusammen, um die Compliance zu bestätigen.

Implementierungsschritte

1. Arbeiten Sie mit Ihrem Sicherheits- und Governance-Team zusammen, um zu ermitteln, welche branchenspezifischen, regulatorischen oder internen Compliance-Regelwerke Ihr Workload einhalten muss. Integrieren Sie die Compliance-Regelwerke in Ihren Workload.
   a. Bestätigen Sie die durchgängige Compliance von AWS-Ressourcen mit Services wie AWS Compute Optimizer und AWS Security Hub.

2. Informieren Sie Ihre Teammitglieder über die Compliance-Anforderungen, damit diese den Workload in Übereinstimmung mit den Anforderungen betreiben und weiterentwickeln können. Die Compliance-Anforderungen sollten bei architektur- und technologiebezogenen Entscheidungen berücksichtigt werden.
3. Je nach Compliance-Regelwerk müssen Sie möglicherweise einen Audit- oder Compliance-Bericht erstellen. Arbeiten Sie mit Ihrer Organisation zusammen, um diesen Prozess so weit wie möglich zu automatisieren.
   a. Verwenden Sie Services wie AWS Audit Manager, um die Compliance zu bestätigen und Audit-Berichte zu erstellen.
   b. AWS-Dokumente zu Sicherheit und Compliance können mit AWS Artifact heruntergeladen werden.

Grad des Aufwands für den Implementierungsplan: mittel. Die Implementierung von Compliance-Regelwerken kann eine Herausforderung darstellen. Das Erstellen von Audit-Berichten oder Compliance-Dokumenten sorgt für zusätzlichen Aufwand.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP03 Identifizieren und Validieren von Kontrollzielen – Sicherheitskontrollziele sind ein wichtiger Bestandteil der allgemeinen Compliance.
- SEC01-BP06 Automatisieren von Tests und Validierung von Sicherheitskontrollen in Pipelines – Validieren Sie die Sicherheitskontrollen als Teil Ihrer Pipelines. Sie können auch eine Compliance-Dokumentation für neue Änderungen erstellen.
- SEC07-BP02 Definieren von Datenschutzkontrollen – Viele Compliance-Regelwerke umfassen Richtlinien für den Umgang mit und die Speicherung von Daten.
- SEC10-BP03 Vorbereiten forensischer Funktionen – Forensische Funktionen können mitunter bei Prüfungen der Compliance verwendet werden.

Zugehörige Dokumente:

- AWS Compliance Center
- AWS-Compliance-Ressourcen
- AWS Risk and Compliance Whitepaper (AWS-Whitepaper: Risiko und Compliance)

- AWS-Modell der geteilten Verantwortung
- AWS-Services im Rahmen des Compliance-Programms

Zugehörige Videos:

- AWS re:Invent 2020: Achieve compliance as code using AWS Compute Optimizer(AWS re:Invent 2020: Mit AWS Compute Optimizer Compliance als Code erzielen)
- AWS re:Invent 2021 - Cloud compliance, assurance, and auditing (AWS re:Invent 2021 – Cloud-Compliance, Sicherheit und Prüfungen)
- AWS Summit ATL 2022 - Implementing compliance, assurance, and auditing on AWS (COP202) (AWS Summit ATL 2022 – Compliance, Sicherheit und Prüfungen für AWS implementieren (COP202))

Zugehörige Beispiele:

- Bewährte Methoden für PCI DSS und AWS Foundational Security auf AWS

Zugehörige Services:

- AWS Artifact
- AWS Audit Manager
- AWS Compute Optimizer
- AWS Security Hub

OPS01-BP05 Bewerten der Bedrohungsszenarien

Bewerten Sie Bedrohungen für das Unternehmen (z. B. Wettbewerb, Geschäftsrisiken und -verpflichtungen, operative Risiken und Bedrohungen der Informationssicherheit) und pflegen Sie aktuelle Informationen in einem Risikoregister. Berücksichtigen Sie die Auswirkungen von Risiken, wenn Sie bestimmen, auf welche Bereiche die Anstrengungen fokussiert werden sollen.

Das Well-Architected Framework legt den Schwerpunkt auf Lernen, Messen und Verbessern. Es bietet einen konsistenten Ansatz, mit dem Sie Architekturen bewerten und Designs implementieren können, die sich im Laufe der Zeit skalieren lassen. AWS stellt das AWS Well-Architected Tool bereit, mit dem Sie Ihren Ansatz vor der Entwicklung, den Status Ihrer Workloads vor der Produktion und den Status Ihrer Workloads in der Produktion überprüfen können. Sie können sie mit den neuesten

bewährten Methoden für die AWS-Architektur vergleichen, den Gesamtstatus Ihrer Workloads überwachen und Einblicke in potenzielle Risiken erhalten.

AWS-Kunden haben auch die Möglichkeit, die Architektur ihrer geschäftskritischen Workloads auf die Einhaltung bewährter AWS-Methoden hin überprüfen zu lassen (Well-Architected Review). Für Kunden mit Enterprise Support wird eine Überprüfung des Betriebs (Operations Review) angeboten. Damit haben sie die Möglichkeit, Lücken in ihrem Cloud-Ansatz aufzuzeigen.

Aufgrund der teamübergreifenden Natur dieser Überprüfungen erhalten Sie ein allgemeines Verständnis Ihrer Workloads und können erkennen, wie Team-Rollen zum Erfolg beitragen. Die bei den Überprüfungen gefundenen Punkte können Ihnen beim Festlegen Ihrer Prioritäten helfen.

AWS Trusted Advisor bietet als Tool Zugriff auf verschiedene wichtige Prüfungen, die Optimierungsempfehlungen ausgeben. Diese Informationen können Ihnen beim Festlegen Ihrer Prioritäten helfen. Kunden mit Business und Enterprise Support erhalten Zugriff auf weitere Prüfungen in den Bereichen Sicherheit, Zuverlässigkeit, Leistung und Kostenoptimierung, die beim Festlegen von Prioritäten noch hilfreicher sind.

Gewünschtes Ergebnis:

- Sie überprüfen regelmäßig Well-Architected und Trusted Advisor-Ergebnisse und reagieren darauf.
- Sie sind über den neuesten Patch-Status Ihrer Services informiert.
- Sie kennen das Risiko und die Auswirkungen bekannter Bedrohungen und handeln entsprechend.
- Sie implementieren bei Bedarf Abhilfemaßnahmen.
- Sie kommunizieren Aktionen und Kontext.

Typische Anti-Muster:

- Sie verwenden in Ihrem Produkt eine alte Version einer Softwarebibliothek. Ihnen ist nicht bewusst, dass für die Bibliothek Sicherheitsaktualisierungen vorliegen, mit denen Probleme behoben werden, die unbeabsichtigte Auswirkungen auf Ihren Workload haben können.
- Ein Mitbewerber hat soeben eine Version seines Produkts veröffentlicht, in der viele Probleme behoben werden, die Kunden an Ihrem Produkt bemängeln. Die Behebung dieser bekannten Probleme hatte für Sie bisher keine Priorität.
- Regulierungsbehörden nehmen Unternehmen wie Ihres, die nicht den gesetzlichen Compliance-Anforderungen entsprechen, verstärkt ins Visier. Sie haben Ihre ausstehenden Compliance-Anforderungen nicht priorisiert.

Vorteile der Einführung dieser bewährten Methode: Sie identifizieren und verstehen die Bedrohungen für Ihre Organisation und Ihren Workload, was Ihnen bei der Entscheidung hilft, welche Bedrohungen angegangen werden müssen, wo die Prioritäten liegen und welche Ressourcen dafür erforderlich sind.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

- Bewerten Sie die Bedrohungslandschaft: Bewerten Sie Bedrohungen für das Unternehmen (z. B. Konkurrenz, Geschäftsrisiken und -verpflichtungen, operative Risiken und Bedrohungen der Informationssicherheit), damit Sie die jeweiligen Auswirkungen berücksichtigen können, wenn Sie bestimmen, auf welche Bereiche die operativen Anstrengungen konzentriert werden sollten.
  - Aktuelle AWS-Sicherheitsmitteilungen
  - AWS Trusted Advisor
- Verwalten Sie ein Bedrohungsmodell: Erstellen und verwalten Sie ein Bedrohungsmodell, in dem potenzielle Bedrohungen, geplante und vorhandene Maßnahmen und deren Priorität festgehalten werden. Untersuchen Sie, wie wahrscheinlich es ist, dass sich Bedrohungen als Vorfälle äußern, wie hoch die Kosten für die Wiederherstellung nach diesen Vorfällen sind, welche Schäden zu erwarten sind und wie viel es kostet, diese Vorfälle zu verhindern. Überarbeiten Sie die Prioritäten, wenn sich der Inhalt des Bedrohungsmodells ändert.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells

Zugehörige Dokumente:

- AWS Cloud-Compliance
- Aktuelle AWS-Sicherheitsmitteilungen
- AWS Trusted Advisor

Zugehörige Videos:

- AWS re:Inforce 2023 – Ein Tool zur Verbesserung Ihrer Bedrohungsmodellierung

OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken

Konkurrierende Interessen mehrerer Parteien können eine Herausforderung darstellen, wenn es darum geht, Anstrengungen zu priorisieren, Fähigkeiten aufzubauen und Ergebnisse zu erzielen, die auf die Geschäftsstrategien abgestimmt sind. So können Sie möglicherweise aufgefordert werden, die Markteinführung neuer Features zu beschleunigen, anstatt die Kosten für die IT-Infrastruktur zu optimieren. Dies kann dazu führen, dass die Interessen zweier Parteien miteinander in Widerspruch stehen. In solchen Situationen muss eine höhere Stelle hinzugezogen werden, um eine Entscheidung zur Lösung des Konflikts zu treffen. Daten sind erforderlich, um den Entscheidungsprozess von emotionalen Komponenten zu befreien.

Ähnliche Herausforderungen können auf taktischer Ebene auftreten. Beispielsweise kann die Wahl zwischen relationalen oder nicht relationalen Datenbanktechnologien erhebliche Auswirkungen auf den Betrieb einer Anwendung haben. Daher ist es wichtig, die voraussichtlichen Ergebnisse verschiedener Entscheidungen zu verstehen.

AWS kann Ihnen helfen, Ihre Teams über AWS und die verfügbaren Services zu schulen, sodass alle Mitarbeiter wissen, welche Auswirkungen ihre Entscheidungen auf Ihren Workload haben können. Nutzen Sie bei der Schulung Ihrer Teams die vom Support (AWS Knowledge Center, AWS-Diskussionsforen und Support Center) bereitgestellten Ressourcen und AWS-Dokumente. Bei weiteren Fragen wenden Sie sich bitte an Support.

AWS teilt auch bewährte operative Methoden und Muster in der Amazon Builders' Library. Eine Vielzahl weiterer nützlicher Informationen finden Sie im AWS-Blogund im offiziellen AWS-Podcast.

Gewünschtes Ergebnis: Ein klar definiertes Framework zur Entscheidungsfindung, das das Treffen wichtiger Entscheidungen auf allen Ebenen Ihrer Cloud-Bereitstellungsorganisation erleichtert. Dieses Framework umfasst Features wie ein Risikoregister, definierte Rollen mit Entscheidungsbefugnissen und definierte Modelle für die einzelnen Entscheidungsebenen. Dieses Framework legt im Voraus fest, wie Konflikte gelöst werden, welche Daten präsentiert werden müssen und wie Optionen priorisiert werden, sodass Sie einmal gefasste Beschlüsse sofort umsetzen können. Das Framework zur Entscheidungsfindung beinhaltet einen standardisierten Ansatz zur Überprüfung und Abwägung der Vorteile und Risiken einzelner Entscheidungen, um die Tragweite etwaiger Kompromisse abzuschätzen. Dazu können externe Faktoren gehören wie die Einhaltung gesetzlicher Vorschriften.

Typische Anti-Muster:

- Ihre Investoren fordern, dass Sie die Compliance mit Payment Card Industry Data Security Standards (PCI DSS) nachweisen. Sie denken nicht über einen möglichen Kompromiss zwischen der Erfüllung dieser Anfrage und der Fortsetzung Ihrer derzeitigen Entwicklungsaktivitäten nach. Stattdessen fahren Sie mit der Entwicklung fort, ohne einen Compliance-Nachweis zu erbringen. Ihre Investoren beenden die Unterstützung Ihres Unternehmens, da sie Bedenken bezüglich der Sicherheit Ihrer Plattform und ihrer Investitionen haben.
- Sie haben sich entschieden, eine Bibliothek einzubinden, die einer Ihrer Entwickler „im Internet entdeckt“ hat. Sie haben keine Bewertung der Risiken durchgeführt, die die Einführung dieser Bibliothek aus einer unbekannten Quelle bergen kann, und wissen nicht, ob sie Schwachstellen oder schädlichen Code enthält.
- Die ursprüngliche geschäftliche Begründung für Ihre Migration basierte auf der Modernisierung von 60 % Ihrer Anwendungsworkloads. Aufgrund technischer Schwierigkeiten wurde jedoch beschlossen, nur 20 % zu modernisieren. Dies führte langfristig zu einer Reduzierung der geplanten Leistungen, zu einem erhöhten Aufwand für die Infrastrukturteams bei der manuellen Wartung von Legacy-Systemen und zu einer stärkeren Abhängigkeit von der Entwicklung neuer Fähigkeiten in Ihren Infrastrukturteams, die diese Änderung nicht geplant hatten.

Vorteile der Einführung dieser bewährten Methode: Umfassende Abstimmung und Unterstützung der Geschäftsprioritäten auf Vorstandsebene, Kenntnis der Erfolgsrisiken, Treffen fundierter Entscheidungen und angemessenes Handeln, wenn Risiken die Erfolgsaussichten trüben. Indem Sie die Auswirkungen und Konsequenzen Ihrer Entscheidungen verstehen, können Sie Ihre Optionen priorisieren und Führungskräfte schneller zu einer Einigung bringen, was zu besseren Geschäftsergebnissen führt. Wenn Sie die Vorteile Ihrer Entscheidungen erkennen und sich der Risiken für Ihre Organisation bewusst sind, können Sie datengestützte Entscheidungen treffen, anstatt sich auf Anekdoten verlassen zu müssen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Abwägung von Nutzen und Risiken sollte von einem Leitungsorgan übernommen werden, das die Anforderungen für wichtige Entscheidungen festlegt. Sie möchten, dass Entscheidungen basierend auf ihrem Nutzen für die Organisation getroffen und priorisiert werden und die damit verbundenen Risiken bekannt sind. Präzise Informationen bilden die Grundlage für die Entscheidungen Ihrer Organisation. Diese sollten auf soliden Messungen beruhen und durch branchenübliche Verfahren der Kosten-Nutzen-Analyse definiert werden. Damit Entscheidungen auf diese Art getroffen werden können, müssen Sie ein Gleichgewicht zwischen zentralisierter und dezentralisierter Autorität

herstellen. Es gibt immer einen Kompromiss. Daher ist es wichtig zu verstehen, wie sich jede Entscheidung auf definierte Strategien und angestrebte Geschäftsergebnisse auswirkt.

Implementierungsschritte

1. Formalisieren Sie die Verfahren zur Leistungsmessung innerhalb eines ganzheitlichen Cloud-Governance-Frameworks.
   a. Bringen Sie die zentrale Kontrolle der Entscheidungsfindung in Einklang mit konkreten dezentralen Entscheidungsbefugnissen.
   b. Machen Sie sich bewusst, dass nicht für jeden Beschluss aufwendige Entscheidungsprozesse vonnöten sind, da sie Sie verlangsamen können.
   c. Integrieren Sie externe Faktoren in Ihren Entscheidungsprozess (wie Compliance-Anforderungen).
2. Richten Sie ein gemeinsames Framework zur Entscheidungsfindung für verschiedene Entscheidungsebenen ein, in dem festgelegt ist, wer Entscheidungen bei widersprüchlichen Interessen trifft.
   a. Zentralisieren Sie einseitige Entscheidungen, die irreversibel sein könnten.
   b. Lassen Sie leicht revidierbare Entscheidungen von Führungskräften auf niedrigerer Ebene treffen.
3. Machen Sie sich mit den Nutzen und Risiken vertraut und wägen Sie sie ab. Wägen Sie den Nutzen von Entscheidungen gegen die damit einhergehenden Risiken ab.
   a. Ermitteln von Vorteilen: Ermitteln Sie die Vorteile auf Basis der geschäftlichen Ziele, Anforderungen und Prioritäten. Beispiele hierfür sind die Auswirkungen auf den Business Case, die Markteinführungszeit, Sicherheit, Zuverlässigkeit, Leistung und Kosten.
   b. Ermitteln von Risiken: Ermitteln Sie die Risiken auf Basis der geschäftlichen Ziele, Anforderungen und Prioritäten. Zu diesen Prioritäten zählen beispielsweise eine kurze Markteinführungszeit, Sicherheit, Zuverlässigkeit, Leistung und Kosten.
   c. Abwägen von Vorteilen und Risiken und Treffen fundierter Entscheidungen: Ermitteln Sie die Auswirkungen von Vorteilen und Risiken basierend auf den Zielen, Bedürfnissen und Prioritäten Ihrer wichtigsten Stakeholder, zu denen auch die Bereiche Betriebswirtschaft, Entwicklung und Operationen zählen. Bewerten Sie den Wert eines Vorteils anhand der Wahrscheinlichkeit, dass sich das Risiko tatsächlich bewahrheitet, sowie der Kosten der jeweiligen Auswirkungen. Eine schnellere Markteinführung zu Lasten der Zuverlässigkeit könnte beispielsweise einen Wettbewerbsvorteil bedeuten. Wenn jedoch Probleme mit der Zuverlässigkeit auftreten, kann dies zu einer verringerten Betriebszeit führen.

4. Setzen Sie wichtige Entscheidungen programmatisch um, um die Einhaltung von Compliance-Anforderungen zu automatisieren.
5. Nutzen Sie branchenübliche Frameworks und Funktionen wie Value Stream Analysis und LEAN, um die aktuelle Leistung und Geschäftsmetriken abzubilden und Iterationen der Fortschritte zur Verbesserung dieser Metriken zu definieren.

Aufwand des Implementierungsplans: mittel bis hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP05 Bewerten der Bedrohungsszenarien

Zugehörige Dokumente:

- Elemente der Day-1-Kultur von Amazon | Schnell gute Entscheidungen treffen
- Cloud-Governance
- Cloud-Umgebung für Management und Governance
- Governance in der Cloud und im digitalen Zeitalter: Teil 1 und 2

Zugehörige Videos:

- Podcast | Jeff Bezos | Über das Treffen von Entscheidungen

Zugehörige Beispiele:

- Fundierte Entscheidungen mithilfe von Daten treffen (The DevOps Sagas)
- Nutzung von Wertstromanalysen in der Entwicklung zur Identifizierung von Einschränkungen bei DevOps-Ergebnissen

OPS 2. Wie strukturieren Sie Ihr Unternehmen, um die gewünschten Geschäftsergebnisse zu erzielen?

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams sollten ihre Rolle für den Erfolg anderer Teams und die Rolle anderer Teams für ihren Erfolg verstehen und

gemeinsame Ziele haben. Wenn sie Verantwortlichkeit, Zuständigkeit und Entscheidungsfindung nachvollziehen können und wissen, wer dazu berechtigt ist, Entscheidungen zu treffen, können ihre Anstrengungen fokussiert und der Nutzen Ihrer Teams maximiert werden.

Bewährte Methoden

- OPS02-BP01 Ressourcen haben feste Verantwortliche
- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer
- OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind
- OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten
- OPS02-BP05 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden
- OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt

OPS02-BP01 Ressourcen haben feste Verantwortliche

Die Ressourcen für Ihren Workload müssen für die Änderungskontrolle, die Fehlerbehebung und andere Funktionen feste Verantwortliche haben. Verantwortliche werden für Workloads, Konten, Infrastruktur, Plattformen und Anwendungen zugewiesen. Die Verantwortlichkeit wird mit Tools wie einem Zentralverzeichnis oder Metadaten zu Ressourcen erfasst. Der Unternehmenswert der Komponenten bestimmt, welche Prozesse und Verfahren auf diese angewendet werden.

Gewünschtes Ergebnis:

- Mithilfe von Metadaten oder einem Zentralverzeichnis werden feste Verantwortliche für die Ressourcen identifiziert.
- Die Teammitglieder können erkennen, wer für eine bestimmte Ressource verantwortlich ist.
- Konten haben wenn möglich einen festen Verantwortlichen.

Typische Anti-Muster:

- Die alternativen Kontakte für Ihre AWS-Konten sind nicht eingepflegt.
- Die Ressourcen sind nicht mit Tags markiert, die kennzeichnen, wer dafür verantwortlich ist.
- Sie haben eine ITSM-Warteschlange ohne E-Mail-Zuordnung.
- Zwei Teams haben sich überschneidende Verantwortlichkeit für einen wichtigen Teil der Infrastruktur.

Vorteile der Nutzung dieser bewährten Methode:

- Dank der zugewiesenen Verantwortlichkeit ist die Änderungskontrolle ganz einfach.
- Wenn Probleme auftreten, können die richtigen Verantwortlichen einbezogen werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Definieren Sie, was Verantwortlichkeit für die Ressourcen-Anwendungsfälle in Ihrer Umgebung bedeutet. Verantwortlichkeit kann bedeuten, Änderungen an der Ressource zu beaufsichtigen, die Ressource während der Fehlerbehebung zu unterstützen oder die finanzielle Verantwortung zu tragen. Legen Sie Verantwortliche für Ressourcen fest und dokumentieren Sie diese. Die Angaben sollten den Namen, die Kontaktinformationen, die Organisation und das Team beinhalten.

Kundenbeispiel

Bei AnyCompany Retail bezeichnet die Verantwortlichkeit das Team oder die Person, das/die für Änderungen und Support für Ressourcen verantwortlich ist. Das Unternehmen verwendet AWS Organizations für die Verwaltung seiner AWS-Konten. Die alternativen Kontakte für die Konten werden mit Gruppenpostfächern konfiguriert. Jede ITSM-Warteschlange ist einem E-Mail-Alias zugeordnet. Tags kennzeichnen, wer für AWS-Ressourcen verantwortlich ist. Für andere Plattformen und Infrastruktur gibt es eine Wiki-Seite, auf der die Verantwortlichkeiten und die Kontaktinformationen angegeben sind.

Implementierungsschritte

1. Beginnen Sie damit, die Verantwortlichkeiten für Ihre Organisation zu definieren. Verantwortlichkeit kann bedeuten, wer für das Risiko für die Ressource oder für Änderungen an der Ressource verantwortlich ist oder wer die Ressource im Fall einer Fehlerbehebung unterstützt. Verantwortlichkeit kann auch die finanzielle oder administrative Verantwortlichkeit für die Ressource umfassen.
2. Verwenden Sie AWS Organizations zum Verwalten der Konten. Sie können die alternativen Kontakte für Ihre Konten zentral verwalten.
   a. Durch die Verwendung von E-Mail-Adressen und Telefonnummern des Unternehmens als Kontaktdaten können Sie auch dann auf sie zugreifen, wenn die Personen, zu denen sie gehören, nicht mehr Teil Ihrer Organisation sind. Erstellen Sie beispielsweise separate E-Mail-Verteilerlisten für die Abrechnung, die Produktion und die Sicherheit und konfigurieren Sie sie

in jedem aktiven AWS-Konto als Abrechnungs-, Sicherheits- und Produktionskontakte. Mehrere Personen erhalten AWS-Benachrichtigungen und können auch dann reagieren, wenn jemand im Urlaub ist, die Rolle wechselt oder das Unternehmen verlässt.

   b. Wenn ein Konto nicht von AWS Organizations verwaltet wird, tragen die alternativen Kontakte für Konten dazu bei, dass AWS nötigenfalls mit den richtigen Mitarbeitern in Kontakt treten kann. Konfigurieren Sie die alternativen Kontakte für ein Konto so, dass sie auf eine Gruppe verweisen, und nicht auf eine Einzelperson.

3. Verwenden Sie Tags, um die Verantwortlichen für AWS-Ressourcen zu kennzeichnen. Sie können die Verantwortlichen und ihre Kontaktdaten in verschiedenen Tags angeben.
   a. Mit Regeln in AWS Config können Sie erzwingen, dass die Ressourcen die erforderlichen Tags zur Verantwortlichkeit aufweisen.
   b. Ausführliche Anleitungen zur Entwicklung einer Tagging-Strategie für Ihre Organisation finden Sie im AWS-Whitepaper „Bewährte Methoden für das Tagging“.
4. Verwenden Sie Amazon Q Business, einen Gesprächsassistenten, der generative KI nutzt, um die Produktivität der Belegschaft zu steigern, Fragen zu beantworten und Aufgaben basierend auf Informationen in Ihren Unternehmenssystemen zu erledigen.
   a. Verbinden Sie Amazon Q Business mit der Datenquelle Ihres Unternehmens. Amazon Q Business bietet vorgefertigte Konnektoren zu über 40 unterstützten Datenquellen, darunter Amazon Simple Storage Service (Amazon S3), Microsoft SharePoint, Salesforce und Atlassian Confluence. Weitere Informationen finden Sie unter Amazon Q Business-Konnektoren.
5. Erstellen Sie für andere Ressourcen, Plattformen und Infrastruktur eine Dokumentation mit Informationen zur jeweiligen Verantwortlichkeit. Diese sollte für alle Teammitglieder zugänglich sein.

Aufwand des Implementierungsplans: niedrig. Nutzen Sie die Kontaktinformationen zum Konto sowie Tags, um die Verantwortlichkeit für AWS-Ressourcen zuzuweisen. Für andere Ressourcen können Sie beispielsweise eine einfache Tabelle in einem Wiki verwenden, um die Verantwortlichkeit und Kontaktinformationen zu erfassen, oder nutzen Sie ein ITSM-Tool, um die Verantwortlichkeit zuzuordnen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer
- OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten

Zugehörige Dokumente:

- AWS-Account Management – Aktualisieren der Kontaktinformationen
- AWS Organizations – Aktualisieren alternativer Kontakte in Ihrer Organisation
- Bewährte Methoden für das Tagging von AWS – Whitepaper
- Erstellung privater und sicherer Apps mit generativer KI für Unternehmen mit Amazon Q Business und AWS IAM Identity Center
- Amazon Q Business, jetzt allgemein verfügbar, zur Steigerung der Mitarbeiterproduktivität mithilfe generativer KI
- Blog zum Thema AWS Cloud-Operations und -Migrationen – Implementierung automatisierter und zentralisierter Tagging-Kontrollen mit AWS Config und AWS Organizations
- Blog zum Thema AWS-Sicherheit – Erweitern von Pre-Commit-Hooks mit AWS CloudFormation Guard
- Blog zum Thema AWS DevOps – Integration von AWS CloudFormation Guard in CI/CD-Pipelines

Zugehörige Workshops:

- AWS-Workshop – Tagging

Zugehörige Beispiele:

- AWS-Config-Regeln – Amazon EC2 mit erforderlichen Tags und gültigen Werten

Zugehörige Services:

- AWS-Config-Regeln – erforderliche Tags
- AWS Organizations

OPS02-BP02 Prozesse und Verfahren haben feste Besitzer

Verschaffen Sie sich einen Überblick darüber, wer für die Definition einzelner Prozesse und Verfahren zuständig ist, warum diese spezifischen Prozesse und Verfahren verwendet werden und warum diese Zuständigkeit besteht. Wenn Sie wissen, warum bestimmte Prozesse und Verfahren verwendet werden, können Sie Verbesserungsmöglichkeiten identifizieren.

Gewünschtes Ergebnis: Ihre Organisation verfügt über gut definierte und verwaltete Prozesse und Verfahren für betriebliche Aufgaben. Der Prozess und die Verfahren werden an einem zentralen Ort gespeichert und stehen Ihren Teammitgliedern zur Verfügung. Prozesse und Verfahren werden regelmäßig aktualisiert, wobei die Zuständigkeit eindeutig zugewiesen wird. Wo möglich, werden Skripte, Vorlagen und Automatisierungsdokumente als Code implementiert.

Typische Anti-Muster:

- Prozesse sind nicht dokumentiert. Es können fragmentierte Skripte auf isolierten Bedienerarbeitsplätzen existieren.
- Das Wissen über den Umgang mit Skripten wird von wenigen Personen oder informell als Teamwissen vermittelt.
- Ein veralteter Prozess muss aktualisiert werden, aber die Zuständigkeit für die Aktualisierung ist unklar, und der ursprüngliche Autor gehört nicht mehr zur Organisation.
- Prozesse und Skripte sind nicht auffindbar und daher nicht sofort verfügbar, wenn sie benötigt werden (z. B. als Reaktion auf einen Vorfall).

Vorteile der Nutzung dieser bewährten Methode:

- Prozesse und Verfahren unterstützen Sie bei der Bewältigung Ihrer Workloads.
- Neue Teammitglieder werden schneller handlungsfähig.
- Die Zeit bis zur Behebung von Vorfällen wird reduziert.
- Verschiedene Teammitglieder (und Teams) können dieselben Prozesse und Verfahren auf einheitliche Weise verwenden.
- Teams können ihre Prozesse durch wiederholbare Prozesse skalieren.
- Standardisierte Prozesse und Verfahren tragen dazu bei, die Auswirkungen der Übertragung von Workload-Verantwortlichkeiten zwischen Teams abzumildern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

- Prozesse und Verfahren haben feste Besitzer, die für ihre Definition verantwortlich sind.
  - Identifizieren Sie die Betriebsaktivitäten, die zur Unterstützung Ihrer Workloads durchgeführt werden. Dokumentieren Sie diese Aktivitäten an einem auffindbaren Ort.

  - Legen Sie die Person oder Personen fest, die für die Spezifikation einer Aktivität verantwortlich sind. Sie sind dafür verantwortlich, sicherzustellen, dass die Aktivität von einem ausreichend qualifizierten Teammitglied durchgeführt wird, das die entsprechenden Berechtigungen, Zugriffsrechte und Tools hat. Wenn bei der Durchführung dieser Aktivität Probleme auftreten, sind die zuständigen Teammitglieder dafür zuständig, detailliertes Feedback bereitzustellen, damit die Aktivität verbessert werden kann.
  - Erfassen Sie die Zuständigkeit in den Metadaten des Aktivitätsartefakts durch Services wie AWS Systems Manager, durch Dokumente und AWS Lambda. Erfassen Sie die Ressourcenzuständigkeit mithilfe von Tags oder Ressourcengruppen und geben Sie Zuständigkeits- und Kontaktinformationen an. Verwenden Sie AWS Organizations, um Markierungsrichtlinien zu erstellen sowie Zuständigkeits- und Kontaktinformationen zu erfassen.
- Mit der Zeit sollten diese Verfahren so weiterentwickelt werden, dass sie als Code ausgeführt werden können, damit weniger menschliche Eingriffe erforderlich sind.
  - Erwägen Sie beispielsweise AWS Lambda-Funktionen, CloudFormation-Vorlagen oder AWS Systems Manager-Automatisierungsdokumente.
  - Führen Sie die Versionskontrolle in den entsprechenden Repositorys durch.
  - Fügen Sie geeignetes Ressourcen-Tagging hinzu, damit Eigentümer und Dokumentation leicht identifiziert werden können.

Kundenbeispiel

AnyCompany Retail legt fest, dass das Team oder die Person, die für die Prozesse einer Anwendung oder einer Gruppe von Anwendungen (die gemeinsame architektonische Praktiken und Technologien nutzen) zuständig ist, der Besitzer ist. Zunächst werden der Prozess und die Verfahren in Form von schrittweisen Anleitungen im Dokumentenverwaltungssystem dokumentiert, die über Tags für das AWS-Konto, das die Anwendung hostet, und für bestimmte Ressourcengruppen innerhalb des Kontos auffindbar sind. Das Unternehmen verwendet AWS Organizations für die Verwaltung seiner AWS-Konten. Im Laufe der Zeit werden diese Prozesse in Code umgewandelt und Ressourcen werden mithilfe von Infrastructure as Code (z. B. CloudFormation oder AWS Cloud Development Kit (AWS CDK)-Vorlagen) definiert. Die Betriebsprozesse werden zu Automatisierungsdokumenten in AWS Systems Manager- oder AWS Lambda-Funktionen, die als geplante Aufgaben, als Reaktion auf Ereignisse wie AWS CloudWatch-Alarme oder AWS EventBridge-Ereignisse oder durch Anfragen innerhalb einer IT-Servicemanagement-Plattform (ITSM) gestartet werden können. Alle Prozesse sind mit Tags versehen, um die Zuständigkeit zu identifizieren. Die Dokumentation für die Automatisierung

und den Prozess wird auf den Wiki-Seiten verwaltet, die vom Code-Repository für den Prozess generiert werden.

Implementierungsschritte

1. Dokumentieren Sie die bestehenden Prozesse und Verfahren.
   a. Überprüfen Sie sie und halten Sie sie auf dem neuesten Stand.
   b. Identifizieren Sie einen Besitzer für jeden Prozess und jede Prozedur.
   c. Stellen Sie sie unter Versionskontrolle.
   d. Wenn möglich, nutzen Sie Prozesse und Verfahren für Workloads und Umgebungen mit gemeinsamen Architekturentwürfen.
2. Richten Sie Mechanismen für Feedback und Verbesserung ein.
   a. Definieren Sie Richtlinien dafür, wie oft Prozesse überprüft werden sollten.
   b. Definieren Sie Prozesse für Prüfende und Genehmigende.
   c. Implementieren Sie Probleme oder eine Ticket-Warteschlange, um Feedback zu geben und zu verfolgen.
   d. Wo immer es möglich ist, sollten Prozesse und Verfahren vorab von einem Gremium zur Genehmigung von Änderungen genehmigt und in eine Risikoklasse eingestuft werden.
3. Stellen Sie sicher, dass Prozesse und Verfahren für diejenigen, die sie ausführen müssen, zugänglich und auffindbar sind.
   a. Verwenden Sie Tags, um anzugeben, wo der Prozess und die Verfahren für den Workload aufgerufen werden können.
   b. Verwenden Sie aussagekräftige Fehler- und Ereignismeldungen, um die geeigneten Prozesse oder Verfahren zur Behebung eines Problems anzugeben.
   c. Verwenden Sie Wikis und Dokumentenmanagement und machen Sie Prozesse und Verfahren organisationsweit durchsuchbar.
4. Automatisieren Sie gegebenenfalls.
   a. Automatisierungen sollten entwickelt werden, wenn Services und Technologien eine API bereitstellen.
   b. Informieren Sie angemessen über Prozesse. Entwickeln Sie die Benutzerszenarien und Anforderungen, um diese Prozesse zu automatisieren.
   c. Messen Sie die erfolgreiche Nutzung Ihrer Prozesse und Verfahren und geben Sie dabei Probleme an, die eine iterative Verbesserung unterstützen.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP01 Ressourcen haben feste Verantwortliche
- OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten
- OPS11-BP04 Wissensmanagement

Zugehörige Dokumente:

- AWS Whitepaper – Einführung in DevOps in AWS
- AWS-Whitepaper – Bewährte Methoden für das Tagging von AWS-Ressourcen
- AWS-Whitepaper – Organisieren der AWS-Umgebung mithilfe mehrerer Konten
- Blog zum Thema AWS Cloud-Betrieb und -Migrationen – Entwicklung eines Cloud-Automatisierungsverfahrens für Operational Excellence: Bewährte Methoden von AWS Managed Services
- Blog zum Thema AWS Cloud-Operations und -Migrationen – Implementierung automatisierter und zentralisierter Tagging-Kontrollen mit AWS Config und AWS Organizations
- Blog zum Thema AWS-Sicherheit – Erweitern von Pre-Commit-Hooks mit AWS CloudFormation Guard
- Blog zum Thema AWS DevOps – Integration von AWS CloudFormation Guard in CI/CD-Pipelines

Zugehörige Workshops:

- Workshop zum Thema AWS Well-Architected Operational Excellence
- AWS-Workshop – Tagging

Zugehörige Videos:

- Automatisierung von IT-Abläufen in AWS
- AWS re:Invent 2020 – Automatisierung mit AWS Systems Manager
- AWS re:Inforce 2022 – Automatisierung der Patch-Verwaltung und -Compliance mit AWS (NIS306)
- Unterstützung durch AWS – Vertiefung in AWS Systems Manager

Zugehörige Services:

- AWS Systems Manager – Automatisierung
- AWS Service Management Connector

OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind

Verschaffen Sie sich einen Überblick darüber, wer für spezifische Aktivitäten in festgelegten Workloads verantwortlich ist und warum diese Zuständigkeit besteht. Wenn Sie wissen, wer für die Durchführung von Aktivitäten verantwortlich ist, können Sie nachvollziehen, wer die Aktivität durchführen, das Ergebnis validieren und dem Besitzer der Aktivität Feedback geben wird.

Gewünschtes Ergebnis:

Ihre Organisation definiert klar die Verantwortlichkeiten, um bestimmte Aktivitäten anhand definierter Workloads durchzuführen und auf Ereignisse zu reagieren, die durch die Workloads verursacht werden. Die Organisation dokumentiert die Zuständigkeit für Prozesse und deren Erfüllung und macht diese Informationen auffindbar. Sie überprüfen und aktualisieren die Zuständigkeiten, wenn organisatorische Änderungen stattfinden, und die Teams verfolgen und messen die Leistung der Aktivitäten zur Identifizierung von Fehlern und Ineffizienzen. Sie implementieren Feedback-Mechanismen, um Fehler und Verbesserungen nachzuverfolgen und iterative Verbesserungen zu unterstützen.

Typische Anti-Muster:

- Sie dokumentieren keine Verantwortlichkeiten.
- Fragmentierte Skripte existieren auf isolierten Bedienerarbeitsplätzen. Nur wenige Personen wissen, wie man sie benutzt, oder bezeichnen sie informell als Teamwissen.
- Ein veralteter Prozess muss aktualisiert werden, aber niemand weiß, wer für den Prozess zuständig ist, und der ursprüngliche Autor gehört nicht mehr zur Organisation.
- Prozesse und Skripte sind nicht auffindbar und nicht sofort verfügbar, wenn sie benötigt werden (z. B. als Reaktion auf einen Vorfall).

Vorteile der Nutzung dieser bewährten Methode:

- Sie wissen, wer die verantwortliche Person für die Durchführung einer Aktivität ist, wer benachrichtigt werden muss, wenn eine Aktion erforderlich ist, und wer die Aktion ausführen, das Ergebnis validieren und dem Besitzer der Aktivität Feedback geben wird.

- Prozesse und Verfahren unterstützen Sie bei der Bewältigung Ihrer Workloads.
- Neue Teammitglieder werden schneller handlungsfähig.
- Sie reduzieren die Zeit, die zur Behebung von Vorfällen benötigt wird.
- Verschiedene Teams verwenden dieselben Prozesse und Verfahren, um Aufgaben auf einheitliche Weise auszuführen.
- Teams können ihre Prozesse durch wiederholbare Prozesse skalieren.
- Standardisierte Prozesse und Verfahren tragen dazu bei, die Auswirkungen der Übertragung von Workload-Verantwortlichkeiten zwischen Teams abzumildern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Um mit der Definition von Verantwortlichkeiten zu beginnen, beginnen Sie mit der vorhandenen Dokumentation, wie Zuständigkeitsmatrizen, Prozessen und Verfahren, Rollen und Verantwortlichkeiten sowie Tools und Automatisierung. Überprüfen und besprechen Sie die Verantwortlichkeiten für dokumentierte Prozesse. Ermitteln Sie gemeinsam mit den Teams, ob Abweichungen zwischen den Dokumenten zu Zuständigkeiten und Prozessen vorliegen. Besprechen Sie die angebotenen Dienstleistungen mit internen Kunden dieses Teams, um unterschiedliche Erwartungen zwischen den Teams zu identifizieren.

Analysieren und beheben Sie die Diskrepanzen. Identifizieren Sie Verbesserungsmöglichkeiten und suchen Sie nach häufig nachgefragten, ressourcenintensiven Aktivitäten, bei denen es sich in der Regel um gute Kandidaten für Verbesserungen handelt. Informieren Sie sich über bewährte Methoden, Muster und präskriptive Anleitungen, um Verbesserungen zu vereinfachen und zu standardisieren. Erfassen Sie Verbesserungsmöglichkeiten und verfolgen Sie die Verbesserungen bis zur Fertigstellung.

Mit der Zeit sollten diese Verfahren so weiterentwickelt werden, dass sie als Code ausgeführt werden, sodass weniger menschliche Eingriffe erforderlich sind. Beispielsweise können Verfahren als AWS Lambda-Funktionen, AWS CloudFormation-Vorlagen oder AWS Systems Manager-Automatisierungsdokumente initiiert werden. Stellen Sie sicher, dass diese Verfahren in den entsprechenden Repositorys versionskontrolliert sind und ein geeignetes Ressourcen-Tagging enthalten, sodass die Teams die Eigentümer und die Dokumentation leicht identifizieren können. Dokumentieren Sie die Verantwortung für die Durchführung der Aktivitäten und überwachen Sie dann die Automatisierungen, um sicherzustellen, dass sie erfolgreich initiiert und ausgeführt werden und dass die gewünschten Ergebnisse erzielt werden.

Kundenbeispiel

AnyCompany Retail legt fest, dass das Team oder die Person, die für die Prozesse einer Anwendung oder einer Gruppe von Anwendungen (die gemeinsame architektonische Praktiken und Technologien nutzen) zuständig ist, der Besitzer ist. Zunächst dokumentiert das Unternehmen die Prozesse und Verfahren als schrittweise Anleitungen im Dokumentenmanagementsystem. Es macht die Verfahren mithilfe von Tags auf dem AWS-Konto, das die Anwendung hostet, und anhand bestimmter Gruppen von Ressourcen innerhalb des Kontos auffindbar und verwendet AWS Organizations zur Verwaltung der AWS-Konten. Im Laufe der Zeit konvertiert AnyCompany Retail diese Prozesse in Code und definiert Ressourcen mithilfe von Infrastructure as Code (über Services wie CloudFormation oder AWS Cloud Development Kit (AWS CDK)-Vorlagen). Die Betriebsprozesse werden zu Automatisierungsdokumenten in AWS Systems Manager- oder AWS Lambda-Funktionen, die als geplante Aufgaben als Reaktion auf Ereignisse wie Amazon CloudWatch-Alarme oder Amazon EventBridge-Ereignisse oder durch Anfragen innerhalb einer IT-Servicemanagement-Plattform (ITSM) gestartet werden können. Alle Prozesse sind mit Tags versehen, um die Zuständigkeit zu identifizieren. Teams verwalten die Dokumentation für die Automatisierung und den Prozess auf den Wiki-Seiten, die vom Code-Repository für den Prozess generiert werden.

Implementierungsschritte

1. Dokumentieren Sie die bestehenden Prozesse und Verfahren.
   a. Überprüfen und vergewissern Sie sich, dass sie auf dem neuesten Stand sind.
   b. Stellen Sie sicher, dass jeder Prozess oder jedes Verfahren einen Besitzer hat.
   c. Stellen Sie die Verfahren unter Versionskontrolle.
   d. Wenn möglich, nutzen Sie Prozesse und Verfahren für Workloads und Umgebungen mit gemeinsamen Architekturentwürfen.
2. Richten Sie Mechanismen für Feedback und Verbesserung ein.
   a. Definieren Sie Richtlinien dafür, wie oft Prozesse überprüft werden sollten.
   b. Definieren Sie Prozesse für Prüfende und Genehmigende.
   c. Implementieren Sie Probleme oder eine Ticket-Warteschlange, um Feedback zu geben und zu verfolgen.
   d. Wo immer es möglich ist, sollten Prozesse und Verfahren vorab von einem Gremium zur Genehmigung von Änderungen genehmigt und in eine Risikoklasse eingestuft werden.
3. Machen Sie Prozesse und Verfahren für Benutzer zugänglich und auffindbar, die sie ausführen müssen.

   a. Verwenden Sie Tags, um anzugeben, wo der Prozess und die Verfahren für den Workload aufgerufen werden können.
   b. Verwenden Sie aussagekräftige Fehler- und Ereignismeldungen, um die geeigneten Prozesse oder Verfahren zur Behebung des Problems anzugeben.
   c. Verwenden Sie Wikis oder Dokumentenmanagement, um Prozesse und Verfahren unternehmensweit durchsuchbar zu machen.
4. Automatisieren Sie, wenn es angemessen ist.
   a. Entwickeln Sie Automatisierungen, wenn Services und Technologien eine API bereitstellen.
   b. Stellen Sie sicher, dass die Prozesse gut verstanden werden, und entwickeln Sie Benutzerberichte und Anforderungen, um diese Prozesse zu automatisieren.
   c. Messen Sie die erfolgreiche Nutzung der Prozesse und Verfahren und unterstützen Sie eine iterative Verbesserung anhand der Problemverfolgung.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP01 Ressourcen haben feste Verantwortliche
- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer
- OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten
- OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeiten und Eigentümerschaft sind vorhanden
- OPS11-BP04 Wissensmanagement

Zugehörige Dokumente:

- AWS Whitepaper | Einführung in DevOps in AWS
- AWSWhitepaper | Bewährte Methoden für das Tagging von AWS-Ressourcen
- AWS-Whitepaper | Organisation der AWS-Umgebung mithilfe mehrerer Konten
- Blog zum Thema AWS Cloud-Betrieb und -Migrationen | Entwicklung eines Cloud-Automatisierungsverfahrens für Operational Excellence: Bewährte Methoden von AWS Managed Services

- AWS-Workshop – Tagging
- AWS Service Management Connector

Zugehörige Videos:

- AWS Knowledge Center Live | Tagging von AWS-Ressourcen
- AWS re:Invent 2020 | Automatisierung mit AWS Systems Manager
- AWS re:Inforce 2022 | Automatisierung der Patch-Verwaltung und -Compliance mit AWS (NIS306)
- Unterstützung durch AWS | Vertiefung in AWS Systems Manager

Zugehörige Beispiele:

- Workshop zum Thema AWS Well-Architected Operational Excellence

OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten

Verstehen Sie die die Verantwortlichkeiten Ihrer Rolle und, wie Sie zu Geschäftsergebnissen beitragen, da Ihnen dieses Wissen ermöglicht, Ihre Aufgaben entsprechend zu priorisieren und die Bedeutung Ihrer Rolle nachzuvollziehen. Auf diese Weise können Teammitglieder Anforderungen erkennen und entsprechend reagieren. Wenn die Teammitglieder ihre Rolle kennen, können sie Verantwortung übernehmen, Verbesserungsmöglichkeiten erkennen und verstehen, wie sie Einfluss nehmen oder entsprechende Änderungen vornehmen können.

Gelegentlich kann es vorkommen, dass eine Verantwortlichkeit keinen eindeutigen Besitzer hat. Entwerfen Sie in diesen Situationen einen Mechanismus, um diese Lücke zu schließen. Erstellen Sie einen klar definierten Eskalationspfad zu jemandem, der die Befugnis hat, die Verantwortung zu übertragen, oder entwickeln Sie einen Plan zur Deckung des Bedarfs.

Gewünschtes Ergebnis: Die Teams in Ihrer Organisation haben klar definierte Verantwortlichkeiten, was auch umfasst, wie sie mit Ressourcen, auszuführenden Maßnahmen, Prozessen und Verfahren zusammenhängen. Diese Verantwortlichkeiten entsprechen den Verantwortlichkeiten und Zielen des Teams sowie den Verantwortlichkeiten anderer Teams. Sie dokumentieren die Eskalationswege auf konsistente und nachvollziehbare Weise und nehmen diese Entscheidungen in Dokumentationsartefakte wie Zuständigkeitsmatrizen, Teamdefinitionen oder Wiki-Seiten auf.

Typische Anti-Muster:

- Die Verantwortlichkeiten des Teams sind mehrdeutig oder schlecht definiert.

- Das Team stimmt Rollen nicht mit Verantwortlichkeiten ab.
- Das Team stimmt seine Ziele und Verantwortlichkeiten nicht aufeinander ab, was es schwierig macht, den Erfolg zu messen.
- Die Verantwortlichkeiten der Teammitglieder sind nicht am Team und der gesamten Organisation ausgerichtet.
- Ihr Team hält die Verantwortlichkeiten nicht auf dem neuesten Stand, was dazu führt, dass sie nicht mit den vom Team ausgeführten Aufgaben übereinstimmen.
- Eskalationswege zur Festlegung von Zuständigkeiten sind nicht definiert oder unklar.
- Eskalationspfade haben keinen eindeutigen Besitzer, um eine zeitnahe Reaktion zu gewährleisten.
- Rollen, Zuständigkeiten und Eskalationswege sind nicht auffindbar und bei Bedarf nicht sofort verfügbar (z. B. als Reaktion auf einen Vorfall).

Vorteile der Nutzung dieser bewährten Methode:

- Wenn Sie wissen, wer verantwortlich oder zuständig ist, können Sie sich an das entsprechende Team oder Teammitglied wenden, um eine Anfrage zu stellen oder eine Aufgabe zu übergeben.
- Um das Risiko von Untätigkeit und ungedecktem Bedarf zu verringern, haben Sie eine Person festgelegt, die befugt ist, Verantwortung und Zuständigkeit zu übertragen.
- Wenn Sie den Umfang einer Verantwortlichkeit klar definieren, gewinnen Ihre Teammitglieder an Autonomie und Eigenverantwortung.
- Ihre Verantwortlichkeiten wirken sich auf Ihre Entscheidungen, Ihre Aktionen und die Übergabe von Aktivitäten an die ordnungsgemäßen Besitzer aus.
- Es ist einfach, aufgegebene Verantwortlichkeiten zu identifizieren, da Sie genau wissen, was außerhalb der Verantwortung Ihres Teams liegt, was die Eskalation zur Aufklärung erleichtert.
- Es kommt innerhalb der Teams zu weniger Verwirrung und Spannungen und sie können ihre Workloads und Ressourcen besser verwalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Legen Sie die Rollen und Verantwortlichkeiten von Teammitgliedern fest und vergewissern Sie sich, dass sie die Anforderungen ihrer Rolle kennen. Diese Informationen sollten leicht auffindbar sein, damit Mitglieder Ihrer Organisation herausfinden können, an wen sie sich für bestimmte Anforderungen wenden müssen (an ein Team oder eine Person). In dem Bestreben,

die Chancen der Migration und Modernisierung auf AWS zu nutzen, können sich auch die Rollen und Verantwortlichkeiten ändern. Sorgen Sie dafür, dass sich Ihre Teams und ihre Mitglieder ihrer Verantwortlichkeiten bewusst sind, und schulen Sie sie angemessen, damit sie ihre Aufgaben während dieser Veränderung erfüllen.

Legen Sie fest, an welche Rolle oder welches Team eskaliert werden soll, um die Verantwortlichkeit und Zuständigkeit zu bestimmen. Dieses Team kann mit verschiedenen Stakeholdern zusammenarbeiten, um eine Entscheidung zu treffen. Es sollte jedoch die Verantwortung für die Verwaltung des Entscheidungsprozesses tragen.

Stellen Sie Mitgliedern Ihrer Organisation zugängliche Mechanismen bereit, um Zuständigkeiten und Verantwortlichkeiten zu ermitteln und zuzuordnen. Diese Mechanismen vermitteln ihnen, an wen sie sich bei spezifischen Bedürfnissen wenden können.

Kundenbeispiel

AnyCompany Retail hat kürzlich eine Migration von Workloads von einer On-Premises-Umgebung zu ihrer Landing Zone in AWS mit einem Lift-and-Shift-Ansatz durchgeführt. Das Unternehmen führte eine Betriebsüberprüfung durch, um festzustellen, wie allgemeine betriebliche Aufgaben erfüllt werden, und verifizierte, dass seine bestehende Verantwortungsmatrix die Abläufe in der neuen Umgebung widerspiegelt. Bei der Migration von On-Premise zu AWS reduzierte es die Verantwortlichkeiten der Infrastrukturteams in Bezug auf die Hardware und die physische Infrastruktur. Dieser Schritt eröffnete auch neue Möglichkeiten, das Betriebsmodell für seine Workloads weiterzuentwickeln.

Es identifizierte, adressierte und dokumentierte die meisten Verantwortlichkeiten, definierte aber auch Eskalationswege für alle Verantwortlichkeiten, die übersehen wurden oder die sich im Zuge der Weiterentwicklung der betrieblichen Abläufe möglicherweise ändern müssen. Um neue Möglichkeiten zur Standardisierung und Effizienzsteigerung Ihrer Workloads zu erkunden, bieten Sie Zugriff auf Betriebstools wie AWS Systems Manager und Sicherheitstools wie AWS Security Hub und Amazon GuardDuty. AnyCompany Retail überprüft die Verantwortlichkeiten und die Strategie auf der Grundlage der Verbesserungen, die zuerst angegangen werden sollen. Wenn das Unternehmen neue Arbeitsweisen und Technologiemuster einführt, passt es seine Verantwortungsmatrix entsprechend an.

Implementierungsschritte

1. Beginnen Sie mit der vorhandenen Dokumentation. Zu den typischen Quelldokumenten gehören möglicherweise:

   a. Verantwortungs- oder RACI-Matrizen (Responsible, Accountable, Consulted, Informed)
   b. Teamdefinitionen oder Wiki-Seiten
   c. Servicedefinitionen und Angebote
   d. Rollen- oder Stellenbeschreibungen
2. Überprüfen und besprechen Sie die dokumentierten Verantwortlichkeiten:
   a. Führen Sie Besprechungen mit den Teams durch, um Abweichungen zwischen den dokumentierten Verantwortlichkeiten und den vom Team üblicherweise wahrgenommenen Verantwortlichkeiten zu identifizieren.
   b. Erörtern Sie mögliche Services, die von internen Kunden angeboten werden, um unterschiedliche Erwartungen zwischen den Teams zu identifizieren.
3. Analysieren und beheben Sie die Diskrepanzen.
4. Identifizieren Sie Verbesserungsmöglichkeiten.
   a. Identifizieren Sie häufig nachgefragte, ressourcenintensive Anfragen, bei denen es sich in der Regel um gute Verbesserungsmöglichkeiten handelt.
   b. Informieren Sie sich über bewährte Methoden, Muster und präskriptive Anleitungen und vereinfachen und standardisieren Sie Verbesserungen anhand dieser Anleitungen.
   c. Erfassen Sie Verbesserungsmöglichkeiten und verfolgen Sie sie bis zur Fertigstellung.
5. Wenn ein Team noch nicht die Verantwortung für die Verwaltung und die Verfolgung der Zuweisung von Verantwortlichkeiten trägt, benennen Sie jemanden im Team, der diese Verantwortung übernimmt.
6. Definieren Sie einen Prozess, nach dem Teams eine Klärung der Verantwortlichkeiten anfordern können.
   a. Überprüfen Sie den Prozess und stellen Sie sicher, dass er klar und einfach umzusetzen ist.
   b. Stellen Sie sicher, dass jemand die Verantwortung für die Eskalationen trägt und sie bis zu ihrem Ende verfolgt.
   c. Legen Sie betriebliche Metriken fest, um die Effektivität zu messen.
   d. Schaffen Sie Feedback-Mechanismen, um sicherzustellen, dass Teams Verbesserungsmöglichkeiten hervorheben können.
   e. Implementieren Sie einen Mechanismus für die regelmäßige Überprüfung.
7. Führen Sie Dokumente an einem auffindbaren und zugänglichen Ort.
   a. Wikis oder das Dokumentationsportal sind gängige Optionen.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP06 Bewerten von Kompromissen
- OPS03-BP02 Teammitglieder sind befugt, Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind
- OPS03-BP03 Eskalation wird empfohlen
- OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten
- OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken
- OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen
- OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung

Zugehörige Dokumente:

- AWS Whitepaper – Einführung in DevOps in AWS
- AWS Whitepaper – AWS Cloud Adoption Framework: Die betriebliche Perspektive
- AWS Well-Architected Framework Operational Excellence – Betriebsmodelltopologien auf Workload-Ebene
- AWS Prescriptive Guidance – Entwicklung Ihres Cloud-Betriebsmodells
- AWS Prescriptive Guidance – Erstellung einer RACI- oder RASCI-Matrix für ein Cloud-Betriebsmodell
- Blog zum Thema AWS Cloud-Betrieb und Migration – Unternehmenswert mit Cloud-Plattform-Teams
- Blog zum Thema AWS Cloud-Betrieb und Migration – Warum ein Cloud-Betriebsmodell?
- Blog zum Thema AWS DevOps – So modernisieren sich Organisationen für den Cloud-Betrieb

Zugehörige Videos:

- AWS Summit Online – Cloud-Betriebsmodelle für eine beschleunigte Transformation
- AWS re:Invent 2023 – Zukunftssichere Cloud-Sicherheit: Ein neues Betriebsmodell

OPS02-BP05 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden

Sie können Anfragen an Verantwortliche für Prozesse, Verfahren und Ressourcen stellen. Die Anfragen umfassen Ergänzungen, Änderungen und Ausnahmen. Diese Anfragen durchlaufen einen Änderungsverwaltungsprozess. Treffen Sie fundierte Entscheidungen, um angemessene Anfragen nach einer Bewertung der Vorteile und Risiken zu genehmigen.

Gewünschtes Ergebnis:

- Sie können Anfragen zum Ändern von Prozessen, Verfahren und Ressourcen basierend auf der zugewiesenen Verantwortlichkeit stellen.
- Änderungen werden nach einem sorgfältigen Abwägen der Vorteile und Risiken vorgenommen.

Typische Anti-Muster:

- Sie müssen die Art und Weise der Bereitstellung Ihrer Anwendung aktualisieren, es gibt jedoch keine Möglichkeit, eine Änderung am Bereitstellungsprozess beim Produktionsteam zu beantragen.
- Der Notfallwiederherstellungsplan muss aktualisiert werden, es ist jedoch kein Verantwortlicher kenntlich gemacht, an den Anträge auf Änderungen übermittelt werden können.

Vorteile der Nutzung dieser bewährten Methode:

- Prozesse, Verfahren und Ressourcen können sich weiterentwickeln, wenn sich die Anforderungen ändern.
- Die Verantwortlichen können fundierte Entscheidungen treffen, wann Änderungen vorgenommen werden sollten.
- Änderungen werden nach sorgfältigen Überlegungen vorgenommen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Um diese bewährte Methode zu implementieren, müssen Sie Änderungen an Prozessen, Verfahren und Ressourcen beantragen können. Der Änderungsverwaltungsprozess kann einfach sein. Dokumentieren Sie den Änderungsverwaltungsprozess.

Kundenbeispiel

AnyCompany Retail verwendet für die Angabe, wer für Änderungen an Prozessen, Verfahren und Ressourcen verantwortlich ist, eine Verantwortlichkeitsmatrix (RACI). Es gibt einen dokumentierten Änderungsverwaltungsprozess, der einfach und leicht zu befolgen ist. Mithilfe der RACI-Matrix und des Prozesses können alle Personen Änderungsanträge übermitteln.

Implementierungsschritte

1. Ermitteln Sie die Prozesse, Verfahren und Ressourcen für Ihren Workload sowie die jeweiligen Verantwortlichen. Dokumentieren Sie sie in Ihrem Wissensmanagementsystem.
   a. Wenn Sie OPS02-BP01 Ressourcen haben feste Verantwortliche, OPS02-BP02 Prozesse und Verfahren haben feste Besitzer oder OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind noch nicht implementiert haben, beginnen Sie damit.
2. Arbeiten Sie mit den Stakeholdern in Ihrer Organisation zusammen, um einen Änderungsverwaltungsprozess zu entwickeln. Der Prozess sollte Ergänzungen, Änderungen und Ausnahmen für Ressourcen, Prozesse und Verfahren umfassen.
   a. Sie können AWS Systems Manager Change Manager als Änderungsverwaltungsplattform für Workload-Ressourcen verwenden.
3. Dokumentieren Sie den Änderungsverwaltungsprozess in Ihrem Wissensmanagementsystem.

Aufwand des Implementierungsplans: mittel. Die Entwicklung eines Änderungsverwaltungsprozesses erfordert die Abstimmung mit mehreren Stakeholdern in Ihrer Organisation.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP01 Ressourcen haben feste Verantwortliche – Bevor Sie einen Änderungsverwaltungsprozess entwickeln können, müssen Verantwortliche für die Ressourcen kenntlich gemacht werden.
- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer – Bevor Sie einen Änderungsverwaltungsprozess entwickeln können, müssen Verantwortliche für die Prozesse kenntlich gemacht werden.
- OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind – Bevor Sie einen Änderungsverwaltungsprozess entwickeln können, müssen Verantwortliche für die Verfahren kenntlich gemacht werden.

Zugehörige Dokumente:

- AWS Prescriptive Guidance – Grundlagen-Playbook für umfassende AWS-Migrationen: RACI-Matrizen erstellen
- Whitepaper Change Management in the Cloud (Änderungsmanagement in der Cloud)

Zugehörige Services:

- AWS Systems Manager Change Manager

OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt

Es gibt definierte oder ausgehandelte Vereinbarungen zwischen Teams, in denen die Zusammenarbeit und gegenseitige Unterstützung beschrieben wird (z. B. Reaktionszeiten, Service-Level-Ziele oder Service-Level-Agreements). Die Kanäle für die teamübergreifende Kommunikation werden dokumentiert. Wenn bekannt ist, welche Auswirkungen die Arbeit der Teams auf die Geschäftsergebnisse und die Ergebnisse anderer Teams und Organisationen hat, können die Teams ihre Aufgaben priorisieren und entsprechend handeln.

Wenn Verantwortlichkeit und Eigentümerschaft nicht definiert oder unbekannt sind, besteht das Risiko, dass sowohl die erforderlichen Aktivitäten nicht rechtzeitig ausgeführt als auch redundante und potenziell widersprüchliche Anstrengungen unternommen werden, um diese Anforderungen zu erfüllen.

Gewünschtes Ergebnis:

- Es werden Vereinbarungen zur teamübergreifenden Zusammenarbeit oder Unterstützung getroffen und dokumentiert.
- Teams, die zusammenarbeiten oder sich gegenseitig unterstützen, verfügen über definierte Kommunikationskanäle und Erwartungen in Bezug auf die Reaktion.

Typische Anti-Muster:

- Während der Produktion tritt ein Problem auf und zwei separate Teams beginnen unabhängig voneinander mit der Fehlersuche. Aufgrund der getrennten Bemühungen verlängert sich der Ausfall.
- Das Produktionsteam benötigt Unterstützung vom Entwicklungsteam, es gibt jedoch keine Vereinbarung in Bezug auf die Reaktionszeit. Die Anfrage wird zurückgestellt.

Vorteile der Nutzung dieser bewährten Methode:

- Die Teams wissen, wie sie miteinander interagieren und sich gegenseitig unterstützen können.
- Die Erwartungen in Bezug auf die Reaktionszeit sind bekannt.
- Die Kommunikationskanäle sind klar definiert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Wenn Sie diese bewährte Methode implementieren, bedeutet dies, dass es in Bezug auf die Zusammenarbeit zwischen Teams keine Unklarheiten gibt. Mithilfe von formellen Vereinbarungen wird festgelegt, wie Teams zusammenarbeiten oder sich gegenseitig unterstützen. Die Kanäle für die teamübergreifende Kommunikation werden dokumentiert.

Kundenbeispiel

Das SRE-Team bei AnyCompany Retail hat ein Service-Level-Agreement mit dem Entwicklungsteam abgeschlossen. Wenn das Entwicklungsteam eine Anfrage über das Ticketing-System einreicht, kann es innerhalb von 15 Minuten eine Antwort erwarten. Bei Standortausfällen übernimmt das SRE-Team mit Unterstützung durch das Entwicklungsteam die Leitung der Untersuchung.

Implementierungsschritte

1. Arbeiten Sie zusammen mit den Stakeholdern in Ihrer Organisation und auf Grundlage der Prozesse und Verfahren Vereinbarungen zwischen Teams aus.
   a. Entwickeln Sie für gemeinsame Prozesse oder Verfahren von zwei Teams ein Runbook für die Zusammenarbeit.
   b. Wenn Abhängigkeiten zwischen Teams bestehen, vereinbaren Sie ein SLA für die Reaktionszeit bei Anfragen.
2. Dokumentieren Sie die Verantwortlichkeiten in Ihrem Wissensmanagementsystem.

Aufwand des Implementierungsplans: mittel. Wenn keine Vereinbarungen zwischen Teams vorhanden sind, kann es mühsam sein, eine Vereinbarung mit den Stakeholdern in Ihrer Organisation zu treffen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer – Die Verantwortlichkeit für Prozesse muss kenntlich gemacht werden, bevor Vereinbarungen zwischen Teams getroffen werden.
- OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind – Die Verantwortlichkeit für Betriebsaktivitäten muss kenntlich gemacht werden, bevor Vereinbarungen zwischen Teams getroffen werden.

Zugehörige Dokumente:

- AWS Executive Insights – Mit dem Zwei-Pizza-Team Innovationen vorantreiben
- Einführung in DevOps in AWS – Zwei-Pizza-Teams

OPS 3. Wie unterstützt Ihre Unternehmenskultur Ihre Geschäftsergebnisse?

Stellen Sie Ihren Teammitgliedern Unterstützung bereit, damit sie effektiver handeln und Ihr Geschäftsergebnis unterstützen können.

Bewährte Methoden

- OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten
- OPS03-BP02 Teammitglieder sind befugt, Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind
- OPS03-BP03 Eskalation wird empfohlen
- OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar
- OPS03-BP05 Experimentieren wird empfohlen
- OPS03-BP06 Teammitglieder werden ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern
- OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten

OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten

Auf höchster Ebene fungiert die Geschäftsleitung als Executive Sponsor, um Erwartungen klar festzulegen und die Richtung für die Ergebnisse der Organisation vorzugeben sowie den Erfolg zu bewerten. Der Sponsor befürwortet und fördert die Einführung von bewährten Methoden und die Weiterentwicklung der Organisation.

Gewünschtes Ergebnis: Organisationen, die sich bemühen, ihren Cloud-Betrieb einzuführen, zu transformieren und zu optimieren, legen klare Führungs- und Rechenschaftslinien für die

gewünschten Ergebnisse fest. Die Organisation kennt jede Fähigkeit, die es benötigt, um ein neues Ergebnis zu erzielen, und überträgt den Funktionsteams die Verantwortung für die Entwicklung dieser Fähigkeiten. Die Führung gibt diese Richtung aktiv vor, weist Verantwortung zu, übernimmt Verantwortung und definiert die Arbeit. Dadurch können Mitarbeiter in der gesamten Organisation mobilisieren, sich inspiriert fühlen und aktiv auf die gewünschten Ziele hinarbeiten.

Typische Anti-Muster:

- Workload-Besitzer sind aufgefordert, Workloads zu AWS zu migrieren ohne klare Unterstützung oder einen Plan für den Cloud-Betrieb. Dies führt dazu, dass Teams nicht gezielt zusammenarbeiten, um ihre operativen Fähigkeiten zu verbessern und weiterzuentwickeln. Der Mangel an Betriebsstandards mit bewährten Methoden führt dazu, dass die Teams überfordert sind (z. B. durch Überarbeitung der Mitarbeiter, Bereitschaftsdienste und technische Schulden) und die Innovation ins Stocken gerät.
- Es wurde ein neues organisationsweites Ziel gesetzt, eine neue Technologie einzuführen, ohne die Führung, den Sponsor und die Strategie anzugeben. Die Teams interpretieren Ziele unterschiedlich, was zu Verwirrung darüber führt, worauf sie sich konzentrieren sollten, warum sie wichtig sind und wie Auswirkungen gemessen werden sollen. Folglich verliert die Organisation bei der Einführung der Technologie an Dynamik.

Vorteile der Einführung dieser bewährten Methode: Wenn das Konzept der Führung klar kommuniziert und auch Vision, Richtung und Ziele mitgeteilt werden, wissen die Teammitglieder, was von ihnen erwartet wird. Wenn sich die Führungskräfte aktiv einbringen, beginnen Einzelpersonen und Teams, ihre Bemühungen intensiv in dieselbe Richtung zu lenken, um festgelegte Ziele zu erreichen. Dadurch maximiert die Organisation ihre Erfolgsfähigkeit. Wenn Sie den Erfolg evaluieren, können Sie Barrieren besser identifizieren um anschließend von der Führung gezielt ausgeräumt werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

- In jeder Phase des Wegs in die Cloud (Migration, Einführung oder Optimierung) erfordert der Erfolg eine aktive Beteiligung auf höchster Führungsebene mit einem leitenden Unterstützer. Der leitende Unterstützer richtet die Denkweise, Fähigkeiten und Arbeitsweisen des Teams an der definierten Strategie aus.
  - Das Warum erläutern: Sorgen Sie für Klarheit und erläutern Sie die Gründe für die Vision und Strategie.

- Erwartungen setzen: Definieren und veröffentlichen Sie Ziele für Ihre Organisationen, einschließlich der Art und Weise, wie Fortschritt und Erfolg gemessen werden.
- Zielerreichung verfolgen: Messen Sie regelmäßig die schrittweise Erreichung von Zielen (nicht nur die Erledigung von Aufgaben). Teilen Sie die Ergebnisse mit, damit geeignete Maßnahmen ergriffen werden können, wenn die Ergebnisse gefährdet sind.
- Erforderliche Ressourcen für die Erreichung Ihrer Ziele bereitstellen: Bringen Sie Mitarbeiter und Teams an einen Tisch, um zusammenzuarbeiten und die richtigen Lösungen zu entwickeln, die zu den angestrebten Ergebnissen führen. Dies reduziert oder beseitigt Reibungspunkte in der Organisation.
- Unterstützen Ihrer Teams: Bleiben Sie mit Ihren Teams in Verbindung, um ihre Leistung im Blick zu behalten und potenzielle externe Negativfaktoren zu erkennen. Identifizieren Sie Hindernisse für den Fortschritt Ihrer Teams. Treten Sie für Ihre Teams ein und beseitigen Sie Hindernisse und unnötige Belastungen. Wenn sich äußere Faktoren negativ auf Ihre Teams auswirken, bewerten Sie die Ziele neu und passen Sie sie entsprechend an.
- Fördern der Übernahme bewährter Methoden: Würdigen Sie bewährte Methoden, die messbare Vorteile bieten, und schenken Sie ihren Entwicklern und Anwendern die gebührende Anerkennung. Ermutigen Sie Ihre Teams zur Annahme dieser Methoden, um die Vorteile zu maximieren.
- Weiterentwicklung Ihrer Teams fördern: Schaffen Sie eine Kultur der kontinuierlichen Verbesserung und lernen Sie proaktiv aus erzielten Fortschritten und Misserfolgen. Fördern Sie Wachstum und Entwicklung sowohl im Persönlichen als auch im Betrieblichen. Entwickeln Sie die Vision und Strategie anhand von Daten und Anekdoten weiter.

Kundenbeispiel

AnyCompany Retail befindet sich inmitten einer Geschäftstransformation mit dem Ziel, das Kundenerlebnis schnell neu zu erfinden, die Produktivität zu steigern und das Wachstum durch generative KI zu beschleunigen.

Implementierungsschritte

1. Ernennen Sie einen einzelnen Verantwortlichen und einen leitenden Unterstützer, der die Transformation leitet und vorantreibt.
2. Definieren Sie klare Geschäftsergebnisse für Ihre Transformation, weisen Sie Verantwortlichkeiten zu und fordern Sie Eigenverantwortung ein. Erteilen Sie der leitenden Führungskraft die Befugnis, wichtige Entscheidungen zu leiten und zu treffen.

3. Stellen Sie sicher, dass Ihre Transformationsstrategie sehr klar ist und vom leitenden Sponsor auf allen Ebenen der Organisation umfassend kommuniziert wird.
   a. Legen Sie klar definierte Geschäftsziele für IT- und Cloud-Initiativen fest.
   b. Dokumentieren Sie wichtige Geschäftsmetriken, um die IT- und Cloud-Transformation voranzutreiben.
   c. Kommunizieren Sie die Vision konsequent an alle Teams und Personen, die für Teile der Strategie verantwortlich sind.
4. Entwickeln Sie Matrizen zur Kommunikationsplanung, die vorgeben, welche Botschaft bestimmten Führungskräften, Managern und einzelnen Mitarbeitern übermittelt werden muss. Legen Sie fest, welche Person oder welches Team diese Nachricht übermitteln soll.
   a. Erfüllen Sie Kommunikationspläne konsistent und zuverlässig.
   b. Setzen und steuern Sie Ihre Erwartungen regelmäßig in persönlichen Meetings.
   c. Nehmen Sie Feedback zur Effektivität der Kommunikation an, passen Sie die Kommunikation an und planen Sie entsprechend.
   d. Planen Sie Kommunikationsveranstaltungen, um die Herausforderungen der Teams proaktiv zur Kenntnis zu nehmen, und richten Sie eine konsistente Feedback-Schleife ein, um den Kurs bei Bedarf zu korrigieren.
5. Beschäftigen Sie sich aktiv mit jeder Initiative aus der Führungsperspektive, um sicherzustellen, dass alle betroffenen Teams die Ergebnisse verstehen, für deren Erreichung sie verantwortlich sind.
6. Bei jedem Status-Meeting sollten die leitenden Unterstützer nach Hindernissen Ausschau halten, etablierte Metriken, Anekdoten oder das Feedback der Teams überprüfen und die Fortschritte bei der Erreichung der Ziele messen.

Aufwand des Implementierungsplans: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar
- OP11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung
- OPS11-BP07 Prüfung von Betriebsmetriken

Zugehörige Dokumente:

- Entwirren im Unternehmensknäuel: Abstimmung auf höchster Ebene
- Die lebende Transformation: Veränderungen pragmatisch angehen
- Auf dem Weg zu einem zukunftsfähigen Unternehmen
- 7 Fehler, die Sie bei der Einrichtung eines CCOE vermeiden sollten
- Navigation in der Cloud: Wichtige Key Performance Indicators für den Erfolg

Zugehörige Videos:

- AWS re:Invent 2023: Ein Leitfaden für Führungskräfte zur generativen KI: Die Geschichte kennen, die Zukunft gestalten (SEG204)

Zugehörige Beispiele:

- Prosci: Rolle und Bedeutung des leitenden Unterstützers

OPS03-BP02 Teammitglieder sind befugt, Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind

Eine von der Führung vermittelte Kultur der Eigenverantwortung führt dazu, dass sich die Mitarbeiter bestärkt fühlen, im Namen des gesamten Unternehmens über ihren definierten Rollen- und Verantwortungsbereich hinaus zu handeln. Die Mitarbeiter können handeln, um auftretende Risiken proaktiv zu erkennen und geeignete Maßnahmen ergreifen. Eine solche Kultur ermöglicht es den Mitarbeitern, die Situation zu überblicken und wichtige Entscheidungen zu treffen.

Amazon verwendet beispielsweise die Führungsprinzipien als Leitlinie, um bei den Mitarbeitern erwünschte Verhaltensweisen zu fördern, damit sie verschiedene Situationen bewältigen, Probleme und Konflikte lösen und geeignete Maßnahmen ergreifen können.

Gewünschtes Ergebnis: Die Führung hat eine neue Kultur geprägt, die es Einzelpersonen und Teams ermöglicht, wichtige Entscheidungen zu treffen, auch auf niedrigeren Führungsebenen (sofern diesen Entscheidungen überprüfbare Befugnisse und Sicherheitsmechanismen zugrunde liegen). Misserfolge werden als Lernerfahrung angesehen, und Teams lernen schrittweise, ihre Entscheidungen und Maßnahmen zu optimieren, um in Zukunft ähnliche Situationen zu bewältigen. Wenn die Maßnahmen einer Person zu einer Verbesserung führen, von der andere Teams profitieren können, werden die aus solchen Maßnahmen gewonnenen Erkenntnisse proaktiv geteilt. Die Geschäftsführung misst betriebliche Verbesserungen und bietet dem Einzelnen sowie der Organisation Anreize für die Übernahme solcher Muster.

Typische Anti-Muster:

- In einer Organisation gibt es keine klaren Leitlinien oder Mechanismen dafür, was zu tun ist, wenn ein Risiko erkannt wird. Wenn ein Mitarbeiter beispielsweise einen Phishing-Angriff bemerkt und dies nicht dem Sicherheitsteam meldet, kann dies zur Folge haben, dass ein großer Teil der Organisation auf den Angriff hereinfällt. Dies führt zu einer Datenschutzverletzung.
- Ihre Kunden beschweren sich über die Nichtverfügbarkeit von Services, die hauptsächlich auf fehlgeschlagene Bereitstellungen zurückzuführen ist. Ihr SRE-Team ist für das Bereitstellungstool verantwortlich, und ein automatisiertes Rollback für Bereitstellungen ist Teil der langfristigen Roadmap. Bei einer kürzlichen Anwendungseinführung entwickelte einer der Engineers eine Lösung, um das Rollback seiner Anwendung auf eine frühere Version zu automatisieren. Obwohl die Lösung zum Vorbild für SRE-Teams werden könnte, wird sie von anderen Teams nicht übernommen, da kein Prozess zur Nachverfolgung solcher Verbesserungen vorhanden ist. Die Organisation wird weiterhin durch fehlgeschlagene Bereitstellungen unter Druck gesetzt, die sich auf die Kunden auswirken und die Reputation des Unternehmens gefährden.
- Zur Wahrung der Compliance überwacht Ihr Infosec-Team einen seit langem etablierten Prozess, bei dem gemeinsam genutzte SSH-Schlüssel im Namen der Betreiber, die eine Verbindung zu ihren Amazon EC2-Linux-Instances herstellen, regelmäßig rotieren. Die InfoSec-Teams brauchen mehrere Tage für die Schlüsselrotation. In dieser Zeit können Sie keine Verbindung zu diesen Instances herstellen. Bislang gab es keine Vorschläge, weder seitens von Infosec noch von außerhalb, zur Nutzung anderer Optionen in AWS, um dasselbe Ergebnis zu erzielen.

Vorteile der Etablierung dieser bewährten Methode: Indem Sie die Entscheidungsbefugnisse dezentralisieren und Ihre Teams in die Lage versetzen, wichtige Entscheidungen zu treffen, können Sie Probleme schneller lösen und die Erfolgsquoten erhöhen. Darüber hinaus beginnen die Teams, ein Gefühl der Eigenverantwortung zu entwickeln, und Misserfolge werden als Lernerfahrungen angesehen. Experimentieren wird zu einem Eckpfeiler der Unternehmenskultur. Manager und Bereichsleiter haben nicht das Gefühl, dass sie in allen Aspekten bis ins kleinste Detail gemanagt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

1. Entwickeln Sie eine Kultur, in der damit gerechnet wird, dass Fehler auftreten können.
2. Definieren Sie klare Verantwortlichkeiten und Zuständigkeiten für verschiedene Funktionsbereiche innerhalb der Organisation.

3. Vermitteln Sie Eigenverantwortung und Rechenschaftspflicht, damit alle wissen, wo sie bei dezentralen Entscheidungen Unterstützung erhalten können.
4. Definieren Sie unumkehrbare und leicht revidierbare Entscheidungen, damit die Mitarbeiter wissen, wann sie Beschlüsse an höhere Führungsebenen eskalieren müssen.
5. Schaffen Sie in der Organisation ein Bewusstsein dafür, dass alle Mitarbeiter in der Lage sind, auf verschiedenen Ebenen Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind. Stellen Sie Ihren Teammitgliedern Unterlagen über Governance, Befugnisebenen, Tools sowie Möglichkeiten zur Verfügung, um die erforderlichen Fähigkeiten für eine effektive Reaktion zu üben.
6. Geben Sie Ihren Teammitgliedern die Möglichkeit, die notwendigen Fähigkeiten zu üben, um auf verschiedene Entscheidungen zu reagieren. Sobald die Entscheidungsebenen festgelegt sind, führen Sie GameDays durch, um sicherzustellen, dass alle Mitarbeiter den Prozess verstehen und umsetzen können.
   a. Stellen Sie alternative sichere Umgebungen bereit, in denen Prozesse und Verfahren getestet und eingeübt werden können.
   b. Erkennen Sie an und schaffen Sie ein Bewusstsein dafür, dass Teammitglieder befugt sind, Maßnahmen zu ergreifen, wenn das Ergebnis ein vordefiniertes Risikoniveau aufweist.
   c. Verschaffen Sie den Teammitgliedern die erforderliche Autorität, um Maßnahmen zu ergreifen, indem Sie ihnen Berechtigungen und Zugriff auf ihre Workloads und Komponenten geben.
7. Bieten Sie Teams die Möglichkeit, ihre Erfahrungen (betriebliche Erfolge und Misserfolge) auszutauschen.
8. Ermöglichen Sie Teams, den Status quo in Frage zu stellen, und stellen Sie Mechanismen zur Verfügung, mit denen Verbesserungen sowie deren Auswirkungen auf die Organisation verfolgt und gemessen werden können.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken
- OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeiten und Eigentümerschaft sind vorhanden

Zugehörige Dokumente:

- AWS-Blogbeitrag | Das agile Unternehmen
- AWS-Blogbeitrag | Erfolgsmessung: Ein Paradoxon und ein Plan
- AWS-Blogbeitrag | Loslassen: Autonomie in Teams ermöglichen
- Zentralisieren oder Dezentralisieren?

Zugehörige Videos:

- re:Invent 2023 | Wie Sie Ihre eigene Transformation nicht sabotieren (SEG201)
- re:Invent 2021 | Die Amazon Builders' Library: Operative Exzellenz von Amazon
- Zentralisierung vs. Dezentralisierung

Zugehörige Beispiele:

- Nutzung von Protokollen über Architekturentscheidungen zur Rationalisierung der technischen Entscheidungsfindung für ein Softwareentwicklungsprojekt

OPS03-BP03 Eskalation wird empfohlen

Die Teammitglieder werden von der Führung ermutigt, Probleme und Bedenken an übergeordnete Entscheidungsträger und Stakeholder zu eskalieren, wenn sie der Meinung sind, dass die gewünschten Ergebnisse gefährdet sind und die erwarteten Standards nicht erfüllt werden. Dies ist ein Merkmal der Organisationskultur und wird auf allen Ebenen vorangetrieben. Die Eskalation sollte frühzeitig und lieber zu oft vorgenommen werden, damit Risiken identifiziert und Vorfälle verhindert werden können. Die Führung tadelt Mitarbeiter nicht dafür, wenn sie ein Problem eskalieren.

Gewünschtes Ergebnis: Mitarbeiter in der gesamten Organisation fühlen sich wohl dabei, Probleme an ihre unmittelbaren Vorgesetzten und höhere Führungsebenen zu eskalieren. Die Führung hat bewusst und gezielt die Erwartung aufgestellt, dass sich ihre Teams sicher fühlen sollen, Probleme zu eskalieren. Es wurde ein Mechanismus eingerichtet, um Probleme auf allen Organisationsebenen zu eskalieren. Wenn Mitarbeiter eine Angelegenheit an ihren Vorgesetzten eskalieren, entscheiden sie gemeinsam über das Ausmaß der Auswirkungen und eine mögliche Eskalation des Problems. Eine Eskalation setzt voraus, dass die Mitarbeiter einen empfohlenen Arbeitsplan zur Behebung des Problems beifügen. Wenn die nächsthöhere Führungsebene nicht rechtzeitig Maßnahmen ergreift, sind die Mitarbeiter angehalten, Probleme an die oberste Führungsebene weiterzuleiten, wenn sie der festen Überzeugung sind, dass die Risiken für die Organisation eine Eskalation rechtfertigen.

Typische Anti-Muster:

- Führungskräfte haken während Ihrer Statusbesprechung zum Cloud-Transformationsprogramm nicht ausreichend nach, um herauszufinden, wo Probleme und Hindernisse auftreten. Stattdessen werden nur gute Nachrichten präsentiert. Die CIO hat deutlich gemacht, dass sie nur gute Nachrichten hören möchte, um zu vermeiden, dass der CEO durch angesprochene Herausforderungen den Eindruck gewinnt, das Programm könne scheitern.
- Sie sind als Cloud-Betriebsentwickler tätig und stellen fest, dass das neue Wissensmanagementsystem von den Anwendungsteams kaum verwendet wird. Das Unternehmen investierte ein Jahr und mehrere Millionen Dollar in die Implementierung dieses neuen Wissensmanagementsystems, aber die Mitarbeiter verfassen ihre Runbooks noch immer lokal und teilen sie in einer internen Cloud-Umgebung, was die Suche nach Wissen erschwert, das für unterstützte Workloads relevant ist. Sie versuchen, die Führungskräfte darauf aufmerksam zu machen, da die konsequente Verwendung dieses Systems die betriebliche Effizienz verbessern kann. Als Sie das Problem der Bereichsleiterin vorlegen, die für die Implementierung des Wissensmanagementsystems zuständig ist, werden Sie von ihr kritisiert, weil dadurch die Investition in Frage gestellt wird.
- Das für die Absicherung der Computing-Ressourcen zuständige Infosec-Team hat beschlossen, einen Prozess einzuführen, bei dem die erforderlichen Scans durchgeführt werden müssen, um die vollständige Absicherung der EC2 Instances zu gewährleisten, bevor das Computing-Team die Ressource freigibt. Dies hat zu einer zusätzlichen Verzögerung von einer Woche für die Bereitstellung von Ressourcen und einer Verletzung des SLA geführt. Das Computing-Team hat Angst, dies über die Cloud an den VP zu eskalieren, da der VP für Informationssicherheit dadurch in ein schlechtes Licht gerückt werden könnte.

Vorteile der Nutzung dieser bewährten Methode:

Komplexe oder kritische Probleme werden angegangen, bevor sie sich auf das Geschäft auswirken. Es wird weniger Zeit verschwendet. Risiken werden minimiert. Teams werden bei der Lösung von Problemen proaktiver und ergebnisorientierter.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Die Bereitschaft und Fähigkeit, auf allen Organisationsebenen uneingeschränkt zu eskalieren, ist eine bedeutende Eigenschaft der Organisation und ihrer Kultur, die bewusst weiterentwickelt werden sollte, und zwar durch gezielte Schulungen, Kommunikationen der Führungsebene, Erwartungssetzung und den Einsatz von Mechanismen auf allen Organisationsebenen.

Implementierungsschritte

1. Definieren Sie Richtlinien, Standards und Erwartungen für Ihre Organisation.
   1. Sorgen Sie für eine breite Anwendung und Kenntnis der Richtlinien, Erwartungen und Standards.
2. Ermutigen, schulen und befähigen Sie die Mitarbeiter, damit sie frühzeitig und häufig eskalieren, wenn die Standards nicht eingehalten werden.
3. Bekräftigen Sie in der Organisation, dass die frühe und häufige Eskalation die bewährte Methode ist. Akzeptieren Sie im Unternehmen, dass sich Eskalationen zwar als unbegründet herausstellen können, es sich aber trotzdem insgesamt lohnt, wenn ein echter Vorfall dadurch verhindert wird.
   a. Entwickeln Sie einen Eskalationsmechanismus (wie ein Andron-Cord-System).
   b. Sorgen Sie für dokumentierte Verfahren, die definieren, wann und wie eine Eskalation erfolgen soll.
   c. Definieren Sie die Abfolge der Personen mit zunehmenden Befugnissen, um Maßnahmen zu ergreifen oder zu genehmigen, sowie die Kontaktinformationen der einzelnen Stakeholder.
4. Im Falle einer Eskalation sollte sie so lange fortgesetzt werden, bis das Teammitglied davon überzeugt ist, dass das Risiko durch entsprechende Maßnahmen der Führung gemindert wurde.
   a. Eskalationen sollten Folgendes beinhalten:
      i. Beschreibung der Situation und Art des Risikos
      ii. Kritikalität der Situation
      iii. Wer oder was betroffen ist
      iv. Umfang der Auswirkungen
      v. Dringlichkeit, falls eine Auswirkung eintritt
      vi. Vorgeschlagene Abhilfemaßnahmen und Risikominderungsplan
   b. Schützen Sie Mitarbeiter, die ein Problem eskalieren. Führen Sie eine Richtlinie ein, die Teammitglieder vor Konsequenzen schützt, wenn sie an einen ablehnend eingestellten Entscheidungsträger oder Stakeholder eskalieren. Schaffen Sie Mechanismen, um solche Szenarien zu erkennen, und leiten Sie entsprechende Maßnahmen ein.
5. Fördern Sie eine Kultur der kontinuierlichen Verbesserung durch Feedback-Schleifen in allen Bereichen der Organisation. Feedback-Schleifen fungieren als kleine Eskalationen an die verantwortlichen Personen und identifizieren Verbesserungsmöglichkeiten, auch wenn eine Eskalation nicht erforderlich ist. Eine Kultur der kontinuierlichen Verbesserung zwingt alle dazu, proaktiver zu werden.

6. Die Führung sollte regelmäßig an die Richtlinien, Standards und Mechanismen erinnern sowie an den Wunsch nach offener Eskalation und kontinuierlichen Feedback-Schleifen ohne Vergeltungsmaßnahmen jedweder Art.

Aufwand des Implementierungsplans: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP05 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden

Zugehörige Dokumente:

- Wie wird eine Kultur der kontinuierlichen Verbesserung und des Lernens von Andon und Eskalationssystemen geschaffen?
- Das Andon Cord (IT-Revolution)
- AWS-DevOps-Anleitung | Etablieren Sie klare Eskalationspfade und fördern Sie konstruktive Auseinandersetzungen

Zugehörige Videos:

- Jeff Bezos über Entscheidungsprozesse (und deren Beschleunigung)
- Das Toyota-Produktsystem: Produktionsstopp, ein Knopf und ein elektronisches Andon-Board
- Andon Cord in der LEAN-Fertigung

Zugehörige Beispiele:

- Arbeiten mit Eskalationsplänen im Incident Manager

OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar

Die Führung ist für eine überzeugende und effektive Kommunikation zuständig, insbesondere wenn die Organisation vor der Einführung neuer Strategien, Technologien oder Arbeitsweisen steht. Führungskräfte sollten Erwartungen an alle Mitarbeiter stellen, damit sie auf die Unternehmensziele hinarbeiten können. Entwickeln Sie Kommunikationsmechanismen für die Bildung und

Aufrechterhaltung des geforderten Bewusstseins in Teams, die für die Durchführung von Plänen verantwortlich sind, die von der Führung finanziert und unterstützt werden. Machen Sie sich die organisationsübergreifende Vielfalt zunutze und hören Sie sich verschiedene einzigartige Perspektiven aufmerksam an. Nutzen Sie diese Perspektiven, um Innovation zu fördern, Ihre Annahmen in Frage zu stellen und das Risiko einer Verzerrung durch automatische Bestätigung zu reduzieren. Stärken Sie Inklusion, Vielfalt und Zugehörigkeit innerhalb Ihrer Teams, um nützliche Perspektiven zu gewinnen.

Gewünschtes Ergebnis: Ihre Organisation entwirft Kommunikationsstrategien, um den Auswirkungen von Veränderungen auf das Unternehmen gerecht zu werden. Die Teams werden informiert und motiviert, weiter miteinander statt gegeneinander zu arbeiten. Einzelpersonen kennen die Bedeutung ihrer Rolle, um die angegebenen Ziele zu erreichen. E-Mail ist nur ein passiver Kommunikationsmechanismus und wird als solcher behandelt. Das Management verbringt Zeit mit seinen einzelnen Mitarbeitern, um ihnen ihre Verantwortung, die zu erledigenden Aufgaben und die Bedeutung ihrer Arbeit zur Gesamtmission zu vermitteln. Bei Bedarf binden Führungskräfte ihre Mitarbeiter an kleineren Veranstaltungsorten direkt ein, um Botschaften zu kommunizieren, und sie stellen sicher, dass diese Botschaften effektiv übermittelt werden. Die Organisation erfüllt oder übertrifft die Erwartungen der Führung mithilfe geeigneter Kommunikationsstrategien. Die Führung begrüßt und fördert unterschiedliche Meinungen innerhalb und zwischen Teams.

Typische Anti-Muster:

- Ihre Organisation hat einen Fünf-Jahres-Plan für die Migration aller Workloads in AWS. Der Business Case für die Cloud beinhaltet die Modernisierung von 25 % aller Workloads, um die Vorteile der Serverless-Technologie zu nutzen. Der CIO kommuniziert diese Strategie direkt unterstellten Mitarbeitern und erwartet, dass die Führungskräfte diese Präsentation ohne persönliche Gespräche an Manager, Bereichsleiter und einzelne Mitarbeiter weiterleiten. Der CIO zieht sich zurück und erwartet, dass seine Organisation die neue Strategie umsetzt.
- Die Führung bietet oder nutzt keine Feedback-Mechanismen, und die Erwartungslücke wächst, was dazu führt, dass einzelne Projekte ins Stocken geraten.
- Sie werden gebeten, eine Änderung an Ihren Sicherheitsgruppen vorzunehmen, ohne konkrete Informationen über die Änderung zu erhalten oder darüber, welche Auswirkungen sie auf alle Workloads haben könnte und bis wann sie umzusetzen ist. Der Manager leitet eine E-Mail vom VP von InfoSec weiter und fügt folgende Nachricht hinzu: "Make this happen."
- An Ihrer Migrationsstrategie wurden Änderungen vorgenommen, die die Anzahl der geplanten Modernisierungen von 25 auf 10 % reduzieren. Dies hat nachgelagerte Auswirkungen auf die Betriebsorganisation. Sie wurden nicht über diese strategische Änderung informiert und verfügen

daher nicht über genügend qualifizierte Mitarbeiter, um einen größeren Lift-and-Shift-Aufwand von Workloads in AWS zu bewältigen.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Organisation ist über neue oder geänderte Strategien hinreichend informiert und die Mitarbeiter sind hochmotiviert, um sich gegenseitig dabei zu unterstützen, die von der Führung festgelegten Gesamtziele und Metriken zu erreichen.
- Es gibt Mechanismen und sie werden angewandt, um Teammitglieder rechtzeitig über bekannte Risiken und geplante Ereignisse zu informieren.
- Neue Arbeitsweisen (einschließlich Änderungen bzgl. Belegschaft, Organisation, Prozessen oder Technologien) werden zusammen mit den erforderlichen Fähigkeiten von der Organisation effektiver übernommen. Darüber hinaus erreicht Ihre Organisation schneller Geschäftsvorteile.
- Die Teammitglieder verfügen über die notwendigen Hintergrundinformationen zu den eingehenden Kommunikationen und können ihre Arbeit effektiver erledigen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Zur Implementierung dieser bewährten Methode müssen Sie mit Beteiligten aus der gesamten Organisation zusammenarbeiten, um Kommunikationsstandards zu vereinbaren. Machen Sie diese Standards in der Organisation bekannt. Bei allen wichtigen IT-Umstellungen kann ein etabliertes Planungsteam die Auswirkungen der Änderungen auf seine Mitarbeiter erfolgreicher bewältigen als eine Organisation, die diese Methode nicht anwendet. In größeren Organisationen können Veränderungen schwieriger umzusetzen sein, da es auf eine hohe Zustimmung aller einzelnen Mitarbeiter zu einer neuen Strategie ankommt. In Ermangelung eines solchen Umstellungsplanungsteams trägt die Führung zu 100 % die Verantwortung für eine effektive Kommunikation. Wenn Sie ein Umstellungsplanungsteam einrichten, weisen Sie die Teammitglieder an, mit der gesamten Organisationsführung zusammenzuarbeiten, um eine effektive Kommunikation auf allen Ebenen zu definieren und zu gewährleisten.

Kundenbeispiel

AnyCompany Retail hat sich für den AWS Enterprise Support registriert und ist für seine Cloud-Betriebsabläufe auf andere Drittanbieter angewiesen. Das Unternehmen nutzt Chat und Chatops als zentrales Kommunikationsmedium für seine betrieblichen Aktivitäten. Warnmeldungen und andere

Informationen ergehen über spezifische Kanäle. Wenn eine Maßnahme erforderlich ist, wird das erwartete Ergebnis klar formuliert, und in vielen Fällen gibt es ein Runbook oder Playbook dafür. Das Unternehmen verwendet einen Änderungskalender für die Planung größerer Änderungen an Produktionssystemen.

Implementierungsschritte

1. Richten Sie innerhalb der Organisation ein Kernteam ein, das für die Erstellung und Initiierung von Kommunikationsplänen für Änderungen verantwortlich ist, die auf mehreren Ebenen innerhalb der Organisation stattfinden.
2. Fordern Sie Eigenverantwortlichkeit, um ein hohes Maß an Übersicht zu fördern. Geben Sie den einzelnen Teams die Möglichkeit, unabhängig voneinander Innovationen zu entwickeln, und sorgen Sie für einen ausgewogenen Einsatz einheitlicher Mechanismen, die das richtige Maß an Einsicht und Zielgerichtetheit ermöglichen.
3. Arbeiten Sie mit allen Stakeholdern in Ihrer Organisation zusammen, um Kommunikationsstandards, -methoden und -pläne zu vereinbaren.
4. Stellen Sie sicher, dass das zentrale Kommunikationsteam mit der Organisations- und Programmleitung zusammenarbeitet, um im Namen der Führungskräfte Botschaften an die zuständigen Mitarbeiter zu verfassen.
5. Entwickeln Sie strategische Kommunikationsmechanismen, um Veränderungen mithilfe von Ankündigungen, gemeinsamen Kalendern, Besprechungen mit allen Mitarbeitern und persönlichen oder Einzelgesprächen zu bewältigen, sodass die Teammitglieder die richtigen Erwartungen bezüglich der zu ergreifenden Maßnahmen haben.
6. Geben Sie den erforderlichen Kontext, Details und die nötige Zeit (wenn möglich), um festzustellen, ob Maßnahmen erforderlich sind. Wenn Maßnahmen erforderlich sind, identifizieren Sie die erforderlichen Maßnahmen und deren Auswirkungen.
7. Implementieren Sie Tools, die eine taktische Kommunikation fördern, z. B. interne Chats, E-Mails und Wissensmanagement.
8. Implementieren Sie Mechanismen, um zu messen und zu überprüfen, ob mit allen Kommunikationen die gewünschten Ergebnisse erreicht werden.
9. Richten Sie eine Feedback-Schleife ein, die die Effektivität aller Kommunikationen misst, insbesondere wenn darin der Widerstand gegen Veränderungen in der Organisation thematisiert wird.

10. Ernennen Sie für alle AWS-Konten alternative Ansprechpartner für Abrechnung, Sicherheit und Betrieb. Idealerweise sollte es sich bei diesen Kontakten um E-Mail-Verteilerlisten und nicht um Einzelpersonen handeln.
11. Erstellen Sie einen Kommunikationsplan für die Eskalation und die umgekehrte Eskalation, um mit Ihren internen und externen Teams, einschließlich AWS Support und anderen Drittanbietern, zusammenzuarbeiten.
12. Initiieren Sie Kommunikationsstrategien und setzen Sie sie während der gesamten Laufzeit jedes Transformationsprogramms konsequent um.
13. Priorisieren Sie Maßnahmen, die nach Möglichkeit wiederholbar sind, um sie sicher und in großem Maßstab zu automatisieren.
14. Wenn Kommunikation in Szenarien mit automatisierten Maßnahmen erforderlich ist, sollte die Kommunikation hauptsächlich der Information der Teams oder Audits dienen oder Teil des Änderungsverwaltungsprozesses sein.
15. Analysieren Sie die Kommunikation Ihrer Warnsysteme auf Fehlalarme oder Warnmeldungen, die ständig generiert werden. Entfernen Sie diese Warnmeldungen oder ändern Sie sie so, dass sie nur ausgelöst werden, wenn menschliches Eingreifen erforderlich ist. Stellen Sie ein Runbook oder Playbook bereit, wenn eine Warnmeldung ausgelöst wird.
    a. Mit AWS Systems Manager-Dokumenten können Sie Runbooks oder Playbooks für Warnmeldung erstellen.
16. Es gibt Mechanismen zur Benachrichtigung über Risiken oder geplante Ereignisse auf eine klare und unterstützende Weise mit ausreichend Zeit für geeignete Maßnahmen. Verwenden Sie E-Mail-Listen oder Chat-Kanäle zum Senden von Benachrichtigungen vor geplanten Ereignissen.
    a. Mit AWS Chatbot können Sie innerhalb der Messaging-Plattform Ihrer Organisation Warnmeldungen senden und auf Ereignisse reagieren.
17. Stellen Sie eine zugängliche Informationsquelle bereit, der geplante Ereignisse zu entnehmen sind. Stellen Sie Benachrichtigungen zu geplanten Ereignissen vom gleichen System bereit.
    a. Mit AWS Systems Manager Change Calendar können Sie Änderungszeitfenster für anstehende Änderungen einrichten. Dadurch werden Teammitglieder benachrichtigt, wann Sie in sicherer Weise Änderungen vornehmen können.
18. Überwachen Sie Benachrichtigungen zu Schwachstellen und Patch-Informationen, um bestehende Schwachstellen und potenzielle Risiken im Zusammenhang mit den Komponenten Ihrer Workloads zu verstehen. Stellen Sie Benachrichtigungen für die Teammitglieder bereit, damit sie Maßnahmen ergreifen können.

a. Sie können AWS Security Bulletins abonnieren, um zu Schwachstellen in AWS benachrichtigt zu werden.

19. Berücksichtigen unterschiedlicher Meinungen und Perspektiven: Ermutigen Sie alle anderen, sich zu Wort zu melden. Geben Sie unterrepräsentierten Gruppen die Möglichkeit, sich in die Kommunikation einzubringen. Rotieren Sie die Rollen und Zuständigkeiten in Meetings.

    a. Erweitern von Rollen und Zuständigkeiten: Bieten Sie Teammitgliedern Möglichkeiten, Rollen zu übernehmen, die ihnen fremd sind. Auf diese Weise können sie Erfahrung sammeln und neue Perspektiven durch die Rolle und den resultierenden Austausch mit neuen Teammitgliedern gewinnen, zu denen sie möglicherweise andernfalls keinen Kontakt hätten. Nicht zuletzt können sie die neue Rolle und die Teammitglieder mit ihren Erfahrungen und Perspektiven bereichern. Mit zunehmender Erfahrung werden Sie aufkommende Geschäftsmöglichkeiten oder neue Verbesserungsmöglichkeiten identifizieren. Rotieren Sie allgemeine Aufgaben zwischen den Mitgliedern innerhalb eines Teams, die normalerweise anderen Tätigkeiten nachgehen, damit sie deren Anforderungen und Auswirkungen verstehen.

    b. Bereitstellen einer sicheren und freundlichen Umgebung: Etablieren Sie Richtlinien und Kontrollen zum Schutz der geistigen und physischen Sicherheit der Teammitglieder in Ihrer Organisation. Die Teammitglieder müssen ohne Angst vor Vergeltungsmaßnahmen zusammenarbeiten können. Wenn sich Teammitglieder sicher und willkommen fühlen, ist die Wahrscheinlichkeit höher, dass sie engagiert und produktiv bleiben. Je vielfältiger Ihre Organisation ist, desto besser verstehen Sie die Personen, die Sie unterstützen, einschließlich Ihrer Kunden. Wenn Ihre Teammitglieder zufrieden sind, ihre Meinung sagen können und sich ernst genommen fühlen, steigt die Wahrscheinlichkeit, dass sie wertvolle Erkenntnisse mitteilen (z. B. Marketingmöglichkeiten, erforderliche Maßnahmen zur Barrierefreiheit, unerschlossene Marktsegmente, unbehandelte Risiken in Ihrer Umgebung).

    c. Ermöglichen einer umfassenden Beteiligung von Teammitgliedern: Stellen Sie die Ressourcen bereit, die Ihre Mitarbeiter zur umfassenden Beteiligung an allen arbeitsbezogenen Aktivitäten benötigen. Teammitglieder haben Fähigkeiten entwickelt, mit denen sie ihre täglichen Herausforderungen meistern. Diese einzigartigen Fähigkeiten können für Ihre Organisation von großem Vorteil sein. Wenn Sie die Teammitglieder mit den notwendigen Ressourcen ausstatten, können Sie den Nutzen ihrer Beiträge maximieren.

Ressourcen

Zugehörige bewährte Methoden:

- OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten

- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen

Zugehörige Dokumente:

- AWS-Blogbeitrag | Eigenverantwortung und Befähigung sind der Schlüssel zu leistungsstarken agilen Organisationen
- AWS Executive Insights | Lernen Sie, Innovation statt Komplexität zu skalieren | Eigenverantwortliche Führungskräfte
- AWS-Sicherheitsberichte
- Open CVE
- Support App in Slack zur Verwaltung von Support-Fällen
- Verwaltung von AWS-Ressourcen in Slack-Kanälen mit Amazon Q Developer in chat applications

Zugehörige Beispiele:

- Well-Architected Labs: Inventory and Patch Management (Level 100) (Well-Architected Labs: Bestands- und Patch-Verwaltung (Stufe 100))

Zugehörige Services:

- Amazon Q Developer in chat applications
- AWS Systems Manager Change Calendar
- AWS Systems Manager-Dokumente

OPS03-BP05 Experimentieren wird empfohlen

Experimente können Katalysatoren für die Umsetzung von Ideen in Produkte und Funktionen sein. Sie beschleunigen Lernprozesse und halten Teammitglieder interessiert und engagiert. Team-Mitglieder sollten oft experimentieren, um Innovationen voranzubringen. Selbst nicht erwünschte Ergebnissen bieten den Vorteil, dass man dadurch weiß, wie man nicht vorgehen sollte. Teammitglieder werden nicht für erfolgreiche Experimente mit unerwünschten Ergebnissen bestraft.

Gewünschtes Ergebnis:

- Ihre Organisation ermutigt zum Experimentieren, um Innovationen voranzubringen.

- Experimente werden genutzt, um daraus zu lernen.

Typische Anti-Muster:

- Sie möchten einen A/B-Test durchführen, es gibt jedoch keinen Mechanismus für das Experiment. Sie stellen eine UI-Änderung bereit, ohne diese testen zu können. Dies beeinträchtigt den Kundenkomfort.
- Ihr Unternehmen verfügt nur über eine Staging- und eine Produktionsumgebung. Es gibt keine Sandbox-Umgebung zum Experimentieren mit neuen Funktionen oder Produkten, weshalb Sie in der Produktionsumgebung experimentieren müssen.

Vorteile der Nutzung dieser bewährten Methode:

- Experimente bringen Innovationen voran.
- Mithilfe von Experimenten können Sie schneller auf Feedback reagieren.
- Ihre Organisation entwickelt eine Lernkultur.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Experimente sollten in sicherer Weise durchgeführt werden. Nutzen Sie mehrere Umgebungen für Experimente, ohne dabei Produktionsressourcen in Gefahr zu bringen. Nutzen Sie A/B-Tests und Feature-Flags für Testexperimente. Geben Sie Teammitgliedern die Möglichkeit, Experimente in einer Sandbox-Umgebung durchzuführen.

Kundenbeispiel

AnyCompany Retail ermuntert seine Mitarbeiter zu Experimenten. Teammitglieder können 20 % ihrer wöchentlichen Arbeitszeit für Experimente oder zum Erlernen neuer Technologien nutzen. Es gibt eine Sandbox-Umgebung zum Ausprobieren von Innovationen. Für neue Funktionen werden A/B-Tests verwendet, um sie mit realem Benutzerfeedback zu prüfen.

Implementierungsschritte

1. Arbeiten Sie mit Führungskräften aus dem gesamten Unternehmen zusammen, um Experimente zu unterstützen. Teammitglieder sollten aufgefordert werden, Experimente in sicherer Weise durchzuführen.

2. Stellen Sie Ihren Teammitgliedern eine Umgebung zur Verfügung, in der sie in sicherer Weise experimentieren können. Sie müssen Zugriff auf eine Umgebung haben, die der Produktionsumgebung stark ähnelt.
   a. Sie können ein separates AWS-Konto verwenden, um eine Sandbox-Umgebung für Experimente einzurichten. AWS Control Tower kann zur Bereitstellung solcher Konten verwendet werden.
3. Verwenden Sie Feature-Flags und A/B-Tests, um in sicherer Weise zu experimentieren und Benutzer-Feedback einzuholen.
   a. AWS AppConfig Feature Flags ermöglicht das Erstellen von Feature-Flags.
   b. Amazon CloudWatch Evidently kann für A/B-Tests für eine begrenzte Bereitstellung verwendet werden.
   c. Mit AWS Lambda-Versionen können Sie eine neue Version einer Funktion für Beta-Tests bereitstellen.

Grad des Aufwands für den Implementierungsplan: hoch. Die Bereitstellung einer Umgebung für Teammitglieder, in der sie in sicherer Weise experimentieren können, kann erhebliche Investitionen erfordern. Möglicherweise muss auch der Anwendungscode modifiziert werden, um Feature-Flags verwenden oder A/B-Tests unterstützen zu können.

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP02 Durchführen von Analysen nach Vorfällen – Das Lernen aus Vorfällen ist zusammen mit Experimenten ein wichtiger Faktor für Innovationen.
- OPS11-BP03 Implementieren von Feedbackschleifen – Feedbackschleifen sind ein wichtiger Bestandteil von Experimenten.

Zugehörige Dokumente:

- An Inside Look at the Amazon Culture: Experimentation, Failure, and Customer Obsession (Ein Insiderblick auf die Kultur bei Amazon: Experimente, Fehler und absolute Kundenorientierung)
- Best practices for creating and managing sandbox accounts in AWS(Bewährte Methoden für das Erstellen und Verwalten von Sandbox-Konten in AWS)
- Create a Culture of Experimentation Enabled by the Cloud (Schaffen einer Experimente-Kultur mithilfe der Cloud)

- Enabling experimentation and innovation in the cloud at SulAmérica Seguros (Ermöglichen von Experimenten und Innovationen in der Cloud bei SulAmérica Seguros)
- Experiment More, Fail Less (Mehr Experimente, weniger Fehlschläge)
- Organizing Your AWS Environment Using Multiple Accounts - Sandbox OU (Organisieren der AWS-Umgebung mithilfe mehrerer Konten – Sandbox-OU)
- Using AWS AppConfig Feature Flags (Verwendung von AWS AppConfig-Feature-Flags)

Zugehörige Videos:

- AWS On Air ft. Amazon CloudWatch Evidently | AWS Events
- AWS On Air San Fran Summit 2022 ft. AWS AppConfig Feature Flags integration with Jira (AWS AppConfig-Feature-Flags-Integration mit Jira)
- AWS re:Invent 2022 - A deployment is not a release: Control your launches w/feature flags (BOA305-R) (AWS re:Invent 2022 – Eine Bereitstellung ist keine Freigabe: Produktstarts mit Feature-Flags kontrollieren (BOA305-R))
- Programmatically Create an AWS-Konto with AWS Control Tower(Ein AWS-Konto mit AWS Control Tower programmgesteuert erstellen)
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations(Eine Multi-Konto-Umgebung in AWS einrichten, in der bewährte Methoden für AWS Organizations verwendet werden)

Zugehörige Beispiele:

- AWS Innovation Sandbox
- End-to-end Personalization 101 for E-Commerce (Einführung in die durchgehende Personalisierung für E-Commerce)

Zugehörige Services:

- Amazon CloudWatch Evidently
- AWS AppConfig
- AWS Control Tower

OPS03-BP06 Teammitglieder werden ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern

Teams müssen ihre Fähigkeiten ausbauen, um neue Technologien nutzen und mit veränderten Anforderungen und Aufgaben Ihrer Workloads umgehen zu können. Neue Fähigkeiten im Umgang mit neuen Technologien erhöhen oftmals die Zufriedenheit der Teammitglieder und ermöglichen Innovationen. Unterstützen Sie Ihre Teammitglieder beim Erlangen und Bewahren von Branchenzertifizierungen, mit denen ihre wachsenden Fähigkeiten bestätigt und anerkannt werden. Führen Sie funktionsübergreifende Trainings durch, um den Wissenstransfer zu fördern und das Risiko signifikanter Auswirkungen zu reduzieren, wenn Sie qualifizierte und erfahrene Teammitglieder mit kritischem Wissen verlieren. Schaffen Sie spezielle strukturierte Lernzeiten.

AWS bietet Ressourcen, darunter das AWS Getting Started Resource Center, AWS Blogs, AWS Online Tech Talks, AWS-Veranstaltungen und Webinare sowie die AWS Well-Architected Labs, die Leitfäden, Beispiele und detaillierte Anleitungen zur Schulung Ihrer Teams bieten.

Ressourcen wie Support, (AWS re:Post, Support Center) und AWS-Documentation helfen dabei, technische Hindernisse zu beseitigen und den Betrieb zu verbessern. Bei Fragen können Sie sich über das Support Center an den Support wenden.

AWS stellt in der The Amazon Builders' Library auch bewährte Methoden und Muster vor, die wir durch den Betrieb von AWS gelernt haben, sowie im AWS-Blog und im offiziellen AWS-Podcast eine Vielzahl anderer nützlicher Lernmaterialien.

AWS Training und die Zertifizierung beinhalten kostenlose Schulungen in digitalen Kursen zum Selbststudium sowie rollen- und bereichsspezifische Lernpläne. Sie können sich auch für eine Schulung mit Kursleiter registrieren, um die AWS-Fähigkeiten Ihres Teams auszubauen.

Gewünschtes Ergebnis: Ihre Organisation evaluiert ständig Qualifikationslücken und schließt sie durch strukturierte Budget- und Investitionspläne. Die Teams ermutigen und unterstützen ihre Mitglieder durch Weiterbildungsaktivitäten wie den Erwerb führender Branchenzertifizierungen. Die Teams nutzen spezielle Programme zum Wissensaustausch wie informelle Schulungen, Immersion Days, Hackathons und GameDays. Ihre Organisation hält ihre Wissenssysteme auf dem aktuellen Stand und relevant für die Schulung von Teammitgliedern, einschließlich Schulungen zur Einarbeitung neuer Mitarbeiter.

Typische Anti-Muster:

- Aufgrund eines fehlenden strukturierten Trainingsprogramms und Budgets entstehen in den Teams Unsicherheit und Zweifel, wenn sie versuchen, mit der technologischen Entwicklung Schritt zu halten, was letztlich zu einer erhöhten Personalabwanderung führt.

- Im Rahmen der Migration zu AWS weist Ihre Organisation Qualifikationslücken auf und die Teams verfügen über unterschiedlich starke Cloud-Kompetenzen. Aufgrund fehlender Fortbildungsprogramme sehen sich die Teams mit der veralteten und ineffizienten Verwaltung der Cloud-Umgebung überfordert, was zu einer Mehrbelastung der Mitarbeiter führt. Diese erschwerten Arbeitsbedingungen erhöhen die Unzufriedenheit der Mitarbeiter.

Vorteile der Einführung dieser bewährten Methode: Wenn Ihre Organisation bewusst in die Verbesserung der Fähigkeiten seiner Teams investiert, wird damit auch die Cloud-Einführung und -Optimierung beschleunigt und skaliert. Gezielte Lernprogramme fördern Innovationen und stärken die operativen Fähigkeiten der Teams, um auf Ereignisse vorbereitet zu sein. Teams investieren bewusst in die Implementierung und Weiterentwicklung von bewährten Methoden. Die Arbeitsmoral im Team ist hoch und die Teammitglieder sind stolz auf ihren Beitrag zum Unternehmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Investieren Sie kontinuierlich in die berufliche Weiterentwicklung Ihrer Teams, um neue Technologien einzuführen, Innovationen voranzutreiben und mit den Veränderungen der Anforderungen und Verantwortlichkeiten Schritt zu halten, um Ihre Workloads zu unterstützen.

Implementierungsschritte

1. Verwendung strukturierter Cloud-Unterstützungsprogramme:AWS Skills Guild bietet beratende Schulungen an, um das Vertrauen in Cloud-Fähigkeiten zu stärken und eine Kultur des kontinuierlichen Lernens anzuregen.
2. Bereitstellung von Ressourcen für Weiterbildungen: Richten Sie eine spezielle strukturierte Lernzeit ein und stellen Sie Schulungsmaterialien und Übungsressourcen bereit. Unterstützen Sie die Teilnahme an Konferenzen und bei Branchenverbänden, die Möglichkeiten zum Lernen von Lehrkräften und anderen Fachleuten bieten. Stellen Sie für Ihre Junior-Teammitglieder den Kontakt zu erfahreneren Teammitgliedern als Mentoren her oder ermöglichen Sie Junior-Teammitgliedern, ihnen bei der Arbeit zuzusehen, um sich mit ihren Methoden und Fähigkeiten vertraut zu machen. Ermutigen Sie dazu, auch etwas über Inhalte zu lernen, die nicht direkt mit der Arbeit zusammenhängen, um den Horizont zu erweitern.
3. Ermutigung zur Nutzung technischer Fachressourcen: Nutzen Sie Ressourcen wie AWS Re:Post, um Zugang zu kuratiertem Wissen und einer lebendigen Community zu erhalten.
4. Aufbau und Pflege eines aktuellen Wissensrepositorys: Verwenden Sie Plattformen für den Wissensaustausch wie Wikis und Runbooks. Erstellen Sie mit AWS re:POST Private Ihre

eigene wiederverwendbare Informationsquelle mit Expertenwissen, um die Zusammenarbeit zu optimieren, die Produktivität zu verbessern und die Einarbeitung von Mitarbeitern zu beschleunigen.

5. Teamschulung und teamübergreifende Zusammenarbeit: Planen Sie die kontinuierlichen Weiterbildungsanforderungen Ihrer Teammitglieder mit ein. Schaffen Sie Gelegenheiten für die Teammitglieder, (vorübergehend oder dauerhaft) in anderen Teams zu arbeiten, damit sie untereinander Fähigkeiten und bewährte Methoden austauschen können, wovon letztendlich die gesamte Organisation profitiert.
6. Unterstützung beim Erlangen und Bewahren von Branchenzertifizierungen: Unterstützen Sie Ihre Teammitglieder bei der Erlangung und dem Erhalt von Branchenzertifizierungen, durch die das Gelernte bestätigt wird und die Erfolge anerkannt werden.

Aufwand für den Implementierungsplans: hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten
- OPS11-BP04 Wissensmanagement

Zugehörige Dokumente:

- AWS Whitepaper | Framework zur Cloud-Einführung: Die Perspektive der Mitarbeiter
- Investition in kontinuierliches Lernen für eine erfolgreiche Zukunft Ihrer Organisation
- AWS Skills Guild
- AWS Training und -Zertifizierung
- Support
- AWS re:Post
- AWS-Ressourcencenter für den Einstieg
- AWS-Blogs
- AWS Cloud-Compliance
- AWS-Dokumentation
- Der offizielle AWS-Podcast.

- AWS Online Tech Talks
- AWS-Veranstaltungen und -Webinare
- AWS Well-Architected Labs
- Die Amazon Builders' Library

Zugehörige Videos:

- AWS re:Invent 2023 | Umschulung im Tempo der Cloud: Aus Mitarbeitern werden Unternehmer
- WS re:Invent 2023 | Aufbau einer Kultur der Neugier durch Gamification

OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten

Setzen Sie die richtige Anzahl kompetenter Teammitglieder ein und stellen Sie Tools und Ressourcen zur Verfügung, um Ihre Workload-Anforderungen zu erfüllen. Eine Überlastung der Teammitglieder erhöht das Risiko menschlicher Fehler. Investitionen in Tools und Ressourcen wie Automatisierung können die Effektivität Ihres Teams steigern und es dabei unterstützen, eine größere Anzahl von Workloads zu bewältigen, ohne zusätzliche Kapazitäten zu benötigen.

Gewünschtes Ergebnis:

- Sie haben Ihr Team personell angemessen ausgestattet, um die erforderlichen Fähigkeiten zu erwerben, um Workloads in AWS entsprechend Ihres Migrationsplans zu betreiben. Da sich Ihr Team im Laufe Ihres Migrationsprojekts vergrößert hat, hat es sich mit den AWS-Kerntechnologien vertraut gemacht, die das Unternehmen bei der Migration oder Modernisierung seiner Anwendungen verwenden möchte.
- Sie haben Ihren Personalplan sorgfältig abgestimmt, um Ressourcen mithilfe von Automatisierung und Workflows effizient zu nutzen. Ein kleineres Team kann jetzt im Auftrag der Anwendungsentwicklungsteams mehr Infrastruktur verwalten.
- Angesichts sich ändernder betrieblicher Prioritäten werden Personalengpässe proaktiv erkannt, um den Erfolg von Geschäftsinitiativen zu sichern.
- Betriebsmetriken, die auf operative Schwierigkeiten (wie Ermüdung des Bereitschaftsdienstes oder übermäßiges Telefonieren) hinweisen, werden überprüft, um eine Überforderung der Mitarbeiter zu vermeiden.

Typische Anti-Muster:

- Ihre Mitarbeiter erwerben keine neuen AWS-Fähigkeiten, während Sie Ihren mehrjährigen Cloud-Migrationsplan entwickeln, was die Unterstützung der Workloads riskiert und die Arbeitsmoral der Mitarbeiter herabsetzt.
- Ihre gesamte IT-Organisation stellt sich auf agile Arbeitsweisen um. Das Unternehmen priorisiert das Produktportfolio und legt Metriken dafür fest, welche Features zuerst entwickelt werden müssen. Ihr agiler Prozess erfordert nicht, dass Teams ihren Arbeitsplänen Story Points zuweisen. Daher ist es unmöglich zu wissen, welche Kapazitäten für den nächsten Arbeitsschritt erforderlich sind oder ob Sie über die dafür notwendigen Fähigkeiten verfügen.
- Sie lassen Ihre Workloads von einem AWS Partner migrieren, und Sie haben keinen Supportübergangsplan für Ihre Teams, sobald der Partner das Migrationsprojekt abgeschlossen hat. Ihre Teams haben Schwierigkeiten, die Workloads effizient und effektiv zu unterstützen.

Vorteile der Einführung dieser bewährten Methode:In Ihrer Organisation stehen Ihnen entsprechend qualifizierte Teammitglieder zur Verfügung, um die Workloads zu unterstützen. Die Ressourcenzuweisung kann an sich ändernde Prioritäten angepasst werden, ohne die Leistung zu beeinträchtigen. Somit können die Teams die Workloads effizient unterstützen und gleichzeitig mehr Zeit mit Innovationen für Kunden aufwenden, was wiederum die Mitarbeiterzufriedenheit erhöht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Ressourcenplanung für Ihre Cloud-Migration sollte auf einer Organisationsebene erfolgen, die Ihrem Migrationsplan sowie dem gewünschten Betriebsmodell entspricht, das zur Unterstützung Ihrer neuen Cloud-Umgebung implementiert wird. Dies erfordert nicht zuletzt ein umfassendes Verständnis, welche Cloud-Technologien für die Geschäfts- und Anwendungsentwicklungsteams eingesetzt werden. Die Infrastruktur- und Betriebsleitung sorgt für eine Analyse von Qualifikationslücken, Schulungen und die Rollendefinition für Ingenieure, die die Cloud-Einführung leiten.

Implementierungsschritte

1. Definieren Sie Erfolgskriterien für den Erfolg des Teams anhand relevanter Betriebsmetriken wie der Mitarbeiterproduktivität (z. B. Kosten für die Unterstützung einer Workload oder Arbeitsstunden, die Mitarbeiter bei Vorfällen aufgewendet haben).
2. Definieren Sie Mechanismen zur Planung und Überprüfung der Kapazität von Ressourcen, um sicherzustellen, dass bei Bedarf ausreichend qualifizierte Ressourcen verfügbar sind und deren Zahl im Laufe der Zeit angepasst werden kann.

3. Schaffen Sie Mechanismen (z. B. das Senden einer monatlichen Umfrage an Teams), um arbeitsbezogene Herausforderungen zu verstehen, die sich auf Teams auswirken (z. B. zunehmende Verantwortlichkeiten, technologische Veränderungen, Personalabwanderung oder wachsende Anzahl unterstützter Kunden).
4. Verwenden Sie diese Mechanismen, um mit Teams in Kontakt zu treten und Trends zu erkennen, die zu Problemen bei der Mitarbeiterproduktivität beitragen können. Wenn sich äußere Faktoren negativ auf Ihre Teams auswirken, bewerten Sie die Ziele neu und passen Sie sie entsprechend an. Identifizieren Sie Hindernisse für den Fortschritt Ihrer Teams.
5. Prüfen Sie regelmäßig, ob Ihre derzeit vorhandenen Ressourcen noch ausreichen oder ob zusätzliche Ressourcen benötigt werden, und nehmen Sie entsprechende Anpassungen an den Support-Teams vor.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS03-BP06 Teammitglieder werden ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern
- OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen
- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen
- OPS10-BP07 Automatisieren von Reaktionen auf Ereignisse

Zugehörige Dokumente:

- AWS Cloud Adoption Framework: Die Perspektive der Mitarbeiter
- Auf dem Weg zu einem zukunftsfähigen Unternehmen
- Priorisieren der Fähigkeiten Ihrer Mitarbeiter, um das Geschäftswachstum voranzutreiben
- Leistungsstarke Organisation – das Zwei-Pizza-Team von Amazon
- Das Erfolgsrezept von Cloud-reifen Unternehmen

Vorbereitung

Fragen

- OPS 4. Wie implementieren Sie die Überwachbarkeit in Ihrem Workload?
- OPS 5. Wie können Sie Fehler reduzieren, die Fehlerbehebung erleichtern und den Ablauf bis zur Produktion verbessern?
- OPS 6. Wie können Sie Bereitstellungsrisiken eindämmen?
- OPS 7. Wie bringen Sie in Erfahrung, ob Sie für die Unterstützung eines Workloads bereit sind?

OPS 4. Wie implementieren Sie die Überwachbarkeit in Ihrem Workload?

Implementieren Sie die Überwachbarkeit in Ihrem Workload, damit Sie dessen Zustand verstehen und datengesteuerte Entscheidungen auf der Grundlage von Geschäftsanforderungen treffen können.

Bewährte Methoden

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung
- OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie
- OPS04-BP05 Implementieren der verteilten Nachverfolgung

OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen

Die Implementierung von Beobachtbarkeit in Ihrem Workload beginnt damit, seinen Status zu verstehen und datengestützte Entscheidungen auf der Grundlage der geschäftlichen Anforderungen zu treffen. Eine der wirksamsten Methoden zur Sicherung der Übereinstimmung von Überwachungsaktivitäten mit den Geschäftszielen ist die Definition und Überwachung von Leistungskennzahlen (KPIs).

Gewünschtes Ergebnis: Effiziente Beobachtbarkeitspraktiken, die eng an den Geschäftszielen ausgerichtet sind und sicherstellen, dass die Überwachungsanstrengungen stets greifbaren Geschäftsergebnissen dienen.

Typische Anti-Muster:

- Undefinierte KPIs: Das Arbeiten ohne klare KPIs kann dazu führen, dass zu viel oder zu wenig überwacht wird und wichtige Signale fehlen.

- Statische KPIs: KPIs werden nicht überarbeitet oder verfeinert, wenn sich der Workload oder die Geschäftsziele ändern.
- Fehlausrichtung: Konzentration auf technische Metriken, die nicht direkt mit Geschäftsergebnissen korrelieren oder schwieriger mit realen Problemen zu korrelieren sind.

Vorteile der Nutzung dieser bewährten Methode:

- Einfache Identifizierung von Problemen: Geschäfts-KPIs machen Probleme oft deutlicher sichtbar als technische Metriken. Ein Rückgang eines Geschäfts-KPIs kann ein Problem effektiver lokalisieren, als die Analyse zahlreicher technischer Metriken.
- Geschäftsausrichtung: Es wird sichergestellt, dass die Überwachungsaktivitäten die Geschäftsziele direkt unterstützen.
- Effizienz: Es erfolgt eine Priorisierung der Ressourcen für die Überwachung und die Konzentration auf wichtige Metriken.
- Proaktivität: Probleme werden erkannt und gelöst, bevor sie weitreichende Auswirkungen auf das Geschäft haben.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

So definieren Sie Workload-KPIs effektiv:

1. Beginnen Sie mit den Geschäftsergebnissen: Bevor Sie sich mit Metriken befassen, sollten Sie sich mit den gewünschten Geschäftsergebnissen vertraut machen. Sind es höhere Umsätze, mehr Benutzerinteraktionen oder schnellere Reaktionszeiten?
2. Stimmen Sie technische Metriken auf Geschäftsziele ab: Nicht alle technischen Metriken wirken sich direkt auf die Geschäftsergebnisse aus. Identifizieren Sie diejenigen, die dies tun. Oft ist es jedoch einfacher, ein Problem anhand eines Geschäfts-KPI zu identifizieren.
3. Verwenden Sie Amazon CloudWatch: Nutzen Sie CloudWatch, um Metriken zu definieren und zu überwachen, die Ihre KPIs repräsentieren.
4. Überprüfen und aktualisieren Sie die KPIs regelmäßig: Sorgen Sie dafür, dass Ihre KPIs relevant bleiben, während sich Ihr Workload und Ihr Unternehmen weiterentwickeln.
5. Beziehen Sie Stakeholder ein: Beziehen Sie sowohl IT- als auch Business-Teams in die Definition und Überprüfung von KPIs ein.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- the section called “OPS04-BP02 Implementieren einer Anwendungstelemetrie”
- the section called “OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung”
- the section called “OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie”
- the section called “OPS04-BP05 Implementieren der verteilten Nachverfolgung”

Zugehörige Dokumente:

- AWS Observability Best Practices (Bewährte Methoden zur Beobachtbarkeit für AWS)
- CloudWatch User Guide (CloudWatch-Benutzerhandbuch)
- AWS Observability Skill Builder Course (Skill-Builder-Kurs zur Beobachtbarkeit in AWS)

Zugehörige Videos:

- Developing an observability strategy (Entwicklung einer Beobachtbarkeitsstrategie)

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit

OPS04-BP02 Implementieren einer Anwendungstelemetrie

Anwendungstelemetrie dient als Grundlage für die Beobachtbarkeit Ihres Workloads. Die ausgegebene Telemetrie muss unbedingt umsetzbare Erkenntnisse zum Status Ihrer Anwendung und zum Erreichen sowohl technischer als auch geschäftlicher Ergebnisse liefern. Ob es um Fehlerbehebung, die Messung der Auswirkungen einer neuen Funktion oder die zuverlässige Ausrichtung auf wichtige Leistungsindikatoren (KPIs) geht – Anwendungstelemetrie liefert Informationen darüber, wie Sie Ihren Workload aufbauen, betreiben und weiterentwickeln können.

Metriken, Protokolle und Traces bilden die drei wichtigsten Säulen der Beobachtbarkeit. Sie dienen als Diagnosetools, die den Status Ihrer Anwendung beschreiben. Im Laufe der Zeit helfen sie bei der Erstellung von Baselines und der Identifizierung von Anomalien. Um sicherzustellen, dass

die Überwachungsaktivitäten und die Geschäftsziele aufeinander abgestimmt sind, ist jedoch die Definition und Überwachung von wichtigen Key Performance Indicators (KPIs) entscheidend. Oft ist es leichter, Probleme anhand von Geschäfts-KPIs zu identifizieren als nur anhand von technischen Metriken.

Andere Telemetriearten, wie Real User Monitoring (RUM) und synthetische Transaktionen, ergänzen diese primären Datenquellen. RUM liefert Echtzeit-Erkenntnisse zu Benutzerinteraktionen, während synthetische Transaktionen potenzielles Benutzerverhalten simulieren und so helfen, Engpässe zu erkennen, bevor echte Benutzer darauf stoßen.

Gewünschtes Ergebnis: Sie erzielen umsetzbare Erkenntnisse zur Leistung Ihres Workloads. Diese Erkenntnisse ermöglichen es Ihnen, proaktive Entscheidungen zur Leistungsoptimierung zu treffen, eine höhere Workload-Stabilität zu erreichen, CI/CD-Prozesse zu rationalisieren und Ressourcen effektiv zu nutzen.

Typische Anti-Muster:

- Unvollständige Beobachtbarkeit: Wenn die Beobachtbarkeit nicht auf jeder Ebene des Workloads berücksichtigt wird, führt dies zu blinden Flecken, die wichtige Erkenntnisse über Systemleistung und Verhalten verschleiern können.
- Fragmentierte Datenansicht: Wenn Daten über mehrere Tools und Systeme verteilt sind, wird es schwierig, einen ganzheitlichen Überblick über den Zustand und die Leistung Ihrer Workloads zu behalten.
- Von Benutzern gemeldete Probleme: Ein Zeichen dafür, dass eine proaktive Problemerkennung durch Telemetrie und Überwachung von Geschäfts-KPIs fehlt.

Vorteile der Nutzung dieser bewährten Methode:

- Fundierte Entscheidungen: Mit Erkenntnissen aus Telemetrie und Geschäfts-KPIs können Sie datengestützte Entscheidungen treffen.
- Verbesserte betriebliche Effizienz: Datengesteuerte Ressourcennutzung führt zu Kosteneffektivität.
- Verbesserte Workload-Stabilität: Schnellere Erkennung und Lösung von Problemen führt zu einer verbesserten Verfügbarkeit.
- Optimierte CI/CD-Prozesse: Erkenntnisse aus Telemetriedaten erleichtern die Verfeinerung von Prozessen und sichern die Codebereitstellung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Verwenden Sie AWS-Services wie Amazon CloudWatch und AWS X-Ray, um Anwendungstelemetrie für Ihren Workload zu implementieren. Amazon CloudWatch bietet eine umfassende Suite von Überwachungstools, mit denen Sie Ihre Ressourcen und Anwendungen in AWS und On-Premises beobachten können. Der Service erfasst, verfolgt und analysiert Metriken, konsolidiert und überwacht Protokolldaten und reagiert auf Änderungen in Ihren Ressourcen, wodurch Sie besser verstehen, wie Ihr Workload funktioniert. Gleichzeitig können Sie mit AWS X-Ray Ihre Anwendungen verfolgen, analysieren und debuggen, um ein umfassendes Verständnis des Verhaltens Ihrer Workloads zu entwickeln. Mit Features wie Service-Maps, Latenzverteilungen und Trace-Zeitplänen liefert AWS X-Ray Ihnen Erkenntnisse zur Leistung Ihres Workloads und zu den Schwachstellen, die ihn beeinträchtigen.

Implementierungsschritte

1. Identifizieren Sie, welche Daten erfasst werden sollen: Ermitteln Sie die wichtigsten Metriken, Protokolle und Traces, die aussagekräftige Erkenntnisse zu Zustand, Leistung und Verhalten Ihres Workloads bieten.
2. Bereitstellen des CloudWatch Agents: Der CloudWatch-Agent ist maßgeblich an der Beschaffung von System- und Anwendungsmetriken und Protokollen von Ihrem Workload und der zugrunde liegenden Infrastruktur beteiligt. Der CloudWatch-Agent kann auch verwendet werden, um OpenTelemetry- oder X-Ray-Traces zu erfassen und an X-Ray zu senden.
3. Implementieren Sie eine Anomalieerkennung für Protokolle und Metriken: Verwenden Sie die CloudWatch Logs-Anomalieerkennung und die CloudWatch Erkennung von Metrikanomalien, um ungewöhnliche Aktivitäten im Betrieb Ihrer Anwendung automatisch zu identifizieren. Diese Tools verwenden Machine-Learning-Algorithmen, um Anomalien zu erkennen und sie zu melden. Dadurch werden Ihre Überwachungsfunktionen verbessert und die Reaktionszeit bei potenziellen Störungen oder Sicherheitsbedrohungen verkürzt. Richten Sie diese Features ein, um den Zustand und die Sicherheit von Anwendungen proaktiv zu verwalten.
4. Schützen Sie vertrauliche Protokolldaten: Verwenden Sie den Amazon CloudWatch Logs Datenschutz, um vertrauliche Informationen in Ihren Protokollen zu maskieren. Dieses Feature trägt zur Wahrung von Datenschutz und Compliance bei, indem sensible Daten automatisch erkannt und maskiert werden, bevor auf sie zugegriffen wird. Implementieren Sie Datenmaskierung, um sensible Daten wie persönlich identifizierbare Informationen (PII) sicher zu handhaben und zu schützen.
5. Definieren und überwachen von Geschäfts-KPIs: Richten Sie benutzerdefinierte Metriken ein, die auf Ihre Geschäftsergebnisse abgestimmt sind.

6. Instrumentieren Ihrer Anwendung mit AWS X-Ray: Neben der Bereitstellung des CloudWatch-Agenten ist es wichtig, Ihre Anwendung so zu instrumentieren, dass sie Trace-Daten ausgibt. Dieser Prozess kann weitere Erkenntnisse zum Verhalten und zur Leistung Ihres Workloads liefern.
7. Standardisieren der Datenerfassung in Ihrer gesamten Anwendung: Standardisieren Sie die Datenerfassungspraktiken in Ihrer gesamten Anwendung. Einheitlichkeit hilft bei der Korrelation und Analyse von Daten und liefert einen umfassenden Überblick über das Verhalten Ihrer Anwendung.
8. Implementieren von kontoübergreifender Beobachtbarkeit: Verbessern Sie die Effizienz der Überwachung mehrerer Konten AWS-Konten mit Amazon CloudWatch kontoübergreifender Beobachtbarkeit. Mit diesem Feature können Sie Metriken, Protokolle und Alarme aus verschiedenen Konten in einer einzigen Ansicht konsolidieren, was die Verwaltung vereinfacht und die Reaktionszeiten bei identifizierten Problemen in der gesamten AWS-Umgebung der Organisation verbessert.
9. Analysieren und Nutzen von Daten: Sobald die Datenerfassung und Normalisierung abgeschlossen sind, verwenden Sie sie Amazon CloudWatch für Metriken- und Protokollanalysen sowie AWS X-Ray für Trace-Analysen. Eine solche Analyse kann wichtige Erkenntnisse über den Zustand, die Leistung und das Verhalten Ihrer Workload liefern und so Ihren Entscheidungsprozess beeinflussen.

Aufwand für den Implementierungsplan: hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Definieren von Workload-KPIs
- OPS04-BP03 Implementieren von Telemetrie für Benutzeraktivitäten
- OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie
- OPS04-BP05 Implementieren einer Transaktionsverfolgung

Zugehörige Dokumente:

- Bewährte Methoden zur Beobachtbarkeit für AWS
- CloudWatch-Benutzerhandbuch
- AWS X-Ray-Entwicklerhandbuch

- Instrumentieren verteilter Systeme für Einblicke in die Betriebsabläufe
- Skill-Builder-Kurs zur Beobachtbarkeit in AWS
- Neuerungen bei Amazon CloudWatch
- Neuerungen bei AWS X-Ray

Zugehörige Videos:

- AWS re:Invent 2022 – Bewährte Überwachungsmethoden bei Amazon
- AWS re:Invent 2022 – Entwicklung einer Überwachungsstrategie

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- AWS-Lösungsbibliothek: Anwendungsüberwachung mit Amazon CloudWatch

OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung

Ein entscheidender Erfolgsfaktor besteht darin, tiefe Einblicke in die Erfahrung Ihrer Kunden und deren Interaktionen mit Ihrer Anwendung zu gewinnen. Zwei leistungsstarke Tools, die diesem Zweck dienen, sind Real User Monitoring (RUM, Reale Benutzerüberwachung) und synthetische Transaktionen. RUM liefert Daten zu echten Benutzerinteraktionen, die ein wahrheitsgetreues Bild der Benutzerzufriedenheit vermitteln. Synthetische Transaktionen hingegen simulieren Benutzerinteraktionen und helfen Ihnen dadurch, potenzielle Probleme zu erkennen, noch bevor sie sich auf echte Benutzer auswirken.

Gewünschtes Ergebnis: Eine ganzheitliche Ansicht des Kundenerlebnisses, die proaktive Erkennung von Problemen und die Optimierung der Benutzerinteraktionen, um nahtlos digitale Erfahrungen zu ermöglichen.

Typische Anti-Muster:

- Anwendungen ohne RUM:
  - Verzögerte Problemerkennung: Ohne RUM werden Sie möglicherweise erst dann auf Leistungsengpässe oder -probleme aufmerksam, wenn sich Benutzer beschweren. Dieser reaktive Ansatz kann bei Ihren Kunden zu Unzufriedenheit führen.

  - Fehlende Einblicke in die Benutzererfahrung: Wenn Sie RUM nicht verwenden, lassen Sie wichtige Daten ungenutzt, die zeigen, wie echte Benutzer mit Ihrer Anwendung interagieren, wodurch Ihre Möglichkeiten zur Optimierung der Benutzererfahrung eingeschränkt bleiben.
- Anwendungen ohne synthetische Transaktionen:
  - Fehlende Grenzfälle: Synthetische Transaktionen helfen Ihnen dabei, Pfade und Funktionen zu testen, die von den meisten Benutzern möglicherweise nicht häufig verwendet werden, aber für bestimmte Geschäftsfunktionen von entscheidender Bedeutung sind. Ohne sie könnten mögliche Fehler bei diesen Pfaden und Funktionen unbemerkt bleiben.
  - Ausbleibende Überprüfung auf Probleme bei inaktiver Anwendung: Regelmäßige synthetische Tests können Situationen simulieren, in denen echte Benutzer nicht aktiv mit Ihrer Anwendung interagieren, wodurch sichergestellt wird, dass das System immer korrekt funktioniert.

Vorteile der Nutzung dieser bewährten Methode:

- Proaktive Problemerkennung: Identifizieren und beheben Sie potenzielle Probleme, bevor sie sich auf echte Benutzer auswirken.
- Optimierte Benutzererfahrung: Kontinuierliches Feedback von RUM hilft Ihnen dabei, die allgemeine Benutzererfahrung zu verfeinern und zu verbessern.
- Erkenntnisse zur Geräte- und Browserleistung: Verstehen Sie, wie gut Ihre Anwendung auf verschiedenen Geräten und Browsern funktioniert, um weitere Optimierungen zu ermöglichen.
- Validierte Geschäftsabläufe: Regelmäßige synthetische Transaktionen stellen sicher, dass Kernfunktionen und kritische Pfade stets betriebsbereit und effizient bleiben.
- Verbesserte Anwendungsleistung: Nutzen Sie Erkenntnisse aus echten Benutzerdaten, um die Reaktionsfähigkeit und Zuverlässigkeit Ihrer Anwendungen zu verbessern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um RUM und synthetische Transaktionen für die Telemetrie von Benutzeraktivitäten zu nutzen, bietet AWS Ihnen Services wie Amazon CloudWatch RUM und Amazon CloudWatch Synthetics. In Verbindung mit Daten zur Benutzeraktivität bieten Metriken, Protokolle und Traces einen umfassenden Überblick über den Betriebsstatus der Anwendung und die Benutzererfahrung zugleich.

Implementierungsschritte

1. Amazon CloudWatch RUM bereitstellen: Integrieren Sie Ihre Anwendung in CloudWatch RUM, um echte Benutzerdaten zu erfassen, zu analysieren und zu präsentieren.
   a. Verwenden Sie die CloudWatch RUM-JavaScript-Bibliothek, um RUM in Ihre Anwendung zu integrieren.
   b. Richten Sie Dashboards ein, um echte Benutzerdaten zu visualisieren und zu überwachen.
2. CloudWatch Synthetics konfigurieren: Erstellen Sie Canaries oder skriptbasierte Routinen, die Benutzerinteraktionen mit Ihrer Anwendung simulieren.
   a. Definieren Sie kritische Anwendungsworkflows und -pfade.
   b. Entwerfen Sie Canaries mit CloudWatch Synthetics-Skripten, um Benutzerinteraktionen für diese Pfade zu simulieren.
   c. Planen und überwachen Sie Canaries so, dass sie in bestimmten Intervallen ausgeführt werden, und sorgen Sie so für einheitliche Leistungsprüfungen.
3. Daten analysieren und Erkenntnisse umsetzen: Nutzen Sie Daten aus RUM und synthetischen Transaktionen, um Erkenntnisse zu gewinnen und korrigierende Maßnahmen zu ergreifen, wenn Anomalien festgestellt werden. Verwenden Sie CloudWatch-Dashboards und Alarme, um auf dem Laufenden zu bleiben.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie
- OPS04-BP05 Implementieren der verteilten Nachverfolgung

Zugehörige Dokumente:

- Leitfaden zu Amazon CloudWatch RUM
- Leitfaden zu Amazon CloudWatch Synthetics

Zugehörige Videos:

- Optimize applications through end user insights with Amazon CloudWatch RUM (Optimierung von Anwendungen durch Endbenutzereinblicke mit Amazon CloudWatch RUM)
- AWS on Air ft. Real-User Monitoring for Amazon CloudWatch (AWS on Air mit RUM für Amazon CloudWatch)

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- Git-Repository für den Amazon CloudWatch RUM-Web-Client
- Verwenden von Amazon CloudWatch Synthetics zur Messung der Seitenladezeit

OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie

Die Abhängigkeitstelemetrie ist für die Überwachung des Status und der Leistung der externen Services und Komponenten, auf die Ihr Workload angewiesen ist, unerlässlich. Sie liefert wertvolle Erkenntnisse zu Erreichbarkeit, Timeouts und anderen kritischen Ereignissen im Zusammenhang mit Abhängigkeiten wie DNS, Datenbanken oder APIs von Drittanbietern. Wenn Sie Ihre Anwendung so instrumentieren, dass sie Metriken, Protokolle und Traces zu diesen Abhängigkeiten ausgibt, gewinnen Sie ein besseres Verständnis von potenziellen Engpässen, Leistungsproblemen oder Ausfällen, die sich auf Ihren Workload auswirken könnten.

Gewünschtes Ergebnis: Die Abhängigkeiten, auf die Ihr Workload angewiesen ist, funktionieren erwartungsgemäß, sodass Sie Probleme proaktiv angehen und eine optimale Workload-Leistung gewährleisten können.

Typische Anti-Muster:

- Nichtbeachtung externer Abhängigkeiten: Sich nur auf interne Anwendungsmetriken konzentrieren und dabei Metriken im Zusammenhang mit externen Abhängigkeiten außer Acht lassen.
- Mangelnde proaktive Überwachung: warten, bis Probleme auftreten, statt den Status und die Leistung von Abhängigkeiten kontinuierlich zu überwachen.
- Isolierte Überwachung: Einsatz mehrerer, unterschiedlicher Überwachungstools, was zu fragmentierten und inkonsistenten Ansichten bezüglich des Überwachungsstatus führen kann.

Vorteile der Nutzung dieser bewährten Methode:

- Verbesserte Zuverlässigkeit der Workloads: Indem sichergestellt wird, dass externe Abhängigkeiten kontinuierlich verfügbar sind und optimal funktionieren.
- Schnellere Problemerkennung und -lösung: Proaktives Identifizieren und Beheben von Problemen mit Abhängigkeiten, bevor sie sich auf den Workload auswirken.
- Umfassender Überblick: Erhalt eines ganzheitlichen Überblicks über interne und externe Komponenten, die den Workload-Status beeinflussen.
- Verbesserte Skalierbarkeit der Workloads: Verständnis der Skalierbarkeitsgrenzen und Leistungsmerkmale externer Abhängigkeiten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Implementieren Sie die Abhängigkeitstelemetrie, indem Sie zunächst die Services, Infrastrukturen und Prozesse identifizieren, von denen Ihr Workload abhängt. Quantifizieren Sie, wie gute Bedingungen aussehen, wenn diese Abhängigkeiten wie erwartet funktionieren, und bestimmen Sie dann, welche Daten zum Messen dieser Bedingungen benötigt werden. Mit diesen Informationen können Sie Dashboards und Warnmeldungen erstellen, die Ihren Operations-Teams Erkenntnisse zum Status dieser Abhängigkeiten liefern. Verwenden Sie AWS-Tools, um die Auswirkungen zu ermitteln und zu quantifizieren, wenn Abhängigkeiten nicht die gewünschten Resultate zeigen. Überarbeiten Sie Ihre Strategie kontinuierlich, um Änderungen der Prioritäten, Ziele und gewonnenen Erkenntnisse Rechnung zu tragen.

Implementierungsschritte

So implementieren Sie die Abhängigkeitstelemetrie auf effiziente Weise:

1. Identifizieren von externen Abhängigkeiten: Arbeiten Sie mit Stakeholdern zusammen, um die externen Abhängigkeiten zu ermitteln, von denen Ihr Workload abhängt. Zu externen Abhängigkeiten zählen Services wie externe Datenbanken, APIs von Drittanbietern, Netzwerkverbindungsrouten zu anderen Umgebungen und DNS-Services. Der erste Schritt zu einer effektiven Abhängigkeitstelemetrie besteht darin, auf ganzer Ebene zu verstehen, welche diese Abhängigkeiten sind.
2. Entwicklung einer Überwachungsstrategie: Sobald Sie sich ein klares Bild von Ihren externen Abhängigkeiten verschafft haben, entwerfen Sie eine darauf zugeschnittene Überwachungsstrategie. Dazu müssen Sie die Wichtigkeit jeder Abhängigkeit, ihr erwartetes Verhalten und alle damit verbundenen Service Level Agreements oder -Ziele verstehen. Richten

Sie proaktive Warnmeldungen ein, die Sie über Statusänderungen oder Leistungsabweichungen informieren.

3. Netzwerküberwachung verwenden: Verwenden Sie Internet Monitor und Network Monitor, die umfassende Einblicke in die globalen Internet- und Netzwerkbedingungen bieten. Diese Tools helfen Ihnen dabei, Ausfälle, Unterbrechungen oder Leistungseinbußen, die sich auf Ihre externen Abhängigkeiten auswirken, zu verstehen und darauf zu reagieren.
4. Informiert bleiben mit dem AWS Health Dashboard: Dieses Dashboard stellt Warnmeldungen bereit und empfiehlt Abhilfemaßnahmen, wenn in AWS Ereignisse eintreten, die möglicherweise Ihre Services betreffen.
   a. Überwachen Sie AWS Health-Ereignisse mithilfe von Amazon EventBridge-Regeln oder integrieren Sie sie programmgesteuert in die AWS Health API, um Aktionen zu automatisieren, wenn Sie AWS Health-Ereignisse erhalten. Dies können allgemeine Aktionen sein, z. B. das Senden aller geplanten Lebenszyklus-Ereignisnachrichten an eine Chat-Oberfläche, oder spezifische Aktionen, wie das Initiieren eines Workflows in einem IT-Servicemanagement-Tool.
   b. Wenn Sie AWS Organizations verwenden, aggregieren Sie AWS Health-Ereignisse kontoübergreifend.
5. Instrumentieren Ihrer Anwendung mit AWS X-Ray: AWS X-Ray bietet Einblicke in die Leistung von Anwendungen und ihren zugrunde liegenden Abhängigkeiten. Verfolgen Sie Anfragen von Anfang bis Ende nach, um Engpässe oder Ausfälle bei den externen Services oder Komponenten zu identifizieren, auf die sich Ihre Anwendung stützt.
6. Verwendung von Amazon DevOps Guru: Dieser Machine-Learning-gestützte Service identifiziert operative Probleme, prognostiziert das Auftreten kritischer Probleme und empfiehlt spezifische Maßnahmen. Dadurch ist er von unschätzbarem Wert, wenn es darum geht, Erkenntnisse zu Abhängigkeiten zu gewinnen und festzustellen, dass sie nicht die Ursache von operativen Problemen sind.
7. Regelmäßige Überwachung: Überwachen Sie kontinuierlich alle Metriken und Protokolle, die sich auf externe Abhängigkeiten beziehen. Richten Sie Warnmeldungen ein, die Sie über unerwartetes Verhalten oder Leistungseinbußen informieren.
8. Validierung nach Änderungen: Überprüfen Sie nach jeder Aktualisierung oder Änderung einer externen Abhängigkeit deren Leistung und Ausrichtung auf die Anforderungen Ihrer Anwendung.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Definieren von Workload-KPIs
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS04-BP03 Implementieren von Telemetrie für Benutzeraktivitäten
- OPS04-BP05 Implementieren einer Transaktionsverfolgung
- OP08-BP04 Erstellen umsetzbarer Warnmeldungen

Zugehörige Dokumente:

- Amazon Personalize AWS Health Dashboard – Benutzerhandbuch
- AWS Internet Monitor – Benutzerhandbuch
- AWS X-Ray-Entwicklerhandbuch
- AWS DevOps Guru-Benutzerhandbuch

Zugehörige Videos:

- Wie sich Internetprobleme auf die Leistung von Apps auswirken
- Einführung in Amazon DevOps Guru
- Verwaltung von Ereignissen im Ressourcenlebenszyklus im großen Maßstab mit AWS Health

Zugehörige Beispiele:

- Operative Erkenntnisse gewinnen mit AIOps und Amazon DevOps Guru
- AWS Health Aware
- Verwenden von tagbasierter Filterung zur Verwaltung von AWS Health Überwachung und Warnmeldungen im großen Maßstab

OPS04-BP05 Implementieren der verteilten Nachverfolgung

Die verteilte Nachverfolgung bietet eine Möglichkeit, Anfragen zu überwachen und zu visualisieren, während sie verschiedene Komponenten eines verteilten Systems durchlaufen. Durch die Erfassung von Trace-Daten aus mehreren Quellen und deren Analyse in einer zentralen Ansicht

können Teams besser verstehen, wie Anfragen ablaufen, wo Engpässe bestehen und worauf Optimierungsbemühungen abzielen sollten.

Gewünschtes Ergebnis: Sie verschaffen sich einen ganzheitlichen Überblick über die Anfragen, die durch Ihr verteiltes System fließen, und ermöglichen so präzises Debugging, optimierte Leistung und verbesserte Benutzererfahrungen.

Typische Anti-Muster:

- Inkonsistente Instrumentierung: Nicht alle Services in einem verteilten System sind für die Nachverfolgung instrumentiert.
- Latenz wird ignoriert: Sie konzentrieren sich nur auf Fehler und berücksichtigen nicht die Latenz oder allmähliche Leistungseinbußen.

Vorteile der Nutzung dieser bewährten Methode:

- Umfassender Systemüberblick: Visualisierung des gesamten Anfragenverlaufs, vom Eingang bis zum Ausgang.
- Verbessertes Debugging: Schnelle Identifizierung von Fehlern oder Leistungsproblemen.
- Verbessertes Benutzererlebnis: Überwachung und Optimierung auf der Grundlage von tatsächlichen Benutzerdaten, um sicherzustellen, dass das System den realen Anforderungen entspricht.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Identifizieren Sie zunächst alle Elemente Ihres Workloads, für die eine Instrumentierung erforderlich ist. Sobald alle Komponenten berücksichtigt sind, können Sie Tools wie AWS X-Ray und OpenTelemetry nutzen, um Trace-Daten für die Analyse mit Tools wie X-Ray und Amazon CloudWatch ServiceLens Map zu erfassen. Nehmen Sie regelmäßig an Besprechungen mit Entwicklern teil und ergänzen Sie diese Diskussionen mit Tools wie Amazon DevOps Guru, X-Ray Analytics und X-Ray Insights, um tiefere Erkenntnisse zu gewinnen. Richten Sie Warnmeldungen anhand von Trace-Daten ein, damit Sie benachrichtigt werden, wenn die im Workload-Überwachungsplan definierten Ergebnisse gefährdet sind.

Implementierungsschritte

So implementieren Sie die verteilte Nachverfolgung auf effektive Weise:

1. Nutzen Sie AWS X-Ray: Integrieren Sie X-Ray in Ihre Anwendung, um Erkenntnisse zu ihrem Verhalten zu gewinnen, ihre Leistung zu verstehen und Engpässe zu lokalisieren. Nutzen Sie X-Ray Insights für die automatische Trace-Analyse.
2. Instrumentieren Sie Ihre Services: Stellen Sie sicher, dass jeder Service, jede AWS Lambda-Funktion und jede EC2-Instance,Trace-Daten sendet. Je mehr Services Sie instrumentieren, desto klarer wird die Gesamtansicht.
3. Integrieren Sie CloudWatch Real User Monitoring und synthetische Überwachung: Integrieren Sie Real User Monitoring (RUM) und synthetische Überwachung mit X-Ray. Auf diese Weise können reale Benutzererfahrungen erfasst und Benutzerinteraktionen simuliert werden, um potenzielle Probleme zu identifizieren.
4. Nutzen Sie den CloudWatch Agent: Der Agent kann Traces entweder von X-Ray oder von OpenTelemetry senden, wodurch die Tiefe der gewonnenen Erkenntnisse verbessert wird.
5. Verwenden Sie Amazon DevOps Guru: DevOps Guru verwendet Daten von X-Ray, CloudWatch, AWS Config und AWS CloudTrail, um umsetzbare Empfehlungen zu liefern.
6. Analysieren Sie Traces: Überprüfen Sie die Trace-Daten regelmäßig, um Muster, Anomalien oder Engpässe zu erkennen, die sich auf die Leistung Ihrer Anwendung auswirken könnten.
7. Richten Sie Benachrichtigungen ein: Konfigurieren Sie Alarme in CloudWatch für ungewöhnliche Muster oder längere Latenzen und ermöglichen Sie dadurch eine proaktive Problembehebung.
8. Kontinuierliche Verbesserung: Überarbeiten Sie Ihre Tracing-Strategie, wenn Services hinzugefügt oder geändert werden, um alle relevanten Datenpunkte zu erfassen.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung
- OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie

Zugehörige Dokumente:

- AWS X-Ray-Entwicklerhandbuch

- Amazon CloudWatch-Benutzerhandbuch für Kundendienstmitarbeiter
- Amazon DevOps Guru-Benutzerhandbuch

Zugehörige Videos:

- Use AWS X-Ray Insights (Nutzung von AWS X-Ray-Erkenntnissen)
- AWS on Air ft. Observability: Amazon CloudWatch and AWS X-Ray (AWS on Air mit Beobachtbarkeit: Amazon CloudWatch und AWS X-Ray)

Zugehörige Beispiele:

- Instrumentierung Ihrer Anwendung mit AWS X-Ray

OPS 5. Wie können Sie Fehler reduzieren, die Fehlerbehebung erleichtern und den Ablauf bis zur Produktion verbessern?

Verwenden Sie Ansätze, die den Fluss von Änderungen in die Produktion verbessern, die Refaktorierung ermöglichen, schnelles Feedback zur Qualität geben und Fehler beheben. Dadurch fließen nützliche Änderungen schneller in die Produktion ein und es treten bei der Bereitstellung weniger Probleme auf. Zudem können Probleme, die durch Bereitstellungsaktivitäten verursacht werden, schnell aufgespürt und gelöst werden.

Bewährte Methoden

- OPS05-BP01 Verwendung einer Versionskontrolle
- OPS05-BP02 Testen und Validieren von Änderungen
- OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung
- OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung.
- OPS05-BP05 Durchführen der Patch-Verwaltung
- OPS05-BP06 Gemeinsame Design-Standards
- OPS05-BP07 Implementieren von Verfahren zur Verbesserung der Codequalität
- OPS05-BP08 Verwenden mehrerer Umgebungen
- OPS05-BP09 Häufige, kleine, reversible Änderungen vornehmen
- OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung

OPS05-BP01 Verwendung einer Versionskontrolle

Aktivieren Sie die Verfolgung von Änderungen und Releases mithilfe einer Versionskontrolle.

Viele AWS-Services bieten Versionskontrollfunktionen. Verwenden Sie ein Revisions- oder Quellcodeverwaltungssystem wie AWS CodeCommit, um Code und andere Artefakte zu verwalten, z. B. versionsgesteuerte AWS CloudFormation -Vorlagen Ihrer Infrastruktur.

Gewünschtes Ergebnis: Ihre Teams arbeiten gemeinsam am Code. Bei der Zusammenführung ist der Code einheitlich und es gehen keine Änderungen verloren. Fehler können durch korrekte Versionierung leicht behoben werden.

Typische Anti-Muster:

- Sie haben Ihren Code auf Ihrer Workstation entwickelt und gespeichert. Es ist ein Speicherfehler bei der Workstation aufgetreten, der nicht rückgängig gemacht werden kann, und Sie haben den Code verloren.
- Nachdem Sie den vorhandenen Code mit Ihren Änderungen überschrieben haben, starten Sie Ihre Anwendung neu, doch sie funktioniert nicht mehr. Sie können die Änderung nicht rückgängig machen.
- Sie arbeiten an einer Berichtsdatei, deshalb ist sie für alle anderen schreibgeschützt, doch ein anderer Benutzer möchte sie bearbeiten. Der Benutzer kontaktiert Sie und bittet darum, die Arbeit daran zu beenden, damit er seine Aufgabe erledigen kann.
- Ihr Forschungsteam arbeitet an einer detaillierten Analyse, die Ihre zukünftige Arbeit prägt. Jemand hat versehentlich seine Einkaufsliste über den endgültigen Bericht gespeichert. Sie können die Änderung nicht rückgängig machen und müssen den Bericht neu erstellen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung von Versionskontrollfunktionen können Sie problemlos auf einen bekanntermaßen funktionierenden Status bzw. frühere Versionen zurücksetzen und so das Risiko von verlorenen Assets begrenzen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Bewahren Sie Ressourcen in Repositorys mit Versionskontrolle auf. Dies ermöglicht die Nachvollziehung von Änderungen, die Bereitstellung neuer Versionen, die Erkennung von Änderungen an bestehenden Versionen und die Rückkehr zu vorherigen Versionen (zum Beispiel bei

einem Fehler die Zurücksetzung auf einen bekanntermaßen funktionierenden Zustand). Integrieren Sie die Versionskontrollfunktionen Ihrer Konfigurationsverwaltungssysteme in Ihre Verfahren.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung.

Zugehörige Dokumente:

- Was ist AWS CodeCommit?

Zugehörige Videos:

- Einführung in AWS CodeCommit

OPS05-BP02 Testen und Validieren von Änderungen

Jede eingesetzte Änderung muss getestet werden, um Fehler in der Produktion zu vermeiden. Diese bewährte Methode konzentriert sich auf das Testen von Änderungen von der Versionskontrolle bis zur Erstellung von Artefakten. Neben Änderungen am Anwendungscode sollten die Tests auch die Infrastruktur, die Konfiguration, die Sicherheitskontrollen und die Betriebsverfahren umfassen. Es gibt viele Formen des Testens, von Tests der Einheiten bis hin zur Softwarekomponentenanalyse (SCA). Wenn Tests im Softwareintegrations- und -bereitstellungsprozess weiter nach links verschoben werden, führt dies zu einer höheren Gewissheit der Artefaktqualität.

Ihr Unternehmen muss Teststandards für alle Software-Artefakte entwickeln. Automatisierte Tests verringern den Arbeitsaufwand und vermeiden manuelle Testfehler. In einigen Fällen können aber auch manuelle Tests notwendig sein. Entwickler müssen Zugang zu automatisierten Testergebnissen haben, um Feedback-Schleifen zur Verbesserung der Softwarequalität zu schaffen.

Gewünschtes Ergebnis: Ihre Softwareänderungen werden vor der Bereitstellung getestet. Die Entwickler haben Zugang zu den Testergebnissen und den Validierungen. Ihre Organisation hat einen Teststandard, der für alle Softwareänderungen gilt.

Typische Anti-Muster:

- Sie stellen eine neue Softwareänderung ohne jegliche Tests bereit. Sie wird in der Produktion nicht ausgeführt, was zu einem Ausfall führt.

- Es werden neue Sicherheitsgruppen mit AWS CloudFormation eingesetzt, ohne in einer Vorproduktionsumgebung getestet zu werden. Durch die Sicherheitsgruppen ist Ihre App für Ihre Kunden unerreichbar.
- Eine Methode wurde geändert, aber es gibt keine Tests der Einheiten. Die Software läuft nicht, wenn sie in der Produktion eingesetzt wird.

Vorteile der Nutzung dieser bewährten Methode: Die Fehlerquote von Änderungen bei Softwarebereitstellungen wird reduziert. Die Qualität der Software wird verbessert. Die Entwickler haben ein größeres Bewusstsein für die Lebensfähigkeit ihres Codes. Sicherheitsrichtlinien können zuverlässig eingeführt werden, um die Compliance des Unternehmens zu unterstützen. Infrastrukturänderungen, wie automatische Aktualisierungen der Skalierungsrichtlinien, werden im Voraus getestet, um den Anforderungen des Datenverkehrs gerecht zu werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Alle Änderungen, vom Anwendungscode bis zur Infrastruktur, werden im Rahmen Ihrer kontinuierlichen Integrationspraxis getestet. Die Testergebnisse werden veröffentlicht, damit die Entwickler schnelles Feedback erhalten. Ihre Organisation hat einen Teststandard, den alle Änderungen erfüllen müssen.

Nutzen Sie die Leistungsfähigkeit generativer KI mit Amazon Q Developer, um die Produktivität und Codequalität von Entwicklern zu verbessern. Amazon Q Developer umfasst die Generierung von Codevorschlägen (basierend auf großen Sprachmodellen), die Erstellung von Komponententests (einschließlich Randbedingungen) und Verbesserungen der Codesicherheit durch die Erkennung und Behebung von Sicherheitsschwachstellen.

Kundenbeispiel

Als Teil der kontinuierlichen Integrationspipeline führt AnyCompany Retail verschiedene Arten von Tests für alle Software-Artefakte durch. Sie praktizieren eine testgesteuerte Entwicklung, sodass die gesamte Software über Tests von Einheiten verfügt. Sobald das Artefakt erstellt ist, führen sie End-to-End-Tests durch. Nach Abschluss dieser ersten Testrunde führen sie einen statischen Anwendungssicherheitsscan durch, bei dem nach bekannten Schwachstellen gesucht wird. Die Entwickler erhalten Meldungen, sobald die einzelnen Prüfpunkte durchlaufen wurden. Sobald alle Tests abgeschlossen wurden, wird der Software-Artefakt in einem Artefakt-Repository gespeichert.

Implementierungsschritte

1. Arbeiten Sie mit den Beteiligten in Ihrem Unternehmen zusammen, um einen Teststandard für Software-Artefakte zu entwickeln. Welche Standardtests sollten alle Artefakte bestehen? Gibt es Compliance- oder Governance-Anforderungen, die bei der Testabdeckung berücksichtigt werden müssen? Müssen Sie die Qualität des Codes testen? Wer muss informiert werden, sobald die Tests abgeschlossen sind?
   1. Die AWS Deployment Pipeline Reference Architecture enthält eine maßgebliche Liste von Testtypen, die als Teil einer Integrationspipeline an Software-Artefakten durchgeführt werden können.
2. Instrumentieren Sie Ihre Anwendung mit den erforderlichen Tests auf der Grundlage Ihres Software-Teststandards. Jeder Testreihe sollte in weniger als zehn Minuten abgeschlossen sein. Tests sollten im Rahmen einer Integrationspipeline durchgeführt werden.
   a. Verwenden Sie Amazon Q Developer, ein generatives KI-Tool, mit dem Sie Komponententestfälle (einschließlich Randbedingungen) erstellen, Funktionen mithilfe von Code und Kommentaren generieren und bekannte Algorithmen implementieren können.
   b. Verwenden Sie Amazon CodeGuru Reviewer, Ihren Anwendungscode auf Fehler zu prüfen.
   c. Mithilfe von AWS CodeBuild können Sie Tests auf Software-Artefakten durchführen.
   d. AWS CodePipeline kann Ihre Softwaretest in eine Pipeline orchestrieren.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP01 Verwendung einer Versionskontrolle
- OPS05-BP06 Gemeinsame Design-Standards
- OPS05-BP07 Implementieren von Verfahren zur Verbesserung der Codequalität
- OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung

Zugehörige Dokumente:

- Einführung eines testgesteuerten Entwicklungsansatzes
- Beschleunigen Ihres Softwareentwicklungszyklus mit Amazon Q
- Amazon Q Developer, jetzt allgemein verfügbar, enthält eine Vorschau auf neue Funktionen, mit denen das Entwicklererlebnis neu gestaltet werden kann

- Der ultimative Spickzettel für die Verwendung von Amazon Q Developer in Ihrer IDE
- Shift-Left-Workload, Nutzung von KI für die Testerstellung
- Amazon Q Developer Center
- 10 Methoden für eine schnellere Entwicklung von Anwendungen mit Amazon CodeWhisperer
- Ein Blick über die Codeabdeckung hinaus mit Amazon CodeWhisperer
- Bewährte Methoden für Prompt-Engineering mit Amazon CodeWhisperer
- Automatisierte AWS CloudFormation-Testpipeline mit TaskCat und CodePipeline
- Erstellen einer End-to-End-AWS-DevSecOps-CI/CD-Pipeline mit Open-Source-SCA-, -SAST- und -DAST-Tools
- Erste Schritte beim Testen von Serverless-Anwendungen
- Meine CI/CD-Pipeline ist mein Release Captain
- Durchführung von Continuous Integration und Continuous Delivery in AWS – Whitepaper

Zugehörige Videos:

- Implementieren einer API mit dem Amazon Q Developer-Agenten für Softwareentwicklung
- Installation, Konfiguration und Verwendung von Amazon Q Developer mit JetBrains-IDEs (Anleitung)
- Beherrschung der Kunst von Amazon CodeWhisperer – YouTube-Playlist
- AWS re:Invent 2020 – Testbare Infrastruktur: Integrationstests auf AWS
- AWS Summit ANZ 2021 – Vorantreiben einer „Test-First“-Strategie mit CDK und testgesteuerter Entwicklung
- Testen Ihrer Infrastruktur as Code mit AWS CDK

Zugehörige Ressourcen:

- Erstellen von Anwendungen mit generativer KI mit Amazon CodeWhisperer
- Amazon CodeWhisperer-Workshop
- Referenzarchitektur für AWS-Bereitstellungs-Pipelines – Anwendung
- AWS Kubernetes DevSecOps Pipeline
- Richtlinie als Code – Workshop – Testgesteuerte Entwicklung
- Tests von Einheiten für eine Node.js-Anwendung aus GitHub mithilfe von AWS CodeBuild

- Serverspec für die testgesteuerte Entwicklung von Infrastrukturcode verwenden

Zugehörige Services:

- Amazon Q Developer
- Amazon CodeGuru Reviewer
- AWS CodeBuild
- AWS CodePipeline

OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung

Verwenden Sie Systeme zur Konfigurationsverwaltung, um Änderungen vorzunehmen und zu verfolgen. Diese Systeme reduzieren Fehler aufgrund von manuellen Prozessen und verringern den Testaufwand.

Bei der statischen Konfigurationsverwaltung werden Werte festgelegt, wenn eine Ressource initialisiert wird, die erwartungsgemäß während der Lebensdauer der Ressource konsistent bleibt. Einige Beispiele sind die Konfiguration eines Web- oder Anwendungsservers auf einer Instance oder die Definition der Konfiguration eines AWS-Service innerhalb der AWS Management Console oder durch die AWS CLI.

Bei der dynamischen Konfigurationsverwaltung werden bei der Initialisierung Werte festgelegt, die sich während der Lebensdauer einer Ressource ändern können oder voraussichtlich ändern werden. So können Sie zum Beispiel durch eine Konfigurationsänderung eine Funktion in Ihrem Code aktivieren oder während eines Vorfalls den Detaillierungsgrad des Protokolls ändern, um mehr Daten zu erfassen, und dann nach dem Vorfall wieder zum Ursprungswert zurückkehren, um unnötige Protokolle und damit verbundene Kosten zu vermeiden.

In AWS können Sie AWS Config zur kontinuierlichen Überwachung Ihrer AWS-Ressourcenkonfigurationen über Konten und Regionen hinweg verwenden. So können Sie den Konfigurationsverlauf besser verfolgen, nachvollziehen, wie sich eine Konfigurationsänderung auf andere Ressourcen auswirkt, und sie im Hinblick auf die erwarteten oder gewünschten Konfigurationen mithilfe von AWS-Config-Regeln und AWS Config Conformance Packs prüfen.

Wenn Sie dynamische Konfigurationen in Ihren Anwendungen haben, die auf Amazon EC2-Instances, AWS Lambda, Containern, Mobilfunkanwendungen oder IoT-Geräten ausgeführt werden, können Sie AWS AppConfig nutzen, um sie in Ihren Umgebungen zu konfigurieren, zu validieren, bereitzustellen und zu überwachen.

In AWS können Sie CI/CD-Pipelines (Continuous Integration/Continuous Deployment) unter Verwendung von Services wie den AWS Developer Tools erstellen (Beispiel: AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeDeployund AWS CodeStar).

Gewünschtes Ergebnis: Sie konfigurieren, validieren und implementieren als Teil Ihrer CI/CD-Pipeline (Continuous Integration, Continuous Delivery). Sie überwachen, um zu überprüfen, ob die Konfigurationen korrekt sind. Dadurch werden die Auswirkungen auf Endbenutzer und Kunden minimiert.

Typische Anti-Muster:

- Sie aktualisieren die Konfigurationen aller Webserver manuell und eine Reihe von Servern reagiert aufgrund von Updatefehlern nicht mehr.
- Sie aktualisieren Ihre Anwendungsserver mehrere Stunden lang auf manuelle Weise. Die Inkonsistenz der Konfiguration während der Änderung führt zu unerwarteten Verhaltensweisen.
- Jemand hat Ihre Sicherheitsgruppen aktualisiert und auf Ihre Webserver kann nicht mehr zugegriffen werden. Sie wissen nicht, was geändert wurde, und verbringen viel Zeit mit der Suche nach dem Problem – die Zeit bis zur Wiederherstellung nimmt zu.
- Sie übertragen eine Vorproduktionskonfiguration ohne Validierung über CI/CD in die Produktion. Sie setzen Benutzer und Kunden falschen Daten und Services aus.

Vorteile der Nutzung dieser bewährten Methode: Die Einführung von Konfigurationsverwaltungssystemen reduziert den Aufwand für die Durchführung und Nachverfolgung von Änderungen sowie die Häufigkeit der durch manuelle Verfahren verursachten Fehler. Konfigurationsverwaltungssysteme liefern Garantien in Bezug auf Governance, Compliance und regulatorische Anforderungen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Konfigurationsverwaltungssysteme werden verwendet, um Änderungen an Anwendungs- und Umgebungskonfigurationen zu verfolgen und zu implementieren. Konfigurationsmanagementsysteme werden auch eingesetzt, um Fehler zu reduzieren, die durch manuelle Prozesse verursacht werden, Konfigurationsänderungen wiederholbar und überprüfbar zu machen und den Aufwand zu reduzieren.

Implementierungsschritte

1. Identifizieren Sie die Verantwortlichen der Konfiguration.

   a. Informieren Sie die Verantwortlichen der Konfigurationen über alle Compliance-, Governance- oder regulatorischen Anforderungen.
2. Identifizieren Sie Konfigurationselemente und Leistungen.
   a. Konfigurationselemente sind alle Anwendungs- und Umgebungskonfigurationen, die von einer Bereitstellung innerhalb Ihrer CI/CD-Pipeline betroffen sind.
   b. Zu den Leistungen gehören Erfolgskriterien, Validierung und was überwacht werden muss.
3. Wählen Sie Tools für die Konfigurationsverwaltung basierend auf Ihren Geschäftsanforderungen und Ihrer Bereitstellungspipeline aus.
4. Ziehen Sie für signifikante Konfigurationsänderungen gewichtete Bereitstellungen wie Canary-Bereitstellungen in Betracht, um die Auswirkungen falscher Konfigurationen zu minimieren.
5. Integrieren Sie Ihre Konfigurationsverwaltung in Ihre CI/CD-Pipeline.
6. Bestätigen Sie alle übermittelten Änderungen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS06-BP01 Einkalkulieren nicht erfolgreicher Änderungen
- OPS06-BP02 Testbereitstellungen
- OPS06-BP03 Einsetzen sicherer Bereitstellungsstrategien
- OPS06-BP04 Automatisieren von Tests und Rollback

Zugehörige Dokumente:

- AWS Control Tower
- Landing Zone Accelerator in AWS
- AWS Config
- Was ist AWS Config?
- AWS AppConfig
- Was ist AWS CloudFormation?
- AWS Developer Tools

Zugehörige Videos:

- AWS re:Invent 2022 - Proactive governance and compliance for AWS workloads (AWS re:Invent 2022 – Proaktive Governance und Compliance für AWS-Workloads)
- AWS re:Invent 2020: Achieve compliance as code using AWS Config (AWS re:Invent 2020: Mit AWS Config Compliance als Code erzielen)
- Manage and Deploy Application Configurations with AWS AppConfig (Verwaltung und Bereitstellung von Anwendungskonfigurationen mit AWS AppConfig)

OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung.

Verwenden Sie Systeme zur Build- und Bereitstellungsverwaltung. Diese Systeme reduzieren Fehler aufgrund von manuellen Prozessen und verringern den Testaufwand.

In AWS können Sie CI/CD-Pipelines (Continuous Integration/Continuous Deployment) unter Verwendung von Services wie den AWS Developer Tools nutzen (z. B. AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeDeployund AWS CodeStar).

Gewünschtes Ergebnis: Ihre Systeme zur Build- und Bereitstellungsverwaltung unterstützen das Continuous Integration Continuous Delivery (CI/CD)-System Ihrer Organisation, das Funktionen zur Automatisierung sicherer Rollouts mit den richtigen Konfigurationen bietet.

Typische Anti-Muster:

- Nachdem Sie Ihren Code auf Ihrem Entwicklungssystem kompiliert haben, kopieren Sie die ausführbare Datei auf Ihre Produktionssysteme und sie kann nicht gestartet werden. Die lokalen Protokolldateien zeigen an, dass die Ausführung aufgrund fehlender Abhängigkeiten fehlgeschlagen ist.
- Sie erstellen Ihre Anwendung erfolgreich mit neuen Funktionen in Ihrer Entwicklungsumgebung und stellen den Code der Quality Assurance (QA, Qualitätsprüfung) zur Verfügung. Die QA-Prüfung schlägt fehl, da statische Komponenten fehlen.
- Am Freitag haben Sie Ihre Anwendung nach großem Aufwand manuell in Ihrer Entwicklungsumgebung erstellt, einschließlich der neu geschriebenen Funktionen. Am Montag können Sie die Schritte, mit denen Sie Ihre Anwendung erfolgreich erstellen konnten, nicht wiederholen.
- Sie führen die Tests durch, die Sie für den neuen Release erstellt haben. Sie verbringen die nächste Woche damit, eine Testumgebung einzurichten und alle vorhandenen Integrationstests durchzuführen, gefolgt von den Leistungstests. Der neue Code bewirkt eine inakzeptable Leistungsbeeinträchtigung und muss neu entwickelt und dann erneut getestet werden.

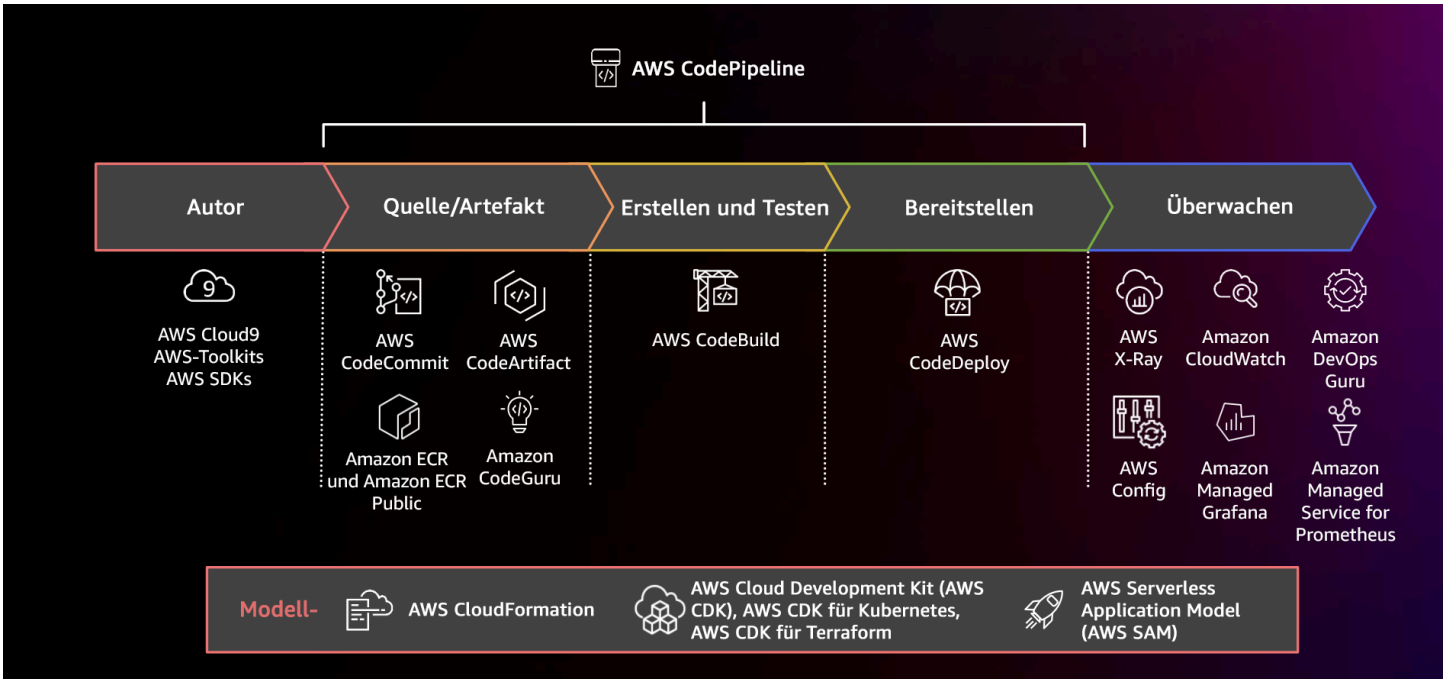
Vorteile der Nutzung dieser bewährten Methode: Mithilfe von Mechanismen zur Verwaltung von Erstellungs- und Bereitstellungsaktivitäten reduzieren Sie den Aufwand für wiederholte Aufgaben, verschaffen Ihren Teammitgliedern die Zeit, sich auf ihre wichtigen Aufgaben zu konzentrieren, und begrenzen die Entstehung von Fehlern durch manuelle Verfahren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Systeme zur Build- und Bereitstellungsverwaltung werden verwendet, um Änderungen nachzuverfolgen und zu implementieren, Fehler zu reduzieren, die durch manuelle Prozesse verursacht werden, und den Aufwand für sichere Implementierungen zu minimieren. Nutzen Sie eine vollständig automatisierte Integrations- und Bereitstellungs-Pipeline vom Einchecken des Codes über das Testen und die Bereitstellung bis hin zur Validierung. Dies reduziert die Vorlaufzeit, senkt die Kosten, ermöglicht häufigere Änderungen, minimiert den Aufwand und verbessert die Zusammenarbeit.

Implementierungsschritte



Diagramm, das eine CI/CD-Pipeline mit AWS CodePipeline und zugehörigen Services zeigt

1. Nutzen Sie AWS CodeCommit zur Versionskontrolle und zum Speichern und Verwalten von Ressourcen (wie Dokumente, Quellcode und Binärdateien).

2. Nutzen Sie CodeBuild, um den Quellcode zu kompilieren, Komponententests auszuführen und Artefakte zu erzeugen, die sofort bereitgestellt werden können.
3. Nutzen Sie CodeDeploy als Bereitstellungsservice, der Anwendungsbereitstellungen für Amazon EC2-Instances, On-Premises-Instances, AWS Lambda-Serverless-Funktionenoder Amazon ECSautomatisiert.
4. Überwachen Sie Ihre Bereitstellungen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS06-BP04 Automatisieren von Tests und Rollback

Zugehörige Dokumente:

- AWS Developer Tools (AWS-Entwicklertools)
- Was ist AWS CodeCommit?
- Was ist AWS CodeBuild?
- AWS CodeBuild
- Was ist AWS CodeDeploy?

Zugehörige Videos:

- AWS re:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS (AWS re:Invent 2022 – AWS Well-Architected Best Practices für DevOps in AWS)

OPS05-BP05 Durchführen der Patch-Verwaltung

Führen Sie eine Patch-Verwaltung durch, um Funktionen zu erhalten, Probleme zu beheben und die Konformität mit der Governance zu gewährleisten. Automatisieren Sie die Patch-Verwaltung, um Fehler aufgrund manueller Prozesse zu reduzieren, zu skalieren und den Aufwand für die Installation von Patches zu verringern.

Patch- und Schwachstellenmanagement sind Teil Ihrer Vorteile- und Risikomanagement-Aktivitäten. Es ist vorzuziehen, unveränderliche Infrastrukturen zu haben und Workloads in verifizierten

bekannten guten Zuständen bereitzustellen. Wenn dies nicht realisierbar ist, ist das Patchen die verbleibende Option.

Amazon EC2 Image Builder stellt Pipelines zur Aktualisierung von Machine Images bereit. Als Teil der Patch-Verwaltung nutzen Amazon Machine Images (AMIs) eine AMI-Image-Pipeline oder Container-Images eine Docker-Image-Pipeline,während AWS Lambda Muster für benutzerdefinierte Lambda-Laufzeiten und zusätzliche Bibliotheken bietet, um Sicherheitslücken zu beseitigen.

Sie sollten Updates für Amazon Machine Images für Linux- oder Windows Server-Images mit Amazon EC2 Image Builderverwalten. Sie können Amazon Elastic Container Registry (Amazon ECR) mit Ihrer bestehenden Pipeline zur Verwaltung von Amazon ECS-Images und von Amazon EKS-Images nutzen. Lambda beinhaltet Versionsmanagementfunktionen.

Patches sollten nicht auf Produktionssystemen ohne erste Tests in einer sicheren Umgebung durchgeführt werden. Patches sollten nur angewendet werden, wenn sie ein betriebliches oder geschäftliches Ergebnis unterstützen. In AWS können Sie AWS Systems Manager Patch Manager verwenden, um das Patchen verwalteter Systeme zu automatisieren und die Aktivitäten mithilfe von Systems Manager-Wartungsfenstern zu planen.

Gewünschtes Ergebnis: Ihre AMI und Container-Images sind gepatcht, aktuell und startbereit. Sie können den Status aller bereitgestellten Images nachverfolgen und wissen, dass die Patches konform sind. Sie können über den aktuellen Status berichten und verfügen über ein Verfahren, mit dem Sie Ihre Compliance-Anforderungen erfüllen können.

Typische Anti-Muster:

- Sie erhalten den Auftrag, alle neuen Sicherheits-Patches innerhalb von zwei Stunden anzuwenden, was zu mehreren Ausfällen aufgrund der Anwendungsinkompatibilität mit bestimmten Patches führt.
- Eine ungepatchte Bibliothek hat unbeabsichtigte Folgen, weil unbekannte Personen Schwachstellen darin ausnutzen, um auf Ihren Workload zuzugreifen.
- Sie patchen die Entwicklerumgebungen automatisch, ohne die Entwickler zu benachrichtigen. Sie erhalten mehrere Beschwerden von den Entwicklern, dass ihre Umgebung nicht mehr wie erwartet funktioniert.
- Sie haben die kommerziell im Handel erhältliche Software auf einer persistenten Instance nicht gepatcht. Als ein Problem mit der Software auftritt und Sie sich an den Anbieter wenden, werden Sie darüber informiert, dass die Version nicht unterstützt wird und Sie bestimmte Patches installieren müssen, um Unterstützung zu erhalten.

- Ein kürzlich veröffentlichter Patch für Ihre verwendete Verschlüsselungssoftware bietet signifikante Leistungsverbesserungen. Ihr ungepatchtes System weist Leistungsprobleme auf, die bestehen bleiben, weil es nicht gepatcht ist.
- Sie werden über eine Zero-Day-Schwachstelle informiert, die eine Notfalllösung erfordert, und Sie müssen alle Ihre Umgebungen manuell patchen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Einrichtung eines Patch-Verwaltungsprozesses, einschließlich Ihrer Patching-Kriterien und Bereitstellungsmethodik für Ihre Umgebungen, können Sie die Patch-Ebenen skalieren und Berichte darüber erstellen. Das gibt Ihnen Sicherheit in Bezug auf Sicherheitspatches und gewährleistet einen klaren Überblick über den Status bekannter Problemlösungen. Dies wiederum fördert die Übernahme der gewünschten Merkmale und Funktionen, das Entfernen von Problemen und die kontinuierliche Compliance. Implementieren Sie Verwaltungssysteme und Automatisierung für Patches, um den Aufwand für die Bereitstellung von Patches zu reduzieren und Fehler zu begrenzen, die durch manuelle Prozesse verursacht werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Installieren Sie auf Ihren Systemen Patches zur Behebung von Problemen, zur Erlangung der gewünschten Funktionen oder Fähigkeiten sowie zur kontinuierlichen Einhaltung der Governance-Richtlinien und der Anforderungen des Lieferantensupport. Nehmen Sie in unveränderlichen Systemen eine Bereitstellung mit einer geeigneten Patch-Gruppe vor, um das gewünschte Ergebnis zu erzielen. Automatisieren Sie den Mechanismus der Patch-Verwaltung, um die Patch-Zeit zu verkürzen, Fehler aufgrund von manuellen Prozessen zu vermeiden und den Aufwand für die Installation von Patches zu verringern.

Implementierungsschritte

Für Amazon EC2 Image Builder:

1. Wenn Sie Amazon EC2 Image Builder verwenden, geben Sie die Pipeline-Details an:
   a. Erstellen Sie eine Image-Pipeline und geben Sie ihr einen Namen.
   b. Definieren Sie den Pipeline-Zeitplan und die Zeitzone.
   c. Konfigurieren Sie alle Abhängigkeiten.
2. Wählen Sie ein Rezept:
   a. Wählen Sie ein vorhandenes Rezept aus oder erstellen Sie ein neues.

   b. Wählen Sie den Image-Typ aus.
   c. Geben Sie Ihrem Rezept einen Namen und eine Versionsnummer.
   d. Wählen Sie Ihr Basis-Image aus.
   e. Fügen Sie Build-Komponenten zur Zielregistrierung hinzu.
3. Optional: Definieren Sie Ihre Infrastrukturkonfiguration.
4. Optional: Definieren Sie die Konfigurationseinstellungen.
5. Überprüfen Sie die Einstellungen.
6. Achten Sie regelmäßig auf die Rezepthygiene.

Für Systems Manager Patch Manager:

1. Erstellen Sie eine Patch-Baseline.
2. Wählen Sie eine Methode für Pfadoperationen aus.
3. Aktivieren Sie Compliance-Berichte und -Scans.

Ressourcen

Zugehörige bewährte Methoden:

- OPS06-BP04 Automatisieren von Tests und Rollback

Zugehörige Dokumente:

- Was ist Amazon EC2 Image Builder?
- Create an image pipeline using the Amazon EC2 Image Builder (Erstellen einer Image-Pipeline mit dem Amazon EC2 Image Builder)
- Create a container image pipeline (Erstellen einer Container-Image-Pipeline)
- AWS Systems Manager Patch Manager
- Working with Patch Manager (Arbeiten mit Patch Manager)
- Working with patch compliance reports (Arbeiten mit Patch-Compliance-Berichten)
- AWS Developer Tools

Zugehörige Videos:

- CI/CD für Serverless Anwendungen in AWS
- Design mit Blick auf die Ops

Zugehörige Beispiele:

- Well-Architected Labs: Bestands- und Patch-Verwaltung
- Anleitungen zu AWS Systems Manager Patch Manager

OPS05-BP06 Gemeinsame Design-Standards

Tauschen Sie teamübergreifend bewährte Methoden aus, um das Bewusstsein zu schärfen und den Nutzen der Entwicklungsarbeit zu maximieren. Dokumentieren Sie sie und halten Sie sie auf dem neuesten Stand, wenn sich Ihre Architektur weiterentwickelt. Wenn gemeinsame Standards in Ihrem Unternehmen durchgesetzt werden, ist es wichtig, dass Mechanismen vorhanden sind, um Ergänzungen, Änderungen und Ausnahmen von Standards abzubilden. Ohne diese Option werden Standards zu einer Einschränkung der Innovation.

Gewünschtes Ergebnis: Designstandards werden von allen Teams in Ihren Organisationen gemeinsam genutzt. Sie werden dokumentiert und mit der Entwicklung bewährter Methoden auf dem neuesten Stand gehalten.

Typische Anti-Muster:

- Zwei Entwicklerteams haben jeweils einen Service zur Authentifizierung von Benutzern erstellt. Ihre Benutzer müssen für jeden Teil des Systems, auf den sie zugreifen möchten, eigene Anmeldeinformationen verwenden.
- Jedes Team verwaltet seine eigene Infrastruktur. Eine neue Compliance-Anforderung erzwingt eine Änderung Ihrer Infrastruktur. Jedes Team implementiert sie auf andere Weise.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung gemeinsamer Standards unterstützt die Umsetzung bewährter Methoden und maximiert den Nutzen der Entwicklungsarbeit. Die Dokumentation und Aktualisierung von Designstandards hält Ihre Organisation auf dem neuesten Stand bezüglich der bewährten Methoden und der Anforderungen an die Sicherheit und Compliance.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nutzen Sie bewährte Methoden, Designstandards, Checklisten, Arbeitsverfahren, Leitlinien und Governance-Anforderungen in allen Teams. Verwenden Sie Verfahren zur Anforderung von Änderungen, Ergänzungen und Ausnahmen von Designstandards, um Verbesserungen und Innovationen zu unterstützen. Stellen Sie sicher, dass die Teams über die veröffentlichten Inhalte informiert sind. Verwenden Sie ein System, um die Designstandards auf dem neuesten Stand zu halten, wenn neue bewährte Methoden eingeführt werden.

Kundenbeispiel

AnyCompany Retail verfügt über ein funktionsübergreifendes Architekturteam, das Softwarearchitekturmuster erstellt. Dieses Team entwickelt die Architektur mit integrierter Compliance und Governance. Teams, die diese gemeinsamen Standards anwenden, profitieren davon, dass Compliance und Governance bereits integriert sind. Sie können schnell auf dem Designstandard aufbauen. Das Architekturteam trifft sich vierteljährlich, um die Architekturmuster zu bewerten und sie gegebenenfalls zu aktualisieren.

Implementierungsschritte

1. Bestimmen Sie ein funktionsübergreifendes Team, das für die Entwicklung und Aktualisierung der Designstandards zuständig ist. Dieses Team sollte mit Stakeholdern in Ihrer gesamten Organisation zusammenarbeiten, um Designstandards, Arbeitsverfahren, Checklisten, Leitlinien und Governance-Anforderungen zu entwickeln. Dokumentieren Sie die Designstandards und geben Sie sie innerhalb Ihrer Organisation weiter.
   a. Mit AWS Service Catalog können Sie Portfolios erstellen, die Designstandards als Infrastructure-as-Code abbilden. Sie können Portfolios über Konten hinweg gemeinsam nutzen.
2. Verwenden Sie ein System, um die Designstandards auf dem neuesten Stand zu halten, wenn neue bewährte Methoden eingeführt werden.
3. Wenn Designstandards zentral durchgesetzt werden, sollten Sie über ein Verfahren verfügen, um Änderungen, Aktualisierungen und Ausnahmen anzufordern.

Aufwand für den Implementierungsplan: Mittel. Die Entwicklung eines Prozesses zur Erstellung und gemeinsamen Nutzung von Designstandards kann die Koordination und Zusammenarbeit mit Stakeholdern in Ihrer gesamten Organisation erforderlich machen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP03 Bewerten der Governance-Anforderungen - Governance-Anforderungen beeinflussen Designstandards.
- OPS01-BP04 Bewerten der Compliance-Anforderungen - Compliance ist ein wichtiger Faktor bei der Erstellung von Designstandards.
- OPS07-BP02 Sicherstellen einer konsistenten Prüfung der betrieblichen Bereitschaft - Checklisten für die operative Einsatzbereitschaft sind ein Mechanismus zur Umsetzung von Designstandards bei der Gestaltung Ihres Workloads.
- OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung - Die Aktualisierung von Designstandards ist ein Teil der kontinuierlichen Verbesserung.
- OPS11-BP04 Wissensmanagement - Als Teil Ihres Wissensmanagements sollten Sie Designstandards dokumentieren und weitergeben.

Zugehörige Dokumente:

- Automate AWS Backups with AWS Service Catalog (Automatisieren von AWS Backups mit AWS Service Catalog)
- AWS Service Catalog Account Factory-Enhanced (Erweiterte Nutzung von AWS Service Catalog Account Factory)
- How Expedia Group built Database as a Service (DBaaS) offering using AWS Service Catalog (So hat die Expedia Gruppe mit AWS Service Catalog ein Database-as-a-Service-Angebot (DBaaS) entwickelt)
- Maintain visibility over the use of cloud architecture patterns (Überblick über die Nutzung von Cloud-Architekturmustern)
- Simplify sharing your AWS Service Catalog portfolios in an AWS Organizations setup (Vereinfachen der gemeinsamen Nutzung Ihrer AWS Service Catalog-Portfolios in einem AWS Organizations-Setup)

Zugehörige Videos:

- AWS Service Catalog – Getting Started (AWS Service Catalog – Erste Schritte)
- AWS re:Invent 2020: Manage your AWS Service Catalog portfolios like an expert (AWS re:Invent 2020: Verwalten Ihrer AWS Service Catalog-Portfolios wie ein Experte)

Zugehörige Beispiele:

- AWS Service Catalog Reference Architecture (AWS Service Catalog-Referenzarchitektur)
- AWS Service Catalog-Workshop

Zugehörige Services:

- Mit AWS Service Catalog

OPS05-BP07 Implementieren von Verfahren zur Verbesserung der Codequalität

Implementieren Sie Verfahren zur Verbesserung der Codequalität und Minimierung von Fehlern. Einige Beispiele sind die testbasierte Entwicklung, Code-Reviews, die Einführung von Standards und Pair-Programming. Integrieren Sie diese Verfahren in Ihren Continuous-Integration- und delivery-Prozess.

Gewünschtes Ergebnis: Ihre Organisation setzt bewährte Methoden wie Code-Reviews oder Pair-Programming ein, um die Codequalität zu verbessern. Entwickler und operative Mitarbeiter nutzen bewährte Methoden zur Codequalität als Teil des Softwareentwicklungslebenszyklus.

Typische Anti-Muster:

- Sie führen ohne Code-Review Commits zum Main-Branch Ihrer Anwendung durch. Die Änderung wird automatisch in der Produktion bereitgestellt und verursacht einen Ausfall.
- Eine neue Anwendung wird ohne Unit-, End-to-End- oder Integrationstests entwickelt. Es gibt keine Möglichkeit, die Anwendung vor der Bereitstellung zu testen.
- Ihre Teams nehmen manuelle Änderungen in der Produktion vor, um Fehler zu beheben. Die Änderungen durchlaufen keine Tests oder Code-Reviews und werden nicht durch kontinuierliche Integrations- und Bereitstellungsprozesse erfasst oder protokolliert.

Vorteile der Nutzung dieser bewährten Methode: Durch die Einführung von Methoden zur Verbesserung der Codequalität können Sie dazu beitragen, Probleme in der Produktion zu minimieren. Die Codequalität erleichtert die Anwendung von bewährten Methoden wie Paarprogrammierung, Codeüberprüfungen und Implementierung von KI-Produktivitätstools.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Implementieren Sie Verfahren zur Verbesserung der Codequalität, um vor der Bereitstellung Fehler zu minimieren. Nutzen Sie Verfahren wie die testbasierte Entwicklung, Code-Reviews und Pair-Programming, um die Qualität Ihrer Entwicklung zu verbessern.

Nutzen Sie die Leistungsfähigkeit generativer KI mit Amazon Q Developer, um die Produktivität und Codequalität von Entwicklern zu verbessern. Amazon Q Developer umfasst die Generierung von Codevorschlägen (basierend auf großen Sprachmodellen), die Erstellung von Komponententests (einschließlich Randbedingungen) und Verbesserungen der Codesicherheit durch die Erkennung und Behebung von Sicherheitsschwachstellen.

Kundenbeispiel

AnyCompany Retail wendet verschiedene Verfahren an, um die Codequalität zu verbessern. Die testbasierte Entwicklung ist der Standard für die Entwicklung von Anwendungen. Bei einigen neuen Funktionen arbeiten die Entwickler während eines Sprints zusammen. Jede Pull-Anforderung wird von einem erfahrenen Entwickler überprüft, bevor sie integriert und bereitgestellt wird.

Implementierungsschritte

1. Setzen Sie bei Ihrem kontinuierlichen Integrations- und Bereitstellungsprozess auf Code-Qualitätsverfahren wie die testbasierte Entwicklung, Code-Reviews und Pair-Programming. Nutzen Sie diese Techniken, um die Softwarequalität zu verbessern.
   a. Verwenden Sie Amazon Q Developer, ein generatives KI-Tool, mit dem Sie Komponententestfälle (einschließlich Randbedingungen) erstellen, Funktionen mithilfe von Code und Kommentaren generieren, bekannte Algorithmen implementieren, Verstöße gegen Sicherheitsrichtlinien und Sicherheitsschwachstellen in Ihrem Code erkennen, Secrets erkennen, Infrastruktur as Code (IaC) scannen, Code dokumentieren und Codebibliotheken von Drittanbietern schneller erlernen können.
   b. Amazon CodeGuru Reviewer kann Machine-Learning-Programmierempfehlungen für Java- und Python-Code bereitstellen.
   c. Mit AWS Cloud9 können Sie gemeinsame Entwicklungsumgebungen schaffen, in denen Sie gemeinsam an der Codeentwicklung arbeiten können.

Aufwand des Implementierungsplans: mittel. Es gibt viele Möglichkeiten zur Umsetzung dieser bewährten Methode. Es kann jedoch schwierig sein, die Akzeptanz im Unternehmen zu erreichen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP02 Testen und Validieren von Änderungen
- OPS05-BP06 Gemeinsame Design-Standards

Zugehörige Dokumente:

- Einführung eines testgesteuerten Entwicklungsansatzes
- Beschleunigen Ihres Softwareentwicklungszyklus mit Amazon Q
- Amazon Q Developer, jetzt allgemein verfügbar, enthält eine Vorschau auf neue Funktionen, mit denen das Entwicklererlebnis neu gestaltet werden kann
- Der ultimative Spickzettel für die Verwendung von Amazon Q Developer in Ihrer IDE
- Shift-Left-Workload, Nutzung von KI für die Testerstellung
- Amazon Q Developer Center
- 10 Methoden für eine schnellere Entwicklung von Anwendungen mit Amazon CodeWhisperer
- Ein Blick über die Codeabdeckung hinaus mit Amazon CodeWhisperer
- Bewährte Methoden für Prompt-Engineering mit Amazon CodeWhisperer
- Leitfaden für agile Software
- Meine CI/CD-Pipeline ist mein Release Captain
- Automatisieren von Code-Reviews mit Amazon CodeGuru Reviewer
- Einführung eines testgesteuerten Entwicklungsansatzes
- So entwickelt DevFactory bessere Anwendungen mit Amazon CodeGuru
- Über Pair-Programming
- RENGA Inc. automatisiert Code-Reviews mit Amazon CodeGuru
- Die Kunst der agilen Entwicklung: Testbasierte Entwicklung
- Warum Code-Reviews wichtig sind (und tatsächlich Zeit sparen!)

Zugehörige Videos:

- Implementieren einer API mit dem Amazon Q Developer-Agenten für Softwareentwicklung

- Installation, Konfiguration und Verwendung von Amazon Q Developer mit JetBrains-IDEs (Anleitung)
- Beherrschung der Kunst von Amazon CodeWhisperer – YouTube-Playlist
- AWS re:Invent 2020: Kontinuierliche Verbesserung der Codequalität mit Amazon CodeGuru
- AWS Summit ANZ 2021 – Vorantreiben einer „Test-First“-Strategie mit CDK und testgesteuerter Entwicklung

Zugehörige Services:

- Amazon Q Developer
- Amazon CodeGuru Reviewer
- Amazon CodeGuru Profiler
- AWS Cloud9

OPS05-BP08 Verwenden mehrerer Umgebungen

Verwenden Sie mehrere Umgebungen, um Ihren Workload auszuprobieren, zu entwickeln und zu testen. Verwenden Sie zunehmende Kontrollstufen, wenn Umgebungen sich der Produktion nähern, um sicherzustellen, dass Ihr Workload bei der Bereitstellung wie beabsichtigt funktioniert.

Gewünschtes Ergebnis: Sie verfügen über mehrere Umgebungen, die Ihre Compliance- und Governance-Anforderungen widerspiegeln. Auf Ihrem Weg zur Produktion testen und promoten Sie Code in Umgebungen.

Typische Anti-Muster:

- Sie führen die Entwicklung in einer gemeinsamen Entwicklungsumgebung durch und ein weiterer Entwickler überschreibt Ihre Codeänderungen.
- Die restriktiven Sicherheitskontrollen Ihrer gemeinsamen Entwicklungsumgebung verhindern, dass Sie mit neuen Services und Funktionen experimentieren können.
- Sie führen Belastungstests auf Ihren Produktionssystemen durch und verursachen einen Ausfall für Ihre Benutzer.
- In der Produktion ist ein kritischer Fehler aufgetreten, der zum Verlust von Daten geführt hat. In Ihrer Produktionsumgebung versuchen Sie, die Bedingungen, die zum Datenverlust geführt haben, nachzustellen, damit Sie die Ursache feststellen und beseitigen können. Um einen weiteren

Datenverlust während des Testens zu verhindern, müssen Sie die Anwendung für Ihre Benutzer deaktivieren.

- Sie betreiben einen Mehrmandanten-Service und können eine Kundenanfrage nach einer eigenen Umgebung nicht erfüllen.
- Möglicherweise testen Sie nicht immer, aber wenn Sie dies tun, testen Sie in Ihrer Produktionsumgebung.
- Sie glauben, dass die Einfachheit einer einzelnen Umgebung die Auswirkungen von Änderungen innerhalb der Umgebung ausgleicht.

Vorteile der Nutzung dieser bewährten Methode: Sie können gleichzeitig mehrere Entwicklungs-, Test- und Produktionsumgebungen unterstützen, ohne Konflikte zwischen Entwicklern oder User-Communities zu erzeugen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Verwenden Sie mehrere Umgebungen und stellen Sie den Entwicklern Sandbox-Umgebungen mit weniger Kontrollen zur Verfügung, in denen sie experimentieren können. Richten Sie individuelle Entwicklungsumgebungen ein, damit parallele Arbeit möglich ist. Dadurch steigern Sie die Agilität der Entwicklung. Implementieren Sie strengere Kontrollen erst in den Umgebungen, die kurz vor der Produktionsaufnahme stehen, damit Entwickler Innovationen schaffen können. Nutzen Sie die Infrastruktur als Code sowie Konfigurationsverwaltungssysteme, um Umgebungen bereitzustellen, die mit den in der Produktion vorhandenen Kontrollen einheitlich konfiguriert sind. Auf diese Weise können Sie sicherstellen, dass die Systeme bei der Bereitstellung wie erwartet funktionieren. Wenn Umgebungen nicht in Gebrauch sind, schalten Sie sie ab, um Kosten für ungenutzte Ressourcen zu vermeiden (z. B. Entwicklungssysteme am Abend und am Wochenende). Stellen Sie beim Belastungstest produktionsgleiche Umgebungen bereit, um die Gültigkeit der Ergebnisse zu verbessern.

Ressourcen

Zugehörige Dokumente:

- Instance Scheduler on AWS (Instance Scheduler in AWS)
- Was ist AWS CloudFormation?

OPS05-BP09 Häufige, kleine, reversible Änderungen vornehmen

Häufige, kleine und reversible Änderungen verringern den Umfang und die Auswirkung einer Änderung. In Verbindung mit Change-Management-Systemen, Systemen zur Konfigurationsverwaltung und Build- und Liefersystemen reduzieren häufige, kleine und reversible Änderungen den Umfang und die Auswirkungen einer Änderung. Dies macht die Fehlersuche effizienter und ermöglicht eine schnellere Korrektur, da die Möglichkeit besteht, Änderungen zurückzusetzen.

Typische Anti-Muster:

- Sie stellen vierteljährlich eine neue Version Ihrer Anwendung mit einem Änderungsfenster bereit, was bedeutet, dass ein zentraler Dienst ausgeschaltet wird.
- Sie nehmen häufig Änderungen an Ihrem Datenbankschema vor, ohne Änderungen in Ihren Managementsystemen nachzuverfolgen.
- Sie führen direkte manuelle Updates durch, überschreiben damit bestehende Installationen und Konfigurationen und haben keinen klaren Rollback-Plan.

Vorteile der Nutzung dieser bewährten Methode: Sie profitieren schneller von den Entwicklungsarbeiten, wenn Sie häufig kleine Änderungen bereitstellen. Wenn die Änderungen klein sind, ist es viel einfacher zu erkennen, ob sie unbeabsichtigte Folgen haben, und sie lassen sich leichter rückgängig machen. Wenn die Änderungen rückgängig gemacht werden können, ist die Implementierung mit geringeren Risiken verbunden, da die Wiederherstellung einfacher ist. Der Änderungsprozess hat ein geringeres Risiko und die Auswirkungen einer fehlgeschlagenen Änderung werden reduziert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Machen Sie häufige, kleine und reversible Änderungen und verringern Sie dadurch den Umfang und die Auswirkung einer Änderung. Dies erleichtert die Fehlersuche, trägt zur Beschleunigung der Fehlerbehebung bei und bietet die Möglichkeit, eine Änderung zurückzusetzen. Außerdem profitiert Ihr Unternehmen schneller von neuen Entwicklungen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung

- OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung.
- OPS06-BP04 Automatisieren von Tests und Rollback

Zugehörige Dokumente:

- Implementieren von Microservices in AWS
- Microservices – Beobachtbarkeit

OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung

Automatisieren Sie den Aufbau, die Bereitstellung und die Tests des Workloads. Dadurch werden Fehler aufgrund von manuellen Prozessen und der Aufwand für die Bereitstellung von Änderungen verringert.

Wenden Sie Metadaten mithilfe von Ressourcen-Tags und AWS Resource Groups nach einer konsistenten Markierungsstrategie an, um die Identifizierung Ihrer Ressourcen zu erleichtern. Versehen Sie Ihre Ressourcen mit Tags für Organisation, Kostenkalkulation, Zugriffssteuerung und Zielrichtung der Ausführung von automatisierten Betriebsaktivitäten.

Gewünschtes Ergebnis: Entwickler verwenden Tools, um Code bereitzustellen und bis zur Produktion zu unterstützen. Entwickler müssen sich nicht bei der AWS Management Console anmelden, um Updates bereitzustellen. Es gibt einen vollständigen Audit Trail für Änderungen und Konfigurationen, der die Governance- und Compliance-Anforderungen erfüllt. Prozesse sind wiederholbar und teamübergreifend standardisiert. Entwickler sind in der Lage, sich auf die Entwicklung und Code-Pushs zu konzentrieren und so die Produktivität zu steigern.

Typische Anti-Muster:

- Am Freitag schließen Sie die Erstellung des neuen Codes für Ihren Funktionszweig ab. Am Montag, nach dem Ausführen Ihrer Skripts für die Codequalitätstests und einzelnen Komponententests, überprüfen Sie Ihren Code für den nächsten geplanten Release.
- Sie erhalten die Aufgabe, eine Korrektur für ein kritisches Problem zu schreiben, das sich auf eine große Anzahl von Kunden in der Produktion auswirkt. Nachdem Sie die Korrektur getestet haben, übergeben Sie Ihren Code und fordern beim Änderungsmanagement die Bereitstellungsgenehmigung zur Produktion an.
- Als Entwickler melden Sie sich bei der AWS Management Console an, um eine neue Entwicklungsumgebung mit nicht standardmäßigen Methoden und Systemen zu erstellen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Implementierung automatisierter Build- und Bereitstellungsverwaltungssysteme reduzieren Sie Fehler aus manuellen Prozessen und den Aufwand für die Bereitstellung von Änderungen, sodass sich Ihre Teammitglieder besser auf die Wertschöpfung konzentrieren können. Sie erhöhen die Liefergeschwindigkeit auf Ihrem Weg zur Produktion.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Verwenden Sie Systeme zur Build- und Bereitstellungsverwaltung für die Verfolgung und Implementierung von Änderungen, die Reduzierung von Fehlern, die durch manuelle Prozesse entstehen, sowie zur Verringerung des Aufwands. Nutzen Sie eine vollständig automatisierte Integrations- und Bereitstellungs-Pipeline vom Einchecken des Codes über das Testen und die Bereitstellung bis hin zur Validierung. Dies reduziert die Vorlaufzeit, fördert häufigere Änderungen, reduziert den Aufwand, beschleunigt die Markteinführung, führt zu einer höheren Produktivität und erhöht die Sicherheit Ihres Codes bis hin zur Produktion.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung
- OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung.

Zugehörige Dokumente:

- Was ist AWS CodeBuild?
- Was ist AWS CodeDeploy?

Zugehörige Videos:

- AWS re\:Invent 2022 - AWS Well-Architected best practices for DevOps on AWS (AWS re\:Invent 2022 – AWS Well-Architected Best Practices für DevOps in AWS)

OPS 6. Wie können Sie Bereitstellungsrisiken eindämmen?

Verwenden Sie Ansätze, die schnelles Feedback zur Qualität liefern und eine schnelle Wiederherstellung bei Änderungen ermöglichen, die nicht zu den gewünschten Ergebnissen

führen. Mit diesen Verfahren können Sie die Auswirkung von Problemen eindämmen, die durch die Bereitstellung von Änderungen entstehen.

Bewährte Methoden

- OPS06-BP01 Einkalkulieren nicht erfolgreicher Änderungen
- OPS06-BP02 Testbereitstellungen
- OPS06-BP03 Einsetzen sicherer Bereitstellungsstrategien
- OPS06-BP04 Automatisieren von Tests und Rollback

OPS06-BP01 Einkalkulieren nicht erfolgreicher Änderungen

Planen Sie Maßnahmen für die Rückkehr zu einem bekanntermaßen funktionierenden Zustand oder die Korrektur in der Produktionsumgebung ein, falls bei der Bereitstellung ein nicht erwünschtes Ergebnis auftritt. Eine Richtlinie zur Festlegung eines solchen Plans hilft allen Teams, Strategien zum Umgang mit fehlgeschlagenen Änderungen zu entwickeln. Einige Beispiele für Strategien sind Bereitstellungs- und Rollback-Schritte, Änderungsrichtlinien, Feature-Flags sowie die Isolierung und Verlagerung von Datenverkehr. Ein einzelner Release kann mehrere zusammengehörige Komponentenänderungen enthalten. Die Strategie sollte die Möglichkeit bieten, dem Ausfall einer Komponentenänderung standzuhalten oder sich danach zu regenerieren.

Gewünschtes Ergebnis: Sie haben einen detaillierten Wiederherstellungsplan für Ihre Änderung erstellt, falls diese nicht erfolgreich sein sollte. Darüber hinaus haben Sie die Größe Ihres Releases reduziert, um die potenziellen Auswirkungen auf andere Workload-Komponenten zu minimieren. Infolgedessen haben Sie die Auswirkungen auf Ihr Unternehmen verringert, indem Sie die potenziellen Ausfallzeiten aufgrund einer fehlgeschlagenen Änderung reduziert und die Flexibilität und Effizienz der Wiederherstellungszeiten erhöht haben.

Typische Anti-Muster:

- Sie haben Code bereitgestellt und Ihre Anwendung ist instabil geworden, aber es befinden sich aktive Benutzer im System. Sie müssen entscheiden, ob Sie die Änderung rückgängig machen und Auswirkungen auf die aktiven Benutzer in Kauf nehmen möchten, oder ob Sie die Änderung erst später rückgängig machen möchten, wodurch möglicherweise trotzdem Auswirkungen auf die Benutzer entstehen könnten.
- Nachdem Sie eine Routineänderung vorgenommen haben, kann auf Ihre neuen Umgebungen zugegriffen werden, aber eines Ihrer Subnetze ist nicht mehr erreichbar. Sie müssen entscheiden, ob Sie die gesamte Änderung rückgängig machen oder versuchen, die Nichtverfügbarkeit des

Subnetzes zu beheben. Während Sie diese Entscheidung abwägen, bleibt das Subnetz nicht erreichbar.

- Ihre Systeme sind nicht so konzipiert, dass sie mit kleineren Releases aktualisiert werden können. Daher haben Sie Schwierigkeiten, die Bulk-Änderungen während einer fehlgeschlagenen Bereitstellung rückgängig zu machen.
- Sie verwenden nicht Infrastructure as Code (IaC) und Sie haben manuelle Aktualisierungen an Ihrer Infrastruktur vorgenommen, die zu einer unerwünschten Konfiguration geführt haben. Sie sind nicht in der Lage, die manuellen Änderungen effektiv zu verfolgen und rückgängig zu machen.
- Da Sie die erhöhte Häufigkeit Ihrer Bereitstellungen nicht gemessen haben, hat Ihr Team keinen Anreiz, den Umfang seiner Änderungen zu reduzieren und seine Rollback-Pläne für jede Änderung zu verbessern. Dies führt zu höheren Risiken und höheren Ausfallraten.
- Sie messen nicht die Gesamtdauer eines Ausfalls, der durch erfolglose Änderungen verursacht wird. Ihr Team ist nicht in der Lage, den Bereitstellungsprozess und die Effektivität des Wiederherstellungsplans zu priorisieren und zu verbessern.

Vorteile der Nutzung dieser bewährten Methode: Ein Plan zur Wiederherstellung nach erfolglosen Änderungen minimiert die mittlere Wiederherstellungszeit (MTTR) und reduziert die Auswirkungen auf Ihr Unternehmen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Mithilfe einer konsistenten, dokumentierten Richtlinie und Praxis, die von den Release-Teams angewendet wird, kann ein Unternehmen planen, was bei nicht erfolgreichen Änderungen passieren soll. Unter bestimmten Umständen sollte die Richtlinie ein Forward-Fixing berücksichtigen. In allen Fällen sollte ein Fix-Forward- oder Rollback-Plan vor der Bereitstellung in der Live-Produktion gut dokumentiert und getestet werden, um die benötigte Zeit zum Rückgängigmachen einer Änderung zu minimieren.

Implementierungsschritte

1. Dokumentieren Sie die Richtlinien, nach denen Teams über wirksame Pläne verfügen müssen, wie Änderungen innerhalb eines bestimmten Zeitraums rückgängig gemacht werden können.
   a. In den Richtlinien sollte festgelegt sein, wann eine Fix-Forward-Situation zulässig ist.
   b. Fordern Sie einen dokumentierten Rollback-Plan, auf den alle Beteiligten zugreifen können.

   c. Geben Sie die Anforderungen für das Rollback an (z. B. wenn festgestellt wird, dass nicht autorisierte Änderungen vorgenommen wurden).

2. Analysieren Sie den Grad der Auswirkungen aller Änderungen für jede Komponente eines Workloads.
   a. Ermöglichen Sie die Standardisierung, Vorlagenerstellung und Vorautorisierung wiederholbarer Änderungen, sofern sie einem konsistenten Workflow folgen, der Änderungsrichtlinien durchsetzt.
   b. Reduzieren Sie die potenziellen Auswirkungen jeder Änderung, indem Sie den Umfang der Änderung verringern, damit die Wiederherstellung weniger Zeit in Anspruch nimmt und weniger Auswirkungen auf das Unternehmen hat.
   c. Stellen Sie sicher, dass die Rollback-Verfahren den Code in einen bekannt funktionierenden Zustand zurückversetzen, um Zwischenfälle nach Möglichkeit zu vermeiden.
3. Integrieren Sie Tools und Workflows, um Ihre Richtlinien programmgesteuert durchzusetzen.
4. Machen Sie Daten zu Änderungen für andere Workload-Besitzer sichtbar, um die Diagnose bei fehlgeschlagenen Änderungen, für die kein Rollback möglich ist, zu beschleunigen.
   a. Messen Sie den Erfolg dieser Methode anhand sichtbarer Änderungsdaten und identifizieren Sie iterative Verbesserungen.
5. Verwenden Sie Überwachungstools, um den Erfolg oder Misserfolg einer Bereitstellung zu überprüfen und so die Entscheidungsfindung beim Rollback zu beschleunigen.
6. Messen Sie die Dauer des Ausfalls bei einer erfolglosen Änderung, um Ihre Wiederherstellungspläne kontinuierlich zu verbessern.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS06-BP04 Automatisieren von Tests und Rollback

Zugehörige Dokumente:

- AWS Builders' Library | Gewährleistung der Rollback-Sicherheit bei Bereitstellungen
- AWS Whitepaper | Änderungsmanagement in der Cloud

Zugehörige Videos:

- re:Invent 2019 | Amazon’s approach to high-availability deployment (re:Invent 2019 | Der Amazon-Ansatz für die Hochverfügbarkeitsbereitstellung)

OPS06-BP02 Testbereitstellungen

Testen Sie Release-Verfahren in der Vorproduktion, indem Sie dieselbe Bereitstellungskonfiguration, dieselben Sicherheitskontrollen, Schritte und Verfahren wie in der Produktion verwenden. Stellen Sie sicher, dass alle bereitgestellten Schritte wie erwartet abgeschlossen wurden, z. B. das Überprüfen von Dateien, Konfigurationen und Services. Testen Sie alle Änderungen darüber hinaus mit Funktions-, Integrations- und Auslastungstests sowie Überwachungsverfahren, z. B. Zustandsprüfungen. Durch diese Tests können Sie Bereitstellungsprobleme frühzeitig erkennen und haben die Möglichkeit, sie vor der Produktion einzuplanen und zu beheben.

Sie können temporäre parallele Umgebungen erstellen, um jede Änderung zu testen. Automatisieren Sie die Bereitstellung der Testumgebungen mithilfe von Infrastructure as Code (IaC), um den Arbeitsaufwand zu reduzieren und Stabilität, Konsistenz und schnellere Funktionsbereitstellung zu gewährleisten.

Gewünschtes Ergebnis: Ihr Unternehmen führt eine testgestützte Entwicklungskultur ein, die Testbereitstellungen einschließt. Dadurch wird sichergestellt, dass sich die Teams darauf konzentrieren, Werte für das Unternehmen zu schaffen, anstatt Releases zu verwalten. Die Teams werden bei der Identifizierung von Bereitstellungsrisiken frühzeitig einbezogen, um die geeigneten Maßnahmen zur Risikominderung festzulegen.

Typische Anti-Muster:

- Während Produktionseinführungen führen ungetestete Bereitstellungen häufig zu Problemen, die eine Fehlerbehebung und Eskalation erfordern.
- Ihr Release enthält Infrastructure as Code (IaC), wodurch vorhandene Ressourcen aktualisiert werden. Sie sind sich nicht sicher, ob IaC erfolgreich ausgeführt wird oder ob es Auswirkungen auf die Ressourcen gibt.
- Sie stellen eine neue Funktion für Ihre Anwendung bereit. Sie funktioniert nicht wie beabsichtigt und dies fällt erst auf, wenn sie von betroffenen Benutzern gemeldet wird.
- Sie aktualisieren Ihre Zertifikate. Sie installieren versehentlich die Zertifikate für die falschen Komponenten, was unentdeckt bleibt und Auswirkungen auf Website-Benutzer hat, da keine sichere Verbindung zur Website hergestellt werden kann.

Vorteile der Nutzung dieser bewährten Methode: Durch umfangreiche Tests der Bereitstellungsverfahren und der durch sie eingeführten Änderungen in der Vorproduktion werden die potenziellen Auswirkungen der Bereitstellungsschritte auf die Produktion minimiert. Dies erhöht das Vertrauen bei der Produktionseinführung und minimiert den Support während des Betriebs, ohne die bereitgestellten Änderungen zu verlangsamen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Das Testen Ihres Bereitstellungsprozesses ist genauso wichtig wie das Testen der Änderungen, die sich aus der Bereitstellung ergeben. Dies kann erreicht werden, indem Sie Ihre Bereitstellungsschritte in einer Vorproduktionsumgebung testen, die die Produktion so genau wie möglich widerspiegelt. Häufig auftretende Probleme, z. B. unvollständige oder falsche Bereitstellungsschritte oder Fehlkonfigurationen, können so vor der Bereitstellung in der Produktionsumgebung erkannt werden. Darüber hinaus können Sie Ihre Wiederherstellungsschritte testen.

Kundenbeispiel

Im Rahmen seiner CI/CD-Pipeline (Continuous Integration and Continuous Delivery) führt AnyCompany Retail die definierten Schritte durch, die zur Veröffentlichung von Infrastruktur- und Softwareupdates für seine Kunden in einer produktionsähnlichen Umgebung erforderlich sind. Die Pipeline besteht aus Vorabprüfungen zur Erkennung von Abweichungen (Erkennung von Änderungen an Ressourcen, die außerhalb von IaC vorgenommen wurden) bei Ressourcen vor der Bereitstellung sowie zur Validierung der Aktionen, die von IaC bei der Initiierung ausgeführt werden. Vor der erneuten Registrierung beim Load Balancer werden Bereitstellungsschritte validiert und z. B. sichergestellt, dass bestimmte Dateien und Konfigurationen vorhanden sind und Services ausgeführt werden und korrekt auf Zustandsprüfungen auf dem lokalen Host reagieren. Darüber hinaus führen alle Änderungen zu einer Reihe automatisierter Tests wie Funktions-, Sicherheits-, Regressions-, Integrations- und Auslastungstests.

Implementierungsschritte

1. Führen Sie Prüfungen vor der Installation durch, um die Vorproduktionsumgebung in der Produktionsumgebung zu spiegeln.
   a. Mit der Abweichungserkennung können Sie erkennen, wann Ressourcen außerhalb von AWS CloudFormation geändert wurden.

   b. Verwenden Sie Änderungssätze, um zu überprüfen, ob die Absicht einer Stack-Aktualisierung mit den Aktionen übereinstimmt, die von AWS CloudFormation bei der Initiierung des Änderungssatzes ausgeführt werden.
2. Dadurch wird ein manueller Genehmigungsschritt in AWS CodePipeline ausgelöst, um die Bereitstellung in der Vorproduktionsumgebung zu autorisieren.
3. Verwenden Sie Bereitstellungskonfigurationen wie AWS CodeDeploy-AppSpec- Dateien zur Definition der Bereitstellungs- und Validierungsschritte.
4. Wo zutreffend, integrieren Sie AWS CodeDeploy in andere AWS-Services oder integrieren Sie AWS CodeDeploy in Produkte und Services von Partnern.
5. Überwachen Sie Bereitstellungen mithilfe von Ereignisbenachrichtigungen von Amazon CloudWatch, AWS CloudTrail und Amazon SNS.
6. Führen Sie nach der Bereitstellung automatisierte Tests durch, einschließlich Funktions-, Sicherheits-, Regressions-, Integrations- und Auslastungstests.
7. Behandlung von Problemen bei der Bereitstellung.
8. Eine erfolgreiche Validierung der zuvor genannten Schritte sollte einen manuellen Genehmigungsworkflow initiieren, um die Bereitstellung in der Produktion zu autorisieren.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP02 Testen und Validieren von Änderungen

Zugehörige Dokumente:

- AWS Builders' Library | Automatisierung sicherer, vollautomatischer Bereitstellungen | Testbereitstellungen
- AWS-Whitepaper | Durchführung von dauerhafter Integration/dauerhafter Bereitstellung in AWS
- The Story of Apollo – Amazon's Deployment Engine (Apollo – die Bereitstellungs-Engine von Amazon)
- Vorgehensweise für den lokalen Test und lokales Debugging von AWS CodeDeploy vor der Auslieferung Ihres Codes

- Integrating Network Connectivity Testing with Infrastructure Deployment (Integration von Netzwerkkonnektivitätstests in die Bereitstellung der Infrastruktur)

Zugehörige Videos:

- re:Invent 2020 | Testing software and systems at Amazon (re:Invent 2020 | Testen von Software und Systemen bei Amazon)

Zugehörige Beispiele:

- Tutorial | Bereitstellen eines Amazon ECS-Services mit einem Validierungstest

OPS06-BP03 Einsetzen sicherer Bereitstellungsstrategien

Sichere Produktionseinführungen steuern den Fluss vorteilhafter Änderungen mit dem Ziel, die von den Kunden wahrgenommenen Auswirkungen dieser Änderungen zu minimieren. Die Sicherheitskontrollen bieten Prüfmechanismen, um die gewünschten Ergebnisse zu validieren und den Umfang der Auswirkungen von Fehlern zu begrenzen, die durch die Änderungen oder durch Fehler bei der Bereitstellung verursacht werden. Zu sicheren Rollouts können Strategien wie Feature-Flags, One-Box, Rolling (Canary-Releases), Immutable, Aufteilung des Datenverkehrs und Blau/Grün-Bereitstellungen gehören.

Gewünschtes Ergebnis: Ihr Unternehmen verwendet ein CI/CD-System (Continuous integration and continuous delivery, kontinuierliche Integration und kontinuierliche Bereitstellung), das Funktionen zur Automatisierung sicherer Rollouts bietet. Die Teams müssen angemessene sichere Rollout-Strategien anwenden.

Typische Anti-Muster:

- Sie stellen eine nicht erfolgreiche Änderung für die gesamte Produktion gleichzeitig bereit. Infolgedessen sind alle Kunden gleichzeitig betroffen.
- Ein Fehler, der bei einer gleichzeitigen Bereitstellung in allen Systemen auftritt, erfordert ein Notfall-Release. Die Korrektur für alle Kunden dauert mehrere Tage.
- Die Verwaltung der Produktionseinführung erfordert die Planung und Beteiligung mehrerer Teams. Dies schränkt Ihre Fähigkeit ein, Features für Ihre Kunden häufig zu aktualisieren.
- Sie führen eine veränderbare Bereitstellung durch, indem Sie Ihre vorhandenen Systeme ändern. Nachdem Sie festgestellt haben, dass die Änderung nicht erfolgreich war, müssen Sie die

Systeme erneut ändern, um die alte Version wiederherzustellen, was die Wiederherstellungsdauer verlängert.

Vorteile der Nutzung dieser bewährten Methode: Automatisierte Bereitstellungen sorgen für ein ausgewogenes Verhältnis zwischen der Geschwindigkeit der Bereitstellungen und der konsistenten Bereitstellung nützlicher Änderungen für die Kunden. Die Begrenzung der Auswirkungen verhindert kostspielige Bereitstellungsfehler und maximiert die Fähigkeit der Teams, effizient auf Ausfälle zu reagieren.

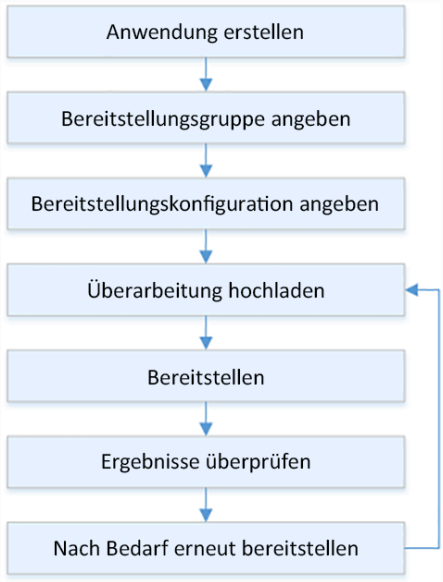
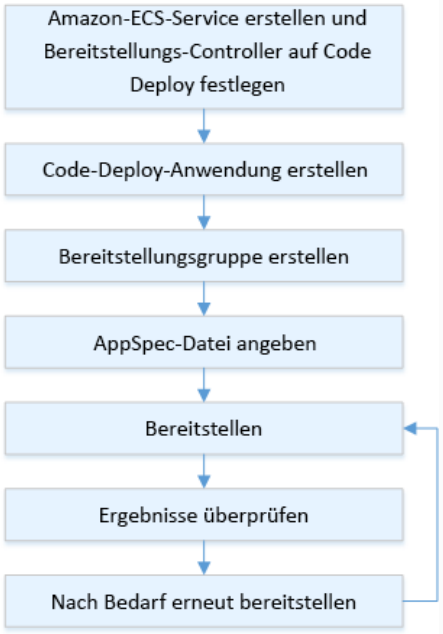
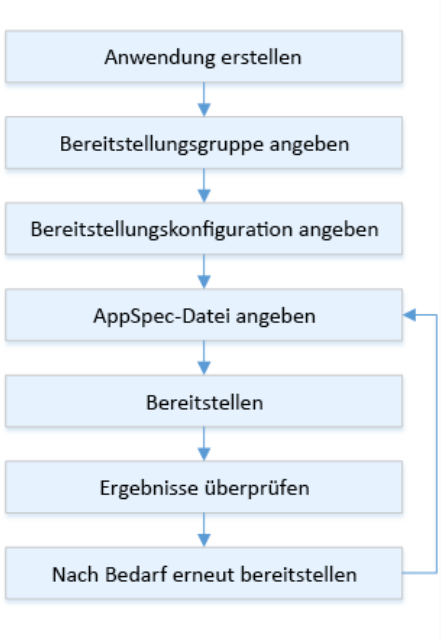
Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Ausfälle bei der kontinuierlichen Bereitstellung können zu einer verringerten Serviceverfügbarkeit und schlechten Kundenerfahrungen führen. Um die Anzahl erfolgreicher Implementierungen zu maximieren, sollten Sie im gesamten Release-Prozess Sicherheitskontrollen zur Minimierung von Bereitstellungsfehlern implementieren. Das Ziel sollte dabei sein, dass keine Bereitstellungsfehler auftreten.

Kundenbeispiel

AnyCompany Retail möchte Bereitstellungen mit minimalen bis gar keinen Ausfallzeiten erreichen, d. h. es soll während der Bereitstellung keine spürbaren Auswirkungen für die Benutzer geben. Um dies zu erreichen, hat das Unternehmen Bereitstellungsmuster festgelegt, z. B. fortlaufende und Blau/Grün-Bereitstellung (siehe nachfolgendes Workflow-Diagramm). Alle Teams übernehmen eines oder mehrere dieser Muster in ihre CI/CD-Pipeline.

CodeDeploy-Workflow für Amazon EC2	CodeDeploy-Workflow für Amazon ECS	CodeDeploy-Workflow für Lambda
		

Implementierungsschritte

1. Verwenden Sie einen Genehmigungsworkflow, um die Reihenfolge der Produktionseinführungsschritte nach der Beförderung zur Produktion einzuleiten.
2. Verwenden Sie ein automatisiertes Bereitstellungssystem wie AWS CodeDeploy. AWS CodeDeploy- Bereitstellungsoptionen schließen lokale Bereitstellungen für EC2/On-Premises und Blau/Grün-Bereitstellungen für EC2/On-Premises ein, AWS Lambda und Amazon ECS (siehe vorhergehendes Workflow-Diagramm).
   a. Wo zutreffend, integrieren Sie AWS CodeDeploy in andere AWS-Services oder integrieren Sie AWS CodeDeploy in Produkte und Services von Partnern.
3. Verwenden Sie Blau/Grün-Bereitstellungen für Datenbanken wie Amazon Aurora und Amazon RDS.
4. Überwachen Sie Bereitstellungen mithilfe von Ereignisbenachrichtigungen von Amazon CloudWatch, AWS CloudTrail und Amazon Simple Notification Service (Amazon SNS).
5. Führen Sie nach der Bereitstellung automatisierte Tests durch, einschließlich Funktions-, Sicherheits-, Regressions-, Integrations- und Auslastungstests.
6. Behandlung von Problemen bei der Bereitstellung.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP02 Testen und Validieren von Änderungen
- OPS05-BP09 Häufige, kleine, reversible Änderungen vornehmen
- OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung

Zugehörige Dokumente:

- AWS Builders' Library | Automatisierung sicherer, vollautomatischer Bereitstellungen | Produktionsbereitstellungen
- AWS Builders' Library | Meine CI/CD-Pipeline ist mein Release Captain | Sichere, automatische Produktionseinführungen
- AWS-Whitepaper | Durchführung von dauerhafter Integration/dauerhafter Bereitstellung in AWS | Bereitstellungsmethoden
- AWS CodeDeploy-Benutzerhandbuch
- Arbeiten mit Bereitstellungskonfigurationen in AWS CodeDeploy
- Einrichten einer API Gateway-Canary-Bereitstellung als Release
- Amazon ECS-Bereitstellungstypen
- Vollständig verwaltete Blau/Grün-Bereitstellungen in Amazon Aurora und Amazon RDS
- Blau/Grün-Bereitstellungen mit AWS Elastic Beanstalk

Zugehörige Videos:

- re:Invent 2020 | Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon
- re:Invent 2019 | Der Amazon-Ansatz für die Hochverfügbarkeitsbereitstellung

Zugehörige Beispiele:

- Testen einer Blau/Grün-Bereitstellung in AWS CodeDeploy
- Workshop | Erstellen von CI/CD-Pipelines für Lambda-Canary-Bereitstellungen mit AWS CDK

- Workshop | Blau/Grün- und Canary-Bereitstellungen für EKS und ECS
- Workshop | Erstellen einer kontenübergreifenden CI/CD-Pipeline

OPS06-BP04 Automatisieren von Tests und Rollback

Um die Geschwindigkeit, Zuverlässigkeit und Sicherheit Ihres Bereitstellungsprozesses zu erhöhen, sollten Sie eine Strategie für automatisierte Test- und Rollback-Funktionen in Vorproduktions- und Produktionsumgebungen entwickeln. Automatisieren Sie Tests bei der Bereitstellung in der Produktion, um Interaktionen zwischen Mensch und System zu simulieren und die bereitgestellten Änderungen zu überprüfen. Automatisieren Sie das Rollback, um schnell zu einem als funktionierend bekannten Zustand zurückkehren zu können. Das Rollback sollte unter vordefinierten Bedingungen automatisch eingeleitet werden, z. B. wenn das gewünschte Ergebnis einer Änderung nicht erreicht wird oder wenn der automatisierte Test fehlschlägt. Die Automatisierung dieser beiden Aktivitäten verbessert Ihre Erfolgsquote bei Bereitstellungen, minimiert die Wiederherstellungszeit und reduziert die potenziellen Auswirkungen auf das Unternehmen.

Gewünschtes Ergebnis: Ihre automatisierten Tests und Rollback-Strategien sind in Ihre CI/CD-Pipeline (Continuous Integration and Continuous Delivery, kontinuierliche Integration und kontinuierliche Bereitstellung) integriert. Ihre Überwachung kann Validierungen anhand Ihrer Erfolgskriterien ausführen und bei einem Fehler ein automatisches Rollback einleiten. Dadurch werden die Auswirkungen auf Endbenutzer und Kunden minimiert. Wenn beispielsweise alle Testergebnisse den Anforderungen entsprechen, übertragen Sie Ihren Code in die Produktionsumgebung, wo automatisierte Regressionstests unter Verwendung derselben Testfälle eingeleitet werden. Wenn die Ergebnisse der Regressionstests nicht den Erwartungen entsprechen, wird im Pipeline-Workflow ein automatisiertes Rollback eingeleitet.

Typische Anti-Muster:

- Ihre Systeme sind nicht so konzipiert, dass sie mit kleineren Releases aktualisiert werden können. Daher haben Sie Schwierigkeiten, die Bulk-Änderungen während einer fehlgeschlagenen Bereitstellung rückgängig zu machen.
- Ihr Bereitstellungsprozess besteht aus einer Reihe manueller Schritte. Nachdem Sie Änderungen an Ihrem Workload bereitgestellt haben, beginnen Sie mit den Tests nach der Bereitstellung. Danach bemerken Sie, dass Ihr Workload nicht mehr funktioniert und die Verbindung der Kunden getrennt wird. Sie starten das Rollback zur vorherigen Version. All diese manuellen Schritte verzögern die allgemeine Systemwiederherstellung und wirken sich nachhaltig auf Ihre Kunden aus.

- Sie haben Zeit dafür aufgewendet, automatisierte Testfälle für Funktionen zu entwickeln, die in Ihrer Anwendung nicht häufig verwendet werden. Dadurch amortisiert sich die Investition in Ihre automatisierten Testfunktionen nur schlecht.
- Ihre Version besteht aus Anwendungs-, Infrastruktur-, Patch- und Konfigurations-Updates, die voneinander unabhängig sind. Sie haben jedoch nur eine CI/CD-Pipeline, die alle Änderungen gleichzeitig bereitstellt. Ein Fehler in einer Komponente zwingt Sie, alle Änderungen rückgängig zu machen, wodurch Ihr Rollback komplex und ineffizient wird.
- Ihr Team schließt die Programmierarbeiten im ersten Sprint ab und beginnt mit dem zweiten Sprint, aber Ihr Plan sieht Tests erst im dritten Sprint vor. Deshalb haben automatisierte Tests Fehler aus dem ersten Sprint aufgedeckt, die behoben werden müssen, bevor mit dem Testen der Ergebnisse von Sprint zwei begonnen werden kann. Der gesamte Release verzögert sich, wodurch der Wert Ihrer automatisierten Tests erheblich verringert wird.
- Ihre automatisierten Regressionstestfälle für die Produktionsversion sind abgeschlossen, aber Sie überwachen den Zustand der Workloads nicht. Da Sie nicht sehen können, ob der Dienst neu gestartet wurde oder nicht, sind Sie sich nicht sicher, ob ein Rollback erforderlich ist oder bereits stattgefunden hat.

Vorteile der Nutzung dieser bewährten Methode: Automatisierte Tests erhöhen die Transparenz Ihres Testprozesses und Ihre Fähigkeit, mehr Funktionen in kürzerer Zeit abzudecken. Durch das Testen und Validieren von Änderungen in der Produktionsphase können Sie Probleme sofort identifizieren. Die Verbesserung der Konsistenz mit automatisierten Testtools ermöglicht eine bessere Fehlererkennung. Durch das automatische Rollback zur vorherigen Version werden die Auswirkungen für Ihre Kunden minimiert. Ein automatisiertes Rollback sorgt letztendlich für mehr Vertrauen in Ihre Bereitstellungsfunktionen, da es die Auswirkungen auf Ihr Unternehmen verringert. Insgesamt verkürzen diese Funktionen die Zeit bis zur Lieferung und stellen gleichzeitig die Qualität sicher.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Automatisieren Sie die Tests von bereitgestellten Umgebungen, um schneller die gewünschten Ergebnisse zu erreichen. Automatisieren Sie den Rollback zu einem bekanntermaßen funktionierenden vorherigen Zustand, wenn die zuvor definierten Ergebnisse nicht erzielt werden. So können Sie die Wiederherstellungszeit minimieren und verringern Fehler, die durch manuelle Prozesse entstehen. Integrieren Sie Testtools in Ihren Pipeline-Workflow, um manuelle Eingaben konsistent zu testen und zu minimieren. Priorisieren Sie die Automatisierung von Testfällen, z. B.

Tests, die die größten Risiken minimieren und die bei jeder Änderung häufig durchgeführt werden müssen. Automatisieren Sie außerdem das Rollback auf Grundlage bestimmter Bedingungen, die in Ihrem Testplan vordefiniert sind.

Implementierungsschritte

1. Richten Sie einen Testlebenszyklus für Ihren Entwicklungslebenszyklus ein, in dem jede Phase des Testprozesses definiert wird. Dies reicht von der Anforderungsplanung über die Testfallentwicklung, die Toolkonfiguration, das automatisierte Testen bis hin zum Abschluss des Testfalls.
   a. Erstellen Sie anhand Ihrer gesamten Teststrategie einen Workload-spezifischen Testansatz.
   b. Ziehen Sie eine Strategie für kontinuierliche Tests während des gesamten Entwicklungszyklus in Erwägung.
2. Wählen Sie in Abhängigkeit von Ihren Geschäftsanforderungen und Pipeline-Investitionen automatisierte Tools für Tests und Rollbacks aus.
3. Entscheiden Sie, welche Testfälle Sie automatisieren möchten und welche manuell durchgeführt werden sollen. Dies kann auf Grundlage des geschäftlichen Nutzens der getesteten Funktion definiert werden. Informieren Sie alle Teammitglieder über diesen Plan und legen Sie fest, wer für die Durchführung manueller Tests verantwortlich ist.
   a. Wenden Sie automatisierte Testfunktionen auf bestimmte Testfälle an, die für die Automatisierung sinnvoll sind, z. B. wiederholbare oder häufig ausgeführte Fälle, Fälle, die sich wiederholende Aufgaben erfordern, oder solche, die für mehrere Konfigurationen erforderlich sind.
   b. Definieren Sie Skripts für die Testautomatisierung sowie die Erfolgskriterien im Automatisierungstool, sodass eine kontinuierliche Workflow-Automatisierung initiiert werden kann, wenn bei bestimmten Fällen Fehler auftreten.
   c. Definieren Sie spezifische Fehlerkriterien für das automatisierte Rollback.
4. Priorisieren Sie die Testautomatisierung, um konsistente Ergebnisse mit einer gründlichen Testfallentwicklung zu erzielen, bei der Komplexität und menschliche Interaktion ein höheres Ausfallrisiko darstellen.
5. Integrieren Sie Ihre automatisierten Test- und Rollback-Tools in Ihre CI/CD-Pipeline.
   a. Entwickeln Sie klare Erfolgskriterien für Ihre Änderungen.
   b. Überwachen und beobachten Sie Ihre Umgebung, um diese Kriterien zu erkennen und Änderungen automatisch rückgängig zu machen, wenn bestimmte Rollback-Kriterien erfüllt werden.

6. Führen Sie verschiedene Arten automatisierter Produktionstests durch, z. B.:
   a. A/B-Tests zur Anzeige von Ergebnissen im Vergleich zur aktuellen Version zwischen zwei Benutzertestgruppen.
   b. Canary-Tests, mit denen Sie Ihre Änderung für eine Untergruppe von Benutzern bereitstellen können, bevor Sie sie für alle freigeben.
   c. Testen mit Feature-Flags, wobei jeweils eine einzelne Funktion der neuen Version außerhalb der Anwendung ein- und ausgeschaltet werden kann, sodass alle neuen Funktionen einzeln validiert werden können.
   d. Regressionstests zur Überprüfung neuer Funktionen mit bestehenden, miteinander verbundenen Komponenten.
7. Überwachen Sie die betrieblichen Aspekte der Anwendung, Transaktionen und Interaktionen mit anderen Anwendungen und Komponenten. Entwickeln Sie Berichte, um den Erfolg von Änderungen nach Workload aufzuzeigen, sodass Sie erkennen können, welche Teile der Automatisierung und des Workflows weiter optimiert werden können.
   a. Entwickeln Sie Testergebnisberichte, anhand derer Sie schnell entscheiden können, ob Rollback-Verfahren eingeleitet werden sollten oder nicht.
   b. Implementieren Sie eine Strategie, die ein automatisiertes Rollback auf Grundlage vordefinierter Fehlerbedingungen ermöglicht, die sich aus einer oder mehreren Ihrer Testmethoden ergeben.
8. Entwickeln Sie Ihre automatisierten Testfälle so, dass sie bei zukünftigen wiederholbaren Änderungen wiederverwendet werden können.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS06-BP01 Einkalkulieren nicht erfolgreicher Änderungen
- OPS06-BP02 Testbereitstellungen

Zugehörige Dokumente:

- AWS Builders' Library | Gewährleistung der Rollback-Sicherheit bei Bereitstellungen
- Erneutes Bereitstellen und Zurücksetzen einer Bereitstellung mit AWS CodeDeploy
- 8 bewährte Methoden beim Automatisieren von Bereitstellungen mit AWS CloudFormation

Zugehörige Beispiele:

- Serverless-Tests für UI mit Selenium, AWS Lambda, AWS Fargate und AWS Developer Tools

Zugehörige Videos:

- re:Invent 2020 | Hands-off: Automating continuous delivery pipelines at Amazon (re:Invent 2020 | Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon)
- re:Invent 2019 | Amazon's approach to high-availability deployment (re:Invent 2019 | Der Amazon-Ansatz für die Hochverfügbarkeitsbereitstellung)

OPS 7. Wie bringen Sie in Erfahrung, ob Sie für die Unterstützung eines Workloads bereit sind?

Bewerten Sie die betriebliche Bereitschaft Ihres Workloads, Prozesse und Verfahren sowie Ihrer Mitarbeiter, damit Sie die betrieblichen Risiken im Zusammenhang mit Ihrer Workload genau kennen.

Bewährte Methoden

- OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter
- OPS07-BP02 Sicherstellen einer konsistenten Prüfung der betrieblichen Bereitschaft
- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen
- OPS07-BP05 Treffen fundierter Entscheidungen für die Bereitstellung von Systemen und Änderungen
- OPS07-BP06 Aktivieren von Supportplänen für Produktions-Workloads

OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter

Nutzen Sie ein System, mit dem Sie validieren können, dass Sie über eine angemessene Anzahl von trainierten Mitarbeitern verfügen, um den Workload zu unterstützen. Sie müssen für die Plattform und die Services, die Ihren Workload ausmachen, trainiert sein. Vermitteln Sie ihnen das für den Betrieb des Workloads erforderliche Wissen. Sie müssen über genügend geschulte Mitarbeiter verfügen, um den normalen Betrieb des Workloads zu unterstützen und auftretende Probleme zu beheben. Sorgen Sie für genügend Mitarbeiter, sodass Sie Bereitschaftsdienste und Urlaubsvertretungen abwechseln können, um Burnouts zu vermeiden.

Gewünschtes Ergebnis:

- Es gibt genügend trainierte Mitarbeiter, um den Workload im Rahmen des Verfügbarkeitszeitraums zu unterstützen.
- Sie trainieren Ihre Mitarbeiter für die Software und Services, die Ihren Workload ausmachen.

Typische Anti-Muster:

- Bereitstellen eines Workloads ohne Teammitglieder, die für den Betrieb der Plattform und der genutzten Services trainiert sind.
- Sie haben nicht genug Mitarbeiter, um wechselnde Bereitschaftsdienste oder Urlaubszeiten abzubilden.

Vorteile der Nutzung dieser bewährten Methode:

- Wenn Sie über qualifizierte Teammitglieder verfügen, können sie Ihren Workload effektiv unterstützen.
- Mit einer ausreichenden Anzahl von Teammitgliedern können Sie den Workload und die Rotation der Bereitschaftsdienste unterstützen und gleichzeitig das Risiko eines Burnouts verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Validieren Sie, ob ausreichend trainierte Mitarbeiter für den Support des Workloads vorhanden sind. Vergewissern Sie sich, dass Sie über genügend Teammitglieder verfügen, um die normalen operativen Aktivitäten, einschließlich Einsatzbereitschaftsdienste, abzudecken.

Kundenbeispiel

AnyCompany Retail sorgt dafür, dass die Teams für den Workload angemessen besetzt und trainiert sind. Es gibt genügend Ingenieure, um wechselnde Bereitschaftsdienste zu unterstützen. Die Mitarbeiter erhalten Training, um die Software und die Workload-Plattform zu nutzen. Sie werden außerdem ermutigt, Zertifizierungen zu erwerben. Es gibt so viele Mitarbeiter, dass Urlaub möglich ist, ohne dass der Workload und die rotierenden Bereitschaftsdienste unterbrochen werden müssen.

Implementierungsschritte

1. Weisen Sie eine ausreichende Anzahl von Mitarbeitern für den Betrieb und den Support Ihres Workloads zu – einschließlich der Bereitschaftsdienste.
2. Trainieren Sie die Mitarbeiter im Umgang mit der Software und den Plattformen, die Ihren Workload ausmachen.
   a. Bei AWS Training und Zertifizierung finden Sie eine Bibliothek mit Kursen zu AWS. Es gibt kostenlose und kostenpflichtige Kurse – online und vor Ort.
   b. AWS hostet Veranstaltungen und Webinare, bei denen Sie von AWS Experten lernen.
3. Bewerten Sie regelmäßig die Größe und die Fähigkeiten des Teams, wenn sich die operativen Bedingungen und der Workload verändern. Passen Sie die Größe und Fähigkeiten des Teams an die operativen Anforderungen an.

Grad des Aufwands für den Implementierungsplan: hoch Das Einstellen und Trainieren eines Teams zur Unterstützung eines Workloads kann einen erheblichen Aufwand darstellen, bietet aber langfristig einen bedeutenden Nutzen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP04 Wissensmanagement - Die Teammitglieder müssen über die notwendigen Informationen verfügen, um den Workload zu betreiben und zu unterstützen. Der Schlüssel dazu ist das Wissensmanagement.

Zugehörige Dokumente:

- AWS-Veranstaltungen und -Webinare
- AWS Training und Zertifizierung

OPS07-BP02 Sicherstellen einer konsistenten Prüfung der betrieblichen Bereitschaft

Verwenden Sie Operational Readiness Reviews (ORRs, Überprüfungen der Einsatzbereitschaft), um zu prüfen, ob Sie Ihren Workload betreiben können. ORR ist ein bei Amazon entwickelter Mechanismus zur Prüfung, ob Teams ihre Workloads in sicherer Weise betreiben können. ORR bezeichnet einen Prüfungs- und Inspektionsprozess anhand einer Checkliste mit Anforderungen. Dies ist ein Self-Service-Vorgang, mit dem Teams ihre Workloads zertifizieren. ORRs beinhalten bewährte Methoden aus unseren jahrelangen Erfahrungen bei der Erstellung von Software.

Eine ORR-Checkliste besteht aus Architekturempfehlungen, betrieblichen Prozessen, Ereignismanagement und Freigabequalität. Unser Correction of Error (CoE)-Prozess ist dafür eine sehr wichtige Grundlage. Ihre eigene Analyse nach einem Vorfall sollte die Weiterentwicklung Ihrer eigenen ORR unterstützen. Bei einer ORR geht es nicht nur um die Umsetzung bewährter Methoden, sondern auch darum, das erneute Auftreten von Ereignissen zu verhindern. Schließlich können auch Sicherheit, Governance und Compliance zu einer ORR gehören.

Führen Sie eine ORR durch, bevor ein Workload zur allgemeinen Verfügbarkeit gestartet wird, und anschließend während des gesamten Softwareentwicklungslebenszyklus. Die Durchführung der ORR vor dem Start verbessert Ihre Fähigkeit zum sicheren Betrieb des Workloads. Führen Sie die ORR auf dem Workload regelmäßig erneut durch, um Abweichungen von bewährten Methoden zu erkennen. Sie können ORR-Checklisten für neue Serviceeinführungen oder für regelmäßige Prüfungen haben. So bleiben Sie hinsichtlich der neuen bewährten Methoden auf dem Laufenden und können Erfahrungen aus Analysen nach Vorfällen einarbeiten. Wenn Sie mit der Cloud immer vertrauter werden, können Sie ORR-Anforderungen als Standardelemente in Ihre Architektur einbauen.

Gewünschtes Ergebnis: Sie haben eine ORR-Checkliste mit bewährten Methoden für Ihre Organisation. ORRs werden vor dem Start von Workloads durchgeführt. ORR werden im Laufe des Workloadlebenszyklus regelmäßig durchgeführt.

Typische Anti-Muster:

- Sie starten einen Workload, ohne zu wissen, ob Sie diesen betreiben können.
- Governance- und Sicherheitsanforderungen gehören nicht zur Zertifizierung eines Workloads für den Start.
- Workloads werden nicht regelmäßig erneut bewertet.
- Workloads werden gestartet, ohne dass erforderliche Verfahren eingerichtet sind.
- Sie erleben die Wiederholung von Ausfällen mit der gleichen Ursache bei mehreren Workloads.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Workloads beinhalten bewährte Methoden für Architektur, Prozess und Management.
- Erkenntnisse werden in Ihren ORR-Prozess integriert.
- Workloads werden gestartet, wenn erforderliche Verfahren eingerichtet sind.
- ORRs werden über den gesamten Softwarelebenszyklus Ihrer Workloads hinweg ausgeführt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Eine ORR ist zweierlei: ein Verfahren und eine Checkliste. Ihr ORR-Verfahren sollte von ihrer Organisation übernommen und von der Unternehmensleitung unterstützt werden. ORRs müssen mindestens durchgeführt werden, bevor Workloads zur allgemeinen Verfügbarkeit gestartet werden. Führen Sie die ORR während des gesamten Lebenszyklus der Softwareentwicklung durch, um ihn bei bewährten Methoden oder neuen Anforderungen aktuell zu halten. Die ORR-Checkliste sollte Konfigurationselemente, Sicherheits- und Governance-Elemente sowie bewährte Methoden aus Ihrer Organisation enthalten. Mit der Zeit können Sie Services wie AWS Config, AWS Security Hubund AWS Control Tower Guardrailsverwenden, um bewährte Methoden aus der ORR in den Integritätsschutz für die automatische Erkennung optimaler Verfahrensweisen aufzunehmen.

Kundenbeispiel

Nach mehreren Produktionsvorfällen entschied sich AnyCompany Retail, einen ORR-Prozess zu implementieren. Das Unternehmen erstellte eine Checkliste mit bewährten Methoden sowie Governance- und Compliance-Anforderungen und Erfahrungen aus früheren Ausfällen. Für neue Workloads werden vor dem Start ORRs durchgeführt. Für jeden Workload wird eine jährliche ORR mit einer Teilmenge der bewährten Methoden durchgeführt, um neue bewährte Methoden und Anforderungen umzusetzen, die der ORR-Checkliste hinzugefügt werden. Mit der Zeit verwendete AnyCompany Retail AWS Config zur Aufdeckung einer bewährter Methoden, was den ORR-Prozess beschleunigte.

Implementierungsschritte

Weitere Informationen zu ORRs finden Sie im Whitepaper zur Überprüfung der betrieblichen Bereitschaft (ORR). Hier finden Sie ausführliche Informationen zur Geschichte des ORR-Verfahrens, zum Aufbau Ihrer eigenen ORR-Praxis und zur Erstellung Ihrer ORR-Checkliste. Die folgenden Schritte sind eine verkürzte Version dieses Dokuments. Für ein vertieftes Verständnis des ORR-Konzepts und der Erstellung eigener ORRs empfehlen wir, das Whitepaper zu lesen.

1. Bringen Sie die wichtigsten Beteiligten zusammen, darunter auch Vertreter aus den Bereichen Sicherheit, Operations und Entwicklung.
2. Lassen Sie alle Beteiligten mindestens eine Anforderung beisteuern. Versuchen Sie für den ersten Durchgang die Anzahl der Elemente auf höchstens dreißig zu beschränken.
   - Anhang B: Beispielfragen für ORRs aus dem ORR-Whitepaper enthält Beispielfragen, die Ihnen beim Start helfen können.

3. Fassen Sie Ihre Anforderungen in einer Tabelle zusammen.
   - Sie können Fokusbereiche in AWS Well-Architected Tool verwenden, um Ihre ORR zu entwickeln und an Ihre Konten und die AWS-Organisation weiterzugeben.
4. Identifizieren Sie einen Workload für die ORR. Ideal ist dafür ein Pre-Launch-Workload oder ein interner Workload.
5. Gehen Sie die ORR-Checkliste durch und notieren Sie alle Erkenntnisse. Diese sind möglicherweise nicht OK, wenn eine Behebung stattfindet. Fügen Sie alle Erkenntnisse ohne Behebung Ihrer Liste hinzu und implementieren Sie die Behebungen vor dem Start.
6. Fügen Sie Ihrer ORR-Checkliste stets weitere bewährte Methoden und Anforderungen hinzu.

Support-Kunden mit Enterprise Support können den Operational Readiness Review Workshop bei ihrem Technical Account Manager anfordern. Der Workshop ist eine interaktive „Working Backwards“- Sitzung zur Entwicklung Ihrer eigenen ORR-Checkliste.

Aufwand für den Implementierungsplan: Hoch. Die Einführung einer ORR-Praxis in Ihrer Organisation erfordert die Unterstützung durch Führungskräfte und alle Beteiligten. Erstellen und aktualisieren Sie die Checkliste mit Beiträgen aus der gesamten Organisation.

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP03 Bewerten der Governance-Anforderungen – Governance-Anforderungen passen perfekt zu einer ORR-Checkliste
- OPS01-BP04 Bewerten der Compliance-Anforderungen – Compliance-Anforderungen werden manchmal auf ORR-Checklisten berücksichtigt. Ansonsten sind sie ein separater Prozess.
- OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten – Die Team-Kapazität ist ein guter Kandidat für eine ORR-Anforderung.
- OPS06-BP01 Einkalkulieren nicht erfolgreicher Änderungen – Vor dem Start Ihres Workloads muss ein Rollback- oder Rollforward-Plan eingerichtet werden.
- OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter – Zur Unterstützung eines Workloads benötigen Sie das erforderliche Personal.
- SEC01-BP03 Identifizieren und Validieren von Kontrollzielen – Sicherheitskontrollziele sind hervorragende ORR-Anforderungen.
- REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten – Notfallwiederherstellungspläne sind eine gute ORR-Anforderung.

- COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen – Kostenmanagementrichtlinien sind für Ihre ORR-Checkliste gut geeignet.

Zugehörige Dokumente:

- AWS Control Tower - Integritätsschutz in AWS Control Tower
- AWS Well-Architected Tool - Fokusbereiche
- Operational Readiness Review Template von Adrian Hornsby
- Whitepaper zur Überprüfung der betrieblichen Bereitschaft (ORR)

Zugehörige Videos:

- AWS -Supports You | Building an Effective Operational Readiness Review (ORR) (AWS -Supports You | Entwickeln einer effektiven Überprüfung der betrieblichen Bereitschaft (ORR))

Zugehörige Beispiele:

- Sample Operational Readiness Review (ORR)-Fokusbereich

Zugehörige Services:

- AWS Config
- AWS Control Tower
- AWS Security Hub
- AWS Well-Architected Tool

OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren

Ein Runbook ist ein dokumentierter Prozess für das Erreichen eines bestimmten Ergebnisses. Runbooks bestehen aus einer Reihe von Schritten, die befolgt werden sollen, um ein Ergebnis zu erzielen. Runbooks werden schon seit den frühen Tagen der Luftfahrt verwendet. Im Cloud-Bereich werden Runbooks verwendet, um die Risiken zu reduzieren und die gewünschten Ergebnisse zu erzielen. In der einfachsten Form ist ein Runbook eine Checkliste für die Durchführung einer Aufgabe.

Runbooks stellen einen kritischen Teil der Ausführung Ihres Workloads dar. Vom Onboarding eines neuen Teammitglieds bis zur Bereitstellung einer Hauptversion – Runbooks stellen kodifizierte

Prozesse dar, mit denen unabhängig von der ausführenden Person konsistente Ergebnisse erzielt werden können. Runbooks sollten an einer zentralen Stelle veröffentlicht werden. Wenn sich der Prozess verändert, sollten sie aktualisiert werden; dies stellt eine zentrale Komponente des Änderungsmanagements dar. Sie sollten auch Anleitungen für Fehlerbehandlung, Tools, Berechtigungen, Ausnahmen und Eskalationen enthalten, falls ein Problem auftritt.

Wenn sich Ihre Organisation entwickelt, sollten Sie mit der Automatisierung von Runbooks beginnen. Sie sollten zunächst Runbooks automatisieren, die kurz sind und häufig verwendet werden. Verwenden Sie Skriptsprachen, um Schritte zu automatisieren oder ihre Ausführung zu vereinfachen. Nach der Automatisierung der ersten Runbooks können Sie komplexere Runbooks automatisieren. Mit der Zeit sollten die meisten Ihrer Runbooks auf die eine oder andere Art automatisiert werden.

Gewünschtes Ergebnis: Ihr Team besitzt eine Sammlung von schrittweisen Anleitungen für die Ausführung von Workload-Aufgaben. Die Runbooks enthalten Angaben zum gewünschten Ergebnis sowie zu notwendigen Tools und Berechtigungen. Darüber hinaus stellen sie Anleitungen für die Fehlerbehandlung bereit. Sie werden an einem zentralen Ort (Versionskontrollsystem) gespeichert und regelmäßig aktualisiert. Ihre Runbooks bieten Ihren Teams beispielsweise die Möglichkeit, AWS Health-Ereignisse für kritische Konten bei Anwendungsalarmen, Betriebsproblemen und geplanten Lebenszyklusereignissen zu überwachen, zu kommunizieren und darauf zu reagieren.

Typische Anti-Muster:

- Verlassen auf das Gedächtnis, um die einzelnen Schritte in einem Prozess durchzuführen.
- Manuelle Bereitstellung von Änderungen ohne Checkliste.
- Verschiedene Teammitglieder führen den gleichen Prozess aus, aber mit unterschiedlichen Schritten oder Ergebnissen.
- Runbooks sind nicht mehr mit Systemänderungen und Automatisierungen synchronisiert.

Vorteile der Nutzung dieser bewährten Methode:

- Reduzierung der Fehlerquoten für manuelle Aufgaben.
- Prozess werden konsistent ausgeführt.
- Neue Teammitglieder können schneller mit der Ausführung von Aufgaben beginnen.
- Runbooks können automatisiert werden, um den Aufwand zu reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Runbooks können verschiedene Formen annehmen, abhängig vom Entwicklungsstand Ihrer Organisation. Sie sollten mindestens aus einem Schritt-für-Schritt-Textdokument bestehen. Das gewünschte Ergebnis sollte klar angegeben werden. Dokumentieren Sie klar die notwendigen Berechtigungen oder Tools. Stellen Sie für den Fall, dass etwas nicht funktioniert, detaillierte Anleitungen für Fehlerbehandlung und Eskalation bereit. Nennen Sie die Person, die für das Runbook verantwortlich ist, und veröffentlichen Sie es an einer zentralen Stelle. Validieren Sie das Runbook, nachdem Sie es dokumentiert haben, indem Sie es von einem Teammitglied ausführen lassen. Mit der weiteren Entwicklung der Verfahren sollten Sie Ihre Runbooks entsprechend Ihrem Prozess für das Änderungsmanagement aktualisieren.

Ihre textbasierten Runbooks sollten mit zunehmender Reife Ihrer Organisation automatisiert werden. Mithilfe von Services wie AWS Systems Manager-Automatisierungen können Sie einfachen Text in Automatisierungen umwandeln, die für Ihr Workload ausgeführt werden können. Diese Automatisierungen können als Reaktion auf Ereignisse ausgeführt werden, was den operativen Aufwand für die Wartung des Workloads reduziert. Die AWSSystems Manager-Automatisierung bietet auch ein visuelles Low-Code-Designerlebnis, mit dem Automatisierungs-Runbooks einfacher erstellt werden können.

Kundenbeispiel

AnyCompany Retail muss während Softwarebereitstellungen die Datenbankschemata aktualisieren. Das Cloud Operations-Team entwickelt gemeinsam mit dem Datenbankverwaltungsteam ein Runbook für die manuelle Bereitstellung dieser Änderungen. In diesem Runbook werden die einzelnen Prozessschritte in Form einer Checkliste aufgelistet. Es enthält für den Fall, dass es ein Problem gibt, auch einen Abschnitt zur Fehlerbehandlung. Das Runbook wird wie die übrigen Runbooks im internen Wiki veröffentlicht. Das Cloud Operations-Team plant, das Runbook in der Zukunft zu automatisieren.

Implementierungsschritte

Wenn Sie noch kein Dokumenten-Repository besitzen, dann ist ein Repository für die Versionskontrolle hervorragend als Grundlage für Ihre Runbook-Bibliothek geeignet. Sie können Ihre Runbooks mithilfe von Markdown erstellen. Wir haben eine Runbook-Beispielvorlage bereitgestellt, die Sie für die Erstellung von Runbooks verwenden können.

```
# Runbook Title
## Runbook Info
```

```
| Runbook ID | Description | Tools Used | Special Permissions | Runbook Author | Last
 Updated | Escalation POC |
|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this runbook for? What is the desired outcome? | Tools | Permissions
 | Your Name | 2022-09-21 | Escalation Name |
## Steps
1. Step one
2. Step two
```

1. Wenn Sie noch kein Dokumentations-Repository oder -Wiki besitzen, sollten Sie in Ihrem Versionskontrollsystem ein neues Versionskontroll-Repository erstellen.
2. Identifizieren Sie einen Prozess, für den es kein Runbook gibt. Ein idealer Prozess hierfür ist ein Prozess, der halbregelmäßig ausgeführt wird, nur wenige Schritte enthält und bei Fehlern nur geringe Auswirkungen hat.
3. Erstellen Sie in Ihrem Dokument-Repository ein neues Markdown-Entwurfsdokument auf der Basis der Vorlage. Füllen Sie den Runbook-Titel und die Pflichtfelder unter Runbook-Informationen aus.
4. Füllen Sie ab dem ersten Schritt den Abschnitt Schritte im Runbook aus.
5. Geben Sie das Runbook einem Teammitglied. Lassen Sie das Teammitglied das Runbook ausführen, um die Schritte zu validieren. Aktualisieren Sie das Runbook, wenn etwas fehlt oder unklar ist.
6. Veröffentlichen Sie das Runbook in Ihrem internen Dokumentationsspeicher. Informieren Sie Ihr Team und die übrigen Stakeholder über das Runbook, nachdem es veröffentlicht wurde.
7. Mit der Zeit entsteht dadurch eine Bibliothek von Runbooks. Beginnen Sie mit der Automatisierung von Runbooks, wenn diese Bibliothek wächst.

Aufwand für den Implementierungsplan: niedrig. Eine schrittweise Anleitung in Textform ist der Mindeststandard für ein Runbook. Die Automatisierung von Runbooks kann den Implementierungsaufwand erhöhen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen
- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen

- OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung
- OPS11-BP04 Wissensmanagement

Zugehörige Dokumente:

- AWS Well-Architected Framework: Konzepte: Runbook-Entwicklung
- Operative Kompetenz durch automatisierte Playbooks und Runbooks
- AWS Systems Manager: Arbeiten mit Runbooks
- Migrations-Playbook für große AWS-Migrationen – Aufgabe 4: Verbesserung Ihrer Migrations-Runbooks
- Verwendung von AWS Systems Manager-Automation-Runbooks zur Lösung operativer Aufgaben

Zugehörige Videos:

- AWS re:Invent 2019: DIY-Leitfaden für Runbooks, Vorfallberichte und Vorfallreaktion
- Automatisierung von IT-Abläufen in AWS | Amazon Web Services
- Integration von Skripts in AWS Systems Manager

Zugehörige Beispiele:

- Well-Architected Labs: Automatisieren von Vorgängen mit Playbooks und Runbooks
- AWS-Blogbeitrag: Aufbau einer Cloud-Automatisierungspraxis für Operational Excellence: Bewährte Methoden von AWS Managed Services
- AWS Systems Manager: Exemplarische Vorgehensweisen zur Automatisierung
- AWS Systems Manager: Runbook für die Wiederherstellung eines Root-Volumes anhand des letzten Snapshots
- Entwicklung eines Runbooks für Vorfallreaktionen in AWS mit Jupyter Notebooks und CloudTrail Lake
- Gitlab – Runbooks
- Rubix – eine Python-Bibliothek für die Erstellung von Runbooks in Jupyter Notebooks
- Verwendung von Document Builder für die Erstellung angepasster Runbooks

Zugehörige Services:

- AWS Systems Manager-Automatisierung

OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen

Playbooks sind schrittweise Anleitungen zur Untersuchung von Vorfällen. Wenn Vorfälle auftreten, werden Playbooks verwendet, um sie zu untersuchen, die Auswirkungen abzuschätzen und Ursachen zu identifizieren. Playbooks werden für verschiedene Szenarien eingesetzt, von fehlgeschlagenen Bereitstellungen bis hin zu Sicherheitsvorfällen. In vielen Fällen identifizieren Playbooks Ursachen, die dann mithilfe eines Runbooks beseitigt werden. Playbooks sind eine sehr wichtige Komponente der Vorfallreaktionspläne Ihrer Organisation.

Ein gutes Playbook weist einige zentrale Merkmale auf. Es leitet den Nutzer Schritt für Schritt durch den Erkennungsprozess. Welche Schritte sollten befolgt werden, um einen Vorfall zu diagnostizieren? Legen Sie im Playbook klar fest, ob bestimmte Tools oder erhöhte Berechtigungen benötigt werden. Ein wichtiger Teil ist ein Kommunikationsplan, um alle Stakeholder über den Status der Untersuchung zu informieren. Für den Fall, dass die eigentliche Ursache des Vorfalls nicht identifiziert werden kann, sollte das Playbook einen Eskalationsplan enthalten. Wenn die Ursache identifiziert wurde, sollte das Playbook auf ein Runbook verweisen, das beschreibt, wie die Ursache zu beheben ist. Playbooks sollten zentral gespeichert und regelmäßig gepflegt werden. Wenn Playbooks für bestimmte Warnungmeldungen verwendet werden, sollte Ihr Team in den Warnungmeldungen auf das Playbook verwiesen werden.

Im Zuge der Weiterentwicklung Ihrer Organisation sollten Sie Ihre Playbooks automatisieren. Beginnen Sie mit Playbooks für Vorfälle mit geringem Risikograd. Automatisieren Sie die Erkennungsschritte mit Skripts. Stellen Sie sicher, dass Sie über begleitende Runbooks für die Behebung typischer Ursachen verfügen.

Gewünschtes Ergebnis: Ihre Organisation verfügt über Playbooks für typische Vorfälle. Die Playbooks werden an einem zentralen Ort gespeichert und sind für Ihre Teammitglieder verfügbar. Playbooks werden häufig aktualisiert. Für alle bekannten Ursachen werden begleitende Runbooks erstellt.

Typische Anti-Muster:

- Es gibt kein Standardverfahren für die Untersuchung von Vorfällen.
- Teammitglieder verlassen sich auf ihr Gedächtnis oder allgemein vorhandenes Wissen, um eine fehlgeschlagene Bereitstellung zu beheben.
- Neue Teammitglieder lernen die Untersuchung von Problemen durch Ausprobieren.

- Es werden keine bewährten Methoden für die Untersuchung von Problemen zwischen Teams ausgetauscht.

Vorteile der Nutzung dieser bewährten Methode:

- Playbooks verbessern Ihre Fähigkeit zum Umgang mit Vorfällen.
- Verschiedene Teammitglieder können dasselbe Playbook verwenden, um Ursachen in konsistenter Weise zu ermitteln.
- Für bekannte Ursachen können Runbooks entwickelt werden, um die Wiederherstellungszeit zu verkürzen.
- Mit Playbooks können Teammitglieder schneller Beiträge leisten.
- Mit wiederholbaren Playbooks können Teams ihre Prozesse skalieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wie Sie Ihre Playbooks aufbauen und verwenden, hängt vom Reifegrad Ihrer Organisation ab. Wenn Sie noch neu in der Cloud sind, erstellen Sie Playbooks in Textform in einem zentralen Dokumenten-Repository. Wenn sich Ihre Organisation weiterentwickelt, können Playbooks mit Skriptsprachen wie Python teilweise automatisiert werden. Diese Skripts können zur Beschleunigung der Untersuchung in einem Jupyter Notebook ausgeführt werden. Fortgeschrittene Organisationen haben vollständig automatisierte Playbooks für häufig auftretende Probleme, die dann mit Runbooks automatisch behoben werden.

Beginnen Sie die Arbeit an Ihren Playbooks mit der Auflistung typischer Vorfälle bei Ihren Workloads. Wählen Sie Playbooks zunächst für Vorfälle mit geringem Risiko, bei denen die Ursache eingegrenzt werden kann. Wenn Sie über Playbooks für einfachere Szenarien verfügen, gehen Sie zu Szenarien mit höheren Risiken oder zu Szenarien über, bei denen die Ursache nicht vollständig klar ist.

Ihre textbasierten Runbooks sollten mit zunehmender Reife Ihrer Organisation automatisiert werden. Mithilfe von Services wie AWS Systems Manager-Automatisierungen kann einfacher Text in Automatisierungen umgewandelt werden. Diese Automatisierungen können dann für Ihren Workload ausgeführt werden, um die Untersuchungen zu beschleunigen. Sie können als Reaktion auf Ereignisse aktiviert werden, wodurch sich der durchschnittliche Zeitaufwand für die Untersuchung und Behebung von Vorfällen reduziert.

Kunden können AWS Systems Manager Incident Manager verwenden, um auf Vorfälle zu reagieren. Dieser Service bietet eine einzige Oberfläche für die Untersuchung von Vorfällen, die Information der Stakeholder über Untersuchung und Abhilfemaßnahmen und die Zusammenarbeit während des gesamten Vorgangs. Er verwendet AWS Systems Manager-Automatisierungen zur Beschleunigung von Untersuchung und Wiederherstellung.

Kundenbeispiel

Ein Produktionsvorfall hat Auswirkungen auf AnyCompany Retail. Der zuständige Techniker untersuchte das Problem mithilfe eines Playbooks. Im Zuge der einzelnen Schritte wurden anhand des aktuellen Playbooks die Beteiligten identifiziert. Der Techniker ermittelte einen Race-Zustand in einem Backend-Service als Ursache für den Vorfall. Mithilfe eines Runbooks startete er den Service neu und brachte AnyCompany Retail so wieder online.

Implementierungsschritte

Wenn Sie noch kein Dokumenten-Repository besitzen, dann sollten Sie ein Versionskontroll-Repository für Ihre Runbook-Bibliothek erstellen. Sie können Ihre Playbooks mit Markdown erstellen, das mit den meisten Playbook-Automatisierungssystemen kompatibel ist. Wenn Sie neu beginnen, verwenden Sie die folgende Beispielvorlage für ein Playbook.

```
# Playbook Title
## Playbook Info
| Playbook ID | Description | Tools Used | Special Permissions | Playbook Author | Last
 Updated | Escalation POC | Stakeholders | Communication Plan |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| RUN001 | What is this playbook for? What incident is it used for? | Tools |
 Permissions | Your Name | 2022-09-21 | Escalation Name | Stakeholder Name | How will
 updates be communicated during the investigation? |
## Steps
1. Step one
2. Step two
```

1. Wenn Sie noch kein Dokumenten-Repository oder -Wiki besitzen, sollten Sie in Ihrem Versionskontrollsystem ein neues Versionskontroll-Repository für Ihre Playbooks erstellen.
2. Identifizieren Sie ein typisches Problem, das eine Untersuchung erfordert. Dies sollte ein Szenario sein, bei dem die Ursache auf wenige Probleme eingegrenzt werden kann und das Risiko insgesamt niedrig ist.
3. Füllen Sie mithilfe der Markdown-Vorlage den Abschnitt Playbook-Name und die Felder unter Playbook-Informationen aus.

4. Geben Sie die Schritte zur Fehlerbehebung ein. Benennen Sie die zu treffenden Maßnahmen bzw. die zu untersuchenden Bereiche so klar wie möglich.
5. Geben Sie das Playbook einem Teammitglied zur Prüfung. Wenn darin etwas fehlt oder nicht klar ist, aktualisieren Sie das Playbook.
6. Veröffentlichen Sie Ihr Playbook in Ihrem Dokumenten-Repository und informieren Sie Ihr Team und alle Stakeholder darüber.
7. Diese Playbook-Bibliothek wächst mit der Zeit an. Sobald Sie mehrere Playbooks haben, beginnen Sie mithilfe von Tools wie AWS Systems Manager Automations mit ihrer Automatisierung.

Aufwand für den Implementierungsplan: niedrig. Ihre Playbooks sollten an einem zentralen Ort gespeicherte Textdokumente sein. Ausgereiftere Organisationen gehen zu automatisierten Playbooks über.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer
- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren
- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen
- OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung
- OPS11-BP04 Wissensmanagement

Zugehörige Dokumente:

- AWS Well-Architected Framework: Konzepte: Playbook-Entwicklung
- Operative Kompetenz durch automatisierte Playbooks und Runbooks
- AWS Systems Manager: Arbeiten mit Runbooks
- Verwendung von AWS Systems Manager-Automation-Runbooks zur Lösung operativer Aufgaben

Zugehörige Videos:

- AWS re:Invent 2019: DIY-Leitfaden für Runbooks, Vorfallberichte und Vorfallreaktion (SEC318-R1)
- AWS Systems Manager Incident Manager – AWS Virtuelle Workshops

- Integration von Skripts in AWS Systems Manager

Zugehörige Beispiele:

- AWS Customer Playbook Framework
- AWS Systems Manager: Exemplarische Vorgehensweisen zur Automatisierung
- Entwicklung eines Runbooks für Vorfallreaktionen in AWS mit Jupyter Notebooks und CloudTrail Lake
- Rubix – Eine Python-Bibliothek für die Erstellung von Runbooks in Jupyter Notebooks
- Verwendung von Document Builder für die Erstellung angepasster Runbooks
- Well-Architected Labs: Automatisieren von Vorgängen mit Playbooks und Runbooks
- Well-Architected Labs: Playbook für Vorfallreaktion mit Jupyter

Zugehörige Services:

- AWS Systems Manager-Automatisierung
- AWS Systems Manager Incident Manager

OPS07-BP05 Treffen fundierter Entscheidungen für die Bereitstellung von Systemen und Änderungen

Nutzen Sie Prozesse für erfolgreiche und erfolglose Änderungen an Ihrem Workload. Eine Pre-mortem-Übung ist eine Übung, bei der ein Team einen Fehler simuliert, um Strategien zur Behebung zu entwickeln. Beugen Sie wo möglich Fehlern vor und stellen Sie entsprechende Abläufe auf. Bewerten Sie den Nutzen und die Risiken der Bereitstellung von Änderungen an Ihrem Workload. Überprüfen Sie, ob alle Änderungen mit der Governance übereinstimmen.

Gewünschtes Ergebnis:

- Sie treffen bei der Bereitstellung von Änderungen an Ihrem Workload fundierte Entscheidungen.
- Änderungen entsprechen der Governance.

Typische Anti-Muster:

- Sie stellen eine Änderung an Ihrem Workload bereit, ohne einen Prozess für die Verarbeitung einer fehlgeschlagenen Bereitstellung zu haben.

- Sie nehmen Änderungen an Ihrer Produktionsumgebung vor, die nicht mit den Governance-Anforderungen vereinbar sind.
- Sie stellen eine neue Version Ihres Workloads bereit, ohne eine Baseline für die Ressourcenauslastung zu erstellen.

Vorteile der Nutzung dieser bewährten Methode:

- Sie sind auf fehlgeschlagene Änderungen an Ihrem Workload vorbereitet.
- Änderungen an Ihrem Workload sind konform mit den Governance-Richtlinien.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Verwenden Sie Pre-Mortem-Übungen, um Prozesse für fehlgeschlagene Änderungen zu entwickeln. Dokumentieren Sie Ihre Prozesse für fehlgeschlagene Änderungen. Stellen Sie sicher, dass alle Änderungen mit der Governance übereinstimmen. Evaluieren Sie die Vorteile und Risiken der Bereitstellung von Änderungen an Ihrem Workload.

Kundenbeispiel

AnyCompany Retail führt regelmäßig Pre-Mortems durch, um die Prozesse für fehlgeschlagene Änderungen zu validieren. Die Prozesse werden in einem gemeinsamen Wiki dokumentiert und regelmäßig aktualisiert. Alle Änderungen entsprechen den Governance-Anforderungen.

Implementierungsschritte

1. Treffen Sie fundierte Entscheidungen, wenn Sie Änderungen an Ihrem Workload bereitstellen. Legen Sie Kriterien für eine erfolgreiche Bereitstellung fest und überprüfen Sie diese. Entwickeln Sie Szenarien oder Kriterien, die ein Rollback einer Änderung auslösen würden. Wägen Sie den Nutzen der Bereitstellung von Änderungen gegen die Risiken einer fehlgeschlagenen Änderung ab.
2. Überprüfen Sie, ob alle Änderungen mit den Governance-Richtlinien übereinstimmen.
3. Planen Sie anhand von Pre-Mortems fehlgeschlagene Änderungen und dokumentieren Sie Strategien zur Schadensbegrenzung. Führen Sie eine Table-Top-Übung durch, um eine fehlgeschlagene Änderung zu modellieren und Rollback-Verfahren zu validieren.

Grad des Aufwands für den Implementierungsplan: moderat. Die Einführung von Pre-Mortems erfordert die Koordination und den Einsatz aller Stakeholder in Ihrer gesamten Organisation

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP03 Bewerten der Governance-Anforderungen - Governance-Anforderungen sind ein Schlüssel bei der Entscheidung zur Bereitstellung einer Änderung.
- OPS06-BP01 Einkalkulieren nicht erfolgreicher Änderungen - Erstellen Sie Pläne zur Eindämmung einer fehlgeschlagenen Bereitstellung und verwenden Sie Pre-Mortems, um diese zu validieren.
- OPS06-BP02 Testbereitstellungen - Jede Softwareänderung sollte vor der Bereitstellung ordnungsgemäß getestet werden, um Fehler in der Produktion zu reduzieren.
- OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter - Ausreichend trainierte Mitarbeiter zur Unterstützung des Workloads sind unerlässlich, um eine fundierte Entscheidung über die Bereitstellung einer Systemänderung zu treffen.

Zugehörige Dokumente:

- Amazon Web Services: Risiko und Compliance
- AWS-Modell der geteilten Verantwortung
- Governance in the AWS Cloud: The Right Balance Between Agility and Safety (Governance in der AWS Cloud: Das richtige Gleichgewicht zwischen Agilität und Sicherheit)

OPS07-BP06 Aktivieren von Supportplänen für Produktions-Workloads

Aktivieren Sie Support für sämtliche Software und Services, auf denen Ihr Produktions-Workload basiert. Wählen Sie ein geeignetes Support-Level für Ihre Servicelevel-Anforderungen in der Produktion. Supportpläne für diese Abhängigkeiten sind wichtig für den Fall von Serviceunterbrechungen oder Softwareproblemen. Dokumentieren Sie Supportpläne sowie die Verfahren zur Anfrage nach Support bei allen Service- und Software-Anbietern. Implementieren Sie Mechanismen zur Prüfung, ob Support-Kontaktpunkte stets aktuell sind.

Gewünschtes Ergebnis:

- Implementieren Sie Supportpläne für Software und Services, auf denen Ihre Workloads basieren.
- Wählen Sie einen geeigneten Supportplan auf der Grundlage Ihrer Service-Level-Anforderungen.

- Dokumentieren Sie die Supportpläne, die Supportlevels und die Vorgehensweise bei Supportanfragen.

Typische Anti-Muster:

- Sie haben keinen Supportplan für einen kritischen Softwareanbieter. Dies beeinflusst Ihren Workload, und Sie haben keine Möglichkeit, schnell einen Fix oder rechtzeitige Updates von dem Anbieter zu erhalten.
- Ein Entwickler, der der primäre Ansprechpartner bei einem Softwareanbieter war, hat das Unternehmen verlassen. Sie können den Support des Anbieters nicht direkt erreichen. Sie müssen Zeit aufwenden, um sich durch generische Kontaktsysteme zu arbeiten, was die Reaktionszeiten verlängert.
- Bei einem Softwareanbieter ereignet sich ein Produktionsausfall. Es gibt keine Dokumentation dazu, wie ein Supportfall einzureichen ist.

Vorteile der Nutzung dieser bewährten Methode:

- Mit dem richtigen Supportlevel können Sie schnell eine Reaktion erhalten, die dem Service-Level entspricht.
- Als Kunde mit Support stehen Ihnen bei Produktionsproblemen Eskalationsmöglichkeiten zur Verfügung.
- Software- und Serviceanbieter können Ihnen bei Vorfällen Unterstützung bei der Fehlerbehebung bieten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Aktivieren Sie Support für sämtliche Software- und Service-Anbieter, von denen Ihr Produktions-Workload abhängt. Richten Sie geeignete Supportpläne ein, um Service-Level einhalten zu können. Für AWS-Kunden bedeutet dies die Aktivierung von AWS Business Support oder einer höheren Stufe für alle Konten mit Produktions-Workloads. Treffen Sie sich regelmäßig mit Supportanbietern, um Neues zu Supportangeboten, -prozessen und -ansprechpartnern zu erfahren. Dokumentieren Sie das Supportverfahren bei Software- und Serviceanbietern, einschließlich der Eskalationsmöglichkeiten bei Ausfällen. Implementieren Sie Mechanismen, um die Supportkontakte stets auf aktuellem Stand zu halten.

Kundenbeispiel

Bei AnyCompany Retail gibt es für alle kommerziellen Software- und Service-Abhängigkeiten Supportpläne. Beispielsweise hat das Unternehmen AWS Enterprise Support für alle Konten mit Produktions-Workloads. Jeder Entwickler kann bei einem Problem einen Supportfall auslösen. Es gibt eine Wiki-Seite mit Informationen zum Verfahren bei Supportanfragen, zu den Ansprechpartnern und zu bewährten Methoden dafür.

Implementierungsschritte

1. Arbeiten Sie mit den Beteiligten in Ihrer Organisation, um Software- und Serviceanbieter zu identifizieren, von denen Ihr Workload abhängt. Dokumentieren Sie diese Abhängigkeiten.
2. Legen Sie die Service-Level-Anforderungen für Ihren Workload fest. Wählen Sie einen Supportplan, der dazu passt.
3. Richten Sie für kommerzielle Software und Services einen Supportplan bei den Anbietern ein.
   a. Ein Abonnement von AWS Business Support oder höher für alle Produktionskonten bietet schnellere Reaktionszeiten von AWS -Support und wird dringend empfohlen. Wenn Sie keinen Premium-Support haben, benötigen Sie einen Aktionsplan für den Umgang mit Problemen, bei denen Hilfe von AWS -Support erforderlich ist. AWS -Support stellt Ihnen verschiedenste Tools und Technologien, Fachpersonal und Programme zur Verfügung, die Sie proaktiv bei der Performance-Optimierung, Kostensenkung und schnelleren Entwicklung neuer Innovationen unterstützen. AWS Business Support bietet zusätzliche Vorteile, darunter den Zugriff auf AWS Trusted Advisor und das AWS Personal Health Dashboard sowie kürzere Reaktionszeiten.
4. Dokumentieren Sie den Supportplan in Ihrem Wissensmanagement-Tool. Berücksichtigen Sie dabei, wie eine Supportanfrage durchgeführt wird, wer in einem solchen Fall zu benachrichtigen ist und wie Vorfälle eskaliert werden können. Ein Wiki ist ein gutes Hilfsmittel, das allen Beteiligten ermöglicht, erforderliche Aktualisierungen der Dokumentation vorzunehmen, wenn ihnen Änderungen bei Supportprozessen oder Ansprechpartnern bekannt werden.

Grad des Aufwands für den Implementierungsplan: niedrig. Die meisten Software- und Serviceanbieter bieten Opt-in-Supportpläne an. Durch die Dokumentation und die Weitergabe bewährter Supportmethoden in Ihrem Wissensmanagementsystem können Sie sicherstellen, dass Ihr Team weiß, was bei einem Produktionsproblem zu tun ist.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP02 Prozesse und Verfahren haben feste Besitzer

Zugehörige Dokumente:

- AWS -Support Plans (AWS -Support-Pläne)

Zugehörige Services:

- AWS Business Support
- AWS Enterprise Support

Betrieb

Fragen

- OPS 8. Wie nutzen Sie die Überwachbarkeit von Workloads in Ihrer Organisation?
- OPS 9. Wie können Sie den Zustand Ihrer Operationen beurteilen?
- OPS 10. Wie bewältigen Sie Workload- und operationsspezifische Ereignisse?

OPS 8. Wie nutzen Sie die Überwachbarkeit von Workloads in Ihrer Organisation?

Sorgen Sie für einen optimalen Zustand des Workloads, indem Sie die Überwachbarkeit nutzen. Nutzen Sie relevante Metriken, Protokolle und Traces, um sich einen umfassenden Überblick über die Leistung Ihres Workloads zu verschaffen und Probleme effizient zu beheben.

Bewährte Methoden

- OPS08-BP01 Analysieren von Workload-Metriken
- OPS08-BP02 Analysieren von Workload-Protokollen
- OPS08-BP03 Analysieren von Workload-Traces
- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen
- OPS08-BP05 Erstellen von Dashboards

OPS08-BP01 Analysieren von Workload-Metriken

Analysieren Sie nach der Implementierung der Anwendungstelemetrie regelmäßig die gesammelten Metriken. Latenz, Anfragen, Fehler und Kapazität (oder Kontingente) liefern zwar Erkenntnisse zur

Systemleistung, es ist jedoch wichtig, die Überprüfung der Metriken zu Geschäftsergebnissen zu priorisieren. Dadurch wird sichergestellt, dass Sie datengestützte Entscheidungen treffen, die auf Ihre Geschäftsziele abgestimmt sind.

Gewünschtes Ergebnis: Präzise Erkenntnisse zur Workload-Leistung, die als Grundlage für datengestützte Entscheidungen dienen und die Abstimmung mit den Geschäftszielen sicherstellen.

Typische Anti-Muster:

- Isolierte Analyse von Metriken, ohne deren Auswirkungen auf die Geschäftsergebnisse zu berücksichtigen.
- Übermäßiges Vertrauen in technische Metriken, während Geschäftsmetriken ignoriert werden.
- Seltene Überprüfung von Metriken, Entscheidungsmöglichkeiten in Echtzeit werden verpasst.

Vorteile der Nutzung dieser bewährten Methode:

- Verbessertes Verständnis des Zusammenhangs zwischen technischer Leistung und Geschäftsergebnissen.
- Verbesserter Entscheidungsprozess auf der Grundlage von Echtzeitdaten.
- Proaktive Identifizierung und Minderung von Problemen, bevor sie sich auf die Geschäftsergebnisse auswirken.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nutzen Sie Tools wie Amazon CloudWatch zur Durchführung metrischer Analysen. Sie können AWS-Services wie AWS Cost Anomaly Detection und Amazon DevOps Guru zur Erkennung von Anomalien verwenden, insbesondere wenn statische Schwellenwerte unbekannt sind oder wenn Verhaltensmuster besser für die Erkennung von Anomalien geeignet sind.

Implementierungsschritte

1. Analysieren und überprüfen Sie Metriken: Überprüfen Sie regelmäßig Ihre Workload-Metriken und werten Sie sie aus.
   a. Priorisieren Sie Metriken zu Geschäftsergebnissen gegenüber rein technischen.
   b. Machen Sie sich mit der Bedeutung von Spitzen, Rückgängen oder Mustern in Ihren Daten vertraut.

2. Nutzen Sie Amazon CloudWatch: Verwenden Sie Amazon CloudWatch für eine zentrale Ansicht und detaillierte Analysen.
   a. Konfigurieren Sie CloudWatch-Dashboards, um Ihre Metriken zu visualisieren und sie im Zeitverlauf zu vergleichen.
   b. Nutzen Sie Perzentile in CloudWatch, um einen klaren Überblick über die metrische Verteilung zu erhalten, der Ihnen helfen kann, SLAs zu verstehen und einzelne Ausreißer nachzuvollziehen.
   c. Richten Sie AWS Cost Anomaly Detection ein, um ungewöhnliche Muster zu identifizieren, ohne sich auf statische Schwellenwerte zu verlassen.
   d. Implementieren Sie die kontenübergreifende Beobachtbarkeit mit CloudWatch, um Anwendungen zu überwachen und Fehler zu beheben, die mehrere Konten innerhalb einer Region betreffen.
   e. Nutzen Sie CloudWatch Metric Insights, um metrische Daten über Konten und Regionen hinweg abzufragen und zu analysieren und Trends und Anomalien zu identifizieren.
   f. Wenden Sie CloudWatch Metric Math an, um Ihre Metriken zu transformieren, zu aggregieren oder Berechnungen für den Erhalt tieferer Einblicke durchzuführen.
3. Machen Sie Gebrauch von Amazon DevOps Guru: Integrieren Sie Amazon DevOps Guru wegen seiner Machine Learning-gestützten Anomalieerkennung, mit der Sie frühzeitig Anzeichen von Betriebsproblemen Ihrer Serverless-Anwendungen erkennen und diese beheben können, bevor sie sich auf Ihre Kunden auswirken.
4. Optimieren Sie auf der Grundlage von Erkenntnissen: Treffen Sie fundierte Entscheidungen auf der Grundlage Ihrer Metrikanalyse, um Ihre Workloads anzupassen und zu verbessern.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie

Zugehörige Dokumente:

- The Wheel Blog - Emphasizing the importance of continually reviewing metrics (Die Bedeutung der kontinuierlichen Überprüfung von Metriken)

- Percentile are important (Perzentile sind wichtig)
- Using AWS Cost Anomaly Detection (Verwendung von AWS Cost Anomaly Detection)
- CloudWatch cross-account observability (kontenübergreifende Beobachtbarkeit mit CloudWatch)
- Query your metrics with CloudWatch Metrics Insights (Metrikabfrage mit CloudWatch Metrics Insights)

Zugehörige Videos:

- Enable Cross-Account Observability in Amazon CloudWatch (Kontenübergreifende Beobachtbarkeit in Amazon CloudWatch aktivieren)
- Introduction to Amazon DevOps Guru (Einführung in Amazon DevOps Guru)
- Continuously Analyze Metrics using AWS Cost Anomaly Detection (Fortlaufende Metrikanalyse mit AWS Cost Anomaly Detection)

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- Gaining operation insights with AIOps using Amazon DevOps Guru (Operative Erkenntnisse gewinnen mit AIOps und Amazon DevOps Guru)

OPS08-BP02 Analysieren von Workload-Protokollen

Die regelmäßige Analyse von Workload-Protokollen ist unerlässlich, um ein tieferes Verständnis der operativen Aspekte Ihrer Anwendung zu erlangen. Durch effizientes Durchsuchen, Visualisieren und Interpretieren von Protokolldaten können Sie die Leistung und Sicherheit von Anwendungen kontinuierlich optimieren.

Gewünschtes Ergebnis: Umfassende Erkenntnisse zum Anwendungsverhalten und zu Operationen, die aus einer gründlichen Protokollanalyse gewonnen wurden und für eine proaktive Problemerkennung und -behebung sorgen.

Typische Anti-Muster:

- Die Analyse von Protokollen vernachlässigen, bis ein kritisches Problem auftritt.
- Die Suite verfügbarer Tools für die Protokollanalyse nicht nutzen und wichtige Erkenntnisse verpassen.

- Alleiniges Vertrauen auf die manuelle Überprüfung von Protokollen, ohne Automatisierungs- und Abfragefunktionen zu nutzen.

Vorteile der Nutzung dieser bewährten Methode:

- Proaktive Identifizierung von operativen Engpässen, Sicherheitsbedrohungen und anderen potenziellen Problemen.
- Effiziente Nutzung von Protokolldaten für die kontinuierliche Anwendungsoptimierung.
- Verbessertes Verständnis des Anwendungsverhaltens, Unterstützung beim Debuggen und bei der Problembehandlung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Amazon CloudWatch Logs ist ein leistungsstarkes Tool für die Protokollanalyse. Integrierte Features wie CloudWatch Logs Insights und Contributor Insights sorgen für eine intuitive und effiziente Ableitung aussagekräftiger Informationen aus Protokollen.

Implementierungsschritte

1. Einrichtung von CloudWatch Logs: Konfigurieren Sie Anwendungen und Services so, dass Protokolle an CloudWatch Logs gesendet werden.
2. Verwendung der Erkennung von Protokollanomalien: Verwenden Sie die Amazon CloudWatch Logs-Anomalieerkennung, um ungewöhnliche Protokollmuster automatisch zu identifizieren und Warnmeldungen zu erhalten. Mit diesem Tool können Sie Anomalien in Ihren Protokollen proaktiv verwalten und potenzielle Probleme frühzeitig erkennen.
3. Einrichten von CloudWatch Logs-Insights: Verwenden Sie CloudWatch Logs-Insights, um Ihre Protokolldaten interaktiv zu durchsuchen und zu analysieren.
   a. Erstellen Sie Abfragen, um Muster zu extrahieren, Protokolldaten zu visualisieren und umsetzbare Erkenntnisse abzuleiten.
   b. Verwenden Sie die Musteranalyse für CloudWatch Logs-Erkenntnisse, um häufige Protokollmuster zu analysieren und zu visualisieren. Dieses Feature hilft Ihnen, allgemeine Betriebstrends und potenzielle Ausreißer in Ihren Protokolldaten nachzuvollziehen.
   c. Verwenden Sie CloudWatch Logs compare (diff), um eine Differenzanalyse zwischen verschiedenen Zeiträumen oder Protokollgruppen vorzunehmen. Verwenden Sie diese

Funktion, um Änderungen zu lokalisieren und deren Auswirkungen auf die Leistung oder das Verhalten Ihres Systems zu bewerten.

4. Überwachen Sie Protokolle in Echtzeit mit Live Tail: Verwenden Sie Amazon CloudWatch Logs Live Tail, um Protokolldaten in Echtzeit anzuzeigen. Sie können die Betriebsaktivitäten Ihrer Anwendung in Echtzeit aktiv überwachen, um sich einen unmittelbaren Einblick in die Systemleistung und potenzielle Probleme zu verschaffen.
5. Nutzung von Contributor Insights: Verwenden Sie CloudWatch Contributor Insights, um Top-Talker in Dimensionen mit hoher Kardinalität wie IP-Adressen oder Benutzeragenten zu identifizieren.
6. Implementieren von CloudWatch Logs-Metrikfiltern: Konfigurieren Sie CloudWatch Logs-Metrikfilter, um Protokolldaten in umsetzbare Metriken umzuwandeln. Auf diese Weise können Sie Alarme einstellen oder Muster näher analysieren.
7. Implementieren von kontoübergreifender CloudWatch-Beobachtbarkeit: Überwachen Sie Anwendungen, die sich über mehrere Konten innerhalb einer Region erstrecken, und beheben Sie Fehler.
8. Regelmäßige Überprüfung und Verfeinerung: Überprüfen Sie regelmäßig Ihre Protokollanalysestrategien, um alle relevanten Informationen zu erfassen und die Anwendungsleistung kontinuierlich zu optimieren.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS08-BP01 Analysieren von Workload-Metriken

Zugehörige Dokumente:

- Analysieren von Protokolldaten mit CloudWatch Logs Insights
- Nutzung von CloudWatch Contributor Insights
- Erstellen und Verwalten von CloudWatch Logs-Metrikfiltern

Zugehörige Videos:

- Analysieren von Protokolldaten mit CloudWatch Logs Insights
- Mit CloudWatch Contributor Insights Daten mit hoher Kardinalität analysieren

Zugehörige Beispiele:

- CloudWatch Logs-Beispielabfragen
- Workshop zur Beobachtbarkeit

OPS08-BP03 Analysieren von Workload-Traces

Die Analyse von Trace-Daten ist entscheidend, wenn es darum geht, einen umfassenden Überblick über den Betriebsverlauf einer Anwendung zu erhalten. Durch die Visualisierung und das Verständnis der Interaktionen zwischen verschiedenen Komponenten können die Leistung optimiert, Engpässe identifiziert und das Benutzererlebnis verbessert werden.

Gewünschtes Ergebnis: Sie verschaffen sich einen klaren Überblick über die verteilten Abläufe Ihrer Anwendung und erzielen dadurch eine schnellere Problemlösung und ein verbessertes Benutzererlebnis.

Typische Anti-Muster:

- Trace-Daten werden übersehen und man verlässt sich ausschließlich auf Protokolle und Metriken.
- Trace-Daten werden nicht mit zugehörigen Protokollen in Zusammenhang gebracht.
- Aus Traces abgeleitete Metriken wie Latenz und Fehlerraten werden ignoriert.

Vorteile der Nutzung dieser bewährten Methode:

- Sie verbessern die Fehlersuche und reduzieren die durchschnittliche Zeit für die Behebung (Mean Time to Resolution, MTTR).
- Sie gewinnen Erkenntnisse über Abhängigkeiten und deren Auswirkungen.
- Sie können Leistungsprobleme rasch identifizieren und beheben.
- Sie nutzen von aus Trace abgeleitete Metriken für fundierte Entscheidungen.
- Sie erzielen ein besseres Benutzererlebnis durch optimierte Komponenteninteraktionen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

AWS X-Ray bietet eine umfassende Suite für die Analyse von Trace-Daten, die einen ganzheitlichen Überblick über Serviceinteraktionen, die Überwachung von Benutzeraktivitäten und die Erkennung von Leistungsproblemen bietet. Features wie ServiceLens, X-Ray Insights, X-Ray Analytics und Amazon DevOps Guru erhöhen die Tiefe verwertbarer Erkenntnisse, die aus Trace-Daten gewonnen werden.

Implementierungsschritte

Die folgenden Schritte bieten einen strukturierten Ansatz zur effektiven Implementierung der Trace-Datenanalyse mithilfe von AWS-Services:

1. Integrate AWS X-Ray: Stellen Sie sicher, dass in Ihre Anwendungen X-Ray integriert ist, um Trace-Daten zu erfassen.
2. Analysieren Sie X-Ray Metriken: Untersuchen Sie anhand von X-Ray Traces abgeleitete Metriken wie Latenz, Anforderungsraten, Fehlerraten und Reaktionszeitverteilungen, und verwenden Sie die Service Map, um den Zustand der Anwendung zu überwachen.
3. Verwendung von ServiceLens: Nutzen Sie die ServiceLens-Map für eine verbesserte Beobachtbarkeit Ihrer Services und Anwendungen. Dies ermöglicht eine integrierte Anzeige von Traces, Metriken, Protokollen, Alarmen und anderen Statusinformationen.
4. Aktivieren Sie X-Ray Insights:
   a. Aktivieren Sie X-Ray Insights für die automatische Erkennung von Anomalien in Traces.
   b. Untersuchen Sie Erkenntnisse, um Muster zu identifizieren und die Ursachen zu ermitteln, z. B. erhöhte Fehlerraten oder Latenzen.
   c. Eine chronologische Analyse der erkannten Probleme finden Sie in der Insights-Timeline.
5. Verwendung von X-Ray Analytics: X-Ray Analytics ermöglicht es Ihnen, Daten gründlich zu untersuchen, Muster zu lokalisieren und Erkenntnisse zu gewinnen.
6. Verwendung von Gruppen in X-Ray: Erstellen Sie Gruppen in X-Ray, um Traces nach Kriterien wie hoher Latenz zu filtern und so eine gezieltere Analyse zu ermöglichen.
7. Integration von Amazon DevOps Guru: Setzen Sie Amazon DevOps Guru ein, um von Machine-Learning-Modellen zu profitieren, die betriebliche Anomalien in Traces lokalisieren.
8. Verwendung von CloudWatch Synthetics: Verwenden Sie CloudWatch Synthetics, um Canaries für die kontinuierliche Überwachung Ihrer Endpunkte und Workflows zu erstellen. Sie können diese Canaries in X-Ray integrieren, um Trace-Daten für eine eingehende Analyse der getesteten Anwendungen bereitzustellen.

9. Verwendung von Real User Monitoring (RUM): Mit AWS X-Ray und CloudWatch RUM können Sie den Anforderungspfad analysieren und debuggen, angefangen bei den Endbenutzern Ihrer Anwendung bis hin zu nachgelagerten AWS-verwalteten Services. Auf diese Weise können Sie Latenztrends und Fehler identifizieren, die sich auf Ihre Endbenutzer auswirken.
10. Korrelieren mit Protokollen: Korrelieren Sie Trace-Daten mit zugehörigen Protokollen in der X-Ray-Trace-Ansicht, um sich einen detaillierten Überblick über das Anwendungsverhalten zu verschaffen. Auf diese Weise können Sie Protokollereignisse anzeigen, die direkt mit verfolgten Transaktionen verknüpft sind.
11. Implementieren von kontoübergreifender CloudWatch-Beobachtbarkeit: Überwachen Sie Anwendungen, die sich über mehrere Konten innerhalb einer Region erstrecken, und beheben Sie Fehler.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS08-BP01 Analysieren von Workload-Metriken
- OPS08-BP02 Analysieren von Workload-Protokollen

Zugehörige Dokumente:

- Verwenden von ServiceLens zur Überwachung des Zustands Ihrer Anwendungen
- Erkunden von Trace-Daten mit X-Ray Analytics
- Mit X-Ray Insights Anomalien in Traces erkennen
- Fortlaufende Überwachung mit CloudWatch Synthetics

Zugehörige Videos:

- Analysieren und Debuggen von Anwendungen mithilfe von Amazon CloudWatch Synthetics und AWS X-Ray
- Nutzung von AWS X-Ray Insights

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- Implementieren von X-Ray mit AWS Lambda
- Vorlagen für CloudWatch Synthetics Canary

OPS08-BP04 Erstellen umsetzbarer Warnmeldungen

Es ist entscheidend, Abweichungen im Verhalten Ihrer Anwendung umgehend zu erkennen und darauf zu reagieren. Besonders wichtig ist es, zu erkennen, wann die auf den wichtigsten Leistungsindikatoren (KPIs) basierenden Ergebnisse gefährdet sind oder unerwartete Anomalien auftreten. Wenn Sie Warnmeldungen auf KPIs basieren, stellen Sie dadurch sicher, dass die Signale, die Sie erhalten, direkt mit geschäftlichen oder betrieblichen Auswirkungen verknüpft sind. Der Ansatz mit umsetzbaren Warnmeldungen fördert proaktive Reaktionen und trägt zur Aufrechterhaltung der Systemleistung und Zuverlässigkeit bei.

Gewünschtes Ergebnis: Sie erhalten rechtzeitig relevante und umsetzbare Warnmeldungen, um potenzielle Probleme schnell zu erkennen und zu beheben, insbesondere wenn die KPI-Ergebnisse gefährdet sind.

Typische Anti-Muster:

- Es werden zu viele unkritische Warnmeldungen eingerichtet, was zu einer Übermüdung durch Warnmeldungen führt.
- Warnmeldungen werden nicht anhand von KPIs priorisiert, was es schwierig macht, die geschäftlichen Auswirkungen von Problemen zu verstehen.
- Die eigentlichen Ursachen werden vernachlässigt, was zu wiederholten Warnmeldungen für dasselbe Problem führt.

Vorteile der Nutzung dieser bewährten Methode:

- Geringere Ermüdung durch Warnmeldungen durch Fokussierung auf umsetzbare und relevante Warnmeldungen.
- Verbesserte Systemverfügbarkeit und -zuverlässigkeit durch proaktive Problemerkennung und -behebung.
- Verbesserte Teamzusammenarbeit und schnellere Problemlösung durch die Integration in übliche Warnmeldungs- und Kommunikationstools.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Um einen effektiven Warnmechanismus zu schaffen, ist es wichtig, Metriken, Protokolle und Trace-Daten zu verwenden, die darauf hinweisen, wenn auf KPIs basierende Ergebnisse gefährdet sind oder Anomalien erkannt werden.

Implementierungsschritte

1. Ermitteln von Key Performance Indicators (KPIs): Identifizieren Sie die KPIs Ihrer Anwendung. Warnmeldungen sollten mit diesen KPIs verknüpft werden, damit sie die Auswirkungen auf das Unternehmen genau widerspiegeln.
2. Implementierung der Erkennung von Anomalien:
   - Verwendung der Amazon CloudWatch-Anomalieerkennung: Richten Sie die Amazon CloudWatch-Anomalieerkennung ein, um ungewöhnliche Muster automatisch zu erkennen, damit Warnmeldungen nur für echte Anomalien generieren werden.
   - Nutzung von AWS X-Ray Insights:
     a. Richten Sie X-Ray Insights ein, um Anomalien in Trace-Daten zu erkennen.
     b. Konfigurieren Sie Benachrichtigungen für X-Ray Insights, um bei erkannten Problemen Warnmeldungen zu erhalten.
   - Integration mit Amazon DevOps Guru:
     a. Nutzung von Amazon DevOps Guru für die Machine-Learning-Fähigkeiten bei der Erkennung betrieblicher Anomalien anhand vorhandener Daten.
     b. Navigieren Sie zu den Benachrichtigungseinstellungen unter DevOps Guru, um Anomaliewarnmeldungen einzurichten.
3. Implementieren umsetzbarer Warnmeldungen: Entwerfen Sie Warnmeldungen, die angemessene Informationen für sofortige Maßnahmen enthalten.
   1. Überwachen Sie AWS Health-Ereignisse mithilfe von Amazon EventBridge-Regeln oder integrieren Sie sie programmgesteuert in die AWS Health API, um Aktionen zu automatisieren, wenn Sie AWS Health-Ereignisse erhalten. Dies können allgemeine Aktionen sein, z. B. das Senden aller geplanten Lebenszyklus-Ereignisnachrichten an eine Chat-Oberfläche, oder spezifische Aktionen, wie das Initiieren eines Workflows in einem IT-Servicemanagement-Tool.
4. Reduzieren der Warnmeldungsmüdigkeit: Minimieren Sie unkritische Warnmeldungen. Wenn Teams mit zahllosen unbedeutenden Warnmeldungen überfordert werden, können sie den

Überblick über kritische Probleme verlieren, was die Gesamteffektivität des Warnmechanismus beeinträchtigt.

5. Einrichten von zusammengesetzten Alarmen: Verwenden Sie zusammengesetzte Amazon CloudWatch-Alarme, um mehrere Alarme zu kombinieren.
6. Integrieren von Warnmeldungs-Tools: Integrieren Sie Tools wie Ops Genie und PagerDuty.
7. Nutzung von Amazon Q Developer in chat applications: Integrieren Sie Amazon Q Developer in chat applications, um Warnmeldungen an Amazon Chime, Microsoft Teams und Slack weiterzuleiten.
8. Warnmeldung basierend auf Protokollen: Verwenden Sie Protokoll-Metrikfilter in CloudWatch, um Alarme basierend auf bestimmten Protokollereignissen zu erstellen.
9. Überprüfen und iterieren: Überprüfen und Sie die Warnkonfigurationen regelmäßig und passen Sie sie an.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung
- OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie
- OPS04-BP05 Implementieren der verteilten Nachverfolgung
- OPS08-BP01 Analysieren von Workload-Metriken
- OPS08-BP02 Analysieren von Workload-Protokollen
- OPS08-BP03 Analysieren von Workload-Traces

Zugehörige Dokumente:

- Verwendung von Amazon CloudWatch-Alarmen
- Erstellung eines zusammengesetzten Alarms
- Erstellung eines CloudWatch-Alarms auf der Grundlage der Anomalieerkennung

- DevOps Guru-Benachrichtigungen
- X-Ray Insights – Benachrichtigungen
- Überwachung, Betrieb und Fehlerbehebung Ihrer AWS-Ressourcen mit interaktiven ChatOps
- Amazon CloudWatch-Integrationsleitfaden | PagerDuty
- Integration von OpsGenie mit Amazon CloudWatch

Zugehörige Videos:

- Erstellung zusammengesetzter Alarme in Amazon CloudWatch
- Amazon Q Developer in chat applications Übersicht
- AWS On Air ft. Veränderliche Befehle in Amazon Q Developer in chat applications

Zugehörige Beispiele:

- Alarme, Vorfallmanagement und Problembehebung in der Cloud mit Amazon CloudWatch
- Tutorial: Erstellen einer Amazon EventBridge-Regel, die Benachrichtigungen an Amazon Q Developer in chat applications sendet
- Workshop zur Beobachtbarkeit

OPS08-BP05 Erstellen von Dashboards

Dashboards sind die anwenderorientierte Sicht auf die Telemetriedaten Ihrer Workloads. Sie stellen zwar eine wichtige visuelle Schnittstelle dar, sollten aber nicht als Ersatz, sondern als Ergänzung für Warnmechanismen dienen. Wenn sie sorgfältig zusammengestellt werden, liefern sie nicht nur schnelle Erkenntnisse zum Status und zur Leistung des Systems, sondern bieten Stakeholdern auch Echtzeitinformationen über Geschäftsergebnisse und die Auswirkungen von Problemen.

Gewünschtes Ergebnis:

Klare, umsetzbare Erkenntnisse zur System- und Geschäftsstabilität mithilfe visueller Darstellungen.

Typische Anti-Muster:

- Überkomplizierte Dashboards mit zu vielen Metriken.
- Sich auf Dashboards verlassen, ohne Warnmeldungen zur Erkennung von Anomalien zu nutzen.
- Fehlende Aktualisierung der Dashboards im Laufe des Workload-Fortschritts.

Vorteile dieser bewährten Methode:

- Sofortiger Einblick in wichtige Systemmetriken und KPIs.
- Verbesserte Kommunikation und mehr Verständnis unter den Interessengruppen.
- Rasche Erkenntnisse zu den Auswirkungen operativer Probleme.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Geschäftsorientierte Dashboards

Dashboards, die auf Geschäfts-KPIs zugeschnitten sind, sprechen ein breiteres Spektrum von Stakeholdern an. Auch wenn diese Personen vielleicht nicht an Systemmetriken interessiert sind, haben sie dennoch großes Interesse daran, die geschäftlichen Auswirkungen dieser Zahlen zu verstehen. Ein geschäftsorientiertes Dashboard stellt sicher, dass alle technischen und betrieblichen Metriken, die überwacht und analysiert werden, auf die übergeordneten Geschäftsziele ausgerichtet sind. Diese Ausrichtung sorgt für Klarheit und stellt sicher, dass alle gleich darüber informiert sind, was wichtig ist und was nicht. Darüber hinaus sind Dashboards, die Geschäfts-KPIs hervorheben, in der Regel leichter umzusetzen. Sie bieten Stakeholdern die Möglichkeit, in kürzester Zeit den Status der Abläufe, die Bereiche, die Aufmerksamkeit erfordern, und die potenziellen Auswirkungen auf die Geschäftsergebnisse zu verstehen.

Vor diesem Hintergrund sollten Sie bei der Erstellung Ihrer Dashboards sicherstellen, dass ein Gleichgewicht zwischen technischen Metriken und Geschäfts-KPIs besteht. Beide sind wichtig, richten sich aber an unterschiedliche Zielgruppen. Idealerweise sollten Sie über Dashboards verfügen, die einen ganzheitlichen Überblick über den Status und die Leistung des Systems bieten und gleichzeitig wichtige Geschäftsergebnisse und deren Auswirkungen hervorheben.

Amazon CloudWatch-Dashboards sind anpassbare Startseiten in der CloudWatch-Konsole zur Überwachung Ihrer Ressourcen in einer einzigen Ansicht, auch wenn sie über verschiedene AWS-Regionen und Konten verteilt sind.

Implementierungsschritte

1. Erstellen eines einfachen Dashboards: Erstellen Sie ein neues Dashboard in CloudWatch und geben Sie ihm einen aussagekräftigen Namen.
2. Verwenden von Markdown-Widgets: Bevor Sie sich mit den Metriken befassen, verwenden Sie Markdown-Widgets, um oben in Ihrem Dashboard inhaltlichen Kontext hinzuzufügen. Dieser

sollte den Inhalt des Dashboards beschreiben und angeben, welche Bedeutung den dargestellten Metriken zukommt. Er kann auch Links zu anderen Dashboards und Tools zur Fehlerbehebung enthalten.

3. Erstellen von Dashboard-Variablen: Integrieren Sie gegebenenfalls Dashboard-Variablen, um dynamische und flexible Dashboard-Ansichten zu ermöglichen.
4. Erstellung von Metrik-Widgets: Fügen Sie Metrik-Widgets hinzu, um verschiedene Metriken zu visualisieren, die Ihre Anwendung ausgibt, und passen Sie diese Widgets so an, dass sie den Systemstatus und die Geschäftsergebnisse effektiv darstellen.
5. Protokollieren von Insights-Abfragen: Nutzen Sie CloudWatch Log Insights, um aus Ihren Protokollen umsetzbare Metriken abzuleiten und diese Erkenntnisse in Ihrem Dashboard anzuzeigen.
6. Einrichten von Alarmen: Integrieren Sie CloudWatch-Alarme in Ihr Dashboard, um sich einen schnellen Überblick über alle Metriken zu verschaffen, die ihre Schwellenwerte überschreiten.
7. Verwenden von Contributor Insights: Integrieren Sie CloudWatch Contributor Insights, um Felder mit hoher Kardinalität zu analysieren und die besten Mitarbeiter Ihrer Ressource zu identifizieren.
8. Entwerfen benutzerdefinierter Widgets: Erwägen Sie die Erstellung von benutzerdefinierten Widgets für spezielle Anforderungen, die von Standard-Widgets nicht erfüllt werden. Diese können Daten aus verschiedenen Quellen abrufen oder sie auf einzigartige Weise darstellen.
9. Verwendung von AWS Health Dashboard: Verwenden Sie AWS Health Dashboard, um detailliertere Einblicke in den Zustand Ihres Kontos, in Ereignisse und bevorstehende Änderungen zu erhalten, die sich auf Ihre Services und Ressourcen auswirken könnten. Sie können auch eine zentrale Übersicht über Statusereignisse in AWS Organizations abrufen oder Ihre eigenen benutzerdefinierten Dashboards erstellen (weitere Informationen finden Sie unter „Verwandte Beispiele“).
10. Iteration und Anpassung: Im Laufe der Entwicklung Ihrer Anwendung sollten Sie Ihr Dashboard regelmäßig überprüfen, um sicherzustellen, dass es weiterhin relevant ist.

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS08-BP01 Analysieren von Workload-Metriken
- OPS08-BP02 Analysieren von Workload-Protokollen
- OPS08-BP03 Analysieren von Workload-Traces

- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen

Zugehörige Dokumente:

- Erstellung von Dashboards für operative Sichtbarkeit
- Verwendung von Amazon CloudWatch-Dashboards

Zugehörige Videos:

- Erstellung von konto- und regionenübergreifenden CloudWatch-Dashboards
- AWS re:Invent 2021 – Mehr Unternehmenstransparenz mit geschäftsorientierten AWS Cloud-Dashboards

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- Anwendungsüberwachung mit Amazon CloudWatch
- Intelligence Dashboards und Erkenntnisse zu AWS Health-Ereignissen
- Visualisieren Sie AWS Health-Ereignisse mit Amazon Managed Grafana

OPS 9. Wie können Sie den Zustand Ihrer Operationen beurteilen?

Definieren, erfassen und analysieren Sie Metriken für Operationen, um einen Einblick in Ereignisse rund um Ihre operativen Abläufe zu erhalten. Dies ist wichtig, damit Sie bei Bedarf entsprechende Maßnahmen ergreifen können.

Bewährte Methoden

- OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken
- OPS09-BP02 Kommunizieren von Status und Trends zur Sicherung der operativen Transparenz
- OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen

OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken

Ermitteln Sie Ziele und KPIs in Ihrem Unternehmen, die operativen Erfolg definieren, und legen Sie Metriken fest, die diese Werte widerspiegeln. Legen Sie Baselines als Bezugspunkt fest und

bewerten Sie diese regelmäßig neu. Entwickeln Sie Mechanismen, um diese Metriken von Teams zur Bewertung zu erfassen.

Gewünschtes Ergebnis:

- Die Ziele und KPIs für die Operations-Teams der Organisation wurden veröffentlicht und geteilt.
- Metriken, die diese KPIs widerspiegeln, wurden festgelegt. Mögliche Beispiele:
  - Tiefe der Ticket-Queue oder Durchschnittsalter der Tickets
  - Anzahl der Tickets, gruppiert nach Art des Problems
  - Aufgewendete Zeit für die Bearbeitung von Problemen mit oder ohne standardisierte Betriebsverfahren (SOP)
  - Zeit, die zur Wiederherstellung nach einem fehlgeschlagenen Code-Push aufgewendet wurde
  - Anrufaufkommen

Typische Anti-Muster:

- Bereitstellungsfristen werden nicht eingehalten, weil Entwickler mit der Lösung von Problemen beauftragt werden. Entwicklerteams fordern mehr Personal, können aber nicht einschätzen, wie viele Personen benötigt werden, da der Zeitaufwand nicht gemessen werden kann.
- Für die Abwicklung von Kundenanrufen wurde ein Problem-Desk Stufe 1 eingerichtet. Im Laufe der Zeit kamen weitere Workloads hinzu, aber dem Problem-Desk Stufe 1 wurde kein zusätzliches Personal zugewiesen. Die Kundenzufriedenheit leidet, da immer mehr Anrufe nötig sind und Probleme länger ungelöst bleiben. Das Management sieht diese Anzeichen jedoch nicht und ermöglicht keine Gegenmaßnahmen.
- Ein problematischer Workload wurde zur Bearbeitung an ein separates Operations-Team übergeben. Im Gegensatz zu anderen Workloads wurde dieser neue Workload nicht mit ordnungsgemäßer Dokumentation und Runbooks geliefert. Daher verbringen Teams mehr Zeit damit, Fehler zu suchen und zu beheben. Es gibt jedoch keine Metriken, die dies dokumentieren, was die Rechenschaftspflicht erschwert.

Vorteile der Nutzung dieser bewährten Methode: Während die Workload-Überwachung den Status unserer Anwendungen und Services anzeigt, liefert die Überwachung von Operations-Teams den Verantwortlichen Erkenntnisse hinsichtlich Veränderungen bei den Nutzern dieser Workloads, wie z. B. sich ändernde Geschäftsanforderungen. Messen Sie die Effektivität dieser Teams und bewerten Sie sie im Hinblick auf Ihre operativen Ziele, indem Sie Metriken erstellen, die den operativen

Status widerspiegeln können. Anhand von Metriken können Supportprobleme aufgezeigt oder Abweichungen von einem angestrebten Servicelevel erkannt werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Planen Sie Meetings mit der Geschäftsleitung und den Stakeholdern, um die allgemeinen Ziele des Services festzulegen. Ermitteln Sie, worin die Aufgaben der verschiedenen Operations-Teams bestehen sollten und mit welchen Herausforderungen sie beauftragt werden könnten. Führen Sie anhand dieser Daten ein Brainstorming der wichtigsten Leistungsindikatoren (KPIs) durch, die diese operativen Ziele widerspiegeln könnten. Dies können Faktoren wie Kundenzufriedenheit, Zeitspanne zwischen Entwurf und Bereitstellung von Funktionen, durchschnittlicher Zeitaufwand für die Problemlösung und andere sein.

Identifizieren Sie anhand der KPIs die Metriken und Datenquellen, die diese Ziele am besten widerspiegeln könnten. Kundenzufriedenheit kann eine Kombination aus verschiedenen Metriken wie Warte- oder Reaktionszeiten bei Anrufen, Zufriedenheitswerte und Art der dargelegten Probleme sein. Die Bereitstellungszeiten können die Summe des Zeitaufwands sein, der für Tests und Bereitstellungen benötigt wird, zuzüglich aller Korrekturen nach der Bereitstellung, die hinzugefügt werden mussten. Statistiken, aus denen hervorgeht, wie viel Zeit für verschiedene Arten von Problemen aufgewendet wurde (oder wie viele dieser Probleme auftraten), können Aufschluss darüber geben, wo gezielte Anstrengungen erforderlich sind.

Ressourcen

Zugehörige Dokumente:

- QuickSight - Using KPIs (Amazon QuickSight – Verwendung von KPIs)
- Amazon CloudWatch - Using Metrics (Amazon CloudWach – Verwendung von Metriken)
- Erstellung von Dashboards
- Wie Sie mit dem KPI-Dashboard Ihre KPIs zur Kostenoptimierung nachverfolgen

OPS09-BP02 Kommunizieren von Status und Trends zur Sicherung der operativen Transparenz

Wenn Sie in Erfahrung bringen wollen, wann Ergebnisse gefährdet sein könnten, ob zusätzliche Workloads unterstützt werden können oder nicht oder welche Auswirkungen Änderungen auf Ihre Teams hatten, müssen Sie unbedingt den Status Ihrer Betriebsabläufe und deren Trendrichtung kennen. Bei Betriebsereignissen können Statusseiten, auf denen Benutzer und Operations-

Teams Informationen abrufen können, den Druck auf die Kommunikationskanäle verringern und Informationen proaktiv verbreiten.

Gewünschtes Ergebnis:

- Betriebsleiter erhalten auf einen Blick Erkenntnisse darüber, welches Anrufvolumen ihre Teams bewältigen müssen und welche Maßnahmen möglicherweise im Gange sind, z. B. Bereitstellungen.
- Wenn Auswirkungen auf den normalen Betrieb auftreten, werden Warnmeldungen an Stakeholder und Nutzergemeinschaften versendet.
- Unternehmensleitung und Stakeholder können als Reaktion auf eine Warnung oder Auswirkung eine Statusseite aufrufen und Informationen zu einem betrieblichen Ereignis abrufen, z. B. Kontaktstellen, Ticketinformationen und erwartete Wiederherstellungszeiten.
- Führungskräften und anderen Stakeholdern werden Berichte zur Verfügung gestellt, damit sie über Betriebsstatistiken wie das Anrufvolumen über einen bestimmten Zeitraum, Nutzerzufriedenheitswerte, Anzahl ausstehender Tickets und deren Alter informiert sind.

Typische Anti-Muster:

- Ein Workload fällt aus und ein Dienst wird nicht verfügbar. Das Anrufvolumen steigt, da Benutzer wissen möchten, was vor sich geht. Manager erhöhen dieses Volumen, da sie nachfragen, wer an dem Problem arbeitet. Verschiedene Operations-Teams bemühen sich doppelt, Untersuchungen durchzuführen.
- Der Wunsch nach neuen Funktionen führt dazu, dass mehrere Mitarbeiter umpositioniert werden, um an einem speziellen technischen Vorhaben zu arbeiten. Dadurch entstehende Lücken werden nicht aufgefüllt und die Problemlösungszeiten steigen. Diese Informationen werden nicht erfasst, und erst nach mehreren Wochen und viel negativem Feedback unzufriedener Nutzer wird die Unternehmensleitung auf das Problem aufmerksam.

Vorteile der Nutzung dieser bewährten Methode: Bei betrieblichen Ereignissen, die das Geschäft beeinträchtigen, wird manchmal viel Zeit und Energie damit verschwendet, Informationen von verschiedenen Teams abzufragen, die versuchen, die Situation zu verstehen. Durch die Einrichtung und Verbreitung von Statusseiten und Dashboards können Stakeholder rasch Informationen darüber abrufen, ob ein Problem festgestellt wurde oder nicht, wer mit der Lösung des Problems beschäftigt ist oder wann mit einer Rückkehr zum normalen Betrieb zu rechnen ist. Dadurch müssen die Teammitglieder nicht zu viel Zeit damit verbringen, anderen den Status mitzuteilen und haben mehr Zeit, Probleme zu lösen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Erstellen Sie Dashboards, die die aktuellen Schlüsselmetriken für Ihre Operations-Teams anzeigen, und machen Sie sie sowohl für die Betriebsleitung als auch für das Management leicht zugänglich.

Erstellen Sie Statusseiten, die schnell aktualisiert werden können, um zu zeigen, wann sich ein Vorfall oder ein Ereignis abspielt, wer dafür verantwortlich ist und wer die Reaktion darauf koordiniert. Kommunizieren Sie auf dieser Seite alle Schritte oder Problemumgehungen, die Benutzer in Betracht ziehen sollten, und machen Sie sie für alle Beteiligten verfügbar. Bitten Sie Benutzer, zuerst diese Seite zu überprüfen, wenn sie mit einem unbekannten Problem konfrontiert werden.

Erfassen Sie Daten und stellen Sie Berichte bereit, die den Zustand der Betriebsabläufe im Zeitverlauf aufzeigen, und verteilen Sie diese an Führungskräfte und Entscheidungsträger, um die Arbeit des Betriebs sowie die Herausforderungen und Bedürfnisse zu veranschaulichen.

Teilen Sie die Metriken und Berichte, die die Ziele und KPIs am besten widerspiegeln, mit den Teams, und zeigen Sie ihnen, wo sie besonders deutlich einen Wandel vorangetrieben haben. Nehmen Sie sich Zeit für diese Aktivitäten, um den Abläufen innerhalb und zwischen Teams mehr Bedeutung beizumessen.

Ressourcen

Zugehörige Dokumente:

- Measure Progress (Fortschritt messen)
- Building Dashboards for Operational Visibility (Erstellung von Dashboards für operative Sichtbarkeit)

Zugehörige Lösungen:

- Datenoperationen

OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen

Durch die Bereitstellung von Zeit und Ressourcen für die Überprüfung des Betriebsstatus wird sichergestellt, dass die Betreuung der täglichen Geschäftstätigkeit weiterhin Priorität hat. Bringen Sie Betriebsleiter und Stakeholder an einen Tisch, um regelmäßig Metriken zu überprüfen, Ziele und Vorgaben zu bestätigen oder zu ändern und Verbesserungen zu priorisieren.

Gewünschtes Ergebnis:

- Betriebsleiter und Mitarbeiter treffen sich regelmäßig, um die Metriken für einen bestimmten Berichtszeitraum zu überprüfen. Herausforderungen werden kommuniziert, Erfolge gefeiert und gewonnene Erkenntnisse geteilt.
- Stakeholder und Unternehmensleiter werden regelmäßig über den Stand der laufenden Operationen informiert und um ihre Meinung gebeten, was Ziele, KPIs und zukünftige Initiativen angeht. Kompromisse zwischen Servicebereitstellung, Betrieb und Wartung werden erörtert und in Zusammenhang gebracht.

Typische Anti-Muster:

- Ein neues Produkt wird auf den Markt gebracht, aber die Operations-Teams der Stufe 1 und 2 sind nicht ausreichend geschult, um Support zu leisten, oder bräuchten zusätzliches Personal. Metriken, die den Anstieg der Bearbeitungsdauer von Tickets und der Anzahl der Vorfälle belegen, werden von Führungskräften nicht berücksichtigt. Erst Wochen später werden Maßnahmen ergriffen, weil die Zahl der Abonnements zu sinken beginnt, da unzufriedene Benutzer die Plattform verlassen.
- Ein manuelles Verfahren zur Durchführung von Wartungsarbeiten an einem Workload gibt es schon lange. Der Wunsch nach Automatisierung war zwar vorhanden, hatte aber angesichts der geringen Bedeutung des Systems nur geringe Priorität. Im Laufe der Zeit hat das System jedoch an Bedeutung gewonnen, und heute nehmen diese manuellen Prozesse einen Großteil der Betriebszeit in Anspruch. Es sind keine Ressourcen für die Bereitstellung von mehr Tools für den Betrieb vorgesehen, was zu einer Überlastung der Mitarbeiter führt, wenn der Workload zunimmt. Die Unternehmensleitung wird sich der Probleme bewusst, als sie erfährt, dass Mitarbeiter zu anderen Wettbewerbern wechseln.

Vorteile der Nutzung dieser bewährten Methode: In einigen Unternehmen kann es zu einer Herausforderung werden, für die Servicebereitstellung die gleiche Zeit und Aufmerksamkeit aufzuwenden, die neuen Produkten oder Angeboten entgegengebracht wird. Wenn dies zutrifft, kann der Geschäftsbereich darunter leiden und das erwartete Serviceniveau verschlechtert sich nach und nach. Dies liegt daran, dass sich der Betrieb nicht mit dem wachsenden Geschäft ändert und weiterentwickelt, wodurch er bald ins Hintertreffen gerät. Ohne eine regelmäßige Überprüfung der Erkenntnisse, die Operations erfasst, wird das Risiko für das Unternehmen möglicherweise erst sichtbar, wenn es zu spät ist. Wenn jedoch sowohl dem Betriebspersonal als auch den Führungskräften Zeit für die Überprüfung von Metriken und Verfahren eingeräumt wird, bleibt die entscheidende Rolle, die der Betrieb spielt, sichtbar und Risiken können erkannt werden, lange

bevor sie ein kritisches Niveau erreichen. Operations-Teams erhalten einen besseren Überblick über bevorstehende Geschäftsänderungen und Initiativen, sodass proaktive Maßnahmen ergriffen werden können. Wenn Führungskräfte die Gelegenheit haben, die Betriebsmetriken zu prüfen, erkennen sie, welche Rolle diese Teams für die Kundenzufriedenheit spielen –sowohl intern als auch extern. So können sie Operations die Möglichkeit geben, Entscheidungen im Hinblick auf Prioritäten besser abzuwägen oder sicherzustellen, dass die Teams über die Zeit und die Ressourcen verfügen, um mit neuen Geschäfts- und Workload-Initiativen zu wachsen und sich weiterzuentwickeln.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nehmen Sie sich Zeit, um die Betriebsmetriken gemeinsam mit Stakeholdern und Operations-Teams zu überprüfen und die Berichtsdaten zu lesen. Stellen Sie diese Berichte in den Kontext der Ziele und Vorgaben der Organisation, um festzustellen, ob sie erreicht werden. Identifizieren Sie Unklarheiten, bei denen die Ziele nicht eindeutig sind oder wo Konflikte bestehen zwischen dem, was verlangt wird, und dem, was gegeben wird.

Identifizieren Sie, wo Zeit, Mitarbeiter und Tools zu Betriebsergebnissen beitragen können. Ermitteln Sie, auf welche KPIs sich dies auswirken würde und welche Erfolgsziele verfolgt werden sollten. Greifen Sie Ihre Überlegungen regelmäßig wieder auf, um sicherzustellen, dass der Betrieb über ausreichende Ressourcen verfügt, um den Geschäftsbereich zu unterstützen.

Ressourcen

Zugehörige Dokumente:

- Amazon Athena
- Amazon CloudWatch metrics and dimensions reference (Referenzinformationen zu Metriken und Dimensionen von Amazon CloudWatch)
- Amazon QuickSight
- AWS Glue
- AWS Glue Data Catalog
- Collect metrics and logs from Amazon EC2 instances and on-premises servers with the Amazon CloudWatch Agent (Erfassen von Metriken und Protokollen aus Amazon EC2-Instances und On-Premises-Servern mit dem Amazon CloudWatch Agent)
- Using Amazon CloudWatch metrics (Verwenden von Amazon CloudWatch-Metriken)

OPS 10. Wie bewältigen Sie Workload- und operationsspezifische Ereignisse?

Erarbeiten und prüfen Sie Verfahren für die Reaktion auf Ereignisse, um Beeinträchtigungen für Ihren Workload zu minimieren.

Bewährte Methoden

- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen
- OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung
- OPS10-BP03 Priorisieren von betrieblichen Ereignissen auf Basis der Auswirkung auf das Unternehmen
- OPS10-BP04 Definieren von Eskalationspfaden
- OPS10-BP05 Definieren eines Kundenkommunikationsplan für Ereignisse, die sich auf den Service auswirken
- OPS10-BP06 Bekanntgeben des Status über Dashboards
- OPS10-BP07 Automatisieren von Reaktionen auf Ereignisse

OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen

Die Fähigkeit, Ereignisse, Vorfälle und Probleme effizient zu verwalten, ist der Schlüssel zur Aufrechterhaltung des Workloads und der Leistung. Es ist wichtig, die Unterschiede zwischen diesen Elementen zu erkennen und zu verstehen, um eine effektive Reaktions- und Lösungsstrategie zu entwickeln. Die Einrichtung und Einhaltung eines klar definierten Prozesses für jeden Aspekt hilft Ihrem Team, alle auftretenden betrieblichen Herausforderungen schnell und effektiv zu bewältigen.

Gewünschtes Ergebnis: Ihr Unternehmen verwaltet betriebliche Ereignisse, Vorfälle und Probleme effektiv durch gut dokumentierte und zentral gespeicherte Prozesse. Diese Prozesse werden ständig aktualisiert, um Änderungen zu berücksichtigen, die Handhabung zu optimieren und eine hohe Servicezuverlässlichkeit und Workload-Leistung aufrechtzuerhalten.

Typische Anti-Muster:

- Sie reagieren eher reaktiv als proaktiv auf Ereignisse.
- Bei verschiedenen Arten von Ereignissen oder Vorfällen werden inkonsistente Ansätze verfolgt.
- Ihr Unternehmen analysiert keine Vorfälle und lernt nicht aus ihnen, um zukünftige Vorfälle zu verhindern.

Vorteile der Nutzung dieser bewährten Methode:

- optimierte und standardisierte Reaktionsprozesse
- geringere Auswirkungen von Vorfällen auf Services und Kunden
- beschleunigte Problemlösung
- kontinuierliche Verbesserung der betrieblichen Abläufe

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Wenn Sie diese bewährte Methode implementieren, bedeutet dies, dass Sie Workload-Ereignisse nachverfolgen. Sie haben Prozesse für den Umgang mit Vorfällen und Problemen. Die Prozesse werden dokumentiert, geteilt und oft aktualisiert. Die Probleme werden identifiziert, priorisiert und behoben.

Verstehen von Ereignissen, Vorfällen und Problemen

- Ereignisse: Bei einem Ereignis kann es sich um eine Beobachtung einer Aktion, eines Vorkommens oder einer Statusänderung handeln. Ereignisse können geplant oder ungeplant sein und sie können intern oder extern zum Workload entstehen.
- Vorfälle: Vorfälle sind Ereignisse, die eine Reaktion erfordern, wie ungeplante Unterbrechungen oder Beeinträchtigungen der Servicequalität. Sie stellen Störungen dar, die sofortige Aufmerksamkeit erfordern, um den normalen Workload-Betrieb wiederherzustellen.
- Probleme: Probleme sind die zugrundeliegenden Ursachen für einen oder mehrere Vorfälle. Bei der Identifizierung und Lösung von Problemen geht es darum, den Vorfällen auf den Grund zu gehen, um zukünftige Vorfälle zu verhindern.

Implementierungsschritte

Ereignisse

1. Überwachen von Ereignissen:
   - Implementieren Sie die Beobachtbarkeit und nutzen Sie die Beobachtbarkeit des Workloads.
   - Monitor-Aktionen, die von einem Benutzer, einer Rolle oder einem AWS-Service ausgeführt werden, werden als Ereignisse in AWS CloudTrailaufgezeichnet.

- Reagieren Sie auf betriebliche Änderungen in Ihren Anwendungen in Echtzeit mit Amazon EventBridge,.
- Kontinuierliche Bewertung, Überwachung und Aufzeichnung von Änderungen der Ressourcenkonfiguration mit AWS Config.

2. Erstellen von Prozessen:
   - Entwickeln Sie ein Verfahren zur Beurteilung, welche Ereignisse signifikant sind und überwacht werden müssen. Dies beinhaltet die Festlegung von Schwellenwerten und Parametern für normale und abnormale Aktivitäten.
   - Legen Sie Kriterien für die Eskalation eines Ereignisses in Bezug auf einen Vorfall fest. Dies kann auf Grundlage des Schweregrads, der Auswirkungen auf die Benutzer oder der Abweichung vom erwarteten Verhalten erfolgen.
   - Überprüfen Sie regelmäßig die Prozesse zur Überwachung und Reaktion auf Ereignisse. Dazu gehören die Analyse früherer Vorfälle, die Anpassung von Schwellenwerten und die Verfeinerung von Warnmechanismen.

Vorfälle

1. Reaktion auf Vorfälle:
   - Nutzen Sie die Erkenntnisse aus den Tools zur Beobachtbarkeit, um Vorfälle schnell zu erkennen und darauf zu reagieren.
   - Implementieren Sie AWS Systems Manager Ops Center , um betriebliche Aufgaben und Vorfälle zu sammeln, zu organisieren und zu priorisieren.
   - Verwenden Sie Services wie Amazon CloudWatch und AWS X-Ray für eingehendere Analysen und Problembehebungen.
   - Ziehen Sie AWS Managed Services (AMS) für ein verbessertes Vorfallmanagement in Betracht, indem Sie die proaktiven, präventiven und detektivischen Fähigkeiten nutzen. AMS erweitert den betrieblichen Support um Services wie Überwachung, Vorfallserkennung und -reaktion sowie Sicherheitsmanagement.
   - Kunden von Enterprise Support können AWS-Vorfallerkennung und -reaktionverwenden, wodurch eine kontinuierliche proaktive Überwachung und ein Vorfallmanagement für Produktions-Workloads ermöglicht wird.
2. Erstellen eines Vorfallmanagementprozesses:
   - Richten Sie einen strukturierten Vorfallmanagementprozess ein, der klare Rollen, Kommunikationsprotokolle und Lösungsschritte umfasst.

- Integrieren Sie das Vorfallmanagement mit Tools wie Amazon Q Developer in chat applications für eine effiziente Reaktion und Koordination.
- Kategorisieren Sie Vorfälle nach Schweregrad mit vordefinierten Plänen zur Vorfallreaktion für die einzelnen Kategorien.

3. Lernen und Verbessern:
   - Führen Sie Analysen nach Vorfällen durch, um die Grundursachen und die Effektivität der Lösung zu verstehen.
   - Aktualisieren und verbessern Sie die Reaktionspläne kontinuierlich auf Grundlage von Überprüfungen und sich entwickelnden Praktiken.
   - Dokumentieren Sie die gewonnenen Erkenntnisse und geben Sie sie an andere Teams weiter, um die betriebliche Widerstandsfähigkeit zu verbessern.
   - Kunden mit Enterprise Support können den Workshop zum Vorfallmanagement bei ihrem Technical Account Manager anfordern. Dieser angeleitete Workshop testet Ihren vorhandenen Reaktionsplan für Vorfälle und hilft Ihnen, Verbesserungsmöglichkeiten zu identifizieren.

Probleme

1. Identifizieren von Problemen:
   - Verwenden Sie Daten aus früheren Vorfällen, um wiederkehrende Muster zu erkennen, die auf tiefere systemische Probleme hinweisen könnten.
   - Nutzen Sie Tools wie AWS CloudTrail und Amazon CloudWatch, um Trends zu analysieren und zugrunde liegende Probleme aufzudecken.
   - Binden Sie funktionsübergreifende Teams ein, einschließlich Betriebs-, Entwicklungs- und Geschäftsbereiche, um unterschiedliche Sichtweisen auf die Grundursachen zu gewinnen.
2. Erstellen eines Problemmanagementprozesses:
   - Entwickeln Sie einen strukturierten Prozess für das Problemmanagement, der sich auf langfristige Lösungen statt auf schnelle Lösungen konzentriert.
   - Integrieren Sie Techniken zur Ursachenanalyse, um die zugrunde liegenden Ursachen von Vorfällen zu untersuchen und zu verstehen.
   - Aktualisieren Sie Betriebsrichtlinien, Verfahren und Infrastruktur auf Grundlage der Ergebnisse, um Wiederholungen zu verhindern.
3. Kontinuierliche Verbesserungen:

- Fördern Sie eine Kultur des ständigen Lernens und der Verbesserung und ermutigen Sie Ihre Teams, potenzielle Probleme proaktiv zu erkennen und anzugehen.
- Überprüfen und überarbeiten Sie regelmäßig die Problemmanagementprozesse und -tools, um sie an die sich entwickelnde Geschäfts- und Technologielandschaft anzupassen.
- Tauschen Sie Erkenntnisse und bewährte Methoden innerhalb des Unternehmens aus, um eine widerstandsfähigere und effizientere Betriebsumgebung zu schaffen.

4. Einsatz von AWS -Support:
   - Verwenden Sie AWS-Support-Ressourcen, wie z. B. AWS Trusted Advisor,für proaktive Anleitungen und Optimierungsempfehlungen.
   - Kunden von Enterprise Support können auf spezielle Programme wie AWS-Countdown zugreifen, um bei kritischen Ereignissen Unterstützung zu erhalten.
   -

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen
- OPS08-BP01 Analysieren von Workload-Metriken
- OPS11-BP02 Durchführen von Analysen nach Vorfällen

Zugehörige Dokumente:

- Leitfaden für AWS Security Incident Response
- AWS-Vorfallerkennung und -reaktion
- AWS Cloud Adoption Framework: Betriebsperspektive – Vorfall- und Problemmanagement
- Vorfallmanagement im Zeitalter von DevOps und SRE
- PagerDuty - What is Incident Management?

Zugehörige Videos:

- Die besten Tipps zur Reaktion auf Vorfälle in AWS
- AWS re:Invent 2022 – Die Amazon Builders' Library: 25 Jahre operative Exzellenz von Amazon
- AWS re:Invent 2022 – AWS-Vorfallerkennung und -reaktion (SUP201)
- Einführung von Incident Manager von AWS Systems Manager

Zugehörige Beispiele:

- AWS Proactive Services – Workshop zum Vorfallmanagement
- Automatisierung der Vorfallbehandlung mit PagerDuty und AWS Systems Manager Incident Manager
- Einbeziehung des Notfallteams in die Bereitschaftsdienstpläne in AWS Systems Manager Incident Manager
- Verbesserung der Sichtbarkeit und Zusammenarbeit bei der Bearbeitung von Vorfällen in AWS Systems Manager Incident Manager
- Vorfallberichte und Serviceanfragen in AMS

Zugehörige Services:

- Amazon EventBridge,

OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung

Die Einrichtung eines klaren und definierten Prozesses für jede Warnmeldung in Ihrem System ist für ein effektives und effizientes Vorfallmanagement unerlässlich. Diese Vorgehensweise stellt sicher, dass jede Warnmeldung zu einer spezifischen, umsetzbaren Reaktion führt, wodurch die Zuverlässigkeit und Reaktionsfähigkeit Ihrer Abläufe verbessert wird.

Gewünschtes Ergebnis: Jede Warnmeldung leitet einen bestimmten, genau definierten Reaktionsplan ein. Wenn möglich, werden die Antworten automatisiert, mit klaren Zuständigkeiten und einem definierten Eskalationspfad. Warnmeldungen sind mit einer aktuellen Wissensdatenbank verknüpft, sodass jeder Bediener konsistent und effektiv reagieren kann. Die Antworten sind schnell und einheitlich, was die betriebliche Effizienz und Zuverlässigkeit erhöht.

Typische Anti-Muster:

- Für Warnmeldungen gibt es keinen vordefinierten Reaktionsprozess, was zu provisorischen und verzögerten Lösungen führt.
- Eine Überlastung mit Warnmeldungen führt dazu, dass wichtige Warnmeldungen übersehen werden.
- Warnmeldungen werden uneinheitlich gehandhabt, da es an klaren Zuständigkeiten und Verantwortlichkeiten mangelt.

Vorteile der Nutzung dieser bewährten Methode:

- Weniger Ermüdungserscheinungen, da nur umsetzbare Warnmeldungen ausgelöst werden.
- Geringere durchschnittliche Zeit bis zur Behebung (MTTR) von Betriebsproblemen.
- Geringere durchschnittliche Zeit bis zur Untersuchung, was zur Verringerung der MTTR beiträgt.
- Verbesserte Fähigkeit, operative Reaktionen zu skalieren.
- Verbesserte Konsistenz und Zuverlässigkeit bei der Behandlung von Betriebsereignissen.

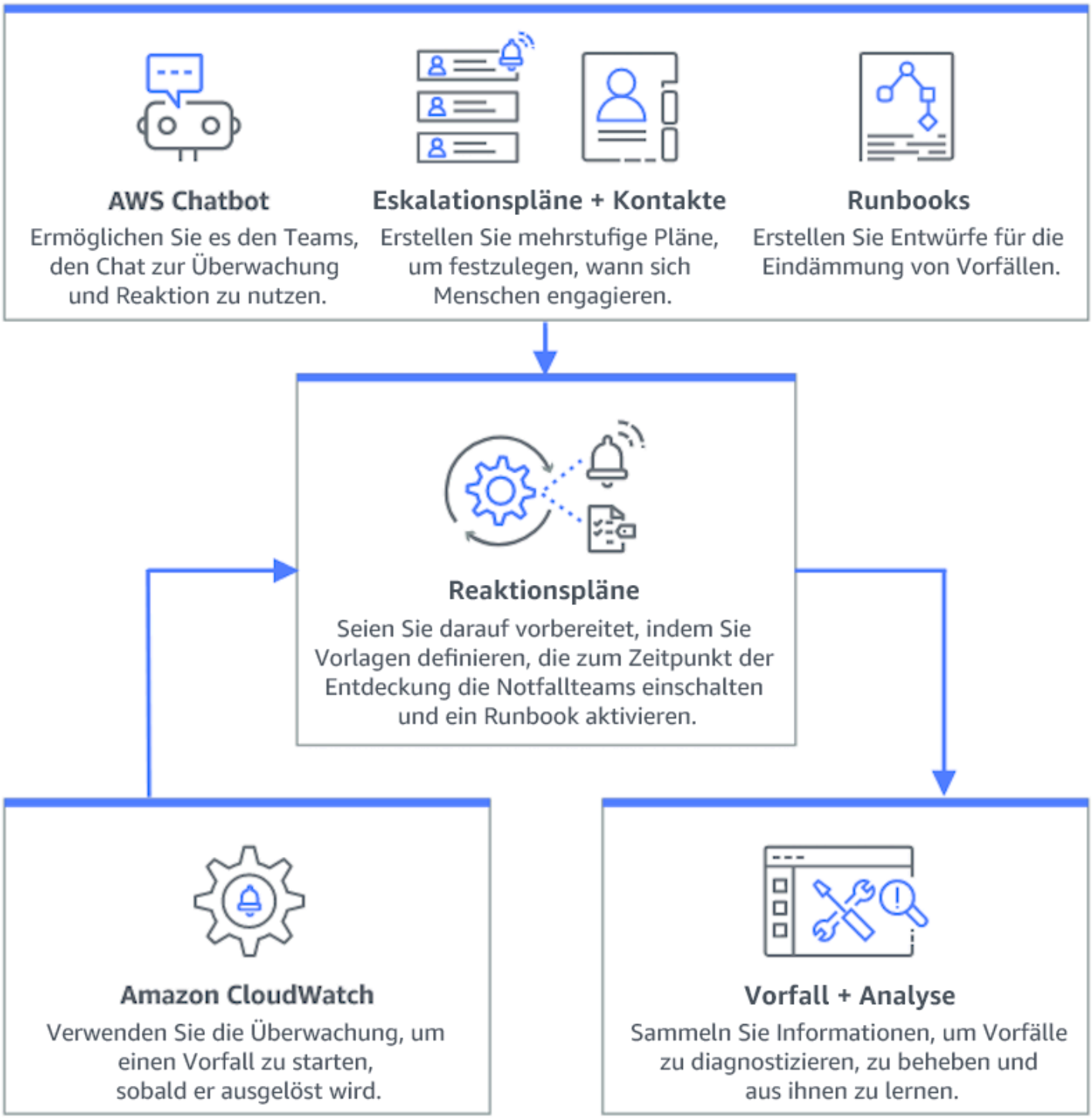
Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Ein Prozess pro Warnmeldung beinhaltet die Erstellung eines klaren Reaktionsplans für jede Warnmeldung, die Automatisierung von Reaktionen (soweit dies möglich ist) und die kontinuierliche Optimierung dieser Prozesse auf Grundlage des betrieblichen Feedbacks und der sich entwickelnden Anforderungen.

Implementierungsschritte

Das folgende Diagramm veranschaulicht den Arbeitsablauf für das Vorfallmanagement in AWS Systems Manager Incident Manager. Es ist so konzipiert, dass es schnell auf betriebliche Probleme reagiert, indem es automatisch Vorfälle als Reaktion auf bestimmte Ereignisse von Amazon CloudWatch oder Amazon EventBridgegeneriert. Wenn ein Vorfall entweder automatisch oder manuell erstellt wird, zentralisiert Incident Manager die Verwaltung des Vorfalls, organisiert relevante Informationen über AWS-Ressourcen und initiiert vordefinierte Reaktionspläne. Dazu gehört das Ausführen von Systems Manager-Automation-Runbooks für sofortige Maßnahmen sowie das Erstellen eines übergeordneten betrieblichen Arbeitselements in OpsCenter, um verwandte Aufgaben und Analysen zu verfolgen. Dieser optimierte Prozess beschleunigt und koordiniert die Reaktion auf Vorfälle in Ihrer gesamten AWS-Umgebung.



1. Zusammengesetzte Alarme, Erstellen Sie zusammengesetzte Alarme in CloudWatch, um zusammenhängende Alarme zu gruppieren, das Rauschen zu reduzieren und sinnvollere Reaktionen zu ermöglichen.

2. Integrieren Sie Amazon CloudWatch-Alarme mit Incident Manager Konfigurieren Sie CloudWatch-Alarme zur automatischen Erstellung von Vorfällen in AWS Systems Manager Incident Manager.
3. Verwenden von Amazon EventBridge mit Incident Manager: Erstellen Sie EventBridge-Regeln, um auf Ereignisse zu reagieren und Vorfälle mithilfe definierter Reaktionspläne zu erstellen.
4. Vorbereitung auf Vorfälle in Incident Manager:
   - Stellen Sie detaillierte Reaktionspläne in Incident Manager für jede Art von Warnmeldung auf.
   - Richten Sie über Amazon Q Developer in chat applications Chat-Kanäle ein, die mit Reaktionsplänen in Incident Manager verknüpft sind und die Echtzeitkommunikation bei Vorfällen über Plattformen wie Slack, Microsoft Teams und Amazon Chime ermöglichen.
   - Integrieren Sie Systems Manager-Automation-Runbooks in Incident Manager, um automatisierte Reaktionen auf Vorfälle zu ermöglichen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen

Zugehörige Dokumente:

- AWS Cloud Adoption Framework: Betriebsperspektive – Vorfall- und Problemmanagement
- Verwenden von Amazon CloudWatch-Alarmen
- Erste Schritte mit AWS Systems Manager Incident Manager
- Vorbereitung auf Vorfälle in Incident Manager

Zugehörige Videos:

- Die besten Tipps zur Reaktion auf Vorfälle in AWS

Zugehörige Beispiele:

- AWS-Workshops – AWS Systems Manager Incident Manager – Automatisierung der Reaktion auf Sicherheitsvorfälle

OPS10-BP03 Priorisieren von betrieblichen Ereignissen auf Basis der Auswirkung auf das Unternehmen

Eine schnelle Reaktion auf Betriebsereignisse ist von entscheidender Bedeutung, aber nicht alle Ereignisse sind gleich. Wenn Sie Ihre Prioritäten auf Grundlage der geschäftlichen Auswirkungen festlegen, müssen Sie sich auch vorrangig mit Ereignissen befassen, die erhebliche Folgen haben könnten, wie z. B. Sicherheit, finanzielle Verluste, Verstöße gegen Vorschriften oder Rufschädigung.

Gewünschtes Ergebnis: Die Reaktionen auf betriebliche Ereignisse werden auf Grundlage der potenziellen Auswirkungen auf die Geschäftsabläufe und -ziele priorisiert. Dadurch werden die Reaktionen effizient und effektiv.

Typische Anti-Muster:

- Jedes Ereignis wird mit der gleichen Dringlichkeit behandelt, was zu Verwirrung und Verzögerungen bei der Behandlung kritischer Probleme führt.
- Sie unterscheiden nicht zwischen Ereignissen mit hoher und geringer Auswirkung, was zu einer Fehlallokation von Ressourcen führt.
- Ihrem Unternehmen fehlt ein klarer Rahmen für die Priorisierung, was zu inkonsistenten Reaktionen auf Betriebsereignisse führt.
- Ereignisse werden in der Reihenfolge ihrer Meldung priorisiert und nicht nach ihrer Auswirkung auf die Geschäftsergebnisse.

Vorteile der Nutzung dieser bewährten Methode:

- Stellt sicher, dass wichtige Geschäftsfunktionen zuerst berücksichtigt werden, um mögliche Schäden zu minimieren.
- Verbessert die Ressourcenzuweisung bei mehreren gleichzeitigen Ereignissen.
- Verbessert die Fähigkeit der Organisation, das Vertrauen zu erhalten und die gesetzlichen Anforderungen zu erfüllen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Wenn Sie mit mehreren betrieblichen Ereignissen konfrontiert sind, ist ein strukturierter Ansatz zur Priorisierung auf Grundlage von Auswirkungen und Dringlichkeit unerlässlich. Dieser Ansatz hilft

Ihnen, fundierte Entscheidungen zu treffen, Ihre Maßnahmen auf die Bereiche zu lenken, wo sie am dringendsten benötigt werden, und das Risiko für die Geschäftskontinuität zu mindern.

Implementierungsschritte

1. Bewertung der Auswirkungen: Entwickeln Sie ein Klassifizierungssystem, um den Schweregrad von Ereignissen im Hinblick auf ihre potenziellen Auswirkungen auf den Geschäftsbetrieb und die Ziele zu bewerten. Das folgende Beispiel zeigt die Wirkungskategorien:

Auswirkungsgrad	Beschreibung
Hoch	Betrifft viele Mitarbeiter oder Kunden, hohe finanzielle Auswirkungen, hoher Reputatio nsschaden oder Verletzungen
Mittel	Betrifft eine Gruppe von Mitarbeitern oder Kunden, mäßige finanzielle Auswirkungen oder mäßiger Reputationsschaden
Niedrig	Betrifft einzelne Mitarbeiter oder Kunden, geringe finanzielle Auswirkungen oder geringer Reputationsschaden

2. Bewertung der Dringlichkeit: Definieren Sie Dringlichkeitsstufen danach, wie schnell auf ein Ereignis reagiert werden muss, und berücksichtigen Sie dabei Faktoren wie Sicherheit, finanzielle Auswirkungen und Service Level Agreements (SLAs). Das folgende Beispiel zeigt die Dringlichkeitskategorien:

Dringlichkeitsstufe	Beschreibung
Hoch	Exponentiell steigender Schaden, Beeinträc htigung zeitkritischer Aufgaben, drohende Eskalation oder betroffene VIP-Benutzer oder Gruppen
Mittel	Der Schaden nimmt im Laufe der Zeit zu oder es ist ein einzelner VIP-Benutzer oder eine Gruppe betroffen

Dringlichkeitsstufe	Beschreibung
Niedrig	Geringfügige Schadenszunahme im Laufe der Zeit oder nicht zeitkritische Arbeit beeinträc htigt

3. Erstellen einer Priorisierungsmatrix:
   - Verwenden Sie eine Matrix, um Auswirkungen und Dringlichkeit miteinander zu vergleichen, und weisen Sie verschiedenen Kombinationen Prioritätsstufen zu.
   - Machen Sie die Matrix allen Teammitgliedern, die für die Reaktion auf betriebliche Ereignisse verantwortlich sind, zugänglich und verständlich.
   - Die folgende Beispielmatrix zeigt den Schweregrad eines Vorfalls nach Dringlichkeit und Auswirkung an:

Dringlichkeit und Auswirkungen	Hoch	Mittel	Niedrig
Hoch	Kritisch	Dringend	Hoch
Mittel	Dringend	Hoch	Normal
Niedrig	Hoch	Normal	Niedrig

4. Trainieren und Kommunizieren: Schulen Sie die Response-Teams im Umgang mit der Prioritätenmatrix und der Wichtigkeit, diese während eines Ereignisses zu befolgen. Kommunizieren Sie den Priorisierungsprozess an alle Beteiligten, um klare Erwartungen zu schaffen.
5. Integration der Vorfallreaktion:
   - Integrieren Sie die Priorisierungsmatrix in Ihre Pläne und Tools zur Reaktion auf Vorfälle.
   - Automatisieren Sie nach Möglichkeit die Klassifizierung und Priorisierung von Ereignissen, um die Reaktionszeiten zu verkürzen.
   - Kunden von Enterprise Support können AWS-Vorfallerkennung und -reaktionverwenden, um die proaktive Überwachung und das Vorfallmanagement für Produktions-Workloads rund um die Uhr zu gewährleisten.

6. Überprüfen und Anpassen: Überprüfen Sie regelmäßig die Effektivität des Priorisierungsprozesses und nehmen Sie Anpassungen auf der Grundlage von Rückmeldungen und Änderungen im Geschäftsumfeld vor.

Ressourcen

Zugehörige bewährte Methoden:

- OPS03-BP03 Eskalation wird empfohlen
- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen
- OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken

Zugehörige Dokumente:

- Atlassian – Verständnis der Schweregrade von Vorfällen
- IT-Prozessplan – Checkliste der Vorfallpriorität

OPS10-BP04 Definieren von Eskalationspfaden

Legen Sie in Ihren Protokollen zur Vorfallreaktion klare Eskalationspfade fest, um rechtzeitige und effektive Maßnahmen zu ermöglichen. Dazu gehören die Festlegung von Aufforderungen zur Eskalation, die detaillierte Beschreibung des Eskalationsprozesses und die vorherige Genehmigung von Maßnahmen, um die Entscheidungsfindung zu beschleunigen und die durchschnittliche Zeit für die Behebung zu verkürzen.

Gewünschtes Ergebnis: Ein strukturierter und effizienter Prozess, der Vorfälle an das entsprechende Personal weiterleitet und so die Reaktionszeiten und Auswirkungen minimiert.

Typische Anti-Muster:

- Mangelnde Klarheit über die Wiederherstellungsverfahren führt zu provisorischen Maßnahmen bei kritischen Vorfällen.
- Das Fehlen von definierten Berechtigungen und Zuständigkeiten führt zu Verzögerungen, wenn dringende Maßnahmen erforderlich sind.
- Stakeholder und Kunden werden nicht erwartungsgemäß informiert.
- Wichtige Entscheidungen verzögern sich.

Vorteile der Nutzung dieser bewährten Methode:

- Optimierte Reaktion auf Vorfälle durch vordefinierte Eskalationsverfahren.
- Reduzierte Ausfallzeiten durch vorab genehmigte Maßnahmen und klare Zuständigkeiten.
- Verbesserte Ressourcenzuweisung und Anpassung der Support-Ebene an den Schweregrad des Vorfalls.
- Verbesserte Kommunikation mit Stakeholdern und Kunden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Richtig definierte Eskalationspfade sind entscheidend für eine schnelle Reaktion auf Vorfälle. AWS Systems Manager Incident Manager unterstützt die Einrichtung strukturierter Eskalations- und Bereitschaftspläne, die die richtigen Mitarbeiter alarmieren, damit sie bei Vorfällen handlungsbereit sind.

Implementierungsschritte

1. Einrichtung von Eskalationsaufforderungen: Richten Sie CloudWatch-Alarme ein, um einen Vorfall in AWS Systems Manager Incident Managerverwenden.
2. Erstellen von Bereitschaftsplänen: Erstellen Sie Bereitschaftspläne in Incident Manager, die mit Ihren Eskalationspfaden übereinstimmen. Statten Sie das Bereitschaftspersonal mit den erforderlichen Berechtigungen und Tools aus, um schnell handeln zu können.
3. Detaillierte Eskalationsverfahren:
   - Legen Sie bestimmte Bedingungen fest, unter denen ein Vorfall eskaliert werden sollte.
   - Erstellen Sie Eskalationspläne in Incident Manager.
   - Eskalationskanäle sollten aus einem Ansprechpartner oder einem Bereitschaftsplan bestehen.
   - Definieren Sie die Rollen und Verantwortlichkeiten des Teams auf jeder Eskalationsstufe.
4. Genehmigung von Schadensbegrenzungsmaßnahmen im Voraus: Arbeiten Sie mit Entscheidungsträgern zusammen, um Maßnahmen für erwartete Szenarien vorab zu genehmigen. Verwenden Sie Systems Manager-Automation-Runbooks, die mit Incident Manager integriert sind, um die Behebung von Vorfällen zu beschleunigen.
5. Angabe der Zuständigkeit: Identifizieren Sie eindeutig die internen Besitzer für jeden Schritt des Eskalationspfads.

6. Details zu Eskalationen mit Drittanbietern:
   - Dokumentieren Sie Service Level Agreements (SLAs) von Drittanbietern und richten Sie sie an internen Zielen aus.
   - Legen Sie klare Protokolle für die Lieferantenkommunikation bei Vorfällen fest.
   - Integrieren Sie Lieferantenkontakte in die Tools zum Vorfallmanagement, um direkten Zugriff zu erhalten.
   - Führen Sie regelmäßige Übungen durch, die Reaktionsszenarien von Drittanbietern beinhalten.
   - Sorgen Sie dafür, dass die Informationen zur Lieferanteneskalation gut dokumentiert und leicht zugänglich sind.
7. Trainieren und Testen von Eskalationsplänen: Schulen Sie Ihr Team im Eskalationsprozess und führen Sie regelmäßig Übungen zur Reaktion auf Vorfälle oder den Ernstfall durch. Kunden mit Enterprise Support können einen Workshop zum Vorfallmanagement anfordernverwenden.
8. Kontinuierliche Verbesserungen: Überprüfen Sie regelmäßig die Wirksamkeit Ihrer Eskalationspfade. Aktualisieren Sie Ihre Prozesse auf Grundlage der Erkenntnisse aus den Nachuntersuchungen von Vorfällen und dem kontinuierlichen Feedback.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen
- OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung
- OPS11-BP02 Durchführen von Analysen nach Vorfällen

Zugehörige Dokumente:

- AWS Systems Manager Incident Manager-Eskalationspläne
- Arbeiten mit Bereitschaftsplänen in Incident Manager
- Erstellen und Verwalten von Runbooks
- Temporäre erweiterte Zugriffsverwaltung mit AWS IAM Identity Center
- Atlassian – Eskalationsrichtlinien für effektives Vorfallmanagement

OPS10-BP05 Definieren eines Kundenkommunikationsplan für Ereignisse, die sich auf den Service auswirken

Eine effektive Kommunikation bei Ereignissen, die sich auf den Service auswirken, ist entscheidend, um das Vertrauen und die Transparenz gegenüber den Kunden aufrechtzuerhalten. Ein klar definierter Kommunikationsplan hilft Ihrem Unternehmen, bei Vorfällen schnell und klar Informationen sowohl intern als auch extern auszutauschen.

Gewünschtes Ergebnis:

- Ein robuster Kommunikationsplan, der Kunden und Interessengruppen bei Ereignissen, die sich auf den Service auswirken, effektiv informiert.
- Transparenz in der Kommunikation, um Vertrauen aufzubauen und Ängste der Kunden abzubauen.
- Minimierung der Auswirkungen von Ereignissen, die sich auf den Service in Bezug auf das Kundenerlebnis und den Geschäftsbetrieb auswirken.

Typische Anti-Muster:

- Eine unzureichende oder verzögerte Kommunikation führt zu Verwirrung und Unzufriedenheit der Kunden.
- Allzu technische oder vage Nachrichten vermitteln nicht die tatsächlichen Auswirkungen auf die Benutzer.
- Es gibt keine vordefinierte Kommunikationsstrategie, was zu inkonsistenten und reaktiven Nachrichten führt.

Vorteile der Nutzung dieser bewährten Methode:

- Mehr Vertrauen und Zufriedenheit bei den Kunden durch proaktive und klare Kommunikation.
- Entlastung der Support-Teams durch präventive Behandlung von Kundenanliegen.
- Verbesserte Fähigkeit, Vorfälle effektiv zu verwalten und zu bewältigen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die Erstellung eines umfassenden Kommunikationsplans für Veranstaltungen, die sich auf den Service auswirken, umfasst mehrere Facetten, von der Auswahl der richtigen Kanäle bis hin zur

Formulierung der Botschaft und des Tonfalls. Der Plan sollte anpassungsfähig und skalierbar sein und verschiedene Ausfallszenarien berücksichtigen.

Implementierungsschritte

1. Definieren von Rollen und Zuständigkeiten:
   - Beauftragen Sie einen Hauptzuständigen für die Vorfallreaktion mit der Überwachung der Maßnahmen.
   - Benennen Sie einen Kommunikationsmanager, der für die Koordination der gesamten externen und internen Kommunikation verantwortlich ist.
   - Beziehen Sie den Support-Manager ein, um eine konsistente Kommunikation über Support-Tickets zu gewährleisten.
2. Identifizieren von Kommunikationskanälen: Wählen Sie Kanäle wie Arbeitsplatz-Chat, E-Mail, SMS, soziale Medien, In-App-Benachrichtigungen und Statusseiten aus. Diese Kanäle sollten robust und in der Lage sein, bei Ereignissen, die den Service beeinträchtigen, unabhängig zu arbeiten.
3. Schnelle, klare und regelmäßige Kommunikation mit Kunden:
   - Entwickeln Sie Vorlagen für verschiedene Szenarien, bei denen Beeinträchtigungen des Serviceangebots vorliegen, und achten Sie dabei auf Einfachheit und wichtige Details. Fügen Sie Informationen über die Beeinträchtigung des Services, die erwartete Lösungszeit und die Auswirkungen hinzu.
   - Verwenden Sie Amazon Pinpoint, um Kunden mithilfe von Push-Benachrichtigungen, In-App-Benachrichtigungen, E-Mails, Textnachrichten, Sprachnachrichten und Nachrichten über benutzerdefinierte Kanäle zu informieren.
   - Verwenden Sie Amazon Simple Notification Service (Amazon SNS), um Subscriber programmgesteuert oder per E-Mail, mobilen Push-Benachrichtigungen und Textnachrichten zu benachrichtigen.
   - Kommunizieren Sie den Status über Dashboards, indem Sie ein Amazon CloudWatch-Dashboard öffentlich teilen.
   - Förderung des Engagements in den sozialen Medien:
     - Verfolgen Sie aktiv die sozialen Medien, um die Stimmung der Kunden zu verstehen.
     - Posten Sie auf Social-Media-Plattformen, um die Öffentlichkeit auf dem Laufenden zu halten und die Community einzubeziehen.
     - Bereiten Sie Vorlagen für eine konsistente und klare Kommunikation in den sozialen Medien vor.

4. Koordinieren Sie die interne Kommunikation: Implementieren Sie interne Protokolle mithilfe von Tools wie Amazon Q Developer in chat applications für die Teamkoordination und Kommunikation. Verwenden Sie CloudWatch-Dashboards, um den Status zu kommunizieren.
5. Organisation der Kommunikation mit speziellen Tools und Services:
   - Verwenden Sie AWS Systems Manager Incident Manager mit Amazon Q Developer in chat applications, um spezielle Chat-Kanäle für die interne Kommunikation und Koordination in Echtzeit bei Vorfällen einzurichten.
   - Verwenden Sie AWS Systems Manager Incident Manager-Runbooks, um Kundenbenachrichtigungen über Amazon Pinpoint, Amazon SNS oder Tools von Drittanbietern wie Social-Media-Plattformen bei Vorfällen zu automatisieren.
   - Integrieren Sie Genehmigungs-Workflows in Runbooks, um optional die gesamte externe Kommunikation vor dem Versand zu überprüfen und zu autorisieren.
6. Praktizieren und verbessern:
   - Führen Sie Trainingkurse zum Einsatz von Kommunikationsmitteln und -strategien durch. Ermöglichen Sie es Teams, bei Vorfällen rechtzeitig Entscheidungen zu treffen.
   - Testen Sie den Kommunikationsplan durch regelmäßige Übungen oder Ernstfallübungen. Mithilfe dieser Tests können Sie Ihre Botschaften präzisieren und die Effektivität der Kanäle bewerten.
   - Implementieren Sie Feedback-Mechanismen, um die Effektivität der Kommunikation bei Vorfällen zu bewerten. Entwickeln Sie den Kommunikationsplan auf Grundlage des Feedbacks und der sich ändernden Bedürfnisse kontinuierlich weiter.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren
- OPS10-BP06 Bekanntgeben des Status über Dashboards
- OPS11-BP02 Durchführen von Analysen nach Vorfällen

Zugehörige Dokumente:

- Atlassian – Bewährte Methoden der Kommunikation bei Vorfällen

- Atlassian – Verfassen eines guten Status-Updates
- PagerDuty – Leitfaden für die Kommunikation bei Vorfällen

Zugehörige Videos:

- Atlassian – Erstellung eines eigenen Kommunikationsplans für Vorfälle: Vorlagen für Zwischenfälle

Zugehörige Beispiele:

- AWS Health-Dashboard
- Beispiel für AWS-Status-Updates

OPS10-BP06 Bekanntgeben des Status über Dashboards

Verwenden Sie Dashboards als strategisches Werkzeug, um den Betriebsstatus und wichtige Metriken in Echtzeit an verschiedene Zielgruppen zu vermitteln, darunter interne technische Teams, Führungskräfte und Kunden. Diese Dashboards bieten eine zentrale, visuelle Darstellung des Systemzustands und der Geschäftsleistung und erhöhen so die Transparenz und die Effizienz der Entscheidungsfindung.

Gewünschtes Ergebnis:

- Ihre Dashboards bieten einen umfassenden Überblick über das System und die Geschäftskennzahlen, die für verschiedene Interessengruppen relevant sind.
- Stakeholder können proaktiv auf Betriebsinformationen zugreifen, sodass keine häufigen Statusanfragen mehr erforderlich sind.
- Die Entscheidungsfindung in Echtzeit wird während des normalen Betriebs und bei Vorfällen verbessert.

Typische Anti-Muster:

- Techniker, die an einem Vorfallmanagementgespräch teilnehmen, benötigen Statusaktualisierungen, um sich auf dem Laufenden zu halten.
- Sie verlassen sich auf die manuelle Berichterstattung für das Management, was zu Verzögerungen und möglichen Ungenauigkeiten führt.
- Die Arbeit der Operations-Teams wird bei Vorfällen häufig für Statusaktualisierungen unterbrochen.

Vorteile der Nutzung dieser bewährten Methode:

- Ermöglicht Stakeholdern den sofortigen Zugriff auf wichtige Informationen und fördert so fundierte Entscheidungen.
- Reduziert betriebliche Ineffizienzen, indem manuelle Berichte und häufige Statusabfragen minimiert werden.
- Erhöht die Transparenz und das Vertrauen durch Echtzeiteinblicke in die Systemleistung und Geschäftskennzahlen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Dashboards vermitteln effektiv den Status Ihrer Systeme und Geschäftskennzahlen und können auf die Bedürfnisse verschiedener Zielgruppen zugeschnitten werden. Mit Tools wie Amazon CloudWatch-Dashboards und Amazon QuickSight können Sie interaktive Echtzeit-Dashboards für die Systemüberwachung und Business Intelligence erstellen.

Implementierungsschritte

1. Ermittlung der Bedürfnisse der Stakeholder: Ermitteln Sie den spezifischen Informationsbedarf verschiedener Zielgruppen, z. B. technische Teams, Führungskräfte und Kunden.
2. Wählen der richtigen Tools: Wählen Sie geeignete Tools wie Amazon CloudWatch-Dashboards für die Systemüberwachung und Amazon QuickSight für interaktive Business Intelligence aus.
3. Entwicklung effektiver Dashboards:
   - Entwickeln Sie Dashboards, um relevante Metriken und KPIs übersichtlich darzustellen und sicherzustellen, dass sie verständlich und umsetzbar sind.
   - Integrieren Sie bei Bedarf Ansichten auf System- und Unternehmensebene.
   - Inkludieren Sie sowohl Dashboards auf hoher Ebene (für umfassende Übersichten) als auch auf niedriger Ebene (für detaillierte Analysen).
   - Integrieren Sie automatische Alarme in Dashboards, um kritische Probleme hervorzuheben.
   - Kommentieren Sie Dashboards mit wichtigen Schwellenwerten und Zielen für sofortige Sichtbarkeit.
4. Integration von Datenquellen:

- Verwenden Sie Amazon CloudWatch, um Metriken von verschiedenen AWS-Services zu aggregieren und anzuzeigen und Metriken aus anderen Datenquellen abzufragen.So erhalten Sie einen einheitlichen Überblick über den Zustand Ihres Systems und Ihre Geschäftsmetriken.
- Nutzen Sie Features wie CloudWatch Logs Insights, um Protokolldaten aus verschiedenen Anwendungen und Services abzufragen und zu visualisieren.

5. Bereitstellung von Selfservice-Zugriff:
   - Teilen Sie CloudWatch-Dashboards mit relevanten Stakeholdern für den Selfservice-Zugriff auf Informationen mithilfe von Dashboard-Freigabe-Features.
   - Stellen Sie sicher, dass Dashboards leicht zugänglich sind und aktuelle Informationen in Echtzeit bereitstellen.
6. Regelmäßige Aktualisierungen und Verbesserungen:
   - Aktualisieren und verbessern Sie die Dashboards kontinuierlich, um sie an die sich entwickelnden Geschäftsanforderungen und das Feedback der Stakeholder anzupassen.
   - Überprüfen Sie die Dashboards regelmäßig, um sicherzustellen, dass sie relevant und effektiv sind, um die erforderlichen Informationen zu vermitteln.

Ressourcen

Zugehörige bewährte Methoden:

- OPS08-BP05 Erstellen von Dashboards

Zugehörige Dokumente:

- Erstellung von Dashboards für operative Sichtbarkeit
- Verwenden von Amazon CloudWatch-Dashboards
- Erstellen flexibler Dashboards mit Dashboard-Variablen
- Freigabe von CloudWatch-Dashboards
- Abfrage von Metriken aus anderen Datenquellen
- Hinzufügen eines benutzerdefinierten Widgets zu einem CloudWatch-Dashboard

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit – Dashboards

OPS10-BP07 Automatisieren von Reaktionen auf Ereignisse

Die Automatisierung von Reaktionen auf Ereignisse ist der Schlüssel für eine schnelle, konsistente und fehlerfreie operative Abwicklung. Erstellen Sie optimierte Prozesse und verwenden Sie Tools, um Ereignisse automatisch zu verwalten und darauf zu reagieren, um manuelle Eingriffe zu minimieren und die betriebliche Effizienz zu steigern.

Gewünschtes Ergebnis:

- weniger menschliche Fehler und schnellere Lösungszeiten durch Automatisierung
- konsistente und zuverlässige Handhabung betrieblicher Ereignisse
- verbesserte betriebliche Effizienz und Systemzuverlässigkeit

Typische Anti-Muster:

- manuelle Behandlung von Ereignissen führt zu Verzögerungen und Fehlern
- bei sich wiederholenden, kritischen Aufgaben wird die Automatisierung übersehen
- sich wiederholende, manuelle Aufgaben führen zu Ermüdungserscheinungen und zum Übersehen kritischer Probleme

Vorteile der Nutzung dieser bewährten Methode:

- beschleunigte Reaktionen auf Ereignisse, wodurch sich die Ausfallzeiten des Systems reduzieren
- zuverlässiger Betrieb mit automatisierter und konsistenter Ereignisbehandlung

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Integrieren Sie Automatisierung, um effiziente Arbeitsabläufe zu schaffen und manuelle Eingriffe zu minimieren.

Implementierungsschritte

1. Identifizieren von Möglichkeiten zur Automatisierung: Bestimmen Sie sich wiederholende Aufgaben für die Automatisierung, wie beispielsweise Problembehebung, Ticketverbesserung, Kapazitätsmanagement, Skalierung, Bereitstellung und Tests.

2. Identifizieren von Automatisierungsaufforderungen:
   - Bewerten und definieren Sie bestimmte Bedingungen oder Metriken, die automatische Reaktionen mithilfe von Amazon CloudWatch-Alarmaktionenauslösen.
   - Verwendung von Amazon EventBridge, um auf Ereignisse in AWS-Services, benutzerdefinierten Workloads und SaaS-Anwendungen zu reagieren.
   - Denken Sie an Initiationsereignisse wie bestimmte Protokolleinträge, Schwellenwerte für Leistungsmetrikenoder Zustandsänderungen in AWS-Ressourcen.
3. Implementieren der ereignisgesteuerten Automatisierung:
   - Verwenden Sie AWS Systems Manager-Automation-Runbooks, um die Wartung, Bereitstellung und Problembehebung zu vereinfachen.
   - Beim Erstellen von Vorfällen in Incident Manager werden automatisch Details zu den betroffenen AWS-Ressourcen erfasst und dem Vorfall hinzugefügt.
   - Überwachen Sie Quoten proaktiv mit Quota Monitor für AWS.
   - Passen Sie die Kapazität mit AWS Auto Scaling automatisch an, um Verfügbarkeit und Leistung aufrechtzuerhalten.
   - Automatisieren Sie Entwicklungspipelines mit Amazon CodeCatalyst.
   - Führen Sie Smoke Tests durch oder überwachen Sie Endpunkte und APIs kontinuierlich mit synthetischer Überwachung.
4. Schadensbegrenzung durch Automatisierung:
   - Implementieren Sie automatisierte Sicherheitsmaßnahmen, um schnell auf Risiken zu reagieren.
   - Verwenden Sie AWS Systems Manager State Manager, um Konfigurationsabweichungen zu reduzieren.
   - Korrigieren Sie nicht konforme Ressourcen mit AWS-Config-Regeln.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen
- OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung

Zugehörige Dokumente:

- Verwendung von Systems-Manager-Automation-Runbooks mit Incident Manager
- Erstellen von Vorfällen in Incident Manager
- AWS Service Quotas
- Überwachen der Ressourcennutzung und Senden von Benachrichtigungen, wenn das Kontingent fast erreicht ist
- AWS Auto Scaling
- Was ist Amazon CodeCatalyst?
- Verwenden von Amazon CloudWatch-Alarmen
- Verwenden von Amazon CloudWatch-Alarmaktionen
- Korrigieren von nicht konformen AWS-Config-Regeln-Ressourcen
- Erstellen von Metriken aus Protokollereignissen mit Filtern
- AWS Systems Manager State Manager

Zugehörige Videos:

- Erstellen von Automation-Runbooks mit AWS Systems Manager
- Automatisierung von IT-Abläufen in AWS
- Automatisierungsregeln für AWS Security Hub
- Amazon CodeCatalyst-Vorlagen sorgen für einen schnellen Start Ihres Softwareprojekts

Zugehörige Beispiele:

- Amazon CodeCatalyst-Tutorial: Erstellen eines Projekts mit der dreistufigen Vorlage für moderne Webanwendungen
- Workshop zur Beobachtbarkeit
- Reaktion auf Vorfälle mit Incident Manager

Weiterentwicklung

Frage

- OPS 11. Wie können Sie Arbeitsvorgänge weiterentwickeln?

OPS 11. Wie können Sie Arbeitsvorgänge weiterentwickeln?

Widmen Sie nahezu kontinuierlichen inkrementellen Verbesserungen Zeit und Ressourcen, um die Effektivität und Effizienz Ihrer operativen Abläufe weiterzuentwickeln.

Bewährte Methoden

- OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung
- OPS11-BP02 Durchführen von Analysen nach Vorfällen
- OPS11-BP03 Implementieren von Feedbackschleifen
- OPS11-BP04 Wissensmanagement
- OPS11-BP05 Definieren von Verbesserungsfaktoren
- OPS11-BP06 Prüfen von Erkenntnissen
- OPS11-BP07 Prüfung von Betriebsmetriken
- OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen
- OPS11-BP09 Einplanen von Zeit für Verbesserungen

OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung

Bewerten Sie Ihren Workload mithilfe bewährter Methoden für interne und externe Architekturen. Führen Sie häufige, bewusste Workload-Überprüfungen durch. Räumen Sie Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsplan Priorität ein.

Gewünschtes Ergebnis:

- Sie analysieren Ihren Workload mindestens einmal im Jahr anhand bewährter Methoden für die Architektur.
- Sie räumen den Features in Ihrem Softwareentwicklungsprozess die gleiche Priorität wie Verbesserungsmöglichkeiten ein.

Typische Anti-Muster:

- Sie haben seit der Bereitstellung Ihres Workloads vor einigen Jahren keine Architekturüberprüfung durchgeführt.
- Verbesserungsmöglichkeiten haben geringere Priorität. Im Vergleich zu neuen Features bleiben diese Möglichkeiten im Backlog.

- In der Organisation gibt keinen Standard für die Umsetzung von Änderungen an bewährten Methoden.

Vorteile der Nutzung dieser bewährten Methode:

- Ihr Workload wird durch bewährte Methoden für die Architektur auf dem aktuellen Stand gehalten.
- Sie entwickeln Ihren Workload gezielt weiter.
- Sie können die bewährten Methoden der Organisation nutzen, um alle Workloads zu verbessern.
- Sie erzielen marginale Gewinne, deren kumulative Wirkung jedoch zu einer höheren Effizienz führen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Führen Sie regelmäßig eine Überprüfung der Architektur Ihres Workloads durch. Bewerten Sie anhand interner und externer bewährter Methoden Ihren Workload und ermitteln Sie Verbesserungsmöglichkeiten. Räumen Sie Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsplan Priorität ein.

Implementierungsschritte

1. Führen Sie in vereinbarten Intervallen Überprüfungen der Architektur Ihrer Produktionsworkloads durch. Verwenden Sie einen dokumentierten Architekturstandard mit AWS-spezifischen bewährten Methoden.
   a. Verwenden Sie Ihre intern definierten Standards für diese Bewertungen. Wenn Sie nicht über einen internen Standard verfügen, verwenden Sie das AWS Well-Architected Framework.
   b. Verwenden Sie AWS Well-Architected Tool, um einen Fokusbereich Ihrer internen bewährten Methoden zu erstellen und Ihre Architekturprüfung durchzuführen.
   c. Wenden Sie sich an Ihren AWS Solution Architect oder Technical Account Manager, um einen geführten Well-Architected Framework Review Ihres Workload durchzuführen.
2. Räumen Sie den während der Überprüfung ermittelten Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsprozess Priorität ein.

Aufwand des Implementierungsplans: niedrig Sie können das AWS Well-Architected Framework zur Durchführung Ihrer jährlichen Architekturprüfung verwenden.

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP02 Durchführen von Analysen nach Vorfällen
- OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen
- OPS04 Implementieren von Beobachtbarkeit

Zugehörige Dokumente:

- AWS Well-Architected Tool – Fokusbereiche
- AWS Well-Architected Whitepaper – Die Überprüfung
- Anpassen von Well-Architected-Prüfungen mit Fokusbereichen und dem AWS Well-Architected Tool
- Implementieren des AWS Well-Architected-Fokusbereich-Lebenszyklus in Ihre Organisation

Zugehörige Videos:

- Well-Architected Labs – Stufe 100: Fokusbereiche auf AWS Well-Architected Tool
- AWS re:Invent 2023 – Skalierung bewährter Methoden von AWS Well-Architected in Ihrer Organisation

Zugehörige Beispiele:

- AWS Well-Architected Tool

OPS11-BP02 Durchführen von Analysen nach Vorfällen

Überprüfen Sie die Ereignisse mit Auswirkungen auf Kunden und bestimmen Sie die beitragenden Faktoren und Präventivmaßnahmen. Entwickeln Sie anhand dieser Informationen Abhilfemaßnahmen, um Wiederholungen einzuschränken oder zu verhindern. Entwickeln Sie Verfahren für schnelle und effektive Reaktionen. Informieren Sie nach Bedarf auf zielgruppengerechte Weise über beitragende Faktoren und Korrekturmaßnahmen.

Gewünschtes Ergebnis:

- Sie haben Prozesse für das Vorfallmanagement eingerichtet, die auch Analysen nach dem Vorfall beinhalten.
- Sie verfügen über Pläne zur Beobachtbarkeit, um Daten über Ereignisse zu sammeln.
- Anhand dieser Daten können Sie Metriken verstehen und erfassen, die Sie bei der Analyse nach einem Vorfall unterstützen.
- Sie lernen aus Vorfällen, um zukünftige Ergebnisse zu verbessern.

Typische Anti-Muster:

- Sie verwalten einen Anwendungsserver. Ungefähr alle 23 Stunden und 55 Minuten werden alle Ihre aktiven Sitzungen beendet. Sie haben versucht, festzustellen, wo der Fehler auf Ihrem Anwendungsserver liegt. Sie vermuten, dass es sich um ein Netzwerkproblem handeln könnte, das Netzwerkteam zeigt sich jedoch unkooperativ, da es für Ihr Anliegen zu beschäftigt ist. Sie haben keinen vordefinierten Prozess, den Sie befolgen könnten, um Support zu erhalten und die nötigen Informationen zu sammeln, um dem Problem auf den Grund zu gehen.
- Bei Ihrem Workload kam es zu Datenverlust. Dies ist das erste Mal, dass dieses Problem aufgetreten ist, und die Ursache ist nicht klar. Sie entscheiden, dass es nicht wichtig ist, da Sie die Daten wiederherstellen können. Datenverluste beginnen mit größerer Häufigkeit aufzutreten und wirken sich auf Ihre Kunden aus. Dadurch steigt auch der betriebliche Aufwand, wenn Sie die fehlenden Daten wiederherstellen.

Vorteile der Nutzung dieser bewährten Methode:

- Durch vordefinierte Prozesse zur Bestimmung der Komponenten, Bedingungen, Maßnahmen und Ereignisse, die zu einem Vorfall beigetragen haben, können Sie Verbesserungsmöglichkeiten ermitteln.
- Sie können Daten aus der Analyse nach einem Vorfall nutzen, um Verbesserungen vorzunehmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Verwenden Sie einen Prozess zur Ermittlung der Faktoren, die dazu beitragen. Überprüfen Sie alle Vorfälle, die sich auf Kunden auswirken. Erarbeiten Sie ein Verfahren, um die beitragenden Faktoren eines Vorfalls zu ermitteln und zu dokumentieren. Damit können Sie Abhilfemaßnahmen entwickeln, um ein erneutes Auftreten einzudämmen oder gänzlich zu verhindern, und Verfahren für

eine rasche und wirksame Reaktion erstellen. Informieren Sie gegebenenfalls über die Ursachen von Vorfällen und passen Sie die Kommunikation an Ihre Zielgruppe an. Teilen Sie Ihre Erkenntnisse offen innerhalb Ihrer Organisation mit.

Implementierungsschritte

1. Erfassen Sie Metriken wie Bereitstellungsänderungen, Konfigurationsänderungen, Startzeit des Vorfalls, Zeitpunkt des Alarms, Zeitpunkt des Einsatzes, Startzeit der Schadensbegrenzung und Zeitpunkt der Behebung des Vorfalls.
2. Beschreiben Sie wichtige Zeitpunkte auf der Zeitleiste, um die Ereignisse des Vorfalls zu verstehen.
3. Stellen Sie die folgenden Fragen:
   a. Könnten Sie die Zeit bis zur Erkennung verkürzen?
   b. Gibt es Aktualisierungen von Metriken und Alarmen, durch die der Vorfall früher erkannt würde?
   c. Können Sie die Zeit bis zur Diagnose verkürzen?
   d. Gibt es Aktualisierungen Ihrer Reaktions- oder Eskalationspläne, mit denen die richtigen Notfallteams früher eingeschaltet werden könnten?
   e. Können Sie die Zeit bis zur Schadensbegrenzung verkürzen?
   f. Gibt es Runbook- oder Playbook-Schritte, die Sie hinzufügen oder verbessern könnten?
   g. Können Sie zukünftige Vorfälle verhindern?
4. Erstellen Sie Checklisten und Aktionen. Verfolgen und führen Sie alle Aktionen durch.

Aufwand für den Implementierungsplan: mittel

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung
- OPS 4 – Implementieren von Beobachtbarkeit

Zugehörige Dokumente:

- Durchführen einer Analyse nach einem Vorfall im Incident Manager
- Überprüfung der Einsatzbereitschaft

OPS11-BP03 Implementieren von Feedbackschleifen

Feedbackschleifen bieten umsetzbare Einblicke zur Unterstützung der Entscheidungsfindung. Integrieren Sie Feedbackschleifen in Ihre Verfahren und Workloads. Damit können Sie Probleme und Bereiche identifizieren, für die Verbesserungen erforderlich sind. Diese validieren auch Investitionen für Verbesserungen. Diese Feedbackschleifen sind die Grundlage für die kontinuierliche Verbesserung Ihres Workloads.

Feedbackschleifen können in zwei Kategorien unterteilt werden: Sofortiges Feedback und nachträgliche Analyse. Sofortiges Feedback wird durch Prüfung der Leistung und der Ergebnisse betrieblicher Aktivitäten eingeholt. Dieses Feedback kommt von Teammitgliedern, Kunden oder der automatisierten Ausgabe der Aktivität. Sofortiges Feedback kommt von Dingen wie A/B-Tests und der Auslieferung neuer Funktionen und ist für das „Schnell scheitern“-Konzept von entscheidender Bedeutung.

Nachträgliche Analysen werden regelmäßig durchgeführt, um Feedback aus der Überprüfung betrieblicher Ergebnisse und Metriken in der Vergangenheit zu erhalten. Dies geschieht am Ende einer Phase, in regelmäßigem Rhythmus oder nach größeren Releases oder Veranstaltungen. Diese Art von Feedbackschleife validiert Investitionen in Betriebsabläufe oder Ihren Workload. Dies hilft Ihnen beim Messen des Erfolgs und bei der Validierung Ihrer Strategie.

Gewünschtes Ergebnis: Sie nutzen sofortiges Feedback und nachträgliche Analysen für weitere Verbesserungen. Es gibt einen Mechanismus zur Erfassung des Feedbacks von Benutzern und Teammitgliedern. Nachträgliche Analysen identifizieren Trends, die Verbesserungen unterstützen können.

Typische Anti-Muster:

- Sie starten einige Funktionen, haben aber keine Möglichkeit, Feedback von den Kunden dazu zu erhalten.
- Nach einer Investition in verbesserte Betriebsabläufe führen Sie keine nachträgliche Analyse für deren Validierung durch.
- Sie holen das Feedback von Kunden ein, überprüfen dies jedoch nicht regelmäßig.
- Feedbackschleifen führen zu vorgeschlagenen Maßnahmen, werden jedoch nicht in den Softwareentwicklungsprozess einbezogen.
- Kunden erhalten kein Feedback zu Verbesserungen, die sie vorgeschlagen haben.

Vorteile der Nutzung dieser bewährten Methode:

- Sie können vom Kunden aus rückwärts arbeiten, um neue Funktionen zu unterstützen.
- Ihre Organisationskultur kann schneller auf Änderungen reagieren.
- Trends dienen zur Identifizierung von Verbesserungsmöglichkeiten.
- Nachträgliche Analysen validieren in Ihre Workloads und Betriebsabläufe getätigte Investitionen.

Risikostufe, wenn diese bewährte Methode nicht genutzt wird: Hoch

Implementierungsleitfaden

Die Implementierung dieser bewährten Methode bedeutet, dass Sie sofortiges Feedback und nachträgliche Analysen verwenden. Diese Feedbackschleifen erleichtern Verbesserungen. Es gibt zahlreiche Mechanismen für sofortiges Feedback, z. B. Umfragen, Kundenbefragungen oder Feedbackformulare. Ihre Organisation nutzt nachträgliche Analysen auch, um Möglichkeiten für Verbesserungen zu identifizieren und Initiativen zu validieren.

Kundenbeispiel

AnyCompany Retail hat ein Webformular erstellt, über das Kunden Feedback abgeben oder Probleme melden können. Bei der wöchentlichen Scrum-Sitzung evaluiert das Softwareentwicklungsteam das Benutzerfeedback. Das Feedback wird regelmäßig genutzt, um die Weiterentwicklung der Plattform zu steuern. Am Ende jeder Etappe wird eine nachträgliche Analyse durchgeführt, um Punkte zu identifizieren, bei denen Verbesserungsbedarf besteht.

Implementierungsschritte

1. Sofortiges Feedback
   - Sie benötigen einen Mechanismus für den Erhalt von Feedback von Kunden und Teammitgliedern. Ihre betrieblichen Aktivitäten können auch so konfiguriert werden, dass Sie automatisiertes Feedback erhalten.
   - Ihre Organisation benötigt einen Prozess zur Prüfung dieses Feedbacks, zum Feststellen der Verbesserungsbereiche und zur Planung der Verbesserungen.
   - Das Feedback muss in Ihren Softwareentwicklungsprozess integriert werden.
   - Wenn Sie Verbesserungen durchführen, informieren Sie die Personen, die dazu Feedback gegeben haben.
     - Sie können AWS Systems Manager OpsCenter verwenden, um diese Verbesserungen als OpsItems nachzuverfolgen.
2. Nachträgliche Analyse

- Führen Sie nachträgliche Analysen am Ende eines Entwicklungszyklus, in regelmäßigen Abständen oder nach einem größeren Release durch.
- Laden Sie an dem Workload beteiligte Personen zu einer Nachbesprechung ein.
- Erstellen Sie auf einem Whiteboard oder in einem Spreadsheet drei Spalten: Beenden, Starten und Beibehalten.
  - Beenden gilt für alles, mit dem Ihr Team aufhören soll.
  - Starten gilt für Ideen, die ab sofort umgesetzt werden sollen.
  - Beibehalten gilt für Elemente, die weiterhin durchgeführt werden sollen.
- Holen Sie das Feedback aller anwesenden beteiligten Personen ein.
- Priorisieren Sie das Feedback. Weisen Sie allen „Starten“- oder „Beibehalten“-Elementen Aktionen und Beteiligte zu.
- Fügen Sie die Aktionen Ihrem Softwareentwicklungsprozess hinzu und halten Sie die Beteiligten bei Ihren Verbesserungen über den Status auf dem Laufenden.

Aufwand für den Implementierungsplan: Mittel. Zur Implementierung dieser bewährten Methode benötigen Sie ein Verfahren zum Einholen und zur Analyse sofortigen Feedbacks. Dazu müssen Sie auch einen Prozess für die nachträgliche Analyse einrichten.

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP01 Kundenbedürfnisse bewerten: Feedbackschleifen sind ein Mechanismus zum Ermitteln der Anforderungen externer Kunden.
- OPS01-BP02 Bedürfnisse interner Kunden bewerten: Interne Beteiligte können Feedbackschleifen nutzen, um Bedürfnisse und Anforderungen zu kommunizieren.
- OPS11-BP02 Durchführen von Analysen nach Vorfällen: Analysen nach einem Vorfall sind eine wichtige Form nachträglicher Analyse nach Vorfällen.
- OPS11-BP07 Prüfung von Betriebsmetriken: Durch die Prüfung betrieblicher Metriken können Sie Trends und Bereiche für Verbesserungen identifizieren.

Zugehörige Dokumente:

- 7 Fehler, die Sie bei der Einrichtung eines CCOE vermeiden sollten
- Atlassian Team Playbook - Retrospectives

- E-Mail-Definitionen: Feedbackschleifen
- Einrichten von Feedbackschleifen mit der AWS Well-Architected Framework Review
- IBM Garage Methodology – Nachträgliche Analysen
- Investopedia – The PDCS Cycle
- Maximizing Developer Effectiveness von Tim Cochran
- Operations Readiness Reviews (ORR) Whitepaper - Iteration
- TIL CSI - Continual Service Improvement
- Toyota und E-Commerce: Lean bei Amazon

Zugehörige Videos:

- Building Effective Customer Feedback Loops (Aufbau effektiver Kundenfeedbackschleifen)

Zugehörige Beispiele:

- Astuto - Open-Source-Tool für Kundenfeedback
- AWS-Lösungen – QnABot auf AWS
- Fider – Eine Plattform zur Organisation von Kundenfeedback

Zugehörige Services:

- AWS Systems Manager OpsCenter

OPS11-BP04 Wissensmanagement

Das Wissensmanagement hilft den Teammitgliedern, die Informationen zu finden, die sie für ihre Arbeit benötigen. In lernenden Organisationen werden Informationen frei geteilt, was jedem Einzelnen die nötigen Kompetenzen eröffnet. Die Informationen können entdeckt oder durchsucht werden. Die Informationen sind korrekt und auf dem neuesten Stand. Es gibt Mechanismen, um neue Informationen zu erstellen, bestehende Informationen zu aktualisieren und veraltete Informationen zu archivieren. Das gängigste Beispiel für eine Wissensmanagement-Plattform ist ein Content-Management-System wie ein Wiki.

Gewünschtes Ergebnis:

- Teammitglieder haben Zugriff auf zeitnahe, präzise Informationen.

- Die Informationen sind durchsuchbar.
- Es gibt Mechanismen zum Hinzufügen, Aktualisieren und Archivieren von Informationen.

Typische Anti-Muster:

- Es gibt keinen zentralen Wissensspeicher. Die Teammitglieder verwalten ihre eigenen Notizen auf ihren lokalen Rechnern.
- Sie haben ein selbst gehostetes Wiki, aber keine Mechanismen zum Verwalten von Informationen, was dazu führt, dass die Informationen veraltet sind.
- Jemand stellt fest, dass Informationen fehlen, aber es gibt keinen Prozess, um das Hinzufügen dieser Informationen zum Team-Wiki anzustoßen. Er fügt sie selbst hinzu, aber versäumt einen wichtigen Schritt, was zu einem Ausfall führt.

Vorteile der Nutzung dieser bewährten Methode:

- Die Teammitglieder werden gestärkt, weil Informationen frei geteilt werden.
- Neue Teammitglieder werden schneller eingearbeitet, weil die Dokumentation aktuell und durchsuchbar ist.
- Die Informationen sind zeitnah, präzise und umsetzbar.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Das Wissensmanagement ist eine wichtige Facette von lernenden Organisationen. Zunächst benötigen Sie ein zentrales Repository, in dem Sie Ihr Wissen speichern (z. B. ein selbst gehostetes Wiki). Sie müssen Prozesse entwickeln, um Wissen hinzuzufügen, zu aktualisieren und zu archivieren. Entwickeln Sie Standards für das, was dokumentiert werden soll, und lassen Sie alle Beteiligten dazu beitragen.

Kundenbeispiel

AnyCompany Retail hostet ein internes Wiki, in dem das gesamte Wissen gespeichert wird. Die Teammitglieder werden ermutigt, die Wissensdatenbank im Rahmen ihrer täglichen Arbeit zu ergänzen. Ein funktionsübergreifendes Team bewertet vierteljährlich, welche Seiten am wenigsten aktualisiert werden, und entscheidet, ob sie archiviert oder aktualisiert werden sollen.

Implementierungsschritte

1. Beginnen Sie damit, das Content-Management-System zu bestimmen, in dem das Wissen gespeichert werden soll. Holen Sie die Zustimmung der Stakeholder in Ihrer Organisation ein.
   a. Wenn Sie kein vorhandenes Content-Management-System haben, können Sie ein selbst gehostetes Wiki oder ein Versionsverwaltungssystem als Ausgangspunkt verwenden.
2. Entwickeln Sie Runbooks für das Hinzufügen, Aktualisieren und Archivieren von Informationen. Informieren Sie Ihr Team über diese Prozesse.
3. Bestimmen Sie, welches Wissen im Content-Management-System gespeichert werden soll. Beginnen Sie mit den täglichen Aktivitäten (Runbooks und Playbooks), die die Teammitglieder ausführen. Arbeiten Sie mit Stakeholdern zusammen, um Prioritäten für das hinzuzufügende Wissen festzulegen.
4. Arbeiten Sie in regelmäßigen Abständen mit Stakeholdern zusammen, um veraltete Informationen zu identifizieren und sie zu archivieren oder auf den neuesten Stand zu bringen.

Grad des Aufwands für den Implementierungsplan: mittel. Wenn Sie kein vorhandenes Content-Management-System haben, können Sie ein selbst gehostetes Wiki oder ein Dokumenten-Repository mit Versionsverwaltung einrichten.

Ressourcen

Zugehörige bewährte Methoden:

- OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen - Das Wissensmanagement erleichtert den Austausch von Informationen über gewonnene Erkenntnisse.

Zugehörige Dokumente:

- Atlassian – Wissensmanagement

Zugehörige Beispiele:

- DokuWiki
- Gollum
- MediaWiki
- Wiki.js

OPS11-BP05 Definieren von Verbesserungsfaktoren

Identifizieren Sie Verbesserungsmöglichkeiten, damit Sie Chancen basierend auf Daten und Feedback-Schleifen bewerten und priorisieren können. Erkunden Sie Verbesserungsmöglichkeiten in Ihren Systemen und Prozessen und automatisieren Sie bei Bedarf.

Gewünschtes Ergebnis:

- Sie verfolgen Daten aus Ihrer gesamten Umgebung.
- Sie korrelieren Ereignisse und Aktivitäten mit Geschäftsergebnissen.
- Sie können Umgebungen und Systeme vergleichen und gegenüberstellen.
- Sie führen einen detaillierten Aktivitätsverlauf Ihrer Bereitstellungen und Ergebnisse.
- Sie sammeln Daten, um Ihren Sicherheitsstatus zu stärken.

Typische Anti-Muster:

- Sie sammeln Daten aus Ihrer gesamten Umgebung, korrelieren jedoch keine Ereignisse und Aktivitäten.
- Sie sammeln detaillierte Daten aus Ihrem gesamten Bestand, was die Aktivität und Kosten von AWS CloudTrail und Amazon CloudWatch in die Höhe treibt. Sie ziehen jedoch keinen sinnvollen Nutzen aus diesen Daten.
- Bei der Definition von Verbesserungsfaktoren berücksichtigen Sie nicht die Geschäftsergebnisse.
- Sie messen nicht die Auswirkungen neuer Features.

Vorteile der Nutzung dieser bewährten Methode:

- Sie minimieren die Auswirkungen ereignisbasierter Motivationen oder emotionaler Investitionen, indem Sie Verbesserungskriterien festlegen.
- Sie reagieren auf alle, nicht nur technische Geschäftsereignisse.
- Sie messen Ihre Umgebung, um Verbesserungsbereiche zu identifizieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

- Kenntnis der Verbesserungsfaktoren: Sie sollten ein System nur dann ändern, wenn das gewünschte Ergebnis auch unterstützt wird.
  - Gewünschte Fähigkeiten: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten die gewünschten Features und Fähigkeiten.
    - Neuerungen bei AWS
  - Nicht akzeptable Probleme: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten nicht akzeptable Probleme, Fehler und Schwachstellen. Informieren Sie sich über Dimensionierungsoptionen und suchen Sie nach Optimierungsmöglichkeiten.
    - Aktuelle AWS-Sicherheitsmitteilungen
    - AWS Trusted Advisor
    - Cloud Intelligence Dashboards
  - Compliance-Anforderungen: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten, welche Updates und Änderungen erforderlich sind, um Vorschriften bzw. Richtlinien einzuhalten oder weiterhin den Support eines Drittanbieters nutzen zu können.
    - AWS-Compliance
    - AWS-Compliance-Programme
    - Aktuelle Neuigkeiten zur AWS-Compliance

Ressourcen

Zugehörige bewährte Methoden:

- OPS01 Organisationsprioritäten
- OPS02 Beziehungen und Eigentümerschaft
- OPS04-BP01 Ermitteln wichtiger Key Performance Indicators
- OPS08 Nutzung der Workload-Beobachtbarkeit
- OPS09 Grundlegendes zum betrieblichen Status
- OPS11-BP03 Implementieren von Feedback-Schleifen

Zugehörige Dokumente:

- Amazon Athena

- QuickSight
- AWS-Compliance
- Aktuelle Neuigkeiten zur AWS-Compliance
- AWS-Compliance-Programme
- AWS Glue
- Aktuelle AWS-Sicherheitsmitteilungen
- AWS Trusted Advisor
- Exportieren Ihrer Protokolldaten zu Amazon S3
- Neuerungen bei AWS
- Die Anforderungen bei kundenorientierter Innovation
- Digitale Transformation: Hype oder strategische Notwendigkeit?

Zugehörige Videos:

- AWS re:Invent 2023 – Verbessern der betrieblichen Effizienz und Belastbarkeit mit Support (SUP310)

OPS11-BP06 Prüfen von Erkenntnissen

Überprüfen Sie Ihre Analyseergebnisse und Reaktionen mit fachbereichsübergreifenden Teams und Geschäftsverantwortlichen. Schaffen Sie mithilfe dieser Prüfungen ein allgemeines Verständnis, ermitteln Sie weitere Auswirkungen und legen Sie einen Maßnahmenkatalog fest. Passen Sie die Reaktionen bei Bedarf an.

Gewünschte Ergebnisse:

- Sie überprüfen regelmäßig Erkenntnisse mit Geschäftsbereichsleitern. Geschäftsbereichsleiter liefern zusätzlichen Kontext zu neu gewonnenen Erkenntnissen.
- Sie überprüfen Erkenntnisse und bitten um Feedback von Fachkollegen, und Sie teilen Ihre Erkenntnisse mit allen Teams.
- Sie veröffentlichen Daten und Erkenntnisse, die andere technische und Geschäftsteams überprüfen können. Sie entwickeln aus Ihren Erkenntnisse neue Methoden für andere Abteilungen.
- Sie fassen neue Erkenntnisse zusammen und besprechen sie mit Führungskräften. Führungskräfte nutzen neue Erkenntnisse, um die Strategie zu definieren.

Typische Anti-Muster:

- Sie veröffentlichen ein neues Feature. Dieses Feature verändert das Verhalten einiger Ihrer Kunden. Ihre Beobachtbarkeit berücksichtigt diese Änderungen nicht. Sie quantifizieren die Vorteile dieser Änderungen nicht.
- Sie veröffentlichen ein neues Update und vernachlässigen die Aktualisierung Ihres CDN. Der CDN-Cache ist nicht mehr mit der aktuellen Version kompatibel. Sie messen den Prozentsatz der Anforderungen mit Fehlern. Alle Ihre Benutzer melden HTTP 400-Fehler bei der Kommunikation mit Backend-Servern. Sie untersuchen die Kundenfehler und stellen fest, dass Sie die Zeit verschwendet haben, weil Sie die falsche Dimension gemessen haben.
- Ihr Service Level Agreement sieht eine Verfügbarkeit von 99,9 % vor, und Ihr Wiederherstellungszeitpunkt liegt bei vier Stunden. Der Servicebesitzer behauptet, dass das System keine Ausfallzeiten hat. Sie implementieren eine teure und komplexe Replikationslösung, die Zeit und Geld verschwendet.

Vorteile der Nutzung dieser bewährten Methode:

- Durch die Prüfung von Erkenntnissen zusammen mit Geschäftsinhabern und Fachexperten bauen Sie ein gemeinsames Verständnis auf und sorgen effektiver für Verbesserungen.
- Sie entdecken verborgene Probleme und berücksichtigen sie bei zukünftigen Entscheidungen.
- Ihr Fokus verlagert sich von technischen Ergebnissen hin zu Geschäftsergebnissen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

- Prüfen von Erkenntnissen: Wenden Sie sich an die Geschäftsinhaber und Fachexperten, um sicherzustellen, dass die Bedeutung der von Ihnen gesammelten Daten allgemein verstanden und vereinbart ist. Ermitteln Sie zusätzliche Bedenken, potenzielle Auswirkungen und bestimmen Sie eine Vorgehensweise.

Ressourcen

Zugehörige bewährte Methoden:

- OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken
- OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt

- OPS11-BP03 Implementieren von Feedback-Schleifen

Zugehörige Dokumente:

- Planung eines Cloud-Kompetenzzentrums (CCOE)

Zugehörige Videos:

- Aufbau von Beobachtbarkeit zur Erhöhung der Resilienz

OPS11-BP07 Prüfung von Betriebsmetriken

Führen Sie regelmäßig teamübergreifend mit Teilnehmern aus verschiedenen Unternehmensbereichen nachträgliche Analysen der operationsspezifischen Metriken durch. Ermitteln Sie mithilfe dieser Prüfungen Verbesserungspotenziale sowie mögliche Maßnahmen und teilen Sie diese Erkenntnisse auch anderen mit. Berücksichtigen Sie bei Ihrer Suche nach Verbesserungsmöglichkeiten all Ihre Umgebungen (z. B. Entwicklungs-, Test- und Produktionsumgebung).

Gewünschtes Ergebnis:

- Sie überprüfen häufig Metriken, die sich auf das Geschäft auswirken.
- Sie erkennen und überprüfen Anomalien mithilfe Ihrer Beobachtbarkeitsfunktionen.
- Sie verwenden Daten, um die Erreichung von Geschäftsergebnissen und Zielen zu unterstützen.

Typische Anti-Muster:

- Ihr Wartungsfenster unterbricht eine wichtige Verkaufsaktion. Das Unternehmen weiß weiterhin nicht, dass es ein Standard-Wartungsfenster gibt, das verzögert werden könnte, wenn sich andere wichtige Ereignisse auf das Geschäft auswirken.
- Sie hatten einen längeren Ausfall, weil in Ihrer Organisation häufig eine veraltete Bibliothek verwendet wird. Inzwischen sind Sie zu einer unterstützten Bibliothek migriert. Die anderen Teams in Ihrer Organisation wissen nicht, dass diese Gefahr besteht.
- Sie überprüfen nicht regelmäßig die Einhaltung der Kunden-SLAs. Sie laufen Gefahr, die mit Kunden vereinbarten SLAs nicht zu erfüllen. Es drohen Geldstrafen bei der Nichteinhaltung von mit Kunden vereinbarten SLAs.

Vorteile der Nutzung dieser bewährten Methode:

- Indem Sie sich regelmäßig treffen, um Betriebsmetriken, Ereignisse und Vorfälle zu überprüfen, sorgen Sie für ein gemeinsames Verständnis aller Teams.
- Ihr Team trifft sich regelmäßig, um Metriken und Vorfälle zu überprüfen und auf diese Weise Maßnahmen gegen Risiken zu ergreifen und Kunden-SLAs zu erkennen.
- Sie teilen Ihre gewonnenen Erkenntnisse, die Daten zur Priorisierung und zur gezielten Verbesserung der Geschäftsergebnisse liefern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

- Führen Sie regelmäßig teamübergreifend mit Teilnehmern aus verschiedenen Unternehmensbereichen nachträgliche Analysen der operationsspezifischen Metriken durch.
- Binden Sie alle Stakeholder, einschließlich der Teams aus den Bereichen Betriebswirtschaft, Entwicklung und Operationen, ein, indem Sie Ihre Erkenntnisse aus dem sofortigen Feedback und der nachträglichen Analyse und gewonnene Erkenntnisse austauschen.
- Machen Sie sich deren Erkenntnisse zunutze, um Verbesserungspotenziale und mögliche Maßnahmen ausfindig zu machen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS08-BP05 Erstellen von Dashboards
- OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen
- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen

Zugehörige Dokumente:

- Amazon CloudWatch
- Referenzinformationen zu Metriken und Dimensionen von Amazon CloudWatch
- Veröffentlichen von benutzerdefinierten Metriken
- Verwendung von Amazon CloudWatch-Metriken

- Dashboards und Visualisierungen mit CloudWatch

OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen

Dokumentieren Sie die Erkenntnisse aus den betrieblichen Aktivitäten und geben Sie diese weiter, damit Sie sie sowohl intern als auch teamübergreifend nutzen können. Die Erkenntnisse Ihres Teams sollten Sie an andere in Ihrer Organisation weitergeben, damit alle davon profitieren. Teilen Sie Informationen und Ressourcen, um vermeidbare Fehler zu verhindern und Entwicklungsbemühungen zu unterstützen, und konzentrieren Sie sich auf die Bereitstellung der angestrebten Features.

Definieren Sie mithilfe von AWS Identity and Access Management (IAM) Berechtigungen, die den gesteuerten Zugriff auf die Ressourcen ermöglichen, die Sie innerhalb von Konten und kontenübergreifend freigeben möchten.

Gewünschtes Ergebnis:

- Anschließend sollten Sie versionsgesteuerte Repositories verwenden, um Anwendungsbibliotheken, skriptbasierte Verfahren, Verfahrens- und andere Systemdokumentationen freizugeben.
- Sie teilen Ihre Infrastrukturstandards als versionskontrollierte AWS CloudFormation-Vorlagen.
- Sie überprüfen die Erkenntnisse, die Sie teamübergreifend gelernt haben.

Typische Anti-Muster:

- Sie erlitten einen längeren Ausfall, weil Ihre Organisation häufig eine fehlerhafte Bibliothek verwendet. Seitdem sind Sie zu einer zuverlässigen Bibliothek migriert. Die anderen Teams in Ihrer Organisation wissen nicht, dass diese Gefahr besteht. Niemand dokumentiert und teilt die Erfahrung mit dieser Bibliothek, und sie sind sich des Risikos nicht bewusst.
- Sie haben einen Grenzfall in einem intern gemeinsam genutzten Microservice ermittelt, der dazu führt, dass Sitzungen unterbrochen werden. Sie rufen den Service jetzt anders auf, um diesen Grenzfall zu vermeiden. Die anderen Teams in Ihrer Organisation wissen nicht, dass diese Gefahr besteht.
- Sie haben eine Möglichkeit gefunden, die Anforderungen an die CPU-Auslastung eines Ihrer Microservices deutlich zu reduzieren. Sie wissen nicht, ob andere Teams auch von diesem Verfahren profitieren könnten.

Vorteile der Etablierung dieser bewährten Methode: Teilen Sie Ihre Erfahrungen mit, um Verbesserungen zu unterstützen und den Nutzen aus Erfahrungen zu maximieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

- Dokumentieren und Weitergeben von Erkenntnissen: Implementieren Sie Verfahren zur Dokumentation der aus der Durchführung von betrieblichen Aktivitäten und nachträglichen Analysen gewonnenen Erkenntnisse, damit auch andere Teams davon profitieren.
- Weitergeben von Erkenntnissen: Nutzen Sie Verfahren für den teamübergreifenden Austausch gewonnener Erkenntnisse und zugehöriger Artefakte. Veröffentlichen Sie beispielsweise aktualisierte Verfahren, Richtlinien, Governance und bewährte Methoden in einem allgemein zugänglichen Wiki. Teilen Sie Skripte, Code und Bibliotheken über ein gemeinsames Repository.
  - Delegieren des Zugriffs auf Ihre AWS-Umgebung
  - Freigeben eines AWS CodeCommit-Repositorys

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt
- OPS05-BP01 Verwendung einer Versionskontrolle
- OPS05-BP06 Gemeinsame Design-Standards
- OPS11-BP03 Implementieren von Feedback-Schleifen
- OPS11-BP07 Prüfung von Betriebsmetriken

Zugehörige Dokumente:

- Reduzieren Sie Projektverzögerungen mit einer Docs-as-Code-Lösung

Zugehörige Videos:

- Delegieren des Zugriffs auf Ihre AWS-Umgebung
- Wie Support Sie unterstützt | Vorstellung der Tabletop-Übungen zum Vorfallmanagement

OPS11-BP09 Einplanen von Zeit für Verbesserungen

Reservieren Sie Zeit und Ressourcen innerhalb Ihrer Prozesse, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen.

Gewünschtes Ergebnis:

- Sie können temporäre Duplikate von Umgebungen erstellen. Das senkt die Risiken, den Aufwand und Kosten, die mit dem Experimentieren und Testen verbunden sind.
- Diese duplizierten Umgebungen können Sie nutzen, um die aus Ihren Analysen gezogenen Rückschlüsse zu testen, Verbesserungen zu entwickeln und geplante Verbesserungen zu testen.
- Sie führen GameDays durch und verwenden Fault Injection Service (FIS), um die Kontrollen und den Integritätsschutz bereitzustellen, die Teams benötigen, um Experimente in einer produktionsähnlichen Umgebung durchzuführen.

Typische Anti-Muster:

- Es besteht ein bekanntes Leistungsproblem auf Ihrem Anwendungsserver. Es wird im Backlog hinter jeder geplanten Feature-Implementierung priorisiert. Bleibt die Rate der hinzugefügten geplanten Features konstant, wird das Leistungsproblem niemals behoben.
- Genehmigen Sie den Administratoren und Entwicklern, dass sie ihre Überstunden zur Auswahl und Implementierung von Verbesserungen nutzen können, um kontinuierliche Verbesserungen zu unterstützen. Es werden niemals Verbesserungen vorgenommen.
- Die Betriebsabnahme ist abgeschlossen, und Sie testen die betrieblichen Praktiken nicht erneut.

Vorteile der Nutzung dieser bewährten Methoden: Indem Sie Zeit und Ressourcen in Ihre Prozesse investieren, ermöglichen Sie kontinuierliche, schrittweise Verbesserungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

- Einplanen von Zeit für Verbesserungen: Reservieren Sie Zeit und Ressourcen innerhalb Ihrer Prozesse, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen.
- Implementieren Sie Änderungen, die zu Verbesserungen führen sollen, und beurteilen Sie deren Ergebnisse.

- Versuchen Sie alternative Vorgehensweisen, wenn die Ergebnisse die Ziele nicht erfüllen und die Verbesserung immer noch Priorität hat.
- Simulieren Sie Produktionsworkloads durch GameDays, und nutzen Sie die Erkenntnisse aus diesen Simulationen, um sich zu verbessern.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP08 Verwenden mehrerer Umgebungen

Zugehörige Videos:

- AWSre:Invent 2023 – Verbessern Sie die Resilienz Ihrer Anwendungen mit AWS Fault Injection Service

Sicherheit

SEC 1. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper der Säule für Sicherheit.

Bereiche für bewährte Methoden

- Sicherheitsgrundlagen
- Identity and Access Management
- Erkennung
- Schutz der Infrastruktur
- Datenschutz
- Vorfallsreaktion
- Anwendungssicherheit

Sicherheitsgrundlagen

Frage

- SEC 1. Wie können Sie Ihren Workload sicher betreiben?

SEC 1. Wie können Sie Ihren Workload sicher betreiben?

SEC 2. Nutzen Sie Anforderungen und Prozesse, die Sie in Operational Excellence definiert haben, auf Organisations- und Workload-Ebene, und wenden Sie sie auf alle Bereiche an. Bleiben Sie auf dem Laufenden mit AWS- und Branchenempfehlungen sowie Bedrohungsinformationen, um Ihr Bedrohungsmodell und Ihre Kontrollziele weiterzuentwickeln. Die Automatisierung von Sicherheitsprozessen, Tests und Validierung ermöglicht es Ihnen, Ihre operativen Abläufe zu skalieren.

Bewährte Methoden

- SEC01-BP01 Trennen von Workloads mithilfe von Konten
- SEC01-BP02 Schutz des Konto-Root-Benutzers und seiner Eigenschaften
- SEC01-BP03 Identifizieren und Validieren von Kontrollzielen
- SEC01-BP04 Sicherstellen der Aktualität von Informationen zu Sicherheitsbedrohungen
- SEC01-BP05 Verringern des Umfangs der Sicherheitsverwaltung
- SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen
- SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells
- SEC01-BP08 Regelmäßiges Bewerten und Implementieren neuer Sicherheitsservices und -features

SEC01-BP01 Trennen von Workloads mithilfe von Konten

Sorgen Sie mit einer Mehrkonten-Strategie für wirksamen Integritätsschutz und Isolierungen zwischen Umgebungen (etwa Produktion, Entwicklung und Test) sowie Workloads. Die Trennung auf Kontoebene wird nachdrücklich angeraten, da diese für die wirksame Isolierung für Sicherheits-, Fakturierungs- und Zugriffszwecke sorgt.

Gewünschtes Ergebnis: eine Kontostruktur, die Cloud-Operationen, nicht zusammengehörige Workloads und Umgebungen in separaten Konten voneinander isoliert, sodass die Sicherheit in der gesamten Cloud-Infrastruktur verbessert wird.

Typische Anti-Muster:

- Platzierung mehrerer nicht zusammengehöriger Workloads mit unterschiedlicher Datensensitivität in einem einzigen Konto

- schlecht definierte Organizational Unit (OU, Organisationseinheit)-Struktur

Vorteile der Nutzung dieser bewährten Methode:

- geringere Auswirkungen bei versehentlichen Zugriffen auf einen Workload
- zentrale Verwaltung des Zugriffes auf AWS-Services, Ressourcen und Regionen
- Wahrung der Sicherheit der Cloud-Infrastruktur durch Richtlinien und die zentralisierte Verwaltung von Sicherheitsservices
- automatisierte Kontoerstellung und Wartungsprozesse
- zentralisierte Prüfung Ihrer Infrastruktur auf Compliance- und regulatorische Anforderungen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

AWS-Konten bieten eine Sicherheitsisolierungsgrenze zwischen Workloads oder Ressourcen, die auf unterschiedlichen Sensitivitätsstufen operieren. AWS bietet Tools, mit denen Sie Ihre umfangreichen Cloud-Workloads über eine Mehrkonten-Strategie verwalten und so die Isolierungsgrenze nutzen können. Für Erläuterungen der Konzepte, Muster und der Implementierung einer Mehrkonten-Strategie auf AWS siehe Organisation Ihrer AWS-Umgebung mit mehreren Konten.

Wenn Sie mehrere AWS-Konten zentral verwalten, sollten Ihre Konten in einer gemäß den Ebenen der Organisationseinheiten (OUs) definierten Hierarchie organisiert sein. Dadurch können Sicherheitskontrollen anhand der OUs und der Mitgliedskonten organisiert und auf diese angewendet werden, was eine konsistente präventive Kontrolle der Mitgliedskonten in der Organisation ermöglicht. Die Sicherheitskontrollen werden weitergegeben, sodass Sie nach verfügbaren Berechtigungen für Mitgliedskonten auf unteren Ebenen der OU-Hierarchie filtern können. Ein gutes Design macht sich diese Weitergabe zunutze, um die Anzahl und die Komplexität der Sicherheitsrichtlinien, die für die erwünschten Sicherheitskontrollen für jedes Mitgliedskonto erforderlich sind, zu reduzieren.

AWS Organizations und AWS Control Tower sind zwei Services, mit denen Sie diese Mehrkontenstruktur in Ihrer AWS-Umgebung implementieren und verwalten können. AWS Organizations ermöglicht die Organisation von Konten in einer von einer oder mehreren Ebenen von OUs definierten Hierarchie, wobei jede OU eine Anzahl von Mitgliedskonten enthält. Service-Kontrollrichtlinien (SCPs) ermöglichen einem Organisationsadministrator die Einrichtung detaillierter präventiver Kontrollen für Mitgliedskonten und AWS Config kann verwendet werden, um proaktive

und erkennende Kontrollen für Mitgliedskonten zu aktivieren. Viele AWS-Services lassen sich in AWS Organizations integrieren und bieten so delegierte administrative Kontrollen und führen servicespezifische Aufgaben für alle Mitgliedskonten in der Organisation durch.

Über AWS Organizations hinaus ermöglicht AWS Control Tower die Einrichtung bewährter Methoden mit einem Klick für eine Mehrkonten-AWS-Umgebung mit einer Landing Zone. Die Landing Zone ist der Einstiegspunkt für die Mehrkonten-Umgebung, eingerichtet von Control Tower. Control Tower bietet verschiedene Vorteile gegenüber AWS Organizations. Hier sind drei Vorteile, die die Kontoverwaltung verbessern:

- integrierter verpflichtender Integritätsschutz, der automatisch auf für die Organisation zugelassene Konten angewendet wird
- optionaler Integritätsschutz, der für einen bestimmten Satz von OUs aktiviert und deaktiviert werden kann
- AWS Control Tower Account Factory bietet eine automatisierte Bereitstellung von Konten mit vorab genehmigten Baselines und Konfigurationsoptionen innerhalb Ihrer Organisation.

Implementierungsschritte

1. Entwurf einer OU-Struktur: Eine korrekt gestaltete OU-Struktur reduziert den Verwaltungsaufwand für die Erstellung und Wahrung von Service-Kontrollrichtlinien und anderen Sicherheitskontrollen. Ihre OU-Struktur sollte an Ihre geschäftlichen Anforderungen, die Sensitivität der Daten und die Workload-Struktur angepasst sein.
2. Erstellen einer Landing Zone für Ihre Mehrkonten-Umgebung: Eine Landing Zone bietet eine konsistente Sicherheits- und Infrastrukturbasis, von der aus Ihre Organisation Workloads schnell entwickeln, starten und bereitstellen kann. Sie können eine individuell erstellte Landing Zone oder AWS Control Tower für die Orchestrierung Ihrer Umgebung verwenden.
3. Einrichtung von Integritätsschutz: Implementieren Sie konsistenten Integritätsschutz für Ihre Umgebung über Ihre Landing Zone. AWS Control Tower bietet eine Liste verpflichtender und optionaler Kontrollen, die bereitgestellt werden können. Verpflichtende Kontrollen werden automatisch bereitgestellt, wenn Control Tower implementiert wird. Überprüfen Sie die Liste nachdrücklich empfohlener sowie optionaler Kontrollen und implementieren Sie diejenigen, die Ihren Anforderungen entsprechen.
4. Einschränken des Zugriffs auf neu hinzugefügte Regionen: Für neue AWS-Regionen werden IAM-Ressourcen, z. B. Benutzer und Rollen, nur an die von Ihnen angegebenen Regionen weitergegeben. Dieser Vorgang kann über die Konsole durchgeführt werden, wenn Sie Control

Tower verwenden, oder durch die Anpassung von IAM-Berechtigungsrichtlinien in AWS Organizations.

5. Erwägen der Verwendung von AWSCloudFormation StackSets: StackSets helfen dabei, Ressourcen wie IAM-Richtlinien, -Rollen und -Gruppen aus einer genehmigten Vorlage in verschiedenen AWS-Konten und Regionen bereitzustellen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter

Zugehörige Dokumente:

- AWS Control Tower
- AWS Security Audit Guidelines (Richtlinien zur AWS-Sicherheitsprüfung)
- IAM Best Practices (Bewährte Methoden für IAM)
- Use CloudFormation StackSets to provision resources across multiple AWS-Konten and regions (Verwendung von CloudFormation StackSets zur Bereitstellung von Ressourcen für mehrere AWS-Konten und Regionen)
- Organizations FAQ (Häufig gestellte Fragen zu Organisationen)
- AWS Organizations terminology and concepts (AO-Terminologie und -Konzepte)
- Best Practices for Service Control Policies in an AWS Organizations Multi-Account Environment (Bewährte Methoden für Service-Kontrollrichtlinien in einer AO-Mehrkonten-Umgebung)
- AWS Account Management Reference Guide (Referenz zur Verwaltung von AWS-Konten)
- Organizing Your AWS Environment Using Multiple Accounts (Organisieren der AWS-Umgebung mithilfe mehrerer Konten)

Zugehörige Videos:

- Enable AWS adoption at scale with automation and governance(AWS-Übernahme in großem Umfang mit Automatisierung und Governance)
- Security Best Practices the Well-Architected Way (Bewährte Sicherheitsmethoden mit durchdachter Architektur)

- Building and Governing Multiple Accounts using AWS Control Tower (Aufbau und Verwaltung mehrerer Konten mit AWS Control Tower)
- Enable Control Tower for Existing Organizations (Aktivierung von Control Tower für bestehende Organisationen)

Zugehörige Workshops:

- Control Tower Immersion Day

SEC01-BP02 Schutz des Konto-Root-Benutzers und seiner Eigenschaften

Der Root-Benutzer ist in einem AWS-Konto der Benutzer mit den meisten Berechtigungen und vollständigem administrativem Zugriff auf alle Ressourcen in dem Konto und kann in manchen Fällen nicht von Sicherheitsrichtlinien eingeschränkt werden. Die Deaktivierung des programmatischen Zugriffs auf den Root-Benutzer, die Einrichtung geeigneter Kontrollen für den Root-Benutzer und das Vermeiden der routinemäßigen Verwendung des Root-Benutzers senken die Risiken einer unbeabsichtigten Offenlegung der Anmeldeinformationen des Root-Benutzers und daraus resultierender ernsthafter Probleme für die Cloud-Umgebung.

Gewünschtes Ergebnis: Der Schutz des Root-Benutzers hilft dabei, die Gefahr zu verringern, dass versehentliche oder beabsichtigte Schäden durch den Missbrauch der Anmeldeinformationen des Root-Benutzers entstehen. Die Einrichtung erkennender Kontrollen kann auch für die Benachrichtigung der richtigen Personen sorgen, wenn Aktionen unter Verwendung des Root-Benutzers durchgeführt werden.

Typische Anti-Muster:

- Verwendung des Root-Benutzers für andere Aufgaben als die wenigen, für die Root-Benutzer-Anmeldeinformationen erforderlich sind
- Versäumnis, Notfallpläne regelmäßig zu testen, um das Funktionieren kritischer Infrastrukturen, Prozesse und des Personals während eines Notfalls zu überprüfen.
- ausschließliche Berücksichtigung des typischen Kontoanmeldungsprozesses und keine Berücksichtigung alternativer Kontowiederherstellungsverfahren
- keine Behandlung von DNS, E-Mail-Servern und Telefonanbietern als Teil des kritischen Sicherheitsperimeters, da diese in den Kontowiederherstellungsabläufen verwendet werden

Vorteile der Nutzung dieser bewährten Methode: Der Schutz des Zugriffs auf den Root-Benutzer stärkt das Vertrauen dazu, dass Aktionen in Ihrem Konto kontrolliert und überwacht werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

AWS bietet zahlreiche Tools für den Schutz Ihres Kontos. Da einige dieser Maßnahmen aber nicht standardmäßig aktiviert sind, müssen Sie sie selbst implementieren. Betrachten Sie diese Empfehlungen als grundlegende Schritte für den Schutz Ihres AWS-Konto. Bei der Implementierung dieser Schritte ist es wichtig, dass Sie einen Prozess für die kontinuierliche Prüfung und Überwachung der Sicherheitskontrollen einrichten.

Wenn Sie ein AWS-Konto anlegen, beginnen Sie mit einer Identität, mit der Sie auf alle mit dem Konto verbundenen AWS-Services und -Ressourcen zugreifen können. Diese Identität wird als der Root-Benutzer des AWS-Konto bezeichnet. Sie können sich mit der E-Mail-Adresse und dem Passwort, die bei der Konto-Erstellung verwendet wurden, als Root-Benutzer anmelden. Da der AWS-Root-Benutzer erweiterte Zugriffsrechte hat, müssen Sie die Verwendung des AWS-Root-Benutzers auf die Aufgaben beschränken, für die er ausdrücklich erforderlich ist. Die Anmeldeinformationen des Root-Benutzers müssen sehr gut geschützt werden, und für den Root-Benutzer des AWS-Konto sollte immer die Multi-Faktor-Authentifizierung (MFA) aktiviert sein.

Zusätzlich zum normalen Authentifizierungsablauf bei der Anmeldung als Root-Benutzer mit einem Benutzernamen, Passwort und einem Gerät zur Multi-Faktor-Authentifizierung (MFA) gibt es Kontowiederherstellungsabläufe für die Anmeldung Ihres AWS-Konto als Root-Benutzer mit Zugriff auf die mit Ihrem Konto verbundene E-Mail-Adresse und die Telefonnummer. Daher ist es ebenso wichtig, das E-Mail-Konto des Root-Benutzers, an das die Wiederherstellungs-E-Mail gesendet wird, und die mit dem Konto verknüpfte Telefonnummer zu sichern. Denken Sie auch an mögliche zirkuläre Abhängigkeiten, bei denen die zum Root-Benutzer gehörende E-Mail-Adresse auf E-Mail-Servern oder DNS (Domain Name Service)-Ressourcen von demselben AWS-Konto gehostet wird.

Bei Verwendung von AWS Organizations gibt es mehrere AWS-Konten, die jeweils einen Root-Benutzer haben. Ein Konto fungiert als Verwaltungskonto und mehrere Ebenen von Mitgliedskonten können dann darunter hinzugefügt werden. Priorisieren Sie den Schutz des Root-Benutzers Ihres Verwaltungskontos und kümmern Sie sich dann um diejenigen der Mitgliedskonten. Die Strategie zum Schutz des Root-Benutzers Ihres Verwaltungskontos kann sich von der für die Root-Benutzer der Mitgliedskonten unterscheiden und Sie können präventive Sicherheitskontrollen für die Root-Benutzer Ihrer Mitgliedskonten einrichten.

Implementierungsschritte

Die folgenden Implementierungsschritte werden für die Einrichtung der Kontrollen für den Root-Benutzer empfohlen. Gegebenenfalls verweisen die Empfehlungen auf CIS AWS Foundations Benchmark, Version 1.4.0. Konsultieren Sie zusätzlich zu diesen Schritten die Richtlinien zu bewährten Methoden für AWS für den Schutz Ihres AWS-Konto und Ihrer Ressourcen.

Präventive Kontrollen

1. Richten Sie korrekte Kontaktinformationen für das Konto ein.
   a. Diese Informationen werden für die Abläufe zur Wiederherstellung verlorener Passwörter, verlorener MFA-Gerätekonten und für die kritische sicherheitsrelevante Kommunikation mit Ihrem Team verwendet.
   b. Verwenden Sie eine von ihrer Unternehmensdomain gehostete E-Mail-Adresse, vorzugsweise eine Verteilerliste, als E-Mail-Adresse des Root-Benutzers. Die Verwendung einer Verteilerliste anstelle einer einzelnen E-Mail-Adresse sorgt für zusätzliche Redundanz und Kontinuität beim Zugriff auf das Root-Konto über längere Zeiträume hinweg.
   c. Die in den Kontaktinformationen angegebene Telefonnummer sollte eine für diesen Zweck speziell eingerichtete und sichere Telefonnummer sein. Diese Telefonnummer sollte nicht eingetragen sein oder an andere weitergegeben werden.
2. Erstellen Sie keine Zugriffsschlüssel für den Root-Benutzer. Wenn Zugriffsschlüssel vorhanden sind, entfernen Sie diese (CIS 1.4).
   a. Entfernen Sie alle langfristigen programmatischen Anmeldeinformationen (Zugriffs- und geheime Schlüssel) für den Root-Benutzer.
   b. Wenn bereits Zugriffsschlüssel für den Root-Benutzer vorhanden sind, sollten Prozesse, die diese Schlüssel verwenden, so umgestaltet werden, dass sie temporäre Zugriffsschlüssel von einer AWS Identity and Access Management (IAM)-Rolle verwenden; löschen Sie dann die Zugriffsschlüssel des Root-Benutzers.
3. Ermitteln Sie, ob Sie Anmeldeinformationen für den Root-Benutzer speichern müssen.
   a. Wenn Sie AWS Organizations zum Erstellen neuer Mitgliedskonten verwenden, wird das ursprüngliche Passwort für den Root-Benutzer in neuen Mitgliedskonten auf einen zufälligen Wert festgelegt, der Ihnen nicht angezeigt wird. Erwägen Sie die Nutzung der Passwortrücksetzung von Ihrem AWS-Organization-Verwaltungskonto, um bei Bedarf Zugriff auf das Mitgliedskonto zu erhalten.

   b. Für Standalone-AWS-Konten oder das AWS-Organization-Verwaltungskonto sollten Sie Anmeldeinformationen für den Root-Benutzer erstellen und sicher speichern. Aktivieren Sie MFA für den Root-Benutzer.

4. Aktivieren Sie präventive Kontrollen für Root-Benutzer von Mitgliedskonten in AWS-Mehrkonten-Umgebungen.
   a. Erwägen Sie die präventive Sicherheitsvorkehrung Erstellung von Zugriffsschlüsseln für den Root-Benutzer nicht zulassen für Mitgliedskonten.
   b. Erwägen Sie die Aktivierung der präventiven Sicherheitsmaßnahme Aktionen als Root-Benutzer nicht zulassen für Mitgliedskonten.
5. Wenn Sie Anmeldeinformationen für den Root-Benutzer benötigen:
   a. Verwenden Sie ein komplexes Passwort.
   b. Aktivieren Sie Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer, besonders für AWS Organizations-Verwaltungskonten (Bezahlerkonten) (CIS 1.5).
   c. Erwägen Sie die Nutzung von Hardware-MFA-Geräten für Resilienz und Sicherheit, da Einweggeräte auf MFA-Funktionen begrenzt sind und so die Wahrscheinlichkeit verringern, dass die Geräte mit Ihren MFA-Codes für andere Zwecke verwendet werden. Stellen Sie sicher, dass batteriebetriebene MFA-Geräte regelmäßig ausgetauscht werden. (CIS 1.6)
      - Befolgen Sie zur Konfiguration der MFA für den Root-Benutzer die Anleitungen für die Aktivierung einer virtuellen MFA oder eines Hardware-MFA-Geräts.
   d. Erwägen Sie die Nutzung mehrerer MFA-Geräte als Backup. Pro Konto sind bis zu 8 MFA-Geräte zulässig.
      - Beachten Sie, dass die Verwendung von mehr als einem Gerät für den Root-Benutzer automatisch den Ablauf für die Wiederherstellung Ihres Kontos bei Verlust des MFA-Geräts deaktiviert.
   e. Speichern Sie das Passwort in sicherer Weise, und beachten Sie zirkuläre Abhängigkeiten bei der elektronischen Speicherung des Passworts. Speichern Sie das Passwort nicht so, dass der Zugriff darauf erforderlich wäreAWS-Konto, um es abzurufen.
6. Optional: Erwägen Sie die Einrichtung einer periodischen Passwortrotation für den Root-Benutzer.
   - Bewährte Methoden für die Verwaltung von Anmeldeinformationen hängen von Ihren jeweiligen regulatorischen und Richtlinienanforderungen ab. Durch MFA geschützte Root-Benutzer sind nicht auf das Passwort als einzigen Authentifizierungsfaktor angewiesen.
   - Die regelmäßige Änderung des Root-Benutzer-Passworts senkt das Risiko, dass ein unbeabsichtigt offengelegtes Passwort missbraucht werden kann.

Aufdeckende Kontrollen

- Erstellen Sie Alarme, um die Verwendung der Root-Anmeldeinformationen zu erkennen (CIS 1.7). Ist Amazon GuardDuty aktiviert, wird die Nutzung der API-Anmeldeinformationen des Root-Benutzers überwacht und Sie werden über das Ergebnis von RootCredentialUsage benachrichtigt.
- Evaluieren und implementieren Sie die AWSim Well-Architected Security Pillar Conformance Pack enthaltenen aufdeckenden Kontrollen für AWS Config oder, falls SieAWS Control Tower verwenden, die nachdrücklich empfohlenen Kontrollen, die in Control Tower verfügbar sind.

Operationale Anleitung

- Legen Sie fest, wer in der Organisation Zugriff auf die Root-Benutzer-Anmeldeinformationen haben sollte.
  - Verwenden Sie eine Zwei-Personen-Regel, damit keine einzelne Person Zugang zu allen erforderlichen Anmeldeinformationen und zur MFA hat, um sich Root-Benutzer-Zugriff zu verschaffen.
  - Stellen Sie sicher, dass die Organisation – und nicht nur eine einzelne Person – die Kontrolle über die mit dem Konto verbundene Telefonnummer und das entsprechende E-Mail-Alias hat (diese werden für die Passwort- und die MFA-Rücksetzung verwendet).
- Verwenden Sie nur im Ausnahmefall den Root-Benutzer (CIS 1.7).
  - Der AWS-Root-Benutzer darf nicht für alltägliche Aktivitäten verwendet werden, auch nicht für administrative. Melden Sie sich nur dann als Root-Benutzer an, wenn Sie AWS-Aufgaben durchführen müssen, für die der Root-Benutzer erforderlich ist. Alle anderen Aktionen sollten von anderen Benutzern mit den entsprechenden Rollen durchgeführt werden.
- Prüfen Sie regelmäßig, ob der Zugriff auf den Root-Benutzer funktioniert, um Prozeduren vor dem Eintreten von Notsituationen zu testen, die die Verwendung der Root-Benutzer-Anmeldeinformationen erfordern.
- Prüfen Sie regelmäßig, ob die mit dem Konto verbundene E-Mail-Adresse und die unter Alternative Kontakte aufgeführten E-Mail-Adressen funktionieren. Überwachen Sie diese E-Mail-Posteingänge auf etwaige Sicherheitsmitteilungen von `<abuse@amazon.com>`. Stellen Sie auch sicher, dass alle mit dem Konto verbundenen Telefonnummern funktionieren.
- Bereiten Sie Notfallreaktionsprozeduren vor, um auf den Missbrauch des Root-Kontos reagieren zu können. Konsultieren Sie den AWS-Reaktionsleitfaden für Sicherheitsvorfälle und die bewährten Methoden im Abschnitt zu Notfallreaktionen im Whitepaper der Säule „Sicherheit“ für weitere Informationen zum Aufbau einer Sicherheitsstrategie für Ihr AWS-Konto.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP01 Trennen von Workloads mithilfe von Konten
- SEC02-BP01 Verwenden von starken Anmeldemechanismen
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen
- SEC03-BP03 Einrichtung eines Notfallzugriffprozesses
- SEC10-BP05 Vorab bereitgestellter Zugriff

Zugehörige Dokumente:

- AWS Control Tower
- AWS Security Audit Guidelines (Richtlinien zur AWS-Sicherheitsprüfung)
- IAM Best Practices (Bewährte Methoden für IAM)
- Amazon GuardDuty – root credential usage alert (Amazon GuardDuty – Alarm bei Verwendung der Root-Anmeldeinformationen)
- Step-by-step guidance on monitoring for root credential use through CloudTrail (Schritt-für-Schritt-Anleitung zur Überwachung der Verwendung von Root-Anmeldeinformationen mit CloudTrail)
- MFA tokens approved for use with AWS (Zur Verwendung mit AWS genehmigte MFA-Tokens)
- Implementing break glass access on AWS (Implementieren des „Break Glass“-Zugriffs in AWS)
- Top 10 security items to improve in your AWS-Konto (Die 10 wichtigsten Sicherheitsverbesserungen für Ihr AWS-Konto)
- What do I do if I notice unauthorized activity in my AWS-Konto? (Was muss ich tun, wenn ich unbefugte Aktivitäten in meinem AWS-Konto erkenne?)

Zugehörige Videos:

- Enable AWS adoption at scale with automation and governance (AWS-Übernahme in großem Umfang mit Automatisierung und Governance)
- Security Best Practices the Well-Architected Way (Bewährte Sicherheitsmethoden mit durchdachter Architektur)

- Limiting use of AWS root credentials from AWS re:inforce 2022 – Security best practices with AWS IAM (Einschränkung der Verwendung der AWS-Root-Anmeldeinformationen von der AWS re:inforce 2022 – Bewährte Sicherheitsmethoden mit AWS IAM)

Zugehörige Beispiele und Workshops:

- Lab: AWS-Konto und Root-Benutzer

SEC01-BP03 Identifizieren und Validieren von Kontrollzielen

Entsprechend Ihren Compliance-Anforderungen und Risiken, die aus Ihrem Bedrohungsmodell identifiziert werden, können Sie die Kontrollziele und Kontrollen ableiten und validieren, die Sie für Ihren Workload benötigen. Die laufende Validierung von Kontrollzielen und Kontrollen hilft Ihnen, die Effektivität der Risikominderung zu messen.

Gewünschtes Ergebnis: Die Kontrollziele Ihres Unternehmens sind klar definiert und auf Ihre Compliance-Anforderungen abgestimmt. Kontrollen werden durch Automatisierung und Richtlinien implementiert und durchgesetzt und kontinuierlich auf ihre Wirksamkeit bei der Erreichung Ihrer Ziele überprüft. Die Belege für die Wirksamkeit sowohl zu einem bestimmten Zeitpunkt als auch über einen bestimmten Zeitraum hinweg sind jederzeit für Prüfer abrufbar.

Typische Anti-Muster:

- Regulatorische Anforderungen, Markterwartungen und Branchenstandards für verlässliche Sicherheit sind in Ihrem Unternehmen nicht hinreichend vertraut.
- Ihr Framework für die Cybersicherheit und Ihre Kontrollziele sind nicht an den Anforderungen Ihres Unternehmens ausgerichtet.
- Die Implementierung der Kontrollen ist nicht messbar auf Ihre Kontrollziele ausgerichtet.
- Sie verwenden keine Automatisierung zur Berichterstattung über die Wirksamkeit Ihrer Kontrollen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Es gibt zahlreiche gängige Frameworks für die Cybersicherheit, die die Grundlage für Ihre Sicherheitskontrollziele bilden können. Berücksichtigen Sie die regulatorischen Anforderungen, die Markterwartungen und die Branchenstandards für Ihr Unternehmen, um festzustellen, welches

Framework Ihre Anforderungen am besten erfüllt. Beispiele hierfür sind u. a. AICPA SOC 2, HITRUST, PCI-DSS, ISO 27001 und NIST SP 800-53.

Für die von Ihnen festgelegten Kontrollziele sollten Sie verstehen, wie die von Ihnen in Anspruch genommenen AWS-Services Ihnen helfen, diese Ziele zu erreichen. Unter AWS Artifact finden Sie Dokumentationen und Berichte, die auf Ihre Zielframeworks abgestimmt sind. Darin wird der Verantwortungsbereich von AWS beschrieben. Ferner können Sie dort Anleitungen erhalten, in denen der verbleibende Umfang, für den Sie verantwortlich sind, beschrieben wird. Weitere servicespezifische Anleitungen, die sich an verschiedenen Regelwerken orientieren, finden Sie in den AWS Customer Compliance Guides.

Während Sie die Kontrollen zur Erreichung Ihrer Ziele definieren, kodifizieren Sie die Durchsetzung mithilfe von präventiven Kontrollen und automatisieren die Abschwächung mithilfe von detektivischen Kontrollen. Verhindern Sie nicht konforme Ressourcenkonfigurationen und Aktionen in Ihrem AWS Organizations mithilfe von Service-Kontrollrichtlinien (SCPs). Implementieren Sie Regeln in AWS Config zur Überwachung und Berichterstattung über nicht konforme Ressourcen und wechseln Sie dann zu einem Durchsetzungsmodell, sobald Sie von deren Verhalten überzeugt sind. Wenn Sie vordefinierte und verwaltete Regeln einsetzen möchten, die sich an Ihren Cybersicherheits-Rahmenbedingungen orientieren, sollten Sie die Verwendung von AWS Security Hub-Standards als erste Wahl in Betracht ziehen. Der Standard „Foundational Service Best Practices (FSBP)“ von AWS und der CIS-AWS-Foundations-Benchmark sind gute Ausgangspunkte mit Kontrollen, die auf zahlreiche Ziele ausgerichtet sind, die in mehreren Standardframeworks gemeinsam genutzt werden. In Fällen, in denen Security Hub nicht über die gewünschten Kontrollmeldungen verfügt, kann es durch AWS Config-Konformitätspakete ergänzt werden.

Verwenden Sie APN-Partner-Pakete, die vom Global Security and Compliance Acceleration (GSCA)-Team von AWS empfohlen werden, um bei Bedarf Unterstützung von Sicherheitsberatern, Beratungsagenturen, Beweissammlungs- und Berichtssystemen, Prüfern und anderen ergänzenden Services zu erhalten.

Implementierungsschritte

1. Bewerten Sie gängige Frameworks für Cybersicherheit und richten Sie Ihre Kontrollziele an den ausgewählten Frameworks aus.
2. Beschaffen Sie sich mithilfe von AWS Artifact einschlägige Unterlagen über Leitlinien und Verantwortlichkeiten für Ihr Framework. Machen Sie sich klar, welche Teile der Compliance in den AWS-Bereich des Modells der gemeinsamen Verantwortung fallen und für welche Teile Sie verantwortlich sind.

3. Verwenden Sie SCPs, Ressourcenrichtlinien, Rollenvertrauensrichtlinien und andere Maßnahmen für den Integritätsschutz, um nicht konforme Ressourcenkonfigurationen und Aktionen zu verhindern.
4. Evaluieren Sie die Implementierung von Security Hub-Standards und AWS Config-Konformitätspaketen, die mit Ihren Kontrollzielen übereinstimmen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC03-BP01 Definieren von Zugriffsanforderungen
- SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung
- SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung
- OPS01-BP03 Bewerten der Governance-Anforderungen
- OPS01-BP04 Bewerten der Compliance-Anforderungen
- PERF01-BP05 Verwenden von Richtlinien und Referenzarchitekturen
- COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen

Zugehörige Dokumente:

- AWS Customer Compliance Guides

Zugehörige Tools:

- AWS Artifact

SEC01-BP04 Sicherstellen der Aktualität von Informationen zu Sicherheitsbedrohungen

Bleiben Sie auf dem Laufenden über die neuesten Bedrohungen und Abhilfemaßnahmen, indem Sie Veröffentlichungen zu Bedrohungsdaten und Datenfeeds der Branche auf Aktualisierungen verfolgen. Prüfen Sie Angebote für verwaltete Services, die automatisch auf der Grundlage der neuesten Bedrohungsdaten aktualisiert werden.

Gewünschtes Ergebnis: Sie bleiben auf dem Laufenden, da die Branchenpublikationen mit den neuesten Bedrohungen und Empfehlungen aktualisiert werden. Sie nutzen die Automatisierung, um potenzielle Schwachstellen und Gefährdungen zu erkennen, während Sie neue Bedrohungen

identifizieren. Sie ergreifen Maßnahmen zur Eindämmung dieser Bedrohungen. Sie übernehmen AWS-Services, die automatisch mit den neuesten Bedrohungsdaten aktualisiert werden.

Typische Anti-Muster:

- kein zuverlässiger und wiederholbarer Mechanismus, um über die neuesten Bedrohungsdaten informiert zu sein
- manuelle Bestandsführung Ihres Technologieportfolios, Ihrer Workloads und Abhängigkeiten, was menschliches Eingreifen im Hinblick auf potenzielle Schwachstellen und Gefährdungen erfordert
- fehlende Mechanismen zur Aktualisierung Ihrer Workloads und Abhängigkeiten auf die neuesten verfügbaren Versionen, die bekannte Bedrohungsabwehrmaßnahmen bieten

Vorteile der Einführung dieser bewährten Methode: Die Verwendung von Bedrohungsdatenquellen, um auf dem Laufenden zu bleiben, verringert das Risiko, wichtige Änderungen in der Bedrohungslandschaft zu verpassen, die sich auf Ihr Unternehmen auswirken können. Wenn Sie Ihre Workloads und deren Abhängigkeiten automatisiert auf potenzielle Schwachstellen oder Gefährdungen prüfen, diese erkennen und beheben, können Sie Risiken im Vergleich zu manuellen Alternativen schnell und vorhersehbar eindämmen. Dies trägt dazu bei, Zeit und Kosten im Zusammenhang mit der Behebung von Schwachstellen zu kontrollieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Verfolgen Sie vertrauenswürdige Veröffentlichungen zu Bedrohungsdaten, um über die Bedrohungslandschaft auf dem Laufenden zu bleiben. Konsultieren Sie die Wissensdatenbank von MITRE ATT&CK. Hier finden Sie Dokumentationen über bekannte gegnerische Taktiken, Techniken und Verfahren (Tactics, Techniques and Procedures, TTPs). Informieren Sie sich in der MITRE-Liste Common Vulnerabilities and Exposures (CVE) über bekannte Sicherheitslücken in Produkten, auf die Sie angewiesen sind. Verstehen Sie kritische Risiken für Webanwendungen mit dem populären Projekt OWASP Top 10 des Open Worldwide Application Security Project (OWASP).

Bleiben Sie auf dem Laufenden über AWS-Sicherheitsereignisse und empfohlene Abhilfemaßnahmen mit AWS-Sicherheitsberichten für CVEs.

Um den Gesamtaufwand für die Aktualisierung zu reduzieren, sollten Sie AWS-Services nutzen. Diese beziehen die neue Bedrohungsdaten im Laufe der Zeit automatisch ein. Zum Beispiel behält Amazon GuardDuty den Überblick über die Bedrohungsdaten der Branche, um anormale

Verhaltensweisen und Bedrohungssignaturen in Ihren Konten zu erkennen. Amazon Inspector hält automatisch eine Datenbank mit den CVEs auf dem neuesten Stand. Diese Datenbank wird für die kontinuierlichen Scan-Funktionen verwendet. Sowohl AWS WAF als auch AWS Shield Advanced bieten verwaltete Regelgruppen, die automatisch aktualisiert werden, wenn neue Bedrohungen auftauchen.

Sehen Sie sich die Säule „Operative Exzellenz“ – AWS-Well-Architected-Framework an, um mehr über automatisiertes Flottenmanagement und Patching zu erfahren.

Implementierungsschritte

- Abonnieren Sie Updates für Bedrohungsinformationen, die für Ihr Unternehmen und Ihre Branche relevant sind. Abonnieren Sie die AWS-Sicherheitsberichte.
- Erwägen Sie die Einführung von Services, die neue Bedrohungsdaten automatisch einbeziehen, wie Amazon GuardDuty und Amazon Inspector.
- Erstellen Sie eine Flottenmanagement- und Patching-Strategie, die sich an den Best Practices der Säule „Operative Exzellenz“ – AWS-Well-Architected-Framework“ orientiert.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells
- OPS01-BP05 Bewerten der Bedrohungsszenarien
- OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung

SEC01-BP05 Verringern des Umfangs der Sicherheitsverwaltung

Ermitteln Sie, ob Sie Ihren Sicherheitsumfang reduzieren können, indem Sie AWS-Services verwenden, die die Verwaltung bestimmter Kontrollen in AWS verlagern (verwaltete Services). Mit diesen Services können Sie Ihre Wartungsaufgaben im Bereich Sicherheit reduzieren, z. B. die Bereitstellung der Infrastruktur, die Einrichtung von Software, Patches oder Backups.

Gewünschtes Ergebnis: Sie berücksichtigen den Umfang Ihrer Sicherheitsverwaltung bei der Auswahl von AWS-Services für Ihren Workload. Die Kosten für den Verwaltungsaufwand und die Wartungsaufgaben (die Gesamtbetriebskosten (Total Cost of Ownership, TCO) werden gegen die Kosten der von Ihnen ausgewählten Services abgewogen. Hinzu kommen weitere Überlegungen im

Rahmen von Well-Architected. Sie integrieren die Kontroll- und Compliance-Dokumentation von AWS in Ihre Kontrollbewertungs- und Verifizierungsverfahren.

Typische Anti-Muster:

- Bereitstellung von Workloads ohne gründliches Verständnis des Modells der geteilten Verantwortung für die von Ihnen ausgewählten Services
- Hosten von Datenbanken und anderen Technologien auf virtuellen Maschinen, ohne einen entsprechenden verwalteten Service evaluiert zu haben
- Nichtberücksichtigung von Sicherheitsverwaltungsaufgaben bei den Gesamtbetriebskosten des Hostings von Technologien auf virtuellen Maschinen im Vergleich zu verwalteten Serviceoptionen

Vorteile der Einführung dieser bewährten Methode: Der Einsatz von verwalteten Services kann Ihren Gesamtaufwand für die Verwaltung der betrieblichen Sicherheitskontrollen verringern, was Ihre Sicherheitsrisiken und Gesamtbetriebskosten reduzieren kann. Die Zeit, die Sie sonst für bestimmte Sicherheitsaufgaben aufwenden müssten, können Sie in Aufgaben investieren, die Ihrem Unternehmen einen größeren Nutzen bringen. Verwaltete Services können auch den Umfang Ihrer Compliance-Anforderungen reduzieren, indem sie einige Kontrollanforderungen in AWS verlagern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Es gibt mehrere Möglichkeiten, wie Sie die Komponenten Ihres Workloads in AWS integrieren können. Die Installation und der Betrieb von Technologien auf Amazon EC2-Instances erfordert häufig, dass Sie den größten Teil der gesamten Sicherheitsverantwortung übernehmen. Um den Aufwand für die Durchführung bestimmter Kontrollen zu verringern, sollten Sie von AWS verwaltete Services identifizieren, die den Umfang Ihrer Seite des Modells der geteilten Verantwortung verringern, und verstehen, wie Sie diese in Ihrer bestehenden Architektur nutzen können. Beispiele sind die Verwendung der Amazon Relational Database Service (Amazon RDS) für die Bereitstellung von Datenbanken, Amazon Elastic Kubernetes Service (Amazon EKS) oder Amazon Elastic Container Service (Amazon ECS) für die Orchestrierung von Containern oder die Verwendung von Serverless-Optionen. Überlegen Sie bei der Entwicklung neuer Anwendungen, welche Services dazu beitragen können, den Zeit- und Kostenaufwand für die Implementierung und Verwaltung von Sicherheitskontrollen zu reduzieren.

Auch Compliance-Anforderungen können bei der Auswahl von Services eine Rolle spielen. Verwaltete Services können die Einhaltung einiger Anforderungen in AWS verlagern. Sprechen Sie

mit Ihrem Compliance-Team darüber, inwieweit es sich mit der Prüfung der von Ihnen betriebenen und verwalteten Services und der Annahme von Kontrollerklärungen in den entsprechenden Audit-Berichten von AWS wohl fühlt. Sie können die in AWS Artifact gefundenen Audit-Artefakte Ihren Prüfern oder Regulierungsbehörden als Nachweis für AWS-Sicherheitskontrollen vorlegen. Sie können bei der Gestaltung Ihrer Architektur auch die Hinweise zur Verantwortung verwenden, die in einigen AWS-Audit-Artefakten enthalten sind, zusammen mit den AWS Customer Compliance Guides. Dieser Leitfaden hilft Ihnen, die zusätzlichen Sicherheitskontrollen zu bestimmen, die Sie einrichten sollten, um die spezifischen Anwendungsfälle Ihres Systems zu unterstützen.

Wenn Sie verwaltete Services nutzen, sollten Sie mit dem Prozess der Aktualisierung ihrer Ressourcen auf neuere Versionen vertraut sein (z. B. die Aktualisierung der Version einer von Amazon RDS verwalteten Datenbank oder einer Laufzeit einer Programmiersprache für eine AWS Lambda-Funktion). Auch wenn der verwaltete Dienst diesen Vorgang für Sie durchführt, sind Sie für die Konfiguration des Zeitpunkts der Aktualisierung und die Auswirkungen auf Ihren Betrieb selbst verantwortlich. Tools wie AWS Health können Ihnen helfen, diese Updates in Ihren Umgebungen zu verfolgen und zu verwalten.

Implementierungsschritte

1. Bewerten Sie die Komponenten Ihres Workloads, die durch einen verwalteten Service ersetzt werden können.
   a. Wenn Sie einen Workload zu AWS migrieren, sollten Sie den geringeren Verwaltungsaufwand (Zeit und Kosten) und die Verringerung des Risikos berücksichtigen, wenn Sie folgende Optionen für Ihren Workload bewerten: Hostwechsel, Faktorwechsel, Plattformwechsel, Rebuild oder Ersatz. Manchmal können zusätzliche Investitionen zu Beginn einer Migration auf lange Sicht erhebliche Einsparungen bringen.
2. Ziehen Sie die Implementierung von verwalteten Services wie Amazon RDS in Betracht, anstatt Ihre eigenen Technologiebereitstellungen zu installieren und zu verwalten.
3. Verwenden Sie die Anleitung zur Verantwortung in AWS Artifact, um die Sicherheitskontrollen zu bestimmen, die Sie für Ihren Workload einrichten sollten.
4. Führen Sie ein Inventar der genutzten Ressourcen und halten Sie sich über neue Services und Ansätze auf dem Laufenden, um neue Möglichkeiten zur Reduzierung des Umfangs zu ermitteln.

Ressourcen

Zugehörige bewährte Methoden:

- PERF02-BP01 Auswählen der besten Datenverarbeitungsoptionen für den Workload
- PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt
- SUS05-BP03 Verwenden verwalteter Services

Zugehörige Dokumente:

- Planned lifecycle events for AWS Health

Zugehörige Tools:

- AWS Health
- AWS Artifact
- AWS Customer Compliance Guides

Zugehörige Videos:

- How do I migrate to an Amazon RDS or Aurora MySQL DB instance using AWS DMS?
- AWS re:Invent 2023 – Manage resource lifecycle events at scale with AWS Health

SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen

Wenden Sie bei der Entwicklung und Bereitstellung von Sicherheitskontrollen, die in Ihren AWS-Umgebungen Standard sind, moderne DevOps-Verfahren an. Definieren Sie Standard-Sicherheitskontrollen und -konfigurationen mithilfe von IaC-Vorlagen (Infrastructure as Code), erfassen Sie Änderungen in einem Versionskontrollsystem, testen Sie Änderungen als Teil einer CI/CD-Pipeline und automatisieren Sie die Bereitstellung von Änderungen in Ihren AWS-Umgebungen.

Gewünschtes Ergebnis: IaC-Vorlagen erfassen standardisierte Sicherheitskontrollen und übergeben sie an ein Versionskontrollsystem. CI/CD-Pipelines sind an Stellen vorhanden, die Änderungen erkennen und das Testen und Bereitstellen Ihrer AWS-Umgebungen automatisieren. Mechanismen zum Integritätsschutz erkennen und warnen vor Fehlkonfigurationen in Vorlagen, bevor die Bereitstellung erfolgt. Workloads werden in Umgebungen bereitgestellt, in denen Standardkontrollen vorhanden sind. Die Teams können genehmigte Servicekonfigurationen über einen Self-Service-Mechanismus bereitstellen. Die Strategien zur Gewährleistung der Sicherheit bei der Sicherung und Wiederherstellung von Kontrollkonfigurationen, Skripten und zugehörigen Daten sind etabliert.

Typische Anti-Muster:

- Manuelle Änderungen an Ihren Standard-Sicherheitskontrollen über eine Webkonsole oder eine Befehlszeilenschnittstelle.
- Sich darauf verlassen, dass die einzelnen Workload-Teams die von einem zentralen Team festgelegten Kontrollen manuell umsetzen.
- Sich auf ein zentrales Sicherheitsteam verlassen, das auf Anfrage eines Workload-Teams Kontrollen auf Workload-Ebene bereitstellt.
- Erlauben, dass dieselben Personen oder Teams Automatisierungsskripte für die Sicherheitskontrolle entwickeln, testen und bereitstellen, ohne dass eine angemessene Aufgabentrennung oder gegenseitige Kontrolle stattfindet.

Vorteile der Einführung dieser bewährten Methode: Die Verwendung von Vorlagen zur Definition Ihrer Standard-Sicherheitskontrollen ermöglicht es Ihnen, Änderungen im Laufe der Zeit mithilfe eines Versionskontrollsystems zu verfolgen und zu vergleichen. Der Einsatz von Automatisierung zum Testen und Bereitstellen von Änderungen schafft Standardisierung und Vorhersehbarkeit, erhöht die Chancen auf eine erfolgreiche Bereitstellung und reduziert manuelle, sich wiederholende Aufgaben. Durch die Bereitstellung eines Self-Service-Mechanismus für Workload-Teams zur Bereitstellung genehmigter Services und Konfigurationen wird das Risiko von Fehlkonfigurationen und Missbrauch verringert. Das hilft ihnen auch dabei, Kontrollen früher in den Entwicklungsprozess einzubauen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wenn Sie die in SEC01-BP01 Trennen von Workloads mithilfe von Konten beschriebenen Verfahrensweisen befolgen, erhalten Sie am Ende mehrere AWS-Konten für verschiedene Umgebungen, die Sie unter Verwendung von AWS Organizations verwalten. Auch wenn jede dieser Umgebungen und Workloads unterschiedliche Sicherheitskontrollen erfordert, können Sie einige Sicherheitskontrollen in Ihrer Organisation standardisieren. Beispiele hierfür sind die Integration zentraler Identitätsanbieter, die Definition von Netzwerken und Firewalls und die Konfiguration von Standardorten für die Speicherung und Analyse von Protokollen. Analog zur Anwendung von Infrastructure as Code (IaC) zur Anwendung der gleichen strikten Vorgehensweise bei der Entwicklung von Anwendungscode auf die Bereitstellung der Infrastruktur können Sie IaC auch zur Definition und Bereitstellung Ihrer Standard-Sicherheitskontrollen verwenden.

Definieren Sie Ihre Sicherheitskontrollen nach Möglichkeit deklarativ, wie z. B. in AWS CloudFormation, und speichern Sie sie in einem Versionskontrollsystem. Nutzen Sie DevOps-

Praktiken, um die Bereitstellung Ihrer Kontrollen zu automatisieren und so besser vorhersehbare Releases, automatisierte Tests mit Tools wie AWS CloudFormation Guard und die Erkennung von Abweichungen zwischen Ihren bereitgestellten Kontrollen und der gewünschten Konfiguration zu ermöglichen. Sie können Services wie AWS CodePipeline, AWS CodeBuild und AWS CodeDeploy verwenden, um eine CI/CD-Pipeline zu erstellen. Berücksichtigen Sie die Hinweise in Organizing Your AWS Environment Using Multiple Accounts, um diese Services in eigenen Konten separat von anderen Bereitstellungspipelines zu konfigurieren.

Sie können auch Vorlagen definieren, um die Definition und Bereitstellung von AWS-Konten, Services und Konfigurationen zu standardisieren. Diese Technik ermöglicht es einem zentralen Sicherheitsteam, diese Definitionen zu verwalten und sie den Workload-Teams über einen Self-Service-Ansatz zur Verfügung zu stellen. Eine Möglichkeit, dies zu erreichen, ist die Verwendung von Service Catalog, wo Sie Vorlagen als Produkte veröffentlichen können, die Workload-Teams in ihre eigenen Pipeline-Bereitstellungen einbinden können. Wenn Sie AWS Control Tower verwenden, sind einige Vorlagen und Kontrollen als Ausgangspunkt verfügbar. Control Tower bietet zudem die Funktion Account Factory, mit der Workload-Teams neue AWS-Konten unter Verwendung der von Ihnen definierten Standards erstellen können. Mit dieser Funktion sind Sie nicht mehr auf ein zentrales Team angewiesen, das neue Konten genehmigt und anlegt, wenn diese von Ihren Workload-Teams als notwendig erachtet werden. Sie benötigen diese Konten möglicherweise, um verschiedene Workload-Komponenten zu isolieren, z. B. aufgrund ihrer Funktion, der Sensibilität der verarbeiteten Daten oder ihres Verhaltens.

Implementierungsschritte

1. Legen Sie fest, wie Sie Ihre Vorlagen in einem Versionskontrollsystem speichern und pflegen wollen.
2. Erstellen Sie CI/CD-Pipelines zum Testen und Bereitstellen Ihrer Vorlagen. Definieren Sie Tests, um zu prüfen, ob Fehlkonfigurationen vorliegen und ob die Vorlagen den Standards Ihres Unternehmens entsprechen.
3. Erstellen Sie einen Katalog mit standardisierten Vorlagen für Workload-Teams zur Bereitstellung von AWS-Konten und -Services gemäß Ihren Anforderungen.
4. Implementieren Sie sichere Sicherungs- und Wiederherstellungsstrategien für die Konfiguration Ihrer Kontrollen, Skripte und zugehörigen Daten.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP01 Verwendung einer Versionskontrolle
- OPS05-BP04 Einsatz von Systemen zur Bild- und Bereitstellungsverwaltung
- REL08-BP05 Automatisieren von Änderungen
- SUS06-BP01 Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen

Zugehörige Dokumente:

- Organizing Your AWS Environment Using Multiple Accounts

Zugehörige Beispiele:

- Automate account creation, and resource provisioning using Service Catalog, AWS Organizations, and AWS Lambda
- Strengthen the DevOps pipeline and protect data with AWS Secrets Manager, AWS KMS, and AWS Certificate Manager

Zugehörige Tools:

- AWS CloudFormation Guard
- Landing Zone Accelerator in AWS

SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells

Führen Sie Bedrohungsmodellierungen zur Identifizierung und Pflege eines aktuellen Registers potenzieller Bedrohungen und entsprechender Abhilfemaßnahmen für Ihren Workload durch. Priorisieren Sie Ihre Bedrohungen und passen Sie Ihre Sicherheitskontrollen an, um zu verhindern, zu erkennen und zu reagieren. Überarbeiten und halten Sie diese Methoden im Kontext Ihres Workloads und der sich entwickelnden Sicherheitslandschaft aktuell.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Was versteht man unter Bedrohungsmodellierung?

„Bedrohungsmodellierung dient der Identifizierung, Kommunikation und dem Verständnis von Bedrohungen und Abhilfemaßnahmen im Kontext des Schutzes von etwas Wertvollem.“ – The Open Web Application Security Project (OWASP) Application Threat Modeling

Wozu dient die Bedrohungsmodellierung?

Systeme sind komplex und werden mit der Zeit immer komplexer und leistungsfähiger. Gleichzeitig liefern sie immer mehr geschäftlichen Wert und verbessern die Kundenzufriedenheit und -bindung. Dies bedeutet, dass Entscheidungen zum IT-Design immer mehr Anwendungsfälle berücksichtigen müssen. Diese Komplexität und die zunehmende Zahl der Anwendungsfälle macht unstrukturierte Konzepte ineffektiv, wenn es um das Erkennen und Bekämpfen von Bedrohungen geht. Stattdessen wird ein systematisches Konzept benötigt, das die potenziellen Bedrohungen für ein System aufführen und Abhilfemaßnahmen benennen und priorisieren kann, um sicherzustellen, dass die begrenzten Ressourcen einer Organisation in maximaler Weise in der Lage sind, die Sicherheitslage des Systems insgesamt zu verbessern.

Die Bedrohungsmodellierung dient zum Aufbau eines solchen systematischen Konzepts, damit Probleme frühzeitig im Designprozess erkannt und angegangen werden können, so lange Abhilfemaßnahmen noch mit niedrigen relativen Kosten und geringem Aufwand verbunden sind, was später im Lebenszyklus nicht mehr der Fall ist. Dieses Konzept entspricht dem Branchenprinzip des Shift-Left-Sicherheitsansatzes. Letztendlich ist die Bedrohungsmodellierung in den Risikomanagementprozess einer Organisation integriert und hilft mit einem auf Bedrohungen ausgerichteten Konzept bei Entscheidungen dazu, welche Kontrollmechanismen zu implementieren sind.

Wann sollte eine Bedrohungsmodellierung durchgeführt werden?

Beginnen Sie mit der Bedrohungsmodellierung so früh wie möglich im Lebenszyklus Ihres Workloads. Dies gibt Ihnen die benötigte Flexibilität im Umgang mit den identifizierten Bedrohungen. Wie bei Softwarebugs gilt auch hier: Je früher Sie Bedrohungen identifizieren, desto kostengünstiger ist es, sie zu beheben. Ein Bedrohungsmodell ist ein lebendiges Dokument, dass stetig weiterentwickelt werden sollte, während sich Ihre Workloads verändern. Überprüfen Sie regelmäßig Ihre Bedrohungsmodelle, vor allem bei größeren Änderungen, bei Änderungen der Bedrohungslandschaft, oder wenn Sie neue Funktionen oder Services einführen.

Implementierungsschritte

Wie wird die Bedrohungsmodellierung durchgeführt?

Es gibt viele verschiedene Möglichkeiten zur Durchführung von Bedrohungsmodellierungen. Ähnlich wie bei Programmiersprachen gibt es Vor- und Nachteile und Sie sollten den Ansatz wählen, der für Sie am besten funktioniert. Ein Konzept besteht darin, mit Shostack's 4 Question Frame for Threat Modeling zu beginnen, das aus offenen Fragen besteht, die Ihre Bedrohungsmodellierung strukturieren:

1. Woran arbeiten wir?

   Diese Frage dient dazu, das von Ihnen aufgebaute System sowie die sicherheitsrelevanten Details zu diesem System zu verstehen. Für die Beantwortung dieser Frage ist es üblich, ein Modell oder Diagramm zur Visualisierung dessen aufzustellen, was aufgebaut wird, etwa in Gestalt eines Datenflussdiagramms. Das Aufschreiben von Annahmen und wichtigen Details zum System hilft ebenfalls beim Verständnis des Umfangs. Dadurch können sich alle, die zum Bedrohungsmodell beitragen, auf dasselbe konzentrieren und zeitraubende Umwege über irrelevante Themen (wie etwa veraltete Versionen des Systems) vermeiden. Wenn Sie beispielsweise eine Web-Anwendung erstellen, ist es wahrscheinlich nicht relevant, sich um die Bedrohungsmodellierung im Zusammenhang mit der Bootsequenz für Browser-Clients in vertrauenswürdigen Betriebssystemen zu kümmern, da Sie darauf ohnehin keinen Einfluss haben.

2. Was kann schief gehen?

   Hier identifizieren Sie die Bedrohungen für Ihr System. Bedrohungen sind versehentliche oder beabsichtigte Handlungen oder Ereignisse, die unerwünschte Folgen haben und die Sicherheit Ihres Systems beeinträchtigen können. Ohne ein klares Verständnis dessen, was schief gehen kann, haben Sie keine Möglichkeit, etwas dagegen zu unternehmen.

   Es gibt keine kanonische Liste dessen, was schief gehen kann. Die Erstellung dieser Liste erfordert Brainstorming und die Zusammenarbeit all Ihrer Teammitglieder und der relevanten Beteiligten an der Bedrohungsmodellierung. Sie können das Brainstorming unterstützen, indem Sie ein Modell zur Identifizierung von Bedrohungen verwenden, z. B. STRIDE, das verschiedene Kategorien zur Bewertung anbietet: Spoofing, Manipulation, Zurückweisung, Offenlegung von Informationen, Denial of Service und Erhöhung der Berechtigung. Dazu sollten Sie zur Inspiration vorhandene Listen und Forschungsergebnisse heranziehen, etwa die OWASP Top 10, den HiTrust Threat Catalog und den eigenen Bedrohungskatalog Ihrer Organisation.

3. Wie gehen wir damit um?

   Wie schon bei der vorherigen Frage gibt es auch hier keine kanonische Liste möglicher Abhilfemaßnahmen. Die Inputs für diesen Schritt sind die identifizierten Bedrohungen, Akteure und Verbesserungsbereiche aus dem vorherigen Schritt.

Sicherheit und Compliance unterliegen der geteilten Verantwortung zwischen Ihnen und AWS. Der Frage „Wie gehen wir damit um?“ sollte unbedingt die Frage „Wer ist für die Maßnahmen verantwortlich?“ angeschlossen werden. Das Verständnis der Verantwortungsverteilung zwischen Ihnen und AWS hilft Ihnen bei der Anpassung der Bedrohungsmodellierung an die Abhilfemaßnahmen, die Ihrer Kontrolle unterliegen und in der Regel aus einer Kombination aus AWS-Servicekonfigurationsoptionen und Ihren eigenen systemspezifischen Abhilfemaßnahmen bestehen.

Für den AWS-Teil der gemeinsamen Verantwortung werden Sie feststellen, dass AWS-Services in den Bereich vieler Compliance-Programme fallen. Diese Programme helfen Ihnen, sich mit den zuverlässigen Kontrollmöglichkeiten bei AWS zur Sicherheitswahrung und Compliance in der Cloud vertraut zu machen. Die Audit-Berichte dieser Programme stehen für AWS-Kunden von AWS Artifact zum Download zur Verfügung.

Unabhängig davon, welche AWS-Services Sie nutzen, gibt es immer ein Element der Kundenverantwortung, und an diese Verantwortungen angepasste Abhilfemaßnahmen sollten Teil Ihres Bedrohungsmodells sein. Für Sicherheitskontrollabhilfen für die AWS-Services selbst sollten Sie die Implementierung von Sicherheitskontrollen über Domains hinweg erwägen, einschließlich Domains wie Identitäts- und Zugriffsmanagement (Authentifizierung und Autorisierung), Datenschutz (im Ruhezustand und während der Übertragung), Infrastruktursicherheit, Protokollierung und Überwachung. Die Dokumentation für jeden AWS-Service enthält ein spezielles Sicherheitskapitel mit Anleitungen zu den Sicherheitskontrollen, die Abhilfemaßnahmen unterstützen können. Wichtig ist, dass Sie den Code, den Sie schreiben, und dessen Abhängigkeiten berücksichtigen und an Kontrollen denken, die Sie für den Umgang mit den damit verbundenen Bedrohungen implementieren können. Bei diesen Kontrollen könnte es sich um Dinge wie Eingabevalidierung, Sitzungsabwicklung und Umgang mit Grenzen handeln. Oft ist der Löwenanteil der Bedrohungen mit benutzerdefiniertem Code verbunden, konzentrieren Sie sich also besonders darauf.

4. Haben wir gute Arbeit geleistet?

   Ihr Team und die Organisation verfolgen das Ziel, die Qualität der Bedrohungsmodelle und die Geschwindigkeit zu verbessern, mit der Sie die Bedrohungsmodellierung im Laufe der Zeit durchführen. Diese Verbesserungen werden durch eine Kombination von Praxis, Lernen, Lehren und Prüfen ermöglicht. Um dies zu vertiefen und praktisch umzusetzen, sollten Sie und Ihr Team den Trainingskurs zum Thema Korrekte Bedrohungsmodellierung für Builder oder den dazugehörigen Workshop absolvieren. Wenn Sie nach Anleitungen zur Integration der Bedrohungsmodellierung in den Anwendungsentwicklungslebenszyklus Ihrer Organisation suchen,

beachten Sie auch den Post zum Thema Bedrohungsmodellierungskonzepte im AWS Security Blog.

Threat Composer

Zur Unterstützung und Anleitung bei der Erstellung von Bedrohungsmodellen können Sie das Threat Composer-Tool verwenden, das darauf ausgerichtet ist, bei der Erstellung von Bedrohungsmodellen die Zeit bis zur Wertschöpfung zu verkürzen. Das Tool hilft Ihnen bei den folgenden Aufgaben:

- Schreiben Sie nützliche, an der Bedrohungsgrammatik ausgerichtete Bedrohungserklärungen, die in einem natürlichen, nicht-linearen Arbeitsablauf funktionieren.
- Generieren Sie ein für Menschen lesbares Bedrohungsmodell.
- Generieren Sie ein maschinenlesbares Bedrohungsmodell, damit Sie Bedrohungsmodelle wie Code behandeln können.
- Mit dem Insights Dashboard können Sie schnell Bereiche identifizieren, in denen die Qualität und die Abdeckung verbessert werden müssen.

Für weitere Informationen rufen Sie Threat Composer auf und wechseln Sie zum systemdefinierten Beispielarbeitsbereich.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP03 Identifizieren und Validieren von Kontrollzielen
- SEC01-BP04 Sicherstellen der Aktualität von Informationen zu Sicherheitsbedrohungen
- SEC01-BP05 Verringern des Umfangs der Sicherheitsverwaltung
- SEC01-BP08 Regelmäßiges Bewerten und Implementieren neuer Sicherheitsservices und -features

Zugehörige Dokumente:

- How to approach threat modeling (AWS Security Blog)
- NIST: Guide to Data-Centric System Threat Modelling

Zugehörige Videos:

- AWS Summit ANZ 2021 - How to approach threat modelling
- AWS Summit ANZ 2022 - Scaling security – Optimise for fast and secure delivery

Zugehöriges Training:

- Threat modeling the right way for builders – AWS Skill Builder virtual self-paced training
- Threat modeling the right way for builders – AWS Workshop

Zugehörige Tools:

- Threat Composer

SEC01-BP08 Regelmäßiges Bewerten und Implementieren neuer Sicherheitsservices und -features

Bewerten und implementieren Sie Sicherheitsservices und -features von AWS und AWS-Partnern, mit denen Sie die Sicherheitsstrategie für Ihren Workload weiterentwickeln können.

Gewünschtes Ergebnis: Sie verfügen über eine Standardmethode, die Sie über neue Features und Services informiert, die von AWS und AWS-Partnern veröffentlicht werden. Sie bewerten, wie sich diese neuen Funktionen auf das Design der aktuellen und neuen Kontrollen für Ihre Umgebungen und Workloads auswirken.

Typische Anti-Muster:

- Sie abonnieren keine Blogs und RSS-Feeds von AWS, um schnell von relevanten neuen Features und Servicesn zu erfahren
- Sie verlassen sich auf Nachrichten und Updates über Sicherheitsservices und Features aus zweiter Hand
- Sie halten AWS-Benutzer in Ihrer Organisation nicht dazu an, sich über die neuesten Updates zu informieren

Vorteile der Einführung dieser bewährten Methode: Indem Sie sich über neue Sicherheitsservices und Features auf dem Laufenden halten, können Sie fundierte Entscheidungen über die Implementierung von Kontrollen in Ihren Cloud-Umgebungen und Workloads treffen. Diese Quellen tragen dazu bei, das Bewusstsein für die sich entwickelnde Sicherheitslandschaft zu schärfen und zu zeigen, wie AWS-Services zum Schutz vor neuen und aufkommenden Bedrohungen genutzt werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

AWS informiert Kunden über neue Sicherheitsservices und Funktionen über verschiedene Kanäle:

- Neuigkeiten zu AWS
- AWS News Blog
- AWS Security Blog
- AWS-Sicherheitsberichte
- Überblick über die AWS-Dokumentation

Sie können ein Thema der AWS Daily Feature Updates mit Amazon Simple Notification Service (Amazon SNS) abonnieren, um eine umfassende tägliche Zusammenfassung der Updates zu erhalten. Einige Sicherheitsservices wie Amazon GuardDuty und AWS Security Hub bieten ihre eigenen SNS-Themen an, um über neue Standards, Erkenntnisse und andere Aktualisierungen für diese speziellen Services informiert zu bleiben.

Neue Services und Funktionen werden auch auf Konferenzen, Veranstaltungen und Webinaren, die jedes Jahr rund um den Globus stattfinden, angekündigt und im Detail beschrieben. Besonders interessant ist dabei die jährliche Sicherheitskonferenz AWS re:Inforce und die breiter angelegte Konferenz AWS re:Invent. In den bereits erwähnten AWS-Nachrichtenkanälen werden diese Konferenzankündigungen über Sicherheit und andere Services geteilt, und Sie können sich Deep-Dive-Breakout-Sitzungen online auf dem AWS-Events-Kanal auf YouTube ansehen.

Sie können auch Ihr AWS-Konto-Team nach den neuesten Updates und Empfehlungen für Sicherheitsservices fragen. Sie können Ihr Team über das Verkaufssupport-Formular erreichen, wenn Ihnen dessen direkte Kontaktinformationen nicht vorliegen. Gleichermaßen erhalten Sie, wenn Sie den AWS-Enterprise-Support abonniert haben, wöchentliche Updates von Ihrem Technical Account Manager (TAM) und können ein regelmäßiges Review-Meeting mit ihm vereinbaren.

Implementierungsschritte

1. Abonnieren Sie die verschiedenen Blogs und Bulletins mit Ihrem bevorzugten RSS-Reader oder die SNS-Thema Daily Features Updates.
2. Überlegen Sie, welche AWS Veranstaltungen Sie besuchen sollten, um sich aus erster Hand über neue Features und Services zu informieren.

3. Vereinbaren Sie Besprechungen mit Ihrem AWS-Konto-Team für alle Fragen zur Aktualisierung von Sicherheitsservices und Features.
4. Ziehen Sie in Erwägung, den Enterprise Support zu abonnieren, um regelmäßige Konsultationen mit einem Technical Account Manager (TAM) zu erhalten.

Ressourcen

Zugehörige bewährte Methoden:

- PERF01-BP01 Informieren über verfügbare Cloud-Services und -Funktionen
- COST01-BP07 Verfolgen neuer Serviceversionen

Identity and Access Management

Fragen

- SEC 2. Was ist bei der Verwaltung der Authentifizierung für Personen und Rechner zu beachten?
- SEC 3. Wie verwalten Sie Berechtigungen für Personen und Maschinen?

SEC 2. Was ist bei der Verwaltung der Authentifizierung für Personen und Rechner zu beachten?

Beim Betrieb sicherer AWS-Workloads gibt es zwei Arten von Identitäten, die Sie verwalten müssen. Wenn Sie verstehen, welche Arten von Identitäten Sie verwalten und Zugriff gewähren müssen, können Sie sicherstellen, dass die richtigen Identitäten unter den richtigen Bedingungen Zugriff auf die richtigen Ressourcen haben.

Menschliche Identitäten: Ihre Administratoren, Entwickler, Bediener und Endbenutzer benötigen eine Identität für den Zugriff auf Ihre AWS-Umgebungen und -Anwendungen. Dies sind Mitglieder Ihrer Organisation oder externe Benutzer, mit denen Sie zusammenarbeiten, und die mit Ihren AWS-Ressourcen über einen Webbrowser, eine Client-Anwendung oder interaktive Befehlszeilen-Tools interagieren.

Maschinenidentitäten: Ihre Service-Anwendungen, betrieblichen Tools und Workloads benötigen eine Identität, um Anforderungen an AWS-Services zu stellen, z. B. um Daten zu lesen. Zu diesen Identitäten gehören Maschinen, die in Ihrer AWS-Umgebung ausgeführt werden, z. B. Amazon EC2-Instances oder AWS Lambda-Funktionen. Sie können auch Maschinenidentitäten für externe

Parteien verwalten, die Zugriff benötigen. Darüber hinaus verfügen Sie möglicherweise auch über Maschinen außerhalb von AWS, die Zugriff auf Ihre AWS-Umgebung benötigen.

Bewährte Methoden

- SEC02-BP01 Verwenden von starken Anmeldemechanismen
- SEC02-BP02 Verwenden von temporären Anmeldeinformationen
- SEC02-BP03 Sicheres Speichern und Verwenden von Secrets
- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter
- SEC02-BP05 Regelmäßiges Überprüfen und Rotieren von Anmeldeinformationen
- SEC02-BP06 Nutzen von Benutzergruppen und Attributen

SEC02-BP01 Verwenden von starken Anmeldemechanismen

Anmeldungen (die Authentifizierung unter Verwendung von Anmeldeinformationen) kann risikobehaftet sein, wenn nicht Mechanismen wie die Multi-Faktor-Authentifizierung (MFA) verwendet werden, besonders in Situationen, in denen Anmeldeinformationen unbeabsichtigt offengelegt wurden oder leicht zu erraten sind. Verwenden Sie starke Anmeldemechanismen in Form von MFA und Richtlinien für sichere Passwörter, um diese Risiken zu reduzieren.

Gewünschtes Ergebnis: Senkung des Risikos unbeabsichtigter Zugriffe auf Anmeldeinformationen in AWS durch die Verwendung starker Anmeldemechanismen für AWS Identity and Access Management (IAM)-Benutzer, den Root-Benutzer des AWS-Konto, AWS IAM Identity Center (Nachfolger von AWS Single Sign-On) und externe Identitätsanbieter. Dies bedeutet das Erfordern von MFA, das Durchsetzen von Richtlinien zur Verwendung starker Passwörter und das Erkennen anomaler Anmeldeverhaltensweisen.

Typische Anti-Muster:

- keine Durchsetzung einer Richtlinie zur Verwendung starker Passwörter für Ihre Identitäten, einschließlich komplexer Passwörter und MFA.
- gemeinsame Nutzung derselben Anmeldeinformationen durch mehrere Benutzer.
- keine Verwendung von Kontrollmechanismen für verdächtige Anmeldevorgänge.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Es gibt viele Möglichkeiten zur Anmeldung für menschliche Identitäten bei AWS. Eine bewährte AWS-Methode besteht darin, einen zentralisierten Identitätsanbieter mit Verbundverfahren (direkter Verbund oder unter Verwendung von AWS IAM Identity Center) für die Authentifizierung bei AWS zu verwenden. In diesem Fall sollten Sie einen sicheren Anmeldeprozess mit Ihrem Identitätsanbieter oder Microsoft Active Directory einrichten.

Wenn Sie ein AWS-Konto zum ersten Mal einrichten, beginnen Sie mit einem Root-Benutzer für das AWS-Konto. Sie sollten den Root-Benutzer des Kontos nur zur Einrichtung des Zugriffs für Ihre Benutzer (und für Aufgaben, die den Root-Benutzer erfordern) verwenden. Es ist wichtig, MFA für den Root-Benutzer des Kontos sofort nach der Einrichtung Ihres AWS-Konto zu aktivieren, und den Root-Benutzer anhand der Anleitung zu bewährten Methoden von AWS zu schützen.

Wenn Sie in AWS IAM Identity Center Benutzer erstellen, dann sollten Sie auch den Anmeldeprozess in diesem Service schützen. Für Verbraucheridentitäten können Sie Amazon Cognito user pools verwenden und den Anmeldeprozess in diesem Service schützen oder indem Sie einen der von Amazon Cognito user pools unterstützten Identitätsanbieter verwenden.

Wenn Sie AWS Identity and Access Management (IAM)-Benutzer verwenden, schützen Sie den Anmeldeprozess mit IAM.

Unabhängig vom Anmeldeverfahren ist es wichtig, eine strenge Anmelderichtlinie durchzusetzen.

Implementierungsschritte

Es folgen allgemeine Empfehlungen für starke Anmeldeverfahren. Die tatsächlich konfigurierten Einstellungen sollten von Ihrer Unternehmensrichtlinie oder von einem Standard wie NIST 800-63 vorgegeben werden.

- Setzen Sie MFA voraus. Ein bewährtes IAM-Verfahren besteht darin, MFA für menschliche Identitäten und Workloads vorauszusetzen. Die Aktivierung von MFA bietet eine zusätzliche Sicherheitsebene, die verlangt, dass Benutzer Anmeldeinformationen und ein Einmalpasswort (OTP) oder eine kryptographisch verifizierte und generierte Zeichenfolge von einem Hardware-Gerät vorlegen.
- Verlangen Sie eine Mindestlänge für Passwörter als primären Faktor für die Passwortstärke.
- Verlangen Sie Passwortkomplexität, um das Erraten von Passwörtern zu erschweren.
- Erlauben Sie Benutzern, Ihr eigenes Passwort zu ändern.

- Erstellen Sie individuelle Identitäten anstelle gemeinsam genutzter Anmeldeinformationen. Durch das Erstellen individueller Identitäten können Sie jedem Benutzer einen einmaligen Satz mit Sicherheitsanmeldeinformationen zuweisen. Individuelle Benutzer bieten die Möglichkeit, die Aktivität der einzelnen Benutzer zu prüfen.

Empfehlungen für IAM Identity Center:

- Bei Verwendung des Standardverzeichnisses bietet IAM Identity Center eine vordefinierte Passwortrichtlinie, die die Passwortlänge, -komplexität und die Anforderungen im Zusammenhang mit der erneuten Verwendung festlegt.
- Aktivieren Sie MFA und konfigurieren Sie die kontextsensitive oder ständig aktive Einstellung für MFA, wenn die Identitätsquelle das Standardverzeichnis, AWS Managed Microsoft AD oder AD Connector ist.
- Erlauben Sie Benutzern die Registrierung ihrer eigenen MFA-Geräte.

Verzeichnisempfehlungen für Amazon Cognito user pools:

- Konfigurieren Sie die Einstellungen für die Passwortstärke.
- Verlangen Sie MFA für Benutzer.
- Verwenden Sie die erweiterten Sicherheitseinstellungen von Amazon Cognito user pools für Funktionen wie die adaptive Authentifizierung, die verdächtige Anmeldeversuche blockieren können.

IAM-Benutzerempfehlungen:

- Idealerweise verwenden Sie IAM Identity Center oder den direkten Verbund. Möglicherweise benötigen Sie aber auch IAM-Benutzer. Richten Sie in diesem Fall eine Passwortrichtlinie für IAM-Benutzer ein. Sie können die Passwortrichtlinie verwenden, um Anforderungen wie Mindestlänge zu definieren oder ob das Passwort nicht-alphanumerische Zeichen beinhalten sollte.
- Erstellen Sie eine IAM-Richtlinie, um die MFA-Anmeldung zu erzwingen, damit Benutzer ihre eigenen Passwörter und MFA-Geräte verwalten können.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP03 Sicheres Speichern und Verwenden von Secrets
- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter
- SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation

Zugehörige Dokumente:

- AWS IAM Identity Center (successor to AWS Single Sign-On) Password Policy (Passwortrichtlinie von AWS IAM Identity Center (Nachfolger von AWS Singe Sign-On))
- IAM-Benutzer-Passwortrichtlinie
- Setting the AWS-Konto root user password (Einrichten des Root-Benutzerpassworts für das AWS-Konto)
- Amazon Cognito-Passwortrichtlinie
- AWS-Anmeldeinformationen
- Bewährte Methoden für die Sicherheit in IAM

Zugehörige Videos:

- Managing user permissions at scale with AWS IAM Identity Center (Verwalten von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center)
- Mastering identity at every layer of the cake (Beherrschen der Identität auf jeder Ebene)

SEC02-BP02 Verwenden von temporären Anmeldeinformationen

Bei Authentifizierungen jeder Art, sollten am besten temporäre anstelle langfristiger Anmeldeinformationen verwendet werden, um Risiken zu reduzieren oder zu eliminieren, etwa durch die unbeabsichtigte Offenlegung, die Weitergabe oder den Diebstahl von Anmeldeinformationen.

Gewünschtes Ergebnis: Senkung des Risikos im Zusammenhang mit langfristigen Anmeldeinformationen durch die Verwendung temporärer Anmeldeinformationen, wo immer dies für menschliche und maschinelle Identitäten möglich ist. Langfristige Anmeldeinformationen sind mit vielen Risiken verbunden, so kann es beispielsweise vorkommen, dass sie in Code in öffentliche GitHub-Repositorys hochgeladen werden. Durch die Verwendung temporärer Anmeldeinformationen können Sie die Gefahr der Kompromittierung von Anmeldeinformationen deutlich senken.

Typische Anti-Muster:

- Entwickler verwenden langfristige Zugriffsschlüssel von IAM users, anstatt sich temporäre Anmeldeinformationen per Verbund von der CLI zu beschaffen.
- Entwickler betten langfristige Zugriffsschlüssel in ihren Code ein und laden diese in öffentliche Git-Repositorys hoch.
- Entwickler betten langfristige Zugriffsschlüssel in Mobil-Apps ein, die dann in App-Stores verfügbar gemacht werden.
- Benutzer geben langfristige Zugriffsschlüssel an andere Benutzer weiter, oder Mitarbeiter verlassen das Unternehmen und besitzen weiterhin langfristige Zugriffsschlüssel.
- Verwendung langfristiger Zugriffsschlüssel für Maschinenidentitäten, obwohl temporäre Anmeldeinformationen verwendet werden könnten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Verwenden Sie temporäre anstelle langfristiger Anmeldeinformationen für alle AWS-API- und -CLI-Anfragen. API- und CLI-Anfragen an AWS müssen in fast jedem Fall mit AWS-Zugriffsschlüsseln signiert werden. Diese Anfragen können mit temporären oder langfristigen Anmeldeinformationen signiert werden. Sie sollten langfristige Anmeldeinformationen (bzw. Zugriffsschlüssel) nur nutzen, wenn Sie einen IAM-Benutzer oder den Root-Benutzer des AWS-Konto verwenden. Wenn Sie einen Verbund mit AWS nutzen oder eine IAM-Rolle über andere Methoden annehmen, werden temporäre Anmeldeinformationen generiert. Selbst wenn Sie mit Anmeldeinformationen auf die AWS Management Console zugreifen, werden für Sie temporäre Anmeldeinformationen für Aufrufe von AWS-Services generiert. Es gibt nur wenige Situationen, in denen Sie langfristige Anmeldeinformationen benötigen, und fast alle Aufgaben lassen sich mit temporären Anmeldeinformationen erledigen.

Das Vermeiden der Verwendung langfristiger zugunsten temporärer Anmeldeinformationen sollte von einer Strategie zur Reduzierung der Verwendung von IAM-Benutzern gegenüber Verbundverfahren und IAM-Rollen begleitet werden. Zwar wurden früher IAM-Benutzer für menschliche und maschinelle Identitäten verwendet, wir empfehlen heute jedoch, dies nicht mehr zu tun, um die mit der Verwendung langfristiger Zugriffsschlüssel verbundenen Risiken auszuschalten.

Implementierungsschritte

Für menschliche Identitäten wie Mitarbeiter, Administratoren, Entwickler, Bediener und Kunden:

- Sie sollten einen zentralisierten Identitätsanbieter nutzen und von menschlichen Benutzern die Verwendung von Verbundverfahren mit einem Identitätsanbieter verlangen, damit mit temporären Anmeldeinformationen auf AWS zugegriffen wird. Ein Verbund für Ihre Benutzer kann per direktem Verbund zu jedem AWS-Konto oder mit AWSIAM Identity Center (Nachfolger von AWS IAM Identity Center) und dem Identitätsanbieter Ihrer Wahl erreicht werden. Ein Verbund bietet eine Reihe von Vorteilen gegenüber der Verwendung von IAM-Benutzern und eliminiert langfristige Anmeldeinformationen. Ihre Benutzer können auch temporäre Anmeldeinformationen aus der Befehlszeile für einen direkten Verbund oder mit IAM Identity Center anfordern. Dies bedeutet, dass es nur wenige Anwendungsfälle gibt, für die IAM-Benutzer oder langfristige Anmeldeinformationen für Ihre Benutzer erforderlich sind.
- Wenn Dritten, wie beispielsweise Anbietern von Software as a Service (SaaS), der Zugriff auf Ressourcen in Ihrem AWS-Konto gewährt wird, können Sie kontoübergreifende Rollen und ressourcenbasierende Richtlinien verwenden.
- Wenn Sie Verbraucheranwendungen oder Kunden Zugriff auf Ihre AWS-Ressourcen gewähren müssen, können Sie Amazon Cognito-Identitätspools oder Amazon Cognito user pools verwenden, um temporäre Anmeldeinformationen bereitzustellen. Die Berechtigungen für die Anmeldeinformationen werden über IAM-Rollen konfiguriert. Sie können auch eine separate IAM-Rolle mit eingeschränkten Berechtigungen für Gastbenutzer definieren, die nicht authentifiziert sind.

Für Maschinenidentitäten müssen Sie möglicherweise langfristige Anmeldeinformationen verwenden. In solchen Fällen sollten Sie verlangen, dass Workloads temporäre Anmeldeinformationen mit IAM-Rollen zum Zugriff auf AWS verwenden.

- Für Amazon Elastic Compute Cloud (Amazon EC2) können Sie Rollen für Amazon EC2 verwenden.

- AWS Lambda ermöglicht die Konfiguration einer Lambda-Ausführungsrolle, um dem Service Berechtigungen zum Ausführen von AWS-Aktionen unter Verwendung temporärer Anmeldeinformationen zu erteilen. Es gibt zahlreiche ähnliche Modelle für AWS-Services zum Gewähren temporärer Anmeldeinformationen mit IAM-Rollen.
- Für IoT-Geräte können Sie den Anmeldeinformationenanbieter von AWS IoT Core zur Anfrage nach temporären Anmeldeinformationen verwenden.
- Für On-Premises-Systeme oder außerhalb von AWS ausgeführte Systeme, die Zugriff auf AWS-Ressourcen benötigen, können Sie IAM Roles Anywhere verwenden.

Es gibt Szenarien, in denen temporäre Anmeldeinformationen nicht in Frage kommen und stattdessen langfristige Anmeldeinformationen verwendet werden müssen. In solchen Fällen sollten Sie die Anmeldeinformationen regelmäßig prüfen und rotieren sowie die Zugriffsschlüssel für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern, regelmäßig wechseln. Beispiele, bei denen langfristige Anmeldeinformationen erforderlich sind, sind etwa WordPress-Plugins und AWS-Clients von Drittanbietern. In Situationen, die langfristige Anmeldeinformationen erfordern, oder für andere Anmeldeinformationen als AWS-Zugriffsschlüssel, wie z. B. Datenbankanmeldungen, können Sie einen Service verwenden, der für die Verwaltung von Secrets gedacht ist, wie etwa AWS Secrets Manager. Secrets Manager erleichtert die Verwaltung, das Rotieren und die Speicherung verschlüsselter Secrets unter Verwendung unterstützter Services. Weitere Informationen zur Rotation langfristiger Anmeldeinformationen finden Sie unter Rotation von Zugriffsschlüsseln.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP03 Sicheres Speichern und Verwenden von Secrets
- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter
- SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation

Zugehörige Dokumente:

- Temporäre Sicherheits-Anmeldeinformationen
- AWS-Anmeldeinformationen
- Bewährte Methoden für die Sicherheit in IAM
- IAM-Rollen
- IAM Identity Center
- Identitätsanbieter und Verbund
- Rotieren der Zugriffsschlüssel
- Partnerlösungen im Bereich Sicherheit: Zugriff und Zugriffssteuerung
- Der Root-Benutzer des AWS-Kontos

Zugehörige Videos:

- Managing user permissions at scale with AWS IAM Identity Center (successor to AWS IAM Identity Center) (Verwalten von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center (Nachfolger von AWS IAM Identity Center))
- Mastering identity at every layer of the cake (Beherrschen der Identität auf jeder Ebene)

SEC02-BP03 Sicheres Speichern und Verwenden von Secrets

Ein Workload muss seine Identität automatisch gegenüber Datenbanken, Ressourcen und Services von Drittanbietern authentifizieren können. Dazu dienen geheime Zugriffsanmeldeinformationen wie etwa API-Zugriffsschlüssel, Passwörter und OAuth-Tokens. Die Verwendung eines dedizierten Services zur Speicherung, Verwaltung und Rotation der Anmeldeinformationen hilft dabei, die Gefahr der Kompromittierung dieser Anmeldeinformationen zu verringern.

Gewünschtes Ergebnis: Implementierung eines Mechanismus für die sichere Verwaltung von Anwendungsanmeldeinformationen, der die folgenden Ziele erreicht:

- Identifikation der für den Workload erforderlichen Secrets
- Reduzierung der Anzahl der erforderlichen langfristigen Anmeldeinformationen durch ihren Austausch gegen kurzfristige Anmeldeinformationen, wo dies möglich ist
- Einrichtung der sicheren Speicherung und der automatischen Rotation der verbleibenden langfristigen Anmeldeinformationen
- Überwachung des Zugriffs auf in dem Workload vorhandene Secrets
- Kontinuierliche Überwachung, um sicherzustellen, dass im Rahmen des Entwicklungsprozesses keine Secrets in den Quellcode eingebettet werden
- Reduzieren der Gefahr unbeabsichtigter Offenlegungen von Anmeldeinformationen

Typische Anti-Muster:

- keine rotierenden Anmeldeinformationen
- Speichern langfristiger Anmeldeinformationen in Quellcode oder Konfigurationsdateien
- Speichern von Anmeldeinformationen im Ruhezustand ohne Verschlüsselung

Vorteile der Nutzung dieser bewährten Methode:

- Secrets werden im Ruhezustand und in Übertragung verschlüsselt gespeichert.

- Organisation des Zugriffs auf Anmeldeinformationen über eine API (vorstellbar als Automat für Anmeldeinformationen)
- Prüfung und Protokollierung des Zugriffs (Lese- und Schreibzugriff) auf Anmeldeinformationen
- Trennung möglicher Problemquellen: Die Rotation der Anmeldeinformationen wird von einer separaten Komponente vorgenommen, die vom Rest der Architektur isoliert werden kann.
- Secrets werden automatisch bei Bedarf an Softwarekomponenten verteilt und die Rotation erfolgt an einem zentralen Ort.
- Der Zugriff auf Anmeldeinformationen kann detailliert kontrolliert werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Früher wurden Anmeldeinformationen für die Authentifizierung bei Datenbanken, APIs von Dritten, Tokens und andere Secrets möglicherweise in eingebettetem Quellcode oder in Umgebungsdateien gespeichert. AWS bietet mehrere Mechanismen, um diese Anmeldeinformationen sicher zu speichern, sie automatisch zu rotieren und ihre Verwendung zu prüfen.

Das beste Verfahren für die Verwaltung von Secrets besteht besteht darin, den Anweisungen zum Entfernen, Ersetzen und Rotieren zu folgen. Die sichersten Anmeldeinformationen sind diejenigen, die Sie nicht speichern, verwalten oder handhaben müssen. Möglicherweise gibt es Anmeldeinformationen, die für die Funktion des Workloads nicht mehr benötigt werden und sicher entfernt werden können.

Bei Anmeldeinformationen, die für die korrekte Funktion des Workloads weiterhin benötigt werden, besteht die Möglichkeit, langfristige Anmeldeinformationen durch temporäre oder kurzfristige zu ersetzen. So könnten Sie beispielsweise anstelle der Hartkodierung eines geheimen AWS-Zugriffsschlüssels diese langfristige Anmeldeinformation durch eine temporäre unter Verwendung von IAM-Rollen ersetzen.

Manche langfristigen Secrets können möglicherweise nicht entfernt oder ersetzt werden. Diese Secrets können in einem Service wie AWS Secrets Manager gespeichert werden, wo sie zentral aufbewahrt, verwaltet und regelmäßig rotiert werden.

Eine Prüfung des Quellcodes und der Konfigurationsdateien des Workloads kann verschiedene Arten von Anmeldeinformationen zutage fördern. Die folgende Tabelle fasst Strategien für den Umgang mit verbreiteten Arten von Anmeldeinformationen zusammen:

Credential type	Description	Suggested strategy
IAM access keys	AWS IAM access and secret keys used to assume IAM roles inside of a workload	Replace: Use IAM-Rollen assigned to the compute instances (such as Amazon EC2 or AWS Lambda) instead. For interoperability with third parties that require access to resources in your AWS-Konto , ask if they support Kontoüber greifender AWS-Zugriff. For mobile apps, consider using temporary credentials through Amazon Cognito-Identitäts pools (Verbundidentitäten). For workloads running outside of AWS, consider IAM Roles Anywhere or AWS Systems Manager Hybride Aktivieru ngen.
SSH keys	Secure Shell private keys used to log into Linux EC2 instances, manually or as part of an automated process	Replace: Use AWS Systems Manager or EC2 Instance Connect to provide programmatic and human access to EC2 instances using IAM roles.
Application and database credentials	Passwords – plain text string	Rotate: Store credentials in AWS Secrets Manager and establish automated rotation if possible.
Amazon RDS and Aurora Admin Database credentials	Passwords – plain text string	Replace: Use the Secrets Manager-Integration mit Amazon RDS or Amazon Aurora. In addition, some RDS

Credential type	Description	Suggested strategy
		database types can use IAM roles instead of passwords for some use cases (for more detail, see IAM-Daten bankauthentifizierung).
OAuth tokens	Secret tokens – plain text string	Rotate: Store tokens in AWS Secrets Manager and configure automated rotation.
API tokens and keys	Secret tokens – plain text string	Rotate: Store in AWS Secrets Manager and establish automated rotation if possible.

Ein typisches Anti-Muster ist die Einbettung von IAM-Zugriffsschlüsseln in Quellcode, Konfigurationsdateien oder Mobil-Apps. Wenn ein IAM-Zugriffsschlüssel für die Kommunikation mit einem AWS-Service erforderlich ist, verwenden Sie temporäre (kurzfristige) Sicherheitsanmeldeinformationen. Diese kurzfristigen Anmeldeinformationen können über IAM-Rollen für EC2-Instances, Ausführungsrollen für Lambda-Funktionen, Cognito-IAM-Rollen für den mobilen Benutzerzugriff und IoT-Core-Richtlinien für IoT-Geräte bereitgestellt werden. Bei Verbindungen mit Drittparteien sollten Sie den Zugriff lieber über eine IAM-Rolle mit dem erforderlichen Zugriff auf die Ressourcen Ihres Kontos delegieren, anstatt einen IAM-Benutzer zu konfigurieren und der Drittpartei den geheimen Zugriffsschlüssel für diesen Benutzer zuzusenden.

Es gibt viele Fälle, in denen der Workload die Speicherung von Secrets erfordert, um mit anderen Services und Ressourcen zusammenwirken zu können. AWS Secrets Manager wurde speziell entwickelt, um solche Anmeldeinformationen sowie die Speicherung, Verwendung und Rotation von API-Tokens, Passwörtern und anderer Anmeldeinformationen sicher zu handhaben.

AWS Secrets Manager bietet fünf entscheidende Funktionen, die für die sichere Speicherung und Handhabung sensitiver Anmeldeinformationen sorgen: Verschlüsselung im Ruhezustand, Verschlüsselung in Übertragung, Umfassende Prüfungen, detaillierte Zugriffssteuerung und erweiterbare Rotation von Anmeldeinformationen. Andere Secret-Managementservices von AWS-Partnern oder lokal entwickelte Lösungen mit ähnlichen Funktionen und Sicherungen sind ebenfalls akzeptabel.

Implementierungsschritte

1. Identifizieren Sie Code-Pfade mit hartkodierten Anmeldeinformationen mithilfe automatisierter Tools wie etwa Amazon CodeGuru.
   - Scannen Sie Ihre Code-Repositorys mit Amazon CodeGuru. Sobald die Prüfung abgeschlossen ist, filtern sie nach `Type=Secrets` in CodeGuru, um problematische Codezeilen zu finden.
2. Identifizieren Sie Anmeldeinformationen, die entfernt oder ersetzt werden können.
   a. Identifizieren Sie Anmeldeinformationen, die nicht mehr benötigt werden, und markieren Sie sie zum Entfernen.
   b. Ersetzen Sie AWS-Geheimschlüssel, die in Quellcode eingebettet sind, durch IAM-Rollen, die mit den erforderlichen Ressourcen verbunden sind. Wenn sich ein Teil Ihres Workloads außerhalb von AWS befindet, er jedoch IAM-Anmeldeinformationen für den Zugriff auf AWS-Ressourcen benötigt, können Sie IAM Roles Anywhere oder AWS Systems Manager Hybride Aktivierungen verwenden.
3. Integrieren Sie für andere langfristige Secrets von Dritten, die die Rotationsstrategie erfordern, Secrets Manager in Ihren Code, um die externen Secrets zur Laufzeit abzurufen.
   a. Die CodeGuru-Konsole kann automatisch ein Secret in Secrets Manager unter Verwendung der erkannten Anmeldeinformationen erstellen.
   b. Integrieren Sie den Secret-Abruf von Secrets Manager in Ihren Anwendungscode.
      - Serverless-Lambda-Funktionen können eine sprachneutrale Lambda-Erweiterung verwenden.
      - Für EC2-Instances oder Container bietet AWS clientseitigen Beispielcode für den Abruf von Secrets von Secrets Manager in verschiedenen verbreiteten Programmiersprachen.
4. Prüfen Sie Ihre Codebasis regelmäßig und wiederholen Sie dies, um sicherzustellen, dass dem Code keine neuen Secrets hinzugefügt wurden.
   - Erwägen Sie die Verwendung eines Tools wie etwa git-secrets, um zu vermeiden, dass neue Secrets in Ihr Quellcode-Repository eingebracht werden.
5. Überwachen Sie die Secrets Manager-Aktivität auf Anzeichen für unerwartete Nutzungen, den unautorisierten Zugriff auf Secrets oder versuche, Secrets zu löschen.
6. Reduzieren Sie menschliche Interaktionen mit Anmeldeinformationen. Schränken Sie den Zugriff zum Lesen, Schreiben und Ändern von Anmeldeinformationen auf eine für diesen Zweck dedizierte IAM-Rolle ein und erlauben Sie die Übernahme dieser Rolle nur einem kleinen Teil der betrieblichen Nutzer.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP02 Verwenden von temporären Anmeldeinformationen
- SEC02-BP05 Regelmäßiges Überprüfen und Rotieren von Anmeldeinformationen

Zugehörige Dokumente:

- Erste Schritte mit AWS Secrets Manager
- Identitätsanbieter und Verbund
- Amazon CodeGuru Introduces Secrets Detector (Amazon CodeGuru stellt Secrets Detector vor)
- How AWS Secrets Manager uses AWS Key Management Service (Wie AWS Secrets Manager AWS Key Management Service verwendet)
- Secret encryption and decryption in Secrets Manager (Secret-Ver- und Entschlüsselung in Secrets Manager)
- Blog-Einträge zu Secrets Manager
- Amazon RDS announces integration with AWS Secrets Manager (Amazon RDS kündigt Integration mit AWS Secrets Manager an)

Zugehörige Videos:

- Best Practices for Managing, Retrieving, and Rotating Secrets at Scale (Bewährte Methoden zum Verwalten, Abrufen und Rotieren von Secrets in großem Umfang)
- Find Hard-Coded Secrets Using Amazon CodeGuru Secrets Detector (Finden hartkodierter Secrets mit CodeGuru Secrets Detector)
- Securing Secrets for Hybrid Workloads Using AWS Secrets Manager (Sichern von Secrets für hybride Workloads mit AWS Secrets Manager)

Zugehörige Workshops:

- Store, retrieve, and manage sensitive credentials in AWS Secrets Manager (Speichern, Abrufen und verwalten sensitiver Anmeldeinformationen in AWS Secrets Manager)
- AWS Systems Manager Hybride Aktivierungen

SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter

Verlassen Sie sich im Zusammenhang mit Identitäten für Ihre Belegschaft (Mitarbeiter und Auftragnehmer) auf einen Identitätsanbieter, mit dem Sie Identitäten zentral verwalten können. Dadurch ist es einfacher, den Zugriff über mehrere Anwendungen und Systeme hinweg zu verwalten, da Sie den Zugriff von einem einzigen Standort aus erstellen, zuweisen, verwalten, widerrufen und überwachen.

Gewünschtes Ergebnis: Sie verfügen über einen zentralen Identitätsanbieter, mit dem Sie Benutzer im Unternehmen, Authentifizierungsrichtlinien (z. B. die Anforderung einer Multi-Faktor-Authentifizierung, MFA) und die Autorisierung für Systeme und Anwendungen zentral verwalten (z. B. die Zuweisung von Zugriffsberechtigungen auf Grundlage der Gruppenmitgliedschaft oder der Attribute eines Benutzers). Die Benutzer in Ihrer Belegschaft melden sich beim zentralen Identitätsanbieter an und bilden einen Verbund (Single Sign-On) mit internen und externen Anwendungen, sodass sich die Benutzer nicht mehrere Anmeldeinformationen merken müssen. Ihr Identitätsanbieter ist in Ihre Personalverwaltungssysteme integriert, sodass Personaländerungen automatisch mit Ihrem Identitätsanbieter synchronisiert werden. Wenn beispielsweise jemand Ihr Unternehmen verlässt, können Sie den Zugriff auf alle Anwendungen und Systeme im Verbund (einschließlich AWS) widerrufen. Sie haben die detaillierte Auditprotokollierung in Ihrem Identitätsanbieter aktiviert und überwachen diese Protokolle auf ungewöhnliches Benutzerverhalten.

Typische Anti-Muster:

- Sie verwenden keinen Verbund mit Single-Sign-On. Die Benutzer in Ihrer Belegschaft erstellen separate Benutzerkonten und Anmeldeinformationen für mehrere Anwendungen und Systeme.
- Sie haben den Lebenszyklus von Identitäten für Benutzer in Ihrer Belegschaft nicht automatisiert, indem Sie beispielsweise Ihren Identitätsanbieter in Ihre Personalverwaltungssysteme integriert haben. Wenn ein Benutzer Ihre Organisation verlässt oder die Position wechselt, folgen Sie einem manuellen Prozess, um seine Datensätze in mehreren Anwendungen und Systemen zu löschen oder zu aktualisieren.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung eines zentralen Identitätsanbieters haben Sie die Möglichkeit, Benutzeridentitäten und Richtlinien für Ihre Mitarbeiter von einem zentralen Ort aus zu verwalten, Benutzern und Gruppen Zugriff auf Anwendungen zuzuweisen und die Anmeldeaktivitäten der Benutzer zu überwachen. Wenn ein Benutzer die Position wechselt, werden durch die Integration in Ihre Personalverwaltungssysteme Änderungen mit dem Identitätsanbieter synchronisiert und die ihm zugewiesenen Anwendungen und Berechtigungen werden automatisch aktualisiert. Wenn ein Benutzer Ihre Organisation verlässt, wird seine Identität

automatisch im Identitätsanbieter deaktiviert, wodurch ihm der Zugriff auf Anwendungen und Systeme im Verbund entzogen wird.

Risikostufe bei fehlender Befolgung dieser Best Practice:: Hoch

Implementierungsleitfaden

Leitfaden für Benutzer im Unternehmen, die auf AWS zugreifen

Benutzer in Ihrer Belegschaft, z. B. Mitarbeiter und Auftragnehmer in Ihrer Organisation, benötigen möglicherweise Zugriff auf AWS über die AWS Management Console oder AWS Command Line Interface (AWS CLI), um ihre Aufgaben auszuführen. Sie können diesen Benutzern Zugriff auf AWS gewähren, indem Sie einen Verbund von Ihrem zentralen Identitätsanbieter zu AWS auf zwei Ebenen einrichten: ein direkter Verbund mit jedem AWS-Konto oder ein Verbund mit mehreren Konten in Ihrem AWS Unternehmen.

- Um die Benutzer in Ihrem Unternehmen direkt mit jedem AWS-Konto zu verbinden, können Sie einen zentralen Identitätsanbieter für den Verbund mit AWS Identity and Access Management in diesem Konto verwenden. Die Flexibilität von IAM ermöglicht es Ihnen, einen separaten SAML 2.0- oder Open ID Connect (OIDC)- Identitätsanbieter für jedes AWS-Konto zu aktivieren und Verbundbenutzerattribute für die Zugriffskontrolle zu verwenden. Die Benutzer in Ihrer Belegschaft verwenden ihren Webbrowser, um sich beim Identitätsanbieter anzumelden, indem sie ihre Anmeldeinformationen (wie Passwörter und MFA-Tokencodes) angeben. Der Identitätsanbieter gibt eine SAML-Zusicherung an den Browser aus, die an die Anmelde-URL der AWS Management Console gesendet wird. Dies ermöglicht den Benutzern das Single Sign-On (SSO) bei der AWS Management Console, indem sie eine IAM-Rolle annehmen. Ihre Benutzer können außerdem temporäre AWS-API-Anmeldeinformationen für die Verwendung in der AWS CLI oder AWS SDKs von AWS STS erhalten, indem sie die IAM-Rolle mit einer SAML-Zusicherung des Identitätsanbieters annehmen.
- Für den Verbund der Benutzer in Ihrer Belegschaft mit mehreren Konten in Ihrer AWS-Organisation können Sie AWS IAM Identity Center verwenden und damit den Zugriff für Ihre Belegschaftsbenutzer auf AWS-Konten und Anwendungen zentral verwalten. Sie aktivieren Identity Center für Ihre Organisation und konfigurieren Ihre Identitätsquelle. IAM Identity Center stellt ein Standard-Identitätsquellenverzeichnis bereit, mit dem Sie Ihre Benutzer und Gruppen verwalten können. Alternativ können Sie eine externe Identitätsquelle auswählen, indem Sie eine Verbindung mit Ihrem externen Identitätsanbieter über SAML 2.0 herstellen und automatisch Benutzer und Gruppen mit SCIM bereitstellen oder eine Verbindung zu Ihrem Microsoft AD-Verzeichnis mit AWS Directory Serviceherstellen. Sobald eine Identitätsquelle konfiguriert wurde,

können Sie Benutzern und Gruppen Zugriff auf AWS-Konten zuweisen, indem Sie Richtlinien nach dem Prinzip der geringsten Berechtigungen in Ihrem Berechtigungssatzdefinieren. Die Benutzer in Ihrer Belegschaft können sich über Ihren zentralen Identitätsanbieter authentifizieren, um sich beim AWS-Zugangsportal anzumelden. Außerdem können sie sich so per Single-Sign-On bei den AWS-Konten und Cloud-Anwendungen anmelden, die ihnen zugewiesen sind. Ihre Benutzer können AWS CLI v2 konfigurieren, um sich bei Identity Center zu authentifizieren und Anmeldeinformationen für die Ausführung von AWS CLI-Befehlen zu erhalten. Identity Center ermöglicht außerdem den Single-Sign-On-Zugriff auf AWS-Anwendungen wie Amazon SageMaker AI Studio und AWS IoT Sitewise Monitor-Portale.

Nachdem Sie die obigen Anweisungen befolgt haben, müssen die Benutzer in Ihrer Belegschaft bei der Verwaltung von Workloads in AWS für den normalen Betrieb keine IAM users und -Gruppen mehr verwenden. Stattdessen werden Ihre Benutzer und Gruppen außerhalb von AWS verwaltet und Benutzer können auf AWS-Ressourcen als Identitätsverbundzugreifen. Bei einem Identitätsverbund werden die Gruppen verwendet, die von Ihrem zentralen Identitätsanbieter definiert wurden. Sie sollten IAM-Gruppen, IAM users und langlebige Benutzeranmeldeinformationen (Passwörter und Zugriffsschlüssel) identifizieren und entfernen, die in Ihren AWS-Konten nicht mehr benötigt werden. Sie können ungenutzte Anmeldeinformationen mit IAM-Berichten zu Anmeldeinformationen suchen, die entsprechenden IAM users löschen und IAM-Gruppen entfernen. Sie können eine Service-Kontrollrichtlinie (SCP) auf Ihre Organisation anwenden, mit der das Erstellen neuer IAM users und -Gruppen verhindert und erzwungen wird, dass der Zugriff auf AWS über Verbundidentitäten erfolgt.

Leitfaden für Benutzer Ihrer Anwendungen

Sie können die Identitäten der Benutzer Ihrer Anwendungen, z. B. einer mobilen App, mithilfe von Amazon Cognito als zentralem Identitätsanbieter verwalten. Amazon Cognito ermöglicht die Authentifizierung, Autorisierung und Benutzerverwaltung für Ihre Web- und mobilen Apps. Amazon Cognito bietet einen Identitätsspeicher, der auf Millionen von Benutzern skaliert werden kann, unterstützt den Identitätsverbund für soziale Netzwerke und Unternehmen und bietet erweiterte Sicherheitsfunktionen zum Schutz Ihrer Benutzer und Ihres Unternehmens. Sie können Ihre benutzerdefinierte Web- oder Mobilanwendung in Amazon Cognito integrieren, um Ihren Anwendungen innerhalb von Minuten Benutzerauthentifizierung und Zugriffskontrolle hinzuzufügen. Amazon Cognito basiert auf offenen Identitätsstandards wie SAML und Open ID Connect (OIDC), unterstützt verschiedene Compliance-Vorschriften und lässt sich in Frontend- und Backend-Entwicklungsressourcen integrieren.

Implementierungsschritte

Schritte für Benutzer im Unternehmen, die auf AWS zugreifen

- Erstellen Sie für die Benutzer in Ihrer Belegschaft unter Verwendung eines zentralen Identitätsanbieters einen Verbund mit AWS. Nutzen Sie dabei einen der folgenden Ansätze:
  - Verwenden Sie IAM Identity Center, um Single Sign-On für mehrere AWS-Konten in Ihrer AWS-Organisation zu aktivieren, indem Sie einen Verbund mit Ihrem Identitätsanbieter erstellen.
  - Verwenden Sie IAM, um Ihren Identitätsanbieter direkt mit jedem AWS-Konto zu verbinden und so einen differenzierten Verbundzugriff zu ermöglichen.
- Identifizieren und entfernen Sie IAM users und -Gruppen, die durch Verbundidentitäten ersetzt werden.

Schritte für Benutzer Ihrer Anwendungen

- Verwenden Sie Amazon Cognito als zentralen Identitätsanbieter für Ihre Anwendungen.
- Integrieren Sie Ihre benutzerdefinierten Anwendungen mithilfe von OpenID Connect und OAuth mit Amazon Cognito. Sie können Ihre benutzerdefinierten Anwendungen mithilfe der Amplify-Bibliotheken entwickeln, die einfache Schnittstellen für die Integration in eine Vielzahl von AWS-Services bieten, z. B. Amazon Cognito für die Authentifizierung.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- SEC02-BP06 Nutzen von Benutzergruppen und Attributen
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen
- SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus

Zugehörige Dokumente:

- AWS-Identitätsverbund
- Bewährte Sicherheitsmethoden in IAM
- Bewährte Methoden für AWS Identity and Access Management
- Getting started with IAM Identity Center delegated administration (Erste Schritte mit der delegierten IAM Identity Center-Verwaltung)

- How to use customer managed policies in IAM Identity Center for advanced use cases (Verwenden von vom Kunden verwalteten Richtlinien in IAM Identity Center für fortgeschrittene Anwendungsfälle)
- AWS CLI v2: IAM Identity Center-Anbieter für Anmeldeinformationen

Zugehörige Videos:

- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive (AWS re:inforce 2022 – AWS Identity and Access Management (IAM) zur Vertiefung)
- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center (AWS re:Invent 2022 – Vereinfachen des vorhandenen Mitarbeiterzugriffs mit IAM Identity Center)
- AWS re:Invent 2018: Mastering identity at every layer of the cake (AWS re:Invent 2018: Beherrschen der Identität auf jeder Ebene)

Zugehörige Beispiele:

- Workshop: Using AWS IAM Identity Center to achieve strong identity management (Verwenden von IAM Identity Center für eine robuste Identitätsverwaltung)
- Workshop: Serverless identity (Serverless-Identität)

Zugehörige Tools:

- AWS Security Competency Partners: Identity and Access Management
- saml2aws

SEC02-BP05 Regelmäßiges Überprüfen und Rotieren von Anmeldeinformationen

Prüfen und rotieren Sie Anmeldeinformationen regelmäßig, um die Zeit zu begrenzen, für die diese zum Zugriff auf Ihre Ressourcen genutzt werden können. Langfristig gültige Anmeldeinformationen sind mit Risiken verbunden, die durch die regelmäßige Rotation dieser Informationen reduziert werden können.

Gewünschtes Ergebnis: Implementierung der Rotation von Anmeldeinformationen zur Reduzierung der mit der Nutzung langfristiger Anmeldeinformationen verbundenen Risiken. Prüfen und korrigieren Sie regelmäßig fehlende Compliance mit Richtlinien zur Rotation von Anmeldeinformationen.

Typische Anti-Muster:

- keine Prüfung der Verwendung von Anmeldeinformationen
- unnötiges Verwenden langfristiger Anmeldeinformationen
- Verwendung langfristiger Anmeldeinformationen, ohne diese regelmäßig zu rotieren

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Wenn Sie sich nicht auf temporäre Anmeldeinformationen verlassen können und langfristige Anmeldeinformationen benötigen, prüfen Sie die definierten Anmeldeinformationen, um sicherzustellen, dass die definierten Kontrollen (z. B. Multi-Faktor-Authentifizierung (MFA)) erzwungen und regelmäßig rotiert werden sowie über die entsprechende Zugriffsebene verfügen.

Eine regelmäßige Validierung, vorzugsweise durch ein automatisiertes Tool, ist notwendig, um zu überprüfen, ob die richtigen Kontrollen angewendet werden. Für Personenidentitäten sollten Sie festlegen, dass Benutzer ihre Passwörter regelmäßig ändern und anstelle von Zugriffsschlüsseln temporäre Anmeldeinformationen verwenden. Wenn Sie von AWS Identity and Access Management (IAM)-Benutzern zu zentralisierten Identitäten übergehen, können Sie einen Anmeldeinformationenbericht für die Prüfung Ihrer Benutzer generieren.

Wir empfehlen außerdem, dass Sie MFA in Ihrem Identitätsanbieter erzwingen. Sie können AWS-Config-Regeln einrichten oder Sicherheitsstandards von AWS Security Hub verwenden, um festzustellen, ob Benutzer MFA aktiviert haben. Erwägen Sie die Nutzung von IAM Roles Anywhere zur Bereitstellung temporärer Anmeldeinformationen für Maschinenidentitäten. In Situationen, in denen die Verwendung von IAM-Rollen und temporären Anmeldeinformationen nicht möglich ist, ist eine häufige Prüfung und Rotation von Zugriffsschlüsseln erforderlich.

Implementierungsschritte

- Prüfen Sie die Anmeldeinformationen regelmäßig: Durch die Prüfung der Identitäten, die in Ihrem Identitätsanbieter und IAM konfiguriert sind, können Sie sicherstellen, dass nur autorisierte Identitäten Zugriff auf Ihre Workload haben. Solche Identitäten können unter anderem IAM-Benutzer, Benutzer von AWS IAM Identity Center, Active-Directory-Benutzer oder Benutzer in einem anderen vorgelagerten Identitätsanbieter sein. Entfernen sie beispielsweise Personen, die die Organisation verlassen. Entfernen Sie auch kontoübergreifende Rollen, die nicht mehr erforderlich sind. Sie benötigen einen Prozess zum regelmäßigen Prüfen von Berechtigungen für die Dienste, auf die eine IAM-Entität zugreift. Dadurch können Sie die Richtlinien identifizieren, die Sie ändern müssen, um nicht genutzte Berechtigungen zu entfernen. Verwenden Sie Berichte

zu Anmeldeinformationen und AWS Identity and Access Management Access Analyzer, um IAM-Anmeldeinformationen und -Berechtigungen zu überprüfen. Sie können mit Amazon CloudWatch Alarme für bestimmte API-Aufrufe innerhalb Ihrer AWS-Umgebung einrichten. Amazon GuardDuty kann Sie auch bei unerwarteten Aktivitäten benachrichtigen, die auf zu großzügige Zugriffsrechte hindeuten können, sowie auf nicht beabsichtigte Zugriffe auf IAM-Anmeldeinformationen.

- Regelmäßige Rotation von Anmeldeinformationen: Wenn Sie keine temporären Anmeldeinformationen verwenden können, rotieren Sie IAM-Zugriffsschlüssel regelmäßig (maximal alle 90 Tage). Wenn ein Zugriffsschlüssel ohne Ihr Wissen kompromittiert wurde, wird dadurch begrenzt, für wie lange die Anmeldeinformationen zum Zugriff auf Ihre Ressourcen genutzt werden können. Weitere Informationen zum Rotieren von Zugriffsschlüsseln für IAM-Benutzer finden Sie unter Rotieren der Zugriffsschlüssel.
- Prüfen Sie die IAM-Berechtigungen: Um die Sicherheit Ihres AWS-Konto zu erhöhen, sollten Sie alle Ihre IAM-Richtlinien regelmäßig überprüfen und überwachen. Stellen Sie sicher, dass die Richtlinien dem Prinzip der geringsten Berechtigung entsprechen.
- Erwägen Sie die Automatisierung der Erstellung und Aktualisierung von IAM-Ressourcen: IAM Identity Center automatisiert viele IAM-Aufgaben wie etwa das Rollen- und Richtlinienmanagement. Alternativ können Sie mit AWS CloudFormation die Bereitstellung von IAM-Ressourcen, einschließlich Rollen und Richtlinien, automatisieren. So lässt sich die Zahl menschlicher Fehler verringern, da die Vorlagen verifiziert und ihre Versionen kontrolliert werden können.
- Verwenden Sie IAM Roles Anywhere, um IAM-Benutzer durch Maschinenidentitäten zu ersetzen: IAM Roles Anywhere ermöglicht die Verwendung von Rollen in Bereichen, in denen dies herkömmlicherweise nicht möglich war, etwa auf On-Premises-Servern. IAM Roles Anywhere verwendet ein vertrauenswürdiges X.509-Zertifikat zur Authentifizierung gegenüber AWS und zum Erhalt temporärer Anmeldeinformationen. Mit IAM Roles Anywhere müssen Sie diese Anmeldeinformationen nicht mehr rotieren, da sie nicht mehr in Ihrer On-Premises-Umgebung gespeichert werden. Beachten Sie, dass Sie das X.509-Zertifikat beobachten und gegen Ende seiner Gültigkeitsdauer austauschen müssen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP02 Verwenden von temporären Anmeldeinformationen
- SEC02-BP03 Sicheres Speichern und Verwenden von Secrets

Zugehörige Dokumente:

- Erste Schritte mit AWS Secrets Manager
- IAM Best Practices (Bewährte Methoden für IAM)
- Identitätsanbieter und Verbund
- Partnerlösungen im Bereich Sicherheit: Zugriff und Zugriffssteuerung
- Temporäre Sicherheits-Anmeldeinformationen
- Getting credential reports for your AWS-Konto (Abrufen von Berichten zu Anmeldeinformationen für Ihr AWS-Konto)

Zugehörige Videos:

- Best Practices for Managing, Retrieving, and Rotating Secrets at Scale (Bewährte Methoden zum Verwalten, Abrufen und Rotieren von Secrets in großem Umfang)
- Managing user permissions at scale with AWS IAM Identity Center (Verwalten von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center)
- Mastering identity at every layer of the cake (Beherrschen der Identität auf jeder Ebene)

Zugehörige Beispiele:

- Well-Architected Lab - Automated IAM User Cleanup (Well-Architected Lab – Automatisierte IAM-Benutzerbereinigung)
- Well-Architected Lab - Automated Deployment of IAM Groups and Roles (Well-Architected Lab – Automatisierte Bereitstellung von IAM-Gruppen und -Rollen)

SEC02-BP06 Nutzen von Benutzergruppen und Attributen

Die Definition von Berechtigungen nach Benutzergruppen und Attributen trägt dazu bei, die Anzahl und Komplexität von Richtlinien zu reduzieren, sodass das Prinzip der geringsten Berechtigung einfacher umgesetzt werden kann. Sie können Benutzergruppen verwenden, um die Berechtigungen für viele Personen an einem Ort zu verwalten, basierend auf der Funktion, die sie in Ihrer Organisation innehaben. Attribute, wie z. B. Abteilung oder Standort, können eine zusätzliche Ebene des Berechtigungsumfangs bieten, wenn Personen eine ähnliche Funktion ausüben, aber für unterschiedliche Teilmengen von Ressourcen.

Gewünschtes Ergebnis: Sie können Änderungen der Berechtigungen auf der Grundlage der Funktion auf alle Benutzer anwenden, die diese Funktion ausführen. Die Gruppenzugehörigkeit und -attribute

regeln die Benutzerberechtigungen, sodass Sie die Berechtigungen nicht mehr auf der Ebene der einzelnen Benutzer verwalten müssen. Die Gruppen und Attribute, die Sie in Ihrem Identitätsanbieter (IDP) definieren, werden automatisch an Ihre AWS-Umgebungen weitergegeben.

Typische Anti-Muster:

- Verwaltung von Berechtigungen für einzelne Benutzer und Duplizierung für viele Benutzer.
- Definition von Gruppen auf einer zu hohen Ebene, Gewährung von zu weitreichenden Berechtigungen.
- Die Definition von Gruppen auf einer zu granularen Ebene, was zu Doppelarbeit und Verwirrung über die Mitgliedschaft führt.
- Verwendung von Gruppen mit doppelten Berechtigungen für Teilmengen von Ressourcen, wenn stattdessen Attribute verwendet werden können.
- Keine Verwaltung von Gruppen, Attributen und Mitgliedschaften über einen standardisierten Identitätsanbieter, der in Ihre AWS-Umgebungen integriert ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

AWS-Berechtigungen werden in Dokumenten definiert, die Richtlinien genannt werden und einem Prinzipal zugeordnet sind, z. B. einem Benutzer, einer Gruppe, einer Rolle oder einer Ressource. So können Sie für Ihre Mitarbeiter Gruppen definieren, die auf der Funktion basieren, die Ihre Benutzer in Ihrer Organisation innehaben, und nicht auf den Ressourcen, auf die sie zugreifen. Eine `WebAppDeveloper`-Gruppe kann zum Beispiel eine Richtlinie für die Konfiguration eines Services wie Amazon CloudFront innerhalb eines Entwicklungskontos enthalten. Eine `AutomationDeveloper` -Gruppe kann einige CloudFront-Berechtigungen mit der `WebAppDeveloper`-Gruppe gemeinsam haben. Diese Berechtigungen können in einer separaten Richtlinie erfasst und mit beiden Gruppen verknüpft werden, anstatt dass Benutzer aus beiden Funktionen zu `CloudFront-Zugriffs`gruppe gehören.

Zusätzlich zu Gruppen können Sie auch Attribute verwenden, um den Zugriff weiter einzuschränken. Sie können z. B. ein `Projekt`-Attribut für Benutzer in Ihrer `WebAppDeveloper`-Gruppe haben, um den Zugriff auf projektspezifische Ressourcen einzuschränken. Mit dieser Technik entfällt die Notwendigkeit, für Anwendungsentwickler, die an verschiedenen Projekten arbeiten, unterschiedliche Gruppen einzurichten, wenn ihre Berechtigungen ansonsten identisch sind. Die Art und Weise, wie Sie sich auf Attribute in Berechtigungsrichtlinien beziehen, hängt von deren Quelle ab, d. h. ob sie als

Teil Ihres Verbundprotokolls (wie SAML, OIDC oder SCIM), als benutzerdefinierte SAML-Assertions oder innerhalb von IAM Identity Center definiert sind.

Implementierungsschritte

1. Legen Sie fest, wo Sie Gruppen und Attribute definieren wollen.
   a. Anhand der Anleitung unter SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter können Sie feststellen, ob Sie Gruppen und Attribute innerhalb Ihres Identitätsanbieters, innerhalb von IAM Identity Center oder mit IAM user-Gruppen in einem bestimmten Konto definieren müssen.
2. Definieren Sie Gruppen.
   a. Legen Sie Ihre Gruppen je nach Funktion und Umfang des erforderlichen Zugriffs fest.
   b. Wenn Sie innerhalb von IAM Identity Center definieren, erstellen Sie Gruppen und ordnen die gewünschte Zugriffsebene mithilfe von Berechtigungsgruppen zu.
   c. Wenn Sie die Definition innerhalb eines externen Identitätsanbieters vornehmen, stellen Sie fest, ob der Anbieter das SCIM-Protokoll unterstützt und erwägen Sie die Aktivierung der automatischen Bereitstellung innerhalb von IAM Identity Center. Diese Funktion synchronisiert die Erstellung, Mitgliedschaft und Löschung von Gruppen zwischen Ihrem Anbieter und IAM Identity Center.
3. Definieren Sie Attribute.
   a. Wenn Sie einen externen Identitätsanbieter verwenden, bieten sowohl das SCIM- als auch das SAML 2.0-Protokoll standardmäßig bestimmte Attribute. Zusätzliche Attribute können über SAML-Assertions unter Verwendung des Attributnamens `https://aws.amazon.com/SAML/Attributes/PrincipalTag` definiert und übergeben werden.
   b. Wenn Sie innerhalb von IAM Identity Center definieren, aktivieren Sie das Feature der attributbasierten Zugriffskontrolle (Attribute-based Access Control, ABAC) und definieren Sie Attribute wie gewünscht.
4. Umfangsberechtigungen basierend auf Gruppen und Attributen.
   a. Erwägen Sie, Bedingungen in Ihre Genehmigungsrichtlinien aufzunehmen, die die Attribute Ihres Prinzipals mit den Attributen der Ressourcen vergleichen, auf die zugegriffen wird. Sie können zum Beispiel eine Bedingung definieren, die den Zugriff auf eine Ressource nur dann erlaubt, wenn der Wert eines `PrincipalTag`-Bedingungsschlüssels mit dem Wert eines gleichnamigen `ResourceTag`-Schlüssels übereinstimmt.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen
- COST02-BP04 Implementieren von Gruppen und Rollen

Zugehörige Dokumente:

- Bewährte Methoden in IAM
- Identitäten verwalten in IAM Identity Center
- What Is ABAC for AWS?
- ABAC in IAM Identity Center

Zugehörige Videos:

- Managing user permissions at scale with AWS IAM Identity Center
- Mastering identity at every layer of the cake

SEC 3. Wie verwalten Sie Berechtigungen für Personen und Maschinen?

Verwalten Sie Berechtigungen zum Steuern des Zugriffs auf Personen- und Maschinenidentitäten, die Zugriff auf AWS und Ihren Workload benötigen. Berechtigungen steuern, wer worauf und unter welchen Bedingungen zugreifen kann.

Bewährte Methoden

- SEC03-BP01 Definieren von Zugriffsanforderungen
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen
- SEC03-BP03 Einrichtung eines Notfallzugriffprozesses
- SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen
- SEC03-BP05 Definieren eines Integritätsschutzes für Berechtigungen in Ihrer Organisation
- SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus
- SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs

- SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation
- SEC03-BP09 Sicheres Teilen von Ressourcen mit Dritten

SEC03-BP01 Definieren von Zugriffsanforderungen

Administratoren, Endbenutzer oder andere Komponenten müssen auf jede Komponente oder Ressource Ihres Workloads zugreifen. Sie müssen eine klare Definition davon haben, wer oder was Zugriff auf die einzelnen Komponenten haben soll. Anschließend wählen Sie den entsprechenden Identitätstyp und die entsprechende Authentifizierungs- und Autorisierungsmethode aus.

Typische Anti-Muster:

- Hartkodierung oder Speicherung von geheimen Daten in Ihrer Anwendung
- Gewähren individueller Berechtigungen für alle Nutzer
- Verwendung langlebiger Anmeldeinformationen

Risikostufe, wenn diese bewährte Methode nicht genutzt wird: Hoch

Implementierungsleitfaden

Administratoren, Endbenutzer oder andere Komponenten müssen auf jede Komponente oder Ressource Ihres Workloads zugreifen. Sie müssen eine klare Definition davon haben, wer oder was Zugriff auf die einzelnen Komponenten haben soll. Anschließend wählen Sie den entsprechenden Identitätstyp und die entsprechende Authentifizierungs- und Autorisierungsmethode aus.

Regulärer Zugriff auf AWS-Konten in der Organisation sollte per Verbundzugriff oder einen zentralen Identitätsanbieter bereitgestellt werden. Sie sollten auch Ihr Identitätsmanagement zentralisieren und sicherstellen, dass es ein etabliertes Verfahren zur Integration des AWS-Zugriffs in den Zugriffslebenszyklus der Mitarbeiter gibt Wenn beispielsweise ein Mitarbeiter in eine Rolle mit einer anderen Zugriffsstufe wechselt, sollte sich auch dessen Gruppenmitgliedschaft so ändern, dass die neuen Zugriffsanforderungen berücksichtigt werden.

Legen Sie bei der Definition der Zugriffsanforderungen für nicht menschliche Identitäten fest, welche Anwendungen und Komponenten Zugriff benötigen und wie die Berechtigungen gewährt werden. Eine empfohlene Vorgehensweise ist die Verwendung von nach dem Modell der geringsten Berechtigung entwickelten IAM-Rollen. AWS-verwaltete Richtlinien bieten vordefinierte IAM-Richtlinien für die meisten typischen Anwendungsfälle.

AWS-Services wie beispielsweise AWS Secrets Manager und AWS Systems Manager Parameter Storekönnen dabei helfen, Secrets in sicherer Weise von Anwendungen oder Workloads zu trennen, wenn es nicht möglich ist, IAM-Rollen zu verwenden. In Secrets Manager können Sie die automatische Rotation Ihrer Anmeldeinformationen einrichten. Mit Systems Manager können Sie auf Parameter in Ihren Skripts, Befehlen, SSM-Dokumenten, Konfigurations- und Automatisierungsworkflows verweisen, indem Sie den bei der Erstellung des Parameters angegebenen eindeutigen Namen verwenden.

Sie können AWS Identity and Access Management Roles Anywhere verwenden, um temporäre Sicherheitsanmeldeinformationen in IAM für Workloads zu erhalten, die außerhalb von AWS ausgeführt werden. Ihre Workloads können dieselben IAM-Richtlinien und IAM-Rollen verwenden, die Sie für AWS-Anwendungen zum Zugriff auf AWS-Ressourcen nutzen.

Verwenden Sie nach Möglichkeit kurzfristige temporäre anstelle langfristiger statischer Anmeldeinformationen. Verwenden Sie für Szenarien, in denen Sie IAM-Nutzer mit programmatischem Zugriff und langfristigen Anmeldeinformationen benötigen, Informationen über die letzte Nutzung von Zugriffsschlüsseln, um Zugriffsschlüssel zu entfernen und zu rotieren.

Ressourcen

Zugehörige Dokumente:

- Attributbasierte Zugriffskontrolle (ABAC)
- AWS IAM Identity Center
- IAM Roles Anywhere
- AWS-verwaltete Richtlinien für IAM Identity Center
- AWS-IAM-Richtlinienbedingungen
- IAM-Anwendungsfälle
- Entfernen von nicht benötigten Anmeldeinformationen
- Arbeiten mit Richtlinien
- Steuerung des Zugriffs auf AWS-Ressourcen auf der Grundlage von AWS-Konto, OU oder Organisation
- Identifizieren, Arrangieren und Verwalten von geheimen Daten mithilfe der erweiterten Suche in AWS Secrets Manager

Zugehörige Videos:

- Become an IAM Policy Master in 60 Minutes or Less (Experte für IAM-Richtlinien in unter 60 Minuten)
- Separation of Duties, Least Privilege, Delegation, and CI/CD (Trennung von Pflichten, geringste Berechtigung, Delegierung und CI/CD)
- Streamlining identity and access management for innovation (Optimieren des Identitäts- und Zugriffsmanagements für Innovation)

SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen

Es hat sich bewährt, nur den Zugriff zu gewähren, den Identitäten benötigen, um bestimmte Aktionen auf bestimmten Ressourcen unter bestimmten Bedingungen durchzuführen. Nutzen Sie Gruppen und Identitätsattribute, um Berechtigungen dynamisch in großem Umfang festzulegen, anstatt Berechtigungen für einzelne Benutzer zu definieren. Sie können beispielsweise einer Gruppe von Entwicklern den Zugriff erlauben, nur die Ressourcen für ihr Projekt zu verwalten. So ist sichergestellt, dass einem Entwickler, der nicht mehr am Projekt arbeitet, automatisch der Zugriff entzogen wird, ohne dass die zugrunde liegenden Zugriffsrichtlinien geändert werden müssen.

Gewünschtes Ergebnis: Die Benutzer sollten nur über die erforderlichen Berechtigungen für ihre Aufgabe verfügen. Die Benutzer sollten nur Zugriff auf Produktionsumgebungen erhalten, um eine bestimmte Aufgabe in einem begrenzten Zeitraum auszuführen. Nach Abschluss der Aufgabe sollte der Zugriff widerrufen werden. Nicht mehr benötigte Berechtigungen sollten widerrufen werden. Dies gilt auch, wenn ein Benutzer zu einem anderen Projekt wechselt oder eine andere Tätigkeit übernimmt. Administratorberechtigungen sollten nur einer kleinen Gruppe von vertrauenswürdigen Administratoren erteilt werden. Die Berechtigungen sollten regelmäßig geprüft werden, um eine schleichende Ausweitung der Berechtigungen zu vermeiden. Maschinen- oder Systemkonten sollten die geringsten Berechtigungen erhalten, die zur Ausführung ihrer Aufgaben benötigt werden.

Typische Anti-Muster:

- Standardmäßige Gewährung von Administratorberechtigungen für Benutzer
- Verwendung des Root-Benutzers für alltägliche Aktivitäten
- Erstellung übermäßig großzügiger Richtlinien, jedoch ohne vollständige Administratorberechtigungen
- Keine Überprüfung der Berechtigungen, um festzustellen, ob sie einen Zugriff mit den geringsten Berechtigungen gewähren

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Das Prinzip der geringsten Berechtigung besagt, dass nur die Berechtigungen für die kleinste Gruppe von Aktionen erteilt werden sollte, die für die Durchführung einer bestimmten Aufgabe notwendig sind. Dies schafft ein Gleichgewicht zwischen Benutzerfreundlichkeit, Effizienz und Sicherheit. Die Anwendung dieses Prinzips trägt dazu bei, den unbeabsichtigten Zugriff zu beschränken und nachzuverfolgen, wer auf welche Ressourcen zugreifen kann. IAM-Benutzer und -Rollen verfügen standardmäßig über keine Berechtigungen. Der Root-Benutzer verfügt standardmäßig über vollen Zugriff und sollte strikt kontrolliert, überwacht und nur für Aufgaben verwendet werden, die Root-Zugriff erfordern.

Mithilfe von IAM-Richtlinien können ausdrücklich Berechtigungen für IAM-Rollen oder bestimmte Ressourcen erteilt werden. So können beispielsweise identitätsbasierte Richtlinien an IAM-Gruppen angefügt werden, während S3-Buckets von ressourcenbasierten Richtlinien kontrolliert werden können.

Wenn Sie eine IAM-Richtlinie erstellen, können Sie die Serviceaktionen, Ressourcen und Bedingungen angeben, die erfüllt sein müssen, damit AWS den Zugriff erlaubt oder verweigert. AWS unterstützt eine Vielzahl von Bedingungen, mit denen Sie den Zugriff einschränken können. Mit dem Bedingungsschlüssel `PrincipalOrgID` können Sie beispielsweise Aktionen verweigern, wenn der Anforderer nicht Ihrer AWS-Organisation angehört.

Sie können auch Anforderungen kontrollieren, die AWS-Services in Ihrem Namen stellen, wie das Erstellen einer AWS Lambda-Funktion durch AWS CloudFormation. Hierfür verwenden Sie den Bedingungsschlüssel `CalledVia`. Sie sollten unterschiedliche Richtlinientypen in Ebenen organisieren, um einen umfassenden Verteidigungsansatz aufzubauen und die Berechtigungen Ihrer Benutzer insgesamt zu begrenzen. Sie können auch Beschränkungen in Bezug darauf festlegen, welche Berechtigungen unter welchen Umständen erteilt werden können. So können Sie beispielsweise Ihren Anwendungsteams gestatten, eigene IAM-Richtlinien für die von ihnen erstellten Systeme zu erstellen, müssen aber auch eine Berechtigungsgrenze anwenden, um die maximalen Berechtigungen zu begrenzen, die das System erhalten kann.

Implementierungsschritte

- Implementieren Sie Richtlinien für geringste Berechtigungen: Weisen Sie IAM-Gruppen und -Rollen Zugriffsrichtlinien zu, die in ihrem Umfang möglichst gering und an die von Ihnen definierte Rolle oder Funktion der Benutzer angepasst sind.
  - Basisrichtlinien zur API-Nutzung: Eine Möglichkeit, herauszufinden, welche Berechtigungen benötigt werden, besteht in der Prüfung der AWS CloudTrail-Protokolle. Diese Prüfung

ermöglicht es Ihnen, Berechtigungen zu erstellen, die auf die Aktionen zugeschnitten sind, die der Benutzer tatsächlich in AWS ausführt. IAM Access Analyzer kann automatisch eine IAM-Richtlinie auf der Grundlage einer Aktivität generieren. Sie können IAM Access Advisor auf Organisations- oder Kontoebene verwenden, um zu verfolgen, auf welche Informationen für eine bestimmte Richtlinie zuletzt zugegriffen wurde.

- Erwägen Sie, von AWS verwaltete Richtlinien für berufliche Funktionen zu verwenden. Beim Erstellen von differenzierten Berechtigungsrichtlinien haben Sie zunächst möglicherweise Schwierigkeiten, herauszufinden, wo Sie beginnen sollten. AWS verfügt über verwaltete Richtlinien für allgemeine Job-Rollen, wie z. B. Fakturierungsmitarbeiter, Datenbankadministratoren und Datenwissenschaftler. Diese Richtlinien können helfen, den Zugriff der Benutzer einzuschränken und gleichzeitig festzulegen, wie die Richtlinien für die geringste Berechtigung implementiert werden sollen.
- Entfernen von unnötigen Berechtigungen: Entfernen Sie nicht benötigte Berechtigungen und schränken Sie zu großzügige Richtlinien ein. Die Richtliniengenerierung von IAM Access Analyzer kann bei der Feinabstimmung von Berechtigungsrichtlinien hilfreich sein.
- Stellen Sie sicher, dass Benutzer nur beschränkten Zugriff auf Produktionsumgebungen haben: Benutzer sollten nur Zugriff auf Produktionsumgebungen haben, wenn ein gültiger Anwendungsfall vorliegt. Nachdem der Benutzer die konkreten Aufgaben ausgeführt hat, für die Zugriff auf die Produktionsumgebung erforderlich war, sollte der Zugriff widerrufen werden. Die Beschränkung des Zugriffs auf Produktionsumgebungen hilft, unbeabsichtigte Vorkommnisse mit Auswirkungen auf die Produktion zu verhindern und das Ausmaß der Auswirkungen eines unbeabsichtigten Zugriffs zu verringern.
- Ziehen Sie Berechtigungsgrenzen in Betracht: Eine Berechtigungsgrenze ist eine Funktion für eine verwaltete Richtlinie. Sie legt die maximalen Berechtigungen fest, die mit einer identitätsbasierten Richtlinie einer IAM-Entität erteilt werden können. Eine Berechtigungsgrenze erlaubt einer Entität nur die Ausführung jener Aktionen, die sowohl nach ihren identitätsbasierten Richtlinien als auch nach ihren Berechtigungsgrenzen zulässig sind.
- Ziehen Sie Ressourcen-Tags für Berechtigungen in Betracht: Ein attributbasiertes Zugriffskontrollmodell, das Ressourcen-Tags verwendet, bietet Ihnen die Möglichkeit, den Zugriff basierend auf dem Zweck der Ressource, dem Besitzer, der Umgebung oder anderen Kriterien zu gewähren. Mithilfe von Ressourcen-Tags können Sie beispielsweise zwischen Entwicklungs- und Produktionsumgebungen unterscheiden. Mit diesen Tags können Sie den Zugriff der Entwickler auf die Entwicklungsumgebung beschränken. Durch die Kombination von Tagging und Berechtigungsrichtlinien können Sie einen differenzierten Ressourcenzugriff erzielen, ohne komplizierte, benutzerdefinierte Richtlinien für jeden Tätigkeitsbereich definieren zu müssen.

- Verwenden Sie Service-Kontrollrichtlinien für AWS Organizations. Service-Kontrollrichtlinien steuern zentral die maximal verfügbaren Berechtigungen für Mitgliedskonten in Ihrer Organisation. Wichtig ist, dass Sie mithilfe von Service-Kontrollrichtlinien die Root-Benutzerberechtigungen in Mitgliedskonten einschränken können. Ziehen Sie auch die Verwendung von AWS Control Tower in Betracht, das präskriptive verwaltete Kontrollen zur Bereicherung von AWS Organizations bietet. Sie können auch Ihre eigenen Kontrollen in Control Tower definieren.
- Erstellen Sie eine Benutzerlebenszyklus-Richtlinie für Ihre Organisation: Benutzerlebenszyklus-Richtlinien definieren Aufgaben, die ausgeführt werden müssen, wenn Benutzer neu in AWS eingebunden werden, ihre Rolle oder ihren Aufgabenbereich ändern oder keinen Zugriff mehr auf AWS benötigen. Bei jedem Schritt im Lebenszyklus eines Benutzers sollten Berechtigungsprüfungen erfolgen, um sicherzustellen, dass die Berechtigungen angemessen restriktiv sind und keine schleichenden Berechtigungserweiterungen stattfinden.
- Legen Sie einen regelmäßigen Zeitplan für die Prüfung von Berechtigungen und das Entfernen nicht benötigter Berechtigungen fest: Sie sollten den Benutzerzugriff regelmäßig prüfen, um sicherzustellen, dass die Benutzer nicht zu viele Zugriffsrechte haben. AWS Config und IAM Access Analyzer können bei der Prüfung der Benutzerberechtigungen hilfreich sein.
- Erstellen Sie eine Job-Rollen-Matrix: In einer Job-Rollen-Matrix sind die verschiedenen Rollen und erforderlichen Zugriffsebenen innerhalb Ihrer AWS-Präsenz visuell dargestellt. Mithilfe einer Job-Rollen-Matrix können Sie Berechtigungen auf der Grundlage von Benutzerzuständigkeiten in Ihrer Organisation definieren und trennen. Verwenden Sie Gruppen, anstatt Berechtigungen direkt auf einzelne Benutzer oder Rollen anzuwenden.

Ressourcen

Zugehörige Dokumente:

- Gewähren der geringsten Berechtigung
- Berechtigungsgrenzen für IAM-Entitäten
- Techniken zum Erstellen von IAM-Richtlinien für geringste Berechtigungen
- IAM Access Analyzer erleichtert die Implementierung geringster Berechtigungen durch die Generierung von IAM-Richtlinien auf der Grundlage der Zugriffsaktivitäten
- Delegieren Sie die Berechtigungsverwaltung an Entwickler und verwenden Sie hierfür IAM-Berechtigungsgrenzen
- Verfeinern der Berechtigungen mithilfe der zuletzt genutzten Informationen
- IAM-Richtlinienarten und wann sie verwendet werden sollten

- Testen von IAM-Richtlinien mit dem IAM-Richtliniensimulator
- Integritätsschutz in AWS Control Tower
- Zero-Trust-Architekturen: Eine AWS-Perspektive
- Implementieren des Prinzips der geringsten Berechtigung mit CloudFormation StackSets
- Attributbasierte Zugriffskontrolle (ABAC)
- Reduzieren des Richtlinienbereichs durch Anzeigen der Benutzeraktivität
- Anzeigen des Rollenzugriffs
- Tagging zum Organisieren Ihrer Umgebung und Stärkung der Rechenschaftspflicht
- AWS-Markierungsstrategien
- Markieren von AWS-Ressourcen

Zugehörige Videos:

- Next-generation permissions management (Berechtigungsmanagement der nächsten Generation)
- Zero Trust: An AWS perspective (Zero Trust: Eine AWS-Perspektive)
- How can I use permissions boundaries to limit users and roles to prevent privilege escalation? (Wie kann ich mit Berechtigungsgrenzen Benutzer und Rollen einschränken, um die Eskalation von Berechtigungen zu vermeiden?)

Zugehörige Beispiele:

- Lab: IAM-Berechtigungsgrenzen – Übertragung der Rollenerstellung
- Lab: IAM-Tag-basierte Zugriffskontrolle für EC2

SEC03-BP03 Einrichtung eines Notfallzugriffprozesses

Erstellen Sie einen Prozess, der im unwahrscheinlichen Fall eines Problems mit Ihrem zentralen Identitätsanbieter den Notfallzugriff auf Ihre Workloads ermöglicht.

Sie müssen Prozesse für verschiedene Ausfallmodi entwerfen, die zu einem Notfallereignis führen können. Unter normalen Umständen verbinden sich die Benutzer Ihrer Belegschaft beispielsweise über einen zentralen Identitätsanbieter mit der Cloud (SEC02-BP04), um ihre Workloads zu verwalten. Wenn der zentrale Identitätsanbieter jedoch ausfällt oder die Konfiguration für den Verbund in der Cloud geändert wird, können sich die Benutzer in Ihrem Unternehmen

möglicherweise nicht mit der Cloud verbinden. Ein Prozess für den Notfallzugriff ermöglicht autorisierten Administratoren den Zugriff auf Ihre Cloud-Ressourcen über alternative Verfahren (z. B. eine alternative Form des Verbunds oder direkter Benutzerzugriff), um Probleme mit Ihrer Verbundkonfiguration oder Ihren Workloads zu beheben. Der Prozess für den Notfallzugriff wird verwendet, bis der normale Verbundmechanismus wiederhergestellt ist.

Gewünschtes Ergebnis:

- Sie haben die Ausfallmodi definiert und dokumentiert, die als Notfall gelten: Berücksichtigen Sie dabei Ihre normalen Abläufe und die Systeme, auf die Ihre Benutzer angewiesen sind, um ihre Workloads zu verwalten. Überlegen Sie, wie jede dieser Abhängigkeiten ausfallen und zu einer Notsituation führen kann. Die Fragen und bewährten Methoden in der Säule „Zuverlässigkeit“ können Sie dabei unterstützen, Ausfallmodi zu identifizieren und widerstandsfähigere Systeme zu entwickeln, um die Wahrscheinlichkeit von Ausfällen zu minimieren.
- Sie haben die Schritte dokumentiert, die befolgt werden müssen, um einen Ausfall als Notfall zu identifizieren. Sie können beispielsweise festlegen, dass Ihre Identitätsadministratoren den Status Ihrer primären und Standby-Identitätsanbieter überprüfen müssen und, falls beide nicht verfügbar sind, ein Notfallereignis für den Ausfall eines Identitätsanbieters feststellen.
- Sie haben einen Prozess für den Notfallzugriff definiert, der für jeden Notfall- oder Ausfallmodus spezifisch ist. Wenn Sie hier möglichst detaillierte Informationen angeben, kann dies der Neigung Ihrer Benutzer entgegenwirken, einen allgemeinen Prozess für alle Arten von Notfällen zu stark zu nutzen. Ihre Prozesse für den Notfallzugriff beschreiben die Umstände, unter denen ein Prozess jeweils verwendet werden sollte, und umgekehrt Situationen, in denen der Prozess nicht verwendet werden sollte. In diesem Fall wird auf alternative Prozesse hingewiesen, die zutreffen können.
- Ihre Prozesse sind mit detaillierten Anweisungen und Playbooks, die schnell und effizient befolgt werden können, gut dokumentiert. Denken Sie daran, dass ein Notfallereignis Stress für Ihre Benutzer bedeuten kann und dass sie unter extremem Zeitdruck stehen können. Gestalten Sie Ihren Prozess daher so einfach wie möglich.

Typische Anti-Muster:

- Sie verfügen nicht über gut dokumentierte und gut getestete Prozesse für den Notfallzugriff. Ihre Benutzer sind nicht auf einen Notfall vorbereitet und nutzen improvisierte Prozesse, wenn er eintritt.
- Ihre Prozesse für den Notfallzugriff hängen von denselben Systemen (z. B. einem zentralen Identitätsanbieter) ab wie Ihre normalen Zugriffsmechanismen. Das bedeutet, dass der Ausfall eines solchen Systems sowohl Ihre normalen Zugriffsmechanismen als auch Ihre

Notfallzugriffsmechanismen betrifft und Ihre Fähigkeit zur Wiederherstellung nach dem Ausfall beeinträchtigen kann.

- Ihre Prozesse für den Notfallzugriff werden in Situationen verwendet, die keine Notfälle sind. Ein Beispiel könnte sein, dass Ihre Benutzer Prozesse für den Notfallzugriff häufig missbrauchen, da es für sie einfacher ist, Änderungen direkt vorzunehmen, als Änderungen über eine Pipeline einzureichen.
- Ihre Prozesse für den Notfallzugriff generieren nicht genügend Protokolle, um sie zu überwachen, oder die Protokolle werden nicht so überwacht, dass Sie bei einem möglichen Missbrauch der Prozesse gewarnt werden.

Vorteile der Nutzung dieser bewährten Methode:

- Durch gut dokumentierte und gut getestete Prozesse für den Notfallzugriff können Sie die Zeit reduzieren, die Ihre Benutzer benötigen, um auf ein Notfallereignis zu reagieren und es zu beheben. Dies kann zu kürzeren Ausfallzeiten und einer höheren Verfügbarkeit der Services führen, die Sie für Ihre Kunden bereitstellen.
- Sie können jede Notfallzugriffsanfrage verfolgen und unbefugte Versuche, den Prozess für Nicht-Notfallereignisse zu missbrauchen, erkennen und darauf hinweisen.

Risikostufe bei fehlender Befolgung dieser Best Practice:: Mittel

Implementierungsleitfaden

Dieser Abschnitt enthält Richtlinien zur Erstellung von Prozessen für den Notfallzugriff für verschiedene Ausfallmodi im Zusammenhang mit Workloads, die in AWS bereitgestellt werden. Zunächst finden Sie allgemeine Leitlinien, die für alle Ausfallmodi gelten, und danach spezifische Anleitungen für die verschiedenen Arten von Ausfallmodi.

Allgemeine Leitlinien für alle Ausfallmodi

Beachten Sie beim Entwerfen eines Prozesses für den Notfallzugriff für einen Ausfallmodus Folgendes:

- Dokumentieren Sie die Voraussetzungen und Annahmen für den Prozess: Wann soll der Prozess verwendet werden und wann nicht? Es ist hilfreich, den Ausfallmodus detailliert zu beschreiben und Annahmen zu dokumentieren, z. B. den Zustand anderer verwandter Systeme. Der Prozess für den Ausfallmodus 2 geht beispielsweise davon aus, dass der Identitätsanbieter verfügbar ist, aber die Konfiguration in AWS geändert wurde oder abgelaufen ist.

- Erstellen Sie im Voraus Ressourcen, die für den Notfallzugriffsprozess benötigt werden (SEC10-BP05). Erstellen Sie beispielsweise vorab das AWS-Konto für den Notfallzugriff (IAM users und -Rollen) und die kontoübergreifenden IAM-Rollen in allen Workload-Konten. So wird sichergestellt, dass diese Ressourcen bereit und verfügbar sind, wenn ein Notfallereignis eintritt. Durch das Erstellen von Ressourcen im Voraus sind Sie nicht abhängig von den APIs der AWS- Steuerebene (zum Erstellen und Ändern von AWS-Ressourcen), die im Notfall möglicherweise nicht verfügbar sind. Wenn Sie IAM-Ressourcen vorab erstellen, müssen Sie außerdem keine möglichen Verzögerungen aufgrund einer letztendlichen Konsistenz berücksichtigen.
- Schließen Sie Prozesse für den Notfallzugriff in Ihre Vorfallmanagementpläne ein (SEC10-BP02). Dokumentieren Sie, wie Notfallereignisse nachverfolgt und an andere in Ihrem Unternehmen, z. B. an Peer-Teams, Führungskräfte und gegebenenfalls extern an Kunden und Geschäftspartner, kommuniziert werden sollen.
- Definieren Sie den Prozess für Notfallzugriffsanfragen in Ihrem bestehenden Workflow-System für Serviceanfragen, falls eines vorhanden ist. In der Regel können Sie mit solchen Workflow-Systemen Eingabeformulare erstellen, um Informationen zur Anfrage zu erfassen, die Anfrage in jeder Phase des Workflows zu verfolgen und sowohl automatisierte als auch manuelle Genehmigungsschritte hinzuzufügen. Ordnen Sie jede Anfrage einem entsprechenden Notfallereignis zu, das in Ihrem Vorfallmanagement-System verfolgt wird. Mit einem einheitlichen System für Notfallzugriffe können Sie diese Anfragen in einem zentralen System verfolgen, Nutzungstrends analysieren und Ihre Prozesse verbessern.
- Stellen Sie sicher, dass Ihre Notfallzugriffsprozesse nur von autorisierten Benutzern initiiert werden können, und legen Sie fest, dass Genehmigungen von Kollegen oder Führungskräften des Benutzers erforderlich sind. Das Genehmigungsverfahren sollte sowohl während als auch außerhalb der Geschäftszeiten funktionieren. Definieren Sie, wie Genehmigungsanfragen sekundäre Genehmiger berücksichtigen, falls die primären Genehmiger nicht verfügbar sind, und wie sie in Ihrer Managementkette nach oben eskaliert werden, bis sie genehmigt wurden.
- Stellen Sie sicher, dass der Prozess detaillierte Auditprotokolle und Ereignisse sowohl für erfolgreiche als auch für fehlgeschlagene Versuche generiert, Notfallzugriff zu erhalten. Überwachen Sie sowohl den Anforderungsprozess als auch den Notfallzugriffsmechanismus, um Missbrauch oder nicht autorisierte Zugriffe zu erkennen. Korrelieren Sie Aktivitäten mit laufenden Notfallereignissen aus Ihrem Vorfallsmanagement-System und senden Sie Benachrichtigungen, wenn Aktionen außerhalb der erwarteten Zeiträume erfolgen. Sie sollten beispielsweise die Aktivitäten im AWS-Konto für den Notfallzugriff überwachen und entsprechende Benachrichtigungen senden, da es im normalen Betrieb nie verwendet werden sollte.

- Testen Sie die Notfallzugriffsprozesse regelmäßig, um sicherzustellen, dass die Schritte klar sind und die richtigen Zugriffsebenen schnell und effizient gewährt werden. Ihre Notfallzugriffsprozesse sollten im Rahmen der Simulation von Vorfallsreaktionen (SEC10-BP07) und Tests der Notfallwiederherstellung (REL13-BP03) getestet werden.

Ausfallmodus 1: Der für den Verbund mit AWS verwendete Identitätsanbieter ist nicht verfügbar

Wie in SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter beschrieben, wird empfohlen, sich auf einen zentralen Identitätsanbieter zu verlassen, der die Benutzer Ihres Unternehmens verbindet, um den Zugriff auf AWS-Konten zu gewähren. Sie können mit IAM Identity Center einen Verbund für mehrere AWS-Konten in Ihrer AWS-Organisation implementieren oder einzelne AWS-Konten mit IAM verbinden. In beiden Fällen authentifizieren sich die Benutzer in Ihrer Belegschaft beim zentralen Identitätsanbieter, bevor sie zu einem AWS-Anmeldeendpunkt für das Single Sign-On weitergeleitet werden.

Im unwahrscheinlichen Fall, dass der zentrale Identitätsanbieter nicht verfügbar ist, können sich die Benutzer Ihrer Belegschaft nicht mit AWS-Konten verbinden oder ihre Workloads verwalten. In einem solchen Notfall können Sie einen Notfallzugriffsprozess für eine kleine Gruppe von Administratoren einrichten, die auf AWS-Konten zugreifen dürfen, um kritische Aufgaben auszuführen, die nicht warten können, bis die zentralen Identitätsanbieter wieder online sind. Nehmen Sie beispielsweise an, dass Ihr Identitätsanbieter für 4 Stunden nicht verfügbar ist und während dieses Zeitraums die Obergrenzen einer Amazon EC2 Auto Scaling-Gruppe in einem Produktionskonto geändert werden müssen, um einen unerwarteten Anstieg des Kundenverkehrs zu bewältigen. Ihre Notfalladministratoren sollten den Notfallzugriffsprozess befolgen, um Zugriff auf das spezifische AWS-Konto in der Produktion zu erhalten und die erforderlichen Änderungen vorzunehmen.

Der Notfallzugriffsprozess basiert auf einem vorab erstellten AWS-Konto für den Notfallzugriff, das ausschließlich für den Notfallzugriff verwendet wird und über AWS-Ressourcen (wie IAM-Rollen und IAM users) zur Unterstützung des Notfallzugriffsprozesses verfügt. Während des normalen Betriebs sollte niemand auf das Notfallzugriffskonto zugreifen. Sie müssen dieses Konto auf Missbrauch überwachen und ggf. Warnungen senden (weitere Informationen finden Sie im vorherigen Abschnitt mit allgemeinen Leitlinien).

Das Notfallzugriffskonto verfügt über IAM-Notfallzugriffsrollen mit der Berechtigung, kontoübergreifende Rollen in den AWS-Konten anzunehmen, für die Notfallzugriff erforderlich ist. Diese IAM-Rollen sind vordefiniert und mit Vertrauensrichtlinien konfiguriert, die den IAM-Rollen des Notfallkontos vertrauen.

Das Notfallzugriffsverfahren kann einen der folgenden Ansätze verwenden:

- Sie können vorab IAM users für Ihre Notfalladministratoren im Notfallzugriffskonto erstellen, denen sichere Passwörtern und MFA-Token zugeordnet sind. Diese IAM users verfügen über Berechtigungen, um die IAM-Rollen anzunehmen, die dann den kontoübergreifenden Zugriff auf das AWS-Konto ermöglichen, für das der Notfallzugriff erforderlich ist. Wir empfehlen, so wenige solcher Benutzer wie möglich zu erstellen und jeden Benutzer einem einzelnen Notfalladministrator zuzuweisen. Während eines Notfalls meldet sich ein Notfalladministrator mit seinem Passwort und seinem MFA-Tokencode beim Notfallzugriffskonto an, wechselt zur IAM-Notfallzugriffsrolle im Notfallkonto und wechselt schließlich zur IAM-Notfallzugriffsrolle im Workload-Konto, um die für den Notfall erforderliche Änderungsaktion durchzuführen. Der Vorteil dieses Ansatzes besteht darin, dass jeder IAM user einem Notfalladministrator zugewiesen ist und Sie anhand der CloudTrail-Ereignisse feststellen können, welcher Benutzer sich angemeldet hat. Der Nachteil ist, dass Sie mehrere IAM users mit den zugehörigen langlebigen Passwörtern und MFA-Token verwalten müssen.
- Sie können den Root-Benutzer für das Notfallzugriff-AWS-Konto verwenden, um sich beim Notfallzugriffskonto anzumelden, die IAM-Rolle für den Notfallzugriff anzunehmen und dann die kontoübergreifende Rolle im Workload-Konto anzunehmen. Wir empfehlen, ein sicheres Passwort und mehrere MFA-Token für den Root-Benutzer festzulegen. Wir empfehlen außerdem, das Passwort und die MFA-Token in einem sicheren Vault für Unternehmensanmeldeinformationen zu speichern, der eine starke Authentifizierung und Autorisierung erzwingt. Sie sollten das Passwort und die Faktoren zum Zurücksetzen des MFA-Tokens sichern: Legen Sie die E-Mail-Adresse für das Konto auf eine E-Mail-Verteilerliste fest, die von Ihren Cloud-Sicherheitsadministratoren überwacht wird. Legen Sie die Telefonnummer des Kontos auf eine gemeinsam genutzte Telefonnummer fest, die ebenfalls von Sicherheitsadministratoren überwacht wird. Der Vorteil dieses Ansatzes besteht darin, dass nur ein Satz von Root-Benutzeranmeldeinformationen verwaltet werden muss. Der Nachteil ist, dass sich mehrere Administratoren als Root-Benutzer anmelden können, da es sich um einen gemeinsam genutzten Benutzer handelt. Sie müssen die Protokollereignisse für den Unternehmens-Vault überprüfen, um festzustellen, welcher Administrator das Passwort für den Root-Benutzer ausgecheckt hat.

Ausfallmodus 2: Die Konfiguration des Identitätsanbieters in AWS wurde geändert oder ist abgelaufen

Um den Verbund der Benutzer in Ihrem Unternehmen mit AWS-Konten zu ermöglichen, können Sie IAM Identity Center mit einem externen Identitätsanbieter konfigurieren oder einen IAM-Identitätsanbieter erstellen (SEC02-BP04). In der Regel konfigurieren Sie diese, indem Sie ein XML-Dokument mit SAML-Metadaten importieren, das von Ihrem Identitätsanbieter bereitgestellt wird. Das

XML-Metadatendokument enthält ein X.509-Zertifikat, das einem privaten Schlüssel entspricht, mit dem der Identitätsanbieter seine SAML-Zusicherungen signiert.

Diese Konfigurationen auf AWS-Seite können versehentlich von einem Administrator geändert oder gelöscht werden. In einem anderen Szenario läuft das in AWS importierte X.509-Zertifikat möglicherweise ab und eine neue XML-Metadatendatei mit einem neuen Zertifikat wurde noch nicht in AWS importiert. In beiden Szenarien kann der Verbund mit AWS für die Benutzer Ihrer Belegschaft unterbrochen werden, was zu einem Notfall führt.

In einem solchen Notfall können Sie Ihren Identitätsadministratoren Zugriff auf AWS gewähren, um die Verbundprobleme zu beheben. Ihr Identitätsadministrator verwendet beispielsweise den Notfallzugriffsprozess, um sich beim AWS-Konto für den Notfallzugriff anzumelden. Er wechselt zu einer Rolle im Identity Center-Administratorkonto und aktualisiert die Konfiguration des externen Identitätsanbieters, indem er das aktuelle XML-Dokument mit SAML-Metadaten von Ihrem Identitätsanbieter importiert, um den Verbund wieder zu aktivieren. Sobald der Verbund wiederhergestellt ist, verwenden die Benutzer in Ihrer Belegschaft weiter den normalen Betriebsprozess, um sich mit ihren Workload-Konten zu verbinden.

Sie können die oben für Ausfallmodus 1 beschriebenen Vorgehensweisen befolgen, um einen Notfallzugriffsprozess zu erstellen. Sie können Ihren Identitätsadministratoren Berechtigungen nach dem Prinzip der geringsten Rechte gewähren, sodass sie nur auf das Identity Center-Administratorkonto zugreifen und nur in diesem Konto Aktionen für Identity Center ausführen können.

Ausfallmodus 3: Störung von Identity Center

Für den unwahrscheinlichen Fall einer Störung von IAM Identity Center oder einer AWS-Region empfehlen wir, eine Konfiguration einzurichten, mit der Sie temporären Zugriff auf die AWS Management Console gewähren können.

Der Notfallzugriffsprozess verwendet einen direkten Verbund von Ihrem Identitätsanbieter zu IAM in einem Notfallkonto. Einzelheiten zu den Prozess- und Entwurfsüberlegungen finden Sie im Artikel zum Einrichten des Notfallzugriffs auf die AWS Management Console.

Implementierungsschritte

Allgemeine Schritte für alle Ausfallmodi

- Erstellen Sie ein AWS-Konto speziell für Notfallzugriffsprozesse. Erstellen Sie vorab die für das Konto benötigten IAM-Ressourcen wie IAM-Rollen oder IAM users und optional IAM-

Identitätsanbieter. Erstellen Sie außerdem vorab kontoübergreifende IAM-Rollen in den AWS-Konten für den Workload mit Vertrauensbeziehungen zu den entsprechenden IAM-Rollen im Notfallzugriffskonto. Nutzen Sie Instrumentierungsservices wie AWS CloudFormation StackSets mit AWS Organizations, um solche Ressourcen in den Mitgliedskonten Ihrer Organisation zu erstellen.

- Erstellen Sie in AWS Organizations Service-Kontrollrichtlinien (Service Control Policies, SCPs), um das Löschen und Ändern der kontoübergreifenden IAM-Rollen in den AWS-Konten der Mitglieder zu verweigern.
- Aktivieren Sie CloudTrail für das AWS-Konto für den Notfallzugriff und senden Sie die Trail-Ereignisse an einen zentralen S3-Bucket im AWS-Konto für die Protokollerfassung. Wenn Sie AWS Control Tower verwenden, um Ihre AWS-Umgebung mit mehreren Konten einzurichten und zu verwalten, ist für jedes Konto, das Sie mit AWS Control Tower erstellen oder in AWS Control Tower registrieren, CloudTrail standardmäßig aktiviert und wird an einen S3-Bucket in einem dedizierten AWS-Konto für das Protokollarchiv gesendet.
- Überwachen Sie die Aktivitäten des Notfallzugriffskontos, indem Sie EventBridge-Regeln erstellen, die bei der Anmeldung in der Konsole und bei API-Aktivitäten durch die IAM-Notfallrollen greifen. Senden Sie Benachrichtigungen an Ihr Security Operations Center, wenn Aktivitäten außerhalb eines laufenden Notfallereignisses stattfinden, das in Ihrem Vorfallmanagement-System nachverfolgt wurde.

Zusätzliche Schritte für Ausfallmodus 1 (Der für den Verbund mit AWS verwendete Identitätsanbieter ist nicht verfügbar) und Ausfallmodus 2 (Die Konfiguration des Identitätsanbieters in AWS wurde geändert oder ist abgelaufen)

- Erstellen Sie vorab Ressourcen, je nachdem, welchen Mechanismus Sie für den Notfallzugriff wählen:
  - Unter Verwendung der IAM users: Erstellen Sie vorab die IAM users mit sicheren Passwörtern und den zugehörigen MFA-Geräten.
  - Unter Verwendung des Root-Benutzers des Notfallkontos: Konfigurieren Sie den Root-Benutzer mit einem sicheren Passwort und speichern Sie das Passwort im Unternehmens-Vault für Anmeldeinformationen. Ordnen Sie dem Root-Benutzer mehrere physische MFA-Geräte zu und bewahren Sie die Geräte an Orten auf, zu denen die Mitglieder Ihres Notfalladministratorteams schnell Zugang haben.

Zusätzliche Schritte für den Ausfallmodus 3 (Störung von Identity Center)

- Erstellen Sie wie im Artikel zum Einrichten des Notfallzugriffs auf die AWS Management Console erläutert im AWS-Konto für den Notfallzugriff einen IAM-Identitätsanbieter, um den direkten SAML-Verbund von Ihrem Identitätsanbieter aus zu ermöglichen.
- Erstellen Sie Notfalleinsatzgruppen in Ihrem Identitätsanbieter ohne Mitglieder.
- Erstellen Sie IAM-Rollen, die den Notfalleinsatzgruppen im Notfallzugriffskonto entsprechen.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen
- SEC10-BP02 Entwickeln von Vorfallmanagementplänen
- SEC10-BP07 Durchführen von Gamedays

Zugehörige Dokumente:

- Set up emergency access to the AWS Management Console (Einrichten des Notfallzugriffs auf die AWS-Managementkonsole)
- Enabling SAML 2.0 federated users to access the AWS Management Console (Aktivieren des Zugriffs von SAML 2.0-Verbundbenutzern auf die AWS-Managementkonsole)
- Break glass access („Break Glass“-Zugriff)

Zugehörige Videos:

- AWS re:Invent 2022 - Simplify your existing workforce access with IAM Identity Center (AWS re:Invent 2022 – Vereinfachen des vorhandenen Mitarbeiterzugriffs mit IAM Identity Center)
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive (AWS re:inforce 2022 – AWS Identity and Access Management (IAM) zur Vertiefung)

Zugehörige Beispiele:

- AWS Break Glass Role (AWS-Rolle „Break Glass“)
- AWS Customer Playbook Framework
- AWS-Beispiele von Playbooks für die Vorfallsreaktion

SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen

Wenn Ihre Teams bestimmen, welchen Zugriff sie benötigen, entfernen Sie unnötige Berechtigungen und erstellen Sie Überprüfungsprozesse, damit jederzeit dem Prinzip der geringsten Berechtigung entsprochen wird. Überwachen Sie Ihre Identitäten kontinuierlich und entfernen Sie ungenutzte Identitäten und Berechtigungen für den Zugriff von Menschen und Maschinen.

Gewünschtes Ergebnis: Berechtigungsrichtlinien sollten dem Prinzip der geringsten Berechtigung folgen. Wenn Zuständigkeiten und Rollen immer besser definiert werden, müssen Sie Ihre Berechtigungsrichtlinien prüfen, um unnötige Berechtigungen zu entfernen. Dieses Konzept verringert die Auswirkungen, wenn Anmeldeinformationen versehentlich offen gelegt werden oder wenn anderweitig ohne Genehmigung darauf zugegriffen wird.

Typische Anti-Muster:

- standardmäßige Gewährung von Administratorberechtigungen für Benutzer
- Erstellung übermäßig lockerer Richtlinien, jedoch ohne vollständige Administratorberechtigungen
- Aufbewahrung von Berechtigungsrichtlinien, nachdem Sie nicht mehr benötigt werden

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wenn Teams und Projekte gerade erst mit der Arbeit beginnen, können lockere Richtlinien verwendet werden, um Innovationen und Agilität zu unterstützen. So könnten beispielsweise Entwickler in einer Entwicklungs- und Testumgebung Zugang zu einer breiten Palette von AWS-Services erhalten. Wir empfehlen, den Zugriff kontinuierlich zu prüfen und auf sServices und Serviceaktionen einzuschränken, die für die anstehende Aufgabe wirklich benötigt werden. Wir empfehlen diese Evaluierung für menschliche und für maschinelle Identitäten. Maschinenidentitäten, manchmal auch als System- oder Servicekonten bezeichnet, sind Identitäten, die AWS den Zugriff auf Anwendungen oder Server ermöglichen. Dieser Zugriff ist besonders in einer Produktionsumgebung wichtig, in der übermäßig lockere Zugriffsregeln weitreichende Auswirkungen haben und möglicherweise Kundendaten offen legen könnten.

AWS bietet mehrere Verfahren zur Unterstützung der Identifizierung nicht verwendeter Benutzer, Rollen, Berechtigungen und Anmeldeinformationen. AWS kann auch bei der Analyse von Zugriffsaktivitäten von IAM-Benutzern und -Rollen helfen, darunter ebenfalls Analysen zu zugehörigen Zugriffsschlüsseln sowie zum Zugriff auf AWS-Ressourcen wie etwa Objekten in

Amazon S3-Buckets. Die Generierung von Richtlinien mit AWS Identity and Access Management Access Analyzer kann Ihnen bei der Erstellung restriktiver Berechtigungsrichtlinien auf der Grundlage der Services und Aktionen helfen, mit denen ein Prinzipal tatsächlich interagiert. Die attributbasierte Zugriffssteuerung (Attribute-based Access Control, ABAC) kann die Verwaltung von Berechtigungen vereinfachen, da Sie Benutzern Berechtigungen auf der Grundlage ihrer Attribute erteilen können, anstatt jedem Benutzer direkt Berechtigungsrichtlinien zuzuweisen.

Implementierungsschritte

- Verwendung von AWS Identity and Access Management Access Analyzer: IAM Access Analyzer hilft bei der Identifizierung von Ressourcen in Ihrer Organisation und in Konten, wie etwa Amazon Simple Storage Service (Amazon S3)-Buckets oder IAM-Rollen, diegemeinsam mit einer externen Entität genutzt werden.
- Verwendung der Richtliniengenerierung von IAM Access Analyzer: Die Richtliniengenerierung von IAM Access Analyzer hilft bei der Erstellung detaillierter Berechtigungsrichtlinien auf der Grundlage eines IAM-Benutzers oder der Zugriffsaktivität einer IAM-Rolle.
- Festlegen eines akzeptablen Zeitrahmens und einer Nutzungsrichtlinie für IAM-Benutzer und -Rollen: Verwenden Sie den Zeitstempel des letzten Zugriffs, um nicht verwendete Benutzer und Rollen zu identifizieren und diese zu entfernen. Prüfen Sie die Informationen zum letzten Zugriff auf Services und Aktionen, um Berechtigungen für bestimmte Benutzer und Rollen zu identifizieren und entsprechend zuzuteilen. Sie können beispielsweise Informationen zum letzten Zugriff verwenden, um die spezifischen Amazon S3-Aktionen zu identifizieren, die Ihre Anwendungsrolle erfordert, und den Zugriff der Rolle auf diese Aktionen beschränken. Funktionen für die zuletzt abgerufenen Informationen sind in der AWS Management Console und programmgesteuert verfügbar, damit Sie sie in Ihre Infrastruktur-Workflows und automatisierten Tools integrieren können.
- Erwägen Sie die Protokollierung von Datenereignissen in AWS CloudTrail: Standardmäßig protokolliert CloudTrail keine Datenereignisse wie Amazon S3-Aktivitäten auf Objektebene (zum Beispiel `GetObject` und `DeleteObject`) oder Amazon DynamoDB-Tabellenaktivitäten (zum Beispiel `PutItem` und `DeleteItem`). Erwägen Sie die Aktivierung der Protokollierung dieser Ereignisse, um zu ermitteln, welche Benutzer und Rollen Zugriff auf bestimmte Amazon S3-Objekte oder DynamoDB-Tabellenelemente benötigen.

Ressourcen

Zugehörige Dokumente:

- Gewähren von geringsten Berechtigungen

- Entfernen von nicht benötigten Anmeldeinformationen
- Was ist AWS CloudTrail?
- Arbeiten mit Richtlinien
- Protokollierung und Überwachung von DynamoDB
- Enabling CloudTrail event logging for Amazon S3 buckets and objects (Aktivieren von CloudTrail-Ereignisprotokollierung für Amazon-S3-Buckets und -Objekte)
- Getting credential reports for your AWS-Konto (Abrufen von Berichten zu Anmeldeinformationen für Ihr AWS-Konto)

Zugehörige Videos:

- Become an IAM Policy Master in 60 Minutes or Less (Experte für IAM-Richtlinien in unter 60 Minuten werden)
- Separation of Duties, Least Privilege, Delegation, and CI/CD (Trennung von Pflichten, geringste Berechtigung, Delegierung und CI/CD)
- AWS re:Inforce 2022 - AWS Identity and Access Management (IAM) deep dive (AWS re:inforce 2022 – AWS Identity and Access Management (IAM) zur Vertiefung)

SEC03-BP05 Definieren eines Integritätsschutzes für Berechtigungen in Ihrer Organisation

Verwenden Sie Maßnahmen zum Integritätsschutz, um den Umfang der verfügbaren Berechtigungen, die Prinzipalen gewährt werden können, einzuschränken. Die Bewertungskette für Berechtigungsrichtlinien umfasst Ihren Integritätsschutz, um die effektiven Berechtigungen eines Prinzipals bei Autorisierungsentscheidungen zu bestimmen. Sie können Maßnahmen zum Integritätsschutz mit einem ebenenbasierten Ansatz definieren. Wenden Sie einige Maßnahmen zum Integritätsschutz allgemein für Ihre gesamte Organisation an und andere granular auf Sitzungen mit temporärem Zugriff.

Gewünschtes Ergebnis: Sie haben eine klare Isolierung der Umgebungen durch separate AWS-Konten. Service-Kontrollrichtlinien (SCPs) werden verwendet, um organisationsweite Maßnahmen zum Integritätsschutz zu definieren. Umfassender angelegte Maßnahmen zu Integritätsschutz werden auf den Hierarchieebenen festgelegt, die der Root Ihrer Organisation am nächsten sind, und strengerer Integritätsschutz wird näher an der Ebene der einzelnen Konten festgelegt. Sofern unterstützt, definieren Ressourcenrichtlinien die Bedingungen, die ein Prinzipal erfüllen muss, um Zugriff auf eine Ressource zu erhalten. Die Ressourcenrichtlinien schränken auch den Umfang

der erlaubten Aktionen ein, wo dies angebracht ist. Berechtigungsgrenzen werden auf Prinzipale verteilt, die Workload-Berechtigungen verwalten und die Verwaltung von Berechtigungen an einzelne Workload-Besitzer delegieren.

Typische Anti-Muster:

- Erstellen eines AWS-Konten als Mitglied innerhalb einer AWS-Organisation, aber keine Verwendung von SCPs, um die Verwendung und die verfügbaren Berechtigungen für ihre Anmeldeinformationen einzuschränken
- Zuweisung von Berechtigungen auf der Grundlage der geringsten Berechtigung, aber kein Integritätsschutz für die maximale Anzahl von Berechtigungen, die gewährt werden können
- Vertrauen auf die implizite Verweigerungsgrundlage von AWS IAM, um Berechtigungen einzuschränken, in der Annahme, dass die Richtlinien keine unerwünschte explizite Erlaubnis erteilen werden
- mehrere Workload-Umgebungen im selben AWS-Konto ausführen und sich dann auf Mechanismen wie VPCs, Tags oder Ressourcenrichtlinien verlassen, um Berechtigungsgrenzen durchzusetzen

Vorteile der Einführung dieser bewährten Methode: Einrichtungen zum Integritätsschutz helfen dabei, Vertrauen zu schaffen, dass unerwünschte Berechtigungen nicht gewährt werden können, selbst wenn eine Berechtigungsrichtlinie dies versucht. Dies kann die Definition und Verwaltung von Berechtigungen vereinfachen, da der maximale Umfang der zu berücksichtigenden Berechtigungen reduziert wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wir empfehlen Ihnen, einen ebenenbasierten Ansatz zu verwenden, um für Maßnahmen für den Integritätsschutz für Ihre Organisation zu definieren. Dieser Ansatz reduziert systematisch die maximale Anzahl der möglichen Berechtigungen, wenn weitere Ebenen hinzugefügt werden. So können Sie den Zugriff nach dem Prinzip der geringsten Berechtigung gewähren und das Risiko eines unbeabsichtigten Zugriffs aufgrund einer falschen Konfiguration der Richtlinie verringern.

Der erste Schritt zur Einrichtung zum Integritätsschutz ist die Isolierung Ihrer Workloads und Umgebungen in getrennten AWS-Konten. Prinzipale eines Kontos können ohne ausdrückliche Erlaubnis nicht auf die Ressourcen eines anderen Kontos zugreifen, selbst wenn sich beide Konten in derselben AWS-Organisation oder unter derselben Organisationseinheit (OE) befinden. Sie können OEs verwenden, um Konten zu gruppieren, die Sie als eine Einheit verwalten möchten.

Der nächste Schritt besteht darin, die maximale Anzahl von Berechtigungen zu reduzieren, die Sie Prinzipalen innerhalb der Mitgliedskonten Ihrer Organisation erteilen können. Zu diesem Zweck können Sie Service-Kontrollrichtlinien (SCPs) verwenden, die Sie entweder auf eine OE oder ein Konto anwenden können. SCPs können allgemeine Zugriffskontrollen durchsetzen, wie z. B. die Beschränkung des Zugriffs auf bestimmte AWS-Regionen, die Verhinderung des Löschens von Ressourcen oder die Deaktivierung potenziell riskanter Serviceaktionen. SCPs, die Sie auf das Root-Verzeichnis Ihrer Organisation anwenden, wirken sich nur auf die Mitgliedskonten aus, nicht auf das Verwaltungskonto. SCPs regeln nur die Prinzipale innerhalb Ihrer Organisation. Ihre SCPs regeln keine Prinzipale außerhalb Ihrer Organisation, die auf Ihre Ressourcen zugreifen.

Ein weiterer Schritt ist die Verwendung von IAM-Ressourcenrichtlinien, um die verfügbaren Aktionen, die Sie für die von ihnen geregelten Ressourcen durchführen können, zusammen mit den Bedingungen, die der handelnde Prinzipal erfüllen muss, zu definieren. Dies kann so weit gefasst sein, dass alle Aktionen erlaubt sind, solange das Prinzipal zu Ihrer Organisation gehört (unter Verwendung des Bedingungsschlüssels PrincipalOrgId), oder so granular, dass nur bestimmte Aktionen von einer bestimmten IAM-Rolle erlaubt sind. Sie können einen ähnlichen Ansatz mit Bedingungen in IAM-Rollenvertrauensrichtlinien verfolgen. Wenn eine Vertrauensrichtlinie für eine Ressource oder Rolle explizit einen Prinzipal im selben Konto wie die Rolle oder Ressource benennt, die sie regelt, benötigt dieser Prinzipal keine angehängte IAM-Richtlinie, die dieselben Berechtigungen gewährt. Wenn der Prinzipal ein anderes Konto hat als die Ressource, dann benötigt der Prinzipal eine angehängte IAM-Richtlinie, die diese Berechtigungen gewährt.

Oft möchte ein Workload-Team die für seinen Workload erforderlichen Berechtigungen verwalten. Dazu muss es möglicherweise neue IAM-Rollen und Berechtigungsrichtlinien erstellen. Sie können den maximalen Umfang der Berechtigungen, die das Team gewähren darf, in einer IAM-Berechtigungsgrenze erfassen und dieses Dokument mit einer IAM-Rolle verknüpfen, die das Team dann zur Verwaltung seiner IAM-Rollen und Berechtigungen verwenden kann. Dieser Ansatz kann ihm die Freiheit geben, ihre Arbeit zu erledigen und gleichzeitig die Risiken eines administrativen IAM-Zugriffs verringern.

Ein detaillierterer Schritt ist die Implementierung von Techniken zur Verwaltung von privilegiertem Zugriff (Privileged Access Management, PAM) und temporärer erweiterter Zugriffsverwaltung (Temporary Elevated Access Management, TEAM). Ein Beispiel für PAM ist die Anforderung an Prinzipale, sich mehrfach zu authentifizieren, bevor sie privilegierte Aktionen durchführen. Weitere Informationen finden Sie unter Configuring MFA-protected API access. TEAM benötigt eine Lösung, die die Genehmigung und den Zeitrahmen verwaltet, in dem ein Prinzipal erweiterten Zugriff haben darf. Eine Möglichkeit besteht darin, den Prinzipal vorübergehend in die Vertrauensrichtlinie für eine IAM-Rolle aufzunehmen, die über einen erweiterten Zugriff verfügt. Ein anderer Ansatz besteht darin,

im Normalbetrieb die Berechtigungen, die einem Prinzipal von einer IAM-Rolle gewährt werden, mit einer Sitzungsrichtlinie einzuschränken und diese Einschränkung dann während des genehmigten Zeitfensters vorübergehend aufzuheben. Weitere Informationen über Lösungen, die AWS und ausgewählte Partner validiert haben, finden Sie unter Temporär erweiterter Zugriff.

Implementierungsschritte

1. Isolieren Sie Ihre Workloads und Umgebungen in separaten AWS-Konten.
2. Verwenden Sie SCPs, um die maximale Anzahl von Berechtigungen zu reduzieren, die Prinzipalen innerhalb der Mitgliedskonten Ihrer Organisation gewährt werden können.
   a. Wir empfehlen Ihnen, Ihre SCPs nach dem Prinzip der Erlaubnisliste zu schreiben, die alle Aktionen verweigert, außer denen, die Sie erlauben, und den Bedingungen, unter denen sie erlaubt sind. Beginnen Sie damit, die Ressourcen zu definieren, die Sie kontrollieren möchten, und setzen Sie den Effekt auf Verweigern. Verwenden Sie das NotAction-Element, um alle Aktionen zu verweigern, außer denen, die Sie angeben. Kombinieren Sie dies mit einer NotLike-Bedingung, um festzulegen, wann diese Aktionen erlaubt sind, falls zutreffend, wie z. B. StringNotLike und ArnNotLike.
   b. Siehe Service control policy examples.
3. Verwenden Sie IAM-Ressourcenrichtlinien, um den Geltungsbereich einzugrenzen und Bedingungen für zulässige Aktionen auf Ressourcen festzulegen. Verwenden Sie Bedingungen in IAM-Rollenvertrauensrichtlinien, um Einschränkungen für die Übernahme von Rollen zu erstellen.
4. Weisen Sie IAM-Berechtigungsgrenzen zu IAM-Rollen zu, die Workload-Teams dann zur Verwaltung ihrer eigenen Workloads IAM-Rollen und -Berechtigungen verwenden können.
5. Evaluieren Sie PAM- und TEAM-Lösungen auf der Grundlage Ihrer Bedürfnisse.

Ressourcen

Zugehörige Dokumente:

- Data perimeters on AWS
- Establish permissions guardrails using data perimeters
- Policy evaluation logic

Zugehörige Beispiele:

- Service control policy examples

Zugehörige Tools:

- AWS Solution: Temporary Elevated Access Management
- Validated security partner solutions for TEAM

SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus

Überwachen Sie die Berechtigungen, die Ihren Prinzipalen (Benutzern, Rollen und Gruppen) während ihres gesamten Lebenszyklus in Ihrer Organisation gewährt werden, und passen Sie sie an. Passen Sie die Gruppenmitgliedschaften an, wenn Benutzer ihre Rolle ändern, und entfernen Sie den Zugriff, wenn ein Benutzer die Organisation verlässt.

Gewünschtes Ergebnis: Sie überwachen und passen die Berechtigungen während des gesamten Lebenszyklus von Prinzipalen innerhalb der Organisation an und verringern so das Risiko unnötiger Privilegien. Sie gewähren den entsprechenden Zugriff, wenn Sie einen Benutzer anlegen. Sie ändern den Zugriff, wenn sich die Aufgaben des Benutzers ändern, und Sie entfernen den Zugriff, wenn der Benutzer nicht mehr aktiv ist oder die Organisation verlassen hat. Sie verwalten Änderungen an Ihren Benutzern, Rollen und Gruppen zentral. Sie verwenden die Automatisierung, um Änderungen in Ihren AWS-Umgebungen zu verbreiten.

Typische Anti-Muster:

- Sie gewähren Identitäten im Voraus übermäßige oder weitreichende Zugriffsrechte, die über das ursprünglich erforderliche Maß hinausgehen.
- Sie unterlassen die Überprüfung der Zugriffsprivilegien und passen sie an, wenn sich die Rollen und Verantwortlichkeiten der Identitäten im Laufe der Zeit ändern.
- Sie verlassen inaktive oder beendete Identitäten mit aktiven Zugriffsrechten. Dies erhöht das Risiko eines unbefugten Zugriffs.
- Sie automatisieren die Verwaltung von Identitätslebenszyklen nicht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Verwalten Sie die Zugriffsprivilegien, die Sie Identitäten (z. B. Benutzern, Rollen, Gruppen) gewähren, sorgfältig und passen Sie sie im Laufe ihres Lebenszyklus an. Dieser Lebenszyklus umfasst die anfängliche Onboarding-Phase, laufende Änderungen der Rollen und Verantwortlichkeiten und schließlich das Offboarding oder die Kündigung. Verwalten Sie den Zugriff proaktiv je nach Stadium

des Lebenszyklus, um die richtige Zugriffsstufe zu erhalten. Halten Sie sich an das Prinzip der geringsten Berechtigung, um das Risiko übermäßiger oder unnötiger Zugriffsberechtigungen zu verringern.

Sie können den Lebenszyklus von IAM users direkt innerhalb des AWS-Konto oder durch den Verbund von Ihrem Identitätsanbieter für die Belegschaft zu AWS-IAM Identity Center verwalten. Für IAM users können Sie innerhalb des AWS-Konto Benutzer und die damit verbundenen Berechtigungen erstellen, ändern und löschen. Für Verbundbenutzer können Sie IAM Identity Center verwenden, um deren Lebenszyklus zu verwalten, indem Sie Benutzer- und Gruppeninformationen vom Identitätsanbieter Ihrer Organisation mit dem Protokoll System für domänenübergreifendes Identitätsmanagement (System for Cross-Domain Identity Management, SCIM) synchronisieren.

SCIM ist ein offenes Standardprotokoll für die automatisierte Bereitstellung und Deprovisionierung von Benutzeridentitäten über verschiedene Systeme hinweg. Durch die Integration Ihres Identitätsanbieters mit IAM Identity Center unter Verwendung von SCIM können Sie Benutzer- und Gruppeninformationen automatisch synchronisieren und so sicherstellen, dass Zugriffsberechtigungen auf der Grundlage von Änderungen in der maßgeblichen Identitätsquelle Ihrer Organisation gewährt, geändert oder entzogen werden.

Wenn sich die Rollen und Zuständigkeiten der Mitarbeiter in Ihrer Organisation ändern, passen Sie ihre Zugriffsrechte entsprechend an. Sie können die Berechtigungssätze von IAM Identity Center verwenden, um verschiedene Job-Rollen oder -Verantwortlichkeiten zu definieren und sie mit den entsprechenden IAM-Richtlinien und -Berechtigungen zu verknüpfen. Wenn sich die Rolle eines Mitarbeiters ändert, können Sie die ihm zugewiesenen Berechtigungen aktualisieren, um die neuen Verantwortlichkeiten zu berücksichtigen. Vergewissern Sie sich, dass sie über den erforderlichen Zugriff verfügen, und halten Sie sich dabei an das Prinzip der geringsten Berechtigung.

Implementierungsschritte

1. Definieren und dokumentieren Sie einen Lebenszyklusprozess für die Zugriffsverwaltung, einschließlich Verfahren für die Gewährung des Erstzugriffs, regelmäßige Überprüfungen und das Offboarding.
2. Implementieren Sie IAM-Rollen, -Gruppen und -Berechtigungsgrenzen, um den Zugriff kollektiv zu verwalten und die maximal zulässigen Zugriffsstufen durchzusetzen.
3. Integrieren Sie einen Anbieter von Verbundidentitäten (wie Microsoft Active Directory, Okta, Ping Identity) als maßgebliche Quelle für Benutzer- und Gruppeninformationen mit IAM Identity Center.
4. Verwenden Sie das SCIM-Protokoll, um Benutzer- und Gruppeninformationen vom Identitätsanbieter mit dem Identitätsspeicher von IAM Identity Center zu synchronisieren.

5. Erstellen Sie in IAM Identity Center Berechtigungssätze, die verschiedene Jobrollen oder Verantwortlichkeiten in Ihrer Organisation repräsentieren. Definieren Sie die entsprechenden IAM-Richtlinien und -Berechtigungen für jeden Berechtigungssatz.
6. Führen Sie regelmäßige Zugriffsüberprüfungen, sofortigen Zugriffsentzug und eine kontinuierliche Verbesserung des Lebenszyklusprozesses der Zugriffsverwaltung ein.
7. Schulung und Sensibilisierung der Mitarbeiter für die bewährten Methoden der Zugriffsverwaltung.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter

Zugehörige Dokumente:

- Manage your identity source
- Manage identities in IAM Identity Center
- Using AWS Identity and Access Management Access Analyzer
- IAM Access Analyzer policy generation

Zugehörige Videos:

- AWS re:Inforce 2023 – Manage temporary elevated access with AWS IAM Identity Center
- AWS re:Invent 2022 – Simplify your existing workforce access with IAM Identity Center
- AWS re:Invent 2022 – Harness power of IAM policies & rein in permissions w/Access Analyzer

SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs

Überwachen Sie kontinuierlich Ergebnisse, die den öffentlichen und kontoübergreifenden Zugriff betreffen. Beschränken Sie den öffentlichen und kontoübergreifenden Zugriff ausschließlich auf Ressourcen, die diese Art von Zugriff benötigen.

Gewünschtes Ergebnis: Wissen, welche Ihrer AWS-Ressourcen für wen freigegeben sind. Überwachen und prüfen Sie kontinuierlich Ihre freigegebenen Ressourcen, um sicherzustellen, dass sie nur für autorisierte Prinzipale freigegeben sind.

Typische Anti-Muster:

- fehlendes Inventar gemeinsam genutzter Ressourcen
- Nichtbefolgung eines Prozesses zur Genehmigung von kontoübergreifendem oder öffentlichem Zugriff auf Ressourcen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Wenn sich Ihr Konto in AWS Organizations befindet, können Sie den Zugriff auf Ressourcen der gesamten Organisation, bestimmten Organisationseinheiten oder einzelnen Konten gewähren. Wenn Ihr Konto nicht zu einer Organisation gehört, können Sie Ressourcen für einzelne Konten freigeben. Sie können direkten kontoübergreifenden Zugriff mithilfe von Richtlinien gewähren, die an Ressourcen angefügt sind – (z. B. Amazon Simple Storage Service (Amazon S3)-Bucket-Richtlinien – oder indem Sie einem Prinzipal erlauben, eine IAM-Rolle in einem anderen Konto anzunehmen. Prüfen Sie bei der Verwendung von Ressourcenrichtlinien, dass der Zugriff nur autorisierten Prinzipalen gewährt ist. Definieren Sie einen Prozess für die Genehmigung aller Ressourcen, die öffentlich verfügbar sein müssen.

AWS Identity and Access Management Access Analyzer verwendet belegbare Sicherheit, um alle Zugriffspfade zu einer Ressource von außerhalb ihres Kontos zu identifizieren. Es überprüft Ressourcenrichtlinien kontinuierlich und meldet Ergebnisse des öffentlichen und kontoübergreifenden Zugriffs, um Ihnen die Analyse potenziell umfassender Zugriffe zu erleichtern. Erwägen Sie die Konfiguration von IAM Access Analyzer mit AWS Organizations, um die Transparenz aller Ihrer Konten sicherzustellen. IAM Access Analyzer ermöglicht Ihnen auch die Voranzeige der Ergebnisse vor der Bereitstellung von Ressourcenberechtigungen. So können Sie sicherstellen, dass mit den Richtlinienänderungen nur der beabsichtigte öffentliche und kontoübergreifende Zugriff auf Ihre Ressourcen gewährt wird. Beim Entwurf des Mehrkonten-Zugriffs können Sie mit Vertrauensrichtlinien steuern, in welchen Fällen eine Rolle angenommen werden kann. So können Sie etwa den Bedingungsschlüssel `PrincipalOrgId` verwenden, um den Versuch, eine Rolle von außerhalb Ihrer AWS Organizations anzunehmen, abzulehnen.

AWS Config kann Ressourcen melden, die nicht korrekt konfiguriert sind, und über AWS Config-Richtlinienprüfungen Ressourcen erkennen, für die der öffentliche Zugriff konfiguriert ist. Services wie AWS Control Tower und AWS Security Hub vereinfachen die Bereitstellung von Prüfungen und Integritätsschutz über AWS Organizations hinweg, um öffentlich zugängliche Ressourcen zu identifizieren und zu korrigieren. Beispielsweise verfügt AWS Control Tower über verwalteten Integritätsschutz, der erkennen kann, ob Amazon EBS-Snapshots von AWS-Konten wiederhergestellt werden können.

Implementierungsschritte

- Erwägen Sie die Aktivierung von AWS Config für AWS Organizations: AWS Config ermöglicht die Aggregierung von Ergebnissen mehrerer Konten in einer AWS Organizations zu einem delegierten Administratorkonto. Dies sorgt für eine umfassende Sicht und ermöglicht die Bereitstellung von AWS-Config-Regeln über mehrere Konten hinweg, um öffentlich zugängliche Ressourcen zu erkennen.
- Konfiguration von AWS Identity and Access Management Access Analyzer: IAM Access Analyzer hilft Ihnen, die Ressourcen in Ihrer Organisation und Ihren Konten zu identifizieren, z. B. Amazon S3-Buckets oder IAM-Rollen, die mit einer externen Entität geteilt werden.
- Verwenden Sie die automatische Korrektur in AWS Config, um auf Änderungen in der Konfiguration des öffentlichen Zugriffs auf Amazon S3-Buckets reagieren zu können: Sie können die Einstellungen zur Blockierung des öffentlichen Zugriffs für Amazon S3-Buckets automatisch erneut aktivieren.
- Implementierung von Überwachung und Benachrichtigung, wenn Amazon S3-Buckets öffentlich zugänglich werden: Sie müssen über Überwachungs- und Benachrichtigungsmechanismen verfügen, um zu erkennen, wenn Amazon S3 Block Public Access deaktiviert ist, und wenn Amazon S3-Buckets öffentlich zugänglich werden. Dazu können Sie bei Verwendung von AWS Organizations eine Servicekontrollrichtlinie erstellen, die Änderungen an Amazon S3-Richtlinien für den öffentlichen Zugriff verhindern. AWS Trusted Advisor prüft auf Amazon S3-Buckets, die Open-Access-Berechtigungen haben. Bucket-Berechtigungen, die allen Benutzern den Zugriff zum Hochladen/Löschen einräumen, bergen ein hohes Potenzial für Sicherheitsrisiken, da alle Personen Elemente in einem Bucket hinzufügen, ändern oder löschen können. Die Prüfung von Trusted Advisor untersucht explizite Bucket-Berechtigungen und zugeordnete Bucket-Richtlinien, die die Bucket-Berechtigungen möglicherweise überschreiben. Sie können auch mit AWS Config Ihre Amazon S3-Buckets für den öffentlichen Zugriff überwachen. Für weitere Informationen vgl. Verwendung von AWS Config zur Überwachung und Reaktion auf Amazon S3-Buckets mit öffentlicher Zugänglichkeit. Bei der Prüfung der Zugänglichkeit ist es wichtig, zu berücksichtigen, welche Art von Daten Amazon S3-Buckets enthalten. Amazon Macie hilft dabei, sensitive Daten wie etwa PII, PHI und Anmeldeinformationen wie private oder AWS-Schlüssel zu erkennen und zu schützen.

Ressourcen

Zugehörige Dokumente:

- Verwendung von AWS Identity and Access Management Access Analyzer

- AWS Control Tower Controls Library
- AWS Foundational Security Best Practices Standard
- AWS Config Managed Rules
- Prüfungsreferenz von AWS Trusted Advisor
- Monitoring AWS Trusted Advisor check results with Amazon EventBridge (Überwachen der Prüfergebnisse von AWS Trusted Advisor mit Amazon EventBridge)
- Managing AWS Config Rules Across All Accounts in Your Organization (Verwaltung von AWS Config-Regeln für alle Konten in Ihrer Organisation)
- AWS Config und AWS Organizations

Zugehörige Videos:

- Best Practices for securing your multi-account environment(Bewährte Methoden für den Schutz Ihrer Mehrkonten-Umgebung)
- Dive Deep into IAM Access Analyzer (Tiefer Einblick in IAM Access Analyzer)

SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation

Wenn die Anzahl der Workloads zunimmt, müssen Sie möglicherweise den Zugriff auf Ressourcen in diesen Workloads ausweiten oder diese Ressourcen mehrfach über mehrere Konten hinweg zugänglich machen. Möglicherweise haben Sie Konstrukte zur Untergliederung Ihrer Umgebung, etwa für Entwicklungs-, Test- und Produktionsumgebungen. Solche Trennungskonstrukte schränken Sie jedoch nicht in der Lage ein, sicher zu teilen. Durch die gemeinsame Nutzung sich überschneidender Ressourcen können Sie übermäßigen betrieblichen Aufwand reduzieren und eine konsistente Umgebung schaffen, ohne dass Sie raten müssen, was Sie vielleicht versäumt haben, wenn Sie eine Ressource mehrmals erstellen.

Gewünschtes Ergebnis: Minimierung unbeabsichtigter Zugriffe durch Verwendung sicherer Verfahren für die Freigabe von Ressourcen innerhalb Ihrer Organisation und die Unterstützung Ihrer Initiative zur Verhinderung von Datenverlusten. Reduzieren Sie Ihren organisatorischen Aufwand gegenüber der Verwaltung einzelner Komponenten, senken Sie die Zahl von Fehlern durch das manuelle mehrmalige Erstellen identischer Ressourcen, und steigern Sie die Skalierbarkeit Ihrer Workloads. Sie können von kürzeren Lösungszeiten in Szenarien mit mehreren Fehlerpunkten profitieren und Ihr Vertrauen in die Bestimmung erhöhen, wann eine Komponente nicht mehr benötigt wird. Anleitungen zur Analyse extern freigegebener Ressourcen finden Sie unter SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs.

Typische Anti-Muster:

- Fehlen eines Prozesses für die kontinuierliche Überwachung und die automatische Benachrichtigung bei unerwarteten externen Freigaben
- Fehlen einer Basislinie dazu, was freigegeben werden sollte und was nicht
- die standardmäßige Verwendung einer sehr offenen Richtlinie, anstatt Ressourcen explizit freizugeben, wenn sie benötigt werden
- manuelle Erstellung grundlegender Ressourcen bei Bedarf, die sich überlappen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Gestalten Sie Ihre Zugriffskontrollen und -muster so, dass die Nutzung freigegebener Ressourcen kontrolliert wird und nur mit vertrauenswürdigen Entitäten möglich ist. Überwachen Sie freigegebene Ressourcen, prüfen Sie kontinuierlich den Zugriff darauf und erhalten Sie Benachrichtigungen bei unangemessenen oder unerwarteten Freigaben. Lesen Sie Analysieren öffentlicher und kontoübergreifender Zugriffe, um Richtlinien einzurichten, die externe Zugriffe auf die Ressourcen beschränken, für die dies erforderlich ist, und um einen Prozess zur kontinuierlichen Überwachung und Benachrichtigung einzurichten.

Die kontoübergreifende Freigabe innerhalb von AWS Organizations wird von einer Reihe von AWS-Services unterstützt, wie etwa AWS Security Hub, Amazon GuardDuty und AWS Backup. Diese Services ermöglichen die Freigabe von Daten für ein zentrales Konto, ihre Zugänglichkeit von einem zentralen Konto aus sowie die Verwaltung von Ressourcen und Daten von einem zentralen Konto aus. Beispielsweise kann AWS Security Hub Ergebnisse von einzelnen Konten auf ein zentrales Konto übertragen, wo Sie alle Ergebnisse einsehen können. AWS Backup kann eine Sicherungskopie einer Ressource kontoübergreifend freigeben. Sie können mit AWS Resource Access Manager (AWS RAM) weitere verbreitete Ressourcen freigeben, wie etwa VPC-Subnetze und Transit Gateway-Anhänge, AWS Network Firewall oder Amazon SageMaker AI-Pipelines.

Um Ihr Konto darauf zu beschränken, Ressourcen nur innerhalb Ihrer Organisation freizugeben, verwenden Sie Service Control Policies (SCPs, Service-Kontrollrichtlinien), um den Zugriff auf externe Prinzipale zu verhindern. Kombinieren Sie bei der Freigabe von Ressourcen identitätsbasierte Kontrollen und Netzwerk-Kontrollen zur Erstellung eines Datenperimeters für Ihre Organisation zum Schutz gegen unbeabsichtigte Zugriffe. Ein Datenperimeter ist ein Satz von präventiven Maßnahmen zum Integritätsschutz, die dabei helfen, sicherzustellen, dass nur vertrauenswürdige Identitäten aus erwarteten Netzwerken auf vertrauenswürdige Ressourcen

zugreifen. Diese Kontrollen begrenzen, welche Ressourcen gemeinsam genutzt werden, und verhindern die gemeinsame Nutzung oder Offenlegung von Ressourcen, die nicht zugelassen werden sollten. So können Sie Beispielsweise als Teil ihres Datenperimeters VPC-Endpunktrichtlinien und die Bedingung `AWS:OrincipalOrgID` verwenden, um sicherzustellen, dass die auf Ihre Amazon S3-Buckets zugreifenden Identitäten zu Ihrer Organisation gehören. Es ist wichtig zu wissen, dass SCPs nicht für serviceverknüpfte Rollen (LSR) oder AWS-Service-Prinzipale gelten.

Bei Verwendung von Amazon S3 sollten Sie ACLs für Ihren Amazon S3-Bucket deaktivieren und IAM-Richtlinien für die Einrichtung der Zugriffskontrollen verwenden. Für die Einschränkung des Zugriffs auf einen Amazon S3-Ursprung von Amazon CloudFront aus migrieren Sie von der Ursprungszugriffsidentität (OAI) zur Ursprungszugriffssteuerung (OAC), die zusätzliche Funktionen wie beispielsweise die serverseitige Verschlüsselung mit AWS Key Management Service unterstützt.

In manchen Fällen möchten Sie möglicherweise die Freigabe von Ressourcen außerhalb Ihrer Organisation zulassen oder einer Drittpartei den Zugriff auf Ihre Ressourcen gewähren. Präskriptive Anleitungen zur Verwaltung von Berechtigungen für die externe Freigabe von Ressourcen finden Sie unterBerechtigungsmanagement.

Implementierungsschritte

1. Nutzen Sie AWS Organizations.

   AWS Organizations ist ein Kontoverwaltungsservice, mit dem Sie mehrere AWS-Konten zu einer zentral erstellten und verwalteten Organisation konsolidieren können. Sie können Ihre Konten in Organisationseinheiten (OUs) gruppieren und jeder OU unterschiedliche Richtlinien zuweisen, um Ihre Budget-, Sicherheits- und Compliance-Anforderungen zu erfüllen. Sie können auch steuern, wie AWS-Services für künstliche Intelligenz (KI) und Machine Learning (ML) Daten erfassen und speichern können, und die Mehrkonten-Verwaltung der mit Organizations integrierten AWS-Services verwenden.

2. Integrieren Sie AWS Organizations mit AWS-Services.

   Wenn Sie einen AWS-Service zur Ausführung von Aufgaben in Ihrem Namen in den Mitgliedskonten Ihrer Organisation aktivieren, erstellt AWS Organizations eine serviceverknüpfte IAM-Rolle für den jeweiligen Service in jedem Mitgliedskonto. Sie sollten den vertrauenswürdigen Zugriff mit der AWS Management Console, den AWS-APIs oder der AWS CLI verwalten. Präskriptive Anleitungen zur Einrichtung vertrauenswürdigen Zugangs finden Sie unter Verwendung von AWS Organizations mit anderen AWS-Services und unter AWS-Services, die Sie mit Organizations verwenden können.

3. Richten Sie einen Datenperimeter ein.

   Der AWS-Perimeter wird typischerweise als von AWS Organizations verwaltete Organisation repräsentiert. Zusammen mit On-Premises-Netzwerken und -Systemen ist der Zugriff auf AWS-Ressourcen das, was viele als den Perimeter von My AWS bezeichnen. Das Ziel des Perimeters besteht darin, zu überprüfen, ob der Zugriff erlaubt ist, wenn die Identität und die Ressource vertrauenswürdig sind und es sich um ein erwartetes Netzwerk handelt.

   a. Definieren und implementieren Sie die Perimeter.

      Befolgen Sie die Schritte unter Perimeter-Implementierung im Whitepaper zum Thema „Aufbau eines Perimeters in AWS“ für jede Autorisierungsbedingung. Eine präskriptive Anleitung zum Schutz von Netzwerkebenen finden Sie unter Schutz von Netzwerken.

   b. Sorgen Sie für kontinuierliche Überwachung und Benachrichtigung.

      AWS Identity and Access Management Access Analyzer hilft bei der Identifizierung von Ressourcen in Ihrer Organisation und in Konten, die gemeinsam mit externen Entitäten genutzt werden. Sie können IAM Access Analyzer mit AWS Security Hub integrieren, um Ergebnisse für eine Ressource von IAM Access Analyzer zu Security Hub zu senden und zu aggregieren und so die Sicherheitssituation ihrer Umgebung zu analysieren. Aktivieren Sie für die Integration IAM Access Analyzer und Security Hub in jeder Region und in jedem Konto. Sie können auch mit AWS-Config-Regeln die Konfiguration prüfen und die jeweilige Partei mit Amazon Q Developer in chat applications mit AWS Security Hub benachrichtigen. Anschließend können Sie mit Automatisierungsdokumenten von AWS Systems Manager nicht-konforme Ressourcen reparieren.

   c. Präskriptive Anleitungen zur Überwachung und kontinuierlichen Beratung zu extern freigegebenen Ressourcen finden Sie unter Analyse des öffentlichen und kontoübergreifenden Zugriffs.

4. Verwenden Sie die Ressourcenfreigabe in AWS-Services, und sorgen Sie für entsprechende Einschränkungen.

   Viele AWS-Services erlauben die Freigabe von Ressourcen für ein anderes Konto oder die Ausrichtung auf eine Ressource in einem anderen Konto, wie etwa Amazon Machine Images (AMIs) und AWS Resource Access Manager (AWS RAM). Schränken Sie die `ModifyImageAttribute`-API auf die Angabe der vertrauenswürdigen Konten für die Freigabe des AMI ein. Geben Sie die Bedingung `ram:RequestedAllowsExternalPrincipals` bei Verwendung von AWS RAM an, um die Freigabe auf Ihre Organisation zu beschränken und

Zugriffe von nicht vertrauenswürdigen Entitäten zu verhindern. Präskriptive Anleitungen und Überlegungen dazu finden Sie unter Ressourcenfreigabe und externe Ziele.

5. Verwenden Sie AWS RAM für sichere Freigaben in einem Konto oder mit anderen AWS-Konten.

   AWS RAM hilft bei der sicheren Freigabe der Ressourcen, die Sie erstellt haben, mit Rollen und Benutzern in Ihrem Konto sowie mit anderen AWS-Konten. In einer Mehrkonten-Umgebung ermöglicht AWS RAM die einmalige Erstellung einer Ressource und ihre Freigabe für andere Konten. Dies reduziert Ihren operationalen Aufwand und sorgt für Konsistenz, Transparenz und Prüfbarkeit durch Integrationen mit Amazon CloudWatch und AWS CloudTrail, die bei Verwendung eines kontoübergreifenden Zugriffs nicht möglich sind.

   Wenn Sie Ressourcen bereits mit einer ressourcenbasierten Richtlinie freigegeben haben, können Sie mit der `PromoteResourceShareCreatedFromPolicy`-API oder einem Äquivalent die Ressourcenfreigabe zu einer vollständigen AWS RAM-Ressourcenfreigabe erhöhen.

   In manchen Fällen müssen Sie möglicherweise weitere Schritte unternehmen, um Ressourcen freizugeben. So müssen Sie etwa für die Freigabe eines verschlüsselten Snapshots einen AWS KMS-Schlüssel freigeben.

Ressourcen

Zugehörige bewährte Methoden:

- SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs
- SEC03-BP09 Sicheres Teilen von Ressourcen mit Dritten
- SEC05-BP01 Erstellen von Netzwerkebenen

Zugehörige Dokumente:

- Bucket-Besitzer gewährt kontoübergreifende Berechtigung für Objekte, die er nicht besitzt
- Verwendung von Vertrauensrichtlinien mit IAM
- Erstellen von Datenperimetern auf AWS
- Verwenden einer externen ID, um Dritten Zugriff auf Ihre AWS-Ressourcen zu gewähren
- AWS-Services, die Sie mit AWS Organizations verwenden können
- Einrichten eines Datenperimeters auf AWS: Zulassen ausschließlich vertrauenswürdiger Identitäten für den Zugriff auf Unternehmensdaten

Zugehörige Videos:

- Granular Access with AWS Resource Access Manager (Granulärer Zugriff mit AWS Resource Access Manager)
- Securing your data perimeter with VPC endpoints (Schutz Ihres Datenperimeters mit VPC-Endpunkten)
- Establishing a data perimeter onAWS(Einrichten eines Datenperimeters auf AWS)

Zugehörige Tools:

- Beispiele für eine Datenperimeterrichtlinie

SEC03-BP09 Sicheres Teilen von Ressourcen mit Dritten

Die Sicherheit Ihrer Cloud-Umgebung endet nicht bei Ihrer Organisation. Möglicherweise stützt sich Ihre Organisation auf eine Drittpartei, um einen Teil Ihrer Daten zu verwalten. Das Berechtigungsmanagement für das von Dritten verwaltete System sollte dem Prinzip des Just-in-time-Zugriffs und dem der geringsten Berechtigung mit temporären Anmeldeinformationen folgen. Durch die enge Zusammenarbeit mit einer Drittpartei können Sie die möglichen Auswirkungen und das Risiko unbeabsichtigter Zugriffe gemeinsam senken.

Gewünschtes Ergebnis: Langfristige AWS Identity and Access Management (IAM)-Anmeldeinformationen, IAM-Zugriffsschlüssel und geheime Schlüssel, die einem Benutzer zugeordnet sind, können von allen verwendet werden, sofern sie gültig und aktiv sind. Die Verwendung einer IAM-Rolle und temporärer Anmeldeinformationen hilft bei der Verbesserung Ihrer allgemeinen Sicherheitsposition durch Reduzierung des Aufwands für die Verwaltung langfristiger Anmeldeinformationen und des operationalen Overheads dieser sensiblen Details. Durch die Verwendung einer universell eindeutigen Kennung (UUID) für die externe ID in der IAM-Vertrauensrichtlinie und die Anbindung der IAM-Richtlinien an die IAM-Rolle unter Ihrer Kontrolle können Sie prüfen und sicherstellen, dass der der Drittpartei gewährte Zugriff nicht zu umfangreich ist. Anleitungen zur Analyse extern freigegebener Ressourcen finden Sie unter SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs.

Typische Anti-Muster:

- Verwendung der Standard-IAM-Vertrauensrichtlinie ohne Bedingungen
- Verwenden langfristiger IAM-Anmeldeinformationen und Zugriffsschlüssel

- Wiederverwendung externer IDs

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Möglicherweise möchten Sie die Freigabe von Ressourcen außerhalb von AWS Organizations zulassen oder einer Drittpartei den Zugriff auf Ihr Konto gewähren. So könnte etwa eine Drittpartei eine Überwachungslösung bereitstellen, die auf Ressourcen in Ihrem Konto zugreifen muss. In solchen Fällen sollten Sie eine kontoübergreifende IAM-Rolle erstellen, die nur über die von der Drittpartei benötigten Berechtigungen verfügt. Definieren Sie dazu eine Vertrauensrichtlinie mit der externen ID-Bedingung. Wenn eine externe ID verwendet wird, können Sie oder die Drittpartei eine eindeutige ID für jede(n) Kunden, Drittpartei oder Tenancy generieren. Die eindeutige ID sollte nach ihrer Erstellung ausschließlich von Ihnen kontrolliert werden. Die Drittpartei muss einen Prozess implementieren, durch den die externe ID in sicherer, prüfbarer und reproduzierbarer Weise dem Kunden zugeordnet wird.

Sie können auch IAM Roles Anywhere verwenden, um IAM-Rollen für Anwendungen außerhalb von AWS zu verwalten, die AWS-APIs verwenden.

Wenn die Drittpartei keinen Zugriff mehr auf Ihre Umgebung benötigt, entfernen Sie die Rolle. Vermeiden Sie die Weitergabe langfristiger Anmeldeinformationen an Dritte. Achten Sie auf andere AWS-Services, die die Freigabe unterstützen. Beispielsweise erlaubt AWS Well-Architected Tool die Freigabe eines Workloads für andere AWS-Konten, und AWS Resource Access Manager hilft Ihnen bei der sicheren Freigabe einer AWS-Ressource, deren Eigentümer Sie sind, für andere Konten.

Implementierungsschritte

1. Verwenden Sie kontoübergreifende Rollen, um Zugriff auf externe Konten zu gewähren.

   Kontoübergreifende Rollen reduzieren den Umfang sensibler Informationen, die von externen Konten und Drittparteien für deren Kunden gespeichert werden. Kontoübergreifende Rollen ermöglichen die sichere Gewährung des Zugriffes auf AWS-Ressourcen in Ihrem Konto für Drittparteien wie etwa AWS Partners oder andere Konten in Ihrer Organisation, bei gleichzeitiger Wahrung der Möglichkeit, diesen Zugriff zu verwalten und zu überprüfen.

   Möglicherweise stellt Ihnen die Drittpartei Dienstleistungen aus einer hybriden Infrastruktur heraus bereit oder ruft Daten zu einem anderen Standort ab. IAM Roles Anywhere hilft Ihnen bei der

Aktivierung von Workloads Dritter zur sicheren Interaktion mit Ihren AWS-Workloads und zur weiteren Reduzierung der Erfordernis langfristiger Anmeldeinformationen.

Sie sollten keine langfristigen Anmeldeinformationen oder mit Benutzern verbundene Zugriffsschlüssel für die externe Gewährung des Zugriffs auf Konten verwenden. Verwenden Sie stattdessen kontoübergreifende Rollen, um kontoübergreifenden Zugriff zu gewähren.

2. Verwenden Sie eine externe ID mit Drittparteien.

   Die Verwendung einer externen ID ermöglicht Ihnen, in einer IAM-Vertrauensrichtlinie festzulegen, wer eine Rolle annehmen kann. Die Vertrauensrichtlinie kann verlangen, dass der Benutzer, der die Rolle annimmt, die Bedingung und das Ziel seiner Aktivität bestätigt. Sie bietet dem Kontoinhaber auch die Möglichkeit, die anzunehmende Rolle nur unter bestimmten Umständen zuzulassen. Die primäre Funktion der externen ID besteht darin, das Confused-Deputy-Problem anzugehen und zu verhindern.

   Verwenden Sie eine externe ID, wenn Sie AWS-Konto-Eigentümer sind und eine Rolle für eine Drittpartei konfiguriert haben, die neben Ihrem auf andere AWS-Konten zugreift, oder wenn Sie Rollen für verschiedene Kunden annehmen. Arbeiten Sie zusammen mit der Drittpartei oder AWS Partner an der Einrichtung einer externen ID-Bedingung für die IAM-Vertrauensrichtlinie.

3. Verwenden Sie universell eindeutige externe IDs.

   Implementieren Sie einen Prozess, der für externe IDs zufällige und eindeutige Werte generiert, etwa eine universell eindeutige Kennung (UUID). Eine Drittpartei, die externe IDs für verschiedene Kunden wiederverwendet, behebt das Confused-Deputy-Problem nicht, da Kunde A möglicherweise unter Verwendung des Rollen-ARN von Kunde B zusammen mit der duplizierten externen ID die Daten von Kunde B einsehen kann. In einer Multi-Tenant-Umgebung, in der eine Drittpartei mehrere Kunden mit verschiedenen AWS-Konten unterstützt, muss die Drittpartei eine andere eindeutige ID als die externe ID für jedes AWS-Konto verwenden. Die Drittpartei ist für das Erkennen doppelter externer IDs und die sichere Zuordnung jedes Kunden zur entsprechenden externen ID verantwortlich. Die Drittpartei muss durch Testen sicherstellen, dass sie die Rolle nur annehmen kann, wenn die externe ID angegeben wird. Die Drittpartei sollte den ARN der Kundenrolle und die externe ID nicht speichern, bis die externe ID benötigt wird.

   Die externe ID wird nicht als Secret behandelt, ihr Wert darf aber nicht leicht zu erraten sein wie etwa eine Telefonnummer, ein Name oder eine Konto-ID. Machen Sie die externe ID zu einem schreibgeschützten Feld, damit sie nicht für illegitime Einrichtungen geändert werden kann.

Sie oder die Drittpartei können/kann die externe ID generieren. Richten Sie einen Prozess ein, um festzulegen, wer für die Generierung der ID verantwortlich ist. Unabhängig von der Entität, die die externe ID erstellt, setzt die Drittpartei Eindeutigkeit und Formate in konsistenter Weise für alle Kunden durch.

4. Nehmen Sie von Kunden bereitgestellte langfristige Anmeldeinformationen außer Betrieb.

   Beenden Sie die Verwendung langfristiger Anmeldeinformationen, und verwenden Sie kontoübergreifende Rollen oder IAM Roles Anywhere. Wenn Sie langfristige Anmeldeinformationen verwendet müssen, formulieren Sie einen Plan für die Migration rollenbasierter Zugriffe. Einzelheiten zur Verwaltung von Schlüsseln finden Sie unterIdentitätsmanagement. Arbeiten Sie auch mit Ihrem AWS-Konto-Team und der Drittpartei daran, ein Runbook zur Risikodämpfung zu erstellen. Präskriptive Anleitungen zur Reaktion auf mögliche Auswirkungen von Sicherheitsvorfällen finden Sie unter Vorfallbehandlung.

5. Prüfen Sie, ob die Einrichtung über präskriptive Anleitungen verfügt oder automatisiert ist.

   Die für den kontoübergreifenden Zugriff in Ihren Konten erstellte Richtlinie muss dem Prinzip der geringsten Berechtigungen folgen. Die Drittpartei muss ein Rollenrichtliniendokument oder einen automatisierten Einrichtungsmechanismus bereitstellen, der eine AWS CloudFormation-Vorlage oder ein Äquivalent verwendet. Dies reduziert die Gefahr von Fehlern durch die manuelle Erstellung von Richtlinien und bietet einen Überwachungspfad. Weitere Informationen zur Verwendung einer AWS CloudFormation-Vorlage für die Erstellung kontoübergreifender Rollen finden Sie unter Kontoübergreifende Rollen.

   Die Drittpartei muss einen automatisierten und prüfbaren Einrichtungsmechanismus bereitstellen. Sie sollten jedoch die Einrichtung der Rolle automatisieren, indem Sie das Rollenrichtliniendokument verwenden, das den erforderlichen Zugriff angibt. Sie sollten mit der AWS CloudFormation-Vorlage oder einem Äquivalent Änderungen überwachen, mit besonderem Augenmerk auf „Drift Detection“.

6. Berücksichtigen Sie Änderungen.

   Ihre Kontostruktur und Ihr Bedarf an einer Drittpartei bzw. deren Serviceangebots können sich über Nacht ändern. Sie sollten Änderungen und Ausfälle antizipieren und mit den richtigen Personen, Prozessen und Technologielösungen entsprechend planen. Prüfen Sie regelmäßig das von Ihnen bereitgestellte Zugriffsniveau und implementieren Sie Erkennungsverfahren, die Sie auf unerwartete Änderungen aufmerksam machen. Überwachen und prüfen Sie die Verwendung der externen Rolle und den Datenspeicher der externen IDs. Sie sollten darauf vorbereitet sein, den Zugriff der Drittpartei temporär oder dauerhaft zu widerrufen, wenn sich

unerwartete Änderungen oder Zugriffsmuster ergeben. Messen Sie auch die Auswirkungen Ihrer Widerrufaktion, einschließlich der dafür benötigten Zeit, der involvierten Personen, der Kosten und der Auswirkungen auf andere Ressourcen.

Präskriptive Anleitungen zu Erkennungsverfahren finden Sie unter Bewährte Erkennungsmethoden.

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP02 Verwenden von temporären Anmeldeinformationen
- SEC03-BP05 Definieren eines Integritätsschutzes für Berechtigungen in Ihrer Organisation
- SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus
- SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs
- SEC04 Detection

Zugehörige Dokumente:

- Bucket-Besitzer gewährt kontoübergreifende Berechtigung für Objekte, die er nicht besitzt
- Verwendung von Vertrauensrichtlinien mit IAM-Rollen
- Delegieren des Zugriffs in allen AWS-Konten mithilfe von IAM-Rollen
- Wie greife ich mit IAM auf Ressourcen in einem anderen AWS-Konto zu?
- Bewährte Sicherheitsmethoden in IAM
- Logik für die kontenübergreifende Richtlinienauswertung
- Verwenden einer externen ID, um Dritten Zugriff auf Ihre AWS-Ressourcen zu gewähren
- Collecting Information from AWS CloudFormation Resources Created in External Accounts with Custom Resources (Erfassen von Informationen von in externen Konten mit benutzerdefinierten Ressourcen erstellten AWS-CloudFormation-Ressourcen)
- Securely Using External ID for Accessing AWS Accounts Owned by Others (Sichere Verwendung einer externen ID für den Zugriff auf AWS-Konten, die anderen gehören)
- Extend IAM roles to workloads outside of IAM with IAM Roles Anywhere (Erweitern von IAM-Rollen auf Workloads außerhalb von IAM mit IAM Roles Anywhere)

Zugehörige Videos:

- How do I allow users or roles in a separate AWS-Konto access to my AWS-Konto? (Wie gewähre ich Benutzern oder Rollen in einem separaten AWS-Konto Zugriff auf mein AWS-Konto?)
- AWS re:Invent 2018: Become an IAM Policy Master in 60 Minutes or Less (AWS re:Invent 2018: Werden Sie in höchstens 60 Minuten zum IAM-Richtlinienexperten)
- AWS Knowledge Center Live: IAM Best Practices and Design Decisions (AWS Knowledge Center Live: Bewährte IAM-Methoden und -Entwurfsentscheidungen)

Zugehörige Beispiele:

- Well-Architected Lab - Lambda cross account IAM role assumption (Level 300) (Well-Architected Lab – Lambda-kontoübergreifende IAM-Rollenannahme)
- Configure cross-account access to Amazon DynamoDB (Konfigurieren des kontoübergreifenden Zugriffs auf Amazon DynamoDB)
- AWS STS Network Query Tool

Erkennung

Frage

- SEC 4. Wie erkennen und untersuchen Sie Sicherheitsereignisse?

SEC 4. Wie erkennen und untersuchen Sie Sicherheitsereignisse?

Erfassen und analysieren Sie Ereignisse mithilfe von Protokollen und Kennzahlen, um Einblick zu erhalten. Ergreifen Sie Maßnahmen bei Sicherheitsereignissen und potenziellen Bedrohungen, um Ihren Workload zu schützen.

Bewährte Methoden

- SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung
- SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten
- SEC04-BP03 Korrelieren und Anreichern von Sicherheitswarnmeldungen
- SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen

SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung

Bewahren Sie Protokolle zu Sicherheitsereignissen von Services und Anwendungen auf. Dies ist ein grundlegendes Sicherheitsprinzip für Prüfungs-, Untersuchungs- und betriebliche Anwendungsfälle und eine übliche Sicherheitsanforderung gemäß Governance-, Risiko- und Compliance (GRC)-Standards, -Richtlinien und -Prozeduren.

Gewünschtes Ergebnis: Eine Organisation sollte in der Lage sein, Sicherheitsereignisprotokolle in zuverlässiger und konsistenter Weise sowie zeitnah aus AWS-Services und -Anwendungen abzurufen, wenn diese für einen internen Prozess oder eine Verpflichtung wie etwa die Reaktion auf einen Sicherheitsvorfall benötigt werden. Erwägen Sie die Zentralisierung von Protokollen für bessere betriebliche Ergebnisse.

Typische Anti-Muster:

- Protokolle werden dauerhaft gespeichert oder zu früh gelöscht.
- jeder kann auf die Protokolle zugreifen.
- Nutzung ausschließlich manueller Prozesse für die Verwaltung und Verwendung von Protokollen
- Speichern aller Arten von Protokollen nur für den Fall, dass sie benötigt werden
- Prüfung der Protokollintegrität nur bei Bedarf

Vorteile der Nutzung dieser bewährten Methode: Implementieren Sie einen Mechanismus für die Ursachenanalyse (RCA) für Sicherheitsvorfälle sowie eine Evidenzquelle für Ihre Governance-, Risiko- und Compliance-Anforderungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Bei einer Sicherheitsuntersuchung oder in anderen bedarfsabhängigen Anwendungsfällen müssen Sie relevante Protokolle konsultieren können, um alle Aspekte und den Zeitrahmen des Vorfalls zu verstehen. Protokolle werden auch für die Generierung von Alarmen benötigt, die darauf hinweisen, dass bestimmte Ereignisse vorgekommen sind. Es ist sehr wichtig, Abfrage-, Abruf- sowie Benachrichtigungsmechanismen auszuwählen, zu aktivieren, zu speichern und einzurichten.

Implementierungsschritte

- Wählen und aktivieren Sie Protokollquellen. Vor einer Sicherheitsuntersuchung müssen Sie relevante Protokolle erfassen, um die Aktivitäten in einem AWS-Konto retroaktiv rekonstruieren zu können. Wählen und aktivieren Sie für Ihre Workloads relevante Protokollquellen.

  Die Kriterien für die Auswahl der Protokollquelle sollten auf den Anwendungsfällen Ihres Unternehmens basieren. Richten Sie einen Trail für jedes AWS-Konto mit AWS CloudTrail oder einen AWS Organizations-Trail ein, und konfigurieren Sie dafür einen Amazon S3-Bucket.

  AWS CloudTrail ist ein Protokollservice, der API-Aufrufe an ein AWS-Konto verfolgt und AWS-Serviceaktivitäten erfasst. Dieser ist standardmäßig mit einer 90-tägigen Aufbewahrung von Managementereignissen aktiviert, die über den CloudTrail-Ereignisverlauf mit der AWS Management Console, der AWS CLI oder einem AWS-SDK abgerufen werden können. Für längere Aufbewahrungszeiten und Abrufbarkeit von Datenereignissen erstellen Sie einen CloudTrail-Trail und verbinden diesen mit einem Amazon S3-Bucket sowie optional mit einer Amazon CloudWatch-Protokollgruppe. Sie können auch einen CloudTrail-Lake erstellen, der CloudTrail-Protokolle bis zu sieben Jahre lang aufbewahrt und eine SQL-basierte Abfragemöglichkeit bietet.

  AWS empfiehlt, dass Kunden, die eine VPC nutzen, Netzwerkdatenverkehr- und DNS-Protokolle mit VPC Flow Logs und Amazon Route 53 Resolver Query Logs einrichten und diese per Stream zu einem Amazon S3-Bucket oder einer CloudWatch-Protokollgruppe leiten. Sie können ein VPC-Flow-Protokoll für eine VPC, ein Subnetz oder eine Netzwerkschnittstelle erstellen. Für VPC-Flow-Protokolle können Sie wählen, wie und wo Flow-Protokolle verwendet werden sollen, um Kosten zu sparen.

  AWS CloudTrail-Protokolle, VPC-Flow-Protokolle und Route 53 Resolver Query Logs sind die grundlegenden Protokollquellen zur Unterstützung von Sicherheitsuntersuchungen in AWS. Sie können auch Amazon Security Lake verwenden, um diese Protokolldaten zu erfassen, zu normalisieren und im Apache Parquet-Format und mit dem Open Cybersecurity Schema Framework (OCSF) zu speichern, das Abfragen ermöglicht. Security Lake unterstützt auch andere AWS-Protokolle sowie Protokolle aus Drittquellen.

  AWS-Services können Protokolle generieren, die von den grundlegenden Protokollquellen nicht erfasst werden, wie etwa Protokolle von Elastic Load Balancing, AWS WAF-Protokolle, Recorder-Protokolle von AWS Config, Amazon GuardDuty-Ergebnisse, Amazon Elastic Kubernetes Service (Amazon EKS)-Prüfprotokolle sowie Instance-Betriebssystem- und Anwendungsprotokolle von Amazon EC2. Eine vollständige Liste von Protokoll- und Überwachungslösungen finden Sie unter Anhang A: Cloud Capability-Definitionen – Protokollierung und Ereignisse in der Anleitung zur Reaktion auf AWS-Sicherheitsvorfälle.

- Untersuchen Sie die Protokollierungsmöglichkeiten für jede(n) AWS-Service und -Anwendung: Jede(r) AWS-Service und -Anwendung bietet Optionen für die Speicherung von Protokollen, jeweils mit eigenen Aufbewahrungs- und Lebenszyklus-Funktionen. Die beiden verbreitetsten Protokollspeicherservices sind Amazon Simple Storage Service (Amazon S3) und Amazon CloudWatch. Für lange Aufbewahrungszeiten wird die Verwendung von Amazon S3 empfohlen, wegen seiner Kosteneffektivität und der flexiblen Lebenszyklus-Funktionen. Wenn die primäre Protokollierungsoption Amazon CloudWatch-Protokolle sind, sollten Sie erwägen, weniger häufig benötigte Protokolle in Amazon S3 zu archivieren.
- Wählen Sie den Protokollspeicher: Die Wahl des Protokollspeichers hängt generell vom verwendeten Abfragetool, den Aufbewahrungsfunktionen, der Vertrautheit damit und den Kosten ab. Die wichtigsten Optionen für die Protokollspeicherung sind ein Amazon S3-Bucket oder eine CloudWatch-Protokollgruppe.

  Ein Amazon S3-Bucket bietet kosteneffektiven und dauerhaften Speicher mit optionaler Lebenszyklusrichtlinie. In Amazon S3-Buckets gespeicherte Protokolle können mit Services wie Amazon Athena abgefragt werden.

  Eine CloudWatch-Protokollgruppe bietet dauerhaften Speicher und eine integrierte Abfragemöglichkeit über CloudWatch Logs Insights.
- Legen Sie die benötigte Aufbewahrungszeit für Protokolle fest: Wenn Sie einen Amazon S3-Bucket oder eine CloudWatch-Protokollgruppe für die Speicherung von Protokollen verwenden, müssen Sie adäquate Lebenszyklen für jede Protokollquelle einrichten, um Speicher- und Abrufkosten zu optimieren. Normalerweise haben Kunden Protokolle zwischen drei Monaten bis einem Jahr für Abfragen verfügbar, bei einer Gesamtaufbewahrungszeit von bis zu sieben Jahren. Die Wahl von Verfügbarkeit und Aufbewahrungszeit sollte sich nach Ihren Sicherheitsanforderungen und einer Kombination aus gesetzlichen, regulatorischen und unternehmensinternen Vorschriften richten.
- Aktivieren Sie die Protokollierung für jede(n) AWS-Service und -Anwendung mit korrekten Aufbewahrungs- und Lebenszyklusrichtlinien: Suchen Sie für jeden AWS-Service oder jede AWS-Anwendung in Ihrer Organisation nach den entsprechenden Anleitungen zur Protokollkonfiguration:
  - Konfigurieren eines AWS CloudTrail-Trails
  - Konfigurieren von VPC-Flow-Protokollen
  - Konfigurieren des Amazon GuardDuty-Ergebnisexports
  - Konfigurieren der AWS Config-Aufzeichnung
  - Konfigurieren des Web-ACL-Datenverkehrs von AWS WAF
  - Konfigurieren der Netzwerkdatenverkehrsprotokolle von AWS Network Firewall

  - Konfigurieren der Zugriffsprotokolle von Elastic Load Balancing
  - Konfigurieren von Resolver-Query-Protokollen von Amazon Route 53
  - Konfigurieren von Amazon RDS-Protokollen
  - Konfigurieren von Amazon EKS-Steuerebenenprotokollen
  - Konfigurieren eines Amazon CloudWatch-Agenten für Amazon EC2-Instances und On-Premises-Server
- Wählen und implementieren Sie Abfragemechanismen für Ihre Protokolle: Für Protokollabfragen können Sie CloudWatch Logs Insights für in CloudWatch-Protokollgruppen gespeicherte Daten sowie Amazon Athena und Amazon OpenSearch Service für in Amazon S3 gespeicherte Daten verwenden. Sie können auch Abfragetools von Drittanbietern wie etwa den SIEM (Security Information and Event Management)-Service verwenden.

  Bei der Auswahl eines Tools zur Protokollabfrage sollten Sie die Personen, die Prozesse und die Technologieaspekte Ihrer Sicherheitsoperationen berücksichtigen. Wählen Sie ein Tool, das betriebliche, geschäftliche und sicherheitsrelevante Aspekte berücksichtigt und langfristig sowohl zugänglich als auch wartbar ist. Denken Sie daran, dass Tools zur Protokollabfrage optimal funktionieren, wenn die Anzahl der zu durchsuchenden Protokolle im Rahmen der Limits des jeweiligen Tools liegt. Es ist nicht ungewöhnlich, aus Kostengründen oder aufgrund technischer Einschränkungen mehrere Abfragetools zu verwenden.

  Beispielsweise können Sie ein SIEM-Tool eines Drittanbieters für Abfragen der letzten 90 Datentage, aber aufgrund der Protokollerfassungskosten für SIEM Athena für Abfragen verwenden, die darüber hinaus gehen. Prüfen Sie unabhängig von der Implementierung, ob Ihr Konzept die Anzahl der für die Maximierung der operationalen Effizienz erforderlichen Tools minimiert, besonders für Untersuchungen von Sicherheitsvorfällen.
- Verwenden Sie Protokolle für Benachrichtigungen: AWS bietet verschiedene Benachrichtigungsmöglichkeiten über mehrere Sicherheitsservices:
  - AWS Config überwacht und zeichnet Ihre AWS-Ressourcenkonfigurationen auf. Darüber hinaus ermöglicht es Ihnen, die Auswertung und Korrektur der gewünschten Konfigurationen zu automatisieren.
  - Amazon GuardDuty ist ein Bedrohungserkennungsservice, der kontinuierlich nach schädlichen Aktivitäten und nicht autorisierten Verhaltensweisen sucht, um Ihr AWS-Konten und Ihre Workloads zu schützen. GuardDuty erfasst, aggregiert und analysiert Informationen aus Quellen wie AWS CloudTrail-Verwaltungs- und Datenereignissen, DNS-Protokollen, VPC-Flow-Protokollen und Amazon EKS-Prüfprotokollen. GuardDuty ruft unabhängige Datenströme direkt

von CloudTrail, VPC-Flow-Protokollen, DNS-Abfrageprotokollen und Amazon EKS ab. Sie müssen keine Amazon S3-Bucket-Richtlinien verwalten oder die Art und Weise der Erfassung und Speicherung von Protokollen verändern. Es wird jedoch empfohlen, diese Protokolle für Ihre eigenen Untersuchungs- und Compliance-Zwecke aufzubewahren.

- AWS Security Hub bietet einen zentralen Ort, an dem Ihre Sicherheitswarnungen oder Ergebnisse von mehreren AWS-Services und optionalen Produkten von Drittanbietern aggregiert, organisiert und priorisiert werden. So erhalten Sie einen umfassenden Überblick über Sicherheitswarnungen und den Compliance-Status.

Sie können auch benutzerdefinierte Alarm-Engines für Sicherheitsalarme verwenden, die von diesen Services nicht abgedeckt werden, bzw. für bestimmte Alarme, die für Ihre Umgebung relevante sind. Für Informationen zur Erstellung dieser Alarm- und Erkennungsmechanismen vgl. Erkennung in der AWS-Sicherheits- und Vorfallreaktionsanleitung.

Ressourcen

Zugehörige bewährte Methoden:

- SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten
- SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements
- SEC10-BP06 Vorabbereitstellen von Tools

Zugehörige Dokumente:

- AWS-Sicherheits- und Vorfallreaktionsanleitung
- Erste Schritte mit Amazon Security Lake
- Erste Schritte: Amazon CloudWatch Logs
- Security Partner Solutions: Logging and Monitoring (Partnerlösungen im Bereich Sicherheit: Protokollierung und Überwachung)

Zugehörige Videos:

- AWS re:Invent 2022 - Introducing Amazon Security Lake (AWS re:Invent 2022 – Vorstellung von Amazon Security Lake)

Zugehörige Beispiele:

- Assisted Log Enabler für AWS
- Historischer Export von Ergebnissen von AWS Security Hub

Zugehörige Tools:

- Snowflake for Cybersecurity

SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten

Sicherheitsteams stützen sich auf Protokolle und Erkenntnisse, um Ereignisse zu analysieren, die auf unbefugte Aktivitäten oder unbeabsichtigte Änderungen hindeuten könnten. Um diese Analyse zu rationalisieren, sollten Sie Sicherheitsprotokolle und Ergebnisse an standardisierten Orten erfassen. Dies macht Datenpunkte von Interesse für die Korrelation verfügbar und kann die Integration von Tools vereinfachen.

Gewünschtes Ergebnis: Sie verfügen über einen standardisierten Ansatz zum Sammeln, Analysieren und Visualisieren von Protokolldaten, Erkenntnissen und Metriken. Sicherheitsteams können Sicherheitsdaten über verschiedene Systeme hinweg effizient korrelieren, analysieren und visualisieren, um potenzielle Sicherheitsereignisse zu erkennen und Anomalien zu identifizieren. Systeme für Sicherheitsinformation und Ereignisverwaltung (Security Information and Event Management, SIEM) oder andere Mechanismen sind integriert, um Protokolldaten abzufragen und zu analysieren, damit Sie zeitnah auf Sicherheitsereignisse reagieren, diese verfolgen und eskalieren können.

Typische Anti-Muster:

- Teams besitzen und verwalten eigenständig Protokolle und Metriksammlungen, die nicht mit der Protokollierungsstrategie der Organisation übereinstimmen.
- Teams verfügen nicht über angemessene Zugriffskontrollen, um die Sichtbarkeit und Veränderung der erfassten Daten einzuschränken.
- Teams regeln ihre Sicherheitsprotokolle, Erkenntnisse und Metriken nicht als Teil ihrer Richtlinie zur Datenklassifizierung.
- Teams vernachlässigen bei der Konfiguration von Datensammlungen die Anforderungen an die Datenhoheit und die Lokalisierung.

Vorteile der Einführung dieser bewährten Methode: Eine standardisierte Protokollierungslösung zur Erfassung und Abfrage von Protokolldaten und -ereignissen verbessert die aus den darin enthaltenen

Informationen gewonnenen Erkenntnisse. Die Konfiguration eines automatisierten Lebenszyklus für die gesammelten Protokolldaten kann die durch die Speicherung von Protokollen entstehenden Kosten reduzieren. Sie können eine fein abgestufte Zugriffskontrolle für die gesammelten Protokollinformationen einrichten, je nachdem, wie sensibel die Daten sind und welche Zugriffsmuster Ihre Teams benötigen. Sie können Tools integrieren, um die Daten zu korrelieren, zu visualisieren und Erkenntnisse daraus abzuleiten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die zunehmende AWS-Nutzung innerhalb einer Organisation führt zu einer wachsenden Anzahl von verteilten Workloads und Umgebungen. Jeder dieser Workloads und jede dieser Umgebungen generiert Daten über die darin stattfindenden Aktivitäten. Die Erfassung und lokale Speicherung dieser Daten stellt eine Herausforderung für den Sicherheitsbetrieb dar. Sicherheitsteams verwenden Tools wie Sicherheitsinformations- und Ereignisverwaltungssysteme (SIEM), um Daten aus verteilten Quellen zu sammeln und Korrelations-, Analyse- und Reaktionsabläufe durchzuführen. Dies erfordert die Verwaltung einer komplexen Reihe von Berechtigungen für den Zugriff auf die verschiedenen Datenquellen und einen zusätzlichen Aufwand beim Betrieb der Extract, Transform, Load (ETL)-Prozesse.

Um diese Herausforderungen zu meistern, sollten Sie alle relevanten Quellen von Sicherheitsprotokolldaten in einem Protokollarchiv-Konto zusammenfassen. Dies ist beschrieben in: Organizing Your AWS Environment Using Multiple Accounts. Dazu gehören alle sicherheitsrelevanten Daten aus Ihrem Workload und Protokolle, die AWS-Services erzeugen, wie AWS CloudTrail, AWS WAF, Elastic Load Balancing und Amazon Route 53. Es hat mehrere Vorteile, diese Daten an standardisierten Orten in einem separaten AWS-Konto mit entsprechenden kontoübergreifenden Berechtigungen zu erfassen. Diese Vorgehensweise hilft, die Manipulation von Protokollen in gefährdeten Workloads und Umgebungen zu verhindern, bietet einen einzigen Integrationspunkt für zusätzliche Tools und bietet ein einfacheres Modell für die Konfiguration der Datenaufbewahrung und des Lebenszyklus. Bewerten Sie die Auswirkungen der Datenhoheit, der Compliance-Bereiche und anderer Vorschriften, um festzustellen, ob mehrere Speicherorte für Sicherheitsdaten und Aufbewahrungsfristen erforderlich sind.

Um die Erfassung und Standardisierung von Protokollen und Erkenntnissen zu erleichtern, bewerten Sie Amazon Security Lake in Ihrem Protokollarchiv-Konto. Sie können Security Lake so konfigurieren, dass Daten aus gängigen Quellen wie CloudTrail, Route 53, Amazon EKS und VPC Flow Logs automatisch aufgenommen werden. Außerdem können Sie AWS Security Hub auch als Datenquelle in Security Lake konfigurieren, sodass Sie Erkenntnisse aus anderen AWS-Services wie Amazon

GuardDuty und Amazon Inspector mit Ihren Protokolldaten korrelieren können. Ferner haben Sie die Möglichkeit, Datenquellen von Drittanbietern zu integrieren oder eigene Datenquellen zu konfigurieren. Alle Integrationen standardisieren Ihre Daten in das Open Cybersecurity Schema Framework (OCSF)-Format und werden in Amazon S3-Buckets als Parquet-Dateien gespeichert, sodass keine ETL-Verarbeitung erforderlich ist.

Die Speicherung von Sicherheitsdaten an standardisierten Orten bietet erweiterte Analysemöglichkeiten. AWS empfiehlt Ihnen die Bereitstellung von Tools für Sicherheitsanalysen, die in einer AWS-Umgebung arbeiten, in einem Security-Tooling-Konto, das von Ihrem Protokollarchiv-Konto getrennt ist. Dieser Ansatz ermöglicht es Ihnen, Kontrollen in der Tiefe zu implementieren, um die Integrität und Verfügbarkeit der Protokolle und des Protokollverwaltungsprozesses zu schützen, und zwar unabhängig von den Tools, die auf sie zugreifen. Erwägen Sie die Nutzung von Services wie Amazon Athena, um On-Demand-Abfragen durchzuführen, die mehrere Datenquellen miteinander in Beziehung setzen. Sie können auch Visualisierungstools wie QuickSight integrieren. KI-gestützte Lösungen werden zunehmend verfügbar und können Funktionen wie die Übersetzung von Erkenntnissen in für Menschen lesbare Zusammenfassungen und Interaktion in natürlicher Sprache übernehmen. Diese Lösungen lassen sich oft leichter integrieren, wenn ein standardisierter Datenspeicher für Abfragen zur Verfügung steht.

Implementierungsschritte

1. Erstellen Sie die Konten „Protokollarchiv“ und „Security Tooling“
   a. Erstellen Sie mit AWS Organizations die Konten „Protokollarchiv“ und „Security Tooling“ unter einer Sicherheitsorganisationseinheit. Wenn Sie AWS Control Tower zur Verwaltung Ihrer Organisation verwenden, werden die Konten für Protokollarchiv und Security Tooling automatisch für Sie erstellt. Konfigurieren Sie bei Bedarf Rollen und Berechtigungen für den Zugriff auf diese Konten und deren Verwaltung.
2. Konfigurieren Sie Ihre standardisierten Speicherorte für Sicherheitsdaten
   a. Legen Sie Ihre Strategie für die Erstellung standardisierter Sicherheitsdatenorte fest. Sie können dies durch Optionen wie allgemeine Data-Lake-Architekturansätze, Datenprodukte von Drittanbietern oder Amazon Security Lake erreichen. AWS empfiehlt, dass Sie Sicherheitsdaten von AWS-Regionen erfassen, die für Ihre Konten aktiviert sind, auch wenn sie nicht aktiv genutzt werden.
3. Konfigurieren Sie die Veröffentlichung von Datenquellen an Ihren standardisierten Standorten
   a. Identifizieren Sie die Quellen für Ihre Sicherheitsdaten und konfigurieren Sie sie so, dass sie an Ihren standardisierten Standorten veröffentlicht werden. Evaluieren Sie Optionen für den automatischen Export von Daten in das gewünschte Format im Gegensatz zu solchen, bei

denen ETL-Prozesse entwickelt werden müssen. Mit Amazon Security Lake können Sie Daten aus unterstützten AWS-Quellen und integrierten Drittsystemen sammeln.

4. Konfigurieren Sie Tools für den Zugriff auf Ihre standardisierten Speicherorte
   a. Konfigurieren Sie Tools wie Amazon Athena, QuickSight oder Lösungen von Drittanbietern, um den erforderlichen Zugriff auf Ihre standardisierten Standorte zu erhalten. Konfigurieren Sie diese Tools so, dass sie über das Security Tooling-Konto mit kontoübergreifendem Zugriff auf das Protokollarchiv-Konto arbeiten, sofern zutreffend. Erstellen Sie Subscriber in Amazon Security Lake, um diesen Tools Zugriff auf Ihre Daten zu geben.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP01 Trennen von Workloads mithilfe von Konten
- SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements
- SEC08-BP04 Durchsetzen der Zugriffskontrolle
- OPS08-BP02 Analysieren von Workload-Protokollen

Zugehörige Dokumente:

- AWS Whitepapers: Organizing Your AWS Environment Using Multiple Accounts
- AWS Prescriptive Guidance: AWS Security Reference Architecture (AWS SRA)
- AWS Prescriptive Guidance: Logging and monitoring guide for application owners

Zugehörige Beispiele:

- Aggregating, searching, and visualizing log data from distributed sources with Amazon Athena and QuickSight
- How to visualize Amazon Security Lake findings with QuickSight
- Generate AI powered insights for Amazon Security Lake using Amazon SageMaker AI Studio and Amazon Bedrock
- Identify cybersecurity anomalies in your Amazon Security Lake data using Amazon SageMaker AI
- Ingest, transform, and deliver events published by Amazon Security Lake to Amazon OpenSearch Service
- How to use AWS Security Hub and Amazon OpenSearch Service for SIEM

Zugehörige Tools:

- Amazon Security Lake
- Amazon Security Lake-Partnerintegrationen
- Open Cybersecurity Schema Framework (OCSF)
- Amazon Athena
- QuickSight
- Amazon Bedrock

SEC04-BP03 Korrelieren und Anreichern von Sicherheitswarnmeldungen

Unerwartete Aktivitäten können mehrere Sicherheitswarnmeldungen aus verschiedenen Quellen auslösen, die eine weitere Korrelation und Anreicherung erfordern, um den gesamten Kontext zu verstehen. Implementieren Sie die automatische Korrelation und Anreicherung von Sicherheitswarnmeldungen, um eine genauere Identifizierung von Vorfällen und eine bessere Reaktion darauf zu ermöglichen.

Gewünschtes Ergebnis: Da die Aktivitäten in Ihren Workloads und Umgebungen unterschiedliche Warnmeldungen erzeugen, korrelieren automatische Mechanismen die Daten und reichern sie mit zusätzlichen Informationen an. Diese Vorverarbeitung ermöglicht ein detaillierteres Verständnis des Ereignisses, was Ihren Ermittlern hilft, die Kritikalität des Ereignisses zu bestimmen und festzustellen, ob es sich um einen Vorfall handelt, der eine formelle Reaktion erfordert. Dieses Verfahren entlastet Ihre Überwachungs- und Untersuchungsteams.

Typische Anti-Muster:

- Verschiedene Personengruppen untersuchen Erkenntnisse und Warnmeldungen, die von verschiedenen Systemen generiert werden, sofern nicht durch Anforderungen der Aufgabentrennung etwas anderes vorgeschrieben ist.
- Ihre Organisation leitet alle Sicherheitserkenntnisse und -warnmeldungen an Standardspeicherorte weiter, verlangt aber von den Ermittlern, dass sie diese manuell korrelieren und anreichern.
- Sie verlassen sich ausschließlich auf die Intelligenz von Bedrohungserkennungssystemen, um über Erkenntnisse zu berichten und die Kritikalität zu bestimmen.

Vorteile der Einführung dieser bewährten Methode: Die automatische Korrelation und Anreicherung von Warnmeldungen trägt dazu bei, die gesamte kognitive Belastung und die manuelle

Datenaufbereitung zu reduzieren, die Ihre Ermittler benötigen. Diese Methode kann die Zeit verkürzen, die benötigt wird, um festzustellen, ob es sich bei dem Ereignis um einen Vorfall handelt, und eine formelle Reaktion einzuleiten. Zusätzlicher Kontext hilft Ihnen auch, den wahren Schweregrad eines Ereignisses genau zu bewerten, da er höher oder niedriger sein kann, als eine einzelne Warnmeldung vermuten lässt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Sicherheitswarnmeldungen können von vielen verschiedenen Quellen innerhalb von AWS stammen, darunter:

- Services wie Amazon GuardDuty, AWS Security Hub, Amazon Macie, Amazon Inspector, AWS Config, AWS Identity and Access Management Access Analyzer und Network Access Analyzer
- Warnmeldungen aus der automatisierten Analyse von AWS-Service-, Infrastruktur- und Anwendungsprotokollen, z. B. von Security Analytics for Amazon OpenSearch Service.
- Warnungen als Reaktion auf Änderungen in Ihrer Abrechnungsaktivität aus Quellen wie Amazon CloudWatch, Amazon EventBridge oder AWS Budgets.
- Quellen von Drittanbietern wie Threat Intelligence Feeds und Security Partner Solutions vom AWS Partner Network
- Kontakt durch AWS-Vertrauen und -Sicherheit oder andere Quellen, wie Kunden oder interne Mitarbeiter.

In ihrer grundlegendsten Form enthalten Warnmeldungen Informationen darüber, wer (Prinzipal oder Identität) was (Aktion, die ergriffen wird) im Hinblick auf (Ressourcen, die betroffen sind) macht. Ermitteln Sie für jede dieser Quellen, ob es Möglichkeiten gibt, Zuordnungen zwischen den Identifikatoren für diese Identitäten, Aktionen und Ressourcen als Grundlage für die Durchführung von Korrelationen zu erstellen. Dies kann in Form einer Integration von Quellen für Warnmeldungen mit einem SIEM-Tool (Security Information and Event Management) erfolgen, das eine automatische Korrelation für Sie durchführt, oder durch den Aufbau eigener Datenpipelines und -verarbeitung oder durch eine Kombination aus beidem.

Ein Beispiel für einen Dienst, der eine Korrelation für Sie durchführen kann, ist Amazon Detective. Detective nimmt laufend Warnmeldungen aus verschiedenen AWS- und Drittquellen auf und nutzt verschiedene Formen von Informationen, um eine visuelle Grafik ihrer Beziehungen zur Unterstützung von Ermittlungen zusammenzustellen.

Während die anfängliche Kritikalität eines Alarms eine Hilfe für die Priorisierung ist, bestimmt der Kontext, in dem der Alarm auftrat, seine wahre Kritikalität. Zum Beispiel kann Amazon GuardDuty eine Warnmeldung ausgeben, dass eine Amazon EC2-Instance innerhalb Ihres Workloads einen unerwarteten Domain-Namen abfragt. GuardDuty könnte dieser Warnmeldung von sich aus eine niedrige Kritikalität zuweisen. Eine automatische Korrelation mit anderen Aktivitäten zum Zeitpunkt der Warnmeldung könnte jedoch aufdecken, dass mehrere hundert EC2-Instances von derselben Identität bereitgestellt wurden, was die Gesamtbetriebskosten erhöht. In diesem Fall könnte GuardDuty diesen korrelierten Ereigniskontext als neue Sicherheitswarnung veröffentlichen und die Kritikalität auf hoch setzen, was die weiteren Maßnahmen beschleunigen würde.

Implementierungsschritte

1. Identifizieren Sie Quellen für Informationen zu Sicherheitswarnmeldungen. Verstehen Sie, wie Warnmeldungen aus diesen Systemen Identität, Aktion und Ressourcen darstellen, um festzustellen, wo eine Korrelation möglich ist.
2. Richten Sie einen Mechanismus zur Erfassung von Warnmeldungen aus verschiedenen Quellen ein. Ziehen Sie zu diesem Zweck Services wie Security Hub, EventBridge und CloudWatch in Betracht.
3. Identifizieren Sie Quellen für die Korrelation und Anreicherung von Daten. Beispiele für Quellen sind CloudTrail, VPC-Flow-Protokolle, Amazon Security Lake sowie Infrastruktur- und Anwendungsprotokolle.
4. Integrieren Sie Ihre Warnmeldungen mit Ihren Datenkorrelations- und -anreicherungsquellen, um detailliertere Kontexte für Sicherheitsereignisse zu erstellen und die Kritikalität zu ermitteln.
   a. Amazon Detective, SIEM-Tools oder andere Lösungen von Drittanbietern können ein gewisses Maß an Erfassung, Korrelation und Anreicherung automatisch durchführen.
   b. Sie können auch AWS-Services nutzen, um Ihre eigenen zu erstellen. Sie können zum Beispiel eine Funktion AWS Lambda aufrufen, um eine Amazon Athena-Abfrage von AWS CloudTrail oder Amazon Security Lake auszuführen, und die Ergebnisse in EventBridge veröffentlichen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC10-BP03 Vorbereiten forensischer Funktionen
- OPS08-BP04 Erstellen umsetzbarer Warnmeldungen
- REL06-BP03 Senden von Benachrichtigungen (Verarbeitung und Benachrichtigung in Echtzeit)

Zugehörige Dokumente:

- AWS-Leitfaden für Security Incident Response

Zugehörige Beispiele:

- How to enrich AWS Security Hub findings with account metadata
- How to use AWS Security Hub and Amazon OpenSearch Service for SIEM

Zugehörige Tools:

- Amazon Detective
- Amazon EventBridge
- AWS Lambda
- Amazon Athena

SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen

Ihre detektivischen Kontrollen können Sie auf Ressourcen aufmerksam machen, die nicht mit Ihren Konfigurationsanforderungen übereinstimmen. Sie können programmatisch definierte Abhilfemaßnahmen einleiten, entweder manuell oder automatisch, um diese Ressourcen zu korrigieren und mögliche Auswirkungen zu minimieren. Wenn Sie Abhilfemaßnahmen programmatisch definieren, können Sie sofort und konsequent handeln.

Automatisierung kann zwar den Sicherheitsbetrieb verbessern, aber Sie sollten die Automatisierung sorgfältig implementieren und verwalten. Schaffen Sie geeignete Überwachungs- und Kontrollmechanismen, um zu überprüfen, ob die automatisierten Antworten effektiv und genau sind und mit den Organisationsrichtlinien und der Risikobereitschaft übereinstimmen.

Gewünschtes Ergebnis: Sie definieren Standards für die Ressourcenkonfiguration und die Schritte zur Behebung, wenn festgestellt wird, dass die Ressourcen nicht konform sind. Wo immer möglich, haben Sie Abhilfemaßnahmen programmatisch definiert, sodass sie entweder manuell oder durch Automatisierung eingeleitet werden können. Es gibt Erkennungssysteme, die nicht konforme Ressourcen identifizieren und Warnungen in zentralisierten Tools veröffentlichen, die von Ihrem Sicherheitspersonal überwacht werden. Diese Tools unterstützen die Durchführung Ihrer programmatischen Korrekturen, entweder manuell oder automatisch. Automatische

Abhilfemaßnahmen verfügen über angemessene Überwachungs- und Kontrollmechanismen, um ihre Verwendung zu steuern.

Typische Anti-Muster:

- Sie implementieren Automatisierung, versäumen es aber, Abhilfemaßnahmen gründlich zu testen und zu validieren. Dies kann unbeabsichtigte Folgen haben, wie z. B. die Unterbrechung legitimer Geschäftsabläufe oder die Instabilität des Systems.
- Sie verbessern die Reaktionszeiten und Verfahren durch Automatisierung, aber ohne angemessene Überwachung und Mechanismen, die bei Bedarf menschliches Eingreifen und Urteilsvermögen ermöglichen.
- Sie verlassen sich ausschließlich auf Abhilfemaßnahmen, anstatt Abhilfemaßnahmen als Teil eines umfassenderen Programms zur Reaktion auf Vorfälle und zur Wiederherstellung zu nutzen.

Vorteile der Einführung dieser bewährten Methode: Automatische Abhilfemaßnahmen können schneller auf Fehlkonfigurationen reagieren als manuelle Prozesse. So können Sie potenzielle Auswirkungen auf Ihr Unternehmen minimieren und das Zeitfenster für unbeabsichtigte Nutzungen verringern. Wenn Sie Abhilfemaßnahmen programmatisch definieren, werden sie konsistent angewendet, was das Risiko menschlicher Fehler verringert. Die Automatisierung kann auch eine größere Anzahl von Alarmen gleichzeitig verarbeiten, was besonders in Umgebungen von großem Maßstab wichtig ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wie unter SEC01-BP03 Identifizieren und Validieren von Kontrollzielen beschrieben, können Services wie AWS Config Ihnen dabei helfen, die Konfiguration der Ressourcen in Ihren Konten auf die Einhaltung Ihrer Anforderungen hin zu überwachen. Wenn nicht konforme Ressourcen entdeckt werden, empfehlen wir Ihnen, den Versand von Warnmeldungen an eine Cloud Security Posture Management (CSPM)-Lösung wie AWS Security Hub zu konfigurieren, um bei der Abhilfe zu unterstützen. Diese Lösungen bieten einen zentralen Ort für Ihre Sicherheitsbeauftragten, um Probleme zu überwachen und Korrekturmaßnahmen zu ergreifen.

Einige Situationen, in denen Ressourcen nicht konform sind, können zwar einzigartig sein und erfordern menschliches Urteilsvermögen, um Abhilfe zu schaffen. Für andere Fälle gibt es jedoch eine Standardreaktion, die Sie programmatisch definieren können. Eine Standardreaktion auf eine falsch konfigurierte VPC-Sicherheitsgruppe könnte zum Beispiel darin bestehen, die unzulässigen

Regeln zu entfernen und den Eigentümer zu benachrichtigen. Antworten können in AWS Lambda-Funktionen, in AWS-Systems Manager-Automation-Dokumenten oder durch andere von Ihnen bevorzugte Code-Umgebungen definiert werden. Vergewissern Sie sich, dass die Umgebung in der Lage ist, sich bei AWS zu authentifizieren, indem Sie eine IAM-Rolle mit der geringsten Berechtigung verwenden, die für die Durchführung von Korrekturmaßnahmen erforderlich ist.

Sobald Sie die gewünschte Abhilfemaßnahme definiert haben, können Sie festlegen, wie Sie diese einleiten möchten. AWS Config kann Abhilfemaßnahmen für Sie einleiten. Wenn Sie Security Hub verwenden, können Sie dies über Angepasste Aktionen tun, wodurch die Suchinformationen in Amazon EventBridge veröffentlicht werden. Eine EventBridge-Regel kann dann Ihre Abhilfe einleiten. Sie können die benutzerdefinierte Aktion in Security Hub so konfigurieren, dass sie entweder automatisch oder manuell ausgeführt wird.

Für programmatische Abhilfemaßnahmen empfehlen wir Ihnen, umfassende Protokolle und Audits für die durchgeführten Maßnahmen sowie deren Ergebnisse zu führen. Prüfen und analysieren Sie diese Protokolle, um die Effektivität der automatisierten Prozesse zu bewerten und Verbesserungsmöglichkeiten zu identifizieren. Erfassen Sie Protokolle in Amazon CloudWatch Logs und Abhilfeergebnisse als Erkenntnis in Security Hub.

Als Ausgangspunkt können Sie Automatische Sicherheitsreaktion in AWS verwenden, das über vorgefertigte Abhilfemaßnahmen zur Behebung häufiger Sicherheitsfehlkonfigurationen verfügt.

Implementierungsschritte

1. Analysieren und priorisieren Sie Warnmeldungen.
   a. Konsolidieren Sie Sicherheitswarnungen von verschiedenen AWS-Services in Security Hub für eine zentrale Übersicht, Priorisierung und Abhilfe.
2. Entwickeln Sie Abhilfemaßnahmen.
   a. Verwenden Sie Services wie Systems Manager und AWS Lambda, um programmatische Korrekturen durchzuführen.
3. Konfigurieren Sie, wie Abhilfemaßnahmen eingeleitet werden.
   a. Definieren Sie mithilfe von Systems Manager benutzerdefinierte Aktionen, die Erkenntnisse an EventBridge veröffentlichen. Konfigurieren Sie diese Aktionen so, dass sie manuell oder automatisch ausgelöst werden.
   b. Sie können auch Amazon Simple Notification Service (SNS) verwenden, um Benachrichtigungen und Warnmeldungen an relevante Beteiligte (wie das Sicherheitsteam

oder das Vorfallsreaktionsteam) zu senden, damit diese bei Bedarf manuell eingreifen oder eskalieren können.

4. Prüfen und analysieren Sie die Protokolle der Abhilfemaßnahmen auf Wirksamkeit und Verbesserung.
   a. Senden Sie die Protokollausgabe an CloudWatch Logs. Erfassen Sie die Ergebnisse als Erkenntnis in Security Hub.

Ressourcen

Zugehörige bewährte Methoden:

- SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs

Zugehörige Dokumente:

- AWS Security Incident Response Guide – Detection

Zugehörige Beispiele:

- Automatische Sicherheitsreaktion in AWS
- Monitor EC2 instance key pairs using AWS Config
- Create AWS Config custom rules by using AWS CloudFormation Guard policies
- Automatically remediate unencrypted Amazon RDS DB instances and clusters

Zugehörige Tools:

- AWS Systems Manager Automation
- Automatische Sicherheitsreaktion in AWS

Schutz der Infrastruktur

Fragen

- SEC 5. Wie schützen Sie Ihre Netzwerkressourcen?
- SEC 6. Wie schützen Sie Ihre Datenverarbeitungsressourcen?

SEC 5. Wie schützen Sie Ihre Netzwerkressourcen?

Alle Workloads, die über eine Art Netzwerkverbindung verfügen, unabhängig davon, ob es sich um das Internet oder ein privates Netzwerk handelt, erfordern mehrere Abwehrebenen, um Schutz vor externen und internen Netzwerkbedrohungen sicherzustellen.

Bewährte Methoden

- SEC05-BP01 Erstellen von Netzwerkebenen
- SEC05-BP02 Kontrollieren des Datenverkehrsflusses innerhalb Ihrer Netzwerkebenen
- SEC05-BP03 Implementieren eines prüfungsbasierten Schutzes
- SEC05-BP04 Automatisieren des Netzwerkschutzes

SEC05-BP01 Erstellen von Netzwerkebenen

Segmentieren Sie Ihre Netzwerktopologie in verschiedene Ebenen, die auf logischen Gruppierungen Ihrer Workload-Komponenten entsprechend ihrer Datensensibilität und Zugriffsanforderungen basieren. Unterscheiden Sie zwischen Komponenten, auf die vom Internet aus zugegriffen werden muss, wie z. B. öffentliche Web-Endpunkte, und solchen, die nur intern erreichbar sein müssen, wie z. B. Datenbanken.

Gewünschtes Ergebnis: Die Ebenen Ihres Netzwerks sind Teil eines ganzheitlichen, tiefgreifenden Sicherheitsansatzes, der die Identitätsauthentifizierungs- und Autorisierungsstrategie Ihrer Workloads ergänzt. Je nach Sensibilität der Daten und den Zugriffsanforderungen werden Ebenen mit entsprechenden Verkehrsfluss- und Kontrollmechanismen eingerichtet.

Typische Anti-Muster:

- Sie erstellen alle Ressourcen in einem einzigen VPC oder Subnetz.
- Sie erstellen Ihre Netzwerkebenen ohne Rücksicht auf die Anforderungen an die Datensensibilität, das Verhalten der Komponenten oder die Funktionalität.
- Sie verwenden VPCs und Subnetze als Standards für alle Aspekte der Netzwerkebenen und berücksichtigen nicht, wie verwaltete AWS-Services Ihre Topologie beeinflussen.

Vorteile der Einführung dieser bewährten Methode: Die Einrichtung von Netzwerkebenen ist der erste Schritt, um unnötige Pfade durch das Netzwerk einzuschränken, insbesondere solche, die zu kritischen Systemen und Daten führen. Dadurch wird es für Unbefugte schwieriger, sich

Zugriff auf Ihr Netzwerk zu verschaffen und zu weiteren Ressourcen darin zu navigieren. Diskrete Netzwerkebenen reduzieren den Umfang der Analyse für Inspektionssysteme, z. B. für die Erkennung von Eindringlingen oder die Verhinderung von Malware, vorteilhaft. Dadurch wird das Potenzial für Fehlalarme und unnötigen Verarbeitungsaufwand reduziert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Beim Entwurf einer Workload-Architektur ist es üblich, die Komponenten je nach ihrer Verantwortlichkeit in verschiedene Ebenen aufzuteilen. Eine Webanwendung kann zum Beispiel eine Präsentationsebene, eine Anwendungsebene und eine Datenebene haben. Bei der Gestaltung Ihrer Netzwerktopologie können Sie einen ähnlichen Ansatz wählen. Die zugrunde liegenden Netzwerkkontrollen können dazu beitragen, die Anforderungen Ihres Workloads an den Datenzugriff durchzusetzen. In einer dreistufigen Webanwendungsarchitektur können Sie zum Beispiel Ihre statischen Präsentationsebenendateien in Amazon S3 speichern und sie von einem Content Delivery Network (CDN) wie Amazon CloudFront aus bereitstellen. Die Anwendungsebene kann öffentliche Endpunkte haben, die ein Application Load Balancer (ALB) in einem Amazon VPC-öffentlichen Subnetz (ähnlich einer demilitarisierten Zone oder DMZ) bedient, während die Backend-Services in privaten Subnetzen bereitgestellt werden. Die Datenebene, die Ressourcen wie Datenbanken und gemeinsam genutzte Dateisysteme hostet, kann sich in anderen privaten Subnetzen befinden als die Ressourcen Ihrer Anwendungsebene. An jeder dieser Ebenengrenzen (CDN, öffentliches Subnetz, privates Subnetz) können Sie Kontrollen bereitstellen, die es nur autorisiertem Datenverkehr erlauben, diese Grenzen zu überqueren.

Ähnlich wie bei der Modellierung von Netzwerkebenen auf der Grundlage des funktionalen Zwecks der Komponenten Ihres Workloads sollten Sie auch die Sensibilität der verarbeiteten Daten berücksichtigen. Wenn Sie das Beispiel der Webanwendung verwenden, kann es sein, dass alle Ihre Workload-Services innerhalb der Anwendungsebene angesiedelt sind, während verschiedene Services Daten mit unterschiedlichen Sensibilitätsstufen verarbeiten. In diesem Fall kann die Aufteilung der Anwendungsebene durch mehrere private Subnetze, verschiedene VPCs in demselben AWS-Konto oder sogar verschiedene VPCs in verschiedenen AWS-Konten für jede Stufe der Datensensibilität je nach Ihren Kontrollanforderungen angemessen sein.

Eine weitere Überlegung für Netzwerkebenen ist die Verhaltenskonsistenz der Komponenten Ihres Workloads. Um das Beispiel fortzusetzen: In der Anwendungsebene haben Sie möglicherweise Services, die Eingaben von Endbenutzern oder externen Systemintegrationen akzeptieren, die von Natur aus risikoreicher sind als die Eingaben für andere Services. Beispiele sind das Hochladen von Dateien, das Ausführen von Skripten, das Scannen von E-Mails und so weiter. Die Unterbringung

dieser Services in einer eigenen Netzwerkebene hilft dabei, eine stärkere Isolationsgrenze um sie herum zu schaffen, und kann verhindern, dass ihr einzigartiges Verhalten falsche positive Alarme in Inspektionssystemen erzeugt.

Berücksichtigen Sie bei Ihrer Planung, wie die Nutzung von AWS verwalteten Services Ihre Netzwerktopologie beeinflusst. Erfahren Sie, wie Services wie Amazon VPC Lattice die Interoperabilität Ihrer Workload-Komponenten über Netzwerkebenen hinweg erleichtern können. Wenn Sie AWS Lambda verwenden, sollten Sie die Bereitstellung in Ihren VPC-Subnetzen vornehmen, es sei denn, es gibt besondere Gründe, die dagegen sprechen. Bestimmen Sie, wo VPC-Endpunkte und AWS PrivateLink die Einhaltung von Sicherheitsrichtlinien, die den Zugriff auf Internet-Gateways beschränken, vereinfachen können.

Implementierungsschritte

1. Überprüfen Sie Ihre Workload-Architektur. Gruppieren Sie Komponenten und Services logisch nach den Funktionen, die sie erfüllen, nach der Sensibilität der verarbeiteten Daten und nach ihrem Verhalten.
2. Für Komponenten, die auf Anfragen aus dem Internet reagieren, sollten Sie Load Balancer oder andere Proxys verwenden, um öffentliche Endpunkte bereitzustellen. Erkunden Sie die Verlagerung der Sicherheitskontrollen durch den Einsatz von verwalteten Services wie CloudFront, Amazon API Gateway, Elastic Load Balancing und AWS Amplify zum Hosten öffentlicher Endpunkte.
3. Für Komponenten, die in Datenverarbeitungsumgebungen ausgeführt werden, wie Amazon EC2-Instances, AWS Fargate-Container oder Lambda-Funktionen, stellen Sie diese in privaten Subnetzen bereit, und zwar basierend auf Ihren Gruppen aus dem ersten Schritt.
4. Für vollständig verwaltete AWS-Services, wie Amazon DynamoDB, Amazon Kinesis oder Amazon SQS, sollten Sie VPC-Endpunkte als Standard für den Zugriff über private IP-Adressen verwenden.

Ressourcen

Zugehörige bewährte Methoden:

- REL02 Planen der Netzwerktopologie
- PERF04-BP01 Verstehen der Auswirkungen des Netzwerks auf die Leistung

Zugehörige Videos:

- AWS re:Invent 2023 – AWS networking foundations

Zugehörige Beispiele:

- VPC-Beispiele
- Access container applications privately on Amazon ECS by using AWS Fargate, AWS PrivateLink, and a Network Load Balancer
- Serve static content in an Amazon S3 bucket through a VPC by using Amazon CloudFront

SEC05-BP02 Kontrollieren des Datenverkehrsflusses innerhalb Ihrer Netzwerkebenen

Verwenden Sie innerhalb der einzelnen Ebenen Ihres Netzwerks eine weitere Segmentierung, um den Datenverkehr auf die für die einzelnen Workloads erforderlichen Flüsse zu beschränken. Konzentrieren Sie sich zunächst auf die Kontrolle des Datenverkehrs zwischen dem Internet oder anderen externen Systemen eines Workloads und Ihrer Umgebung (Nord-Süd-Verkehr). Betrachten Sie anschließend die Ströme zwischen verschiedenen Komponenten und Systemen(Ost-West-Verkehr).

Gewünschtes Ergebnis: Sie lassen nur die Netzwerkflüsse zu, die für die Kommunikation der Komponenten Ihrer Workloads untereinander, mit ihren Clients und mit allen anderen Services, von denen sie abhängig sind, erforderlich sind. Ihr Design berücksichtigt Überlegungen wie öffentlichen im Vergleich zu privatem Ingress und Egress, Datenklassifizierung, regionale Vorschriften und Protokollanforderungen. Wo immer es möglich ist, bevorzugen Sie Punkt-zu-Punkt-Flüsse gegenüber Netzwerk-Peering im Rahmen des Prinzips der geringsten Berechtigung.

Typische Anti-Muster:

- Sie verfolgen bei der Netzwerksicherheit einen Perimeter-basierten Ansatz und kontrollieren den Datenverkehr nur an den Grenzen Ihrer Netzwerkebenen.
- Sie gehen davon aus, dass der gesamte Verkehr innerhalb einer Netzwerkebene authentifiziert und autorisiert ist.
- Sie kontrollieren entweder den eingehenden oder den ausgehenden Datenverkehr, aber nicht beide.
- Sie verlassen sich bei der Authentifizierung und Autorisierung des Datenverkehrs ausschließlich auf Ihre Workload-Komponenten und Netzwerkkontrollen.

Vorteile der Einführung dieser bewährten Methode: Diese Vorgehensweise trägt dazu bei, das Risiko unbefugter Bewegungen innerhalb Ihres Netzwerks zu verringern, und fügt Ihren Workloads eine zusätzliche Autorisierungsebene hinzu. Durch die Kontrolle des Datenverkehrs können Sie den Umfang der Auswirkungen eines Sicherheitsvorfalls begrenzen und die Erkennung und Reaktion beschleunigen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Netzwerkebenen helfen zwar bei der Abgrenzung von Komponenten Ihres Workloads, die eine ähnliche Funktion, eine ähnliche Datensensibilität und ein ähnliches Verhalten aufweisen. Sie können jedoch eine wesentlich feinere Ebene der Datenverkehrskontrolle schaffen, indem Sie Techniken zur weiteren Segmentierung von Komponenten innerhalb dieser Ebenen einsetzen, die dem Prinzip der geringsten Berechtigung folgen. Innerhalb von AWS werden Netzwerkebenen in erster Linie über Subnetze entsprechend den IP-Adressbereichen innerhalb eines Amazon VPC definiert. Ebenen können auch über verschiedene VPCs definiert werden, z. B. für die Gruppierung von Microservice-Umgebungen nach Business Domain. Wenn Sie mehrere VPCs verwenden, vermitteln Sie das Routing mit einer AWS Transit Gateway. Dies ermöglicht zwar die Kontrolle des Datenverkehrs auf Layer-4-Ebene (IP-Adressen- und Portbereiche) mithilfe von Sicherheitsgruppen und Routing-Tabellen, aber Sie können mit zusätzlichen Services, wie AWS PrivateLink, Amazon Route 53-Resolver-DNS-Firewall, AWS Network Firewall und AWS WAF weitere Kontrolle erlangen.

Verstehen und inventarisieren Sie den Datenfluss und die Kommunikationsanforderungen Ihrer Workloads in Bezug auf verbindungsauslösende Parteien, Ports, Protokolle und Netzwerkebenen. Prüfen Sie die verfügbaren Protokolle für den Verbindungsaufbau und die Datenübertragung, um diejenigen auszuwählen, die Ihre Schutzanforderungen erfüllen (z. B. HTTPS statt HTTP). Erfassen Sie diese Anforderungen sowohl an den Grenzen Ihrer Netzwerke als auch innerhalb jeder Ebene. Sobald diese Anforderungen identifiziert sind, prüfen Sie die Möglichkeiten, um nur den erforderlichen Datenverkehr an jedem Verbindungspunkt zuzulassen. Ein guter Ausgangspunkt ist die Verwendung von Sicherheitsgruppen innerhalb Ihrer VPC, da sie an Ressourcen angehängt werden können, die eine Elastic-Network-Schnittstelle (ENI) verwenden, wie Amazon EC2-Instances, Amazon ECS-Aufgaben, Amazon EKS-Pods oder Amazon RDS-Datenbanken. Im Gegensatz zu einer Layer-4-Firewall kann eine Sicherheitsgruppe eine Regel haben, die den Datenverkehr einer anderen Sicherheitsgruppe anhand ihrer Kennung zulässt, wodurch Aktualisierungen minimiert werden, wenn sich die Ressourcen innerhalb der Gruppe im Laufe der Zeit ändern. Sie können den Datenverkehr auch mithilfe von Sicherheitsgruppen nach eingehenden und ausgehenden Regeln filtern.

Wenn sich der Datenverkehr zwischen VPCs bewegt, ist es üblich, VPC-Peering für einfaches Routing oder AWS Transit Gateway für komplexes Routing zu verwenden. Mit diesen Ansätzen erleichtern Sie den Datenverkehrsfluss zwischen dem Bereich der IP-Adressen des Quell- und des Zielnetzwerks. Wenn Ihr Workload jedoch nur Datenverkehrsflüsse zwischen bestimmten Komponenten in verschiedenen VPCs erfordert, sollten Sie eine Punkt-zu-Punkt-Verbindung mit AWS PrivateLink verwenden. Bestimmen Sie dazu, welcher Service als Produzent und welcher als Verbraucher fungieren soll. Stellen Sie einen kompatiblen Load Balancer für den Produzenten bereit, schalten Sie PrivateLink entsprechend ein und akzeptieren Sie dann eine Verbindungsanfrage des Verbrauchers. Dem Produzenten-Service wird dann eine private IP-Adresse aus der VPC des Verbrauchers zugewiesen, die der Verbraucher für nachfolgende Anfragen verwenden kann. Dieser Ansatz reduziert die Notwendigkeit, die Netzwerke zu peeren. Beziehen Sie die Kosten für die Datenverarbeitung und den Load Balancer in die Bewertung von PrivateLink mit ein.

Sicherheitsgruppen und PrivateLink tragen zwar dazu bei, den Fluss zwischen den Komponenten Ihrer Workloads zu kontrollieren. Eine weitere wichtige Überlegung ist jedoch, wie Sie kontrollieren können, auf welche DNS-Domains Ihre Ressourcen zugreifen dürfen (falls überhaupt). Abhängig von der DHCP-Konfiguration Ihrer VPCs können Sie zwei verschiedene AWS-Services für diesen Zweck in Betracht ziehen. Die meisten Kunden verwenden den standardmäßigen Route 53-Resolver DNS-Service (auch Amazon-DNS-Server oder AmazonProvidedDNS genannt), der für VPCs unter der +2-Adresse ihres CIDR-Bereichs verfügbar ist. Mit diesem Ansatz können Sie DNS-Firewall-Regeln erstellen und diese mit Ihrer VPC verknüpfen, die festlegen, welche Aktionen für die von Ihnen bereitgestellten Domain-Listen durchgeführt werden sollen.

Wenn Sie nicht den Route 53-Resolver verwenden, oder wenn Sie den Resolver mit tieferen Prüf- und Flusskontrollfunktionen als der Domain-Filterung ergänzen wollen, sollten Sie die Bereitstellung eines AWS Network Firewall erwägen. Dieser Service prüft einzelne Pakete anhand von zustandslosen oder zustandsbehafteten Regeln, um zu entscheiden, ob der Datenverkehr verweigert oder zugelassen werden soll. Einen ähnlichen Ansatz können Sie für die Filterung des eingehenden Internetdatenverkehrs zu Ihren öffentlichen Endpunkten mit AWS WAF verfolgen. Weitere Hinweise zu diesen Services finden Sie unter SEC05-BP03 Implement inspection-based protection.

Implementierungsschritte

1. Identifizieren Sie die erforderlichen Datenflüsse zwischen den Komponenten Ihrer Workloads.
2. Wenden Sie mehrere Kontrollen mit einem Ansatz der Tiefenverteidigung sowohl für den eingehenden als auch für den ausgehenden Datenverkehr an, einschließlich der Verwendung von Sicherheitsgruppen und Routing-Tabellen.

3. Verwenden Sie Firewalls, um eine feinkörnige Kontrolle über den Netzwerkverkehr in, aus und zwischen Ihren VPCs zu definieren, wie z. B. die Route 53 Resolver DNS Firewall, AWS Network Firewall, und AWS WAF. Erwägen Sie den Einsatz von AWS Firewall Manager für die zentrale Konfiguration und Verwaltung Ihrer Firewall-Regeln in Ihrer Organisation.

Ressourcen

Zugehörige bewährte Methoden:

- REL03-BP01 Segmentierung Ihres Workloads
- SEC09-BP02 Erzwingen einer Verschlüsselung bei der Übertragung

Zugehörige Dokumente:

- Security best practices for your VPC
- AWS Network Optimization Tips
- Guidance for Network Security on AWS
- Secure your VPC's outbound network traffic in the AWS Cloud

Zugehörige Tools:

- AWS Firewall Manager

Zugehörige Videos:

- AWS Transit Gateway reference architectures for many VPCs
- Application Acceleration and Protection with Amazon CloudFront, AWS WAF, and AWS Shield
- AWS re:Inforce 2023: Firewalls and where to put them

Zugehörige Beispiele:

- Lab: CloudFront for Web Application

SEC05-BP03 Implementieren eines prüfungsbasierten Schutzes

Richten Sie Kontrollpunkte für den Datenverkehr zwischen Ihren Netzwerkebenen ein, um sicherzustellen, dass die Daten während der Übertragung den erwarteten Kategorien und Mustern entsprechen. Analysieren Sie Datenverkehrsströme, Metadaten und Muster, um Ereignisse effektiver zu identifizieren, zu erkennen und darauf zu reagieren.

Gewünschtes Ergebnis: Der Datenverkehr, der zwischen Ihren Netzwerkebenen verläuft, wird geprüft und autorisiert. Entscheidungen über das Zulassen oder Verweigern von Zugriffen beruhen auf expliziten Regeln, Informationen über Bedrohungen und Abweichungen vom Grundverhalten. Der Schutz wird strenger, je näher der Datenverkehr an sensible Daten heranrückt.

Typische Anti-Muster:

- Ausschließlich auf Firewall-Regeln vertrauen, die auf Ports und Protokollen basieren Vorteile intelligenter Systeme außer Acht lassen
- Erstellen von Firewall-Regeln auf der Grundlage bestimmter aktueller Bedrohungsmuster, die sich ändern können
- Überprüfung des Datenverkehrs beschränkt auf den Übergang von privaten zu öffentlichen Subnetzen oder von öffentlichen Subnetzen zum Internet
- Sie verfügen nicht über eine Basisansicht Ihres Netzwerkdatenverkehrs, die Sie auf Verhaltensanomalien hin überprüfen können.

Vorteile der Einführung dieser bewährten Methode: Prüfungssysteme ermöglichen es Ihnen, intelligente Regeln zu erstellen, z. B. den Datenverkehr nur dann zuzulassen oder zu verweigern, wenn bestimmte Bedingungen in den Datenverkehrsdaten vorliegen. Profitieren Sie von verwalteten Regelsätzen von AWS und Partnern, die auf den neuesten Bedrohungsdaten basieren, da sich die Bedrohungslandschaft im Laufe der Zeit verändert. Dadurch verringert sich der Aufwand für die Pflege von Regeln und die Suche nach Indikatoren für eine Gefährdung, wodurch das Potenzial für Fehlalarme reduziert wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Kontrollieren Sie Ihren zustandsbehafteten und zustandslosen Netzwerkverkehr im Detail mit AWS Network Firewall oder anderen Firewalls und Intrusion Prevention Systems (IPS) in AWS Marketplace, die Sie hinter einer (GWLB) bereitstellen können. AWS Network Firewall unterstützt Suricata-kompatible Open-Source-IPS-Spezifikationen zum Schutz Ihres Workloads.

Sowohl die Lösung AWS Network Firewall als auch die Lösungen der Anbieter, die eine GWLB verwenden, unterstützen verschiedene Modelle für die Bereitstellung von Inline-Prüfungen. Sie können zum Beispiel Prüfungen pro VPC durchführen, die Prüfungen in einer VPC zentralisieren oder in einem hybriden Modell bereitstellen, bei dem der Ost-West-Verkehr durch eine Prüfungs-VPC fließt und der Internet-Eingang pro VPC geprüft wird. Eine weitere Frage ist, ob die Lösung das Unwrapping von Transport Layer Security (TLS) unterstützt und damit eine Deep Packet Inspection für Datenverkehrsflüsse in beide Richtungen ermöglicht. Weitere Informationen und ausführliche Details zu diesen Konfigurationen finden Sie in den AWS Network Firewall Leitlinien für bewährte Methoden.

Wenn Sie Lösungen verwenden, die Out-of-Band-Prüfungen durchführen, wie z. B. die pcap-Analyse von Paketdaten von Netzwerkschnittstellen, die im Promiscuous-Modus arbeiten, können Sie die VPC traffic mirroring konfigurieren. Gespiegelter Datenverkehr wird auf die verfügbare Bandbreite Ihrer Schnittstellen angerechnet und unterliegt denselben Datenübertragungsgebühren wie nicht gespiegelter Datenverkehr. Sie können sehen, ob virtuelle Versionen dieser Appliances auf der AWS Marketplace verfügbar sind, die möglicherweise eine Inline-Bereitstellung hinter einer GWLB unterstützen.

Bei Komponenten, die über HTTP-basierte Protokolle abgewickelt werden, schützen Sie Ihre Anwendung mit einer Web Application Firewall (WAF) vor gängigen Bedrohungen. AWS WAF ist eine Web Application Firewall, mit der Sie HTTP(S)-Anfragen, die Ihren konfigurierbaren Regeln entsprechen, überwachen und blockieren können, bevor sie an Amazon API Gateway, Amazon CloudFront, AWS AppSync oder Application Load Balancer gesendet werden. Wenn Sie die Bereitstellung Ihrer Web Application Firewall prüfen, sollten Sie eine Deep Packet Inspection in Betracht ziehen, da einige Firewalls verlangen, dass Sie TLS vor der Überprüfung des Datenverkehrs beenden. Um mit AWS WAF zu beginnen, können Sie Von AWS verwaltete Regeln in Kombination mit Ihren eigenen oder mit bestehenden Partner-Integrationen verwenden.

Sie können Sicherheitsgruppen für AWS WAF, AWS Shield Advanced, AWS Network Firewall und Amazon VPC in Ihrer gesamten AWS-Organisation mit AWS Firewall Manager zentral verwalten.

Implementierungsschritte

1. Legen Sie fest, ob Sie die Inspektionsregeln weit fassen können, z. B. durch eine Inspektions-VPC, oder ob Sie einen granulareren Ansatz pro VPC benötigen.
2. Für Inline-Prüfungslösungen:

   a. Wenn Sie AWS Network Firewall verwenden, erstellen Sie Regeln, Firewall-Richtlinien und die Firewall selbst. Sobald diese konfiguriert sind, können Sie den Datenverkehr an den Endpunkt der Firewall leiten, um die Prüfung zu aktivieren.
   b. Wenn Sie eine Appliance eines Drittanbieters mit einem Gateway Load Balancer (GWLB) verwenden, stellen Sie Ihre Appliance in einer oder mehreren Verfügbarkeitszonen bereit und konfigurieren sie. Dann erstellen Sie Ihre GWLB, den Endservice, den Endpunkt und konfigurieren das Routing für Ihren Datenverkehr.
3. Für Out-of-Band-Prüfungslösungen:
   1. Aktivieren Sie die VPC-Datenverkehrsspiegelung auf den Schnittstellen, auf denen der ein- und ausgehende Datenverkehr gespiegelt werden soll. Sie können Amazon EventBridge-Regeln verwenden, um eine AWS Lambda-Funktion aufzurufen, die die Datenverkehrsspiegelung auf Schnittstellen aktiviert, wenn neue Ressourcen erstellt werden. Richten Sie die Sitzungen zur Datenverkehrsspiegelung auf den Network Load Balancer vor Ihrer Appliance, der den Datenverkehr verarbeitet.
4. Für Lösungen für eingehenden Internetdatenverkehr:
   a. Um AWS WAF zu konfigurieren, beginnen Sie mit der Konfiguration einer Internet-Zugriffssteuerungsliste (Web Access Control List, web ACL). Die web ACL ist eine Sammlung von Regeln mit einer seriell verarbeiteten Standardaktion (ALLOW oder DENY), die definiert, wie Ihre WAF den Datenverkehr behandelt. Sie können Ihre eigenen Regeln und Gruppen erstellen oder verwaltete Regelgruppen von AWS in Ihrer web ACL verwenden.
   b. Sobald Ihre web ACL konfiguriert ist, verknüpfen Sie die Web-ACL mit einer AWS-Ressource (z. B. einer Application Load Balancer, API Gateway-REST-API oder CloudFront-Distribution), um den Webverkehr zu schützen.

Ressourcen

Zugehörige Dokumente:

- What is Traffic Mirroring?
- Implementing inline traffic inspection using third-party security appliances
- AWS Network Firewall example architectures with routing
- Centralized inspection architecture with AWS Gateway Load Balancer and AWS Transit Gateway

Zugehörige Beispiele:

- Best practices for deploying Gateway Load Balancer
- TLS inspection configuration for encrypted egress traffic and AWS Network Firewall

Zugehörige Tools:

- AWS Marketplace IDS/IPS

SEC05-BP04 Automatisieren des Netzwerkschutzes

Automatisieren Sie die Bereitstellung Ihres Netzwerkschutzes mit DevOps-Verfahren wie Infrastructure as Code (IaC) und CI/CD-Pipelines. Diese Praktiken können Ihnen helfen, Änderungen an Ihrem Netzwerkschutz über ein Versionskontrollsystem zu verfolgen, den Zeitaufwand für die Bereitstellung von Änderungen zu reduzieren und zu erkennen, wenn Ihr Netzwerkschutz von der gewünschten Konfiguration abweicht.

Gewünschtes Ergebnis: Sie definieren Netzwerkschutzmaßnahmen mit Vorlagen und übertragen diese in ein Versionskontrollsystem. Automatisierte Pipelines werden initiiert, wenn neue Änderungen vorgenommen werden, die ihre Prüfung und Bereitstellung orchestrieren. Richtlinienprüfungen und andere statische Tests dienen der Validierung von Änderungen vor der Bereitstellung. Sie stellen die Änderungen in einer Staging-Umgebung bereit, um zu überprüfen, ob die Kontrollen wie erwartet funktionieren. Die Bereitstellung in Ihrer Produktionsumgebung erfolgt ebenfalls automatisch, sobald die Kontrollen genehmigt sind.

Typische Anti-Muster:

- darauf vertrauen, dass die einzelnen Workload-Teams ihren kompletten Netzwerkstack, Schutzmaßnahmen und Automatisierungen selbst definieren keine zentrale Veröffentlichung von Standardaspekten des Netzwerkstapels und der Schutzmechanismen für Workload-Teams zur Nutzung
- auf ein zentrales Netzwerkteam vertrauen, das alle Aspekte des Netzwerks, der Schutzmaßnahmen und der Automatisierungen definiert Verzicht auf die Delegation von Workload-spezifischen Aspekten des Netzwerkstacks und der Schutzmaßnahmen an das Team des Workloads
- Beibehalten eines ausgewogenen Verhältnisses zwischen Zentralisierung und Delegation zwischen einem Netzwerkteam und Workload-Teams, aber keine Anwendung konsistenter Test- und Bereitstellungsstandards über Ihre IaC-Vorlagen und CI/CD-Pipelines hinweg Unterlassen der Erfassung erforderlicher Konfigurationen in Tools, die Ihre Vorlagen auf Einhaltung überprüfen

Vorteile der Einführung dieser bewährten Methode: Durch die Verwendung von Vorlagen zur Definition Ihres Netzwerkschutzes können Sie Änderungen im Laufe der Zeit mit einem Versionskontrollsystem verfolgen und vergleichen. Der Einsatz von Automatisierung zum Testen und Bereitstellen von Änderungen schafft Standardisierung und Vorhersehbarkeit, erhöht die Chancen auf eine erfolgreiche Bereitstellung und reduziert die sich wiederholenden manuellen Konfigurationen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Eine Reihe von Netzwerkschutzkontrollen, die in SEC05-BP02 Control traffic flows within your network layers und SEC05-BP03 Implement inspection-based protection beschrieben sind, verfügen über verwaltete Regelsysteme, die automatisch auf der Grundlage der neuesten Bedrohungsdaten aktualisiert werden können. Beispiele für den Schutz Ihrer Web-Endpunkte sind AWS WAF verwaltete Regeln und AWS Shield Advanced automatische DDoS-Abwehr auf Anwendungsebene. Verwenden Sie AWS Network Firewall-verwaltete Regelgruppen, um auch bei Domain-Listen mit geringer Reputation und Bedrohungssignaturen auf dem Laufenden zu bleiben.

Neben den verwalteten Regeln empfehlen wir Ihnen, DevOps-Praktiken einzusetzen, um die Bereitstellung Ihrer Netzwerkressourcen, Schutzmaßnahmen und der von Ihnen festgelegten Regeln zu automatisieren. Sie können diese Definitionen in AWS CloudFormation oder einem anderen Infrastructure as Code (IaC)-Tool Ihrer Wahl erfassen, sie an ein Versionskontrollsystem übergeben und sie über CI/CD-Pipelines bereitstellen. Nutzen Sie diesen Ansatz, um die traditionellen Vorteile von DevOps für die Verwaltung Ihrer Netzwerkkontrollen zu nutzen, wie z. B. besser vorhersehbare Releases, automatisierte Tests mit Tools wie AWS CloudFormation Guard und die Erkennung von Abweichungen zwischen Ihrer bereitgestellten Umgebung und Ihrer gewünschten Konfiguration.

Basierend auf den Entscheidungen, die Sie im Rahmen von SEC05-BP01 Erstellen von Netzwerkebenen getroffen haben, verfügen Sie möglicherweise über einen zentralen Verwaltungsansatz für die Erstellung von VPCs, die für Ingress-, Egress- und Inspektionsflüsse bestimmt sind. Diese VPCs können Sie, wie in der AWS Security Reference Architecture (AWS SRA) beschrieben, in einem speziellen Netzwerkinfrastrukturkonto definieren. Sie können ähnliche Techniken verwenden, um die von Ihren Workloads in anderen Konten verwendeten VPCs, deren Sicherheitsgruppen, AWS Network Firewall-Bereitstellungen, Route 53-Resolver-Regeln und DNS-Firewall-Konfigurationen sowie andere Netzwerkressourcen zentral zu definieren. Sie können diese Ressourcen mit Ihren anderen Konten mit der AWS Resource Access Manager teilen. Mit diesem Ansatz können Sie das automatisierte Testen und die Bereitstellung Ihrer Netzwerkkontrollen für das Netzwerkkonto vereinfachen, da Sie nur ein Ziel verwalten müssen. Sie können dies in einem

hybriden Modell tun, bei dem Sie bestimmte Kontrollen zentral bereitstellen und gemeinsam nutzen und andere Kontrollen an die einzelnen Workload-Teams und ihre jeweiligen Konten delegieren.

Implementierungsschritte

1. Legen Sie fest, welche Aspekte des Netzwerks und des Schutzes zentral definiert werden und welche Ihre Workload-Teams verwalten können.
2. Erstellen Sie Umgebungen zum Testen und Bereitstellen von Änderungen an Ihrem Netzwerk und dessen Schutzmaßnahmen. Verwenden Sie zum Beispiel ein Netzwerk-Testkonto und ein Netzwerk-Produktionskonto.
3. Legen Sie fest, wie Sie Ihre Vorlagen in einem Versionskontrollsystem speichern und pflegen wollen. Speichern Sie zentrale Vorlagen in einem Repository, das sich von den Workload-Repositories unterscheidet, während Workload-Vorlagen in Repositories gespeichert werden können, die speziell für diesen Workload gelten.
4. Erstellen Sie CI/CD-Pipelines zum Testen und Bereitstellen von Vorlagen. Definieren Sie Tests, um zu prüfen, ob Fehlkonfigurationen vorliegen und ob die Vorlagen den Standards Ihres Unternehmens entsprechen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen

Zugehörige Dokumente:

- AWS Security Reference Architecture – Network account

Zugehörige Beispiele:

- AWS Deployment Pipeline Reference Architecture
- NetDevSecOps to modernize AWS networking deployments
- Integrating AWS CloudFormation security tests with AWS Security Hub and AWS CodeBuild reports

Zugehörige Tools:

- AWS CloudFormation
- AWS CloudFormation Guard
- cfn_nag

SEC 6. Wie schützen Sie Ihre Datenverarbeitungsressourcen?

Datenverarbeitungsressourcen in Ihrem Workload erfordern mehrere Ebenen der Abwehr zum Schutz vor externen und internen Bedrohungen. Zu den Datenverarbeitungsressourcen zählen EC2-Instances, Container, AWS Lambda-Funktionen, Datenbankservices, IoT-Geräte und mehr.

Bewährte Methoden

- SEC06-BP01 Schwachstellenmanagement
- SEC06-BP02 Bereitstellen von Datenverarbeitung über gehärtete Images
- SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs
- SEC06-BP04 Validieren der Softwareintegrität
- SEC06-BP05 Automatisieren des Datenverarbeitungsschutzes

SEC06-BP01 Schwachstellenmanagement

Überprüfen und Patchen Sie Ihren Code, Ihre Abhängigkeiten und Ihre Infrastruktur häufig auf Schwachstellen, um sich vor neuen Bedrohungen zu schützen.

Gewünschtes Ergebnis: Erstellen und Verwalten eines Programms für das Schwachstellenmanagement. Überprüfen und Patchen Sie regelmäßig Ressourcen wie Amazon EC2-Instances, Amazon Elastic Container Service (Amazon ECS)-Container und Amazon Elastic Kubernetes Service (Amazon EKS)-Workloads. Konfigurieren Sie Wartungszeitfenster für AWS-verwaltete Ressourcen wie Amazon Relational Database Service (Amazon RDS)-Datenbanken. Verwenden Sie statisches Code-Scanning, um Anwendungsquellcode auf verbreitete Probleme zu überprüfen. Ziehen Sie Penetrationstests für Webanwendungen in Betracht, wenn Ihre Organisation über die entsprechenden Fähigkeiten verfügt oder externe Unterstützung erhalten kann.

Typische Anti-Muster:

- Fehlen eines Programms für das Schwachstellenmanagement
- Durchführung von System-Patches ohne Berücksichtigung des Schweregrads oder der Risikovermeidung

- Verwendung von Software nach dem vom Anbieter angegebenen Lebenszyklusenddatum
- Bereitstellung von Code für die Produktion, bevor dieser auf Sicherheitsprobleme untersucht wurde

Vorteile der Nutzung dieser bewährten Methode:

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Ein Programm für das Schwachstellenmanagement beinhaltet Sicherheitsbewertungen, die Identifizierung von Problemen sowie die Priorisierung und Durchführung von Patching-Vorgängen im Rahmen der Behebung der Probleme. Automatisierung ist der Schlüssel zur kontinuierlichen Prüfung von Workloads auf Probleme und unbeabsichtigte Offenlegung in Netzwerken sowie für die Durchführung von Abhilfemaßnahmen. Die Automatisierung der Erstellung und Aktualisierung von Ressourcen spart Zeit und senkt die Gefahr von Konfigurationsfehlern, die zu weiteren Problemen führen können. Ein gut gestaltetes Programm für das Schwachstellenmanagement sollte auch Schwachstellentests in den Entwicklungs- und Bereitstellungsphasen des Softwarelebenszyklus beinhalten. Die Implementierung des Schwachstellenmanagements während der Entwicklung und der Bereitstellung verringert die Gefahr, dass eine Schwachstelle in Ihre Produktionsumgebung gelangt.

Die Implementierung eines Programms für das Schwachstellenmanagement erfordert ein gutes Verständnis des AWS-Modells der geteilten Verantwortung und seiner Beziehung zu Ihren spezifischen Workloads. In diesem Modell der geteilten Verantwortung ist AWS für den Schutz der Infrastruktur der AWS Cloud verantwortlich. Diese Infrastruktur umfasst die Hardware, Software, Netzwerke und Einrichtungen, in bzw. auf denen AWS Cloud-Services ausgeführt werden. Sie sind für die Sicherheit in der Cloud verantwortlich, zum Beispiel für die eigentlichen Daten, die Sicherheitskonfiguration und Verwaltungsaufgaben für Amazon EC2-Instances sowie für die Sicherstellung, dass Ihre Amazon S3-Objekte korrekt klassifiziert und konfiguriert sind. Ihr Konzept für das Schwachstellenmanagement kann auch je nach den von Ihnen genutzten Services variieren. So verwaltet beispielsweise AWS die Patches für unseren verwalteten relationalen Datenbankservice Amazon RDS, Sie sind jedoch selbst für das Patchen selbst gehosteter Datenbanken verantwortlich.

AWS bietet eine Reihe von Services zur Unterstützung Ihres Programms für das Schwachstellenmanagement. Amazon Inspector untersucht kontinuierlich AWS-Workloads auf Softwareprobleme und nicht beabsichtigte Netzwerkzugriffe. AWS Systems Manager Patch Manager hilft bei der Verwaltung des Patchings für Ihre Amazon EC2-Instances. Amazon Inspector und Systems Manager können in AWS Security Hub angezeigt werden. Dieser Managementservice

für den Cloud-Sicherheitsstatus hilft dabei, AWS-Sicherheitsprüfungen zu automatisieren und Sicherheitsbenachrichtigungen zu zentralisieren.

Amazon CodeGuru kann mit der Analyse von statischem Code dabei helfen, potenzielle Probleme in Java- und Python-Anwendungen zu erkennen.

Implementierungsschritte

- Konfigurieren Sie Amazon Inspector: Amazon Inspector erkennt automatisch neu gestartete Amazon EC2-Instances, Lambda-Funktionen und infrage kommende Container-Images, die an Amazon ECR übertragen wurden, und untersucht diese sofort auf Softwareprobleme, potenzielle Fehler und unbeabsichtigte Netzwerkoffenlegung.
- Untersuchen Sie den Quellcode: Überprüfen Sie Bibliotheken und Abhängigkeiten auf Probleme und Fehler. Amazon CodeGuru kann diese Überprüfungen vornehmen und Empfehlungen zur Behebung verbreiteter Sicherheitsprobleme für Java- und Python-Anwendungen bereitstellen. Die OWASP Foundation veröffentlicht eine Liste von Quellcodeanalysetools (auch als SAST-Tools bezeichnet).
- Implementieren Sie einen Mechanismus zur Untersuchung und zum Patching Ihrer bestehenden Umgebung sowie zur Untersuchung im Rahmen eines CI/CD-Pipeline-Erstellungsprozesses: Implementieren Sie einen Mechanismus zur Untersuchung und zum Patching von Problemen in Ihren Abhängigkeiten und Betriebssystemen, um Schutz gegen neue Bedrohungen zu bieten. Lassen Sie diesen Mechanismus regelmäßig laufen. Das Software-Schwachstellenmanagement ist wichtig, um zu verstehen, wo Patches angebracht oder Softwareprobleme behoben werden müssen. Priorisieren Sie die Abhilfemaßnahmen zu potenziellen Sicherheitsproblemen durch die frühzeitige Einbettung von Schwachstellenanalysen in Ihre Pipeline für kontinuierliche Integration und kontinuierliche Bereitstellung (Continuous Integration/Continuous Delivery, CI/CD). Ihr Konzept kann je nach den von Ihnen genutzten AWS-Services variieren. Fügen Sie zur Prüfung auf potenzielle Probleme in der Software, die in Amazon EC2-Instances ausgeführt wird, Ihrer Pipeline Amazon Inspector hinzu, damit Sie benachrichtigt werden und den Prozess anhalten können, wenn Probleme oder mögliche Fehler erkannt werden. Amazon Inspector überwacht Ressourcen kontinuierlich. Sie können auch Open-Source-Produkte wie OWASP Dependency-Check, Snyk, OpenVAS, Paketmanager oder AWS Partner-Tools für das Schwachstellenmanagement verwenden.
- Verwenden Sie AWS Systems Manager: Sie sind für das Patch-Management für Ihre AWS-Ressourcen verantwortlich, einschließlich Amazon Elastic Compute Cloud (Amazon EC2)-Instances, Amazon Machine Images (AMIs) und anderer Datenverarbeitungsressourcen. AWS Systems Manager Patch Manager automatisiert das Patchen verwalteter Instances

mit sicherheitsrelevanten und anderen Arten von Updates. Patch Manager kann für die Durchführung von Patches auf Amazon EC2-Instances für Betriebssysteme und Anwendungen verwendet werden, darunter Microsoft-Anwendungen, Windows-Service Packs und kleinere Versionsaktualisierungen für auf Linux basierende Instances. Zusätzlich zu Amazon EC2 kann Patch Manager auch für das Patching von On-Premises-Servern genutzt werden.

Eine Liste der unterstützten Betriebssysteme finden Sie unter Unterstützte Betriebssysteme im Systems Manager-Benutzerhandbuch. Sie können Instances scannen, um nur fehlende Patches anzuzeigen, oder Sie können scannen und automatisch alle fehlenden Patches installieren.

- Verwenden Sie AWS Security Hub: Security Hub bietet eine umfassende Ansicht Ihres Sicherheitszustands in AWS. Es erfasst Sicherheitsdaten über mehrere AWS-Services hinweg und stellt diese Ergebnisse in einem standardisierten Format bereit, damit Sie die Sicherheitsergebnisse für AWS-Services priorisieren können.
- Verwenden Sie AWS CloudFormation: AWS CloudFormation ist ein Infrastructure-as-Code (Iac)-Service, der das Schwachstellenmanagement durch die Automatisierung der Ressourcenbereitstellung und die Standardisierung der Ressourcenarchitektur über mehrere Konten und Umgebungen hinweg unterstützt.

Ressourcen

Zugehörige Dokumente:

- AWS Systems Manager
- Security Overview of AWS Lambda (Übersicht zur Sicherheit von AWS Lambda)
- Amazon CodeGuru
- Improved, Automated Vulnerability Management for Cloud Workloads with a New Amazon Inspector (Verbessertes und automatisiertes Schwachstellenmanagement für Cloud-Workloads mit einem neuen Amazon Inspector)
- Automate vulnerability management and remediation in AWS using Amazon Inspector and AWS Systems Manager – Part 1 (Automatisierung des Schwachstellenmanagements und von Abhilfemaßnahmen in AWS mit Amazon Inspector und AWS Systems Manager – Teil 1)

Zugehörige Videos:

- Securing Serverless and Container Services (Schutz von Serverless- und Container-Services)

- Security best practices for the Amazon EC2 instance metadata service (Bewährte Sicherheitsmethoden für den Amazon EC2-Instance-Metadatenservice)

SEC06-BP02 Bereitstellen von Datenverarbeitung über gehärtete Images

Bieten Sie weniger Möglichkeiten für einen unbeabsichtigten Zugriff auf Ihre Laufzeitumgebungen, indem Sie sie über gehärtete Images bereitstellen. Beziehen Sie Laufzeit-Abhängigkeiten wie Container-Images und Anwendungsbibliotheken nur von vertrauenswürdigen Registern und überprüfen Sie deren Signaturen. Erstellen Sie Ihre eigenen privaten Register, um vertrauenswürdige Images und Bibliotheken für die Verwendung in Ihren Build- und Bereitstellungsprozessen zu speichern.

Gewünschtes Ergebnis: Ihre Datenverarbeitungsressourcen werden über gehärtete Baseline-Images bereitgestellt. Sie rufen externe Abhängigkeiten, wie Container-Images und Anwendungsbibliotheken, nur aus vertrauenswürdigen Registern ab und überprüfen deren Signaturen. Diese werden in privaten Registern gespeichert, auf die Ihre Build- und Bereitstellungsprozesse verweisen können. Sie überprüfen und aktualisieren Images und Abhängigkeiten regelmäßig, um sich vor neu entdeckten Schwachstellen zu schützen.

Typische Anti-Muster:

- Abrufen von Images und Bibliotheken aus vertrauenswürdigen Registern, ohne deren Signaturen zu überprüfen oder Schwachstellen zu scannen, bevor sie eingesetzt werden
- Härtung von Images, ohne sie regelmäßig auf neue Schwachstellen zu testen oder auf die neueste Version zu aktualisieren
- Installation oder Nichtentfernung von Softwarepaketen, die während des erwarteten Lebenszyklus des Images nicht benötigt werden
- Vertrauen auf Patches als einzige Methode, um Datenverarbeitungsressourcen in der Produktion auf dem neuesten Stand zu halten Die alleinige Verwendung von Patches kann immer noch dazu führen, dass Datenverarbeitungsressourcen im Laufe der Zeit von dem gehärteten Standard abweichen. Patches sind außerdem nicht in der Lage, Malware zu entfernen, die möglicherweise von einem Bedrohungsakteur während eines Sicherheitsvorfalls installiert wurde.

Vorteile der Einführung dieser bewährten Methode: Das Härten von Images trägt dazu bei, die Anzahl der in Ihrer Laufzeitumgebung verfügbaren Pfade zu reduzieren, die unbeabsichtigten Zugriff auf nicht autorisierte Benutzer oder Services ermöglichen können. Auch das Ausmaß der Auswirkungen eines unbeabsichtigten Zugriffs kann damit verringert werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Um Ihre Systeme abzusichern, sollten Sie mit den neuesten Versionen von Betriebssystemen, Container-Images und Anwendungsbibliotheken beginnen. Wenden Sie Patches auf bekannte Probleme an. Reduzieren Sie das System auf ein Minimum, indem Sie nicht benötigte Anwendungen, Services, Gerätetreiber, Standardbenutzer und andere Anmeldeinformationen entfernen. Ergreifen Sie alle weiteren erforderlichen Maßnahmen, wie z. B. das Deaktivieren von Ports, um eine Umgebung zu schaffen, die nur über die von Ihren Workloads benötigten Ressourcen und Fähigkeiten verfügt. Von dieser Baseline aus können Sie dann Software, Agenten oder andere Prozesse installieren, die Sie für Zwecke wie die Überwachung des Workloads oder die Verwaltung von Schwachstellen benötigen.

Sie können den Aufwand für die Systemhärtung verringern, indem Sie Anleitungen nutzen, die von vertrauenswürdigen Quellen bereitgestellt werden, wie z. B. dem Center for Internet Security (CIS) und die Security Technical Implementation Guides (STIGs) der Defense Information Systems Agency (DISA). Wir empfehlen Ihnen, mit einem Amazon Machine Image (AMI) zu beginnen, das von AWS oder einem APN-Partner veröffentlicht wurde. Ferner empfehlen wir die Verwendung von AWS EC2 Image Builder, um die Konfiguration gemäß einer geeigneten Kombination von CIS- und STIG-Kontrollen zu automatisieren.

Es gibt zwar gehärtete Images und EC2 Image Builder-Rezepte, die die CIS- oder DISA-STIG-Empfehlungen anwenden. Sie werden jedoch möglicherweise feststellen, dass deren Konfiguration die erfolgreiche Ausführung Ihrer Software verhindert. In dieser Situation können Sie von einem nicht gehärteten Basis-Image ausgehen, Ihre Software installieren und dann schrittweise CIS-Kontrollen anwenden, um deren Auswirkungen zu testen. Testen Sie bei jeder CIS-Kontrolle, die die Ausführung Ihrer Software verhindert, ob Sie stattdessen die detaillierteren Härtungsempfehlungen der DISA implementieren können. Behalten Sie den Überblick über die verschiedenen CIS-Kontrollen und DISA-STIG-Konfigurationen, die Sie erfolgreich anwenden können. Verwenden Sie diese, um Ihre Rezepte für die Imagehärtung in EC2 Image Builder entsprechend zu definieren.

Für containerisierte Workloads sind gehärtete Images von Docker im öffentlichen Repository Amazon Elastic Container Registry (ECR) verfügbar. Sie können EC2 Image Builder verwenden, um Container-Images neben AMIs zu härten.

Ähnlich wie bei Betriebssystemen und Container-Images können Sie auch Code-Pakete (oder Bibliotheken) aus öffentlichen Repositorys beziehen, und zwar mithilfe von Tools wie pip, npm, Maven und NuGet. Wir empfehlen Ihnen, Code-Pakete zu verwalten, indem Sie private Repositorys,

wie z. B. innerhalb von AWS CodeArtifact, mit vertrauenswürdigen öffentlichen Repositorys verbinden. Diese Integration kann das Abrufen, Speichern und Aktualisieren von Paketen für Sie übernehmen. Ihre Anwendungserstellungsprozesse können dann die neueste Version dieser Pakete zusammen mit Ihrer Anwendung abrufen und testen, wobei Techniken wie Software Composition Analysis (SCA), Static Application Security Testing (SAST) und Dynamic Application Security Testing (DAST) zum Einsatz kommen.

Für Serverless Workloads, die AWS Lambda verwenden, vereinfachen Sie die Verwaltung von Paketabhängigkeiten mit Lambda-Ebenen. Verwenden Sie Lambda-Ebenen, um einen Satz von Standardabhängigkeiten, die von verschiedenen Funktionen gemeinsam genutzt werden, in einem eigenständigen Archiv zu konfigurieren. Sie können Ebenen durch einen eigenen Erstellungsprozess erstellen und pflegen, sodass Ihre Funktionen immer auf dem neuesten Stand sind.

Implementierungsschritte

- Härten des Betriebssystems. Verwenden Sie Basis-Images aus vertrauenswürdigen Quellen als Grundlage für die Erstellung Ihrer gehärteten AMIs. Mit EC2 Image Builder können Sie die auf Ihren Images installierte Software anpassen.
- Härten von containerisierten Ressourcen. Konfigurieren Sie containerisierte Ressourcen so, dass sie den bewährten Methoden im Bereich Sicherheit entsprechen. Implementieren Sie bei der Verwendung von Containern ECR Image Scanning in Ihrer Build-Pipeline und in regelmäßigen Abständen im Vergleich mit Ihrem Image-Repository, um nach CVEs in Ihren Containern zu suchen.
- Wenn Sie eine Serverless-Implementierung mit AWS Lambda verwenden, nutzen Sie Lambda-Ebenen, um den Funktionscode der Anwendung und gemeinsam genutzte abhängige Bibliotheken zu segmentieren. Konfigurieren Sie Codesignierung für Lambda, um sicherzustellen, dass nur vertrauenswürdiger Code in Ihren Lambda-Funktionen ausgeführt wird.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP05 Durchführen der Patch-Verwaltung

Zugehörige Videos:

- Deep dive into AWS Lambda security

Zugehörige Beispiele:

- Quickly build STIG-compliant AMI using EC2 Image Builder
- Building better container images
- Using Lambda layers to simplify your development process
- Develop & Deploy AWS Lambda Layers using Serverless Framework
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST and DAST tools

SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs

Nutzen Sie Automatisierung für die Bereitstellung, Konfiguration, Wartung und Untersuchung, wo immer dies möglich ist. Erwägen Sie den manuellen Zugriff auf Datenverarbeitungsressourcen in Notfällen oder in sicheren (Sandbox-)Umgebungen, wenn keine Automatisierung möglich ist.

Gewünschtes Ergebnis: Programmatische Skripte und Automatisierungsdokumente (Runbooks) erfassen autorisierte Aktionen in Ihren Datenverarbeitungsressourcen. Diese Runbooks werden entweder automatisch durch Systeme zur Erkennung von Änderungen oder manuell ausgelöst, wenn ein menschliches Urteilsvermögen erforderlich ist. Der direkte Zugriff auf Datenverarbeitungsressourcen wird nur in Notfällen gewährt, wenn keine Automatisierung verfügbar ist. Alle manuellen Aktivitäten werden protokolliert und in einen Überprüfungsprozess einbezogen, um Ihre Automatisierungsmöglichkeiten kontinuierlich zu verbessern.

Typische Anti-Muster:

- Interaktiver Zugriff auf Amazon EC2-Instances mit Protokollen wie SSH oder RDP.
- Verwalten einzelner Benutzeranmeldungen wie `/etc/passwd` oder lokale Windows-Benutzer.
- Gemeinsame Nutzung eines Passworts oder privaten Schlüssels für den Zugriff auf eine Instance durch mehrere Benutzer.
- Manuelles Installieren von Software und Erstellen oder Aktualisieren von Konfigurationsdateien.
- Manuelles Aktualisieren oder Patchen von Software.
- Einloggen in eine Instance, um Probleme zu beheben.

Vorteile der Einführung dieser bewährten Methode: Die Durchführung automatisierter Aktionen hilft Ihnen, das betriebliche Risiko unbeabsichtigter Änderungen und Fehlkonfigurationen zu

verringern. Durch das Entfernen von Secure Shell (SSH) und Remote Desktop Protocol (RDP) für den interaktiven Zugriff wird der Umfang des Zugriffs auf Ihre Datenverarbeitungsressourcen reduziert. Damit wird ein gängiger Weg für unbefugte Aktionen abgeschnitten. Die Erfassung Ihrer Aufgaben zur Verwaltung von Datenverarbeitungsressourcen in Automatisierungsdokumenten und programmatischen Skripten bietet einen Mechanismus, mit dem Sie den gesamten Umfang der autorisierten Aktivitäten bis ins kleinste Detail definieren und überprüfen können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Das Protokollieren einer Instance ist eine klassische Methode der Systemverwaltung. Nach der Installation des Server-Betriebssystems würden sich die Benutzer normalerweise manuell anmelden, um das System zu konfigurieren und die gewünschte Software zu installieren. Während der Lebensdauer des Servers melden sich die Benutzer möglicherweise an, um Software-Updates durchzuführen, Patches anzuwenden, Konfigurationen zu ändern und Probleme zu beheben.

Der manuelle Zugriff birgt jedoch eine Reihe von Risiken. Er erfordert einen Server, der auf Anfragen achtet, wie z. B. einen SSH- oder RDP-Service, der einen potenziellen Pfad für unbefugten Zugriff darstellen kann. Außerdem erhöht sich dadurch das Risiko menschlicher Fehler bei der Durchführung manueller Schritte. Diese können zu Störungen des Workloads, zur Beschädigung oder Zerstörung von Daten oder zu anderen Sicherheitsproblemen führen. Der menschliche Zugriff erfordert außerdem Schutzmaßnahmen gegen die Weitergabe von Anmeldeinformationen, was zusätzlichen Verwaltungsaufwand bedeutet.

Um diese Risiken abzuschwächen, können Sie eine agentenbasierte Remotezugriffslösung implementieren, wie z. B. AWS Systems Manager. Der AWS Systems Manager-Agent (SSM Agent) initiiert einen verschlüsselten Kanal und ist daher nicht darauf angewiesen, auf von außen initiierte Anfragen zu achten. Erwägen Sie, SSM Agent so zu konfigurieren, dass er diesen Kanal über einen VPC-Endpunkt aufbaut.

Systems Manager gibt Ihnen eine fein abgestufte Kontrolle darüber, wie Sie mit Ihren verwalteten Instances interagieren können. Sie legen fest, welche Automatisierungen ausgeführt werden sollen, wer sie ausführen darf und wann sie ausgeführt werden können. Systems Manager ist in der Lage, Patches anzuwenden, Software zu installieren und Konfigurationsänderungen ohne interaktiven Zugriff auf die Instance vorzunehmen. Systems Manager kann außerdem den Zugriff auf eine entfernte Shell ermöglichen und jeden während der Sitzung aufgerufenen Befehl und seine Ausgabe in Protokollen und Amazon S3 protokollieren. AWS CloudTrail zeichnet Aufrufe von Systems Manager-APIs zur Überprüfung auf.

Implementierungsschritte

1. Installieren Sie AWS Systems Manager Agent (SSM Agent) auf Ihren Amazon EC2-Instances. Prüfen Sie, ob der SSM-Agent als Teil Ihrer AMI-Basiskonfiguration enthalten ist und automatisch gestartet wird.
2. Überprüfen Sie, ob die IAM-Rollen, die mit Ihren EC2-Instance-Profilen verbunden sind, die verwaltete IAM-Richtlinie `AmazonSSMManagedInstanceCore` enthalten.
3. Deaktivieren Sie SSH, RDP und andere Remotezugriffsservices, die auf Ihren Instances ausgeführt werden. Sie können dies tun, indem Sie Skripte ausführen, die im Abschnitt Benutzerdaten Ihrer Startvorlagen konfiguriert sind, oder indem Sie mit Tools wie EC2 Image Builder angepasste AMIs erstellen.
4. Vergewissern Sie sich, dass die für Ihre EC2-Instances geltenden Ingress-Regeln der Sicherheitsgruppe keinen Zugriff auf Port 22/tcp (SSH) oder Port 3389/tcp (RDP) zulassen. Implementieren Sie die Erkennung und Alarmierung bei falsch konfigurierten Sicherheitsgruppen mit Services wie AWS Config.
5. Definieren Sie entsprechende Automatisierungen, Runbooks und Run Commands in Systems Manager. Verwenden Sie IAM-Richtlinien, um festzulegen, wer diese Aktionen durchführen darf und unter welchen Bedingungen sie erlaubt sind. Testen Sie diese Automatisierungen gründlich in einer nicht produktiven Umgebung. Rufen Sie diese Automatisierungen bei Bedarf auf, anstatt interaktiv auf die Instance zuzugreifen.
6. Verwenden Sie AWS Systems Manager Session Manager, um bei Bedarf interaktiven Zugriff auf Instances zu ermöglichen. Aktivieren Sie die Protokollierung der Sitzungsaktivitäten, um einen Audit Trail zu erstellen, in Amazon CloudWatch Logs oder Amazon S3.

Ressourcen

Zugehörige bewährte Methoden:

- REL08-BP04 Bereitstellung mit einer unveränderlichen Infrastruktur

Zugehörige Beispiele:

- Ersetzen des SSH-Zugriffs zur Reduzierung des Verwaltungs- und Sicherheitsaufwands durch AWS Systems Manager

Zugehörige Tools:

- AWS Systems Manager

Zugehörige Videos:

- Kontrolle des Zugriffs von Benutzersitzungen auf Instances in AWS Systems Manager-Sitzungsmanager

SEC06-BP04 Validieren der Softwareintegrität

Verwenden Sie die kryptografische Überprüfung, um die Integrität von Software-Artefakten (einschließlich Images) zu überprüfen, die Ihr Workload verwendet. Signieren Sie Ihre Software kryptografisch, um sie vor unbefugten Änderungen in Ihren Computerumgebungen zu schützen.

Gewünschtes Ergebnis: Alle Artefakte werden aus vertrauenswürdigen Quellen bezogen. Die Zertifikate der Website des Anbieters sind validiert. Heruntergeladene Artefakte werden anhand ihrer Signaturen kryptographisch verifiziert. Ihre eigene Software ist kryptografisch signiert und wird von Ihren Computerumgebungen überprüft.

Typische Anti-Muster:

- Vertrauen auf die Websites seriöser Anbieter, um Software-Artefakte zu erhalten, aber Hinweise zum Ablauf von Zertifikaten ignorieren Fortfahren mit dem Herunterladen, ohne zu bestätigen, dass die Zertifikate gültig sind
- Validieren der Zertifikate von Anbieter-Websites, aber keine kryptografische Überprüfung der heruntergeladenen Artefakte von diesen Websites
- Prüfen der Integrität von Software ausschließlich anhand von Digests oder Hashes Hashes stellen sicher, dass Artefakte gegenüber der ursprünglichen Version nicht verändert wurden, aber sie bestätigen nicht ihre Quelle.
- Nicht signieren Ihrer eigene Software, Ihres eigenen Codes oder Ihrer eigenen Bibliotheken, selbst wenn Sie sie nur in Ihren eigenen Bereitstellungen verwenden.

Vorteile der Einführung dieser bewährten Methode: Die Überprüfung der Integrität von Artefakten, von denen Ihr Workload abhängt, hilft zu verhindern, dass Malware in Ihre Computerumgebungen eindringt. Das Signieren Ihrer Software schützt Sie davor, dass sie von Unbefugten in Ihrer Computerumgebung ausgeführt wird. Sichern Sie Ihre Softwarelieferkette durch Signieren und Verifizieren von Code.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Betriebssystem-Images, Container-Images und Code-Artefakte werden oft mit verfügbaren Integritätsprüfungen verteilt, z. B. durch einen Digest oder Hash. Diese ermöglichen es den Clients, die Integrität zu überprüfen, indem sie ihren eigenen Hash der Nutzdaten berechnen und überprüfen, ob er mit dem veröffentlichten Hash übereinstimmt. Diese Überprüfungen helfen zwar dabei, sicherzustellen, dass die Nutzdaten nicht manipuliert wurden, aber sie bestätigen nicht, dass die Nutzdaten von der ursprünglichen Quelle (ihrer Herkunft) stammen. Zur Überprüfung der Herkunft ist ein Zertifikat erforderlich, das eine vertrauenswürdige Stelle ausstellt, um das Artefakt digital zu signieren.

Wenn Sie in Ihrem Workload eine heruntergeladene Software oder Artefakte verwenden, prüfen Sie, ob der Anbieter einen öffentlichen Schlüssel für die Überprüfung der digitalen Signatur bereitstellt. Hier sind einige Beispiele dafür, wie AWS einen öffentlichen Schlüssel und Verifizierungsanweisungen für die von uns veröffentlichte Software bereitstellt:

- EC2 Image Builder: Verify the signature of the AWSTOE installation download
- AWS Systems Manager: Verifying the signature of SSM Agent
- Amazon CloudWatch: Verifying the signature of the CloudWatch agent package

Integrieren Sie die Überprüfung digitaler Signaturen in die Prozesse, die Sie zur Beschaffung und Härtung von Images verwenden, wie in SEC06-BP02 Bereitstellen von Datenverarbeitung über gehärtete Images beschrieben.

Sie können AWS Signer verwenden, um die Überprüfung von Signaturen sowie Ihren eigenen Lebenszyklus der Codesignatur für Ihre eigene Software und Artefakte zu verwalten. Sowohl AWS Lambda als auch Amazon Elastic Container Registry bieten Integrationen mit Signer, um die Signaturen Ihres Codes und Ihrer Images zu überprüfen. Mit den Beispielen im Abschnitt Ressourcen können Sie Signer in Ihre Continuous Integration und Delivery (CI/CD) Pipelines einbinden, um die Überprüfung von Signaturen und die Signierung Ihres eigenen Codes und Ihrer Images zu automatisieren.

Ressourcen

Zugehörige Dokumente:

- Cryptographic Signing for Containers

- Best Practices to help secure your container image build pipeline by using AWS Signer
- Announcing Container Image Signing with AWS Signer and Amazon EKS
- Configuring code signing for AWS Lambda
- Best practices and advanced patterns for Lambda code signing
- Code signing using AWS Certificate Manager Private CA and AWS Key Management Service asymmetric keys

Zugehörige Beispiele:

- Automate Lambda code signing with Amazon CodeCatalyst and AWS Signer
- Signing and Validating OCI Artifacts with AWS Signer

Zugehörige Tools:

- AWS Lambda
- AWS Signer
- AWS Certificate Manager
- AWS Key Management Service
- AWS CodeArtifact

SEC06-BP05 Automatisieren des Datenverarbeitungsschutzes

Automatisieren Sie den Datenverarbeitungsschutz, um das Erfordernis menschlichen Eingreifens zu reduzieren. Nutzen Sie automatisierte Scans, um potenzielle Probleme in Ihren Datenverarbeitungsressourcen zu erkennen und mit automatisierten programmatischen Reaktionen oder Flottenmanagement-Vorgängen zu beheben. Integrieren Sie die Automatisierung in Ihre CI/CD-Prozesse, um vertrauenswürdige Workloads mit aktuellen Abhängigkeiten bereitzustellen.

Gewünschtes Ergebnis: Automatisierte Systeme führen alle Scans und Patches von Datenverarbeitungsressourcen durch. Sie verwenden die automatische Überprüfung, um sicherzustellen, dass Software-Images und Abhängigkeiten aus vertrauenswürdigen Quellen stammen und nicht manipuliert wurden. Workloads werden automatisch auf aktuelle Abhängigkeiten geprüft und signiert, um die Vertrauenswürdigkeit in AWS-Datenverabeitungsumgebungen zu gewährleisten. Automatisierte Abhilfemaßnahmen werden eingeleitet, wenn nicht konforme Ressourcen entdeckt werden.

Typische Anti-Muster:

- Verfolgen des Ansatzes einer unveränderlichen Infrastruktur, aber ohne eine Lösung für Notfall-Patches oder den Austausch von Produktionssystemen
- Verwenden von Automatisierung, um falsch konfigurierte Ressourcen zu korrigieren, ohne dass ein manueller Überschreibungsmechanismus vorhanden ist Es können Situationen entstehen, in denen Sie die Anforderungen anpassen müssen, und es kann sein, dass Sie die Automatisierungen aussetzen müssen, bis Sie diese Änderungen vorgenommen haben.

Vorteile der Einführung dieser bewährten Methode: Die Automatisierung kann das Risiko des unbefugten Zugriffs und der Nutzung Ihrer Datenverarbeitungsressourcen verringern. Sie hilft zu verhindern, dass Fehlkonfigurationen in Produktionsumgebungen gelangen, und Fehlkonfigurationen zu erkennen und zu beheben, wenn sie auftreten. Die Automatisierung hilft auch bei der Erkennung von unbefugtem Zugriff und der Nutzung von Datenverarbeitungsressourcen, um Ihre Reaktionszeit zu verkürzen. Dies wiederum kann den Gesamtumfang der Auswirkungen des Problems verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Sie können die in den Methoden der Sicherheitssäule beschriebenen Automatisierungen zum Schutz Ihrer Datenverarbeitungsressourcen anwenden. SEC06-BP01 Schwachstellenmanagement beschreibt, wie Sie Amazon Inspector sowohl in Ihren CI/CD-Pipelines als auch für die kontinuierliche Überprüfung Ihrer Laufzeitumgebungen auf bekannte CVEs (Common Vulnerabilities and Exposures) einsetzen können. Sie können AWS Systems Manager verwenden, um Patches anzuwenden oder neue Images über automatisierte Runbooks bereitzustellen, damit Ihre Computerflotte stets mit der neuesten Software und den neuesten Bibliotheken ausgestattet ist. Nutzen Sie diese Techniken, um den Bedarf an manuellen Prozessen und interaktivem Zugriff auf Ihre Datenverarbeitungsressourcen zu reduzieren. Siehe SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs, um mehr zu erfahren.

Die Automatisierung spielt auch eine Rolle bei der Bereitstellung von Workloads, die vertrauenswürdig sind. Dies wird in SEC06-BP02 Bereitstellen von Datenverarbeitung über gehärtete Images und SEC06-BP04 Validieren der Softwareintegrität beschrieben. Sie können Services wie EC2 Image Builder, AWS Signer, AWS CodeArtifact, und Amazon Elastic Container Registry (ECR) verwenden, um gehärtete und genehmigte Images und Code-Abhängigkeiten herunterzuladen, zu überprüfen, zu erstellen und zu speichern. Neben Inspector kann jeder von ihnen eine Rolle in Ihrem CI/CD-Prozess spielen, sodass Ihr Workload nur dann in die Produktion geht, wenn sichergestellt ist,

dass seine Abhängigkeiten aktuell sind und aus vertrauenswürdigen Quellen stammen. Ihr Workload ist außerdem signiert, damit AWS-Datenverarbeitungsumgebungen wie AWS Lambda und Amazon Elastic Kubernetes Service (EKS) überprüfen können, dass er nicht manipuliert wurde, bevor sie ihn ausführen.

Über diese präventiven Kontrollen hinaus können Sie die Automatisierung auch bei den detektivischen Kontrollen für Ihre Datenverarbeitungsressourcen einsetzen. Ein Beispiel: AWS Security Hub bietet den Standard NIST 800-53 Rev. 5, der Prüfungen wie [EC2.8] EC2-Instances should use Instance Metadata Service Version 2 (IMDSv2) enthält. IMDSv2 verwendet die Techniken der Sitzungsauthentifizierung, des Blockierens von Anfragen, die einen X-Forwarded-For HTTP-Header enthalten, und eine Netzwerk-TTL von 1, um den von externen Quellen stammenden Datenverkehr zum Abrufen von Informationen über die EC2-Instance zu stoppen. Diese Prüfung in Security Hub kann erkennen, wenn EC2 Instances IMDSv1 verwenden und eine automatische Abhilfe einleiten. Erfahren Sie mehr über automatische Erkennung und Abhilfemaßnahmen in SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen.

Implementierungsschritte

1. Automatisieren Sie die Erstellung sicherer, konformer und gehärteter AMIs mit EC2 Image Builder. Sie können Images erstellen, die Kontrollen aus den Center for Internet Security (CIS)-Benchmarks oder Security Technical Implementation Guide (STIG)-Standards aus Basis- AWS und APN-Partner-Images enthalten.
2. Automatische Konfigurationsverwaltung. Erzwingen und validieren Sie sichere Konfigurationen in Ihren Datenverarbeitungsressourcen automatisch. Verwenden Sie dazu einen Service oder ein Tool zur Konfigurationsverwaltung.
   a. Automatisiertes Konfigurationsmanagement mit AWS Config
   b. Automatisiertes Sicherheits- und Compliance-Management mit AWS Security Hub
3. Automatisieren Sie das Patchen oder Ersetzen von Amazon Elastic Compute Cloud (Amazon EC2)-Instances. AWS Systems Manager Patch Manager automatisiert das Patchen verwalteter Instances mit sicherheitsrelevanten und anderen Arten von Updates. Sie können Patch Manager verwenden, um Patches für Betriebssysteme und Anwendungen anzuwenden.
   a. AWS Systems Manager Incident Manager
4. Automatisieren Sie das Scannen von Datenverarbeitungsressourcen auf häufige Schwachstellen und Gefährdungen (CVEs) und betten Sie Sicherheitsscan-Lösungen in Ihre Build-Pipeline ein.
   a. Amazon Inspector
   b. ECR Image Scanning

5. Ziehen Sie Amazon GuardDuty für die automatische Erkennung von Malware und Bedrohungen in Betracht, um Datenverarbeitungsressourcen zu schützen. GuardDuty kann außerdem mögliche Probleme identifizieren, wenn eine AWS Lambda-Funktion in Ihrer AWS-Umgebung aufgerufen wird.
   a. Amazon GuardDuty
6. Ziehen Sie AWS-Partnerlösungen in Betracht. AWS-Partner bieten branchenführende Produkte an, die mit vorhandenen Kontrollen in Ihren lokalen Umgebungen gleichwertig oder identisch sind oder sich in diese integrieren lassen. Diese Produkte ergänzen die vorhandenen AWS-Services, sodass Sie eine umfassende Sicherheitsarchitektur bereitstellen und eine nahtlosere Erfahrung in Ihren Cloud- und On-Premises-Umgebungen ermöglichen können.
   a. Sicherheit der Infrastruktur

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen

Zugehörige Dokumente:

- Get the full benefits of IMDSv2 and disable IMDSv1 across your AWS infrastructure

Zugehörige Videos:

- Security best practices for the Amazon EC2 instance metadata service

Datenschutz

Fragen

- SEC 7. Wie klassifizieren Sie Ihre Daten?
- SEC 8. Wie schützen Sie Ihre ruhenden Daten?
- SEC 9. Wie schützen Sie Ihre Daten bei der Übertragung?

SEC 7. Wie klassifizieren Sie Ihre Daten?

Die Datenklassifizierung bietet eine Möglichkeit, Daten basierend auf Wichtigkeit und Sensibilität zu kategorisieren, um Ihnen dabei zu helfen, angemessene Schutz- und Aufbewahrungskontrollen zu bestimmen.

Bewährte Methoden

- SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung
- SEC07-BP02 Anwenden von Datenschutzkontrollen basierend auf der Sensibilität der Daten
- SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung
- SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements

SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung

Machen Sie sich ein Bild von der Klassifizierung der Daten, die Ihr Workload verarbeitet, den Anforderungen an die Verarbeitung, den damit verbundenen Geschäftsprozessen, dem Ort, an dem die Daten gespeichert sind, sowie dem Eigentümer der Daten. Ihr Schema für die Klassifizierung und den Umgang mit Daten sollte die geltenden rechtlichen und Compliance-Anforderungen Ihres Workloads und die erforderlichen Datenkontrollen berücksichtigen. Das Verständnis der Daten ist der erste Schritt zur Datenklassifizierung.

Gewünschtes Ergebnis: Die in Ihrem Workload vorhandenen Datentypen sind gut verstanden und dokumentiert. Es gibt angemessene Kontrollen zum Schutz sensibler Daten auf der Grundlage ihrer Klassifizierung. Diese Kontrollen regeln z. B., wer auf die Daten zugreifen darf und zu welchem Zweck, wo die Daten gespeichert werden, die Verschlüsselungsrichtlinie für diese Daten und wie Verschlüsselungsschlüssel verwaltet werden, den Lebenszyklus der Daten und die Anforderungen an die Aufbewahrung, angemessene Vernichtungsprozesse, welche Sicherungs- und Wiederherstellungsprozesse vorhanden sind und die Überprüfung des Zugriffs.

Typische Anti-Muster:

- Fehlen einer formalen Richtlinie zur Datenklassifizierung, um die Sensibilitätsebenen und die Anforderungen an die Handhabung von Daten zu definieren
- Mangel an Wissen über die Sensibilitätsebenen der Daten innerhalb Ihres Workloads und fehlende Erfassung dieser Informationen in der Architektur- und Betriebsdokumentation

- Versäumnis, angemessene Kontrollen für Ihre Daten anzuwenden, die auf deren Sensibilität und Anforderungen basieren, wie in Ihrer Richtlinie zur Datenklassifizierung und -verarbeitung festgelegt
- Unterlassen von Feedback über die Anforderungen an die Datenklassifizierung und -verarbeitung an die Eigentümer der Richtlinien

Vorteile der Einführung dieser bewährten Methode: Diese Vorgehensweise beseitigt Unklarheiten über den angemessenen Umgang mit Daten innerhalb Ihres Workloads. Die Anwendung einer formellen Richtlinie, die die Sensibilitätsebenen der Daten in Ihrer Organisation und die erforderlichen Schutzmaßnahmen definiert, kann Ihnen helfen, gesetzliche Vorschriften und andere Bescheinigungen und Zertifizierungen im Bereich der Cybersicherheit einzuhalten. Besitzer von Workloads können sich darauf verlassen, dass sie wissen, wo sensible Daten gespeichert sind und welche Schutzkontrollen vorhanden sind. Wenn Sie diese in der Dokumentation festhalten, können neue Team-Mitglieder sie besser verstehen und schon früh in ihrer Amtszeit Kontrollen durchführen. Diese Praktiken können auch dazu beitragen, die Kosten zu senken, indem die Kontrollen für jede Art von Daten richtig dimensioniert werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Wenn Sie einen Workload entwerfen, überlegen Sie vielleicht intuitiv, wie Sie sensible Daten schützen können. Bei einer mandantenfähigen Anwendung ist es beispielsweise intuitiv, die Daten jedes Mandanten als sensibel zu betrachten und Schutzmaßnahmen zu ergreifen, damit ein Mandant nicht auf die Daten eines anderen Mandanten zugreifen kann. Ebenso können Sie intuitiv Zugriffskontrollen so gestalten, dass nur Administratoren Daten ändern können, während andere Benutzer nur Lesezugriff oder gar keinen Zugriff haben.

Indem Sie diese Datensensibilitätsebenen zusammen mit den entsprechenden Datenschutzanforderungen definieren und in Richtlinien festhalten, können Sie formell feststellen, welche Daten sich in Ihrem Workload befinden. Sie können dann feststellen, ob die richtigen Kontrollen vorhanden sind, ob die Kontrollen überprüft werden können und welche Reaktionen angemessen sind, wenn ein falscher Umgang mit Daten festgestellt wird.

Um die Kategorisierung von sensiblen Daten innerhalb Ihres Workloads zu erleichtern, sollten Sie, sofern verfügbar, Ressourcen-Tags verwenden. Sie können zum Beispiel ein Tag mit dem Tag-Schlüssel `Klassifizierung` und dem Tag-Wert `PHI` für geschützte Gesundheitsinformationen (Protected Health Information, PHI) und ein weiteres Tag mit dem Tag-Schlüssel `Sensibilität`

und dem Tag-Wert `Hoch` verwenden. Mit Services wie AWS Config können Sie diese Ressourcen auf Änderungen überwachen und eine Warnung ausgeben, wenn sie so verändert werden, dass sie Ihren Schutzanforderungen nicht mehr genügen (z. B. durch Änderung der Verschlüsselungseinstellungen). Sie können die Standarddefinition Ihrer Tag-Schlüssel und zulässigen Werte mit Tag-Richtlinien, einer Funktion von AWS Organizations, erfassen. Es wird nicht empfohlen, dass der Tag-Schlüssel oder -Wert private oder sensible Daten enthält.

Implementierungsschritte

1. Verstehen Sie das Datenklassifizierungsschema und die Schutzanforderungen Ihrer Organisation.
2. Identifizieren Sie die Arten von sensiblen Daten, die von Ihren Workloads verarbeitet werden.
3. Vergewissern Sie sich, dass sensible Daten in Ihrem Workload gemäß Ihrer Richtlinie gespeichert und geschützt werden. Nutzen Sie Techniken wie automatisierte Tests, um die Wirksamkeit Ihrer Kontrollen zu überprüfen.
4. Erwägen Sie die Verwendung von Markierungen auf Ressourcen- und Datenebene, sofern verfügbar, um Daten mit ihrer Sensibilitätsstufe und anderen operativen Metadaten zu versehen, die bei der Überwachung und der Reaktion auf Vorfälle helfen können.
   a. AWS Organizations-Tag-Richtlinien können verwendet werden, um Tagging-Standards durchzusetzen.

Ressourcen

Zugehörige bewährte Methoden:

- SUS04-BP01 Implementieren einer Richtlinie für die Klassifizierung von Daten

Zugehörige Dokumente:

- Data Classification whitepaper
- Best Practices for Tagging AWS Resources

Zugehörige Beispiele:

- AWS Organizations Tag Policy Syntax and Examples

Zugehörige Tools

- AWS-Tag-Editor

SEC07-BP02 Anwenden von Datenschutzkontrollen basierend auf der Sensibilität der Daten

Wenden Sie Datenschutzkontrollen an, die ein angemessenes Maß an Kontrolle für jede in Ihrer Klassifizierungsrichtlinie definierte Datenklasse bieten. Auf diese Weise können Sie sensible Daten vor unbefugtem Zugriff und unbefugter Nutzung schützen und gleichzeitig die Verfügbarkeit und Nutzung der Daten aufrechterhalten.

Gewünschtes Ergebnis: Sie verfügen über eine Klassifizierungsrichtlinie, die die verschiedenen Sensibilitätsstufen für Daten in Ihrer Organisation definiert. Für jede dieser Sensibilitätsebenen haben Sie klare Richtlinien für zugelassene Speicher- und Bearbeitungsservices und -orte sowie deren erforderliche Konfiguration veröffentlicht. Sie implementieren die Kontrollen für jede Ebene entsprechend dem erforderlichen Schutzniveau und den damit verbundenen Kosten. Sie verfügen über Überwachungs- und Warnsysteme, um zu erkennen, wenn sich Daten an nicht autorisierten Orten befinden, in nicht autorisierten Umgebungen verarbeitet werden, nicht autorisierte Akteure darauf zugreifen oder die Konfiguration der zugehörigen Services nicht mehr konform ist.

Typische Anti-Muster:

- Anwenden des gleichen Maßes an Schutzkontrollen für alle Daten Dies kann dazu führen, dass zu viele Sicherheitskontrollen für wenig sensible Daten bereitgestellt werden oder hochsensible Daten nicht ausreichend geschützt werden.
- Unterlassen, die relevanten Stakeholder aus Sicherheits-, Compliance- und Geschäftsteams bei der Definition von Datenschutzkontrollen einzubeziehen
- Vernachlässigen des betrieblichen Aufwands und der Kosten, die mit der Implementierung und Pflege von Datenschutzkontrollen verbunden sind
- Fehlen von regelmäßigen Überprüfungen der Datenschutzkontrollen, um die Übereinstimmung mit den Klassifizierungsrichtlinien zu gewährleisten

Vorteile der Einführung dieser bewährten Methode: Indem Sie Ihre Kontrollen auf die Klassifizierungsstufe Ihrer Daten abstimmen, kann Ihre Organisation bei Bedarf in höhere Kontrollstufen investieren. Dies kann eine Aufstockung der Ressourcen für die Sicherung, Überwachung, Messung, Behebung und Berichterstattung beinhalten. Wo weniger Kontrollen angebracht sind, können Sie die Zugänglichkeit und Vollständigkeit der Daten für Ihre Mitarbeiter, Kunden oder Wähler verbessern. Dieser Ansatz bietet Ihrer Organisation die größtmögliche

Flexibilität bei der Datennutzung, während gleichzeitig die Datenschutzanforderungen eingehalten werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Die Implementierung von Datenschutzkontrollen auf der Grundlage von Datensensibilitätsebenen umfasst mehrere wichtige Schritte. Ermitteln Sie zunächst die verschiedenen Datensensibilitätsebenen innerhalb Ihrer Workload-Architektur (z. B. öffentlich, intern, vertraulich und eingeschränkt) und bewerten Sie, wo Sie diese Daten speichern und verarbeiten. Als Nächstes definieren Sie Isolationsgrenzen um die Daten herum, basierend auf ihrer Sensibilitätsebene. Wir empfehlen Ihnen, Daten in verschiedene AWS-Konten zu unterteilen und Service-Kontrollrichtlinien(SCPs) zu verwenden, um die für die einzelnen Sensibilitätsebenen zulässigen Services und Aktionen einzuschränken. Auf diese Weise können Sie starke Isolationsgrenzen schaffen und das Prinzip der geringsten Berechtigung durchsetzen.

Nachdem Sie die Isolationsgrenzen definiert haben, implementieren Sie geeignete Schutzkontrollen auf der Grundlage der Sensibilitätsebenen der Daten. Beachten Sie die bewährten Methoden zum Schutz von Daten im Ruhezustand und zum Schutz von Daten während der Übertragung , um entsprechende Kontrollen wie Verschlüsselung, Zugriffskontrollen und Audits zu implementieren. Ziehen Sie Techniken wie Tokenisierung oder Anonymisierung in Betracht, um die Sensibilität Ihrer Daten zu verringern. Vereinfachen Sie die Anwendung konsistenter Datenrichtlinien in Ihrem Unternehmen mit einem zentralisierten System für Tokenisierung und De-Tokenisierung.

Überwachen und testen Sie fortlaufend die Wirksamkeit der implementierten Kontrollen. Überprüfen und aktualisieren Sie das Datenklassifizierungsschema, die Risikobewertungen und die Schutzkontrollen regelmäßig, wenn sich die Datenlandschaft und die Bedrohungen in Ihrer Organisation weiterentwickeln. Richten Sie die implementierten Datenschutzkontrollen an den einschlägigen Branchenvorschriften, Standards und gesetzlichen Anforderungen aus. Sorgen Sie außerdem für ein Sicherheitsbewusstsein und bieten Sie Schulungen an, damit die Mitarbeiter das Datenklassifizierungsschema und ihre Verantwortung im Umgang mit sensiblen Daten und deren Schutz verstehen.

Implementierungsschritte

1. Identifizieren Sie die Klassifizierungs- und Sensibilitätsstufen der Daten innerhalb Ihres Workloads.
2. Definieren Sie Isolationsgrenzen für jede Ebene und legen Sie eine Durchsetzungsstrategie fest.

3. Bewerten Sie die von Ihnen definierten Kontrollen, die den Zugriff, die Verschlüsselung, die Prüfung, die Aufbewahrung und andere von Ihrer Datenklassifizierungsrichtlinie geforderte Punkte regeln.
4. Prüfen Sie gegebenenfalls Optionen zur Verringerung der Sensibilität der Daten, z. B. durch Tokenisierung oder Anonymisierung.
5. Überprüfen Sie Ihre Kontrollen durch automatische Tests und die Überwachung Ihrer konfigurierten Ressourcen.

Ressourcen

Zugehörige bewährte Methoden:

- PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt
- COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung

Zugehörige Dokumente:

- Data Classification whitepaper
- Best Practices for Security, Identify, & Compliance
- AWS KMS Best Practices
- Encryption best practices and features for AWS services

Zugehörige Beispiele:

- Building a serverless tokenization solution to mask sensitive data
- How to use tokenization to improve data security and reduce audit scope

Zugehörige Tools:

- AWS Key Management Service (AWS KMS)
- AWS CloudHSM
- AWS Organizations

SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung

Durch die Automatisierung der Identifizierung und Klassifizierung von Daten können Sie die richtigen Kontrollen implementieren. Der Einsatz von Automatisierung als Ergänzung zur manuellen Ermittlung verringert das Risiko menschlicher Fehler und das Risiko einer Gefährdung.

Gewünschtes Ergebnis: Sie sind in der Lage zu überprüfen, ob die richtigen Kontrollen auf der Grundlage Ihrer Klassifizierungs- und Bearbeitungsrichtlinien vorhanden sind. Automatisierte Tools und Services helfen Ihnen bei der Identifizierung und Klassifizierung der Sensibilitätsebene Ihrer Daten. Die Automatisierung hilft Ihnen auch bei der kontinuierlichen Überwachung Ihrer Umgebungen, um zu erkennen und zu melden, wenn Daten auf unzulässige Weise gespeichert oder verarbeitet werden, sodass schnell Abhilfemaßnahmen ergriffen werden können.

Typische Anti-Muster:

- Vertrauen auf ausschließlich manuelle Prozesse, die fehleranfällig und zeitaufwendig sein können, um Daten zu identifizieren und zu klassifizieren Dies kann zu einer ineffizienten und inkonsistenten Datenklassifizierung führen, insbesondere wenn das Datenvolumen wächst.
- Fehlen von Mechanismen zur Verfolgung und Verwaltung von Datenbeständen in der gesamten Organisation
- Vernachlässigen der Notwendigkeit einer kontinuierlichen Überwachung und Klassifizierung von Daten, während sie sich innerhalb der Organisation bewegen und weiterentwickeln

Vorteile der Einführung dieser bewährten Methode: Die Automatisierung der Identifizierung und Klassifizierung von Daten kann zu einer konsistenteren und präziseren Anwendung von Datenschutzkontrollen führen und das Risiko menschlicher Fehler verringern. Die Automatisierung kann auch den Zugriff auf und die Bewegung von sensiblen Daten transparent machen, sodass Sie unautorisierten Umgang mit diesen Daten erkennen und Korrekturmaßnahmen ergreifen können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Auch wenn die Klassifizierung von Daten in den ersten Entwurfsphasen eines Workloads häufig nach menschlichem Ermessen erfolgt, sollten Sie zur Vorbeugung Systeme einsetzen, die die Identifizierung und Klassifizierung von Testdaten automatisieren. Beispielsweise können Entwickler ein Tool oder einen Dienst erhalten, um repräsentative Daten zu scannen und ihre Sensibilität zu bestimmen. Innerhalb von AWS können Sie Datensätze in Amazon S3 hochladen und sie

unter Verwendung von Amazon Macie, Amazon Comprehend oder Amazon Comprehend Medical scannen. Ziehen Sie auch in Betracht, Daten im Rahmen von Modultests und Integrationstests zu scannen, um festzustellen, wo sensible Daten nicht erwartet werden. Eine Warnung vor sensiblen Daten in dieser Phase kann vor der Bereitstellung in der Produktion auf Schutzlücken hinweisen. Andere Funktionen wie die Erkennung sensibler Daten in AWS Glue, Amazon SNS und Amazon CloudWatch können ebenfalls verwendet werden, um PII zu erkennen und geeignete Abhilfemaßnahmen zu ergreifen. Verstehen Sie bei jedem automatisierten Tool oder Dienst, wie es sensible Daten definiert, und ergänzen Sie es mit anderen menschlichen oder automatisierten Lösungen, um eventuelle Lücken zu schließen.

Nutzen Sie die kontinuierliche Überwachung Ihrer Umgebungen als detektivische Kontrolle, um festzustellen, ob sensible Daten auf nicht konforme Weise gespeichert werden. Dies kann dazu beitragen, Situationen zu erkennen, in denen sensible Daten ohne ordnungsgemäße De-Identifizierung oder Schwärzung in Protokolldateien ausgegeben oder in eine Datenanalyseumgebung kopiert werden. Daten, die in Amazon S3 gespeichert sind, können mit Amazon Macie kontinuierlich auf sensible Daten überwacht werden.

Implementierungsschritte

1. Führen Sie einen ersten Scan Ihrer Umgebungen zur automatischen Identifizierung und Klassifizierung durch.
   a. Ein erster vollständiger Scan Ihrer Daten kann dazu beitragen, ein umfassendes Verständnis darüber zu erlangen, wo sich sensible Daten in Ihren Umgebungen befinden. Wenn ein vollständiger Scan nicht erforderlich ist oder aus Kostengründen nicht im Voraus durchgeführt werden kann, sollten Sie prüfen, ob Stichprobenverfahren geeignet sind, um Ihre Ziele zu erreichen. Zum Beispiel kann Amazon Macie so konfiguriert werden, dass eine umfassende automatische Erkennung sensibler Daten in Ihren S3 Buckets durchgeführt wird. Diese Funktion nutzt Stichprobenverfahren, um kosteneffizient eine Vorabanalyse darüber durchzuführen, wo sensible Daten gespeichert sind. Eine tiefergehende Analyse von S3 Buckets kann dann mit einem Auftrag zur Erkennung sensibler Daten durchgeführt werden. Auch andere Datenspeicher können in S3 exportiert werden, um von Macie durchsucht zu werden.
2. Konfigurieren Sie laufende Scans Ihrer Umgebungen.
   a. Die automatische Erkennungsfunktion für sensible Daten von Macie kann für laufende Scans Ihrer Umgebungen verwendet werden. Bekannte S3 Buckets, die für die Speicherung sensibler Daten autorisiert sind, können mit einer Zulassen-Liste in Macie ausgeschlossen werden.
3. Integrieren Sie die Identifizierung und Klassifizierung in Ihre Build- und Testprozesse.

   a. Identifizieren Sie Tools, mit denen Entwickler Daten auf Sensibilität prüfen können, während Workloads entwickelt werden. Verwenden Sie diese Tools als Teil der Integrationstests, um bei unerwarteten sensiblen Daten Alarm zu schlagen und eine weitere Bereitstellung zu verhindern.
4. Implementieren Sie ein System oder Runbook, um Maßnahmen zu ergreifen, wenn sensible Daten an nicht autorisierten Orten gefunden werden.

Ressourcen

Zugehörige Dokumente:

- AWS Glue: Detect and process sensitive data
- Using managed data identifiers in Amazon SNS
- Amazon CloudWatch Logs: Help protect sensitive log data with masking

Zugehörige Beispiele:

- Enabling data classification for Amazon RDS database with Macie
- Detecting sensitive data in DynamoDB with Macie

Zugehörige Tools:

- Amazon Macie
- Amazon Comprehend
- Amazon Comprehend Medical
- AWS Glue

SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements

Machen Sie sich mit den Anforderungen an den Lebenszyklus Ihrer Daten in Bezug auf die verschiedenen Ebenen der Datenklassifizierung und -verarbeitung vertraut. Dazu kann gehören, wie Daten behandelt werden, wenn sie zum ersten Mal in Ihre Umgebung gelangen, wie Daten umgewandelt werden und welche Regeln für ihre Vernichtung gelten. Berücksichtigen Sie Faktoren wie Aufbewahrungsfristen, Zugriff, Prüfung und Nachvollziehbarkeit der Herkunft.

Gewünschtes Ergebnis: Sie klassifizieren die Daten so nah wie möglich an dem Punkt und dem Zeitpunkt der Datenerfassung. Wenn die Klassifizierung von Daten eine Maskierung, Tokenisierung

oder andere Prozesse zur Verringerung der Sensibilitätsebene erfordert, führen Sie diese Aktionen so nah wie möglich am Zeitpunkt der Datenerfassung durch.

Sie löschen Daten in Übereinstimmung mit Ihrer Richtlinie, wenn sie aufgrund ihrer Klassifizierung nicht mehr aufbewahrt werden sollten.

Typische Anti-Muster:

- Implementieren eines Einheitsansatzes für die Verwaltung des Lebenszyklus von Daten, ohne Berücksichtigung unterschiedlicher Sensibilitätsebenen und Zugriffsanforderungen
- Beschränken der Betrachtung des Lebenszyklusmanagements auf entweder nutzbare Daten oder gesicherte Daten, statt auf beide
- Annehmen, dass Daten, die in Ihren Workload eingegeben wurden, gültig sind, ohne ihren Wert oder ihre Herkunft zu ermitteln
- Vertrauen auf die Haltbarkeit von Daten als Ersatz für Datensicherungen und -schutz
- Beibehalten von Daten über ihre Nützlichkeit und die erforderliche Aufbewahrungsfrist hinaus

Vorteile der Einführung dieser bewährten Methode: Eine gut definierte und skalierbare Strategie für die Verwaltung des Lebenszyklus von Daten hilft bei der Einhaltung gesetzlicher Vorschriften, verbessert die Datensicherheit, optimiert die Speicherkosten und ermöglicht einen effizienten Datenzugriff und -austausch unter Beibehaltung angemessener Kontrollen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Daten innerhalb eines Workloads sind oft dynamisch. Die Form, in der die Daten in Ihre Workload-Umgebung gelangen, kann sich von der Form unterscheiden, in der sie gespeichert oder in der Geschäftslogik, der Berichterstattung, der Analyse oder dem Machine Learning verwendet werden. Außerdem kann sich der Wert der Daten im Laufe der Zeit ändern. Einige Daten sind zeitlich begrenzt und verlieren an Wert, wenn sie älter werden. Überlegen Sie, wie sich diese Änderungen an Ihren Daten auf die Bewertung nach Ihrem Datenklassifizierungsschema und die damit verbundenen Kontrollen auswirken. Verwenden Sie nach Möglichkeit einen automatisierten Lebenszyklus-Mechanismus wie Amazon S3-Lebenszyklus-Richtlinien und Amazon Data Lifecycle Manager, um Ihre Datenaufbewahrung, Archivierung und Ablaufprozesse zu konfigurieren.

Unterscheiden Sie zwischen Daten, die zur Verwendung zur Verfügung stehen, und Daten, die als Backup gespeichert sind. Ziehen Sie die Verwendung von AWS Backup in Betracht, um die

Sicherung von Daten über AWS-Services hinweg zu automatisieren. Amazon EBS-Snapshots bieten eine Möglichkeit, ein EBS-Volume zu kopieren und es unter Verwendung von S3-Features zu speichern, einschließlich Lebenszyklus, Datenschutz und Zugriff auf Schutzmechanismen. Zwei dieser Mechanismen sind S3 Object Lock und AWS Backup Vault Lock, die Ihnen zusätzliche Sicherheit und Kontrolle über Ihre Backups bieten können. Verwalten Sie eine klare Aufgabentrennung und Zugriffsrechte für Backups. Isolieren Sie Backups auf Kontoebene, um während eines Ereignisses eine Trennung von der betroffenen Umgebung zu gewährleisten.

Ein weiterer Aspekt des Lifecycle-Managements ist die Aufzeichnung des Datenverlaufs, während diese Ihren Workload durchlaufen. Dies wird als Nachverfolgung der Datenherkunft bezeichnet. Dadurch können Sie sicher sein, dass Sie wissen, woher die Daten stammen, welche Transformationen durchgeführt wurden, welcher Eigentümer oder Prozess diese Änderungen vorgenommen hat und wann. Dieser Verlauf hilft bei der Fehlersuche und bei der Untersuchung möglicher Sicherheitsvorfälle. Sie können zum Beispiel Metadaten über Transformationen in einer Amazon DynamoDB-Tabelle protokollieren. Innerhalb eines Data Lake können Sie Kopien der transformierten Daten in verschiedenen S3-Buckets für jede Stufe der Datenpipeline aufbewahren. Speichern Sie Schema- und Zeitstempelinformationen in einem AWS Glue Data Catalog. Unabhängig von Ihrer Lösung sollten Sie die Anforderungen Ihrer Endbenutzer berücksichtigen, um die geeigneten Tools für die Berichterstattung über die Herkunft Ihrer Daten zu bestimmen. So können Sie feststellen, wie Sie Ihre Herkunft am besten verfolgen können.

Implementierungsschritte

1. Analysieren Sie die Datentypen, Sensibilitätsebenen und Zugriffsanforderungen des Workloads, um die Daten zu klassifizieren und geeignete Strategien für das Lebenszyklusmanagement zu definieren.
2. Entwerfen und implementieren Sie Richtlinien für die Datenaufbewahrung und automatisierte Vernichtungsprozesse, die mit den rechtlichen, regulatorischen und organisatorischen Anforderungen übereinstimmen.
3. Etablieren Sie Prozesse und Automatisierungen für die kontinuierliche Überwachung, Prüfung und Anpassung von Strategien, Kontrollen und Richtlinien für die Verwaltung des Datenlebenszyklus, wenn sich die Anforderungen an den Workload und die Vorschriften weiterentwickeln.

Ressourcen

Zugehörige bewährte Methoden:

- COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung

- SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien

Zugehörige Dokumente:

- Data Classification Whitepaper
- AWS Blueprint for Ransomware Defense
- DevOps Guidance: Improve traceability with data provenance tracking

Zugehörige Beispiele:

- How to protect sensitive data for its entire lifecycle in AWS
- Build data lineage for data lakes using AWS Glue, Amazon Neptune, and Spline

Zugehörige Tools:

- AWS Backup
- Amazon Data Lifecycle Manager
- AWS Identity and Access Management Access Analyzer

SEC 8. Wie schützen Sie Ihre ruhenden Daten?

Schützen Sie Ihre Daten im Ruhezustand, indem Sie mehrere Kontrollen implementieren, um das Risiko eines unbefugten Zugriffs oder eines Missbrauchs zu reduzieren.

Bewährte Methoden

- SEC08-BP01: Implementieren einer sicheren Schlüsselverwaltung
- SEC08-BP02 Erzwingen der Verschlüsselung im Ruhezustand
- SEC08-BP03 Automatisieren des Schutzes von Daten im Ruhezustand
- SEC08-BP04 Durchsetzen der Zugriffskontrolle

SEC08-BP01: Implementieren einer sicheren Schlüsselverwaltung

Eine sichere Schlüsselverwaltung umfasst die Speicherung, Rotation, Zugriffskontrolle und Überwachung von Schlüsseldaten, die zur Sicherung von Daten im Ruhezustand für Ihre Workloads erforderlich sind.

Gewünschtes Ergebnis: Ein skalierbarer, wiederholbarer und automatisierter Schlüsselverwaltungsmechanismus. Der Mechanismus sollte die Möglichkeit bieten, den Zugriff mit den geringsten Berechtigungen auf Schlüsseldaten zu erzwingen, und das richtige Gleichgewicht zwischen Schlüsselverfügbarkeit, Vertraulichkeit und Integrität bieten. Der Zugriff auf Schlüssel sollte überwacht werden und Schlüsseldaten sollten mit einem automatisierten Prozess rotiert werden. Schlüsseldaten sollten niemals für menschliche Identitäten zugänglich sein.

Typische Anti-Muster:

- Personen haben Zugriff auf unverschlüsselte Schlüsseldaten.
- Es werden benutzerdefinierte kryptografische Algorithmen erstellt.
- Die Berechtigungen für den Zugriff auf Schlüsseldaten sind zu weit gefasst.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie einen sicheren Mechanismus für die Schlüsselverwaltung für Ihren Workload einrichten, können Sie dazu beitragen, Ihre Inhalte vor unbefugtem Zugriff zu schützen. Darüber hinaus gelten möglicherweise gesetzliche Anforderungen zur Verschlüsselung Ihrer Daten. Eine effektive Schlüsselverwaltungslösung kann technische Mechanismen bereitstellen, die diesen Vorschriften zum Schutz von Schlüsseldaten entsprechen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Viele regulatorische Anforderungen und bewährte Methoden beinhalten die Verschlüsselung von Daten im Ruhezustand als grundlegende Sicherheitskontrolle. Um diese Bedingung zu erfüllen, benötigt Ihr Workload einen Mechanismus, mit dem Schlüsseldaten, die zur Verschlüsselung Ihrer Daten im Ruhezustand verwendet werden, sicher gespeichert und verwaltet werden können.

AWS bietet AWS Key Management Service (AWS KMS) zur dauerhaften, sicheren und redundanten Speicherung von AWS KMS-Schlüsseln. Viele AWS-Services lassen sich in AWS KMS integrieren, um die Verschlüsselung Ihrer Daten zu unterstützen. AWS KMS verwendet FIPS 140-2 Level 3-validierte Hardware-Sicherheitsmodule zum Schutz Ihrer Schlüssel. Es gibt keinen Mechanismus zum Exportieren von AWS KMS-Schlüsseln als Klartext.

Bei der Bereitstellung von Workloads mit einer Strategie für mehrere Konten gilt es als bewährte Methode, AWS KMS-Schlüssel im selben Konto zu speichern wie der Workload, der sie verwendet. In diesem verteilten Modell liegt die Verantwortung für die Verwaltung der AWS KMS-Schlüssel beim Anwendungsteam. In anderen Anwendungsfällen können sich Unternehmen dafür entscheiden,

AWS KMS-Schlüssel in einem zentralen Konto zu speichern. Diese zentralisierte Struktur erfordert zusätzliche Richtlinien, um den kontoübergreifenden Zugriff zu ermöglichen, der benötigt wird, damit das Workload-Konto auf Schlüssel zugreifen kann, die im zentralen Konto gespeichert sind. Dieses Verfahren kann jedoch in Anwendungsfällen, in denen ein einzelner Schlüssel von mehreren AWS-Konten gemeinsam genutzt wird, besser geeignet sein.

Unabhängig davon, wo die Schlüsseldaten gespeichert werden, sollte der Zugriff auf den Schlüssel durch Schlüsselrichtlinien und IAM-Richtlinien streng kontrolliert werden. Schlüsselrichtlinien sind die wichtigste Methode, um den Zugriff auf einen AWS KMS-Schlüssel zu kontrollieren. Darüber hinaus können AWS KMS-Schlüsselzuweisungen den Zugriff auf AWS-Services ermöglichen, mit denen Daten in Ihrem Namen ver- und entschlüsselt werden. Nehmen Sie sich Zeit, um die bewährten Methoden für die Steuerung des Zugriffs auf AWS KMS-Schlüsseldurchzugehen.

Es hat sich bewährt, die Verwendung von Verschlüsselungsschlüsseln zu überwachen, um ungewöhnliche Zugriffsmuster zu erkennen. Vorgänge, die mit von AWS verwalteten Schlüsseln und kundenseitig verwalteten Schlüsseln ausgeführt werden, die in AWS KMS gespeichert sind, können in AWS CloudTrail protokolliert werden. Sie sollten regelmäßig überprüft werden. Besondere Aufmerksamkeit sollte dabei der Überwachung von Schlüsselzerstörungsereignissen gelten. Um die versehentliche oder böswillige Zerstörung von Schlüsseldaten zu verhindern, werden Schlüsseldaten bei Schlüsselzerstörungsereignissen nicht sofort gelöscht. Für Versuche, Schlüssel in AWS KMS zu löschen, gilt eine Wartezeit, die standardmäßig auf 30 Tage festgelegt ist. So haben Administratoren Zeit, diese Aktionen zu überprüfen und die Anfrage gegebenenfalls rückgängig zu machen.

Die meisten AWS-Services verwenden AWS KMS auf eine Weise, die für Sie transparent ist. Sie müssen lediglich entscheiden, ob Sie einen in AWS verwalteten oder einen kundenseitig verwalteten Schlüssel verwenden möchten. Wenn Ihr Workload die direkte Verwendung von AWS KMS zum Verschlüsseln oder Entschlüsseln von Daten erfordert, empfiehlt sich eine Umschlagverschlüsselung zum Schutz Ihrer Daten. Das AWS-Verschlüsselungs-SDK kann Ihren Anwendungen clientseitige Verschlüsselungsprimitive bereitstellen, um die Umschlagverschlüsselung zu implementieren und eine Integration in AWS KMS zu ermöglichen.

Implementierungsschritte

1. Ermitteln Sie die geeigneten Schlüsselverwaltungsoptionen (von AWS verwaltet oder vom Kunden verwaltet) für den Schlüssel.
   - Aus Gründen der Benutzerfreundlichkeit bietet AWS für die meisten Services AWS-eigene und von AWS verwaltete Schlüssel. Diese stellen eine Funktion für die Verschlüsselung von Daten im Ruhezustand bereit, ohne dass Schlüsseldaten oder -richtlinien verwaltet werden müssen.

- Wenn Sie kundenseitig verwaltete Schlüssel verwenden, sollten Sie den Standard-Schlüsselspeicher in Betracht ziehen, um das beste Gleichgewicht zwischen Agilität, Sicherheit, Datenhoheit und Verfügbarkeit zu erzielen. Andere Anwendungsfälle erfordern möglicherweise die Verwendung von benutzerdefinierten Schlüsselspeichern mit AWS CloudHSM oder einem externen Schlüsselspeicher.

2. Gehen Sie die Liste der Services durch, die Sie für Ihren Workload verwenden, um zu verstehen, wie AWS KMS in den Service integriert wird. EC2-Instances können beispielsweise verschlüsselte EBS-Volumes verwenden, um zu überprüfen, dass die von diesen Volumes erstellten Amazon EBS-Snapshots auch mit einem kundenseitig verwalteten Schlüssel verschlüsselt werden. So wir die versehentliche Offenlegung unverschlüsselter Snapshot-Daten verhindert.
   - So nutzen AWS-Services AWS KMS
   - Ausführliche Informationen zu den Verschlüsselungsoptionen, die ein AWS-Service bietet, finden Sie im Thema „Verschlüsselung im Ruhezustand“ im Benutzerhandbuch oder Entwicklerhandbuch für den Service.
3. Implementieren Sie AWS KMS: AWS KMS erleichtert Ihnen das Erstellen und Verwalten von Schlüsseln sowie die Kontrolle der Verschlüsselung in einer Vielzahl von AWS-Services und in Ihren Anwendungen.
   - Erste Schritte mit AWS Key Management Service (AWS KMS)
   - Lesen Sie die bewährten Methoden für die Steuerung des Zugriffs auf AWS KMS-Schlüssel.
4. Erwägen Sie die Verwendung des AWS Encryption SDK: Verwenden Sie das AWS Encryption SDK mit AWS KMS-Integration, wenn Ihre Anwendung Daten clientseitig verschlüsseln muss.
   - AWS Encryption SDK
5. Aktivieren Sie IAM Access Analyzer, um automatisch zu überprüfen und benachrichtigt zu werden, wenn zu weit gefasste AWS KMS-Schlüsselrichtlinien vorhanden sind.
6. Aktivieren Sie Security Hub, um Benachrichtigungen zu erhalten, wenn falsch konfigurierte Schlüsselrichtlinien, Schlüssel mit geplanter Löschung oder Schlüssel ohne aktivierte automatische Rotation vorhanden sind.
7. Ermitteln Sie die für Ihre AWS KMS-Schlüssel geeignete Protokollierungsstufe. Da Aufrufe von AWS KMS, einschließlich schreibgeschützter Ereignisse, protokolliert werden, können die CloudTrail-Protokolle für AWS KMS sehr umfangreich werden.
   - Einige Organisationen ziehen es vor, die AWS KMS-Protokollierungsaktivitäten in einem eigenen Pfad zu separieren. Weitere Details finden Sie im Abschnitt Logging AWS KMS API calls with CloudTrail (Protokollieren von AWS KMS-API-Aufrufen mit CloudTrail) im AWS KMS-Entwicklerhandbuch.

Ressourcen

Zugehörige Dokumente:

- AWS Key Management Service
- AWS cryptographic services and tools (Kryptografische AWS-Services und -Tools)
- Protecting Amazon S3 Data Using Encryption (Schutz von S3-Daten durch Verschlüsselung)
- Umschlagverschlüsselung
- Das Versprechen zu digitaler Souveränität
- Demystifying AWS KMS key operations, bring your own key, custom key store, and ciphertext portability (Das Geheimnis von AWS KMS-Schlüsselvorgängen, Bring Your Own Key, benutzerdefinierten Schlüsselspeichern und Portabilität von Geheimtext)
- AWS Key Management Service cryptographic details (Kryptografische Details in AWS Key Management Service)

Zugehörige Videos:

- How Encryption Works in AWS (So funktioniert die Verschlüsselung in AWS)
- Securing Your Block Storage on AWS (Sichern Ihres Blockspeichers in AWS)
- AWS data protection: Using locks, keys, signatures, and certificates (Datenschutz in AWS: Verwenden von Schlössern, Schlüsseln, Signaturen und Zertifikaten)

Zugehörige Beispiele:

- Implement advanced access control mechanisms using AWS KMS (Implementieren erweiterter Zugriffskontrollmechanismen mit AWS KMS)

SEC08-BP02 Erzwingen der Verschlüsselung im Ruhezustand

Sie sollten die Verwendung der Verschlüsselung von Daten im Ruhezustand erzwingen. Durch die Verschlüsselung wird die Vertraulichkeit sensitiver Daten im Falle eines unautorisierten Zugriffs oder einer unbeabsichtigten Offenlegung gewahrt.

Gewünschtes Ergebnis: Private Daten sollten im Ruhezustand standardmäßig verschlüsselt werden. Die Verschlüsselung wahrt die Vertraulichkeit der Daten und bietet eine zusätzliche Schutzebene gegen beabsichtigte oder unbeabsichtigte Datenoffenlegung oder Exfiltration. Verschlüsselte Daten

können ohne vorherige Entschlüsselung nicht gelesen oder genutzt werden. Alle unverschlüsselt gespeicherten Daten sollten inventarisiert und kontrolliert werden.

Typische Anti-Muster:

- keine Verwendung von Konfigurationen mit standardmäßiger Verschlüsselung
- Bereitstellung von Zugriffsmöglichkeiten mit zu vielen Berechtigungen für Entschlüsselungsschlüssel
- fehlende Überwachung der Ver- und Entschlüsselungsschlüssel
- Speichern von Daten ohne Verschlüsselung
- Verwendung desselben Verschlüsselungsschlüssels für alle Daten, ohne Berücksichtigung von Datennutzung, -typen und -klassifizierung

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Ordnen Sie den Datenklassifizierungen in Ihren Workloads Verschlüsselungsschlüssel zu. Dies hilft beim Schutz vor Zugriffsmöglichkeiten mit zu vielen Berechtigungen bei Verwendung eines einzigen oder sehr weniger Verschlüsselungsschlüssel für Ihre Daten (vgl. SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung).

AWS Key Management Service (AWS KMS) kann in viele AWS-Services integriert werden, um die Verschlüsselung Ihrer Daten im Ruhezustand zu vereinfachen. In Amazon Simple Storage Service (Amazon S3) können Sie beispielsweise die Standardverschlüsselung für einen Bucket festlegen, sodass neue Objekte automatisch verschlüsselt werden. Berücksichtigen Sie bei der Verwendung von AWS KMS, wie eng die Daten eingeschränkt werden müssen. Standard- und servicegesteuerte AWS KMS-Schlüssel werden für Sie von AWS verwaltet und verwendet. Ziehen Sie für sensible Daten, die einen differenzierten Zugriff auf den zugrunde liegenden Verschlüsselungsschlüssel erfordern, kundenverwaltete Schlüssel (CMKs) in Betracht. Sie haben die vollständige Kontrolle über CMKs, einschließlich Rotation und Zugriffsmanagement mithilfe von Schlüsselrichtlinien.

Zudem unterstützen Amazon Elastic Compute Cloud (Amazon EC2) und Amazon S3 das Erzwingen der Verschlüsselung durch Festlegen einer Standardverschlüsselung. Sie können AWS-Config-Regeln verwenden, um automatisch zu überprüfen, ob Sie die Verschlüsselung nutzen, z. B. für Amazon Elastic Block Store (Amazon EBS)-Volumes, Amazon Relational Database Service (Amazon RDS)-Instances und Amazon S3-Buckets.

AWS bietet auch Optionen für die clientseitige Verschlüsselung, mit der Sie Daten vor dem Laden in die Cloud verschlüsseln können. Das AWS Encryption SDK bietet eine Möglichkeit zur Verschlüsselung Ihrer Daten mit Umschlagverschlüsselung. Sie stellen den Wrapping-Schlüssel bereit und das AWS Encryption SDK generiert einen eindeutigen Datenschlüssel für jedes verschlüsselte Datenobjekt. Ziehen Sie AWS CloudHSM in Betracht, wenn Sie ein verwaltetes Single-Tenant-Hardware-Sicherheitsmodul (HSM) benötigen. Mit AWS CloudHSM können Sie kryptographische Schlüssel auf einem nach FIPS 140-2 Level 3 validierten HSM generieren, importieren und verwalten. Einige Anwendungsfälle von AWS CloudHSM umfassen den Schutz privater Schlüssel für die Ausgabe einer Zertifizierungsstelle (Certificate authority, CA) und die Aktivierung der transparenten Datenverschlüsselung (Transparent Data Encryption, TDE) für Oracle-Datenbanken. Das AWS CloudHSM-Client-SDK bietet Software, die die clientseitige Verschlüsselung von Daten mit innerhalb von AWS CloudHSM gespeicherten Schlüsseln ermöglicht, bevor die Daten zu AWS geladen werden. DerAmazon DynamoDB Encryption Client ermöglicht darüber hinaus das Verschlüsseln und Signieren von Elementen vor dem Laden in eine DynamoDB-Tabelle.

Implementierungsschritte

- Erzwingen Sie die Verschlüsselung von Daten im Ruhezustand für Amazon S3: Implementieren Sie die Standardverschlüsselung für Amazon S3-Buckets.

  Konfigurieren Sie die Standardverschlüsselung für neue Amazon EBS-Volumes: Legen Sie fest, dass alle neu erstellten Amazon EBS-Volumes verschlüsselt erstellt werden sollen. Dabei können Sie den von AWS bereitgestellten Standardschlüssel oder einen von Ihnen erstellten Schlüssel verwenden.

  Konfigurieren Sie verschlüsselte Amazon Machine Images (AMIs): Beim Kopieren eines vorhandenen AMI mit aktivierter Verschlüsselung werden Root-Volumes und Snapshots automatisch verschlüsselt.

  Konfigurieren Sie die Amazon RDS-Verschlüsselung: Konfigurieren Sie die Verschlüsselung für Ihre Amazon RDS-Datenbank-Cluster und Snapshots im Ruhezustand durch Aktivieren der Verschlüsselungsoption.

  Erstellen und konfigurieren Sie AWS KMS-Schlüssel mit Richtlinien, die den Zugriff für jede Datenklassifizierung auf die jeweiligen Prinzipale beschränken: Erstellen Sie beispielsweise einen AWS KMS-Schlüssel für die Verschlüsselung von Produktionsdaten und einen anderen Schlüssel für Entwicklungs- oder Testdaten. Sie können den Schlüsselzugriff auch für andere AWS-Konten gewähren. Ziehen Sie die Nutzung verschiedener Konten für Ihre Entwicklungs- und Produktionsumgebungen in Betracht. Wenn Ihre Produktionsumgebung

Artefakte im Entwicklungskonto entschlüsseln muss, können Sie die zur Verschlüsselung der Entwicklungsartefakte verwendete CMK-Richtlinie so bearbeiten, dass das Produktionskonto diese Artefakte entschlüsseln kann. Die Produktionsumgebung kann dann die entschlüsselten Daten zur Verwendung in der Produktion einlesen.

Konfigurieren Sie Verschlüsselung in weiteren AWS-Services: Sehen Sie sich die Sicherheitsdokumentation zu anderen verwendeten AWS-Services an, um die entsprechenden Verschlüsselungsoptionen festzustellen.

Ressourcen

Zugehörige Dokumente:

- AWS Crypto Tools
- Dokumentation zu AWS
- AWS Encryption SDK
- Whitepaper: Einführung in die kryptografischen Details von AWS KMS
- AWS Key Management Service
- AWS cryptographic services and tools (Kryptografische Services und Tools von AWS)
- Amazon EBS-Verschlüsselung
- Default encryption for Amazon EBS volumes (Standardverschlüsselung für Amazon EBS-Volumes)
- Verschlüsseln von Amazon RDS-Ressourcen
- How do I enable default encryption for an Amazon S3 bucket? (Wie kann ich die Standardverschlüsselung für einen Amazon S3-Bucket aktivieren?)
- Protecting Amazon S3 Data Using Encryption (Schutz von Amazon S3-Daten durch Verschlüsselung)

Zugehörige Videos:

- How Encryption Works in AWS (So funktioniert die Verschlüsselung in AWS)
- Securing Your Block Storage on AWS (Sichern Ihres Blockspeichers in AWS)

SEC08-BP03 Automatisieren des Schutzes von Daten im Ruhezustand

Nutzen Sie die Automatisierung, um Daten im Ruhezustand zu validieren und zu kontrollieren. Nutzen Sie automatisierte Scans, um Fehlkonfigurationen Ihrer Datenspeicherlösungen zu erkennen, und führen Sie, wenn möglich, Abhilfemaßnahmen durch automatisierte programmatische Reaktionen durch. Integrieren Sie die Automatisierung in Ihre CI/CD-Prozesse, um Fehlkonfigurationen des Datenspeichers zu erkennen, bevor sie in der Produktion bereitgestellt werden.

Gewünschtes Ergebnis: Automatisierte Systeme scannen und überwachen Datenspeicher auf Fehlkonfigurationen der Kontrollen, unbefugten Zugriff und unerwartete Nutzung. Die Erkennung von falsch konfigurierten Speicherorten leitet automatische Abhilfemaßnahmen ein. Automatisierte Prozesse erstellen Datensicherungen und speichern unveränderliche Kopien außerhalb der ursprünglichen Umgebung.

Typische Anti-Muster:

- Keine Berücksichtigung von Optionen zur Aktivierung der Verschlüsselung in den Standardeinstellungen, sofern unterstützt.
- Keine Berücksichtigung von Sicherheitsereignissen neben den betrieblichen Ereignissen bei der Formulierung einer automatisierten Backup- und Wiederherstellungsstrategie.
- Keine Durchsetzung der Einstellungen für den öffentlichen Zugriff auf Speicherservices.
- Keine Überwachung und Prüfung Ihrer Kontrollen zum Schutz von Daten im Ruhezustand.

Vorteile der Einführung dieser bewährten Methode: Die Automatisierung hilft, das Risiko einer Fehlkonfiguration Ihrer Datenspeicher zu vermeiden. Dieses Vorgehen hilft zu verhindern, dass Fehlkonfigurationen in Ihre Produktionsumgebungen gelangen. Diese bewährte Methode trägt außerdem dazu bei, Fehlkonfigurationen zu erkennen und zu beheben, falls sie auftreten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Automatisierung zieht sich wie ein roter Faden durch die Praktiken zum Schutz Ihrer Daten im Ruhezustand. SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen beschreibt, wie Sie die Konfiguration Ihrer Ressourcen mithilfe von Infrastructure as Code (IaC)-Vorlagen erfassen können, z. B. mit AWS CloudFormation. Diese Vorlagen werden in ein Versionskontrollsystem übertragen und zur Bereitstellung von Ressourcen in AWS über eine CI/CD-

Pipeline verwendet. Diese Techniken gelten auch für die Automatisierung der Konfiguration Ihrer Datenspeicherlösungen, z. B. für die Verschlüsselungseinstellungen in Amazon S3-Buckets.

Sie können die Einstellungen, die Sie in Ihren IaC-Vorlagen definieren, mithilfe von Regeln in AWS CloudFormation Guard auf Fehlkonfigurationen in Ihren CI/CD-Pipelines überprüfen. Sie können Einstellungen, die noch nicht in CloudFormation oder anderen IaC-Tools verfügbar sind, mit AWS Config auf Fehlkonfigurationen überwachen. Warnungen, die Config für Fehlkonfigurationen erzeugt, können automatisch behoben werden, wie in SEC04-BP04 Initiate remediation for non-compliant resources beschrieben.

Der Einsatz von Automatisierung als Teil Ihrer Strategie zur Verwaltung von Berechtigungen ist ebenfalls ein wesentlicher Bestandteil des automatisierten Datenschutzes. SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen und SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen beschreiben die Konfiguration von Zugriffsrichtlinien mit geringsten Berechtigungen, die kontinuierlich vom AWS Identity and Access Management Access Analyzer überwacht werden, um Erkenntnisse zu generieren, wenn die Berechtigung reduziert werden kann. Über die Automatisierung der Überwachung von Berechtigungen hinaus können Sie Amazon GuardDuty konfigurieren, um auf anomales Datenzugriffsverhalten für Ihre EBS-Volumes (über eine EC2-Instance), S3-Buckets und unterstützte Amazon Relational Database Service-Datenbanken zu achten.

Automatisierung spielt auch eine Rolle bei der Erkennung, wenn sensible Daten an nicht autorisierten Orten gespeichert sind. SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung beschreibt, wie Amazon Macie Ihre S3-Buckets auf unerwartete sensible Daten überwachen und Warnungen generieren kann, die eine automatisierte Reaktion auslösen können.

Befolgen Sie die Praktiken in REL09 Daten sichern, um eine automatisierte Datensicherungs- und Wiederherstellungsstrategie zu entwickeln. Datensicherung und -wiederherstellung sind für die Wiederherstellung nach Sicherheitsereignissen ebenso wichtig wie für betriebliche Ereignisse.

Implementierungsschritte

1. Erfassen Sie die Konfiguration des Datenspeichers in IaC-Vorlagen. Verwenden Sie automatische Prüfungen in Ihren CI/CD-Pipelines, um Fehlkonfigurationen zu erkennen.
   a. Sie können <ulink type="marketing" url="cloudformation">&CFN;</ulink> für Ihre IaC-Vorlagen verwenden und CloudFormation Guard um Vorlagen auf Fehlkonfigurationen zu überprüfen.
   b. Verwenden Sie AWS Config, um Regeln in einem proaktiven Bewertungsmodus auszuführen. Verwenden Sie diese Einstellung, um die Konformität einer Ressource als Schritt in Ihrer CI/CD-Pipeline zu prüfen, bevor Sie sie erstellen.

2. Überwachen Sie Ressourcen auf Fehlkonfigurationen des Datenspeichers.
   a. Legen Sie AWS Config fest, um Datenspeicher-Ressourcen auf Änderungen der Kontrollkonfigurationen zu überwachen und Warnungen zu generieren, um Abhilfemaßnahmen aufzurufen, wenn eine Fehlkonfiguration entdeckt wird.
   b. Weitere Hinweise zu automatischen Abhilfemaßnahmen finden Sie unter SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen.
3. Überwachen und reduzieren Sie die Datenzugriffsberechtigungen kontinuierlich durch Automatisierung.
   a. IAM Access Analyzer kann kontinuierlich ausgeführt werden, um Warnungen zu generieren, wenn die Berechtigungen möglicherweise reduziert werden können.
4. Überwachen Sie anomales Datenzugriffsverhalten und geben Sie entsprechende Warnmeldungen.
   a. GuardDuty überwacht sowohl bekannte Bedrohungssignaturen als auch Abweichungen vom grundlegenden Zugriffsverhalten auf Datenspeicherressourcen wie EBS-Volumes, S3-Buckets und RDS-Datenbanken.
5. Überwachen Sie sensible Daten, die an unerwarteten Orten gespeichert sind, und geben Sie entsprechende Warnmeldungen.
   a. Verwenden Sie Amazon Macie, um Ihre S3-Buckets kontinuierlich auf sensible Daten zu überprüfen.
6. Automatisieren Sie sichere und verschlüsselte Backups Ihrer Daten.
   a. AWS Backup ist ein verwalteter Service, der verschlüsselte und sichere Backups von verschiedenen Datenquellen in AWS erstellt. Mit Elastic Disaster Recovery können Sie komplette Workloads von Servern kopieren und einen kontinuierlichen Datenschutz mit einem in Sekunden gemessenen Recovery Point Objective (RPO) gewährleisten. Sie können beide Services so konfigurieren, dass sie zusammenarbeiten, um die Erstellung von Datensicherungen und das Kopieren der Daten an Failover-Standorte zu automatisieren. Dies kann dazu beitragen, dass Ihre Daten auch dann verfügbar bleiben, wenn sie durch betriebliche oder sicherheitsrelevante Ereignisse beeinträchtigt werden.

Ressourcen

Zugehörige bewährte Methoden:

- SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen
- SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen

- SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen
- SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung
- REL09-BP02 Schützen und Verschlüsseln von Backups
- REL09-BP03 Automatische Daten-Backups

Zugehörige Dokumente:

- AWS Prescriptive Guidance: Automatically encrypt existing and new Amazon EBS volumes
- Ransomware Risk Management on AWS Using the NIST Cyber Security Framework (CSF)

Zugehörige Beispiele:

- How to use AWS Config proactive rules and AWS CloudFormation Hooks to prevent creation of noncompliant cloud resources
- Automate and centrally manage data protection for Amazon S3 with AWS Backup
- AWS re:Invent 2023 – Implement proactive data protection using Amazon EBS snapshots
- AWS re:Invent 2022 – Build and automate for resilience with modern data protection

Zugehörige Tools:

- AWS CloudFormation Guard
- AWS CloudFormation Guard Rules Registry
- IAM Access Analyzer
- Amazon Macie
- AWS Backup
- Elastic Disaster Recovery

SEC08-BP04 Durchsetzen der Zugriffskontrolle

Um Ihre Daten im Ruhezustand zu schützen, sollten Sie Zugriffskontrollen über Mechanismen wie das Isolieren und die Versionsverwaltung durchsetzen und das Prinzip der geringsten Berechtigung anwenden. Verhindern Sie den öffentlichen Zugriff auf Ihre Daten.

Gewünschtes Ergebnis: Sie stellen sicher, dass nur autorisierte Benutzer auf Daten zugreifen können, wenn dies unbedingt erforderlich ist. Sie schützen Ihre Daten mit regelmäßigen Backups und

Versionsverwaltung vor beabsichtigten oder unbeabsichtigten Änderungen oder Löschungen. Sie isolieren wichtige Daten von anderen Daten, um die Vertraulichkeit und Datenintegrität zu schützen.

Typische Anti-Muster:

- gemeinsame Speicherung von Daten mit unterschiedlichen Anforderungen hinsichtlich Vertraulichkeit oder verschiedenen Klassifizierungen
- Verwendung von übermäßig großzügigen Berechtigungen für Entschlüsselungsschlüssel
- inkorrekte Klassifizierung von Daten
- keine Aufbewahrung von Sicherheitskopien wichtiger Daten
- Ermöglichen des dauerhaften Zugriffs auf Produktionsdaten
- keine Prüfung des Datenzugriffs bzw. keine regelmäßige Prüfung der Berechtigungen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Mehrere Kontrollen können zum Schutz Ihrer Daten im Ruhezustand beitragen, einschließlich Zugriff (unter Verwendung des Prinzips der geringsten Berechtigung), Isolierung und Versionsverwaltung. Der Zugriff auf Ihre Daten sollte mit Erkennungsmechanismen wie beispielsweise AWS CloudTrail und Service-Level-Protokollen (z. B. Amazon Simple Storage Service (Amazon S3)-Zugriffsprotokolle) überprüft werden. Sie sollten inventarisieren, welche Daten öffentlich zugänglich sind, und einen Plan erstellen, wie Sie die Menge an öffentlich verfügbaren Daten im Laufe der Zeit reduzieren können.

Amazon S3 Glacier Vault Lock und Amazon S3 Object Lock bieten eine obligatorische Zugriffskontrolle für Objekte in Amazon S3. Sobald eine Tresorrichtlinie mit der Compliance-Option gesperrt ist, kann sie nicht einmal der Root-Benutzer ändern, bis die Sperre abläuft.

Implementierungsschritte

- Erzwingen der Zugriffskontrolle: Erzwingen Sie die Zugriffskontrolle nach dem Prinzip der geringsten Berechtigung, einschließlich des Zugriffs auf Verschlüsselungsschlüssel.
- Trennen von Daten anhand unterschiedlicher Klassifizierungsstufen: Verwenden Sie unterschiedliche AWS-Konten für die Datenklassifizierungsstufen und verwalten Sie diese Konten mit AWS Organizations.
- Überprüfen von AWS Key Management Service (AWS KMS)-Richtlinien: Überprüfen Sie die gewährte Zugriffsebene in den AWS KMS-Richtlinien.

- Überprüfen der Berechtigungen für Amazon S3-Buckets und -Objekte: Überprüfen Sie regelmäßig den in S3-Bucket-Richtlinien gewährten Zugriff. Als bewährte Methode gilt, keine öffentlich lesbaren oder schreibbaren Buckets zu haben. Erwägen Sie, AWS Config zur Erkennung von öffentlich verfügbaren Buckets und Amazon CloudFront für die Bereitstellung von Inhalten aus Amazon S3 zu verwenden. Stellen Sie sicher, dass Buckets, die den öffentlichen Zugriff nicht gewähren sollten, so konfiguriert sind, dass ein öffentlicher Zugriff verhindert wird. Standardmäßig sind alle S3 Buckets privat. Der Zugriff ist nur für Benutzer möglich, denen der Zugriff ausdrücklich gewährt wurde.
- Aktivieren von AWS IAM Access Analyzer: IAM Access Analyzer analysiert Amazon S3-Buckets und generiert ein Ergebnis, wenn eine S3-Richtlinie Zugriff auf eine externe Entität gewährt.
- Aktivieren der Amazon S3-Versionsverwaltung und der Objektsperre, wenn dies angemessen ist.
- Verwenden von Amazon S3 Inventory: Amazon S3 Inventory kann verwendet werden, um den Replikations- und Verschlüsselungsstatus Ihrer S3-Objekte zu prüfen und zu melden.
- Überprüfen von Amazon EBS- und AMI-Freigabeberechtigungen: Mit Freigabeberechtigungen können Images und Volumes für AWS-Konten außerhalb Ihres Workloads freigegeben werden.
- Regelmäßiges Überprüfen der Freigaben von AWS Resource Access Manager, um zu bestimmen, ob Ressourcen weiterhin freigegeben werden sollten. Resource Access Manager ermöglicht die Freigabe von Ressourcen wie beispielsweise Richtlinien für AWS Network Firewall, Amazon Route 53-Resolver-Regeln und Subnetzen innerhalb Ihrer Amazon VPCs. Überprüfen Sie die freigegebenen Ressourcen regelmäßig und beenden Sie die Freigabe von Ressourcen, die keine Freigabe mehr erfordern.

Ressourcen

Zugehörige bewährte Methoden:

- SEC03-BP01 Definieren von Zugriffsanforderungen
- SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen

Zugehörige Dokumente:

- Whitepaper: Einführung in die kryptografischen Details von AWS KMS
- Einführung in die Verwaltung von Zugriffsberechtigungen für Ihre Amazon S3-Ressourcen
- Übersicht über die Verwaltung des Zugriffs auf Ihre AWS KMS-Ressourcen
- AWS-Config-Regeln

- Amazon S3 + Amazon CloudFront: A Match Made in the Cloud (Amazon S3 + Amazon CloudFront: Die perfekte Kombination in der Cloud)
- Verwenden der Versionsverwaltung
- Locking Objects Using Amazon S3 Object Lock (Sperren von Objekten mit der Amazon S3-Objektsperre)
- Teilen eines Amazon EBS-Snapshots
- Gemeinsame AMIs
- Hosting a single-page application on Amazon S3 (Hosten einer Single-Page-Anwendung in Amazon S3)

Zugehörige Videos:

- Securing Your Block Storage on AWS (Sichern Ihres Blockspeichers in AWS)

SEC 9. Wie schützen Sie Ihre Daten bei der Übertragung?

Schützen Sie Ihre Daten während der Übertragung, indem Sie mehrere Kontrollen implementieren, um das Risiko eines unbefugten Zugriffs oder Verlusts zu reduzieren.

Bewährte Methoden

- SEC09-BP01 Implementieren einer sicheren Schlüssel- und Zertifikatverwaltung
- SEC09-BP02 Erzwingen einer Verschlüsselung bei der Übertragung
- SEC09-BP03 Authentifizieren der Netzwerkkommunikation

SEC09-BP01 Implementieren einer sicheren Schlüssel- und Zertifikatverwaltung

Transport Layer Security-Zertifikate (TLS) werden verwendet, um die Netzwerkkommunikation zu sichern und die Identität von Websites, Ressourcen und Workloads über das Internet sowie in privaten Netzwerken festzulegen.

Gewünschtes Ergebnis: Ein sicheres Zertifikatverwaltungssystem, das Zertifikate in einer Public-Key-Infrastruktur (PKI) bereitstellen, speichern und verlängern kann. Ein sicherer Schlüssel- und Zertifikatsverwaltungsmechanismus verhindert die Offenlegung von Zertifikatsmaterial mit privaten Schlüsseln und erneuert das Zertifikat automatisch in regelmäßigen Abständen. Es lässt sich auch in andere Services integrieren, um eine sichere Netzwerkkommunikation und Identität für

Maschinenressourcen innerhalb Ihres Workloads zu gewährleisten. Schlüsseldaten sollten niemals für menschliche Identitäten zugänglich sein.

Typische Anti-Muster:

- Während der Bereitstellung oder Verlängerung von Zertifikaten werden manuelle Schritte ausgeführt.
- Beim Entwurf einer privaten Zertifizierungsstelle (Certificate Authority, CA) wird die Hierarchie der Zertifizierungsstelle nicht ausreichend beachtet.
- Für öffentliche Ressourcen werden selbstsignierte Zertifikate verwendet.

Vorteile der Nutzung dieser bewährten Methode:

- Die Zertifikatverwaltung wird durch automatisierte Bereitstellung und Verlängerung vereinfacht.
- Die Verschlüsselung von Daten während der Übertragung wird mithilfe von TLS-Zertifikaten gefördert.
- Sicherheit und Überprüfbarkeit der von der Zertifizierungsstelle ausgeführten Zertifikataktionen werden gesteigert.
- Verwaltungsaufgaben werden auf verschiedenen Ebenen der CA-Hierarchie angeordnet.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Moderne Workloads nutzen verschlüsselte Netzwerkkommunikation mithilfe von PKI-Protokollen wie TLS in großem Umfang. Die Verwaltung von PKI-Zertifikaten kann komplex sein, durch automatisierte Bereitstellung und Verlängerung von Zertifikaten können aber Reibungsverluste im Zusammenhang mit der Zertifikatverwaltung verringert werden.

AWS bietet zwei Services zur Verwaltung von allgemeinen PKI-Zertifikaten: AWS Certificate Manager und AWS Private Certificate Authority (AWS Private CA). ACM ist der primäre Service, den Kunden für die Bereitstellung und Verwaltung von Zertifikaten sowohl für öffentliche als auch für private AWS-Workloads verwenden. ACM stellt Zertifikate mithilfe von AWS Private CA aus und lässt sich in viele andere verwaltete AWS-Services zur Bereitstellung sicherer TLS-Zertifikate für Workloads integrieren.

AWS Private CA ermöglicht es Ihnen, Ihre eigene Stamm- oder untergeordnete Zertifizierungsstelle einzurichten und TLS-Zertifikate über eine API auszustellen. Sie können diese Art von Zertifikaten

in Szenarien verwenden, in denen Sie die Vertrauenskette auf der Clientseite der TLS-Verbindung kontrollieren und verwalten. Zusätzlich zu TLS-Anwendungsfällen kann AWS Private CA für die Ausstellung von Zertifikaten für Kubernetes-Pods, Matter-Geräteproduktbescheinigungen, Codesignaturen und andere Anwendungsfälle verwendet werden, und zwar mit einer benutzerdefinierten Vorlage. Sie können auch IAM Roles Anywhere verwenden, um temporäre IAM-Anmeldeinformationen für On-Premises-Workloads bereitzustellen, für die von Ihrer privaten CA signierte X.509-Zertifikate ausgestellt wurden.

Zusätzlich zu ACM und AWS Private CA bietet AWS IoT Core spezielle Unterstützung für die Bereitstellung und Verwaltung von PKI-Zertifikaten für IoT-Geräte. AWS IoT Core bietet spezielle Mechanismen für das das groß angelegte Onboarding von IoT-Geräten in Ihre Public-Key-Infrastruktur.

Überlegungen zur Einrichtung einer privaten CA-Hierarchie

Wenn Sie eine private Zertifizierungsstelle einrichten müssen, ist es wichtig, dass Sie besonders darauf achten, die CA-Hierarchie im Voraus richtig zu entwerfen. Es hat sich bewährt, beim Erstellen einer privaten CA-Hierarchie jede Ebene der Hierarchie in separaten AWS-Konten bereitzustellen. Dieser gezielte Schritt reduziert die Oberfläche für jede Ebene in der CA-Hierarchie, wodurch es einfacher wird, Anomalien in CloudTrail-Protokolldaten zu erkennen und den Umfang des Zugriffs oder die Auswirkungen eines unbefugten Zugriffs auf eines der Konten zu reduzieren. Die Stammzertifizierungsstelle sollte sich in einem eigenen separaten Konto befinden und nur zur Ausstellung eines oder mehrerer Zertifikate für eine Zwischenzertifizierungsstelle verwendet werden.

Erstellen Sie dann eine oder mehrere Zwischenzertifizierungsstellen in Konten, die vom Konto der Stammzertifizierungsstelle getrennt sind, um Zertifikate für Endbenutzer, Geräte oder andere Workloads auszustellen. Stellen Sie abschließend Zertifikate von Ihrer Stammzertifizierungsstelle an die Zwischenzertifizierungsstellen aus, die wiederum Zertifikate für die Endbenutzer oder Geräte ausstellen. Weitere Informationen zur Planung Ihrer CA-Bereitstellung und zum Entwerfen einer CA-Hierarchie, einschließlich Planung von Ausfallsicherheit, regionsübergreifender Replikation, gemeinsamer Nutzung von Zertifizierungsstellen in Ihrer Organisation und mehr, finden Sie unter Planung Ihrer AWS Private CA-Bereitstellung .

Implementierungsschritte

1. Ermitteln Sie die relevanten AWS-Services, die für Ihren Anwendungsfall erforderlich sind:
   - Viele Anwendungsfälle können die bestehende Public-Key-Infrastruktur von AWS mithilfe von AWS Certificate Managernutzen. ACM kann zur Bereitstellung von TLS-Zertifikaten für

Webserver, Load Balancer oder für andere Zwecke für öffentlich vertrauenswürdige Zertifikate verwendet werden.

- Erwägen Sie die AWS Private CA , wenn Sie Ihre eigene private Zertifizierungsstellenhierarchie einrichten müssen oder Zugriff auf exportierbare Zertifikate benötigen. Mit ACM können dann viele Arten von Endentitätszertifikaten mit dem AWS Private CA ausgegeben werden.
- Für Anwendungsfälle, in denen Zertifikate für eingebettete Geräte des Internet der Dinge (IoT) in großem Umfang bereitgestellt werden müssen, erwägen Sie den Einsatz von AWS IoT Core.

2. Implementieren Sie nach Möglichkeit eine automatische Zertifikatsverlängerung:
   - Verwenden Sie ACM verwaltete Verlängerung für Zertifikate, die von ACM zusammen mit integrierten AWS Managed Services ausgestellt wurden.
3. Richten Sie die Sie Protokollierung und Prüfpfade ein:
   - Aktivieren Sie CloudTrail-Protokolle, um Zugriff auf die Konten zu verfolgen, die Zertifizierungsstellen enthalten. Erwägen Sie, die Integritätsprüfung der Protokolldatei in CloudTrail zu konfigurieren, um die Authentizität der Protokolldaten zu überprüfen.
   - Generieren und überprüfen Sie regelmäßig Auditberichte, in denen die Zertifikate aufgeführt werden, die Ihre private CA ausgestellt oder widerrufen hat. Diese Berichte können in einen S3-Bucket exportiert werden.
   - Wenn Sie eine private CA bereitstellen, müssen Sie auch einen S3-Bucket einrichten, um die CRL (Certificate Revocation List, Zertifikatssperrliste) zu speichern. Anleitungen zur Konfiguration dieses S3-Buckets basierend auf den Anforderungen Ihres Workloads finden Sie unter Planung einer Zertifikatssperrliste (CRL).

Ressourcen

Zugehörige bewährte Methoden:

- SEC02-BP02 Verwenden von temporären Anmeldeinformationen
- SEC08-BP01: Implementieren einer sicheren Schlüsselverwaltung
- SEC09-BP03 Authentifizieren der Netzwerkkommunikation

Zugehörige Dokumente:

- Hosten und Verwalten einer ganzen privaten Zertifikatinfrastruktur in AWS
- Sichern einer ACM Private CA-Hierarchie auf Unternehmensebene für die Automobil- und Produktionsbranche

- Bewährte Private-CA-Methoden
- So verwenden Sie AWS RAM, um Ihre ACM Private CA kontoübergreifend zu teilen

Zugehörige Videos:

- Aktivieren von AWS Certificate Manager Certificate Manager Private CA (Workshop)

Zugehörige Beispiele:

- Private CA-Workshop
- Workshop zur IOT-Geräteverwaltung (einschließlich der Gerätebereitstellung)

Zugehörige Tools:

- Plugin für Kubernetes-Zertifikatmanager für die Verwendung von AWS Private CA

SEC09-BP02 Erzwingen einer Verschlüsselung bei der Übertragung

Erzwingen Sie Ihre definierten Verschlüsselungsanforderungen basierend auf den Richtlinien, regulatorischen Verpflichtungen und Standards Ihrer Organisation, damit Sie Ihre Unternehmens-, Rechts- und Compliance-Anforderungen erfüllen können. Verwenden Sie nur Protokolle mit Verschlüsselung, wenn Sie vertrauliche Daten außerhalb Ihrer Virtual Private Cloud (VPC) übertragen. Verschlüsselung hilft bei der Wahrung der Datenvertraulichkeit auch dann, wenn die Daten nicht vertrauenswürdige Netzwerke durchqueren.

Gewünschtes Ergebnis: Alle Daten sollten während der Übertragung mithilfe von sicheren TLS-Protokollen und Verschlüsselungssammlungen verschlüsselt werden. Der Netzwerkverkehr zwischen Ihren Ressourcen und dem Internet muss verschlüsselt werden, um nicht autorisierten Zugriff auf die Daten zu verhindern. Nur der Netzwerkverkehr in Ihrer internen AWS-Umgebung sollte wenn möglich mit TLS verschlüsselt werden. Das interne AWS-Netzwerk ist standardmäßig verschlüsselt und der Netzwerkverkehr innerhalb einer VPC kann nicht manipuliert oder analysiert werden, es sei denn, eine unbefugte Partei hat sich Zugang zu der Ressource verschafft, die den Datenverkehr generiert (wie beispielsweise Amazon EC2-Instances und Amazon ECS-Container). Überlegen Sie, ob Sie den Netzwerk-zu-Netzwerk-Datenverkehr mit einem IPsec Virtual Private Network (VPN) schützen sollten.

Typische Anti-Muster:

- Verwendung veralteter Versionen von SSL, TLS und Komponenten von Verschlüsselungssammlungen (z. B. SSL v3.0, RSA-Schlüssel mit 1 024 Bit und RC4-Verschlüsselung)
- Zulassen von unverschlüsseltem (HTTP-)Datenverkehr zu oder von öffentlich zugänglichen Ressourcen
- keine Überwachung und kein Ersatz von X.509-Zertifikaten, bevor sie ablaufen
- Verwendung von selbstsignierten X.509-Zertifikaten für TLS

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

AWS-Services bieten HTTPS-Endpunkte, die für die Kommunikation TLS nutzen. Dadurch werden die Daten bei der Kommunikation mit den AWS-APIs während der Übertragung verschlüsselt. Unsichere Protokolle wie HTTP können in einer VPC durch die Verwendung von Sicherheitsgruppen überprüft und blockiert werden. HTTP-Anfragen können in Amazon CloudFront oder einem Application Load Balancer auch automatisch an HTTPS umgeleitet werden. Sie haben uneingeschränkte Kontrolle über Ihre Datenverarbeitungsressourcen und können die Verschlüsselung während der Übertragung in alle Ihre Services implementieren. Darüber hinaus können Sie die VPN-Konnektivität mit Ihrer VPC von einem externen Netzwerk oder AWS Direct Connect aus verwenden, um die Verschlüsselung des Datenverkehrs zu erleichtern. Stellen Sie sicher, dass Ihre Kunden AWS-API-Aufrufe mindestens mit TLS 1.2 tätigen, da AWS die Verwendung von TLS 1.0 und 1.1 im Juni 2023 einstellt. Sollten Sie besondere Anforderungen haben, finden Sie Lösungen von Drittanbietern im AWS Marketplace.

Implementierungsschritte

- Erzwingen der Verschlüsselung bei der Übertragung: Die definierten Verschlüsselungsanforderungen sollten sich nach den neuesten Standards und bewährten Methoden richten und nur sichere Protokolle zulassen. Konfigurieren Sie beispielsweise eine Sicherheitsgruppe, die nur das HTTPS-Protokoll für einen Application Load Balancer oder eine Amazon EC2-Instance zulässt.
- Konfigurieren von sicheren Protokollen bei Edge-Services: Konfigurieren Sie HTTPS mit Amazon CloudFront und verwenden Sie ein für Ihren Sicherheitsstatus und Ihren Anwendungsfall geeignetes Sicherheitsprofil.

- Verwenden eines VPN für die externe Konnektivität: Ziehen Sie ein IPsec-VPN in Betracht, um Punkt-zu-Punkt- oder Netzwerk-zu-Netzwerk-Verbindungen zu sichern und so den Datenschutz und die Datenintegrität zu gewährleisten.
- Konfigurieren von sicheren Protokollen bei Load Balancern: Wählen Sie eine Sicherheitsrichtlinie aus, die die stärksten Verschlüsselungssammlungen bereitstellt, die von den Clients unterstützt werden, die eine Verbindung mit dem Listener herstellen. Erstellen Sie einen HTTPS-Listener für Ihren Application Load Balancer.
- Konfigurieren von sicheren Protokollen in Amazon Redshift: Konfigurieren Sie Ihren Cluster so, dass eine Verbindung über Secure Socket Layer (SSL) or Transport Layer Security (TLS) vorgeschrieben ist.
- Konfigurieren von sicheren Protokollen: Sehen Sie sich die AWS-Servicedokumentation an, um die Funktionen zur Verschlüsselung während der Übertragung zu bestimmen.
- Konfigurieren von sicherem Zugriff beim Hochladen in Amazon S3-Buckets: Verwenden Sie die Richtlinienkontrolle für Amazon S3-Buckets, um sicheren Zugriff auf Daten zu erzwingen.
- Erwägen der Verwendung von AWS Certificate Manager: ACM ermöglicht das Bereitstellen und Verwalten von öffentlichen TLS-Zertifikaten zur Verwendung mit AWS-Services.
- Erwägen der Verwendung von AWS Private Certificate Authority für private PKI-Anforderungen: AWS Private CA ermöglicht das Erstellen privater Zertifizierungsstellenhierarchien, um X.509-Endentitätszertifikate auszustellen, die zum Erstellen verschlüsselter TLS-Kanäle verwendet werden können.

Ressourcen

Zugehörige Dokumente:

- Dokumentation zu AWS
- Verwenden von HTTPS mit CloudFront
- Verbinden Ihrer VPC mit Remote-Netzwerken über AWS Virtual Private Network
- Create an HTTPS listener for your Application Load Balancer (Erstellen eines HTTPS-Listeners für Ihren Application Load Balancer)
- Tutorial: SSL/TLS unter Amazon Linux 2 konfigurieren
- Verwenden von SSL/TLS für die Verschlüsselung einer Verbindung zu einer DB-Instance
- Konfigurieren von Sicherheitsoptionen für Verbindungen

SEC09-BP03 Authentifizieren der Netzwerkkommunikation

Überprüfen Sie die Identität der Kommunikation mithilfe von Protokollen, die die Authentifizierung unterstützen, wie Transport Layer Security (TLS) oder IPsec.

Entwerfen Sie Ihren Workload so, dass bei der Kommunikation zwischen Services, Anwendungen oder Benutzern sichere, authentifizierte Netzwerkprotokolle verwendet werden. Die Verwendung von Netzwerkprotokollen, die Authentifizierung und Autorisierung unterstützen, bietet eine strengere Kontrolle über den Netzwerkfluss und reduziert die Auswirkungen von nicht autorisiertem Zugriff.

Gewünschtes Ergebnis: Ein Workload mit klar definierten Datenflüssen auf der Daten- und Steuerebene zwischen den Services. Die Datenflüsse verwenden authentifizierte und verschlüsselte Netzwerkprotokolle, sofern dies technisch möglich ist.

Typische Anti-Muster:

- Unverschlüsselte oder unauthentifizierte Datenflüsse innerhalb Ihres Workloads
- Wiederverwendung von Authentifizierungsdaten für mehrere Benutzer oder Entitäten
- Die alleinige Verwendung von Netzwerkkontrollen als Zugriffskontrolle
- Erstellen eines benutzerdefinierten Authentifizierungsmechanismus, anstatt sich auf die Standard-Authentifizierungsmechanismen der Branche zu verlassen
- Übermäßig freizügige Datenflüsse zwischen Servicekomponenten oder anderen Ressourcen in der VPC

Vorteile der Nutzung dieser bewährten Methode:

- Schränkt den Umfang der Auswirkungen eines unberechtigten Zugriffs auf einen Teil des Workloads ein
- Bietet ein höheres Maß an Sicherheit, dass Aktionen nur von authentifizierten Personen durchgeführt werden können
- Verbessert die Entkopplung von Diensten, indem die vorgesehenen Schnittstellen für die Datenübertragung klar definiert und durchgesetzt werden
- Verbessert die Überwachung, Protokollierung und Reaktion auf Vorfälle durch die Zuordnung von Anfragen und gut definierte Kommunikationsschnittstellen
- Bietet durch die Kombination von Netzwerkkontrollen mit Authentifizierungs- und Autorisierungskontrollen einen umfassenden Schutz für Ihre Workloads

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Die Netzwerkverkehrsmuster Ihres Workloads lassen sich in zwei Kategorien einteilen:

- Der Ost-West-Verkehr steht für Datenflüsse zwischen Services, die einen Workload ausmachen.
- Der Nord-Süd-Verkehr stellt die Datenflüsse zwischen Ihrem Workload und den Verbrauchern dar.

Während es üblich ist, den Nord-Süd-Verkehr zu verschlüsseln, ist die Sicherung des Ost-West-Verkehrs mit authentifizierten Protokollen weniger verbreitet. Moderne Sicherheitspraktiken empfehlen, dass das Netzwerkdesign allein noch keine vertrauenswürdige Beziehung zwischen zwei Entitäten gewährleistet. Auch wenn sich zwei Services innerhalb einer gemeinsamen Netzwerkgrenze befinden, ist es immer noch die beste Methode, die Kommunikation zwischen diesen Services zu verschlüsseln, zu authentifizieren und zu autorisieren.

Beispielsweise verwenden AWS-Service-APIs das Signaturprotokoll AWS Signature Version 4 (SigV4), um den Anforderer zu authentifizieren, unabhängig davon, aus welchem Netzwerk die Anfrage stammt. Diese Authentifizierung stellt sicher, dass AWS-APIs die Identität des Anforderers der Aktion überprüfen können. Diese Identität kann dann mit Richtlinien kombiniert werden, um eine Autorisierungsentscheidung zu treffen, ob die Aktion erlaubt werden soll oder nicht.

Mit Services wie Amazon VPC Lattice und Amazon API Gateway können Sie das gleiche SigV4-Signaturprotokoll verwenden, um den Ost-West-Verkehr in Ihren eigenen Workloads zu authentifizieren und zu autorisieren. Wenn Ressourcen außerhalb Ihrer AWS-Umgebung mit Services kommunizieren müssen, die eine SigV4-basierte Authentifizierung und Autorisierung erfordern, können Sie AWS Identity and Access Management (IAM) Roles Anywhere auf der AWS-fremden Ressource verwenden, um temporäre AWS-Anmeldeinformationen zu erhalten. Diese Anmeldeinformationen können verwendet werden, um Anfragen an Services zu signieren, die mit SigV4 den Zugriff autorisieren.

Ein weiterer gängiger Mechanismus zur Authentifizierung des Ost-West-Verkehrs ist die gegenseitige TLS-Authentifizierung (mTLS). Viele Internet der Dinge (IoT)- und Business-to-Business-Anwendungen sowie Microservices verwenden mTLS, um die Identität beider Seiten einer TLS-Kommunikation durch die Verwendung von X.509-Zertifikaten auf Client- und Server-Seite zu validieren. Diese Zertifikate können von AWS Private Certificate Authority (AWS Private CA) ausgestellt werden. Sie können Services wie Amazon API Gateway und AWS App Mesh verwenden, um die mTLS-Authentifizierung für die Kommunikation zwischen oder innerhalb eines Workloads

bereitzustellen. Während mTLS Authentifizierungsinformationen für beide Seiten einer TLS-Kommunikation bereitstellt, bietet es keinen Mechanismus zur Autorisierung.

Nicht zuletzt sind OAuth 2.0 und OpenID Connect (OIDC) zwei Protokolle, die in der Regel für die Kontrolle des Zugriffs von Benutzern auf Services verwendet werden, jetzt aber auch für den Datenverkehr von Service zu Service immer beliebter werden. API Gateway bietet einen JSON Web Token (JWT) Authorizer, der es Workloads ermöglicht, den Zugriff auf API-Routen mithilfe von JWTs zu beschränken, die von OIDC- oder OAuth-2.0-Identitätsanbietern ausgestellt wurden. OAuth2-Bereiche können als Quelle für grundlegende Autorisierungsentscheidungen verwendet werden, aber die Autorisierungsprüfungen müssen immer noch in der Anwendungsschicht implementiert werden. Und OAuth2-Bereiche allein können komplexere Autorisierungsanforderungen nicht unterstützen.

Implementierungsschritte

- Definieren und Dokumentieren der Netzwerkflüsse Ihres Workloads: Der erste Schritt bei der Implementierung einer umfassenden Verteidigungsstrategie ist die Definition der Datenflüsse Ihres Workloads.
  - Erstellen Sie ein Datenflussdiagramm, das klar definiert, wie Daten zwischen den verschiedenen Services, aus denen Ihr Workload besteht, übertragen werden. Dieses Diagramm ist der erste Schritt zur Durchsetzung dieser Datenflüsse über authentifizierte Netzwerkkanäle.
  - Nutzen Sie Ihre Workloads in der Entwicklungs- und Testphase, um zu überprüfen, ob das Datenflussdiagramm das Verhalten der Workloads zur Laufzeit korrekt wiedergibt.
  - Ein Datenflussdiagramm kann auch bei der Durchführung einer Bedrohungsmodellierung nützlich sein, wie in SEC01-BP07 Identifizierung von Bedrohungen und Priorisierung von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells beschrieben.
- Einrichten von Netzwerkkontrollen: Erwägen Sie AWS-Funktionen, um Netzwerkkontrollen einzurichten, die auf Ihre Datenflüsse abgestimmt sind. Netzwerkgrenzen sollten zwar nicht die einzige Sicherheitskontrolle sein, aber sie stellen eine Stufe der umfassenden Verteidigungsstrategie zum Schutz Ihres Workloads dar.
  - Verwenden Sie Sicherheitsgruppen, um den Datenfluss zwischen Ressourcen zu definieren und einzuschränken.
  - Verwenden Sie AWS PrivateLink, um sowohl mit AWS als auch mit Drittanbieterservices zu kommunizieren, die AWS PrivateLink unterstützen. Daten, die über einen AWS PrivateLink-Schnittstellen-Endpunkt gesendet werden, bleiben innerhalb des AWS-Netzwerk-Backbones und durchlaufen nicht das öffentliche Internet.

- Implementieren von Authentifizierung und Autorisierung für alle Services in Ihrem Workload: Wählen Sie die AWS-Services aus, die am besten geeignet sind, um authentifizierte, verschlüsselte Datenflüsse in Ihrem Workload bereitzustellen.
  - Ziehen Sie Amazon VPC Lattice in Erwägung, um die Kommunikation von Service zu Service zu sichern. VPC Lattice kann SigV4-Authentifizierung in Kombination mit Authentifizierungsrichtlinien verwenden, um den Zugriff von Service zu Service zu kontrollieren.
  - Für die serviceübergreifende Kommunikation mit mTLS sollten Sie API Gateway oder App Mesh in Betracht ziehen. AWS Private CA kann verwendet werden, um eine private CA-Hierarchie einzurichten, die Zertifikate für die Verwendung mit mTLS ausstellen kann.
  - Bei der Integration mit Services, die OAuth 2.0 oder OIDC verwenden, sollten Sie API Gateway unter Verwendung des JWT-Genehmigers in Betracht ziehen.
  - Für die Kommunikation zwischen Ihrem Workload und IoT-Geräten sollten Sie AWS IoT Core in Betracht ziehen, das mehrere Optionen für die Verschlüsselung und Authentifizierung des Netzwerkverkehrs bietet.
- Überwachung auf nicht autorisierten Zugriff: Überwachen Sie kontinuierlich unbeabsichtigte Kommunikationskanäle, nicht autorisierte Auftraggeber, die versuchen, auf geschützte Ressourcen zuzugreifen, und andere unzulässige Zugriffsmuster.
  - Wenn Sie VPC Lattice zur Verwaltung des Zugriffs auf Ihre Services verwenden, sollten Sie die Zugriffsprotokolle von VPC Lattice aktivieren und überwachen. Diese Zugriffsprotokolle enthalten Informationen über die anfragende Entität, Netzwerkinformationen einschließlich Quell- und Ziel-VPC und Metadaten der Anfrage.
  - Erwägen Sie die Aktivierung von VPC Flow-Protokollen, um Metadaten zu Netzwerkflüssen zu erfassen und regelmäßig auf Anomalien zu überprüfen.
  - Weitere Hinweise zum Planen, Simulieren und Reagieren auf Sicherheitsvorfälle finden Sie im AWS Security Incident Response Guide und im Abschnitt Vorfallreaktion der Säule „Sicherheit“ des AWS-Well-Architected-Framework.

Ressourcen

Zugehörige bewährte Methoden:

- SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs
- SEC02-BP02 Verwenden von temporären Anmeldeinformationen
- SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells

Zugehörige Dokumente:

- Evaluating access control methods to secure Amazon API Gateway APIs
- Configuring mutual TLS authentication for a REST API
- How to secure API Gateway HTTP endpoints with JWT authorizer
- Authorizing direct calls to AWS services using AWS IoT Core credential provider
- AWS Security Incident Response Guide

Zugehörige Videos:

- AWS re:invent 2022: Introducing VPC Lattice
- AWS re:invent 2020: Serverless API authentication for HTTP APIs on AWS

Zugehörige Beispiele:

- Amazon VPC Lattice Workshop
- Zero-Trust Episode 1 – The Phantom Service Perimeter workshop

Vorfallsreaktion

Frage

- SEC 10. Wie können Sie Vorfälle voraussagen, darauf reagieren und diese beheben?

SEC 10. Wie können Sie Vorfälle voraussagen, darauf reagieren und diese beheben?

Obwohl die präventiven und Erkennungskontrollen mittlerweile ausgereift sind, sollte Ihr Unternehmen Verfahren etablieren, um auf Sicherheitsvorfälle reagieren und mögliche Auswirkungen mindern zu können. Ihre Vorbereitung wirkt sich stark auf die Fähigkeit Ihrer Teams aus, während eines Vorfalls effektiv zu arbeiten, Probleme zu isolieren, einzudämmen und forensisch zu untersuchen sowie den Betrieb in einem bekannten guten Zustand wiederherzustellen. Durch die Bereitstellung von Tools und Zugriff vor einem Sicherheitsvorfall und die routinemäßige Reaktion auf Vorfälle im Alltag können Sie sicherstellen, dass Sie eine Wiederherstellung durchführen und die Betriebsunterbrechung minimieren können.

Bewährte Methoden

- SEC10-BP01 Identifizieren wichtiger Mitarbeiter und externer Ressourcen
- SEC10-BP02 Entwickeln von Vorfallmanagementplänen
- SEC10-BP03 Vorbereiten forensischer Funktionen
- SEC10-BP04 Entwickeln und Testen von Playbooks für die Reaktion auf Sicherheitsvorfälle
- SEC10-BP05 Vorab bereitgestellter Zugriff
- SEC10-BP06 Vorabbereitstellen von Tools
- SEC10-BP07 Durchführen von Simulationen
- SEC10-BP08 Entwickeln eines Frameworks, um aus Vorfällen zu lernen

SEC10-BP01 Identifizieren wichtiger Mitarbeiter und externer Ressourcen

Identifizieren Sie internes und externes Personal, Ressourcen und rechtliche Anforderungen, die Ihre Organisation bei der Reaktion auf einen Vorfall unterstützen.

Gewünschtes Ergebnis: Sie haben eine Liste der wichtigsten Mitarbeiter, deren Kontaktinformationen und die Rollen, die sie bei der Reaktion auf ein Sicherheitsereignis spielen. Sie überprüfen diese Informationen regelmäßig und aktualisieren sie, um personelle Veränderungen aus Sicht der internen und externen Tools zu berücksichtigen. Bei der Dokumentation dieser Informationen berücksichtigen Sie alle Drittanbieter und Dienstleister, einschließlich Sicherheitspartner, Cloud-Anbieter und Software as a Service (SaaS)-Anwendungen. Während eines Sicherheitsereignisses stehen Mitarbeiter mit dem entsprechenden Maß an Verantwortung, Kontext und Zugriff zur Verfügung, um zu reagieren und sich zu erholen.

Typische Anti-Muster:

- Fehlen einer aktualisierten Liste der wichtigsten Mitarbeiter mit Kontaktinformationen, ihren Aufgaben und ihren Verantwortlichkeiten bei der Reaktion auf Sicherheitsvorfälle
- Voraussetzen, dass jeder die Menschen, Abhängigkeiten, Infrastruktur und Lösungen bei der Reaktion auf ein Ereignis und bei der Wiederherstellung nach einem Ereignis versteht
- Fehlen eines Dokuments oder eines Wissensspeichers, der die wichtigsten Infrastruktur- oder Anwendungsdesigns darstellt
- Fehlen von angemessenen Einarbeitungsprozessen für neue Mitarbeiter, um effektiv zur Reaktion auf ein Sicherheitsereignis beizutragen, wie z. B. die Durchführung von Ereignissimulationen
- Fehlen eines Eskalationspfades für den Fall, dass wichtige Mitarbeiter vorübergehend nicht verfügbar sind oder bei Sicherheitsereignissen nicht reagieren können

Vorteile der Einführung dieser bewährten Methode: Diese Praxis reduziert die Triage- und Reaktionszeit, die für die Identifizierung der richtigen Mitarbeiter und ihrer Rollen während eines Ereignisses aufgewendet wird. Minimieren Sie Zeitverluste während eines Ereignisses, indem Sie eine aktualisierte Liste der wichtigsten Mitarbeiter und ihrer Rollen führen, damit Sie die richtigen Personen für die Triage und die Wiederherstellung nach einem Ereignis einsetzen können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Identifizieren Sie wichtige Personen in Ihrer Organisation: Führen Sie eine Kontaktliste der Personen in Ihrer Organisation, die Sie einbeziehen müssen. Überprüfen und aktualisieren Sie diese Informationen regelmäßig bei personellen Veränderungen wie organisatorischen Änderungen, Beförderungen und Teamwechseln. Dies ist besonders wichtig für Schlüsselpositionen wie Incident Manager, Incident Responder und Communications Lead.

- Incident Manager: Incident Managers haben die Gesamtverantwortung für die Reaktion auf das Ereignis.
- Incident Responder: Incident Responders sind für Untersuchungen und Abhilfemaßnahmen zuständig. Diese Personen können sich je nach Art des Ereignisses unterscheiden, sind aber in der Regel Entwickler und Betriebs-Teams, die für die betroffene Anwendung verantwortlich sind.
- Communications Lead: Communications Leads sind für die interne und externe Kommunikation verantwortlich, insbesondere mit Behörden, Regulierungsbehörden und Kunden.
- Fachexperten (Subject Matter Experts, SMEs): Im Falle von verteilten und autonomen Teams empfehlen wir Ihnen, für geschäftskritische Workloads SMEs zu bestimmen. Sie bieten Einblicke in den Betrieb und die Datenklassifizierung von kritischen Workloads, die an dem Ereignis beteiligt sind.

Benutzen Sie die Funktion AWS Systems Manager Incident Manager, um wichtige Kontakte zu erfassen, einen Reaktionsplan zu definieren, Bereitschaftspläne zu automatisieren und Eskalationspläne zu erstellen. Automatisieren und rotieren Sie alle Mitarbeiter durch einen Bereitschaftsdienstplan, sodass die Verantwortung für den Workload auf alle Eigentümer verteilt wird. Dies fördert gute Praktiken wie die Ausgabe relevanter Metriken und Protokolle sowie die Definition von Alarmschwellen, die für den Workload von Bedeutung sind.

Externe Partner identifizieren: Unternehmen nutzen Tools, die von unabhängigen Softwareanbietern (ISVs), Partnern und Subunternehmern entwickelt wurden, um differenzierte Lösungen für ihre Kunden zu erstellen. Engagieren Sie wichtige Mitarbeiter dieser Parteien, die Ihnen bei der Reaktion

auf einen Vorfall und bei dessen Bewältigung helfen können. Wir empfehlen Ihnen, sich für die entsprechende Stufe von Support anzumelden, um über einen Supportfall sofortigen Zugang zu AWS Fachexperten zu erhalten. Erwägen Sie ähnliche Vereinbarungen mit allen Anbietern kritischer Lösungen für die Workloads. Einige Sicherheitsereignisse machen es erforderlich, dass börsennotierte Unternehmen die zuständigen Behörden und Aufsichtsbehörden über das Ereignis und dessen Auswirkungen informieren. Pflegen und aktualisieren Sie die Kontaktinformationen der relevanten Abteilungen und der zuständigen Personen.

Implementierungsschritte

1. Richten Sie eine Lösung für das Vorfallmanagement ein.
   a. Erwägen Sie die Bereitstellung von Incident Manager in Ihrem Security Tooling-Konto.
2. Definieren Sie Kontakte in Ihrer Lösung für das Vorfallmanagement.
   a. Definieren Sie für jeden Kontakt mindestens zwei Arten von Kontaktkanälen (z. B. SMS, Telefon oder E-Mail), um die Erreichbarkeit während eines Vorfalls sicherzustellen.
3. Definieren Sie einen Reaktionsplan.
   a. Ermitteln Sie die am besten geeigneten Ansprechpartner für einen Vorfall. Definieren Sie Eskalationspläne, die sich an den Rollen der einzuschaltenden Mitarbeiter orientieren, und nicht an einzelnen Ansprechpartnern. Erwägen Sie die Aufnahme von Kontakten, die für die Benachrichtigung externer Stellen zuständig sein könnten, auch wenn diese nicht direkt an der Lösung des Vorfalls beteiligt sind.

Ressourcen

Zugehörige bewährte Methoden:

- OPS02-BP03 Betriebsaktivitäten haben feste Eigentümer, die für ihre Leistung verantwortlich sind

Zugehörige Dokumente:

- AWS-Leitfaden für Security Incident Response

Zugehörige Beispiele:

- AWS Customer Playbook Framework
- Prepare for and respond to security incidents in your AWS environment

Zugehörige Tools:

- AWS Systems Manager Incident Manager

Zugehörige Videos:

- Amazon's approach to security during development

SEC10-BP02 Entwickeln von Vorfallmanagementplänen

Das erste Dokument, das für die Vorfallreaktion entwickelt werden muss, ist der Vorfallreaktionsplan. Der Vorfallreaktionsplan ist als Grundlage für Ihr Vorfallreaktionsprogramm und Ihre Vorfallreaktionsstrategie konzipiert.

Vorteile der Nutzung dieser bewährten Methode: Die Entwicklung gründlicher und klar definierter Prozesse zur Vorfallreaktion ist der Schlüssel zu einem erfolgreichen und skalierbaren Vorfallreaktionsprogramm. Wenn ein Sicherheitsereignis eintritt, können Ihnen klare Schritte und Workflows dabei helfen, rechtzeitig zu reagieren. Möglicherweise verfügen Sie bereits über bestehende Prozesse zur Vorfallreaktion. Unabhängig von Ihrem aktuellen Status ist es wichtig, Ihre Prozesse zur Vorfallreaktion regelmäßig zu aktualisieren, zu wiederholen und zu testen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Ein Vorfallreaktionsplan ist von entscheidender Bedeutung, um auf Sicherheitsvorfälle zu reagieren, sie einzudämmen und ihre potenziellen Folgen zu beheben. Ein Vorfallmanagementplan ist ein strukturierter Prozess für die Identifizierung und Behebung von Sicherheitsvorfällen sowie die zeitgerechte Reaktion darauf.

In der Cloud gibt es viele der betrieblichen Rollen und Anforderungen, die auch für eine On-Premises-Umgebung typisch sind. Bei der Erstellung eines Vorfallmanagementplans ist es wichtig, Reaktions- und Wiederherstellungsstrategien zu berücksichtigen, die optimal zu Ihren Anforderungen an geschäftliche Ergebnisse und Compliance passen. Wenn Sie beispielsweise Workloads in AWS bearbeiten, die mit FedRAMP in den USA kompatibel sind, sollten Sie den NIST SP 800-61 Computer Security Handling Guide berücksichtigen. Ähnlich gilt beim Betrieb von Workloads mit persönlich identifizierbaren Informationen (PII) in Europa, dass Sie an Szenarien denken sollten, in denen Sie diese schützen und auf Probleme reagieren müssen, die im Zusammenhang mit den Bestimmungen

zu Datenspeicherorten der Regulierungen der Datenschutz-Grundverordnung (DSGVO) der EU stehen.

Wenn Sie einen Vorfallmanagementplan für Ihre Workloads in AWS erstellen, beginnen Sie mit dem AWS-Modell der geteilten Verantwortung zum Aufbau eines gründlichen Verteidigungskonzepts im Rahmen Ihrer Vorfallreaktionen. In diesem Modell kümmert sich AWS um die Sicherheit der Cloud und Sie sind für die Sicherheit in der Cloud verantwortlich. Dies bedeutet, dass Sie die Kontrolle behalten und für die Sicherheitskontrollen verantwortlich sind, für deren Implementierung Sie sich entscheiden. Der Leitfaden für AWS Security Incident Response enthält zentrale Konzepte und grundlegende Anleitungen für den Aufbau eines cloudbasierten Vorfallmanagementplans.

Ein effektiver Vorfallmanagementplan muss kontinuierlich iteriert und stets an die Ziele Ihrer Cloud-Operationen angepasst werden. Erwägen Sie die Verwendung der nachfolgend erläuterten Implementierungspläne für die Erstellung und Weiterentwicklung Ihres Vorfallmanagementplans.

Implementierungsschritte

Definieren von Rollen und Zuständigkeiten

Der Umgang mit Sicherheitsereignissen erfordert organisationsübergreifende Disziplin und Handlungsbereitschaft. Innerhalb Ihrer Organisationsstruktur sollte es viele Personen geben, die für einen Vorfall verantwortlich, rechenschaftspflichtig, konsultiert oder auf dem Laufenden gehalten werden, z. B. Vertreter der Personalabteilung (HR), des Führungsteams und der Rechtsabteilung. Berücksichtigen Sie diese Rollen und Verantwortlichkeiten und ob Dritte beteiligt sein müssen. Beachten Sie, dass in vielen Regionen lokale Gesetze gelten, die regeln, was getan werden sollte und was nicht. Auch wenn es bürokratisch erscheinen mag, ein Diagramm für Verantwortung, Rechenschaftspflicht, Berater und zu Informierende (RACI) für Ihre Sicherheitspläne zu erstellen, erleichtert dies eine schnelle und direkte Kommunikation und gibt einen klaren Überblick über die Führungskräfte in den verschiedenen Phasen des Ereignisses.

Bei einem Vorfall ist es von entscheidender Bedeutung, die Eigentümer und Entwickler der betroffenen Anwendungen und Ressourcen einzubeziehen, da es sich um Fachexperten (SMEs) handelt, die Informationen und Zusammenhänge bereitstellen können, um die Auswirkungen zu messen. Üben Sie und bauen Sie Beziehungen zu den Entwicklern und Anwendungsbesitzern auf, bevor Sie sich bei der Vorfallreaktion auf deren Fachwissen verlassen. Anwendungsinhaber oder SMEs, wie Ihre Cloud-Administratoren oder Techniker, müssen möglicherweise in Situationen handeln, in denen die Umgebung nicht vertraut oder komplex ist oder in denen die Handelnden keinen Zugriff haben.

Schließlich könnten vertrauenswürdige Partner in die Untersuchung oder Reaktion einbezogen werden, da sie zusätzliches Fachwissen und wertvolle Einblicke bereitstellen können. Wenn Sie in Ihrem eigenen Team nicht über diese Fähigkeiten verfügen, sollten Sie eine externe Partei mit der Unterstützung beauftragen.

Die AWS-Reaktionsteams und der Support

- AWS -Support
  - Support bietet eine Reihe von Tarifen, die den Zugriff auf Tools und Fachwissen ermöglichen, um den Erfolg und die Betriebssicherheit Ihrer AWS-Lösungen zu unterstützen. Wenn Sie technischen Support und weitere Ressourcen benötigen, um Ihre AWS-Umgebung zu planen, bereitzustellen und zu optimieren, können Sie einen Supportplan auswählen, der am besten zu Ihrem AWS-Anwendungsfall passt.
  - Das Support-Center in der AWS Management Console (Anmeldung erforderlich) ist Ihre zentrale Anlaufstelle, um Unterstützung bei Problemen zu erhalten, die sich auf Ihre AWS-Ressourcen auswirken. Der Zugriff auf den Support wird über AWS Identity and Access Management gesteuert. Weitere Informationen zum Zugriff auf Support-Funktionen finden Sie unter Erste Schritte mit Support.
- AWS-Kundenvorfallreaktionsteam (CIRT)
  - Das AWS-Kundenvorfallreaktionsteam (CIRT) ist ein spezialisiertes globales, rund um die Uhr verfügbares AWS-Team, das Kunden bei aktiven Sicherheitsereignissen auf Kundenseite des AWS-Modells der geteilten Verantwortung.
  - Wenn das AWS-CIRT Sie unterstützt, bietet es Hilfe bei der Fehlererkennung und Wiederherstellung eines aktiven Sicherheitsereignisses auf AWS an. Sie können mithilfe von AWS-Serviceprotokollen bei der Ursachenanalyse helfen und Ihnen Empfehlungen für die Wiederherstellung geben. Sie können Ihnen auch Sicherheitsempfehlungen und bewährte Methoden an die Hand geben, mit denen Sie Sicherheitsereignisse in Zukunft vermeiden können.
  - AWS-Kunden können das AWS-CIRT über einen Support-Fall.
- Unterstützung für DDoS-Response
  - AWS bietet AWS Shield, das einen verwalteten Distributed Denial of Service (DDoS)-Schutzservice bereitstellt, der laufende Webanwendungen auf AWS schützt. Shield bietet eine ständig aktive Erkennung und automatische Inline-Schutzmaßnahmen, mit denen Ausfallzeiten und Latenz von Anwendungen minimiert werden können. Sie müssen also nicht Support kontaktieren, um vom DDoS-Schutz zu profitieren. Es gibt zwei Stufen von Shield: AWS Shield

Standard und AWS Shield Advanced. Weitere Informationen zu den Unterschieden zwischen diesen beiden Stufen finden Sie unter Shield-Funktionsdokumentation.

- AWS Managed Services (AMS)
  - AWS Managed Services (AMS) stellt eine fortlaufende Verwaltung Ihrer AWS-Infrastruktur bereit, damit Sie sich auf Ihre Anwendungen konzentrieren können. AMS trägt durch eine Implementierung bewährter Methoden zur Verwaltung Ihrer Infrastruktur dazu bei, den Betriebsaufwand zu reduzieren und das Risiko zu senken. Außerdem automatisiert AMS häufige Aktivitäten wie Änderungsanforderungen, Überwachung, Patch-Verwaltung, Sicherheit sowie Backup-Services und bietet während der gesamten Lebensdauer Services zum Bereitstellen, Ausführen und Unterstützen Ihrer Infrastruktur.
  - AMS übernimmt die Verantwortung für die Bereitstellung einer Reihe von Sicherheitskontrollen und bietet rund um die Uhr Erstreaktion auf Warnmeldungen an. Wenn eine Warnung ausgelöst wird, befolgt AMS eine Reihe automatisierter und manueller Standard-Playbooks, um sicherzustellen, dass eine konsistente Reaktion gewährleistet ist. Diese Playbooks werden den AMS-Kunden während des Onboardings zur Verfügung gestellt, damit sie eine Antwort entwickeln und mit AMS abstimmen können.

Erstellen des Vorfallreaktionsplans

Der Vorfallreaktionsplan ist als Grundlage für Ihr Vorfallreaktionsprogramm und Ihre Vorfallreaktionsstrategie konzipiert. Er sollte immer formell schriftlich festgehalten werden. Ein Vorfallreaktionsplan enthält in der Regel folgende Abschnitte:

- Ein Überblick über das Vorfallreaktionsteam: Er enthält die Ziele und Funktionen des Vorfallreaktionsteams.
- Rollen und Zuständigkeiten: Hier sind die für die Vorfallreaktion zuständigen Interessenvertreter aufgeführt und ihre Rollen im Falle eines Vorfalls werden beschrieben.
- Ein Kommunikationsplan: Dieser enthält Kontaktinformationen und gibt an, wie Sie während eines Vorfalls kommunizieren.
- Alternative Kommunikationsmethoden: Es hat sich bewährt, Out-of-Band-Kommunikation als Backup für die Kommunikation bei Vorfällen zu verwenden. Ein Beispiel für eine Anwendung, die einen sicheren Out-of-Band-Kommunikationskanal bereitstellt, ist AWS Wickr.
- Phasen der Vorfallreaktion und zu ergreifende Maßnahmen: Hier sind die Phasen der Vorfallreaktion aufgeführt (z. B. Erkennung, Analyse, Beseitigung, Eindämmung und Wiederherstellung), einschließlich der in diesen Phasen zu ergreifenden allgemeinen Maßnahmen.

- Definitionen des Schweregrads und der Priorisierung des Vorfalls: Hier wird erläutert, wie der Schweregrad eines Vorfalls klassifiziert wird, wie der Vorfall priorisiert wird und wie sich die Schweregraddefinitionen dann auf die Eskalationsverfahren auswirken.

Diese Abschnitte sind zwar in Unternehmen verschiedener Größen und Branchen üblich, der Vorfallreaktionsplan ist jedoch für jedes Unternehmen einzigartig. Sie müssen einen Vorfallreaktionsplan erstellen, der für Ihr Unternehmen am besten geeignet ist.

Ressourcen

Zugehörige bewährte Methoden:

- SEC04 (Wie erkenne und untersuche ich Sicherheitsereignisse?)

Zugehörige Dokumente:

- Leitfaden für AWS Security Incident Response
- NIST: Computer Security Incident Handling Guide

SEC10-BP03 Vorbereiten forensischer Funktionen

Im Vorfeld eines Sicherheitsvorfalls sollten Sie erwägen, forensische Funktionen zur Unterstützung der Untersuchung von Sicherheitsereignissen zu entwickeln.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Konzepte aus der traditionellen On-Premises-Forensik gelten für AWS. Wichtige Informationen für den Einstieg in den Aufbau forensischer Funktionen finden Sie AWS Cloud in den Strategien für forensische Untersuchungsumgebungen in der AWS Cloud.

Sobald Sie Ihre Umgebung und AWS-Konto-Struktur für die Forensik eingerichtet haben, definieren Sie die Technologien, die für die effektive Durchführung forensisch fundierter Methoden in den vier Phasen erforderlich sind:

- Sammlung: Erfassen Sie relevante AWS-Protokolle wie AWS CloudTrail, AWS Config, VPC Flow Logs und Protokolle auf Host-Ebene. Erfassen Sie Snapshots, Backups und Speicherabbilder der betroffenen AWS-Ressourcen, sofern verfügbar.
- Prüfung: Prüfen Sie die erfassten Daten, indem Sie die relevanten Informationen extrahieren und bewerten.

- Analyse: Analysieren Sie die erfassten Daten, um den Vorfall zu verstehen und daraus Schlüsse zu ziehen.
- Berichterstellung: Präsentieren Sie die Informationen, die sich aus der Analysephase ergeben.

Implementierungsschritte

Vorbereiten Ihrer forensischen Umgebung

AWS Organizations hilft Ihnen bei der zentralen Verwaltung und Steuerung einer AWS-Umgebung, während Sie AWS-Ressourcen erweitern und skalieren. Eine AWS-Organisation konsolidiert Ihre AWS-Konten, sodass Sie sie als eine einzige Einheit verwalten können. Sie können Organisationseinheiten (OEs) verwenden, um Konten zu gruppieren und als eine einzige Einheit zu verwalten.

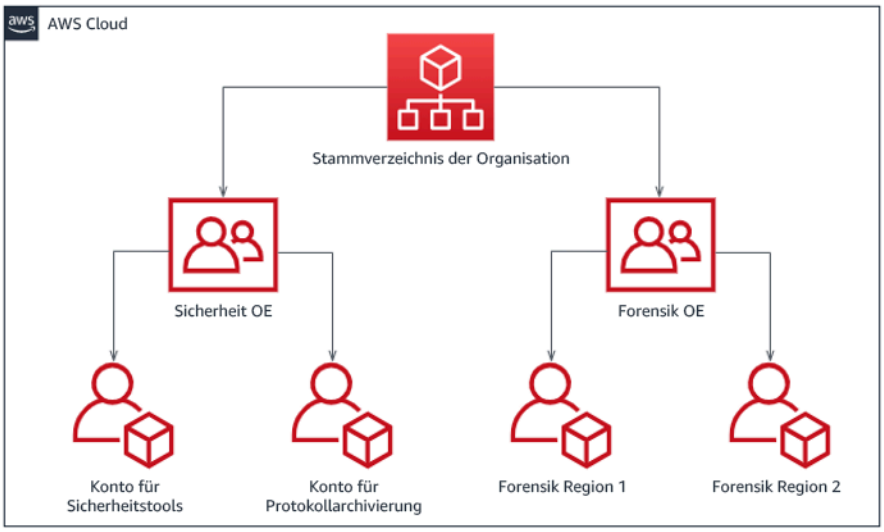
Für die Reaktion auf Vorfälle ist es hilfreich, eine AWS-Konto-Struktur zu haben, die die Funktionen der Vorfallsreaktion unterstützt. Dazu gehören eine Sicherheits-OE und eine Forensik-OE. Innerhalb der Sicherheits-OE sollten Sie Konten für Folgendes haben:

- Archivierung des Protokolls: Aggregieren Sie Protokolle in einem AWS-Konto für Protokollarchivierung mit eingeschränkten Berechtigungen.
- Sicherheitstools: Zentralisieren Sie Sicherheitsservices in einem AWS-Konto für Sicherheitstools. Dieses Konto fungiert als delegierter Administrator für Sicherheitsservices.

Innerhalb der Forensik-OE haben Sie die Möglichkeit, für jede Region, in der Sie tätig sind, ein oder mehrere forensische Konten zu implementieren, je nachdem, welche für Ihr Geschäfts- und Betriebsmodell am besten geeignet ist. Wenn Sie ein forensisches Konto pro Region erstellen, können Sie die Erstellung von AWS-Ressourcen außerhalb dieser Region blockieren und so das Risiko verringern, dass Ressourcen in eine unbeabsichtigte Region kopiert werden. Wenn Sie beispielsweise nur in US East (N. Virginia) Region (`us-east-1`) und US West (Oregon) (`us-west-2`) arbeiten, hätten Sie zwei Konten in der forensischen Organisationseinheit: eine für `us-east-1` und eine für `us-west-2`.

Sie können ein forensisches AWS-Konto für mehrere Regionen erstellen. Sie sollten beim Kopieren von AWS-Ressourcen auf dieses Konto Vorsicht walten lassen, um sicherzustellen, dass Sie Ihre Anforderungen an die Datensouveränität einhalten. Da die Bereitstellung neuer Konten einige Zeit in Anspruch nimmt, ist es unerlässlich, die forensischen Konten rechtzeitig vor einem Vorfall einzurichten und zu instrumentieren, damit die Notfallteams darauf vorbereitet sind, sie effektiv für die Reaktion zu nutzen.

Das folgende Diagramm zeigt eine Beispiel-Kontenstruktur mit einer Forensik-OE mit regionalen forensischen Konten:



Regionale Kontenstruktur für die Vorfallsreaktion

Erfassen von Backups und Snapshots

Die Einrichtung von Backups wichtiger Systeme und Datenbanken ist für die Wiederherstellung nach einem Sicherheitsvorfall und für forensische Zwecke von entscheidender Bedeutung. Mit vorhandenen Backups können Sie Ihre Systeme in ihren vorherigen sicheren Zustand zurückversetzen. In AWS können Sie Snapshots von verschiedenen Ressourcen erstellen. Snapshots bieten Ihnen zeitpunktbezogene Backups dieser Ressourcen. Es gibt viele AWS-Services, die Sie beim Backup und der Wiederherstellung unterstützen können. Einzelheiten zu diesen Services und Ansätzen für Backup und Wiederherstellung finden Sie unter Präskriptive Leitlinien für Backup und Wiederherstellung und Verwendung von Backups zur Wiederherstellung nach Sicherheitsvorfällen.

Vor allem, wenn es um Situationen wie Ransomware geht, ist es wichtig, dass Ihre Backups gut geschützt sind. Hinweise zur Sicherung Ihrer Backups finden Sie in den 10 besten Sicherheitsmethoden für die Sicherung von Backups in AWS. Zusätzlich zur Sicherung Ihrer Backups sollten Sie Ihre Backup- und Wiederherstellungsprozesse regelmäßig testen, um sicherzustellen, dass die vorhandenen Technologien und Prozesse wie erwartet funktionieren.

Automatisieren der Forensik

Während eines Sicherheitsereignisses muss Ihr Vorfallsreaktionsteam in der Lage sein, schnell Nachweise zu sammeln und zu analysieren und gleichzeitig die Genauigkeit für den Zeitraum rund

um das Ereignis aufrechtzuerhalten (z. B. das Erfassen von Protokollen zu einem bestimmten Ereignis oder einer bestimmten Ressource oder das Erfassen von Speicherabbildern einer Amazon EC2-Instance). Für das Vorfallsreaktionsteam ist es sowohl schwierig als auch zeitaufwändig, die relevanten Beweise manuell zu erfassen, insbesondere bei einer großen Anzahl von Instances und Konten. Darüber hinaus kann die manuelle Erfassung anfällig für menschliche Fehler sein. Aus diesen Gründen sollten Sie die Automatisierung für die Forensik so weit wie möglich entwickeln und implementieren.

AWS bietet eine Reihe von Automatisierungsressourcen für die Forensik, die im Abschnitt Ressourcen unten aufgeführt sind. Diese Ressourcen sind Beispiele für forensische Muster, die wir entwickelt und Kunden implementiert haben. Obwohl sie für den Anfang eine nützliche Referenzarchitektur sein können, sollten Sie erwägen, sie zu ändern oder neue forensische Automatisierungsmuster zu erstellen, die auf Ihrer Umgebung, Ihren Anforderungen, Tools und forensischen Prozessen basieren.

Ressourcen

Zugehörige Dokumente:

- AWS-Sicherheits- und Vorfallsreaktionsanleitung – Forensische Funktionen entwickeln
- AWS-Sicherheits- und Vorfallsreaktionsanleitung – Forensische Ressourcen
- Strategien für forensische Untersuchungsumgebungen in der AWS Cloud
- Automatisieren der forensischen Datenträgererfassung in AWS
- Präskriptive AWS-Anleitung – Automatisieren der Vorfallsreaktion und Forensik

Zugehörige Videos:

- Automating Incident Response and Forensics

Zugehörige Beispiele:

- Framework für automatisierte Vorfallsreaktion und Forensik
- Automatisierter forensischer Orchestrator für Amazon EC2

SEC10-BP04 Entwickeln und Testen von Playbooks für die Reaktion auf Sicherheitsvorfälle

Ein wichtiger Teil der Vorbereitung Ihrer Prozesse zur Vorfallsreaktion ist die Entwicklung von Playbooks. Playbooks für die Vorfallsreaktion enthalten eine Reihe von präskriptiven Anleitungen und Schritten, die Sie befolgen müssen, wenn ein Sicherheitsereignis eintritt. Eine klare Struktur und klare Schritte vereinfachen die Reaktion und verringern die Wahrscheinlichkeit menschlicher Fehler.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Playbooks sollten für Vorfallsszenarien wie die folgenden erstellt werden:

- Erwartete Vorfälle: Sie sollten Playbooks für zu erwartende Vorfälle erstellen. Dazu gehören Bedrohungen wie Denial of Service (DoS), Ransomware und die Kompromittierung von Anmeldeinformationen.
- Bekannte Sicherheitserkenntnisse oder Warnungen: Sie sollten Playbooks für Ihre bekannten Sicherheitserkenntnisse und Warnmeldungen wie GuardDuty-Ergebnisse erstellen. Möglicherweise erhalten Sie eine GuardDuty-Erkenntnis und denken: „Wie geht es weiter?“ Um zu verhindern, dass eine GuardDuty-Erkenntnis unsachgemäß gehandhabt oder ignoriert wird, sollten Sie für jede potenzielle GuardDuty-Erkenntnis ein Playbook erstellen. Einige Einzelheiten und Anleitungen zur Mängelbeseitigung finden Sie in der GuardDuty-Dokumentation. Es ist erwähnenswert, dass GuardDuty standardmäßig nicht aktiviert ist und dafür Kosten anfallen. Weitere Informationen finden GuardDuty Sie in Anhang A: Definitionen der Cloud-Funktionen –Sichtbarkeit und Warnmeldungen.

Playbooks sollten technische Schritte enthalten, die ein Sicherheitsanalyst ausführen muss, um einen potenziellen Sicherheitsvorfall angemessen zu untersuchen und darauf zu reagieren.

Implementierungsschritte

Zu den Elementen, die in ein Playbook aufgenommen werden sollten, gehören:

- Playbook-Übersicht: Welches Risiko- oder Vorfallszenario behandelt dieses Playbook? Was ist das Ziel des Playbooks?
- Voraussetzungen: Welche Protokolle, Erkennungsmechanismen und automatisierten Tools sind für dieses Vorfallszenario erforderlich? Wie lautet die erwartete Benachrichtigung?
- Kommunikations- und Eskalationsinformationen: Wer ist beteiligt und wie lauten ihre Kontaktinformationen? Welche Verantwortlichkeiten haben die einzelnen Interessenvertreter?

- Reaktionsschritte: Welche taktischen Maßnahmen sollten in allen Phasen der Vorfallsreaktion ergriffen werden? Welche Abfragen sollte ein Analyst ausführen? Welcher Code sollte ausgeführt werden, um das gewünschte Ergebnis zu erzielen?
  - Erkennen: Wie wird der Vorfall erkannt?
  - Analysieren: Wie wird der Umfang der Auswirkungen bestimmt?
  - Eindämmen: Wie wird der Vorfall isoliert, um den Umfang zu begrenzen?
  - Beseitigen: Wie wird die Bedrohung aus der Umgebung entfernt?
  - Wiederherstellen: Wie wird das betroffene System oder die betroffene Ressource wieder in der Produktion bereitgestellt?
- Erwartete Ergebnisse: Was ist das erwartete Ergebnis des Playbooks, nachdem Abfragen und Code ausgeführt wurden?

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- SEC10-BP02 – Entwickeln von Vorfallmanagementplänen

Zugehörige Dokumente:

- Framework für Playbooks für die Vorfallsreaktion
- Entwickeln eigener Playbooks für die Vorfallsreaktion
- Beispiele von Playbooks für die Vorfallsreaktion
- Entwicklung eines Runbooks für die Vorfallsreaktion in AWS mit Jupyter Playbooks und CloudTrail Lake

SEC10-BP05 Vorab bereitgestellter Zugriff

Stellen Sie sicher, dass Notfallteams über den richtigen vorab bereitgestellten Zugriff in AWS verfügen, um die Zeit von der Untersuchung bis zur Wiederherstellung zu verkürzen.

Typische Anti-Muster:

- Verwenden des Root-Kontos für die Reaktion auf Vorfälle
- Verändern bestehender Benutzerkonten

- Direkte Manipulation von IAM-Berechtigungen bei Bereitstellung von Just-in-time-Berechtigungserhöhungen

Risikostufe, wenn diese bewährte Methode nicht genutzt wird: Mittel

Implementierungsleitfaden

AWS empfiehlt die Reduzierung oder Ausschaltung der Abhängigkeit von langlebigen Anmeldeinformationen wenn möglich und ihren Ersatz durch Just-in-Time-Berechtigungseskalationsmechanismen. Langlebige Anmeldeinformationen sind anfällig für Sicherheitsrisiken und erhöhen den Verwaltungsaufwand. Für die meisten Managementaufgaben sowie für Vorfallreaktionsaufgaben empfehlen wir die Implementierung eines Identitätsverbunds neben der temporären Eskalierung für den administrativen Zugriff. In diesem Modell beantragt ein Benutzer seine Erhöhung auf eine höhere Berechtigungsstufe (etwa zu einer Vorfallreaktionsrolle). Anschließend wird, sofern der Benutzer grundsätzlich dafür infrage kommt, eine Anfrage an einen Genehmiger gesendet. Wenn die Anfrage genehmigt wurde, erhält der Benutzer einen Satz temporärer AWS-Anmeldeinformationen für die Durchführung seiner Aufgaben. Wenn diese Anmeldeinformationen ablaufen, muss der Benutzer eine neue Erhöhungsanfrage stellen.

Wir empfehlen für die meisten Vorfallreaktionsszenarien die Verwendung temporärer Berechtigungseskalierungen. Die korrekte Vorgehensweise ist die Verwendung von AWS Security Token Service und von Sitzungsrichtlinien zur Festlegung der Zugriffsbereiche.

Es gibt Szenarien, in denen Verbundidentitäten nicht verfügbar sind, zum Beispiel:

- Ausfall durch Problem mit einem Identitätsanbieter (IdP)
- Fehlerhafte Konfiguration oder menschlicher Fehler, die/der das Managementsystem für den Verbundzugriff beschädigt
- Böswillige Aktivität, z. B. ein DDoS-Angriff (Distributed Denial of Service) oder anderweitig verursachte Nichtverfügbarkeit des Systems

Für diese Fälle sollte Notfall- „Break Glass“- Zugriff konfiguriert werden, um Untersuchungen und die schnelle Behebung des Vorfalls zu ermöglichen. Wir empfehlen die Verwendung eines IAM-Benutzers mit ausreichenden Berechtigungen für die Durchführung von Aufgaben und den Zugriff auf AWS-Ressourcen. Verwenden Sie die Root-Anmeldeinformationen nur für Aufgaben, die Root-Benutzerzugriff erfordern. Zur Prüfung, ob die Vorfallreaktionskräfte über die korrekte Zugriffsstufe auf AWS und andere relevante Systeme verfügen, empfehlen wir die Bereitstellung dedizierter Benutzerkonten. Die Benutzerkonten erfordern privilegierten Zugriff und müssen eng kontrolliert und

überwacht werden. Die Konten müssen mit den geringstmöglichen Berechtigungen versehen sein, die für die erforderlichen Aufgaben benötigt werden, und die Zugriffsstufe muss auf den Playbooks basieren, die Teil des Vorfallmanagementplans sind.

Verwenden Sie als bewährte Methode zweckgerichtet erstellte und dedizierte Benutzer und Rollen. Die vorübergehende Eskalierung des Zugriffs eines Benutzers oder einer Rolle über IAM-Richtlinien macht es unklar, welche Zugriffsmöglichkeiten Benutzer während eines Vorfalls hatten, und birgt die Gefahr, dass die eskalierten Berechtigungen später nicht widerrufen werden.

Es ist wichtig, so viele Abhängigkeiten wie möglich zu entfernen, um sicherzustellen, dass Zugriff bei einer möglichst großen Anzahl von Ausfallszenarien möglich ist. Erstellen Sie deshalb ein Playbook, um sicherzustellen, dass Vorfallreaktionsbenutzer als AWS Identity and Access Management-Benutzer in einem dedizierten Sicherheitskonto erstellt und nicht durch einen vorhandenen Verbund oder eine Single Sign-On (SSO)-Lösung verwaltet werden. Alle einzelnen Reaktionskräfte müssen ein eigenes benanntes Konto haben. Die Kontokonfiguration muss eine Richtlinie für sichere Passwörter und Multi-Faktor-Authentifizierung (MFA) durchsetzen. Wenn die Playbooks zur Vorfallreaktion nur Zugriff auf die AWS Management Console benötigen, sollten für den Benutzer keine Zugriffsschlüssel konfiguriert werden und er sollte auch explizit keine Zugriffsschlüssel erstellen dürfen. Dies kann mit IAM-Richtlinien oder Service-Kontrollrichtlinien (SCPs) konfiguriert werden, wie in den bewährten AWS-Sicherheitsmethoden für AWS Organizations SCPs erläutert. Die Benutzer sollten keine Berechtigungen außer der Möglichkeit zur Übernahme von Vorfallreaktionsrollen in anderen Konten haben.

Während eines Vorfalls kann es erforderlich sein, anderen internen oder externen Personen Zugriff zu gewähren, um Untersuchungs-, Korrektur- oder Wiederherstellungsaktivitäten zu unterstützen. Verwenden Sie in diesem Fall den vorher erwähnten Playbook-Mechanismus. Darüber hinaus muss ein Prozess vorhanden sein, um sicherzustellen, dass jeglicher zusätzliche Zugriff sofort nach Abschluss des Vorfalls widerrufen wird.

Zur Sicherstellung, dass die Verwendung von Vorfallreaktionsrollen in korrekter Weise überwacht und geprüft werden kann, ist es entscheidend, dass die für diesen Zweck erstellten IAM-Benutzerkonten nicht zwischen Personen weitergegeben werden und dass der AWS-Konto-Root-Benutzer nicht verwendet wird, sofern dies nicht für eine bestimmte Aufgabe erforderlich ist. Wenn der Root-Benutzer erforderlich ist (zum Beispiel wenn der IAM-Zugriff auf ein bestimmtes Konto nicht verfügbar ist), verwenden Sie einen separaten Prozess mit einem Playbook, um die Verfügbarkeit des Root-Benutzer-Passworts und des MFA-Tokens zu prüfen.

Erwägen Sie zur Konfiguration der IAM-Richtlinien für die Vorfallreaktionsrollen die Verwendung von IAM Access Analyzer zum Erstellen von Richtlinien auf der Grundlage von AWS CloudTrail-

Protokollen. Gewähren Sie dazu der Vorfallreaktionsrolle in einem Nicht-Produktionskonto Administratorzugriff und durchlaufen Sie das Playbook. Sobald dies geschehen ist, kann eine Richtlinie erstellt werden, die nur die entsprechenden Aktionen zulässt. Diese Richtlinie kann dann auf alle Vorfallreaktionsrollen über alle Konten hinweg angewendet werden. Möglicherweise möchten Sie eine separate IAM-Richtlinie für jedes Playbook erstellen, um Management und Auditing zu vereinfachen. Beispiel-Playbooks können Reaktionspläne für Ransomware-Angriffe, Datenschutzverletzungen, Verlust von produktionsrelevantem Zugriff oder andere Szenarien enthalten.

Verwenden Sie die Vorfallreaktionsbenutzerkonten zur Annahme dedizierter Vorfallreaktions-IAM-Rollen in anderen AWS-Konten. Diese Rollen müssen so konfiguriert sein, dass sie nur von Benutzern im Sicherheitskonto angenommen werden können, und das Vertrauensverhältnis muss erfordern, dass der aufrufende Prinzipal per MFA authentifiziert wurde. Die Rollen müssen eng gefasste IAM-Richtlinien verwenden, um den Zugriff zu kontrollieren. Stellen Sie sicher, dass alle `AssumeRole-` Anfragen für diese Rollen in CloudTrail protokolliert und gemeldet werden und dass alle mit diesen Rollen durchgeführten Aktivitäten protokolliert werden.

Es wird nachdrücklich empfohlen, die IAM-Benutzerkonten und die IAM-Rollen deutlich zu benennen, damit sie in CloudTrail-Protokollen leicht zu finden sind. Ein Beispiel ist die Benennung der IAM-Konten als `<USER_ID>-BREAK-GLASS` und der IAM-Rollen als `BREAK-GLASS-ROLE`.

CloudTrail wird verwendet, um API-Aktivitäten in Ihren AWS-Konten zu protokollieren, und sollte zur Konfiguration von Alarmen zur Nutzung der Vorfallreaktionsrollen eingesetzt werden. Weitere Informationen finden Sie im Blog-Beitrag zur Konfiguration von Alarmen bei Verwendung von Root-Schlüsseln. Die Anweisungen können können geändert werden, um die Metrik Amazon CloudWatch so zu konfigurieren, dass sie nach `AssumeRole-` Ereignissen gefiltert wird, die mit der Vorfallreaktions-IAM-Rolle zusammenhängen.

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =
 "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !
= "AwsServiceEvent" }
```

Da die Vorfallreaktionsrollen sehr wahrscheinlich eine hohe Zugriffsstufe haben, ist es wichtig, dass diese Alarme an eine breite Gruppe gehen und dass sofort darauf reagiert wird.

Während eines Vorfalls kann es geschehen, dass eine Reaktionskraft Zugriff auf Systeme benötigt, die nicht direkt von IAM gesichert sind. Dazu können Amazon Elastic Compute Cloud-Instances, Amazon Relational Database Service-Datenbanken oder SaaS-Plattformen gehören. Es wird nachdrücklich empfohlen, anstelle nativer Protokolle wie SSH oder RDP AWS Systems Manager

Session Manager für alle administrativen Zugriffe auf Amazon EC2-Instances zu verwenden. Dieser Zugriff kann mit IAM (sicher und geprüft) kontrolliert werden. Es kann auch möglich sein, Teile Ihrer Playbooks mit AWS Systems Manager-Run-Command-Dokumentenzu automatisieren, wodurch sich möglicherweise Benutzerfehler reduzieren und Wiederherstellungszeiten verkürzen lassen. Für den Zugriff auf Datenbanken und Tools von Drittanbietern empfehlen wir die Speicherung von Anmeldeinformationen in AWS Secrets Manager und die Gewährung des Zugriffs auf die Vorfallreaktionsrollen.

Schließlich sollte die Verwaltung der Vorfallreaktions-IAM-Benutzerkonten Ihren Joiners-, Movers- und Leavers-Prozessen hinzugefügt sowie regelmäßig geprüft und getestet werden, um sicherzustellen, dass nur die beabsichtigten Zugriffsrechte gewährt werden.

Ressourcen

Zugehörige Dokumente:

- Verwaltung des vorübergehend erhöhten Zugriffs auf Ihre AWS-Umgebung
- Leitfaden für AWS Security Incident Response
- AWS Elastic Disaster Recovery
- AWS Systems Manager Incident Manager
- Einrichten einer Kontopasswortrichtlinie für IAM-Benutzer
- Verwenden der Multi-Faktor-Authentifizierung (MFA) in AWS
- Konfigurieren des kontoübergreifenden Zugriffs mit MFA
- Verwenden von IAM Access Analyzer zum Erstellen von IAM-Richtlinien
- Bewährte Methoden für AWS Organizations-Servicekontrollrichtlinien in einer Mehrkontenumgebung
- Empfang von Benachrichtigungen, wenn die Root-Zugriffsschlüssel Ihres AWS-Kontos verwendet werden
- Erstellen detaillierter Sitzungsberechtigungen mithilfe von IAM-verwalteten Richtlinien

Zugehörige Videos:

- Automating Incident Response and ForensicsAWS (Automatisieren der Vorfallreaktion und Forensik in AWS)
- DIY guide to runbooks, incident reports, and incident response (DIY-Leitfaden für Runbooks, Vorfallberichte und Vorfallreaktion)

- Prepare for and respond to security incidents in your AWS environment (Vorbereiten und Reagieren auf Sicherheitsvorfälle in Ihrer AWS-Umgebung)

Zugehörige Beispiele:

- Übung: AWS-Kontoeinrichtung und Root-Benutzer
- Übung: Vorfallreaktion mit AWS-Konsole und CLI

SEC10-BP06 Vorabbereitstellen von Tools

Stellen Sie sicher, dass Sicherheitspersonal über die richtigen Tools verfügt, um die Zeit von der Untersuchung bis zur Wiederherstellung zu verkürzen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Zur Automatisierung von Sicherheitsreaktionen und Betriebsfunktionen können Sie eine umfassende Palette von APIs und Tools von AWS verwenden. Sie können die Identitätsverwaltung, Netzwerksicherheit, Datenschutz und Überwachungsfunktionen vollständig automatisieren und diese mithilfe gängiger Softwareentwicklungsmethoden bereitstellen, die Sie bereits eingerichtet haben. Wenn Sie die Sicherheitsautomatisierung erstellen, kann Ihr System eine Reaktion überwachen, prüfen und initiieren, statt nur Ihre Sicherheitslage zu überwachen und manuell auf Ereignisse zu reagieren.

Wenn Ihre Vorfallreaktionsteams auf Warnungen weiterhin auf die gleiche Weise reagieren, riskieren sie eine Abstumpfung der Warnung. Im Laufe der Zeit kann das Team für Warnungen desensibilisiert werden und entweder Fehler bei der Verarbeitung normaler Situationen machen oder außergewöhnliche Warnungen übersehen. Automatisierung hilft, eine Abstumpfung von Warnungen zu vermeiden, indem Funktionen verwendet werden, die sich wiederholende und gewöhnliche Warnungen verarbeiten, sodass Mitarbeiter die nötigen freien Kapazitäten haben, um sich um sensible und einzigartige Vorfälle zu kümmern. Die Integration von Systemen zur Erkennung von Anomalien wie Amazon GuardDuty, AWS CloudTrail Insights und Amazon CloudWatch Anomaly Detection kann den durch schwellenwertbasierte Warnmeldungen verursachten Aufwand reduzieren.

Sie können manuelle Prozesse verbessern, indem Sie die Schritte im Prozess automatisieren. Nachdem Sie das Korrekturmuster für ein Ereignis definiert haben, können Sie dieses Muster in umsetzbare Logik zerlegen und den Code schreiben, um diese Logik auszuführen. Notfallteams

können anschließend diesen Code ausführen, um das Problem zu beheben. Mit der Zeit können Sie immer mehr Schritte automatisieren und schließlich häufige Vorfälle automatisch verarbeiten.

Bei einer Sicherheitsuntersuchung müssen Sie relevante Protokolle konsultieren können, um alle Aspekte und den Zeitrahmen des Vorfalls zu verstehen. Protokolle werden auch für die Generierung von Alarmen benötigt, die darauf hinweisen, dass bestimmte Ereignisse vorgekommen sind. Es ist sehr wichtig, Abfrage- und Abrufmechanismen auszuwählen, zu aktivieren, zu speichern und einzurichten sowie die Alarmierung einzurichten. Darüber hinaus besteht eine effektive Möglichkeit zur Nutzung von Tools zum Durchsuchen von Protokolldaten in Amazon Detective.

AWS bietet über 200 Cloud-Services und Tausende von Funktionen. Wir empfehlen Ihnen, die Services zu konsultieren, die Ihre Strategie zur Vorfallsreaktion unterstützen und vereinfachen können.

Zusätzlich zur Protokollierung sollten Sie eine Markierungsstrategie entwickeln und implementieren. Die Markierung kann dabei helfen, einen Kontext zum Zweck einer AWS-Ressource bereitzustellen. Die Markierung kann auch für die Automatisierung verwendet werden.

Implementierungsschritte

Auswählen und Einrichten von Protokollen für die Analyse und Alarmierung

In der folgenden Dokumentation finden Sie Informationen zur Konfiguration der Protokollierung für die Vorfallsreaktion:

- Protokollierungsstrategien für die Reaktion auf Sicherheitsvorfälle
- SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung

Aktivieren von Sicherheitsservices zur Unterstützung von Erkennung und Reaktion

AWS bietet native Erkennungs-, Präventions- und Reaktionsfunktionen, und andere Services können für den Aufbau benutzerdefinierter Sicherheitslösungen verwendet werden. Eine Liste der wichtigsten Services für die Reaktion auf Sicherheitsvorfälle finden Sie unter Definitionen der Cloud-Funktionen.

Entwickeln und Implementieren einer Markierungsstrategie

Es kann schwierig sein, kontextbezogene Informationen zum geschäftlichen Anwendungsfall und zu relevanten internen Interessenvertretern rund um eine AWS-Ressource zu erhalten. Eine Möglichkeit, dies zu tun, sind Tags, die Ihren AWS-Ressourcen Metadaten zuweisen und aus einem

benutzerdefinierten Schlüssel und Wert bestehen. Sie können Tags erstellen, um Ressourcen nach Zweck, Besitzer, Umgebung, Art der verarbeiteten Daten und anderen Kriterien Ihrer Wahl zu kategorisieren.

Eine konsistente Markierungsstrategie kann die Reaktionszeiten verkürzen und den Zeitaufwand für den organisatorischen Kontext minimieren, da Sie Kontextinformationen zu einer AWS-Ressource schnell identifizieren und erkennen können. Tags können auch als Mechanismus zur Initiierung von Reaktionsautomatisierungen dienen. Weitere Informationen über zu markierende Elemente finden Sie unter Markieren Ihrer AWS-Ressourcen. Sie sollten zunächst die Tags definieren, die Sie in Ihrer Organisation implementieren möchten. Danach implementieren Sie diese Markierungsstrategie und setzen sie durch. Weitere Einzelheiten zur Umsetzung und Durchsetzung finden Sie unter Implementieren einer Markierungsstrategie für AWS-Ressourcen mithilfe von AWS-Markierungsrichtlinien und Service-Kontrollrichtlinien (SCPs).

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung
- SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten

Zugehörige Dokumente:

- Protokollierungsstrategien für die Reaktion auf Sicherheitsvorfälle
- Cloud-Capability-Definitionen für die Vorfallsreaktion

Zugehörige Beispiele:

- Bedrohungserkennung und -reaktion mit Amazon GuardDuty und Amazon Detective
- Security-Hub-Workshop
- Management von Schwachstellen mit Amazon Inspector

SEC10-BP07 Durchführen von Simulationen

Ebenso wie Unternehmen im Laufe der Zeit wachsen und sich weiterentwickeln, wächst auch die Bedrohungslandschaft. Daher ist es wichtig, Ihre Fähigkeiten zur Vorfallreaktion kontinuierlich zu überprüfen. Die Durchführung von Simulationen (auch bekannt als Gamedays) ist eine

Methode, mit der diese Bewertung durchgeführt werden kann. Bei Simulationen werden reale Sicherheitsereignisse als Szenarien verwendet, die die Taktiken, Techniken und Verfahren (TTPs) eines Bedrohungsakteurs nachahmen und es einer Organisation ermöglichen, ihre Fähigkeiten zur Vorfallreaktion einzusetzen und zu bewerten, indem sie auf diese simulierten Cyberereignisse so reagieren, wie sie es im Ernstfall tun würden.

Vorteile der Nutzung dieser bewährten Methode: Simulationen haben eine Vielzahl von Vorteilen:

- Validierung der Cybersicherheit und Stärkung des Vertrauens Ihres Vorfallreaktionsteams
- Testen der Genauigkeit und Effizienz von Tools und Workflows
- Optimierung der Kommunikations- und Eskalationsmethoden Ihres Vorfallreaktionsplans
- Die Möglichkeit, auf weniger verbreitete Vektoren zu reagieren

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Es gibt drei Hauptarten von Simulationen:

- Tabletop-Übungen: Der Tabletop-Ansatz für Simulationen besteht aus einer Diskussionsrunde, in der die verschiedenen Interessenvertreter des Bereichs Vorfallreaktion teilnehmen, um Rollen und Verantwortlichkeiten zu üben und etablierte Kommunikationstools und Playbooks zu verwenden. Die Übung kann in der Regel an einem ganzen Tag an einem virtuellen Ort, einem physischen Veranstaltungsort oder einer Kombination daraus durchgeführt werden. Da sie auf Diskussionen basiert, konzentriert sich die Tabletop-Übung auf Prozesse, Menschen und Zusammenarbeit. Technologie ist ein integraler Bestandteil der Diskussion, aber der tatsächliche Einsatz von Tools oder Skripten für die Vorfallreaktion ist in der Regel kein Teil der praktischen Übung.
- Lila Teamübungen: Lila Teamübungen verbessern die Zusammenarbeit zwischen dem Vorfallreaktionsteam (blaues Team) und den simulierten Bedrohungsakteuren (rotes Team). Das blaue Team besteht aus Mitgliedern des Security Operations Center (SOC), kann aber auch andere Interessenvertreter einbeziehen, die an einem tatsächlichen Cyberereignis beteiligt wären. Das rote Team besteht aus einem Penetrationstest-Team oder wichtigen Interessenvertretern, die in offensiver Sicherheit trainiert sind. Das rote Team arbeitet bei der Planung eines Szenarios mit den Übungsleitern zusammen, damit das Szenario korrekt und durchführbar ist. Bei den lila Teamübungen liegt das Hauptaugenmerk auf den Erkennungsmechanismen, den Tools und den Standard-Betriebsabläufen (SOPs), mit denen die Maßnahmen zur Vorfallreaktion unterstützt werden.

- Übungen des roten Teams: Bei einer Übung des roten Teams führt das Offensivteam (rotes Team) eine Simulation durch, um ein bestimmtes Ziel oder eine Reihe von Zielen aus einem vorher festgelegten Umfang zu erreichen. Die Verteidiger (blaues Team) kennen nicht unbedingt den Umfang und die Dauer der Übung, was eine realistischere Einschätzung darüber ermöglicht, wie sie auf einen tatsächlichen Vorfall reagieren würden. Da es sich bei den Übungen des roten Teams um invasive Tests handeln kann, sollten Sie vorsichtig sein und Kontrollen implementieren, um sicherzustellen, dass die Übung Ihrer Umgebung nicht tatsächlich schadet.

Erwägen Sie, in regelmäßigen Abständen Cybersimulationen durchzuführen. Jeder Übungstyp kann den Teilnehmern und der gesamten Organisation einzigartige Vorteile bieten. Sie können also mit weniger komplexen Simulationstypen beginnen (z. B. mit Tabletop-Übungen) und zu komplexeren Simulationstypen übergehen (Übungen des roten Teams). Wählen Sie einen Simulationstyp anhand Ihres Sicherheitsgrads, Ihrer Ressourcen und der gewünschten Ergebnisse aus. Einige Kunden entscheiden sich aufgrund der Komplexität und der Kosten möglicherweise gegen Übungen des roten Teams.

Implementierungsschritte

Unabhängig von der Art der gewählten Simulation folgen diese im Allgemeinen den folgenden Implementierungsschritten:

1. Definieren Sie die wichtigsten Übungselemente: Definieren Sie das Simulationsszenario und die Ziele der Simulation. Beide sollten von den Führungskräften akzeptiert werden.
2. Identifizieren Sie die wichtigsten Interessenvertreter: Für eine Übung sind mindestens Übungsleiter und Teilnehmer erforderlich. Je nach Szenario können weitere Interessengruppen wie Recht, Kommunikation oder Geschäftsleitung einbezogen werden.
3. Erstellen und testen Sie das Szenario: Das Szenario muss möglicherweise während der Erstellung neu definiert werden, falls bestimmte Elemente nicht realisierbar sind. Als Ergebnis dieser Phase wird ein fertiges Szenario erwartet.
4. Führen Sie die Simulation durch: Die Art der Simulation bestimmt die Durchführung (ein Szenario auf Papier im Vergleich zu einem hochtechnischen, simulierten Szenario). Die Übungsleiter sollten ihre Moderationstaktiken an den Übungsobjekten ausrichten und alle Übungsteilnehmer nach Möglichkeit einbeziehen, um den größtmöglichen Nutzen zu erzielen.
5. Arbeiten Sie den After-Action Report (AAR, Abschlussbericht) aus: Identifizieren Sie Bereiche, die gut gelaufen sind, diejenigen, die verbessert werden können, und potenzielle Lücken. Der AAR sollte die Effektivität der Simulation sowie die Reaktion des Teams auf das simulierte Ereignis messen, damit der Fortschritt mit zukünftigen Simulationen im Laufe der Zeit verfolgt werden kann.

Ressourcen

Zugehörige Dokumente:

- AWS Incident Response Guide

Zugehörige Videos:

- AWS GameDay – Sicherheitsausgabe

SEC10-BP08 Entwickeln eines Frameworks, um aus Vorfällen zu lernen

Die Implementierung eines Erkenntnis-Frameworks für Erkenntnisse und der Fähigkeit zur Ursachenanalyse trägt nicht nur dazu bei, die Reaktionsfähigkeit auf Vorfälle zu verbessern, sondern auch zu verhindern, dass sich der Vorfall wiederholt. Indem Sie aus jedem Vorfall lernen, können Sie verhindern, dass dieselben Fehler, Risiken oder Fehlkonfigurationen wiederholt werden. Dies verbessert nicht nur Ihre Sicherheitslage, sondern minimiert auch den Zeitverlust durch vermeidbare Situationen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Die Implementierung eines Erkenntnis-Frameworks ist wichtig, der die folgenden Punkte allgemein festlegt und erreicht:

- Wann finden Erkenntnisse statt?
- Was beinhaltet der Erkenntnisprozess?
- Wie werden Erkenntnisse durchgeführt?
- Wer ist am Prozess beteiligt und wie?
- Wie werden verbesserungswürdige Bereiche identifiziert?
- Wie stellen Sie sicher, dass Verbesserungen effektiv verfolgt und implementiert werden?

Das Framework sollte sich nicht auf Einzelpersonen konzentrieren oder ihnen die Schuld geben, sondern stattdessen den Fokus auf die Verbesserung der Tools und Prozesse legen.

Implementierungsschritte

Abgesehen von den zuvor aufgeführten Ergebnissen auf hoher Ebene ist es wichtig, sicherzustellen, dass Sie die richtigen Fragen stellen, um den größtmöglichen Nutzen (Informationen, die zu umsetzbaren Verbesserungen führen) aus dem Prozess zu ziehen. Beachten Sie die folgenden Fragen, um Ihre Diskussionen über Erkenntnisse zu fördern:

- Was ist vorgefallen?
- Wann wurde der Vorfall zum ersten Mal identifiziert?
- Wie wurde er identifiziert?
- Welche Systeme haben über die Aktivität alarmiert?
- Welche Systeme, Services und Daten waren beteiligt?
- Was ist konkret passiert?
- Was hat gut funktioniert?
- Was hat nicht gut funktioniert?
- Welcher Prozess oder welche Verfahren haben versagt oder konnten nicht skaliert werden, um auf den Vorfall zu reagieren?
- Was kann in den folgenden Bereichen verbessert werden:
  - Mitarbeiter
    - Waren die Mitarbeiter, die kontaktiert werden mussten, tatsächlich verfügbar und war die Kontaktliste auf dem neuesten Stand?
    - Fehlten den Mitarbeitern Trainings oder Fähigkeiten, die erforderlich waren, um effektiv auf den Vorfall reagieren und ihn untersuchen zu können?
    - Waren die erforderlichen Ressourcen bereit und verfügbar?
  - Prozess
    - Wurden Prozesse und Verfahren eingehalten?
    - Wurden Prozesse und Verfahren für diese(n) (Art von) Vorfall dokumentiert und waren sie dafür verfügbar?
    - Fehlten die erforderlichen Prozesse und Verfahren?
    - Konnten die Notfallteams rechtzeitig auf die erforderlichen Informationen zugreifen, um auf das Problem zu reagieren?
  - Technologie
    - Haben die bestehenden Warnsysteme die Aktivität effektiv identifiziert und gemeldet?

- Wie hätten wir die Zeit bis zur Erkennung um 50 % reduzieren können?
- Müssen bestehende Warnungen verbessert werden oder müssen neue Warnungen für diese(n) (Art von) Vorfall erstellt werden?
- Ermöglichten die vorhandenen Tools eine effektive Untersuchung (Suche/Analyse) des Vorfalls?
- Was kann getan werden, um diese(n) (Art von) Vorfall früher zu erkennen?
- Was kann getan werden, um zu verhindern, dass sich diese(r) (Art von) Vorfall wiederholt?
- Wem gehört der Verbesserungsplan und wie testen Sie, ob er umgesetzt wurde?
- Wie sieht der Zeitplan für die Implementierung und das Testen zusätzlicher Überwachungs- oder präventiver Kontrollen und Prozesse aus?

Diese Liste ist nicht vollständig, soll aber als Ausgangspunkt dienen, um zu ermitteln, was die Organisations- und Geschäftsanforderungen sind und wie Sie diese analysieren können, um am effektivsten aus Vorfällen zu lernen und Ihre Sicherheitslage kontinuierlich zu verbessern. Am wichtigsten ist es, zunächst die Erkenntnisse als Standardbestandteil Ihres Prozesses zur Vorfallsreaktion, der Dokumentation und der Erwartungen der Interessenvertreter zu berücksichtigen.

Ressourcen

Zugehörige Dokumente:

- AWS-Sicherheits- und Vorfallreaktionsanleitung – Entwickeln eines Frameworks, um aus Vorfällen zu lernen
- NCSC-CAF-Leitfaden – Erkenntnisse

Anwendungssicherheit

Frage

- SEC 11. Wie beziehen Sie die Sicherheitseigenschaften von Anwendungen während des gesamten Entwurfs-, Entwicklungs- und Bereitstellungslebenszyklus ein und validieren sie?

SEC 11. Wie beziehen Sie die Sicherheitseigenschaften von Anwendungen während des gesamten Entwurfs-, Entwicklungs- und Bereitstellungslebenszyklus ein und validieren sie?

Das Schulen von Personen, das Testen mithilfe von Automatisierung, ein Verständnis der Abhängigkeiten und die Validierung der Sicherheitseigenschaften von Tools und Anwendungen helfen dabei, die Wahrscheinlichkeit eines Sicherheitsproblems bei Produktions-Workloads zu verringern.

Bewährte Methoden

- SEC11-BP01 Für Anwendungssicherheit schulen
- SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren
- SEC11-BP03 Regelmäßig Penetrationstests durchführen
- SEC11-BP04 Manuelle Codeüberprüfungen
- SEC11-BP05 Services für Pakete und Abhängigkeiten zentralisieren
- SEC11-BP06 Software programmgesteuert bereitstellen
- SEC11-BP07 Die Sicherheitseigenschaften der Pipelines regelmäßig bewerten
- SEC11-BP08 Ein Programm entwickeln, das den Workload-Teams die Verantwortung für die Sicherheit überträgt

SEC11-BP01 Für Anwendungssicherheit schulen

Bieten Sie den Entwicklern in Ihrer Organisation Schulungsmöglichkeiten zu allgemeinen Praktiken für die sichere Entwicklung und den sicheren Betrieb von Anwendungen. Die Einführung sicherheitsbezogener Entwicklungsmethoden hilft, die Wahrscheinlichkeit von Problemen zu verringern, die nur während der Phase der Sicherheitsüberprüfung erkannt werden.

Gewünschtes Ergebnis: Beim Entwerfen und Entwickeln von Software sollte Sicherheit berücksichtigt werden. Wenn Entwickler in einer Organisation hinsichtlich sicherer Entwicklungspraktiken, die mit einem Bedrohungsmodell beginnen, geschult sind, wird die gesamte Qualität und Sicherheit der entwickelten Software verbessert. Mithilfe dieses Ansatzes kann die Zeit bis zum Ausliefern von Software oder Funktionen verringert werden, da der Überarbeitungsaufwand nach Sicherheitsüberprüfungen kleiner ist.

Für den Zweck dieser bewährten Methode bezieht sich sichere Entwicklung auf die Software, die geschrieben wird, und die Tools oder Systeme, die den Softwareentwicklungs-Lebenszyklus (SDLC) unterstützen.

Typische Anti-Muster:

- Auf eine Sicherheitsüberprüfung warten und dann die Sicherheitseigenschaften eines Systems berücksichtigen.
- Alle sicherheitsbezogenen Entscheidungen dem Sicherheitsteam überlassen.
- Nicht kommunizieren, wie sich die im Softwareentwicklungs-Lebenszyklus getroffenen Entscheidungen auf die allgemeinen Sicherheitserwartungen- oder -richtlinien der Organisation beziehen.
- Den Sicherheitsüberprüfungsprozess zu spät einsetzen.

Vorteile der Nutzung dieser bewährten Methode:

- Bessere Kenntnis der Unternehmensanforderungen hinsichtlich Sicherheit früh im Entwicklungszyklus.
- Raschere Lieferung von Funktionen durch das schnelle Identifizieren und Lösen potenzieller Sicherheitsproblemen.
- Verbesserte Qualität von Software und Systemen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Bieten Sie den Entwicklern in Ihrem Unternehmen Schulungen. Ein Kurs über Bedrohungsmodellierung ist ein guter Start, um einen Grundstein für Sicherheitsschulungen zu legen. Idealerweise sollten Entwickler selbständig auf die Informationen zugreifen können, die für ihre Workloads relevant sind. Dieser Zugriff hilft ihnen dabei, informierte Entscheidungen zu den Sicherheitseigenschaften der Systeme zu treffen, die sie entwickelt haben, ohne ein anderes Team kontaktieren zu müssen. Der Vorgang zum Einbinden von Sicherheitsteams in Überprüfungen sollte klar definiert und einfach zu befolgen sein. Die Schritte des Überprüfungsprozesses sollten Inhalt der Sicherheitsschulung sein. Dort, wo bekannte Implementierungsmuster oder -vorlagen verfügbar sind, sollten sie einfach zu finden und mit den allgemeinen Sicherheitsanforderungen verknüpft sein. Erwägen Sie, AWS CloudFormation, AWS Cloud Development Kit (AWS CDK)-Konstrukte, Service

Catalog oder andere Vorlagen-Tools zu verwenden, um den Bedarf nach einer benutzerspezifischen Konfiguration zu verringern.

Implementierungsschritte

- Ein Kurs über Bedrohungsmodellierung ist für Ihre Entwickler ein guter Start, um einen Grundstein für Sicherheitsüberlegungen zu legen.
- Bieten Sie Zugriff auf AWS Training and Certification und Branchen- oder AWS-Partner-Schulungen.
- Bieten Sie Schulungen zum Sicherheitsüberprüfungsprozess Ihres Unternehmens an, die die Aufteilung von Verantwortlichkeiten zwischen Sicherheitsteams, Workload-Teams und anderen Beteiligten klären.
- Veröffentlichen Sie Self-Service-Anweisungen zum Erfüllen von Sicherheitsanforderungen, einschließlich Codebeispielen und Vorlagen, wenn verfügbar.
- Erhalten Sie regelmäßig Feedback von Entwicklerteams zu ihrer Erfahrung mit dem Sicherheitsüberprüfungsprozess und -schulungen und verwenden Sie dieses Feedback, um Verbesserungen zu implementieren.
- Führen Sie Ernstfallübungen oder Kampagnen zum Beseitigen von Bugs durch, um die Anzahl von Fehlern zu verringern und die Fähigkeiten Ihrer Entwickler auszuweiten.

Ressourcen

Zugehörige bewährte Methoden:

- SEC11-BP08 Ein Programm entwickeln, das den Workload-Teams die Verantwortung für die Sicherheit überträgt

Zugehörige Dokumente:

- AWS Training und Zertifizierung
- How to think about cloud security governance (Über Cloud-Sicherheits-Governance nachdenken)
- How to approach threat modeling (Konzepte für Bedrohungsmodellierung)
- Accelerating training – The AWS Skills Guild (Schulungen beschleunigen – AWS Skills Guild)

Zugehörige Videos:

- Proactive security: Considerations and approaches (Proaktive Sicherheit: Überlegungen und Ansätze)

Zugehörige Beispiele:

- Workshop on threat modeling (Workshop zur Bedrohungsmodellierung)
- Industry awareness for developers (Branchenbewusstsein für Entwickler)

Zugehörige Services:

- AWS CloudFormation
- AWS Cloud Development Kit (AWS CDK) (AWS CDK) Konstrukte
- Service Catalog
- AWS BugBust

SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren

Automatisieren Sie das Testen der Sicherheitseigenschaften während des Entwicklungs- und Veröffentlichungslebenszyklus. Automatisierung vereinfacht die kontinuierliche und wiederholbare Identifizierung potenzieller Probleme. Dadurch wird das Risiko von Sicherheitsproblemen bei der bereitgestellten Software verringert.

Gewünschtes Resultat: Das Ziel von automatisiertem Testen ist, eine programmatische Möglichkeit zur frühen Erkennung von potenziellen Problemen – häufig im Laufe des Entwicklungslebenszyklus – zu bieten. Wenn Sie Regressionstests automatisieren, können Sie funktionale und nicht-funktionale Tests erneut durchführen, um zu überprüfen, ob zuvor getestete Software nach einer Änderung weiterhin wie erwartet funktioniert. Wenn Sie Sicherheitstests für Komponenten definieren, um nach häufigen Fehlkonfigurationen zu suchen, wie einer fehlerhaften oder fehlenden Authentifizierung, können Sie diese Fehler früh im Entwicklungsprozess identifizieren und beheben.

Testautomatisierung verwendet speziell entwickelte Testfälle zur Anwendungsvalidierung auf Basis der Anforderungen und der gewünschten Funktionalität der Anwendung. Das Ergebnis von automatisiertem Testen basiert auf dem Vergleich zwischen der erstellten Testausgabe und der erwarteten Ausgabe, wodurch der gesamte Lebenszyklus des Testens beschleunigt wird. Testmethoden wie Regressionstests und Komponententestsuites eignen sich am besten zur Automatisierung. Durch die Automatisierung des Testens von Sicherheitseigenschaften

können Entwickler automatisiertes Feedback erhalten, ohne auf eine Sicherheitsüberprüfung warten zu müssen. Automatisierte Tests in Form von statischer oder dynamischer Codeanalyse können die Qualität von Code erhöhen und dabei helfen, potenzielle Softwareprobleme früh im Entwicklungslebenszyklus zu erkennen.

Typische Anti-Muster:

- Testfälle und Testergebnisse des automatisierten Testens nicht kommunizieren.
- Automatisiertes Testen nur vor einer Veröffentlichung durchführen.
- Testfälle mit sich häufig ändernden Anforderungen automatisieren.
- Keine Anweisungen für den Umgang mit den Ergebnissen von Sicherheitstests bieten.

Vorteile der Nutzung dieser bewährten Methode:

- Verringerte Abhängigkeit von Menschen, um die Sicherheitseigenschaften eines Systems zu evaluieren.
- Beständige Resultate bei mehreren Arbeitsabläufen verbessern die Konsistenz.
- Verringerte Wahrscheinlichkeit, dass Sicherheitsprobleme in die Softwareproduktion eingeschleppt werden.
- Kürzeres Zeitfenster zwischen der Erkennung und Lösung von Softwareproblemen, da sie früher entdeckt werden.
- Erhöhte Sichtbarkeit von systemischem oder wiederholtem Verhalten bei mehreren Arbeitsabläufen, dank derer unternehmensweite Verbesserungen vorangetrieben werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Setzen Sie während der Entwicklung Ihrer Software unterschiedliche Mechanismen für das Testen von Software ein, um sicherzustellen, dass Sie Ihre Anwendung sowohl auf funktionale Anforderungen – basierend auf Ihrer Geschäftslogik – als auch auf nicht-funktionale Anforderungen testen, die sich auf die Zuverlässigkeit, Leistung und Sicherheit der Anwendung konzentrieren.

Statisches Anwendungssicherheitstesten (SAST) untersucht Ihren Quellcode auf Anomalien bei Sicherheitsmustern und bietet Hinweise auf einen fehleranfälligen Code. SAST nutzt

statische Eingaben, wie Dokumentation (Anforderungsspezifikationen, Designdokumentation und Designspezifikationen) und den Anwendungscode, um Tests in Bezug auf eine Reihe von bekannten Sicherheitsproblemen durchzuführen. Statische Code-Analyzer helfen dabei, die Analyse von großen Codemengen zu beschleunigen. Die NIST Quality Group bietet einen Vergleich von Source Code Security Analyzers, die Open-Source-Tools für Byte Code Scanner und Binary Code Scanner enthalten.

Ergänzen Sie Ihr statisches Testen mit Methodologien zum dynamischen Anwendungssicherheitstesten (DAST), wobei die Anwendung bei ihrer Ausführung getestet wird, um potenzielles unerwartetes Verhalten zu identifizieren. Dynamisches Testen kann verwendet werden, um potenzielle Probleme zu erkennen, die über die statische Analyse nicht gefunden werden können. Das Testen der Code-Repository-, Build- und Pipeline-Stadien ermöglicht Ihnen, nach unterschiedlichen Arten potenzieller Fehler in Ihrem Code zu suchen. Amazon CodeWhisperer bietet Codeempfehlungen, einschließlich Sicherheitsscans in der IDE des Entwicklers. Amazon CodeGuru Reviewer kann kritische Fehler, Sicherheitsprobleme und schwer zu findende Bugs während der Anwendungsentwicklung identifizieren und bietet Empfehlungen zur Verbesserung der Codequalität.

Der Workshop „Security for Developers“ verwendet AWS-Entwickler-Tools, wie AWS CodeBuild, AWS CodeCommit und AWS CodePipeline für die Automatisierung der Veröffentlichungs-Pipeline, die SAST- und DAST-Testmethodologien umfasst.

Richten Sie beim Durchlaufen Ihres Softwareentwicklungs-Lebenszyklus einen iterativen Prozess ein, der regelmäßige Anwendungsüberprüfungen mit Ihrem Sicherheitsteam enthält. Aus diesen Sicherheitsüberprüfungen gewonnenes Feedback sollte adressiert und im Rahmen der Bereitschaftsüberprüfung Ihrer Softwareversion validiert werden. Diese Überprüfungen schaffen einen robusten Sicherheitsstatus der Anwendungen und bieten Entwicklern umsetzbares Feedback, um Maßnahmen zum Beheben von Problemen zu ergreifen.

Implementierungsschritte

- Implementieren Sie eine integrierte Entwicklungsumgebung, Codeüberprüfung und CI/CD-Tools, die Sicherheitstests enthalten.
- Überlegen Sie, wo im Softwareentwicklungs-Lebenszyklus Pipelines blockiert werden können, anstatt Entwickler darüber zu informieren, dass Probleme behoben werden müssen.
- Der Workshop „Security for Developers“ bietet ein Beispiel für das Integrieren von statischem und dynamischem Testen in eine Veröffentlichungs-Pipeline.
- Das Durchführen von Tests oder Codeanalyse mithilfe von automatisierten Tools, wie Amazon CodeWhisperer, das mit IDEs von Entwicklern integriert ist, und Amazon CodeGuru Reviewer

für das Scannen von Code beim Commit, ermöglicht Entwicklern, Feedback zur richtigen Zeit zu erhalten.

- Beim Entwickeln mithilfe von AWS Lambda können Sie Amazon Inspector verwenden, um den Anwendungscode in Ihren Funktionen zu scannen.
- Der AWS CI/CD-Workshop bietet einen Ausgangspunkt für das Entwickeln von CI/CD-Pipelines auf AWS.
- Wenn automatisiertes Testen bei CI/CD-Pipelines enthalten ist, sollten Sie ein Ticketing-System verwenden, um das Melden und Lösen von Softwareproblemen nachzuverfolgen.
- Bei Sicherheitstests, die möglicherweise Erkenntnisse liefern, sollten Sie Lösungsanweisungen bieten, damit Entwickler die Codequalität verbessern können.
- Analysieren Sie von automatisierten Tools gewonnenen Einblicke, um die nächste Automatisierung, Entwicklerschulung oder Bewusstmachungskampagne zu planen.

Ressourcen

Zugehörige Dokumente:

- Continuous Delivery und Continuous Deployment

- AWS DevOps Competency Partners (AWS-Dev-Ops-Kompetenzpartner)
- AWS Security Competency Partners for Application Security (Sicherheitskompetenzpartner für Anwendungssicherheit)
- Choosing a Well-Architected CI/CD approach (Auswählen eines Well-Architected-CI/CD-Ansatzes)
- Monitoring CodeCommit events in Amazon EventBridge and Amazon CloudWatch Events (Überwachen von AWS-CodeCommit-Ereignissen in Amazon EventBridge und Amazon CloudWatch Events)
- Secrets detection in Amazon CodeGuru Review (Secrets-Erkennung bei der Code-Überprüfung in CodeGuru Reviewer)
- Accelerate deployments on AWS with effective governance (Beschleunigen von Bereitstellungen auf AWS mit effektiver Governance)
- How AWS approaches automating safe, hands-off deployments (Wie AWS die Automatisierung sicherer, vollautomatischer Bereitstellungen durchführt)

Zugehörige Videos:

- Hands-off: Automating continuous delivery pipelines at Amazon (Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon)
- Automating cross-account CI/CD pipelines (Automatisieren von kontoübergreifenden CI/CD-Pipelines)

Zugehörige Beispiele:

- Industry awareness for developers (Branchenbewusstsein für Entwickler)
- AWS CodePipeline Governance (GitHub)
- Workshop „Security for Developers“ (Workshop „Sicherheit für Entwickler“)
- AWS-CI/CD-Workshop

SEC11-BP03 Regelmäßig Penetrationstests durchführen

Führen Sie regelmäßige Penetrationstests bei Ihrer Software durch. Dieser Mechanismus hilft bei der Identifizierung potenzieller Softwareprobleme, die bei automatisierten Tests oder einer manuellen Überprüfung des Codes nicht erkannt werden können. Er kann Ihnen außerdem dabei helfen, die Wirksamkeit Ihrer Erkennungskontrollen zu verstehen. Penetrationstests sollten feststellen, ob es möglich ist, die Software so zu beeinflussen, dass sie auf unerwartete Weise ausgeführt wird, beispielsweise das Freigeben von Daten, die geschützt sein sollten, oder die Gewährung umfassenderer Berechtigungen als erwartet.

Gewünschtes Ergebnis: Penetrationstests werden verwendet, um die Sicherheitseigenschaften Ihrer Anwendung zu erkennen, zu lösen und zu validieren. Regelmäßige und geplante Penetrationstests sollten als Teil des Softwareentwicklungs-Lebenszyklus durchgeführt werden. Die aus Penetrationstests gewonnenen Erkenntnisse sollten vor der Veröffentlichung der Software adressiert werden. Sie sollten die Ergebnisse von Penetrationstests verwenden, um festzustellen, ob es sich um Probleme handelt, die mithilfe von Automatisierung gefunden werden könnten. Ein regelmäßiger und wiederholbarer Prozess für Penetrationstests, der einen aktiven Feedback-Mechanismus umfasst, fließt in die Anweisungen für Entwickler ein und verbessert die Softwarequalität.

Typische Anti-Muster:

- Penetrationstests nur für bekannte oder weit verbreitete Sicherheitsprobleme verwenden.
- Penetrationstests bei Anwendungen ohne abhängige Drittanbieter-Tools und -Bibliotheken durchführen.

- Penetrationstests nur bei Paketsicherheitsproblemen durchführen und die implementierte Geschäftslogik nicht evaluieren.

Vorteile der Nutzung dieser bewährten Methode:

- Gesteigertes Vertrauen in die Sicherheitseigenschaften der Software vor der Veröffentlichung.
- Die Möglichkeit, bevorzugte Anwendungsmuster zu identifizieren, wodurch die Softwarequalität erhöht wird.
- Verbresserte Sicherheitseigenschaften von Software durch eine Feedbackschleife, die früher im Entwicklungszyklus bestimmt, wo Automatisierung oder zusätzliche Schulungen erforderlich sind.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Penetrationstests sind eine strukturierte Sicherheitstestübung, wobei Sie Szenarios mit geplanten Sicherheitsverstößen durchführen, um Sicherheitskontrollen zu erkennen, zu lösen und zu validieren. Penetrationstests starten mit einer Erkundung, bei der Daten basierend auf dem aktuellen Design der Anwendung und ihrer Abhängigkeiten erfasst werden. Eine kuratierte Liste an sicherheitsspezifischen Testszenarios wird entwickelt und ausgeführt. Der wesentliche Zweck dieser Tests ist, die Sicherheitsprobleme in Ihrer Anwendung aufzudecken, die dazu genutzt werden könnten, unbeabsichtigten Zugriff auf Ihre Umgebung oder unautorisierten Zugriff auf Daten zu erhalten. Sie sollten Penetrationstests durchführen, wenn Sie neue Funktionen einführen oder wenn bei Ihrer Anwendung wesentliche Änderungen hinsichtlich der Funktion oder technischen Implementierung erfolgt sind.

Sie sollten in Ihrem Entwicklungslebenszyklus die am besten geeignete Phase bestimmen, um Penetrationstests durchzuführen. Das Testen sollte so spät stattfinden, dass sich das System nahe am vorgesehenen Veröffentlichungszustand befindet, aber es sollte ausreichend Zeit vorhanden sein, damit Probleme behoben werden können.

Implementierungsschritte

- Implementieren Sie einen strukturierten Prozess für den Umfang der Penetrationstests und dieser Prozess sollte auf einem Bedrohungsmodell basieren, um den Kontext zu bewahren.
- Bestimmen Sie den geeigneten Zeitpunkt im Entwicklungszyklus zum Durchführen von Penetrationstests. Penetrationstests sollten dann erfolgen, wenn die geringsten Änderungen an der Anwendung erwartet werden, aber noch ausreichend Zeit für die Fehlerbehebung übrig ist.

- Schulen Sie Ihre Entwickler in Bezug darauf, was sie von den Ergebnissen von Penetrationstests erwarten und wie Informationen zur Mängelbeseitigung erhalten können.
- Verwenden Sie Tools zum Beschleunigen des Penetrationstestvorgangs, indem Sie gängige oder wiederholbare Tests automatisieren.
- Analysieren Sie Ergebnisse von Penetrationstests, um systemische Sicherheitsprobleme zu identifizieren, und verwenden Sie diese Daten, um sie in zusätzliche automatisierte Tests und fortlaufende Entwicklerschulungen einfließen zu lassen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC11-BP01 Für Anwendungssicherheit schulen
- SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren

Zugehörige Dokumente:

- AWS-Penetrationstest bieten ausführliche Anweisungen für Penetrationstests mit AWS
- Accelerate deployments on AWS with effective governance (Beschleunigen von Bereitstellungen auf AWS mit effektiver Governance)
- AWS Security Competency Partners (AWS-Kompetenzpartner für Sicherheit)
- Modernize your penetration testing architecture on AWS Fargate (Modernisieren Ihrer Penetrationstestarchitektur auf AWS Fargate)
- AWS Fault Injection Simulator

Zugehörige Beispiele:

- Automate API testing with AWS CodePipeline (Automatisieren von API-Testen mit AWS Codepipeline mit Postman) (GitHub)
- Automated security helper (Automatisierter Sicherheitshelfer) (GitHub)

SEC11-BP04 Manuelle Codeüberprüfungen

Führen Sie eine manuelle Codeüberprüfung der von Ihnen produzierten Software durch. Dieser Prozess hilft zu verifizieren, dass die Person, die den Code geschrieben hat, die Qualität des Codes nicht allein überprüft.

Gewünschtes Ergebnis: Das Hinzufügen einer manuellen Codeüberprüfung während der Entwicklung erhöht die Qualität der geschriebenen Software, hilft dabei, weniger erfahrene Teammitglieder weiterzubilden, und bietet eine Möglichkeit, Stellen zum Einsetzen von Automatisierung zu identifizieren. Manuelle Codeüberprüfungen können von automatisierten Tools und Tests unterstützt werden.

Typische Anti-Muster:

- Keine Codeüberprüfungen vor der Bereitstellung durchführen.
- Die gleiche Person zum Schreiben und Überprüfen des Codes einsetzen.
- Keine Automatisierung zum Unterstützen und Orchestrieren von Codeüberprüfungen einsetzen.
- Entwickler nicht hinsichtlich Anwendungssicherheit schulen, bevor sie Code überprüfen.

Vorteile der Nutzung dieser bewährten Methode:

- Verbesserte Codequalität.
- Erhöhte Konsistenz bei der Codeentwicklung durch das erneute Verwenden von gängigen Ansätzen.
- Verringerte Anzahl von Schwierigkeiten, die bei Penetrationstests und in späteren Phasen entdeckt werden.
- Verbesserter Wissenstransfer innerhalb des Teams.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Der Überprüfungsschritt sollte als Teil des allgemeinen Codeverwaltungs-Flows implementiert werden. Die Details hängen vom Ansatz an, der für Verzweigen, Pull-Anforderungen und Zusammenführen verwendet wird. Sie verwenden möglicherweise AWS CodeCommit oder Drittanbieterlösungen wie GitHub, GitLab oder Bitbucket. Welche Methode auch immer Sie verwenden – es ist wichtig, dass Sie verifizieren, dass Ihre Prozesse eine Überprüfung von Code

erfordern, bevor dieser in einer Produktionsumgebung bereitgestellt wird. Das Verwenden von Tools wie Amazon CodeGuru Reviewer kann das Orchestrieren des Codeüberprüfungsvorgangs vereinfachen.

Implementierungsschritte

- Implementieren Sie einen Schritt zur manuellen Überprüfung als Teil Ihres Codeverwaltungs-Flows und führen Sie diese Überprüfung durch, bevor Sie fortfahren.
- Erwägen Sie Amazon CodeGuru Reviewer für das Verwalten und Unterstützen bei Codeüberprüfungen.
- Implementieren Sie einen Genehmigungs-Workflow, bei dem eine Codeüberprüfung erforderlich ist, bevor Code zur nächsten Stufe übergehen kann.
- Verifizieren Sie, dass es einen Vorgang gibt, um Probleme bei manuellen Codeüberprüfungen zu finden, die automatisch erkannt werden könnten.
- Integrieren Sie den Schritt zur manuellen Codeüberprüfung auf eine Weise, die mit Ihren Codeentwicklungspraktiken übereinstimmt.

Ressourcen

Zugehörige bewährte Methoden:

- SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren

Zugehörige Dokumente:

- Working with pull requests in AWS CodeCommit repositories (Arbeiten Mit Pull-Anforderungen in AWS CodeCommit)
- Working with approval rule templates in AWS CodeCommit (Arbeiten mit Genehmigungsregelvorlagen in AWS CodeCommit)
- About pull requests in GitHub (Informationen über Pull-Anforderungen auf GitHub)
- Automate code reviews with Amazon CodeGuru Reviewer (Automatisieren von Codeüberprüfungen mit Amazon CodeGuru Reviewer)
- Automating detection of security vulnerabilities and bugs in CI/CD pipelines using Amazon CodeGuru Reviewer CLI (Automatisieren der Erkennung von Sicherheitsschwachstellen und Bugs in CI/CD-Pipelines mithilfe der CLI von Amazon CodeGuru Reviewer

Zugehörige Videos:

- Continuous improvement of code quality with Amazon CodeGuru (Kontinuierliche Verbesserung der Codequalität mit Amazon CodeGuru)

Zugehörige Beispiele:

- Security for Developers workshop (Workshop „Sicherheit für Entwickler“)

SEC11-BP05 Services für Pakete und Abhängigkeiten zentralisieren

Stellen Sie zentralisierte Services für Entwicklungsteams bereit, sodass sie Softwarepakete und andere Abhängigkeiten erhalten können. Dadurch können Pakete validiert werden, bevor sie in die von Ihnen geschriebene Software integriert werden, und es kann eine Datenquelle für die Analyse der Software bereitgestellt werden, die in Ihrer Organisation verwendet wird.

Gewünschtes Ergebnis: Software besteht aus einem Set aus anderen Softwarepaketen zusätzlich zum Code, der geschrieben wird. Dadurch wird die Implementierung von häufig verwendeten Funktionen vereinfacht, wie einem JSON-Parser oder einer Verschlüsselungsbibliothek. Das logische Zentralisieren der Quellen und Abhängigkeiten für diese Pakete bietet einen Mechanismus für Sicherheitsteams, damit diese die Eigenschaften der Pakete validieren können, bevor sie verwendet werden. Dieser Ansatz verringert auch das Risiko, dass ein unerwartetes Problem durch die Änderung eines vorhandenen Pakets verursacht wird oder dass Entwicklungsteams beliebige Pakete direkt aus dem Internet einbeziehen. Verwenden Sie diesen Ansatz zusammen mit manuellem und automatischem Testen, um das Vertrauen in die Qualität der entwickelten Software zu steigern.

Typische Anti-Muster:

- Pakete aus beliebigen Repositorys im Internet abrufen.
- Neue Pakete nicht testen, bevor sie für Entwickler verfügbar gemacht werden.

Vorteile der Nutzung dieser bewährten Methode:

- Besseres Verständnis darüber, welche Pakete in der entwickelten Software verwendet werden.
- Benachrichtigung von Workload-Teams, wenn ein Paket aktualisiert werden muss – basierend auf dem Verständnis davon, wer was verwendet.
- Geringeres Risiko, dass ein Paket mit Problemen in Ihrer Software enthalten ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Stellen Sie zentralisierte Services für Pakete und Abhängigkeiten so bereit, dass sie von Entwicklern einfach verwendet werden können. Zentralisierte Services können logisch zentral sein, anstatt als monolithisches System implementiert zu werden. Mit diesem Ansatz können Sie Services anbieten, die die Anforderungen Ihrer Entwickler erfüllen. Sie sollten eine effiziente Möglichkeit zum Hinzufügen von Paketen zum Repository implementieren, wenn Updates erfolgen oder neue Anforderungen aufkommen. Mithilfe von AWS-Services wie AWS CodeArtifact oder ähnlichen AWS-Partnerlösungen kann diese Funktion geboten werden.

Implementierungsschritte:

- Implementieren Sie einen logisch zentralisierten Repository-Service, der in allen Umgebungen, in welchen die Software entwickelt wird, verfügbar ist.
- Fügen Sie den Zugriff auf das Repository als Teil des AWS-Konto-Vergabeprozesses hinzu.
- Entwickeln Sie eine Automatisierung zum Testen von Paketen, bevor diese in einem Repository veröffentlicht werden.
- Pflegen Sie Metriken der am häufigsten verwendeten Pakete, Sprachen und Teams mit den häufigsten Änderungen.
- Stellen Sie Entwicklungsteams einen automatisierten Mechanismus bereit, damit sie neue Pakete anfordern und Feedback abgeben können.
- Scannen Sie regelmäßig Pakete in Ihrem Repository, um die Auswirkungen von kürzlich entdecken Problemen zu identifizieren.

Ressourcen

Zugehörige bewährte Methoden:

- SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren

Zugehörige Dokumente:

- Accelerate deployments on AWS with effective governance (Beschleunigen von Bereitstellungen auf AWS mit effektiver Governance)

- Tighten your package security with CodeArtifact Package Origin Control toolkit (Erhöhen Ihrer Paketsicherheit mit dem Toolkit von CodeArtifact Package Origin Control)
- Detecting security issues in logging with Amazon CodeGuru Reviewer (Erkennen von Sicherheitsproblemen beim Protokollieren mit Amazon CodeGuru Reviewer)
- Supply chain Levels for Software Artifacts (SLSA) (Lieferkettenebenen für Software-Artefakte)

Zugehörige Videos:

- Proactive security: Considerations and approaches (Proaktive Sicherheit: Überlegungen und Ansätze)
- The AWS Philosophy of Security (re:Invent 2017) (Die AWS-Philosophie zu Sicherheit)
- When security, safety, and urgency all matter: Handling Log4Shell (Wenn Sicherheit und Dringlichkeit von Bedeutung sind: Umgang mit Log4Shell)

Zugehörige Beispiele:

- Multi Region Package Publishing Pipeline (Mehrregions-Veröffentlichungs-Pipeline für Pakete) (GitHub)
- Publishing Node.js Modules on AWS CodeArtifact using AWS CodePipeline (Node.js-Module auf AWS CodeArtifact mithilfe von AWS CodePipeline veröffentlichen) (GitHub)
- AWS CDK Java CodeArtifact Pipeline Sample (Beispiel für eine Java-CodeArtifact-Pipeline) (GitHub)
- Distribute private .NET NuGet packages with AWS CodeArtifact (Verteilen von privaten .NET-NuGet-Pakete mit AWS CodeArtifact) (GitHub)

SEC11-BP06 Software programmgesteuert bereitstellen

Führen Sie Bereitstellungen von Software möglichst programmgesteuert durch. Dieser Ansatz verringert die Wahrscheinlichkeit eines Bereitstellungsfehlers oder der Einführung eines unerwarteten Problem aufgrund eines menschlichen Fehlers.

Gewünschtes Ergebnis: Menschen von Daten fernhalten ist eines der Prinzipien für sicheres Entwickeln in der AWS Cloud. Dieses Prinzip umfasst, wie Sie Ihre Software bereitstellen.

Wenn Sie sich nicht auf Menschen verlassen müssen, um Software bereitzustellen, bietet dies den Vorteil, dass Sie mehr Vertrauen darin haben können, dass das, was getestet wird, auch das ist, was

bereitgestellt wird, und dass die Bereitstellung jedes Mal konsistent durchgeführt wird. Die Software sollte nicht geändert werden müssen, um in unterschiedlichen Umgebungen zu funktionieren. Mithilfe der Prinzipien der 12-Faktor-Anwendungsentwicklung, insbesondere dem Externalisieren der Konfiguration, können Sie denselben Code ohne Änderungen in mehreren Umgebungen bereitstellen. Das kryptografische Signieren von Softwarepaketen ist eine gute Möglichkeit, zu verifizieren, dass sich zwischen den Umgebungen nichts geändert hat. Das Gesamtergebnis dieses Ansatzes ist die Risikoverringerung bei Ihrem Änderungsprozess und die Verbesserung der Konsistenz von Softwareveröffentlichungen.

Typische Anti-Muster:

- Software manuell in die Produktion bereitstellen.
- Manuelle Änderungen an Software durchführen, um unterschiedliche Umgebungen zu bedienen.

Vorteile der Nutzung dieser bewährten Methode:

- Gesteigertes Vertrauen in den Prozess der Softwareveröffentlichung.
- Verringertes Risiko, dass eine fehlgeschlagene Änderung, die Geschäftsfunktionen beeinträchtigt.
- Erhöhte Veröffentlichungsfrequenz, aufgrund eines geringeren Änderungsrisikos.
- Automatische Rollback-Funktion für unerwartete Ereignisse während der Bereitstellung.
- Die Möglichkeit, kryptografisch zu beweisen, dass es sich bei der getesteten Software um die bereitgestellte Software handelt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Entwickeln Sie Ihre AWS-Konto-Struktur, um den fortlaufenden menschlichen Zugriff über Umgebungen zu verhindern und CI/CD-Tools zum Durchführen von Bereitstellungen zu verwenden. Entwerfen Sie Ihre Anwendungen so, dass umgebungsspezifische Konfigurationsdaten von externen Quellen gewonnen werden, wie AWS Systems Manager Parameter Store. Signieren Sie Pakete, nachdem sie getestet wurden, und validieren Sie diese Signaturen während der Bereitstellung. Konfigurieren Sie Ihre CI/CD-Pipelines, um den Anwendungscode zu übertragen und verwenden Sie Canaries, um die erfolgreiche Bereitstellung zu bestätigen. Verwenden Sie Tools wie AWS CloudFormation oder AWS CDK, um Ihre Infrastruktur zu definieren, und verwenden Sie dann AWS CodeBuild und AWS CodePipeline, um CI/CD-Vorgänge durchzuführen.

Implementierungsschritte

- Entwicklen Sie gut definierte CI/CD-Pipelines, um den Bereitstellungsprozess zu optimieren.
- Die Verwendung von AWS CodeBuild und AWS Code Pipeline, um die CI/CD-Funktionalität zu bieten, vereinfacht das Integrieren von Sicherheitstesten in Ihre Pipelines.
- Befolgen Sie die Anweisungen für die Trennung von Umgebungen im Whitepaper Organisation Ihrer AWS-Umgebung mit mehreren Konten.
- Verifzieren Sie, dass es keinen fortlaufenden Zugriff durch Personen auf Umgebungen gibt, in welchen Produktions-Workloads ausgeführt werden.
- Entwickeln Sie Ihre Anwendungen so, dass sie die Externalisierung von Konfigurationsdaten unterstützen.
- Ziehen Sie eine Bereitstellung mithilfe eines Blau/Grün-Modells in Betracht.
- Setzen Sie Canaries ein, um die erfolgreiche Bereitstellung der Software zu validieren.
- Verwenden Sie kryptografische Tools wie AWS Signer oder AWS Key Management Service (AWS KMS), um die Softwarepakete, die Sie bereitstellen, zu signieren und zu verifizieren.

Ressourcen

Zugehörige bewährte Methoden:

- SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren

Zugehörige Dokumente:

- AWS-CI/CD-Workshop
- Accelerate deployments on AWS with effective governance (Beschleunigen von Bereitstellungen auf AWS mit effektiver Governance)
- Automating safe, hands-off deployments (Automatisierung sicherer, vollautomatischer Bereitstellungen)
- Code signing using AWS Certificate Manager Private CA and AWS Key Management Service asymmetric keys (Codesignatur mithilfe von AWS Certificate Manager Private CA und asymmetrischen Schlüsseln von AWS Key Management Service)
- Code Signing, a Trust and Integrity Control for AWS Lambda (Codesignatur, eine Vertrauens- und Integritätskontrolle für AWS Lambda)

Zugehörige Videos:

- Hands-off: Automating continuous delivery pipelines at Amazon (Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon)

Zugehörige Beispiele:

- Blue/Green deployments with AWS Fargate (Blau/Grün-Bereitstellungen mit AWS Fargate)

SEC11-BP07 Die Sicherheitseigenschaften der Pipelines regelmäßig bewerten

Wenden Sie die Prinzipien der Säule der Well-Architected-Sicherheit bei Ihren Pipelines an und achten Sie dabei besonders auf die Trennung von Berechtigungen. Bewerten Sie die Sicherheitseigenschaften Ihrer Pipeline-Infrastruktur regelmäßig. Durch die effektive Verwaltung der Pipeline-Sicherheit können Sie bei der Software, die diese Pipelines durchläuft, für Sicherheit sorgen.

Gewünschtes Ergebnis: Die Pipelines, die zum Entwickeln und Bereitstellen Ihrer Software verwendet werden, sollten dieselben empfohlenen Praktiken wie jeder andere Workload in Ihrer Umgebung befolgen. Die Tests, die in den Pipelines implementiert sind, sollten nicht von Entwicklern bearbeitet werden können, die sie verwenden. Die Pipelines sollten nur Berechtigungen für die Bereitstellungen haben, die sie durchführen, und sollten Sicherheitsmaßnahmen zum Verhindern von Bereitstellungen in den falschen Umgebungen implementieren. Pipelines sollten sich nicht auf langfristige Anmeldeinformationen verlassen und sollten konfiguriert sein, um den Status auszugeben, sodass die Integrität der Entwicklungsumgebung validiert werden kann.

Typische Anti-Muster:

- Sicherheitstests können von Entwicklern umgangen werden.
- Berechtigungen für Bereitstellungs-Pipelines sind übermäßig breit gefasst.
- Pipelines sind nicht konfiguriert, um Eingaben zu validieren.
- Berechtigungen in Zusammenhang mit Ihrer CI/CD-Infrastruktur werden nicht regelmäßig überprüft.
- Langfristige oder fest codierte Anmeldeinformationen werden verwendet.

Vorteile der Nutzung dieser bewährten Methode:

- Größeres Vertrauen in die Integrität der Software, die über die Pipelines entwickelt und bereitgestellt wird.

- Eine Bereitstellung kann angehalten werden, wenn es verdächtige Aktivitäten gibt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Durch den Beginn mit CI/CD-Services, die IAM-Rollen unterstützen, wird das Risiko von Anmeldeinformationslecks verringert. Durch das Anwenden der Prinzipien der Säule „Sicherheit“ auf Ihre CI/CD-Pipeline-Infrastruktur können Sie bestimmen, wo Sicherheitsverbesserungen durchgeführt werden können. Das Befolgen der AWS Deployment Pipelines Reference Architecture (Referenzarchitektur für AWS-Bereitstellungs-Pipelines) ist ein guter Startpunkt für das Erstellen Ihrer eigenen CI/CD-Umgebungen. Regelmäßige Überprüfungen der Pipeline-Implementierung und Untersuchungen von Protokollen auf unerwartetes Verhalten können Ihnen dabei helfen, die Verwendungsmuster der Pipelines, die zum Bereitstellen der Software verwendet werden, besser zu verstehen.

Implementierungsschritte

- Beginnen Sie mit der AWS Deployment Pipelines Reference Architecture (Referenzarchitektur für AWS-Bereitstellungs-Pipelines).
- Erwägen Sie, AWS IAM Access Analyzer zu verwenden, um für die Pipelines programmatisch IAM-Richtlinien mit der geringsten Berechtigung zu erstellen.
- Integrieren Sie Ihre Pipelines mit Überwachung und Benachrichtigung, sodass Sie über unerwartete oder abnorme Aktivitäten benachrichtigt werden. Bei von AWS verwalteten Services können Sie mithilfe von Amazon EventBridge Daten zu Zielen wie AWS Lambda oder Amazon Simple Notification Service (Amazon SNS) umleiten.

Ressourcen

Zugehörige Dokumente:

- AWS Deployment Pipelines Reference Architecture (Referenzarchitektur für AWS-Bereitstellungs-Pipelines)
- Monitoring AWS CodePipeline (Überwachen von AWS CodePipeline)
- Security best practices for AWS CodePipeline (Bewährte Methoden für die Sicherheit mit AWS CodePipeline)

Zugehörige Beispiele:

- DevOps monitoring dashboard (DevOps-Überwachungs-Dashboard) (GitHub)

SEC11-BP08 Ein Programm entwickeln, das den Workload-Teams die Verantwortung für die Sicherheit überträgt

Entwickeln Sie ein Programm oder einen Mechanismus, der es Entwicklerteams ermöglicht, Entscheidungen bezüglich der Sicherheit der von ihnen erstellten Software zu treffen. Zwar muss Ihr Sicherheitsteam diese Entscheidungen immer noch während einer Überprüfung validieren, doch macht das Übertragen der Sicherheitsverantwortlichkeit auf Entwicklerteams eine schnellere und sicherer Workload-Erstellung möglich. Zudem fördert dieser Mechanismus eine Kultur der Verantwortlichkeit, die einen positiven Einfluss auf den Betrieb der von Ihnen entwickelten Systeme hat.

Gewünschtes Ergebnis: Um Entwicklungsteams Verantwortung und Entscheidungsfindung zu überlassen, können Sie entweder Entwickler in Bezug darauf schulen, wie sie über Sicherheit nachdenken, oder Sie können ihre Schulung mithilfe von Sicherheitsexperten verbessern, die Teil des Entwicklungsteams sind oder damit in Kontakt stehen. Beide Ansätze sind valide und ermöglichen dem Team, bessere Sicherheitsentscheidungen früher im Entwicklungszyklus zu treffen. Dieses Verantwortungsmodell basiert auf Schulungen in Anwendungssicherheit. Wenn Sie mit einem Bedrohungsmodell für den bestimmten Workload beginnen, hilft Ihnen dies dabei, das Design Thinking auf den entsprechenden Kontext zu konzentrieren. Ein weiterer Vorteil, eine Community an sicherheitsorientierten Entwicklern oder eine Gruppe an Sicherheitstechnikern zu haben, die mit Entwicklungsteams zusammenarbeiten, ist, dass Sie ein besseres Verständnis darüber erlangen, wie Code geschrieben wird. Dieses Verständnis hilft Ihnen dabei, die nächsten verbesserungswürdigen Bereiche bei Ihrem Automatisierungsunterfangen zu bestimmen.

Typische Anti-Muster:

- Einem Sicherheitsteam alle Entscheidungen bezüglich des Sicherheitsdesigns überlassen.
- Sicherheitsanforderungen nicht früh genug im Entwicklungsprozess adressieren.
- Kein Feedback bezüglich des Programmbetriebs von Entwicklern und Sicherheitsexperten einholen.

Vorteile der Nutzung dieser bewährten Methode:

- Kürzere Dauer zum Abschließen von Sicherheitsüberprüfungen.
- Verringerung von Sicherheitsproblemen, die nur auf der Ebene der Sicherheitsüberprüfung erkannt werden.
- Verbesserung der gesamten Qualität der Software, die geschrieben wird.
- Die Möglichkeit, systemische Probleme oder Bereiche mit hoher Wertverbesserung zu identifizieren und zu verstehen.
- Verringerung der erforderlichen Überarbeitung aufgrund von Erkenntnissen in Bezug auf Sicherheit.
- Verbesserung der Wahrnehmung von Sicherheitsfunktionen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Beginnen Sie mit den Anweisungen unter SEC11-BP01 Für Anwendungssicherheit schulen. Bestimmen Sie danach das Betriebsmodell für das Programm, von dem Sie denken, dass es am besten für Ihr Unternehmen funktioniert. Die zwei Hauptmuster bestehen daraus, Entwickler zu schulen oder Sicherheitsexperten in in Entwicklungsteams zu positionieren. Nachdem Sie sich für eine anfängliche Verfahrensweise entschieden haben, sollten Sie einen Pilotlauf mit einem einzelnen Team oder einer kleinen Gruppe von Workload-Teams durchführen, um zu bestätigen, dass das Modell für Ihr Unternehmen funktioniert. Unterstützung der Führungskräfte aus den Entwicklungs- und Sicherheitsbereichen des Unternehmens hilft Ihnen beim Durchführen und dem Erfolg des Programms. Während Sie dieses Programm entwickeln, ist es wichtig, Metriken auszuwählen, die auf den Wert des Programms hinweisen. Zu erfahren, wie AWS mit diesem Problem umgegangen ist, bietet eine gute Lernerfahrung. Die bewährte Methode konzentriert sich auf die Veränderung und Kultur des Unternehmens. Die von Ihnen eingesetzten Tools sollten die Zusammenarbeit zwischen den Entwicklung- und Sicherheits-Communities unterstützen.

Implementierungsschritte

- Beginnen Sie damit, Ihre Entwickler im Bereich der Anwendungssicherheit zu schulen.
- Schaffen Sie eine Community und ein Onboarding-Programm zum Schulen der Entwickler.
- Geben Sie dem Programm einen Namen. Guardians, Champions oder Advocates werden häufig verwendet.
- Bestimmen Sie das Modell, das verwendet werden soll: Schulen Sie Entwickler und bringen Sie Sicherheitstechniker oder andere verwandte Sicherheitsrollen ein.

- Identifizieren Sie Projektsponsoren aus Sicherheitsexperten, Entwicklern und anderen potenziell relevanten Gruppen.
- Verfolgen Sie Metriken für die Anzahl der im Programm involvierten Personen, die für Überprüfungen erforderliche Zeit und das Feedback von Entwicklern und Sicherheitsexperten. Nutzen Sie diese Metriken, um Verbesserungen vorzunehmen.

Ressourcen

Zugehörige bewährte Methoden:

- SEC11-BP01 Für Anwendungssicherheit schulen
- SEC11-BP02 Das Testen während des Entwicklungs- und Veröffentlichungslebenszyklus automatisieren

Zugehörige Dokumente:

- How to approach threat modeling (Konzepte für Bedrohungsmodellierung)
- How to think about cloud security governance (Über Cloud-Sicherheits-Governance nachdenken)

Zugehörige Videos:

- Proactive security: Considerations and approaches (Proaktive Sicherheit: Überlegungen und Ansätze)

Zuverlässigkeit

Die Säule „Zuverlässigkeit“ umfasst die Fähigkeit eines Workloads, die beabsichtigte Funktion erwartungsgemäß korrekt und konsistent auszuführen. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule „Zuverlässigkeit“.

Bereiche für bewährte Methoden

- Grundlagen
- Workload-Architektur
- Änderungsverwaltung
- Fehlerverwaltung

Grundlagen

Fragen

- REL 1. Wie verwalten Sie Servicekontingente und Einschränkungen?
- REL 2. Was ist bei der Planung der Netzwerktopologie zu beachten?

REL 1. Wie verwalten Sie Servicekontingente und Einschränkungen?

Für cloudbasierte Workload-Architekturen gibt es Servicekontingente (die auch als Servicelimits bezeichnet werden). Diese Kontingente dienen dazu, nicht versehentlich mehr Ressourcen bereitzustellen als nötig und Anfrageraten für API-Vorgänge zu begrenzen, um Services vor Missbrauch zu schützen. Darüber hinaus gibt es Ressourceneinschränkungen, z. B. die Rate, mit der Bits durch ein Glasfaserkabel geschleust werden können, oder die Speichermenge auf einer physischen Festplatte.

Bewährte Methoden

- REL01-BP01 Kenntnis von Servicekontingenten und Einschränkungen
- REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen
- REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur
- REL01-BP04 Überwachen und Verwalten von Kontingenten
- REL01-BP05 Automatisieren der Kontingentverwaltung
- REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist

REL01-BP01 Kenntnis von Servicekontingenten und Einschränkungen

Sie wissen über die Standardkontingente Bescheid und verwalten Anfragen zur Kontingenterhöhung für Ihre Workload-Architektur. Außerdem wissen Sie, welche Ressourceneinschränkungen, z. B. bezüglich Datenträgern oder Netzwerken, potenziell große Auswirkungen haben.

Gewünschtes Ergebnis: Kunden können eine Beeinträchtigung oder Unterbrechung ihrer Services in ihrer AWS-Konten verhindern, indem sie geeignete Richtlinien für die Überwachung von Schlüsselkennzahlen, Infrastrukturüberprüfungen und Automatisierungsschritte zur Behebung von Problemen einführen, um sicherzustellen, dass Service Quotas und Einschränkungen, die eine Beeinträchtigung oder Unterbrechung der Dienste verursachen könnten, nicht erreicht werden.

Typische Anti-Muster:

- Bereitstellung eines Workloads ohne Kenntnis der harten oder weichen Quoten und ihrer Grenzen für die verwendeten Services.
- Bereitstellung eines Ersatz-Workloads, ohne die erforderlichen Quoten zu analysieren und neu zu konfigurieren oder den Support im Voraus zu kontaktieren.
- Annehmen, dass Cloud-Services keine Grenzen haben und die Service ohne Berücksichtigung von Tarifen, Grenzen, Zählungen und Mengen genutzt werden können.
- Annehmen, dass die Quoten automatisch erhöht werden.
- Keine Kenntnis des Prozesses und der Zeitleiste von Quotenanforderungen.
- Annehmen, dass das Standardkontingent für Cloud-Services für jeden Service im regionalen Vergleich identisch ist.
- Annehmen, dass die Servicebeschränkungen überschritten werden können und die Systeme automatisch skalieren oder das Limit über die Beschränkungen der Ressource hinaus erhöhen.
- Die Anwendung nicht bei Spitzenbelastungen testen, um die Auslastung der Ressourcen zu strapazieren.
- Bereitstellung der Ressource ohne Analyse der erforderlichen Ressourcengröße.
- Überbereitstellung von Kapazitäten durch Auswahl von Ressourcentypen, die weit über den tatsächlichen Bedarf oder die erwarteten Spitzen hinausgehen.
- Keine Bewertung des Kapazitätsbedarfs für neue Datenverkehrsniveaus im Vorfeld eines neuen Kundenereignisses und keine Einführung einer neuen Technologie.

Vorteile der Nutzung dieser bewährten Methode: Durch die Überwachung und automatisierte Verwaltung von Service Quotas und Ressourcenbeschränkungen können Ausfälle proaktiv reduziert werden. Änderungen in den Datenverkehrsmustern für den Service eines Kunden können zu einer Unterbrechung oder Verschlechterung führen, wenn die bewährten Methoden nicht befolgt werden. Durch die Überwachung und Verwaltung dieser Werte in allen Regionen und auf allen Konten können die Anwendungen bei ungünstigen oder ungeplanten Ereignissen besser geschützt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Service Quotas ist ein AWS-Service, mit dem Sie Ihre Kontingente für über 250 AWS-Services von einem Standort aus verwalten können. Neben der Suche nach den Kontingentwerten können Sie auch Kontingenterhöhungen über die Service Quotas-Konsole oder über das AWS SDK anfordern

und nachverfolgen. AWS Trusted Advisor bietet eine Servicekontingent-Prüfung, die Ihre Nutzung und Ihre Kontingente für bestimmte Aspekte einiger Services anzeigt. Die Standardkontingente pro Service finden Sie ebenfalls in der AWS-Dokumentation für den jeweiligen Service. Weitere Informationen finden Sie unter Amazon-VPC-Kontingente).

Einige Servicelimits wie Ratenlimits für gedrosselte APIs werden innerhalb des Amazon API Gateway selbst festgelegt. Dazu wird ein Nutzungsplan konfiguriert. Andere Limits, die für ihre jeweiligen Services konfiguriert werden, sind bereitgestellte IOPS, zugewiesener Amazon RDS-Speicher und Amazon EBS-Volume-Zuweisungen. Amazon Elastic Compute Cloud verfügt über ein eigenes Service Limits-Dashboard, mit dem Sie Ihre Limits für Instances, Amazon Elastic Block Store und Elastic IP-Adressen verwalten können. Wenn Sie einen Anwendungsfall haben, bei dem sich Servicekontingente auf die Leistung Ihrer Anwendung auswirken und eine Anpassung an Ihre Anforderungen nicht möglich ist, wenden Sie sich an den Support, um zu ermitteln, ob es Lösungen gibt.

Service Quotas können spezifisch für eine Region oder auch global sein. Ein AWS-Service, der sein Kontingent erreicht hat, verhält sich bei normaler Nutzung nicht wie erwartet und es kann zu Unterbrechungen oder Beeinträchtigungen des Services kommen. Beispielsweise begrenzt ein Servicekontingent die Anzahl der DL Amazon EC2, die in einer Region genutzt werden können, und dieses Limit kann während eines Ereignisses zur Skalierung des Datenverkehrs durch Auto Scaling-Gruppen (ASG) erreicht werden.

Service Quotas für die einzelnen Konten sollten regelmäßig auf ihre Nutzung hin überprüft werden, um festzustellen, welche Servicelimits für das jeweilige Konto angemessen sind. Diese Service Quotas dienen als betrieblicher Integritätsschutz, um zu verhindern, dass versehentlich mehr Ressourcen bereitgestellt werden, als Sie benötigen. Sie begrenzen auch die Anfrageraten bei API-Operationen, um Services vor Missbrauch zu schützen.

Serviceeinschränkungen und Service Quotas unterscheiden sich voneinander. Serviceeinschränkungen stellen die Limits einer bestimmten Ressource dar, wie sie durch diesen Ressourcentyp definiert sind. Dabei kann es sich um die Speicherkapazität (z. B. hat gp2 eine Größenbegrenzung von 1 GB bis 16 TB) oder den Festplattendurchsatz (10.0000 iops) handeln. Es ist von entscheidender Bedeutung, dass die Beschränkung eines Ressourcentyps konstruiert und ständig auf eine Nutzung geprüft wird, durch die das Limit erreicht werden könnte. Wenn eine Beschränkung unerwartet erreicht wird, können die Anwendungen oder Services des Kontos beeinträchtigt oder unterbrochen werden.

Wenn es einen Anwendungsfall gibt, bei dem sich Service Quotas auf die Leistung Ihrer Anwendung auswirken und eine Anpassung an die Anforderungen nicht möglich ist, wenden Sie sich an

den Support, um zu ermitteln, ob es Lösungen gibt. Weitere Einzelheiten zur Anpassung fester Kontingente finden Sie unter REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur.

Es gibt eine Reihe von AWS-Services und -Tools, die Sie bei der Überwachung und Verwaltung von Service Quotas unterstützen. Der Service und die Tools sollten genutzt werden, um automatische oder manuelle Überprüfungen der Kontingente zu ermöglichen.

- AWS Trusted Advisor bietet eine Servicekontingent-Prüfung, die Ihre Nutzung und Ihre Kontingente für einige Aspekte einiger Services anzeigt. Es kann dabei helfen, Services zu identifizieren, die ihr Kontingent fast erreicht haben.
- AWS Management Console bietet Methoden, um Service-Quota-Werte für Services anzuzeigen, zu verwalten, neue Kontingente anzufordern, den Status von Kontingentanforderungen zu überwachen und den Verlauf von Kontingenten anzuzeigen.
- AWS CLI und CDKs bieten programmatische Methoden zur automatischen Verwaltung und Überwachung von Servicekontingenten und deren Nutzung.

Implementierungsschritte

Für Service Quotas:

- Überprüfen Sie AWS Service Quotas.
- Bestimmen Sie die verwendeten Services (wie IAM Access Analyzer), damit Sie Ihre bestehenden Service Quotas kennen. Es gibt etwa 250 AWS-Services, für die Service Quotas gelten. Bestimmen Sie dann den spezifischen Service-Quota-Namen, der für jedes Konto und jede Region verwendet werden kann. Pro Region gibt es etwa 3 000 Service-Quota-Namen.
- Ergänzen Sie diese Kontingentanalyse um AWS Config, um alle AWS-Ressourcen zu finden, die in Ihrer AWS-Konten verwendet werden.
- Bestimmen Sie anhand von AWS CloudFormation-Daten Ihre verwendeten AWS-Ressourcen. Sehen Sie sich die Ressourcen an, die in der AWS Management Console oder über den Befehl `list-stack-resources` AWS CLI in der Befehlszeilenschnittstelle erstellt wurden. Sie können zudem Ressourcen anzeigen, die für die Bereitstellung in der Vorlage selbst konfiguriert sind.
- Ermitteln Sie alle für die Workload erforderlichen Services durch Untersuchung des Bereitstellungscodes.
- Ermitteln Sie die geltenden Servicekontingente. Nutzen Sie die programmgesteuert über Trusted Advisor und Service Quotas zugänglichen Informationen.

- Richten Sie eine automatisierte Überwachungsmethode ein (siehe REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen und REL01-BP04 Überwachen und Verwalten von Kontingenten), um zu warnen und zu informieren, wenn die Service Quotas fast erschöpft sind oder ihr Limit erreicht haben.
- Richten Sie eine automatische, programmatische Methode ein, um zu überprüfen, ob ein Service Quota in einer Region, aber nicht in anderen Regionen desselben Kontos geändert wurde (siehe REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen und REL01-BP04 Überwachen und Verwalten von Kontingenten).
- Automatisieren Sie das Scannen von Anwendungsprotokollen und Metriken, um festzustellen, ob Fehler beim Kontingent oder bei Serviceeinschränkungen vorliegen. Falls Fehler vorhanden sind, senden Sie Warnmeldungen an das Überwachungssystem.
- Führen Sie technische Verfahren zur Berechnung der erforderlichen Kontingentänderung ein (siehe REL01-BP05 Automatisieren der Kontingentverwaltung), wenn festgestellt wird, dass für bestimmte Services größere Kontingente erforderlich sind.
- Erstellen Sie einen Bereitstellungs- und Genehmigungs-Workflow, um Änderungen am Service Quota anzufordern. Dies sollte einen Ausnahme-Workflow für den Fall umfassen, dass ein Antrag abgelehnt oder nur teilweise genehmigt wird.
- Erstellen Sie eine technische Methode zur Überprüfung von Service Quotas vor der Bereitstellung und Nutzung neuer AWS-Services, und zwar vor dem Rollout in Produktionsumgebungen oder Umgebungen mit Last (z. B. Lasttestkonto).

Bei Serviceeinschränkungen:

- Führen Sie Überwachungs- und Messmethoden ein, um auf Ressourcen aufmerksam zu machen, die ihre Ressourceneinschränkungen fast erreicht haben. Nutzen Sie CloudWatch gegebenenfalls für Metriken oder Protokollüberwachung.
- Legen Sie Warnschwellenwerte für jede Ressource fest, die eine für die Anwendung oder das System bedeutsame Einschränkung hat.
- Erstellen Sie Verfahren für die Verwaltung von Workflows und Infrastrukturen, um den Ressourcentyp zu ändern, wenn die Nutzungseinschränkung fast erreicht ist. Dieser Workflow sollte Lasttests beinhalten, um zu überprüfen, ob der neue Typ der richtige Ressourcentyp mit den neuen Einschränkungen ist.
- Migrieren Sie die identifizierte Ressource unter Verwendung bestehender Verfahren und Prozesse auf den empfohlenen neuen Ressourcentyp.

Ressourcen

Zugehörige bewährte Methoden:

- REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen
- REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur
- REL01-BP04 Überwachen und Verwalten von Kontingenten
- REL01-BP05 Automatisieren der Kontingentverwaltung
- REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist
- REL03-BP01 Segmentierung Ihres Workloads
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering

Zugehörige Dokumente:

- AWS Well-Architected Framework's Reliability Pillar: Availability (Säule für Zuverlässigkeit des AWS Well-Architected Framework)
- AWS Service Quotas (früher als Service Limits bezeichnet)
- Bewährte AWS Trusted Advisor-Prüfungsmethoden (siehe Abschnitt „Servicelimits")
- AWS Limit Monitor in AWS Answers
- Amazon EC2 Service Limits
- Was ist Service Quotas?
- How to Request Quota Increase (So beantragen Sie eine Kontingenterhöhung)
- Service endpoints and quotas (Service-Endpunkte und -Quoten)
- Service Quotas-Benutzerhandbuch
- Quota Monitor for AWS (Kontingentüberwachung für AWS)
- AWS Fault Isolation Boundaries (AWS-Grenzen für die Fehlerisolierung)
- Availability with redundancy (Verfügbarkeit mit Redundanz)

- AWS für Daten
- What is Continuous Integration? (Was ist Continuous integration?)
- What is Continuous Delivery? (Was ist Continuous Delivery?)
- APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS (Verwaltung des Kontolebenszyklus in SaaS-Umgebungen mit Konto pro Mandant auf AWS)
- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations (Umfangreiche AWS Trusted Advisor-Empfehlungen mit AWS Organizations anzeigen)
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower (Automatisieren von Service-Limit-Erhöhungen und Enterprise Support mit AWS Control Tower)

Zugehörige Videos:

- AWS Live re:Inforce 2019 – Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas (Kontingente für AWS Services, die Service Quotas verwenden, anzeigen und verwalten)
- AWS IAM Quotas Demo (AWS IAM-Kontingente – Demo)

Zugehörige Tools:

- Amazon CodeGuru Reviewer
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager

- AWS Marketplace

REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen

Wenn Sie mehrere Konten oder Regionen verwenden, fordern Sie die entsprechenden Kontingente in allen Umgebungen an, in denen die Produktions-Workloads ausgeführt werden.

Gewünschtes Ergebnis: Services und Anwendungen sollten bei Konfigurationen, die sich über Konten oder Regionen erstrecken oder die über ein Resilienzdesign mit Zonen-, Regions- oder Konto-Failover verfügen, nicht von der Erschöpfung des Service Quota betroffen sein.

Typische Anti-Muster:

- Es wird zugelassen, dass die Ressourcennutzung in einer Isolationsregion zunimmt, ohne dass es einen Mechanismus zur Aufrechterhaltung der Kapazität in den anderen Zonen gibt.
- Alle Kontingente werden manuell und in jeder Isolationsregion einzeln festgelegt.
- Nichtberücksichtigung der Auswirkungen von Ausfallsicherheitsarchitekturen (wie aktiv oder passiv) auf den künftigen Kontingentbedarf bei einer Verschlechterung in der nicht primären Region.
- Keine regelmäßige Bewertung der Kontingente und Durchführung der erforderlichen Änderungen in jeder Region und jedem Konto, in dem die Workload ausgeführt wird.
- Keine Nutzung von Vorlagen für Kontingentanforderungen, um Erhöhungen für mehrere Regionen und Konten zu beantragen.
- Keine Aktualisierung von Service Quotas, weil man fälschlicherweise davon ausgeht, dass eine Erhöhung der Kontingente Kosten nach sich zieht, wie z. B. Anforderungen von Rechenkapazitäten.

Vorteile der Einführung dieser bewährten Methode: Überprüfen, ob Sie Ihre aktuelle Last in sekundären Regionen oder Konten bewältigen können, falls regionale Services nicht mehr verfügbar sind. Dies kann dazu beitragen, die Anzahl von Fehlern oder Verschlechterungen zu verringern, die beim Verlust von Regionen auftreten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Service Quotas werden pro Konto aufgezeichnet. Sofern nicht anders angegeben, gilt jedes Kontingent für eine bestimmte AWS-Region. Zusätzlich zu den Produktionsumgebungen verwalten

Sie auch Kontingente in allen anwendbaren Nicht-Produktionsumgebungen, damit Tests und Entwicklung nicht behindert werden. Die Aufrechterhaltung eines hohen Maßes an Ausfallsicherheit setzt voraus, dass die Service Quotas ständig überprüft werden (entweder automatisch oder manuell).

Da durch die Implementierung von Designs mit den Ansätzen Aktiv/Aktiv, Aktiv/Passiv – Hot, Aktiv/Passiv – Cold und Aktiv/Passiv – Pilot Light immer mehr Workloads auf die Regionen verteilt werden, ist es wichtig, alle Kontingente für Regionen und Konten zu kennen. Frühere Datenverkehrsmuster sind nicht immer ein guter Indikator dafür, ob das Service Quota korrekt eingestellt ist.

Ebenso wichtig ist, dass das Namenslimit für das Service Quota nicht immer für alle Regionen gleich ist. In einer Region kann der Wert fünf sein, in einer anderen zehn. Die Verwaltung dieser Kontingente muss sich auf dieselben Services, Konten und Regionen erstrecken, um eine gleichmäßige Ausfallsicherheit unter Last zu gewährleisten.

Stimmen Sie alle Unterschiede zwischen den Service Quotas in den verschiedenen Regionen (aktive oder passive Region) ab und schaffen Sie Prozesse, um diese Unterschiede kontinuierlich abzugleichen. Die Testpläne für passive Regions-Failover sind selten auf die aktive Spitzenkapazität skaliert, was bedeutet, dass es im Ernstfall oder bei Tabletop-Übungen nicht gelingen kann, Unterschiede bei den Service Quotas zwischen den Regionen festzustellen und die korrekten Limits einzuhalten.

Service-Quota-Abweichung, d. h. der Umstand, dass die Service-Quota-Limits für ein bestimmtes benanntes Kontingent in einer Region und nicht in allen Regionen geändert werden, müssen unbedingt verfolgt und bewertet werden. Es sollte erwogen werden, die Kontingente in Regionen mit Datenverkehr oder potenziellem Datenverkehr zu ändern.

- Wählen Sie relevante Konten und Regionen anhand von Serviceanforderungen, regulatorischen Anforderungen sowie Anforderungen für die Latenz und die Notfallwiederherstellung aus.
- Ermitteln Sie Servicekontingente für alle relevanten Konten, Regionen und Availability Zones. Die Limits gelten für ein Konto und eine Region. Diese Werte sollten auf Unterschiede hin verglichen werden.

Implementierungsschritte

- Überprüfen Sie die Service Quotas-Werte, die über eine Risikostufe der Nutzung hinausgehen. AWS Trusted Advisor bietet Warnungen bei Überschreitung der Schwellenwerte von 80 % und 90 %.

- Überprüfen Sie die Werte für Service Quotas in allen passiven Regionen (in einem Aktiv/Passiv-Design). Stellen Sie sicher, dass die Last in den sekundären Regionen bei einem Ausfall in der primären Region erfolgreich ausgeführt werden kann.
- Automatisieren Sie die Bewertung, ob es zu einer Verschiebung der Service Quotas zwischen den Regionen desselben Kontos gekommen ist, und handeln Sie entsprechend, um die Limits zu ändern.
- Wenn die Organisationseinheiten (OU) des Kunden in der unterstützten Weise strukturiert sind, sollten die Vorlagen für Service Quotas aktualisiert werden, um Änderungen an Kontingenten widerzuspiegeln, die auf mehrere Regionen und Konten angewendet werden sollen.
  - Erstellen Sie eine Vorlage und weisen Sie der Kontingentänderung Regionen zu.
  - Überprüfen Sie alle bestehenden Vorlagen für Service Quotas auf erforderliche Änderungen (Region, Limits und Konten).

Ressourcen

Zugehörige bewährte Methoden:

- REL01-BP01 Kenntnis von Servicekontingenten und Einschränkungen
- REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur
- REL01-BP04 Überwachen und Verwalten von Kontingenten
- REL01-BP05 Automatisieren der Kontingentverwaltung
- REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist
- REL03-BP01 Segmentierung Ihres Workloads
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering

Zugehörige Dokumente:

- AWS Well-Architected Framework’s Reliability Pillar: Availability (Säule für Zuverlässigkeit des AWS Well-Architected Framework)

- AWS Service Quotas (früher als Service Limits bezeichnet)
- Bewährte AWS Trusted Advisor-Prüfungsmethoden (siehe Abschnitt „Servicelimits“)
- AWS Limit Monitor in AWS Answers
- Amazon EC2 Service Limits
- Was ist Service Quotas?
- How to Request Quota Increase (So beantragen Sie eine Kontingenterhöhung)
- Service endpoints and quotas (Service-Endpunkte und -Quoten)
- Service Quotas-Benutzerhandbuch
- Quota Monitor for AWS (Kontingentüberwachung für AWS)
- AWS Fault Isolation Boundaries (AWS-Grenzen für die Fehlerisolierung)
- Availability with redundancy (Verfügbarkeit mit Redundanz)
- AWS für Daten
- What is Continuous Integration? (Was ist Continuous integration?)
- What is Continuous Delivery? (Was ist Continuous Delivery?)
- APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS (Verwaltung des Kontolebenszyklus in SaaS-Umgebungen mit Konto pro Mandant auf AWS)
- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations (Umfangreiche AWS Trusted Advisor-Empfehlungen mit AWS Organizations anzeigen)
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower (Automatisieren von Service-Limit-Erhöhungen und Enterprise Support mit AWS Control Tower)

Zugehörige Videos:

- AWS Live re:Inforce 2019 – Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas (Kontingente für AWS Services, die Service Quotas verwenden, anzeigen und verwalten)
- AWS IAM Quotas Demo (AWS IAM-Kontingente – Demo)

Zugehörige Services:

- Amazon CodeGuru Reviewer
- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur

Achten Sie auf nicht veränderbare Service-Kontingente, Service-Einschränkungen und physische Ressourcenbeschränkungen. Entwerfen Sie Architekturen für Anwendungen und Services, um zu verhindern, dass sich diese Beschränkungen auf die Zuverlässigkeit auswirken.

Beispiele hierfür sind die Netzwerkbandbreite, die Datengröße beim Aufrufen von Serverless-Funktionen, die Drosselung der Burst-Rate eines API-Gateways und die gleichzeitig mit einer Datenbank verbundenen Benutzer.

Gewünschtes Ergebnis: Die Anwendung oder der Service erbringt unter normalen Bedingungen und bei hohem Datenverkehr die erwartete Leistung. Sie wurden so konzipiert, dass sie innerhalb der für diese Ressource festgelegten Beschränkungen oder Service-Kontingente arbeiten.

Typische Anti-Muster:

- Auswahl eines Designs, das eine Ressource eines Service verwendet, ohne zu wissen, dass es Design-Einschränkungen gibt, die dazu führen, dass dieses Design beim Skalieren versagt.
- Sie führen ein Benchmarking durch, das unrealistisch ist und mit dem während der Tests die festen Kontingente für den Service erreicht werden. Sie führen beispielsweise Tests mit einem Burst-Limit durch, diese aber für einen längeren Zeitraum.
- Sie wählen ein Design aus, das nicht skaliert oder geändert werden kann, wenn feste Service-Kontingente überschritten werden müssen. Ein Beispiel wäre ein SQS-Payload von 256 KB.

- Die Überwachungsfunktion wurde nicht zur Überwachung und Benachrichtigung von/für Schwellenwerte/n für Service-Kontingente entwickelt und implementiert, die bei hohem Datenverkehr gefährdet sein könnten.

Vorteile der Nutzung dieser bewährten Methode: Es wird sichergestellt, dass die Anwendung unter allen prognostizierten Last-Levels der Services ohne Unterbrechung oder Beeinträchtigung läuft.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Im Gegensatz zu Soft-Kontingenten für Services oder Ressourcen, die durch Einheiten mit höherer Kapazität ersetzt werden können, können feste Kontingente für AWS-Services nicht geändert werden. Das bedeutet, dass alle AWS-Services dieser Art auf potenzielle harte Kapazitätsgrenzen geprüft werden müssen, wenn sie in einer Anwendung zum Einsatz kommen.

Feste Beschränkungen werden in der Service Quotas-Konsole angezeigt. Wenn die Spalten `ANPASSBAR = Nein`anzeigen, gibt es eine feste Beschränkung für den Service. Auch auf einigen Konfigurationsseiten für Ressourcen werden feste Beschränkungen angezeigt. Für Lambda gibt es zum Beispiel bestimmte feste Beschränkungen, die nicht angepasst werden können.

Wenn Sie beispielsweise eine Python-Anwendung entwerfen, die in einer Lambda-Funktion ausgeführt werden soll, sollte die Anwendung daraufhin geprüft werden, ob die Möglichkeit besteht, dass Lambda länger als 15 Minuten läuft. Wenn die Codeausführung länger als dieses Service-Kontingent dauert, müssen alternative Technologien oder Designs in Betracht gezogen werden. Wird diese Beschränkung nach der Bereitstellung in der Produktion erreicht, wird die Anwendung beeinträchtigt und gestört, bis sie wiederhergestellt werden kann. Im Gegensatz zu Soft-Kontingenten gibt es keine Möglichkeit, diese Beschränkungen zu ändern – selbst wenn ein Ereignis des Schweregrads 1 eintritt.

Sobald die Anwendung in einer Testumgebung bereitgestellt wurde, sollten Strategien eingesetzt werden, um herauszufinden, ob feste Beschränkungen erreicht werden könnten. Stresstests, Lasttests und Chaostests sollten Teil des Einführungstestplans sein.

Implementierungsschritte

- Sehen Sie sich die vollständige Liste der AWS-Services an. Diese können Sie in der Entwurfsphase der Anwendung verwenden.

- Sehen Sie sich die Soft-Kontingentbeschränkungen und Hard-Kontingentbeschränkungen der Services an. Nicht alle Beschränkungen werden in der Service Quotas-Konsole angezeigt. Einige Services zeigen die Beschränkungen an anderen Stellen an.
- Prüfen Sie bei der Entwicklung Ihrer Anwendung die geschäftlichen und technologischen Faktoren Ihres Workloads, wie z. B. Geschäftsergebnisse, Anwendungsfälle, abhängige Systeme, Verfügbarkeitsziele und Objekte für die Notfallwiederherstellung. Lassen Sie sich von Ihren geschäftlichen und technologischen Faktoren leiten, um das richtige verteilte System für Ihren Workload zu finden.
- Analysieren Sie die Last des Services über Regionen und Konten hinweg. Viele feste Beschränkungen für Services basieren auf Regionen. Einige Beschränkungen sind jedoch kontobasiert.
- Analysieren Sie die Architekturen zur Ausfallsicherheit der Ressourcen bei einem zonenbezogenen Fehler und einem Fehler in einer Region. Bei der Entwicklung von Multi-Regionen-Designs mit Aktiv/Aktiv-, Aktiv/Passiv-Hot-, Aktiv/Passiv-Cold- und Aktiv/Passiv-Pilot-Light-Ansätzen werden diese Fehlerfälle eine höhere Auslastung verursachen. Dies schafft einen potenziellen Anwendungsfall für feste Beschränkungen.

Ressourcen

Zugehörige bewährte Methoden:

- REL01-BP01 Kenntnis von Servicekontingenten und Einschränkungen
- REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen
- REL01-BP04 Überwachen und Verwalten von Kontingenten
- REL01-BP05 Automatisieren der Kontingentverwaltung
- REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist
- REL03-BP01 Segmentierung Ihres Workloads
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering

Zugehörige Dokumente:

- AWS Well-Architected Framework's Reliability Pillar: Availability (Säule für Zuverlässigkeit des AWS Well-Architected Framework)
- AWS Service Quotas (früher als Service Limits bezeichnet)
- Bewährte AWS Trusted Advisor-Prüfungsmethoden (siehe Abschnitt „Servicelimits")
- AWS Limit Monitor in AWS Answers
- Amazon EC2 Service Limits
- Was ist Service Quotas?
- How to Request Quota Increase (So beantragen Sie eine Kontingenterhöhung)
- Service endpoints and quotas (Service-Endpunkte und -Quoten)
- Service Quotas-Benutzerhandbuch
- Quota Monitor for AWS (Kontingentüberwachung für AWS)
- AWS Fault Isolation Boundaries (AWS-Grenzen für die Fehlerisolierung)
- Availability with redundancy (Verfügbarkeit mit Redundanz)
- AWS für Daten
- What is Continuous Integration? (Was ist Continuous integration?)
- What is Continuous Delivery? (Was ist Continuous Delivery?)
- APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS (Verwaltung des Kontolebenszyklus in SaaS-Umgebungen mit Konto pro Mandant auf AWS)
- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations (Umfangreiche AWS Trusted Advisor-Empfehlungen mit AWS Organizations anzeigen)
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower (Automatisieren von Service-Limit-Erhöhungen und Enterprise Support mit AWS Control Tower)
- Aktionen, Ressourcen und Bedingungsschlüssel für Service Quotas

Zugehörige Videos:

- AWS Live re:Inforce 2019 – Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas (Kontingente für AWS Services, die Service Quotas verwenden, anzeigen und verwalten)

- AWS IAM Quotas Demo (AWS IAM-Kontingente – Demo)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (AWS re:Invent 2018: Details und Strategien: Wie man die Kontrolle über große und kleine Systeme übernimmt)

Zugehörige Tools:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP04 Überwachen und Verwalten von Kontingenten

Überprüfen Sie die potenzielle Nutzung und erhöhen Sie Ihre Kontingente entsprechend, um einen geplanten Nutzungsanstieg zu ermöglichen.

Gewünschtes Ergebnis: Es wurden aktive und automatisierte Verwaltungs- und Überwachungssysteme bereitgestellt. Diese operativen Lösungen reagieren, wenn die Schwellenwerte für die Kontingentnutzung fast erreicht werden. Sie lösen die Situation durch die proaktiven Änderungen des Kontingents.

Typische Anti-Muster:

- Keine Konfigurationsüberwachung zur Prüfung von Schwellenwerten für das Service-Kontingent.
- Keine Konfigurationsüberwachung für feste Beschränkungen, auch wenn diese Werte nicht geändert werden können.
- Sie gehen davon aus, dass eine Änderung des Soft-Kontingents direkt stattfindet oder nur wenig Zeit erfordert.

- Es werden Warnungen für den Fall konfiguriert, dass Servicekontingente erreicht werden, aber es gibt keinen Prozess für die Reaktion auf eine entsprechende Warnung.
- Es werden nur Alarme für Services konfiguriert, die von AWS Service Quotas unterstützt werden, und es erfolgt keine Überwachung anderer AWS-Services.
- Keine Berücksichtigung der Verwaltung von Kontingenten für die Ausfallsicherheit mehrerer Regionen, wie z. B. Aktiv/Aktiv-, Aktiv/Passiv-Hot-, Aktiv/Passiv-Cold- und Aktiv/Passiv-Pilot-Light-Ansätze.
- Keine Bewertung der Kontingentunterschiede zwischen den Regionen.
- Keine Bewertung des Bedarfs in jeder Region für eine bestimmte Kontingentserhöhung.
- Keine Nutzung von Vorlagen für die Verwaltung von Kontingenten für mehrere Regionen.

Vorteile der Nutzung dieser bewährten Methode: Automatische Verfolgung der AWS Service Quotas und die Überwachung der Nutzung dieser Kontingente ermöglichen die Erkennung von nahen Kontingentbeschränkungen. Sie können diese Überwachungsdaten außerdem nutzen, um Verschlechterungen aufgrund einer Kontingentausschöpfung zu begrenzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Bei unterstützten Services können Sie Ihre Kontingente überwachen, indem Sie verschiedene Services zur Bewertung und anschließenden Versendung von Warnungen konfigurieren. Auf diese Weise können Sie die Nutzung überwachen und werden auf sich nähernde Kontingentgrenzen aufmerksam gemacht. Diese Warnungen können von AWS Config, Lambda-Funktionen, Amazon CloudWatch oder von AWS Trusted Advisor ausgelöst werden. Sie können außerdem metrische Filter auf CloudWatch Logs verwenden, um Muster in den Protokollen zu suchen und zu extrahieren, und so festzustellen, ob sich die Nutzung den Schwellenwerten für Kontingente nähert.

Implementierungsschritte

Für die Überwachung:

- Erfassen Sie den aktuellen Ressourcenverbrauch (z. B. Buckets oder Instances). Nutzen Sie Service-API-Operationen, wie z. B. die Amazon EC2 `DescribeInstances` API, um die aktuelle Nutzung von Ressourcen zu erfassen.
- Erfassen Sie Ihre aktuellen Kontingente, die für die Services wesentlich und anwendbar sind. Nutzen Sie dazu:

  - AWS Service Quotas
  - AWS Trusted Advisor
  - AWS-Dokumentation
  - Entsprechende Seiten von AWS-Services
  - AWS Command Line Interface (AWS CLI)
  - AWS Cloud Development Kit (AWS CDK)
- Verwenden Sie AWS Service Quotas, ein AWS-Service, der Sie bei der Verwaltung von mehr als 250 AWS-Services an einem einzigen Ort unterstützt.
- Nutzen Sie Trusted Advisor-Service-Beschränkungen, um Ihre aktuellen Service-Beschränkungen zu verschiedenen Schwellenwerten zu überwachen.
- Nutzen Sie die Historie der Service-Kontingente (Konsole oder AWS CLI), um regionale Erhöhungen zu prüfen.
- Vergleichen Sie die Änderungen der Service-Kontingente in jeder Region und jedem Konto, um bei Bedarf auszugleichen.

Für die Verwaltung:

- Automatisiert: Richten Sie eine angepasste AWS Config-Regel ein, um Service-Kontingente in den Regionen zu prüfen und Abweichungen zu ermitteln.
- Automatisiert: Richten Sie eine geplante Lambda-Funktion ein, um Service-Kontingente in den Regionen zu scannen und Abweichungen zu ermitteln.
- Manuell: Scannen Sie Service-Kontingente über AWS CLI, die API oder die AWS-Konsole, um Service-Kontingente in den Regionen zu scannen und Abweichungen zu ermitteln. Erstellen Sie einen Bericht zu den Abweichungen.
- Wenn Abweichungen in den Kontingenten zwischen den Regionen festgestellt werden, fordern Sie bei Bedarf eine Kontingentänderung an.
- Überprüfen Sie das Ergebnis aller Anforderungen.

Ressourcen

Zugehörige bewährte Methoden:

- REL01-BP01 Kenntnis von Servicekontingenten und Einschränkungen
- REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen

- REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur
- REL01-BP05 Automatisieren der Kontingentverwaltung
- REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist
- REL03-BP01 Segmentierung Ihres Workloads
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering

Zugehörige Dokumente:

- AWS Well-Architected Framework's Reliability Pillar: Availability (Säule für Zuverlässigkeit des AWS Well-Architected Framework)
- AWS Service Quotas (früher als Service Limits bezeichnet)
- Bewährte AWS Trusted Advisor-Prüfungsmethoden (siehe Abschnitt „Servicelimits“)
- AWS Limit Monitor in AWS Answers
- Amazon EC2 Service Limits
- Was ist Service Quotas?
- How to Request Quota Increase (So beantragen Sie eine Kontingenterhöhung)
- Service endpoints and quotas (Service-Endpunkte und -Quoten)
- Service Quotas-Benutzerhandbuch
- Quota Monitor for AWS (Kontingentüberwachung für AWS)
- AWS Fault Isolation Boundaries (AWS-Grenzen für die Fehlerisolierung)
- Availability with redundancy (Verfügbarkeit mit Redundanz)
- AWS für Daten
- What is Continuous Integration? (Was ist Continuous integration?)
- What is Continuous Delivery? (Was ist Continuous Delivery?)
- APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS (Verwaltung des Kontolebenszyklus in SaaS-Umgebungen mit Konto pro Mandant auf AWS)

- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations (Umfangreiche AWS Trusted Advisor-Empfehlungen mit AWS Organizations anzeigen)
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower (Automatisieren von Service-Limit-Erhöhungen und Enterprise Support mit AWS Control Tower)
- Aktionen, Ressourcen und Bedingungsschlüssel für Service Quotas

Zugehörige Videos:

- AWS Live re:Inforce 2019 – Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas (Kontingente für AWS Services, die Service Quotas verwenden, anzeigen und verwalten)
- AWS IAM Quotas Demo (AWS IAM-Kontingente – Demo)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (AWS re:Invent 2018: Details und Strategien: Wie man die Kontrolle über große und kleine Systeme übernimmt)

Zugehörige Tools:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru
- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL01-BP05 Automatisieren der Kontingentverwaltung

Implementieren Sie Tools, um vor dem Erreichen von Schwellenwerten benachrichtigt zu werden. Durch die Verwendung von AWS Service Quotas-APIs können Sie Anfragen zur Kontingenterhöhung automatisieren.

Wenn Sie Ihre Konfigurationsmanagementdatenbank (CMDB) oder das Ticketing-System mit Service Quotas integrieren, können Sie die Verfolgung von Kontingenterhöhungsanfragen und von aktuellen Kontingenten automatisieren. Zusätzlich zum AWS SDK bietet Service Quotas Automatisierung unter Verwendung der AWS Command Line Interface (AWS CLI).

Gängige Antimuster:

- Die Kontingente und die Nutzung werden in Tabellen verfolgt.
- Es werden Berichte zur täglichen, wöchentlichen oder monatlichen Nutzung ausgeführt und anschließend wird die Nutzung mit den Kontingenten verglichen.

Vorteile der Einführung dieser bewährten Methode: Durch die automatisierte Nachverfolgung der AWS-Servicekontingente und die Überwachung ihrer Nutzung können Sie feststellen, wann ein Kontingent zu Neige geht. Sie können die Automatisierung einrichten, damit Sie beim Anfordern einer Kontingenterhöhung bei Bedarf unterstützt werden. Wenn sich Ihre Nutzung in die entgegengesetzte Richtung entwickelt, sollten Sie einige Kontingente reduzieren, um von den verringerten Risiken (im Falle von kompromittierten Anmeldeinformationen) und von Kosteneinsparungen zu profitieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Richten Sie eine automatisierte Überwachung ein. Implementieren Sie Tools mithilfe von SDKs, um vor dem Erreichen von Schwellenwerten benachrichtigt zu werden.
  - Nutzen Sie Service Quotas und erweitern Sie den Service mit einer Lösung zur automatisierten Kontingentüberwachung, z. B. mit AWS Limit Monitor oder einem Angebot aus AWS Marketplace.
    - Was ist Service Quotas?
    - Quota Monitor on AWS – AWS-Lösung
  - Richten Sie automatische Reaktionen anhand von Schwellenwerten für Kontingente mit Amazon SNS- und AWS Service Quotas-APIs ein.
  - Testen Sie die Automatisierung.

- Konfigurieren Sie Limit-Schwellenwerte.
- Integrieren Sie Änderungsereignisse von AWS Config-Bereitstellungspipelines, Amazon EventBridge oder Ereignisse von Drittanbietern.
- Legen Sie unnatürlich niedrige Schwellenwerte für Kontingente fest, um die Reaktionen zu testen.
- Richten Sie Trigger ein, damit bei Benachrichtigungen geeignete Maßnahmen ergriffen werden und bei Bedarf der AWS -Support kontaktiert wird.
- Lösen Sie Änderungsereignisse manuell aus.
- Führen Sie eine Ernstfallübung aus, um den Prozess für die Kontingenterhöhung zu testen.

Ressourcen

Ähnliche Dokumente:

- APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können
- AWS Marketplace: CMDB-Produkte zur Nachverfolgung von Limits
- AWS Service Quotas (früher als Service Limits bezeichnet)
- Bewährte AWS Trusted Advisor Trusted Advisor-Methoden (Prüfungen) (siehe Abschnitt „Servicelimits“)
- Quota Monitor on AWS – AWS-Lösung
- Amazon EC2 Service Limits
- Was ist Service Quotas?

Ähnliche Videos:

- AWS Live re:Inforce 2019 – Service Quotas

REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist

Wenn eine Ressource ausfällt oder nicht erreichbar ist, wird diese Ressource möglicherweise noch auf ein Kontingent angerechnet, bis sie erfolgreich beendet wird. Überprüfen Sie, ob Ihre Kontingente die Überschneidung von ausgefallenen oder nicht zugreifbaren Ressourcen und deren Ersatz abdecken. Bei der Berechnung dieser Lücke sollten Sie Anwendungsfälle wie Netzwerkfehler, Fehler in der Availability Zone oder Fehler in einer Region berücksichtigen.

Gewünschtes Ergebnis: Kleine oder große Fehler bei Ressourcen oder der Ressourcenzugänglichkeit können innerhalb der aktuellen Service-Schwellenwerte abgedeckt werden. Zonenfehler, Netzwerkfehler oder sogar regionale Fehler wurden bei der Ressourcenplanung berücksichtigt.

Typische Anti-Muster:

- Es werden Servicekontingente auf Grundlage des aktuellen Bedarfs eingerichtet, ohne dass Failover-Szenarien berücksichtigt werden.
- Keine Berücksichtigung des Prinzips der statischen Stabilität bei der Berechnung des Spitzenkontingents für einen Service.
- Keine Berücksichtigung des Potenzials nicht zugreifbarer Ressourcen bei der Berechnung des für jede Region benötigten Gesamtkontingents.
- Keine Berücksichtigung der AWS-Grenzen für die Fehlerisolierung bei einigen Services und ihrer potenziell anormalen Nutzungsmuster.

Vorteile der Nutzung dieser bewährten Methode: Wenn die Verfügbarkeit von Anwendungen durch eine Service-Störung beeinträchtigt wird, bietet Ihnen die Cloud die Möglichkeit zur Implementierung von Strategien zur Abschwächung dieser Ereignisse oder der Wiederherstellung. Zu solchen Strategien gehört oft die Erstellung zusätzlicher Ressourcen, um ausgefallene oder unzugängliche Ressourcen zu ersetzen. Ihre Kontingent-Strategie muss diese Failover-Bedingungen berücksichtigen und würde nicht zu einer zusätzlichen Verschlechterung aufgrund des Erreichens von Service-Beschränkungen führen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Berücksichtigen Sie bei der Bewertung der Kontingente auch Failover-Fälle, die aufgrund einer Verschlechterung auftreten können. Die folgenden Arten von Failover-Fällen sollten in Betracht gezogen werden:

- Eine VPC, die gestört oder auf die nicht zugreifbar ist.
- Ein Subnetz, auf das nicht mehr zugegriffen werden kann.
- Eine Availability Zone wurde so stark beeinträchtigt, dass die Erreichbarkeit vieler Ressourcen beeinträchtigt ist.
- Verschiedene Netzwerk-Routen oder Ingress- und Egress-Punkte sind blockiert oder verändert.

- Eine Region ist so stark gestört, dass die Erreichbarkeit vieler Ressourcen beeinträchtigt ist.
- Es gibt mehrere Ressourcen, aber nicht alle sind von einem Fehler in einer Region oder einer Availability Zone betroffen.

Fehler wie in der obigen Liste können der Auslöser für ein Failover-Ereignis sein. Die Entscheidung für einen Failover ist für jede Situation und jeden Kunden individuell, da die Auswirkungen auf den Geschäftsbetrieb sehr unterschiedlich sein können. Wenn Sie sich jedoch operativ für einen Failover von Anwendungen oder Services entscheiden, müssen Sie sich vor dem Ereignis mit der Kapazitätsplanung der Ressourcen am Failover-Standort und den entsprechenden Kontingenten befassen.

Überprüfen Sie die Service-Kontingente für jeden Service und berücksichtigen Sie dabei die möglichen Spitzenwerte. Diese Spitzen können mit Ressourcen zusammenhängen, die über Netzwerkproblemen oder Berechtigungen zwar noch aktiv, aber nicht erreichbar sind. Nicht beendete aktive Ressourcen werden weiterhin auf das Kontingent des Service angerechnet.

Implementierungsschritte

- Vergewissern Sie sich, dass zwischen Ihrem Service-Kontingent und Ihrer maximalen Nutzung genügend Spielraum besteht, um einen Failover oder den Verlust der Erreichbarkeit aufzufangen.
- Ermitteln Sie die Servicekontingente unter Berücksichtigung von Bereitstellungsmustern, der Verfügbarkeitsanforderungen und des Nutzungsanstiegs.
- Fordern Sie bei Bedarf Kontingenterhöhungen an. Planen Sie den erforderlichen Zeitraums bis zur Bewilligung von Kontingenterhöhungen.
- Bestimmen Sie Ihre Anforderungen an die Zuverlässigkeit (Anzahl der Neunen).
- Legen Sie Fehlerszenarien fest (z. B. Verlust einer Komponente, Availability Zone oder Region).
- Führen Sie eine Bereitstellungsmethode ein (z. B. Canary, Blau/Grün-Bereitstellung, Rot/Schwarz-Bereitstellung oder schrittweise).
- Berücksichtigen Sie einen angemessenen Puffer (z. B. 15 %) in aktuelle Limits.
- Berücksichtigen Sie gegebenenfalls Berechnungen zur statischen Stabilität (zonenbezogen und regional).
- Planen Sie den Nutzungsanstieg (z. B. durch Überwachen des Nutzungstrends).
- Berücksichtigen Sie die Auswirkungen der statischen Stabilität für Ihre kritischsten Workloads. Bewerten Sie Ressourcen entsprechend eines statisch stabilen Systems in allen Regionen und Availability Zones.

- Ziehen Sie den Einsatz von On-Demand-Kapazitätsreservierungen in Betracht, um vor einem Failover Kapazitäten zu reservieren. Diese Strategie kann während kritischer Geschäftszeiten sinnvoll sein, um potenzielle Risiken bei der Beschaffung der richtigen Menge und Art von Ressourcen während eines Failovers zu verringern.

Ressourcen

Zugehörige bewährte Methoden:

- REL01-BP01 Kenntnis von Servicekontingenten und Einschränkungen
- REL01-BP02 Verwalten von Servicekontingenten für mehrere Konten und Regionen
- REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur
- REL01-BP04 Überwachen und Verwalten von Kontingenten
- REL01-BP05 Automatisieren der Kontingentverwaltung
- REL03-BP01 Segmentierung Ihres Workloads
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering

Zugehörige Dokumente:

- AWS Well-Architected Framework's Reliability Pillar: Availability (Säule für Zuverlässigkeit des AWS Well-Architected Framework)
- AWS Service Quotas (früher als Service Limits bezeichnet)
- Bewährte AWS Trusted Advisor-Prüfungsmethoden (siehe Abschnitt „Servicelimits“)
- AWS Limit Monitor in AWS Answers
- Amazon EC2 Service Limits
- Was ist Service Quotas?
- How to Request Quota Increase (So beantragen Sie eine Kontingenterhöhung)
- Service endpoints and quotas (Service-Endpunkte und -Quoten)
- Service Quotas-Benutzerhandbuch

- Quota Monitor for AWS (Kontingentüberwachung für AWS)
- AWS Fault Isolation Boundaries (AWS-Grenzen für die Fehlerisolierung)
- Availability with redundancy (Verfügbarkeit mit Redundanz)
- AWS für Daten
- What is Continuous Integration? (Was ist Continuous integration?)
- What is Continuous Delivery? (Was ist Continuous Delivery?)
- APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können
- Managing the account lifecycle in account-per-tenant SaaS environments on AWS (Verwaltung des Kontolebenszyklus in SaaS-Umgebungen mit Konto pro Mandant auf AWS)
- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- View AWS Trusted Advisor recommendations at scale with AWS Organizations (Umfangreiche AWS Trusted Advisor-Empfehlungen mit AWS Organizations anzeigen)
- Automating Service Limit Increases and Enterprise Support with AWS Control Tower (Automatisieren von Service-Limit-Erhöhungen und Enterprise Support mit AWS Control Tower)
- Aktionen, Ressourcen und Bedingungsschlüssel für Service Quotas

Zugehörige Videos:

- AWS Live re:Inforce 2019 – Service Quotas
- View and Manage Quotas for AWS Services Using Service Quotas (Kontingente für AWS Services, die Service Quotas verwenden, anzeigen und verwalten)
- AWS IAM Quotas Demo (AWS IAM-Kontingente – Demo)
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (AWS re:Invent 2018: Details und Strategien: Wie man die Kontrolle über große und kleine Systeme übernimmt)

Zugehörige Tools:

- AWS CodeDeploy
- AWS CloudTrail
- Amazon CloudWatch
- Amazon EventBridge
- Amazon DevOps Guru

- AWS Config
- AWS Trusted Advisor
- AWS CDK
- AWS Systems Manager
- AWS Marketplace

REL 2. Was ist bei der Planung der Netzwerktopologie zu beachten?

REL 3. Dazu gehören mehrere Cloud-Umgebungen (öffentlich zugängliche und private) und möglicherweise die vorhandene Infrastruktur Ihres Rechenzentrums. Die Pläne müssen Netzwerkaspekte umfassen, wie z. B. die Konnektivität innerhalb und zwischen Systemen, die Verwaltung öffentlicher und privater IP-Adressen und die Auflösung von Domänennamen.

Bewährte Methoden

- REL02-BP01 Bereitstellen einer hochverfügbaren Netzwerkkonnektivität für öffentliche Endpunkte der Workload
- REL02-BP02 Bereitstellen redundanter Konnektivität zwischen privaten Netzwerken in der Cloud und in On-Premises-Umgebungen
- REL02-BP03 Berücksichtigen von Erweiterungen und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze
- REL02-BP04 Vorziehen von Hub-and-Spoke-Topologien gegenüber M-zu-N-Netzen
- REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht

REL02-BP01 Bereitstellen einer hochverfügbaren Netzwerkkonnektivität für öffentliche Endpunkte der Workload

Der Aufbau einer hochverfügbaren Netzwerkkonnektivität zu öffentlichen Endpunkten Ihres Workloads kann Ihnen helfen, Ausfallzeiten aufgrund von Konnektivitätsverlusten zu reduzieren und die Verfügbarkeit und SLA Ihres Workloads zu verbessern. Verwenden Sie dazu hochverfügbares DNS, Content Delivery Networks (CDNs), API-Gateways, Load-Balancing oder Reverse-Proxies.

Gewünschtes Ergebnis: Es ist von entscheidender Bedeutung, eine hochverfügbare Netzwerkkonnektivität für Ihre öffentlichen Endpunkte zu planen, aufzubauen und in Betrieb zu nehmen. Wenn Ihr Workload aufgrund eines Konnektivitätsverlustes nicht mehr erreichbar ist, sehen

Ihre Kunden Ihr System als ausgefallen an – selbst wenn Ihr Workload läuft und verfügbar ist. Durch die Kombination einer hochverfügbaren und stabilen Netzwerkkonnektivität für die öffentlichen Endpunkte Ihres Workloads mit einer stabilen Architektur für Ihren Workload selbst können Sie Ihren Kunden die bestmögliche Verfügbarkeit und das bestmögliche Serviceniveau bieten.

AWS Global Accelerator, Amazon CloudFront, Amazon API Gateway, AWS Lambda-Funktions-URLs, AWS AppSync-APIs und Elastic Load Balancing (ELB) bieten alle hochverfügbare öffentliche Endpunkte. Amazon Route 53 bietet einen hochverfügbaren DNS-Service für die Auflösung von Domänennamen, um sicherzustellen, dass die Adressen Ihrer öffentlichen Endpunkte aufgelöst werden können.

Sie können außerdem AWS Marketplace-Software-Appliances für das Load-Balancing und für Proxys nutzen.

Typische Anti-Muster:

- Entwurf eines hochverfügbaren Workloads, ohne eine DNS- und Netzwerkkonnektivität mit hoher Verfügbarkeit einzuplanen.
- Verwendung öffentlicher Internetadressen auf einzelnen Instances oder Containern und Verwalten der Konnektivität zu diesen per DNS.
- Verwendung von IP-Adressen anstelle von Domänennamen zur Lokalisierung von Services.
- Keine Tests von Szenarien, in denen die Konnektivität zu Ihren öffentlichen Endpunkten verloren geht.
- Keine Analyse des Bedarfs für den Netzwerkdurchsatz und die Verteilungsmuster im Netzwerk.
- Keine Tests und Planungen für Szenarien, in denen die Internet-Netzwerkkonnektivität zu Ihren öffentlichen Endpunkten der Workloads unterbrochen werden könnte.
- Bereitstellen von Inhalten (z. B. Webseiten, statische Komponenten oder Mediendateien) für ein großes geografisches Gebiet ohne Verwendung eines Content-Delivery-Networks.
- Keine Planung für Distributed Denial of Service (DDoS)-Angriffe. Bei DDoS-Angriffen besteht die Gefahr, dass der legitime Datenverkehr unterbrochen wird und die Verfügbarkeit für Ihre Benutzer sinkt.

Vorteile der Nutzung dieser bewährten Methode: Die Planung einer hochverfügbaren und stabilen Netzwerkkonnektivität stellt sicher, dass Ihr Workload für Ihre Benutzer zugreifbar und verfügbar ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Das Wichtigste beim Aufbau einer hochverfügbaren Netzwerkkonnektivität zu Ihren öffentlichen Endpunkten ist das Routing des Datenverkehrs. Um sicherzustellen, dass Ihr Datenverkehr die Endpunkte erreichen kann, muss das DNS in der Lage sein, die Domänennamen in die entsprechenden IP-Adressen aufzulösen. Verwenden Sie ein hochverfügbares und skalierbares Domain Name System (DNS) wie Amazon Route 53, um die DNS-Einträge Ihrer Domäne zu verwalten. Sie können außerdem die von Amazon Route 53 bereitgestellten Zustandsprüfungen verwenden. Die Zustandsprüfungen überprüfen, ob Ihre Anwendung erreichbar, verfügbar und funktionstüchtig ist. Sie können so eingerichtet werden, dass sie das Verhalten Ihres Benutzers nachahmen, z. B. das Anfordern einer Webseite oder einer bestimmten URL. Im Falle eines Fehlers reagiert Amazon Route 53 auf DNS-Auflösungsanfragen und leitet den Datenverkehr nur an Health-Endpunkte weiter. Sie können außerdem die von Amazon Route 53 angebotenen Funktionen für Geo-DNS und latenzbasiertes Routing nutzen.

Um zu überprüfen, ob Ihr Workload selbst hochverfügbar ist, verwenden Sie Elastic Load Balancing (ELB). Amazon Route 53 kann verwendet werden, um den Datenverkehr an ELB zu leiten, das den Datenverkehr an die Ziel-Computing-Instances verteilt. Sie können Amazon API Gateway außerdem zusammen mit AWS Lambda für eine Serverless-Lösung verwenden. Kunden können Workloads zudem in mehreren AWS-Regionen ausführen. Mit einem Multi-Site Aktiv/Aktiv-Muster kann der Workload den Datenverkehr aus mehreren Regionen bedienen. Bei einem Multi-Site Aktiv/Passiv-Muster bedient der Workload den Datenverkehr aus der aktiven Region, während die Daten in die sekundäre Region repliziert werden, die im Falle eines Fehlers in der primären Region aktiv wird. Mit Route 53-Zustandsprüfungen können Sie dann das DNS-Failover von einem beliebigen Endpunkt in einer primären Region zu einem Endpunkt in einer sekundären Region steuern und so sicherstellen, dass Ihr Workload erreichbar und für Ihre Benutzer verfügbar ist.

Amazon CloudFront bietet eine einfache API für die Verteilung von Inhalten mit geringer Latenz und hohen Datenübertragungsraten, indem Anfragen über ein Netzwerk von Edge-Standorten auf der ganzen Welt bedient werden. Content Delivery Networks (CDNs) dienen den Kunden, indem sie Inhalte bereitstellen, die sich in der Nähe des Benutzers befinden oder dort zwischengespeichert werden. Dies verbessert auch die Verfügbarkeit Ihrer Anwendung, da die Last der Inhalte von Ihren Servern auf die Edge-Standorte von CloudFront verlagert wird. Die Edge-Standorte und regionalen Edge-Caches halten zwischengespeicherte Kopien Ihrer Inhalte in der Nähe Ihrer Benutzer vor, was einen schnellen Abruf ermöglicht und die Erreichbarkeit und Verfügbarkeit Ihres Workloads erhöht.

Bei Workloads mit geografisch verteilten Benutzern hilft AWS Global Accelerator Ihnen, die Verfügbarkeit und Leistung der Anwendungen zu verbessern. AWS Global Accelerator bietet

statische Anycast-IP-Adressen, die als fester Zugangspunkt zu Ihrer Anwendung dienen, die in einer oder mehreren AWS-Regionen gehostet wird. Dadurch kann der Datenverkehr so nah wie möglich an Ihren Benutzern in das globale AWS Netzwerk geleitet werden, was die Erreichbarkeit und Verfügbarkeit Ihres Workloads verbessert. AWS Global Accelerator überwacht außerdem den Zustand Ihrer Anwendungsendpunkte mithilfe von TCP-, HTTP- und HTTPS-Zustandsprüfungen. Jede Änderung im Zustand oder in der Konfiguration Ihrer Endpunkte leitet den Benutzerverkehr auf funktionierende Endpunkte weiter, die Ihren Benutzern die beste Leistung und Verfügbarkeit bieten. Darüber hinaus verfügt AWS Global Accelerator über ein fehlerisolierendes Design, das zwei statische IPv4-Adressen verwendet, die von unabhängigen Netzwerkzonen bedient werden und die Verfügbarkeit Ihrer Anwendungen erhöhen.

Um Kunden vor DDoS-Angriffen zu schützen, bietet AWS AWS Shield Standard. Shield Standard wird automatisch aktiviert und schützt vor gängigen Infrastrukturangriffen (Layer 3 und 4) wie SYN/UDP-Floods und Reflection-Angriffen, um die hohe Verfügbarkeit Ihrer Anwendungen auf AWS zu unterstützen. Für zusätzlichen Schutz vor ausgefeilteren und größeren Angriffen (wie UDP-Floods), State-Exhaustion-Angriffen (wie TCP-SYN-Floods) und zum Schutz Ihrer Anwendungen, die auf Amazon Elastic Compute Cloud (Amazon EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator und Route 53 ausgeführt werden, können Sie AWS Shield Advanced verwenden. Zum Schutz vor Angriffen auf der Anwendungsebene wie HTTP-POST- oder GET-Floods verwenden Sie AWS WAF. AWS WAF kann IP-Adressen, HTTP-Header, HTTP-Body, URI-Strings, SQL-Injections und Cross-Site-Scripting-Bedingungen verwenden, um zu bestimmen, ob eine Anfrage blockiert oder zugelassen werden soll.

Implementierungsschritte

1. Richten Sie ein hochverfügbares DNS ein: Amazon Route 53 ist ein hochverfügbarer und skalierbarer Domain Name System (DNS)-Webservice. Route 53 verbindet Benutzeranfragen mit Internetanwendungen, die auf AWS oder on-premises ausgeführt werden. Weitere Informationen finden Sie unter Konfigurieren von Amazon Route 53 als DNS-Service.
2. Richten Sie Zustandsprüfungen ein: Wenn Sie Route 53 verwenden, vergewissern Sie sich, dass nur korrekt funktionierende Ziele auflösbar sind. Starten Sie mit der Erstellung von Route 53-Zustandsprüfungen und der Konfiguration des DNS-Failovers. Bei der Einrichtung von Zustandsprüfungen sind die folgenden Aspekte zu beachten:
   a. So ermittelt Amazon Route 53, ob eine Zustandsprüfung fehlerfrei ist
   b. Erstellen, Aktualisieren und Löschen von Zustandsprüfungen
   c. Den Status von Zustandsprüfungen überwachen und Benachrichtigungen erhalten
   d. Bewährte Methoden für Amazon Route 53-DNS

3. Verbinden Sie Ihren DNS-Service mit Ihren Endpunkten.
   a. Wenn Sie Elastic Load Balancing als Ziel für Ihren Datenverkehr verwenden, erstellen Sie einen Alias-Eintrag mit Amazon Route 53, der auf den regionalen Endpunkt Ihres Load-Balancers verweist. Setzen Sie bei der Erstellung des Alias-Eintrags die Option „Zielzustand evaluieren“ auf „Ja“.
   b. Verwenden Sie bei der Nutzung von API Gateway für Serverless-Workloads oder private APIs Route 53, um den Datenverkehr zu API Gateway zu routen.
4. Entscheiden Sie sich für ein Content Delivery Netzwerk.
   a. Informieren Sie sich zunächst über die Art und Weise, wie CloudFront-Inhalte über Edge-Standorte in der Nähe des Benutzers bereitgestellt werden.
   b. Starten Sie mit einer einfachen CloudFront-Verteilung. CloudFront weiß dann, von wo aus die Inhalte ausgeliefert werden sollen, und kennt die Details zur Nachverfolgung und Verwaltung der Content-Bereitstellung. Die folgenden Aspekte sollten Sie kennen und berücksichtigen, wenn Sie die CloudFront-Verteilung einrichten:
      i. Funktionsweise der Zwischenspeicherung mit CloudFront-Edge-Standorten
      ii. Erhöhen des Anteils der Anforderungen, die direkt von den CloudFront-Caches bereitgestellt werden (Cache-Trefferverhältnis)
      iii. Verwenden von Amazon CloudFront Origin Shield
      iv. Optimieren der Hochverfügbarkeit mit CloudFront-Ursprungs-Failover
5. Einrichten des Schutzes auf der Anwendungsebene: AWS WAF hilft Ihnen, sich gegen gängige Web-Exploits und Bots zu schützen, die die Verfügbarkeit beeinträchtigen, die Sicherheit gefährden oder übermäßig viele Ressourcen verbrauchen können. Um ein tieferes Verständnis zu erlangen, lesen Sie How AWS WAF works (Funktionsweise von AWS WAF). Wenn Sie bereit sind, den Schutz vor HTTP-POST- und -GET-Floods auf der Anwendungsebene zu implementieren, lesen Sie Getting started with AWS WAF (Erste Schritte mit AWS WAF). Sie können außerdem AWS WAF mit CloudFront verwenden. In der Dokumentation erfahren Sie, wie wie AWS WAF mit Amazon CloudFront-Funktionen arbeitet.
6. Richten Sie einen zusätzlichen DDoS-Schutz ein: Standardmäßig erhalten alle Kunden von AWS mit AWS Shield Standard ohne zusätzliche Kosten einen Schutz gegen die gängigsten DDoS-Angriffe auf Netzwerk- und Transportebene, die sich gegen Ihre Website oder Anwendung richten. Für zusätzlichen Schutz von Anwendungen, die auf Amazon EC2, Elastic Load Balancing, Amazon CloudFront, AWS Global Accelerator und Amazon Route 53 ausgeführt werden, können Sie AWS Shield Advanced einsetzen und sich Beispiele für DDoS-resistente Architekturen ansehen.

Um Ihren Workload und Ihre öffentlichen Endpunkte vor DDoS-Angriffen zu schützen, lesen Sie Getting started with AWS Shield Advanced (Erste Schritte mit AWS Shield Advanced).

Ressourcen

Zugehörige bewährte Methoden:

- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL10-BP02 Auswählen der geeigneten Standorte für Ihre Multi-Standort-Bereitstellung
- REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung
- REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können
- AWS Marketplace für Netzwerkinfrastruktur
- Was ist AWS Global Accelerator?
- Was ist Amazon CloudFront?
- Was ist Amazon Route 53?
- Was ist Elastic Load Balancing?
- Network Connectivity capability – Establishing Your Cloud Foundations (Funktionalität zur Netzwerkkonnektivität – Etablieren Ihrer Cloud-Grundlagen)
- Was ist Amazon API Gateway?
- What are AWS WAF, AWS Shield, and AWS Firewall Manager? (Was sind AWS WAF, AWS Shield und AWS Firewall Manager?)
- Was ist Amazon Route 53 Application Recovery Controller?
- Benutzerdefinierte Zustandsprüfungen für das DNS-Failover konfigurieren

Zugehörige Videos:

- AWS re:Invent 2022 – Improve performance and availability with AWS Global Accelerator (AWS re:Invent 2022 – Verbessern der Leistung und Verfügbarkeit mit AWS Global Accelerator)

- AWS re:Invent 2020: Global traffic management with Amazon Route 53 (AWS re:Invent 2020: Globales Datenverkehrsmanagement mit AWS)
- AWS re:Invent 2022 – Operating highly available Multi-AZ applications (AWS re:Invent 2022 – Betrieb hochverfügbarer Multi-AZ Anwendungen)
- AWS re:Invent 2022 – Dive deep on AWS networking infrastructure (AWS re:Invent 2022 – Details zur AWS-Netzwerkinfrastruktur)
- AWS re:Invent 2022 – Building resilient networks (AWS re:Invent 2022 – Aufbau widerstandsfähiger Netzwerke)

Zugehörige Beispiele:

- Disaster Recovery with Amazon Route 53 Application Recovery Controller (ARC) (Notfallwiederherstellung mit Amazon Route 53 Application Recovery Controller (ARC))
- Workshops zur Zuverlässigkeit
- AWS Global Accelerator-Workshop

REL02-BP02 Bereitstellen redundanter Konnektivität zwischen privaten Netzwerken in der Cloud und in On-Premises-Umgebungen

Implementieren Sie Redundanz in Ihren Verbindungen zwischen privaten Netzwerken in der Cloud und On-Premises-Umgebungen, um die Stabilität der Konnektivität zu erreichen. Dies kann erreicht werden, indem zwei oder mehr Verbindungen und Datenverkehrspfade bereitgestellt werden, sodass die Konnektivität bei Netzwerkausfällen erhalten bleibt.

Typische Anti-Muster:

- Sie verlassen sich auf nur eine Netzwerkverbindung, was zu einer einzigen Fehlerquelle führt.
- Sie verwenden nur einen VPN-Tunnel oder mehrere Tunnel, die in derselben Availability Zone enden.
- Sie verlassen sich bei der VPN-Konnektivität auf einen ISP, was bei ISP-Ausfällen zu kompletten Ausfällen führen kann.
- Keine Implementierung dynamischer Routing-Protokolle wie BGP, die für die Umleitung des Datenverkehrs bei Netzwerkunterbrechungen von entscheidender Bedeutung sind.
- Sie ignorieren die Bandbreitenbeschränkungen von VPN-Tunneln und überschätzen deren Backup-Fähigkeiten.

Vorteile der Implementierung dieser bewährten Methoden: Durch die Implementierung redundanter Konnektivität zwischen Ihrer Cloud-Umgebung und Ihrer Unternehmens- bzw. On-Premises-Umgebung wird die sichere Kommunikation der abhängigen Services zwischen den beiden Umgebungen gewährleistet.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Wenn Sie AWS Direct Connect verwenden, um Ihr On-Premises-Netzwerk mit AWS zu verbinden, können Sie maximale Netzwerkstabilität (SLA von 99,99 %) erreichen, indem Sie separate Verbindungen verwenden, die auf verschiedenen Geräten an mehr als einem On-Premises-Standort und mehr als einem AWS Direct Connect-Standort enden. Diese Topologie bietet Widerstandsfähigkeit gegen Geräteausfälle, Verbindungsprobleme und komplette Standortausfälle. Alternativ können Sie eine hohe Ausfallsicherheit (SLA von 99,9 %) erreichen, indem Sie zwei einzelne Verbindungen zu mehreren Standorten verwenden (jeder On-Premises-Standort ist mit einem einzigen Direct-Connect-Standort verbunden). Dieser Ansatz schützt vor Verbindungsunterbrechungen, die durch getrennte Glasfaserkabel oder Geräteausfälle verursacht werden, und trägt dazu bei, die Auswirkungen kompletter Standortausfälle zu mindern. Das AWS Direct Connect Resiliency Toolkit unterstützt Sie beim Entwerfen Ihrer AWS Direct Connect-Topologie.

Sie können auch erwägen, dass AWS Site-to-Site VPN auf einem AWS Transit Gateway endet, um ein kostengünstiges Backup Ihrer primären Verbindung mit AWS Direct Connect zu erhalten. Dieses Setup ermöglicht Equal-Cost-Multipath (ECMP)-Routing über mehrere VPN-Tunnel und ermöglicht einen Durchsatz von bis zu 50 Gbit/s, obwohl jeder VPN-Tunnel auf 1,25 Gbit/s begrenzt ist. Es ist jedoch wichtig zu beachten, dass AWS Direct Connect immer noch die effektivste Wahl ist, um Netzwerkunterbrechungen zu minimieren und eine stabile Konnektivität bereitzustellen.

Wenn Sie VPNs über das Internet verwenden, um Ihre Cloud-Umgebung mit Ihrem On-Premises-Rechenzentrum zu verbinden, konfigurieren Sie zwei VPN-Tunnel als Teil einer einzigen Site-to-Site-VPN-Verbindung. Jeder Tunnel sollte aus Gründen der Hochverfügbarkeit in einer anderen Availability Zone enden und redundante Hardware verwenden, um Ausfälle von On-Premises-Geräten zu verhindern. Erwägen Sie außerdem mehrere Internetverbindungen von verschiedenen Internetdienstanbietern (ISPs) an Ihrem On-Premises-Standort, um eine vollständige Unterbrechung der VPN-Konnektivität durch den Ausfall eines einzigen ISP zu vermeiden. Die Auswahl von ISPs mit unterschiedlichem Routing und Infrastruktur, insbesondere solchen mit separaten physischen Pfaden zu AWS-Endpunkten, sorgt für eine hohe Konnektivitätsverfügbarkeit.

Neben der physischen Redundanz mit mehreren AWS Direct Connect-Verbindungen und VPN-Tunneln (oder einer Kombination aus beiden) ist auch die Implementierung des dynamischen Routings des Border Gateway Protocol (BGP) von entscheidender Bedeutung. Dynamisches BGP ermöglicht die automatische Umleitung des Datenverkehrs von einem Pfad zum nächsten, basierend auf Netzwerkbedingungen in Echtzeit und konfigurierten Richtlinien. Dieses dynamische Verhalten ist besonders vorteilhaft zur Aufrechterhaltung der Netzwerkverfügbarkeit und Servicekontinuität bei Verbindungs- oder Netzwerkausfällen. Es wählt schnell alternative Pfade aus und verbessert so die Ausfallsicherheit und Zuverlässigkeit des Netzwerks.

Implementierungsschritte

- Erwerben Sie hochverfügbare Konnektivität zwischen AWS und Ihrer On-Premises-Umgebung.
  - Verwenden Sie mehrere AWS Direct Connect-Verbindungen oder VPN-Tunnel zwischen separat bereitgestellten privaten Netzwerken.
  - Verwenden Sie für eine hohe Verfügbarkeit mehrere AWS Direct Connect-Standorte.
  - Wenn Sie mehrere AWS-Regionen verwenden, sorgen Sie in mindestens zwei davon für Redundanz.
- Verwenden Sie wenn möglich AWS Transit Gateway, um Ihre VPN-Verbindung zu beenden.
- Beurteilen sie AWS Marketplace-Appliances, um VPNs zu beenden oder erweitern Sie Ihr SD-WAN auf AWS. Stellen Sie bei Verwendung von AWS Marketplace-Appliances redundante Instances bereit, um eine hohe Verfügbarkeit in verschiedenen Availability Zones zu gewährleisten.
- Stellen Sie auf Ihrer On-Premises-Umgebung eine redundante Verbindung her.
  - Möglicherweise benötigen Sie redundante Verbindungen zu mehreren AWS-Regionen, um die erforderliche Verfügbarkeit zu gewährleisten.
  - Verwenden Sie das AWS Direct Connect Resiliency Toolkit, um loszulegen.

Ressourcen

Zugehörige Dokumente:

- AWS Direct Connect-Resilienzempfehlungen
- Verwendung redundanter Site-to-Site VPN-Verbindungen zur Bereitstellung eines Failovers
- Routing-Richtlinien und BGP-Communities
- Aktiv/Aktiv- und Aktiv/Passiv-Konfigurationen in AWS Direct Connect
- APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können

- AWS Marketplace für Netzwerkinfrastruktur
- Amazon Virtual Private Cloud-Konnektivitätsoptionen – Whitepaper
- Erstellen einer skalierbaren und sicheren Multi-VPC-AWS-Netzwerkinfrastruktur
- Verwendung redundanter Site-to-Site VPN-Verbindungen zur Bereitstellung eines Failovers
- Erste Schritte mit dem AWS Direct Connect Resiliency Toolkit
- VPC-Endpunkte und VPC-Endpunktservices (AWS PrivateLink)
- Was ist Amazon VPC?
- Was ist ein Transit-Gateway?
- Was ist AWS Site-to-Site VPN?
- Arbeiten mit Direct-Connect-Gateways

Zugehörige Videos:

- AWS re:Invent 2018: Erweitertes VPC-Design und neue Funktionen für Amazon VPC
- AWS re:Invent 2019: AWS Transit Gateway-Referenzarchitekturen für viele VPCs

REL02-BP03 Berücksichtigen von Erweiterungen und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze

Die IP-Adressbereiche für Amazon VPC müssen ausreichend groß sein, um die Anforderungen einer Workload zu erfüllen. Dabei sind zukünftige Erweiterungen und Zuweisungen von IP-Adressen zu Subnetzen in verschiedenen Availability Zones zu berücksichtigen. Dies betrifft Load Balancer, EC2-Instances sowie containerbasierte Anwendungen.

Wenn Sie Ihre Netzwerktopologie planen, besteht der erste Schritt in der Definition des IP-Adressbereichs. Private IP-Adressbereiche (gemäß RFC 1918-Richtlinien) sollten jeder VPC zugewiesen werden. Berücksichtigen Sie im Rahmen dieses Prozesses die folgenden Anforderungen:

- Ermöglichen Sie einen IP-Adressbereich für mehr als eine VPC pro Region.
- Planen Sie innerhalb einer VPC Platz für mehrere Subnetze ein, damit Sie mehrere Availability Zones abdecken können.
- Lassen Sie für eine zukünftige Erweiterung stets Raum für nicht verwendete CIDR-Blöcke innerhalb einer VPC.

- Stellen Sie sicher, dass ein IP-Adressbereich vorhanden ist, um die Anforderungen von temporären Amazon EC2-Instances zu erfüllen, die Sie möglicherweise verwenden, z. B. Spot-Flotten für Machine Learning, Amazon EMR-Cluster oder Amazon Redshift-Cluster. Ähnliche Überlegungen sollten für Kubernetes-Cluster wie Amazon Elastic Kubernetes Service (Amazon EKS) getroffen werden, da jedem Kubernetes-Pod standardmäßig eine routbare Adresse aus dem VPC-CIDR-Block zugewiesen wird.
- Beachten Sie, dass die ersten vier IP-Adressen und die letzte IP-Adresse in jedem Subnetz-CIDR-Block reserviert und nicht für Sie verfügbar sind.
- Beachten Sie, dass der VPC CIDR-Block, der anfänglich Ihrer VPC zugewiesen war, nicht geändert oder gelöscht werden kann. Sie können der VPC jedoch zusätzliche, nicht überlappende CIDR-Blöcke hinzufügen. IPv4-CIDRs für Subnetze können nicht geändert werden, IPv6 CIDRs jedoch schon.
- Der größte mögliche VPC-CIDR-Block entspricht /16 und der kleinste /28.
- Berücksichtigen Sie andere verbundene Netzwerke (VPC, On-Premises oder sonstige Cloud-Anbieter) und stellen Sie sicher, dass sich der IP-Adressraum nicht überschneidet. Weitere Informationen finden Sie unter REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht.

Gewünschtes Ergebnis: Ein skalierbares IP-Subnetz kann Ihnen helfen, zukünftiges Wachstum zu bewältigen und unnötige Verschwendung zu vermeiden.

Typische Anti-Muster:

- Wenn zukünftiges Wachstum nicht berücksichtigt wird, sind die CIDR-Blöcke zu klein und müssen neu konfiguriert werden, was zu Ausfallzeiten führen kann.
- Es wird falsch eingeschätzt, wie viele IP-Adressen ein Elastic Load Balancer verwenden kann.
- Es werden viele Load Balancer mit hohem Datenverkehr in denselben Subnetzen bereitgestellt.
- Es werden automatische Skalierungsmechanismen verwendet, während der Verbrauch von IP-Adressen nicht überwacht wird.
- Die Definition übermäßig großer CIDR-Bereiche liegt weit über den zukünftigen Wachstumserwartungen, was zu Schwierigkeiten beim Peering mit anderen Netzwerken mit überlappenden Adressbereichen führen kann.

Vorteile der Nutzung dieser bewährten Methode: So wird sichergestellt, dass Sie das Wachstum Ihrer Workloads bewältigen können und beim Hochskalieren weiterhin die entsprechende Verfügbarkeit bereitstellen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Berücksichtigen Sie bei der Planung Ihres Netzwerks Ihr zukünftiges Wachstum, die Einhaltung gesetzlicher Vorschriften sowie die Kompatibilität mit anderen Netzwerken. Das Wachstum kann unterschätzt werden, gesetzliche Vorschriften können sich ändern, und bei Unternehmensübernahmen oder privaten Netzwerkverbindungen kann die Implementierung ohne fundierte Planung zur Herausforderung werden.

- Wählen Sie relevante AWS-Konten und Regionen anhand von Serviceanforderungen, regulatorischen Anforderungen sowie Anforderungen für die Latenz und die Notfallwiederherstellung aus.
- Identifizieren Sie Ihre Anforderungen bezüglich regionaler VPC-Bereitstellungen.
- Ermitteln Sie die erforderliche Größe der VPCs.
  - Ermitteln Sie, ob Multi-VPC-Konnektivität bereitgestellt werden soll.
    - Was ist ein Transit-Gateway?
    - Multi-VPC-Konnektivität in einer Region
  - Ermitteln Sie, ob aufgrund von Compliance-Anforderungen getrennte Netzwerke erforderlich sind.
  - Erstellen Sie VPCs mit CIDR-Blöcken in geeigneter Größe, um Ihren aktuellen und zukünftigen Anforderungen gerecht zu werden.
    - Wenn Sie unbekannte Wachstumsprognosen haben, sollten Sie sich für größere CIDR-Blöcke entscheiden, um das Potenzial einer zukünftigen Neukonfiguration zu verringern.
  - Erwägen Sie die Verwendung von IPv6-Adressierung für Subnetze als Teil einer Dual-Stack-VPC. IPv6 eignet sich gut für den Einsatz in privaten Subnetzen, die Flotten kurzlebiger Instances oder Container enthalten, für die andernfalls eine große Anzahl von IPv4-Adressen erforderlich wäre.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können
- AWS Marketplace für Netzwerkinfrastruktur
- Amazon Virtual Private Cloud-Konnektivitätsoptionen – Whitepaper
- Hochverfügbare Netzwerkkonnektivität zwischen mehreren Rechenzentren
- Multi-VPC-Konnektivität in einer Region
- Was ist Amazon VPC?
- IPv6 in AWS
- IPv6 in Referenzarchitekturen
- Amazon Elastic Kubernetes Service startet IPv6-Support

Zugehörige Videos:

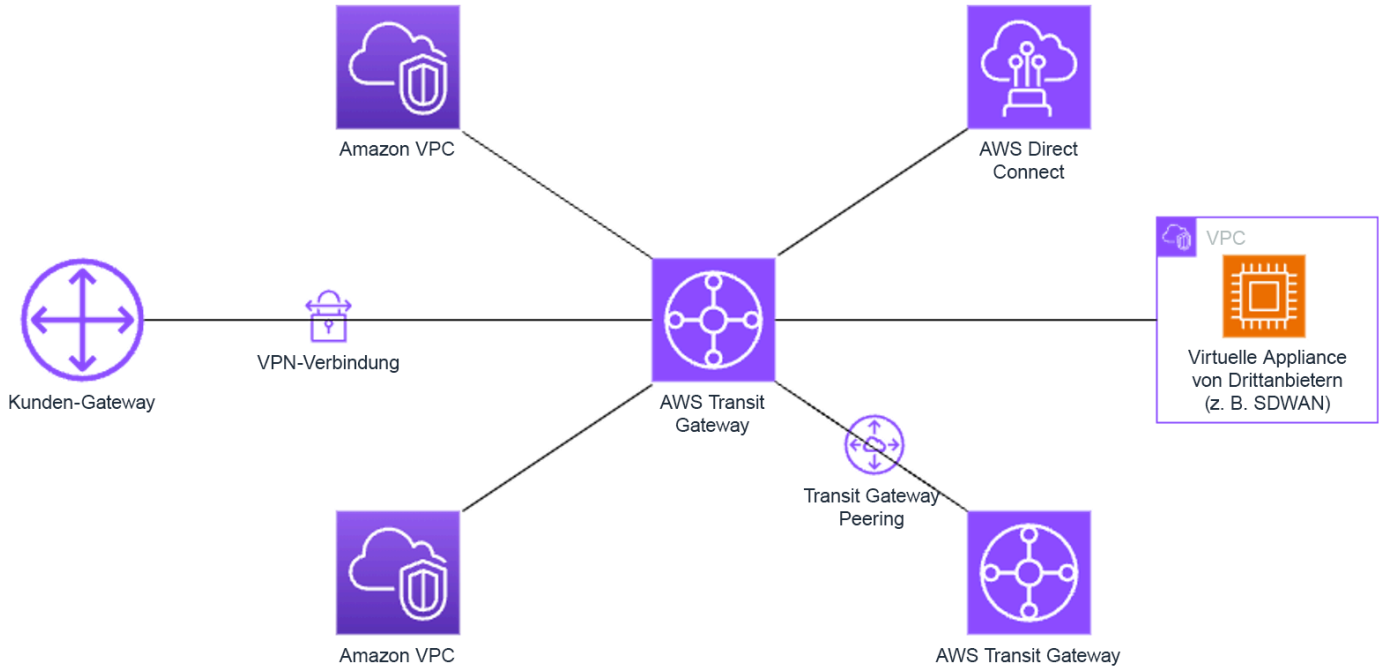
- AWS re:Invent 2018: Erweitertes VPC-Design und neue Funktionen für Amazon VPC (NET303)
- AWS re:Invent 2019: AWS Transit Gateway-Referenzarchitekturen für viele VPCs (NET406-R1)
- AWS re:Invent 2023: AWS – Sind Sie bereit für Neues? Gestaltung von Netzwerken für Wachstum und Flexibilität (NET310)

REL02-BP04 Vorziehen von Hub-and-Spoke-Topologien gegenüber M-zu-N-Netzen

Wenn Sie mehrere private Netzwerke wie Virtual Private Clouds (VPCs) und On-Premises-Netzwerke verbinden, sollten Sie sich für eine Hub-and-Spoke-Topologie statt für eine verflochtene Topologie entscheiden. Im Gegensatz zu verflochtenen Topologien, bei denen jedes Netzwerk direkt mit dem anderen verbunden ist, was die Komplexität und den Verwaltungsaufwand erhöht, zentralisiert die Hub-and-Spoke-Architektur Verbindungen über einen einzigen Hub. Diese Zentralisierung vereinfacht die Netzwerkstruktur und verbessert deren Bedienbarkeit, Skalierbarkeit und Kontrolle.

AWS Transit Gateway ist ein verwalteter, skalierbarer und hochverfügbarer Service, der für den Aufbau von Hub-and-Spoke-Netzwerken auf AWS entwickelt wurde. Er dient als zentraler Knotenpunkt Ihres Netzwerks, der Netzwerksegmentierung, zentralisiertes Routing und die vereinfachte Verbindung zu Cloud- und On-Premises-Umgebungen ermöglicht. Die folgende

Abbildung zeigt, wie Sie Ihre Hub-and-Spoke-Topologie mit AWS Transit Gateway entwickeln können.



Typische Anti-Muster:

- Sie verkomplizieren Routing-Richtlinien in einer Hub-and-Spoke-Architektur zu sehr, was die Netzwerkeffizienz verringert und sowohl die Fehlerbehebung als auch die proaktive Verwaltung erschwert.
- Eine unzureichende routingbasierte Segmentierung innerhalb des Hubs kann zu Sicherheitsschwachstellen führen, die das Netzwerk potenziell unbefugten Zugriffen aussetzen.
- Ohne sorgfältige Optimierung kann der über den Hub geleitete Verkehr zu höheren Datenübertragungskosten führen, insbesondere für den Verkehr, der Availability Zones und Regions passiert. Effektive Verkehrsmanagementstrategien sind für die Kostenkontrolle unerlässlich.

Vorteile der Einführung dieser bewährten Methode: Während die Anzahl der verbundenen Netzwerke zunimmt, wird die Verwaltung und Erweiterung der verflochtenen Konnektivität immer schwieriger. AWS Transit Gateway bietet einen skalierbaren und zuverlässigen verwalteten Hub für den Aufbau und Betrieb Ihrer Hub-and-Spoke-Topologien. Wenn Sie AWS Transit Gateway verwenden,

können Sie Verbindungen herstellen und das Routing des Datenverkehrs über mehrere Netzwerke zentralisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

- Planen Sie Ihr Netzwerk.
- Erstellen Sie Ihr AWS Transit Gateway.
- Fügen Sie Ihre VPCs an.
- Erstellen Sie bei Bedarf VPN-Verbindungen oder Direct Connect-Gateways und verknüpfen Sie sie mit dem Transit Gateway.
- Definieren Sie durch die Konfiguration Ihrer Transit Gateway-Routing-Tabellen, wie der Verkehr zwischen den verbundenen VPCs und anderen Verbindungen weitergeleitet wird.
- Verwenden Sie Amazon CloudWatch, um Konfigurationen zu überwachen und bei Bedarf zur Leistungs- und Kostenoptimierung anzupassen.

Ressourcen

Zugehörige Dokumente:

- Was ist Transit Gateway?
- Erstellen einer skalierbaren und sicheren Multi-VPC-AWS-Netzwerkinfrastruktur
- Aufbau eines globalen Netzwerks mithilfe von regionsübergreifendem AWS Transit Gateway-Peering
- Amazon Virtual Private Cloud-Konnektivitätsoptionen
- APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können
- AWS Marketplace für Netzwerkinfrastruktur

Zugehörige Videos:

- AWS re:Invent 2023 – AWS-Netzwerkgrundlagen
- AWS re:Invent 2023 – Fortschrittliche VPC-Designs und neue Funktionen

REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht

Die IP-Adressbereiche Ihrer VPCs dürfen sich nicht überschneiden, wenn sie per Peering oder über Transit Gateway oder VPN verbunden sind. Vermeiden Sie IP-Adresskonflikte zwischen einer VPC und On-Premises-Umgebungen oder anderen verwendeten Cloud-Anbietern. Sie müssen bei Bedarf auch die Möglichkeit haben, private IP-Adressbereiche zuzuweisen. Ein IP-Adressenverwaltungssystem (IPAM) kann bei der Automatisierung helfen.

Gewünschtes Ergebnis:

- Keine Konflikte mit IP-Adressbereichen zwischen VPCs, On-Premises-Umgebungen oder anderen Cloud-Anbietern.
- Eine angemessene IP-Adressverwaltung ermöglicht eine einfachere Skalierung der Netzwerkinfrastruktur, um wachsenden und sich wandelnden Netzwerkanforderungen gerecht zu werden.

Typische Anti-Muster:

- Verwendung desselben IP-Bereichs in Ihrer VPC wie On-Premises, in Ihrem Unternehmensnetzwerk oder bei anderen Cloud-Anbietern.
- Keine Verfolgung von IP-Bereichen von VPCs, die zur Bereitstellung der Workloads verwendet werden.
- Alleinige Nutzung manueller IP-Adressverwaltungsprozesse wie Tabellenkalkulationen.
- Über- oder Unterdimensionierung von CIDR-Blöcken, was zu einer Verschwendung von IP-Adressen oder zu wenig Adressbereichen für Ihren Workload führt.

Vorteile der Nutzung dieser bewährten Methode: Mit der aktiven Planung des Netzwerks stellten Sie sicher, dass dieselbe IP-Adresse in miteinander verbunden Netzwerken nicht mehrmals vorkommt. So wird verhindert, dass Routing-Probleme in Teilen der Workload auftreten, die die verschiedenen Anwendungen verwenden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Verwenden Sie ein IPAM, z. B. den Amazon VPC IP Address Manager, um Ihre CIDR-Nutzung zu überwachen und zu verwalten. Im AWS Marketplace stehen auch mehrere IPAMs zur Verfügung.

Bewerten Sie die potenzielle Nutzung in AWS, fügen Sie vorhandenen VPCs CIDR-Bereiche hinzu und erstellen Sie neue VPCs, um das geplante Wachstum abzudecken.

Implementierungsschritte

- Ermitteln Sie den aktuellen CIDR-Umfang (z. B. VPCs und Subnetze).
  - Erfassen Sie über die Service API den aktuellen CIDR-Umfang.
  - Verwenden Sie den Amazon VPC IP Address Manager, um Ressourcen zu entdecken.
- Erfassen Sie die aktuelle Subnetzauslastung.
  - Ermitteln Sie über die Service-API die in jeder Region pro VPC vorhandenen Subnetze.
  - Verwenden Sie den Amazon VPC IP Address Manager, um Ressourcen zu entdecken.
- Zeichnen Sie die aktuelle Auslastung auf.
- Prüfen Sie, ob sich IP-Bereiche überschneiden.
- Berechnen Sie die freie Kapazität.
- Identifizieren Sie sich überschneidende IP-Bereiche. Sie können wahlweise zu einem neuen Adressbereich migrieren oder Techniken wie privates NAT-Gateway oder AWS PrivateLink verwenden, wenn Sie die sich überschneidenden Bereiche verbinden müssen.

Ressourcen

Zugehörige bewährte Methoden:

- Schutz von Netzwerken

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können
- AWS Marketplace für Netzwerkinfrastruktur
- Amazon Virtual Private Cloud-Konnektivitätsoptionen – Whitepaper
- Hochverfügbare Netzwerkkonnektivität zwischen mehreren Rechenzentren
- Netzwerke mit sich überschneidenden IP-Bereichen verbinden
- Was ist Amazon VPC?
- Was ist IPAM?

Zugehörige Videos:

- AWS re:Invent 2023 – Fortschrittliche VPC-Designs und neue Funktionen
- AWS re:Invent 2019: AWS Transit Gateway-Referenzarchitekturen für viele VPCs
- AWS re:Invent 2023 – Sind Sie bereit für Neues? Gestaltung von Netzwerken für Wachstum und Flexibilität
- AWS re:Invent 2021 – {Neuer Launch} Verwaltung Ihrer IP-Adressen im großen Maßstab in AWS

Workload-Architektur

Fragen

- REL 3. Wie entwerfen Sie Ihre Workload-Service-Architektur?
- REL 4. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?
- REL 5. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle abgemildert oder bewältigt werden?

REL 3. Wie entwerfen Sie Ihre Workload-Service-Architektur?

Erstellen Sie hoch skalierbare und zuverlässige Workloads mithilfe einer serviceorientierten Architektur (SOA) oder einer Microservices-Architektur. Eine serviceorientierte Architektur (SOA) hat zum Ziel, Softwarekomponenten über Service-Schnittstellen wiederverwendbar zu machen. Die Microservices-Architektur geht noch weiter, um Komponenten kleiner und einfacher zu machen.

Bewährte Methoden

- REL03-BP01 Segmentierung Ihres Workloads
- REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren
- REL03-BP03 Bereitstellen von Serviceverträgen pro API

REL03-BP01 Segmentierung Ihres Workloads

Die Workload-Segmentierung ist wichtig, wenn es um die Festlegung der Resilienzanforderungen Ihrer Anwendung geht. Eine monolithische Architektur sollte vermieden werden, wann immer

möglich. Stattdessen sollten Sie sorgfältig überlegen, welche Anwendungskomponenten in Microservices aufgeteilt werden können. Abhängig von den Anforderungen Ihrer Anwendung könnte es sich im Endergebnis um eine Kombination aus einer serviceorientierten Architektur (SOA) und Microservices handeln, wenn dies möglich ist. Workloads, die zustandslos sein können, können eher als Microservices bereitgestellt werden.

Gewünschtes Ergebnis: Workloads sollten unterstützbar, skalierbar und so lose miteinander verbunden sein wie möglich.

Wiegen Sie bei Entscheidungen zur Segmentierung von Workloads die Vorteile und die Komplexitäten miteinander ab. Was für ein neues Produkt richtig ist, das gerade auf dem Markt eingeführt wird, unterscheidet sich von den Anforderungen eines Workloads, der von Anfang an skalierbar sein muss. Bei einem Faktorwechsel für einen vorhandenen Monolith müssen Sie berücksichtigen, wie gut dieser aufgeteilt und in zustandslose Anwendungen transformiert werden kann. Die Aufteilung von Services in kleinere Teile ermöglicht kleinen, klar definierten Teams, diese weiterzuentwickeln und zu verwalten. Kleinere Services können jedoch Komplexitäten wie eine möglicherweise erhöhte Latenz, ein komplexeres Debugging und einen erhöhten operativen Aufwand einführen.

Typische Anti-Muster:

- Der Microservice Death Star ist eine Situation, in der die einzelnen Komponenten so stark voneinander abhängig werden, dass der Ausfall einer einzigen Komponente einen wesentlich größeren Ausfall bewirkt. Das bedeutet, dass die Komponenten so starr und anfällig wie ein Monolith sind.

Vorteile der Einrichtung dieser Best Practice:

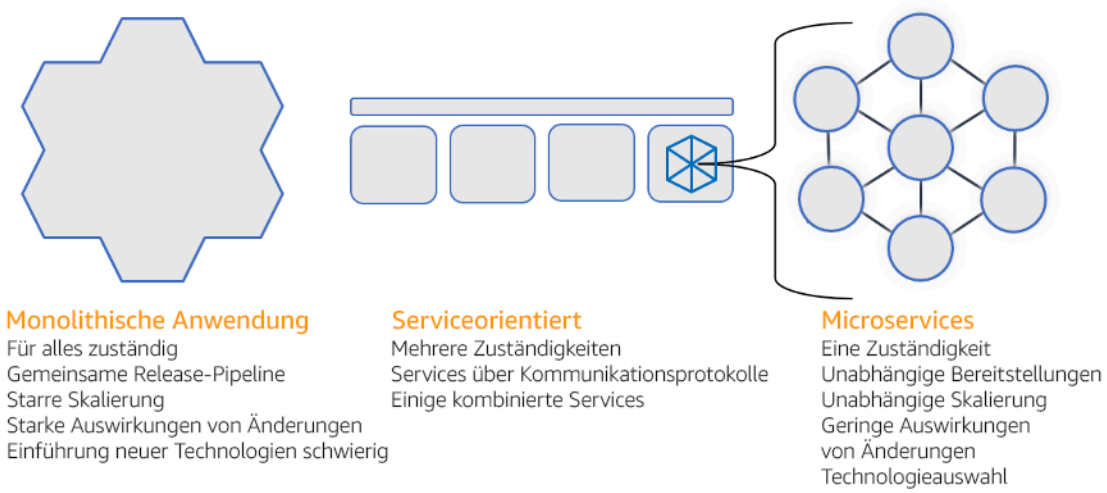
- Spezifischere Segmente führen zu einer größeren Agilität, zu organisatorischer Flexibilität und zu Skalierbarkeit.
- Die Auswirkungen von Service-Unterbrechungen werden reduziert.
- Die einzelnen Komponenten einer Anwendung besitzen möglicherweise unterschiedliche Anforderungen an die Verfügbarkeit, die von einer stärkeren Segmentierung besser unterstützt werden können.
- Die Verantwortlichkeiten der Teams, die den Workload unterstützen, sind klar definiert.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Wählen Sie Ihren Architekturtyp basierend auf der Segmentierung Ihres Workloads aus. Wählen Sie eine serviceorientierte Architektur (SOA) oder eine Microservices-Architektur aus. (In seltenen Fällen ist möglicherweise auch eine monolithische Architektur geeignet.) Auch wenn Sie mit einer monolithischen Architektur beginnen möchten, müssen Sie sicherstellen, dass diese modular ist und zu einer SOA oder zu Microservices weiterentwickeln werden kann, wenn Ihr Produkt aufgrund der zunehmenden Einführung durch Benutzer skaliert wird. SOA und Microservices ermöglichen eine kleinteiligere Segmentierung, die als moderne skalierbare und zuverlässige Architektur bevorzugt wird. Es gibt jedoch auch Nachteile, die besonders bei der Bereitstellung einer Microservice-Architektur berücksichtigt werden sollten.

Aufgrund ihrer verteilten Computing-Architektur kann es schwieriger sein, die Latenzanforderungen von Benutzern zu erfüllen. Außerdem sind das Debugging und die Nachverfolgung von Benutzerinteraktionen komplexer. Zur Lösung dieses Problems können Sie AWS X-Ray verwenden. Ein weiterer Effekt ist die erhöhte operative Komplexität, da die Anzahl der von Ihnen verwalteten Anwendungen zunimmt. In der Folge müssen Sie eine größere Zahl voneinander unabhängiger Komponenten bereitstellen.



Monolithische, serviceorientierte und Microservice-Architekturen

Implementierungsschritte

- Ermitteln Sie die richtige Architektur für den Faktorwechsel oder die Entwicklung Ihrer Anwendung. SOA und Microservices bieten eine jeweils kleinere Segmentierung, die als moderne skalierbare und zuverlässige Architektur bevorzugt wird. SOA kann ein guter Kompromiss für das Erreichen

einer kleineren Segmentierung sein, während die Komplexität von Microservices zum Teil vermieden wird. Weitere Informationen finden Sie in Kompromisse bei Microservices.

- Wenn Ihre Workload für sie zugänglich ist und Ihre Organisation sie unterstützen kann, sollten Sie eine Microservices-Architektur verwenden, um die beste Agilität und Zuverlässigkeit zu erzielen. Weitere Informationen finden Sie in Implementieren von Microservices in AWS.
- Sie sollten das Muster mit der Bezeichnung Strangler Fig („Würgefeige“) verwenden, um einen Faktorwechsel für einen Monolithen durchzuführen, bei dem Sie diesen in kleinere Komponenten aufteilen. Dies umfasst die schrittweise Ersetzung spezifischer Anwendungskomponenten durch neue Anwendungen und Services. AWS Migration Hub Refactor Spaces dient als Ausgangspunkt für den inkrementellen Faktorwechsel. Weitere Informationen finden Sie in Nahtlose Integration ältere On-Premises-Workloads unter Anwendung eines Strangler-Fig-Musters.
- Die Implementierung von Microservices erfordert möglicherweise einen Mechanismus für die Entdeckung von Services, damit diese verteilten Services miteinander kommunizieren können. AWS App Mesh kann mit serviceorientierten Architekturen verwendet werden, um eine zuverlässige Erkennung von Services und den Zugriff auf sie zu unterstützen. AWS Cloud Map kann für die dynamische, DNS-basierte Serviceerkennung verwendet werden.
- Wenn Sie von einem Monolithen zur SOA migrieren, kann Amazon MQ helfen, als Service-Bus die Lücke zu überbrücken, wenn Sie ältere Anwendungen in der Cloud neu entwerfen.
- Im Fall vorhandener Monolithen mit einer einzigen, geteilten Datenbank müssen Sie entscheiden, wie Sie die Daten neu in kleineren Segmenten organisieren. Dabei kann es sich um Geschäftsbereiche, Zugriffsmuster oder Datenstrukturen handeln. An diesem Punkt des Faktorwechsel-Prozesses sollten Sie entscheiden, ob Sie eine relationale oder eine nicht relationale (NoSQL) Datenbank verwenden. Weitere Informationen finden Sie in Von SQL zu NoSQL.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren

Zugehörige Dokumente:

- Amazon API Gateway: Konfigurieren einer REST-API mit OpenAPI
- Was ist eine serviceorientierte Architektur?
- Bounded Context (Begrenzter Kontext) (ein zentrales Muster im domänengesteuerten Design)
- Implementieren von Microservices in AWS
- Kompromisse bei Microservices
- Microservices – eine Definition dieses neuen Architekturbegriffs
- Microservices in AWS
- Was ist AWS App Mesh?

Zugehörige Beispiele:

- Workshop für die iterative App-Modernisierung

Zugehörige Videos:

- Kompetenz mit Microservices in AWS

REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren

Eine serviceorientierte Architektur (SOA) definiert Services mit genau abgegrenzten Funktionen, die von Geschäftsanforderungen definiert werden. Microservices verwenden Domänenmodelle und begrenzten Kontext, um Servicegrenzen entlang der Grenzen des Geschäftskontextes zu ziehen. Die Konzentration auf Geschäftsdomänen und Funktionen hilft Teams dabei, unabhängige Zuverlässigkeitsanforderungen für ihre Services zu definieren. Begrenzte Kontexte isolieren und kapseln die Geschäftslogik, sodass Teams besser überlegen können, wie mit Fehlern umzugehen ist.

Gewünschtes Ergebnis: Ingenieure und geschäftliche Interessenvertreter definieren gemeinsam begrenzte Kontexte und verwenden sie, um Systeme als Services zu entwerfen, die bestimmte Geschäftsfunktionen erfüllen. Diese Teams verwenden etablierte Praktiken wie Event Storming, um Anforderungen zu definieren. Neue Anwendungen sind als Services mit klar definierten Grenzen und losen Verkopplungen definiert. Bestehende Monolithe werden in begrenzte Kontexte zerlegt und Systemdesigns bewegen sich in Richtung SOA- oder Microservice-Architekturen. Bei der Refaktorisierung von Monolithen kommen etablierte Ansätze wie Bubble-Kontexte und Monolith-Zerlegung zur Anwendung.

Domänenorientierte Services werden als ein oder mehrere Prozesse ausgeführt, die keinen gemeinsamen Zustand haben. Sie reagieren selbstständig auf Nachfrageschwankungen und behandeln Störszenarien anhand domänenspezifischer Anforderungen.

Typische Anti-Muster:

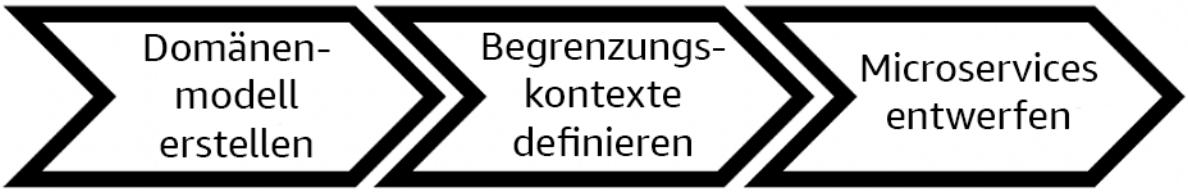
- Teams werden für bestimmte technische Bereiche wie UI und UX, Middleware oder Datenbank gebildet, anstatt für bestimmte Geschäftsdomänen.
- Anwendungen erstrecken sich über die Zuständigkeiten der einzelnen Bereiche. Services, die sich über begrenzte Kontexte erstrecken, können schwieriger zu verwalten sein, erfordern einen größeren Testaufwand und erfordern die Teilnahme mehrerer Domänenteams an Softwareupdates.
- Domänenabhängigkeiten wie Domain-Entity-Bibliotheken werden von allen Services gemeinsam genutzt, sodass Änderungen für eine Servicedomäne Änderungen an anderen Service-Domains erfordern.
- Serviceverträge und Geschäftslogik formulieren Entities nicht in einer gemeinsamen und konsistenten Domänensprache, was zu Übersetzungsebenen führt, die Systeme komplizieren und den Debugging-Aufwand erhöhen.

Vorteile der Nutzung dieser bewährten Methode: Anwendungen sind als unabhängige Services konzipiert, die durch Geschäftsdomänen begrenzt sind und eine gemeinsame Geschäftssprache verwenden. Services sind unabhängig voneinander testbar und einsetzbar. Services erfüllen die domänenspezifischen Resilienzanforderungen für die implementierte Domäne.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Domain-driven Decision (DDD, Domänengesteuerte Entscheidung) ist der grundlegende Ansatz für das Entwerfen und Entwickeln von Software rund um Geschäftsdomänen. Bei der Entwicklung von Services, die sich auf Geschäftsdomänen konzentrieren, ist es hilfreich, mit einem vorhandenen Framework zu arbeiten. Wenn Sie mit bestehenden monolithischen Anwendungen arbeiten, können Sie die Vorteile von Zerlegungsmustern nutzen, die etablierte Techniken zur Modernisierung von Anwendungen in Services bereitstellen.



Domänengesteuerte Entscheidung

Implementierungsschritte

- Teams können Event-Storming-Workshops veranstalten, um rasch Ereignisse, Befehle, Mengen und Domänen in einem unkomplizierten Notizformat zu sammeln.
- Sobald Domain-Entities und -Funktionen in einem Domänenkontext gebildet wurden, können Sie Ihre Domäne mithilfe eines begrenzten Kontextsweiter in kleinere Modelle unterteilt, wobei Entities mit ähnlichen Funktionen und Attributen in Gruppen sortiert werden. Wenn das Modell in Kontexte unterteilt ist, entsteht eine Vorlage für die Begrenzung von Microservices.
  - Für die Website Amazon.com können Entities beispielsweise Pakete, Zustellung, Zeitplan, Preise, Rabatte und Währung enthalten.
  - Paket, Zustellung und Zeitplan werden dem Versandkontext zugeordnet, während Preis, Rabatt und Währung dem Preiskontext zugeordnet sind.
- Zerlegung von Monolithen in Microservices skizziert Muster für das Refactoring von Microservices. Die Verwendung von Mustern für die Unterteilung nach Geschäftsfähigkeit, Subdomäne oder Transaktion passt gut zu domänengesteuerten Ansätzen.
- Taktische Techniken wie der Bubble-Kontext ermöglichen es Ihnen, DDD in bestehenden oder älteren Anwendungen einzuführen, ohne dass Sie im Voraus Änderungen vornehmen und sich voll und ganz auf DDD verlassen müssen. Bei einem Bubble-Kontext-Ansatz wird mithilfe von Service-Mapping und -koordination ein kleiner begrenzter Kontext oder eine Ebene zur Korruptionsbekämpfungerstellt, die das neu definierte Domänenmodell vor äußeren Einflüssen schützt.

Nachdem die Teams eine Domänenanalyse durchgeführt und Entities und Serviceverträge definiert haben, können sie AWS-Services nutzen, um ihr domänengesteuertes Design als Cloud-basierte Services zu implementieren.

- Beginnen Sie Ihre Entwicklung, indem Sie Tests definieren, die die Geschäftsregeln Ihrer Domäne anwenden. Test-driven Development (TDD, Testgetriebene Entwicklung) und Behavior-driven Development (BDD, verhaltensgetriebene Entwicklung) helfen Teams dabei, die Services auf die Lösung von Geschäftsproblemen zu konzentrieren.
- Wählen Sie die AWS-Services, die den Anforderungen Ihrer Geschäftsdomänen und Ihrer Microservice-Architekturam besten entsprechen:
  - AWS Serverless ermöglicht es Ihrem Team, sich auf eine bestimmte Domänenlogik zu konzentrieren, anstatt Server und Infrastruktur zu verwalten.
  - Container in AWS vereinfachen die Verwaltung Ihrer Infrastruktur, sodass Sie sich auf Ihre Domänenanforderungen konzentrieren können.
  - Speziell entwickelte Datenbanken helfen Ihnen dabei, Ihre Domänenanforderungen dem am besten geeigneten Datenbanktyp zuzuordnen.
- Hexagonale Architekturen auf AWS skizzieren ein Framework zur Integration von Geschäftslogik in Services. Dabei wird rückwärts von der Geschäftsdomäne aus gearbeitet, um funktionale Anforderungen zu erfüllen und dann Integrationsadapter zu implementieren. Muster, die Schnittstellendetails von der Geschäftslogik mit AWS-Services trennen, helfen Teams, sich auf die Funktionalität der Domäne zu konzentrieren und die Softwarequalität zu verbessern.

Ressourcen

Zugehörige bewährte Methoden:

- REL03-BP01 Segmentierung Ihres Workloads
- REL03-BP03 Bereitstellen von Serviceverträgen pro API

Zugehörige Dokumente:

- AWS Microservices
- Implementieren von Microservices in AWS
- How to break a Monolith into Microservices (Aufschlüsseln eines Monolithen in Microservices)
- Getting Started with DDD when Surrounded by Legacy Systems (Erste Schritte mit DDD, wenn die Umgebung aus Legacy-Systemen besteht)
- Domain-Driven Design: Tackling Complexity in the Heart of Software (Domänengesteuertes Design: Umgang mit der Komplexität im Herzen der Software)
- Hexagonale Architekturen auf AWS

- Zerlegung von Monolithen in Microservices
- Event Storming
- Nachrichten zwischen begrenzten Kontexten
- Microservices
- Testgetriebene Entwicklung
- Verhaltensgetriebene Entwicklung

Zugehörige Beispiele:

- Workshop „Enterprise Cloud Native“
- Designing Cloud Native Microservices on AWS (from DDD/EventStormingWorkshop) (Entwerfen Cloud-nativer Microservices in AWS (aus DDD/EventStormingWorkshop))

Zugehörige Tools:

- AWS Cloud-Datenbanken
- Serverless auf AWS
- Container in AWS

REL03-BP03 Bereitstellen von Serviceverträgen pro API

Serviceverträge sind dokumentierte Vereinbarungen zwischen API-Herstellern und Verbrauchern, die in einer maschinenlesbaren API-Definition festgehalten sind. Eine Vertragsversionsverwaltungsstrategie ermöglicht es Verbrauchern, die vorhandene API weiter zu verwenden und ihre Anwendungen auf eine neuere API zu migrieren, wenn sie bereit sind. Die Bereitstellung durch den Produzenten kann jederzeit erfolgen, solange der Vertrag eingehalten wird. Die Serviceteams können den Technologie-Stack ihrer Wahl verwenden, um den API-Vertrag zu erfüllen.

Gewünschtes Ergebnis:

Typische Anti-Muster: Anwendungen, die mit serviceorientierten Architekturen oder Microservice-Architekturen erstellt wurden, können unabhängig voneinander arbeiten und verfügen gleichzeitig über eine integrierte Laufzeitabhängigkeit. Änderungen, die für einen API-Verbraucher oder -Hersteller bereitgestellt werden, beeinträchtigen die Stabilität des Gesamtsystems nicht, wenn

beide Seiten einen gemeinsamen API-Vertrag einhalten. Komponenten, die über Service-APIs kommunizieren, können unabhängige funktionale Releases, Upgrades von Laufzeitabhängigkeiten oder ein Failover auf eine Notfallwiederherstellung (DR) ausführen, ohne dass sich dies gegenseitig beeinträchtigt. Darüber hinaus können spezialisierte Services unabhängig voneinander skaliert werden und können dabei den Ressourcenbedarf absorbieren, ohne dass andere Services ebenfalls skaliert werden müssen.

- Erstellung von Service-APIs ohne stark typisierte Schemata. Dies führt zu APIs, die nicht zum Generieren von API-Bindungen und Payloads verwendet werden können, die nicht programmgesteuert validiert werden können.
- Keine Versionsverwaltungsstrategie, weshalb API-Verbraucher dazu gezwungen sind, Updates zu installieren, Releases einzuspielen oder eine Notfallwiederherstellung durchzuführen, wenn sich Serviceverträge weiterentwickeln.
- Fehlermeldungen, die Details der zugrundeliegenden Service-Implementierung preisgeben, anstatt Integrationsfehler im Kontext und in der Sprache der Domäne zu beschreiben.
- Keine Verwendung von API-Verträgen zur Entwicklung von Testfällen und zur Simulation von API-Implementierungen, um unabhängige Tests von Servicekomponenten zu ermöglichen.

Vorteile der Nutzung dieser bewährten Methode: Verteilte Systeme, die aus Komponenten bestehen, die über API-Serviceverträge kommunizieren, können die Zuverlässigkeit verbessern. Entwickler können potenzielle Probleme schon früh im Entwicklungsprozess erkennen, indem sie während der Kompilierung eine Typprüfung durchführen, um sicherzustellen, dass Anfragen und Antworten dem API-Vertrag entsprechen und die erforderlichen Felder vorhanden sind. API-Verträge bieten eine übersichtliche, selbstdokumentierende Schnittstelle für APIs und sorgen für eine bessere Interoperabilität zwischen verschiedenen Systemen und Programmiersprachen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Sobald Sie Geschäftsbereiche identifiziert und Ihre Workload-Segmentierung festgelegt haben, können Sie Ihre Service-APIs entwickeln. Definieren Sie zunächst maschinenlesbare Serviceverträge für APIs und implementieren Sie dann eine Strategie zur API-Versionsverwaltung. Wenn Sie bereit sind, Services über gängige Protokolle wie REST, GraphQL oder asynchrone Ereignisse zu implementieren, können Sie AWS-Services in Ihre Architektur einbinden, um Ihre Komponenten mit stark typisierten API-Verträgen zu integrieren.

AWS-Services für API-Serviceverträge

Implementieren Sie AWS-Services wie Amazon API Gateway, AWS AppSyncund Amazon EventBridge in Ihre Architektur, um API-Serviceverträge in Ihrer Anwendung zu verwenden. Amazon API Gateway hilft Ihnen bei der direkten Integration in native AWS-Services und andere Webservices. API Gateway unterstützt die OpenAPI-Spezifikation sowie die Versionsverwaltung. AWS AppSync ist ein verwalteter GraphQL -Endpunkt, den Sie konfigurieren, indem Sie ein GraphQL-Schema definieren, um eine Serviceschnittstelle für Abfragen, Mutationen und Abonnements festzulegen. Amazon EventBridge verwendet Ereignisschemata, um Ereignisse zu definieren und Codebindungen für Ihre Ereignisse zu generieren.

Implementierungsschritte

- Definieren Sie zunächst einen Vertrag für Ihre API. In einem Vertrag werden die Funktionen einer API festgehalten und stark typisierte Datenobjekte und Felder für die API-Eingabe und -Ausgabe definiert.
- Wenn Sie APIs in API Gateway konfigurieren, können Sie OpenAPI-Spezifikationen für Ihre Endpunkte importieren und exportieren.
  - Eine OpenAPI-Definition zu importieren, vereinfacht die Erstellung Ihrer API und kann in AWS-Infrastrukturen wie AWS Serverless Application Model und AWS Cloud Development Kit (AWS CDK) integriert werden.
  - Eine API-Definition zu exportieren, vereinfacht die Integration in API-Testtools und bietet Servicekunden eine Integrationsspezifikation.
- Definieren und verwalten Sie GraphQL-APIs mit AWS AppSync, indem Sie eine GraphQL-Schema-Datei definieren, um Ihre Vertragsschnittstelle zu generieren und die Interaktion mit komplexen REST-Modellen, mehreren Datenbanktabellen oder Legacy-Services zu vereinfachen.
- AWS Amplify -Projekte, die in AWS AppSync integriert sind, generieren stark typisierte JavaScript-Abfragedateien, die Sie sowohl in Ihrer Anwendung als auch in einer AWS AppSync-GraphQL-Client-Bibliothek für Amazon DynamoDB -Tabellen verwenden können.
- Wenn Sie Serviceereignisse aus Amazon EventBridge verarbeiten, befolgen diese Ereignisse Schemata, die bereits in der Schemaregistrierung existieren oder die Sie mit der OpenAPI-Spezifikation definieren. Mit einem in der Registrierung definierten Schema können Sie auch Client-Bindungen aus dem Schemavertrag generieren, um Ihren Code in Ereignisse zu integrieren.
- API erweitern oder versionieren Die Erweiterung einer API ist eine einfachere Option, wenn Felder hinzugefügt werden, die mit optionalen Feldern oder Standardwerten für Pflichtfelder konfiguriert werden können.
  - JSON-basierte Verträge für Protokolle wie REST und GraphQL können sich gut für eine Vertragserweiterung eignen.

  - XML-basierte Verträge für Protokolle wie SOAP sollten mit Service-Verbrauchern getestet werden, um festzustellen, ob eine Vertragserweiterung durchführbar ist.
- Erwägen Sie bei der Versionsverwaltung einer API die Implementierung einer Proxy-Versionsverwaltung, bei der eine Fassade zur Unterstützung von Versionen verwendet wird, sodass die Logik in einer einzigen Codebasis verwaltet werden kann.
  - Mit API Gateway können Sie Anfrage- und von Antwortzuordnungen nutzen, um Vertragsänderungen einfacher zu übernehmen. Hierzu wird eine Fassade eingerichtet, die Standardwerte für neue Felder bereitstellt oder entfernte Felder aus einer Anfrage oder Antwort herausnimmt. Mit diesem Ansatz kann der zugrunde liegende Service mit einer einzelnen Codebasis betrieben werden.

Ressourcen

Zugehörige bewährte Methoden:

- REL03-BP01 Segmentierung Ihres Workloads
- REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren
- REL04-BP02 Implementieren lose gekoppelter Abhängigkeiten
- REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen
- REL05-BP05 Festlegen von Client-Zeitüberschreitungen

Zugehörige Dokumente:

- Was ist eine API (Anwendungsprogrammierschnittstelle)?
- Implementieren von Microservices in AWS
- Kompromisse bei Microservices
- Microservices – eine Definition dieses neuen Architekturbegriffs
- Microservices in AWS
- Arbeiten mit API Gateway-Erweiterungen für OpenAPI
- OpenAPI-Spezifikation
- GraphQL: Schemata und Typen
- Amazon EventBridge-Codebindungen

Zugehörige Beispiele:

- Amazon API Gateway: Konfigurieren einer REST-API mit OpenAPI
- Amazon API Gateway zu Amazon DynamoDB CRUD-Anwendung mit OpenAPI
- Moderne Anwendungsintegrationsmuster in einem serverlosen Zeitalter: API Gateway-Serviceintegration
- Implementieren einer Header-basierten API Gateway-Versionsverwaltung mit Amazon CloudFront
- AWS AppSync: Erstellen einer Client-Anwendung

Zugehörige Videos:

- Verwenden von OpenAPI in AWS SAM zur Verwaltung von API Gateway

Zugehörige Tools:

- Amazon API Gateway
- AWS AppSync
- Amazon EventBridge

REL 4. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?

Verteilte Systeme nutzen Kommunikationsnetzwerke, um Komponenten wie Server oder Services miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder höherer Latenz in diesen Netzwerken zuverlässig ausgeführt werden. Komponenten des verteilten Systems müssen so funktionieren, dass sie keine negativen Auswirkungen auf andere Komponenten oder die Workload haben. Diese bewährten Methoden verhindern Ausfälle und verbessern die mittlere Zeit zwischen Ausfällen (MTBF).

Bewährte Methoden

- REL04-BP01 Bestimmen Sie, von welcher Art von verteilten Systemen Sie abhängig sind
- REL04-BP02 Implementieren lose gekoppelter Abhängigkeiten
- REL04-BP03 Konstante Ausführung
- REL04-BP04 Festlegen aller Reaktionen als idempotent

REL04-BP01 Bestimmen Sie, von welcher Art von verteilten Systemen Sie abhängig sind

Verteilte Systeme verwenden die synchrone, asynchrone oder Stapelverarbeitung. Synchrone Systeme müssen Anfragen so schnell wie möglich verarbeiten und miteinander kommunizieren, indem sie synchrone Anfrage- und Antwortaufrufe mithilfe von HTTP/S-, REST- oder RPC-Protokollen (Remote Procedure Call) durchführen. Asynchrone Systeme kommunizieren miteinander, indem sie Daten asynchron über einen Zwischenservice austauschen, ohne einzelne Systeme zu koppeln. Systeme mit Stapelverarbeitung empfangen eine große Menge an Eingabedaten, führen automatisierte Datenprozesse ohne menschliches Eingreifen aus und generieren Ausgabedaten.

Gewünschtes Ergebnis: Entwerfen Sie einen Workload, der effektiv mit synchronen, asynchronen und Batch-Abhängigkeiten interagiert.

Typische Anti-Muster:

- Der Workload wartet auf unbestimmte Zeit auf eine Antwort von seinen Abhängigkeiten, was dazu führen kann, dass Workload-Clients das Zeitlimit überschreiten und nicht wissen, ob ihre Anfrage eingegangen ist.
- Der Workload verwendet eine Kette von abhängigen Systemen, die sich gegenseitig synchron aufrufen. Der Erfolg der gesamten Kette hängt davon ab, dass jedes System verfügbar ist und Anfragen erfolgreich verarbeitet, was zu instabilem Verhalten und eingeschränkter Gesamtverfügbarkeit führen kann.
- Der Workload kommuniziert asynchron mit seinen Abhängigkeiten und stützt sich auf das Konzept der garantierten einmaligen Zustellung von Nachrichten, obwohl es oft immer noch möglich ist, doppelte Nachrichten zu empfangen.
- Der Workload verwendet keine geeigneten Tools zur Batchplanung und ermöglicht die gleichzeitige Ausführung desselben Batchjobs.

Vorteile der Einführung dieser bewährten Methode: Es ist üblich, dass ein bestimmter Workload einen oder mehrere Kommunikationsstile der synchronen, asynchronen und Stapelverarbeitung implementiert. Diese bewährte Methode hilft Ihnen dabei, die verschiedenen Kompromisse zu identifizieren, die mit den einzelnen Kommunikationsstilen verbunden sind, damit Ihr Workload Störungen in allen Abhängigkeiten tolerieren kann.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Die folgenden Abschnitte enthalten sowohl allgemeine als auch spezifische Implementierungshinweise für jede Art von Abhängigkeit.

Allgemeine Orientierungshilfe

- Stellen Sie sicher, dass die Leistungs- und Zuverlässigkeits-Servicelevel-Ziele (SLOs), die Ihre Abhängigkeiten bieten, den Leistungs- und Zuverlässigkeitsanforderungen Ihres Workloads entsprechen.
- Verwenden Sie AWS Observability Services, um Reaktionszeiten und Fehlerraten zu überwachen und so sicherzustellen, dass Ihre Abhängigkeit den von Ihrem Workload benötigten Service bietet.
- Identifizieren Sie die potenziellen Herausforderungen, mit denen Ihr Workload bei der Kommunikation mit seinen Abhängigkeiten konfrontiert sein könnte. Verteilte Systeme sind mit einer Vielzahl von Herausforderungen verbunden, die die architektonische Komplexität, den Betriebsaufwand und die Kosten erhöhen können. Zu den häufigsten Herausforderungen gehören Latenz, Netzwerkunterbrechungen, Datenverlust, Skalierung und Verzögerungen bei der Datenreplikation.
- Implementieren Sie eine robuste Fehlerbehandlung und -protokollierung, um Probleme zu beheben, wenn es in Ihrer Abhängigkeit zu Problemen kommt.

Synchrone Abhängigkeit

Bei synchroner Kommunikation sendet Ihr Workload eine Anfrage an seine Abhängigkeit und blockiert den Vorgang, der auf eine Antwort wartet. Wenn ihre Abhängigkeit die Anfrage erhält, versucht sie, sie so schnell wie möglich zu bearbeiten, und sendet eine Antwort zurück an den Workload. Eine große Herausforderung bei synchroner Kommunikation besteht darin, dass sie zu einer zeitlichen Kopplung führt, was erfordert, dass der Workload und dessen Abhängigkeiten gleichzeitig verfügbar sind. Beachten Sie die folgenden Hinweise, wenn der Workload synchron mit seinen Abhängigkeiten kommunizieren muss:

- Der Workload sollte sich nicht auf mehrere synchrone Abhängigkeiten stützen, um eine einzelne Funktion auszuführen. Diese Kette von Abhängigkeiten erhöht die allgemeine Instabilität, da alle Abhängigkeiten im Pfad verfügbar sein müssen, damit die Anfrage erfolgreich abgeschlossen werden kann.
- Wenn eine Abhängigkeit fehlerhaft oder nicht verfügbar ist, bestimmen Sie Ihre Strategie zur Fehlerbehandlung und versuchen Sie es erneut. Vermeiden Sie bimodales Verhalten. Bimodales

Verhalten liegt vor, wenn sich der Workload im Normalmodus und im Fehlermodus unterschiedlich verhält. Weitere Informationen zu bimodalem Verhalten finden Sie unter REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität.

- Denken Sie daran, dass es besser ist, schnell zu scheitern, als Ihren Workload warten zu lassen. Im AWS Lambda Entwicklerhandbuch wird beispielsweise beschrieben, wie Wiederholungen und Fehlschläge beim Aufrufen von Lambda-Funktionen behandelt werden.
- Legen Sie Timeouts fest, wenn Ihr Workload seine Abhängigkeit aufruft. Dadurch wird vermieden, zu lange oder unbegrenzt auf eine Antwort zu warten. Eine hilfreiche Diskussion zu diesem Problem finden Sie unter Optimieren der AWS Java SDK HTTP-Anforderungseinstellungen für latenzfähige Amazon DynamoDB-Anwendungen.
- Reduzieren Sie die Anzahl der Aufrufe von Ihrem Workload an seine Abhängigkeit, um eine einzelne Anfrage zu erfüllen. Durch zahlreiche Aufrufe erhöht sich die Kopplung und Latenz.

Asynchrone Abhängigkeit

Um den Workload zeitlich von dessen Abhängigkeiten zu entkoppeln, sollte die Kommunikation asynchron erfolgen. Bei einem asynchronen Ansatz kann der Workload mit jeder anderen Verarbeitung fortfahren, ohne auf eine Antwort der Abhängigkeit oder Kette von Abhängigkeiten warten zu müssen.

Beachten Sie die folgenden Hinweise, wenn der Workload asynchron mit seiner Abhängigkeit kommunizieren muss:

- Entscheiden Sie je nach Anwendungsfall und Anforderungen, ob Sie Messaging oder Ereignis-Streaming verwenden möchten. Beim Messaging kann der Workload mit seiner Abhängigkeit kommunizieren, indem er Nachrichten über einen Message Broker sendet und empfängt. Beim Ereignis-Streaming können der Workload und seine Abhängigkeiten einen Streaming-Dienst verwenden, um Ereignisse zu veröffentlichen und zu abonnieren, die als kontinuierliche Datenströme bereitgestellt werden und so schnell wie möglich verarbeitet werden müssen.

- Messaging und Ereignis-Streaming behandeln Nachrichten unterschiedlich, sodass Sie basierend auf den folgenden Faktoren Abwägungsentscheidungen treffen müssen:
  - Nachrichtenpriorität: Message Broker können Nachrichten mit hoher Priorität vor normalen Nachrichten verarbeiten. Beim Ereignis-Streaming haben alle Nachrichten dieselbe Priorität.

  - Nachrichtenverbrauch: Message Broker stellen sicher, dass die Verbraucher die Nachricht erhalten. Ereignis-Streaming-Verbraucher müssen den Überblick über die zuletzt gelesene Nachricht behalten.
  - Nachrichtenreihenfolge: Beim Messaging ist der Empfang von Nachrichten in der genauen Reihenfolge, in der sie gesendet wurden, nicht garantiert, es sei denn, Sie verwenden einen FIFO-Ansatz (First-in-First-Out). Beim Ereignis-Streaming wird immer die Reihenfolge beibehalten, in der die Daten erzeugt wurden.
  - Löschen von Nachrichten: Beim Messaging muss der Verbraucher die Nachricht nach der Verarbeitung löschen. Der Ereignis-Streaming-Dienst hängt die Nachricht an einen Stream an und verbleibt dort, bis die Aufbewahrungsfrist der Nachricht abläuft. Diese Löschrichtlinie macht das Ereignis-Streaming für die Wiedergabe von Nachrichten geeignet.
- Definieren Sie, wie Ihr Workload weiß, wann seine Abhängigkeit seine Arbeit beendet hat. Wenn der Workload beispielsweise asynchron eine Lambda-Funktion aufruft, stellt Lambda das Ereignis in eine Warteschlange und gibt eine Erfolgsantwort ohne zusätzliche Informationen zurück. Nach Abschluss der Verarbeitung kann die Lambda-Funktion das Ergebnis an ein Ziel senden, das je nach Erfolg oder Misserfolg konfiguriert werden kann.
- Entwickeln Sie Ihren Workload so, dass er doppelte Nachrichten verarbeiten kann, indem Sie Idempotenz nutzen. Idempotenz bedeutet, dass sich die Ergebnisse des Workloads nicht ändern, auch wenn der Workload mehrmals für dieselbe Nachricht generiert wird. Es ist wichtig, darauf hinzuweisen, dass Messaging- oder Streaming-Dienste eine Nachricht erneut übermitteln, wenn ein Netzwerkausfall auftritt oder wenn keine Bestätigung eingegangen ist.
- Wenn Ihr Workload keine Antwort von seiner Abhängigkeit erhält, muss er die Anfrage erneut einreichen. Erwägen Sie, die Anzahl der Wiederholungsversuche zu begrenzen, um die CPU-, Arbeitsspeicher- und Netzwerkressourcen des Workloads für die Bearbeitung anderer Anfragen zu schonen. Die AWS Lambda-Dokumentation zeigt, wie Fehler beim asynchronen Aufruf behandelt werden.
- Nutzen Sie geeignete Beobachtbarkeits-, Debugging- und Tracing-Tools, um die asynchrone Kommunikation des Workloads mit seinen Abhängigkeiten zu verwalten und zu betreiben. Sie können Amazon CloudWatch verwenden, um Nachrichten- und Ereignis-Streaming-Dienste zu überwachen. Sie können auch Ihren Workload mit AWS X-Ray instrumentieren, um schnell Erkenntnisse zur Problembehebung zu gewinnen.

Batch-Abhängigkeit

Batch-Systeme nehmen Eingabedaten auf, initiieren eine Reihe von Aufgaben, um sie zu verarbeiten, und erzeugen einige Ausgabedaten, ohne dass manuelles Eingreifen erforderlich ist. Je nach Datengröße können Aufgaben in wenigen Minuten oder in einigen Fällen sogar in mehreren Tagen ausgeführt werden. Beachten Sie die folgenden Hinweise, wenn der Workload mit seiner Batch-Abhängigkeit kommuniziert:

- Definieren Sie das Zeitfenster, in dem der Workload den Batchjob ausführen soll. Der Workload kann ein Wiederholungsmuster einrichten, um ein Batchsystem aufzurufen, beispielsweise jede Stunde oder am Ende eines jeden Monats.
- Ermitteln Sie den Ort der Dateneingabe und der verarbeiteten Datenausgabe. Wählen Sie einen Speicherservice wie Amazon Simple Storage Services (Amazon S3), Amazon Elastic File System (Amazon EFS) und Amazon FSx for Lustre, der es Ihrem Workload ermöglicht, Dateien in großem Umfang zu lesen und zu schreiben.
- Wenn Ihr Workload mehrere Batchjobs aufrufen muss, können Sie AWS Step Functions nutzen, um die Orchestrierung von Batchjobs, die in AWS oder On-Premises ausgeführt werden, zu vereinfachen. In diesem Beispielprojekt wird die Orchestrierung von Batchjobs mithilfe von Step Functions, AWS Batch und Lambda demonstriert.
- Überwachen Sie Batchjobs, um nach Auffälligkeiten zu suchen, z. B. wenn die Ausführung eines Jobs länger dauert, als sie sollte. Sie könnten Tools wie CloudWatch Container Insights verwenden, um AWS Batch-Umgebungen und Jobs zu überwachen. In diesem Fall würde Ihr Workload den Beginn des nächsten Jobs unterbrechen und die zuständigen Mitarbeiter über die Ausnahme informieren.

Ressourcen

Zugehörige Dokumente:

- AWS Cloud-Betrieb: Überwachung und Beobachtbarkeit
- Die Amazon's Builder Library: Herausforderungen für verteilte Systeme
- REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität
- AWS Lambda-Entwicklerhandbuch: Fehlerbehandlung und automatische Wiederholungsversuche in AWS Lambda
- Optimieren der AWS Java-SDK-HTTP-Anforderungseinstellungen für latenzfähige Amazon DynamoDB-Anwendungen
- AWS-Messaging

- Was sind Streaming-Daten?
- AWS Lambda Entwicklerhandbuch: Asynchroner Aufruf
- Amazon Simple Queue Service FAQ: FIFO-Warteschlangen
- Amazon Kinesis Data Streams Entwicklerhandbuch: Umgang mit doppelten Datensätzen
- Amazon Simple Queue Service Entwicklerhandbuch: Verfügbare CloudWatch-Metriken für Amazon SQS
- Amazon Kinesis Data Streams Entwicklerhandbuch: Überwachung des Amazon Kinesis Data Streams-Service mit Amazon CloudWatch
- AWS X-Ray Entwicklerhandbuch: AWS X-Ray-Konzepte
- AWS-Beispiele auf GitHub: AWS Step Functions Complex Orchestrator App
- AWS Batch Benutzerhandbuch: AWS Batch CloudWatch Container Insights

Zugehörige Videos:

- AWS Summit SF 2022 – Full-Stack-Beobachtbarkeit und -Überwachung von Anwendungen mit AWS (COP310)

Zugehörige Tools:

- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS X-Ray
- Amazon Simple Storage Services (Amazon S3)
- Amazon Elastic File System (Amazon EFS)
- Amazon FSx for Lustre
- AWS Step Functions
- AWS Batch

REL04-BP02 Implementieren lose gekoppelter Abhängigkeiten

Abhängigkeiten etwa zwischen Warteschlangensystemen, Streaming-Systemen, Workflows und Load Balancern sind lose gekoppelt. Eine lose Verkoppelung hilft, das Verhalten einer Komponente von anderen Komponenten zu isolieren, die von ihr abhängig sind. Dies verbessert Resilienz und Agilität.

In eng gekoppelten Systemen können Änderungen an einer Komponente Änderungen an anderen Komponenten erforderlich machen, die von ihr abhängen, was die Leistung aller Komponenten beeinträchtigt. Die lose Verkoppelung unterbricht diese Abhängigkeit, sodass abhängige Komponenten nur die versionierte und veröffentlichte Schnittstelle kennen müssen. Die Implementierung einer losen Kopplung zwischen Abhängigkeiten isoliert einen Ausfall. So wird verhindert, dass er sich auf andere Komponenten auswirkt.

Die lose Verkoppelung ermöglicht Ihnen, einer Komponente zusätzlichen Code oder Features hinzuzufügen und gleichzeitig das Risiko für Komponenten zu minimieren, die von ihr abhängig sind. Sie ermöglicht auch eine granulare Ausfallsicherheit auf Komponentenebene, bei der Sie die zugrunde liegende Implementierung der Abhängigkeit aufskalieren oder sogar ändern können.

Um die Ausfallsicherheit durch lose Kopplung weiter zu verbessern, legen Sie Komponenten-Interaktionen nach Möglichkeit als asynchron fest. Dieses Modell eignet sich für jede Interaktion, bei der keine sofortige Antwort benötigt wird, sondern die Bestätigung ausreicht, dass eine Anfrage registriert wurde. Es umfasst eine Komponente, die Ereignisse generiert, und eine andere Komponente, die sie konsumiert. Die beiden Komponenten lassen sich nicht durch direkte Punkt-zu-Punkt-Interaktion integrieren, sondern in der Regel über eine temporäre, robuste Speicherschicht, z. B. eine Amazon SQS-Warteschlange oder eine Streaming-Datenplattform wie Amazon Kinesis oder AWS Step Functions.

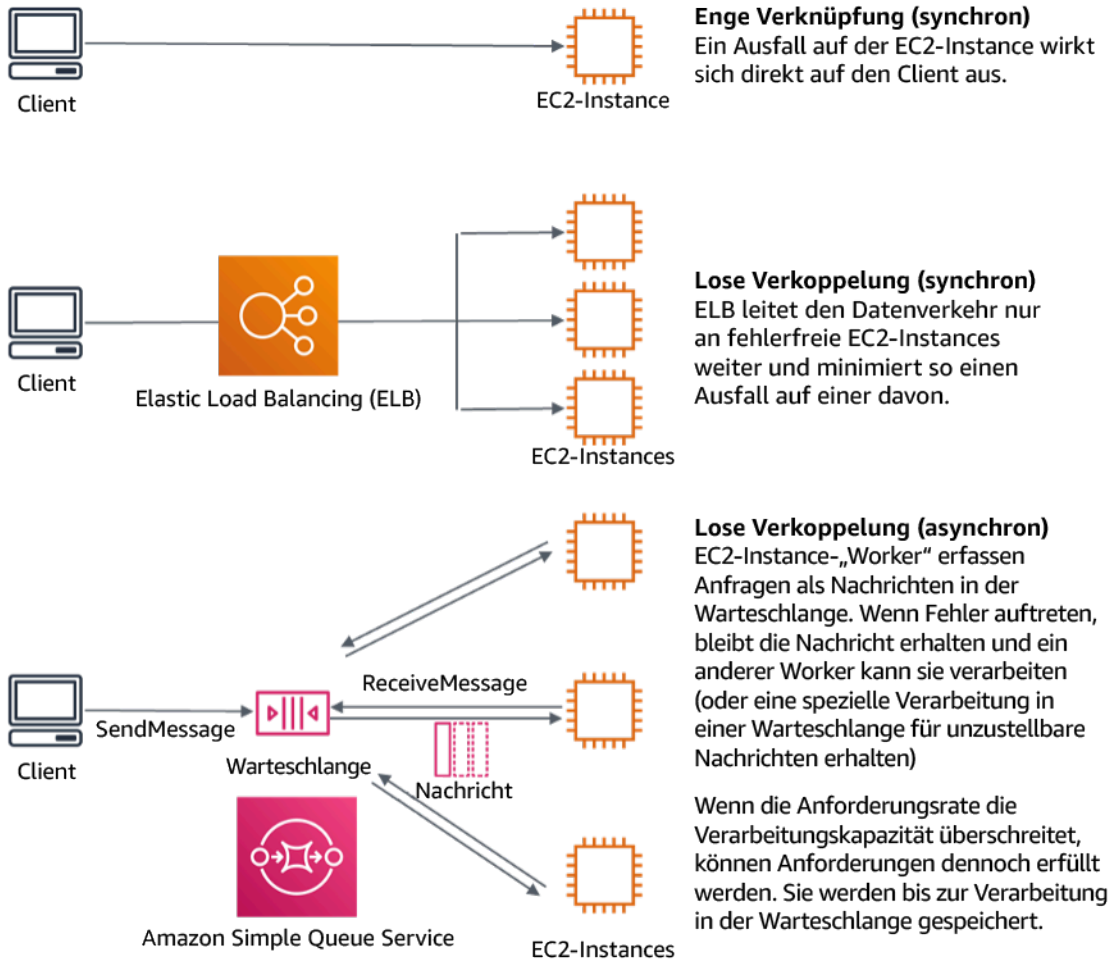


Abbildung 4: Abhängigkeiten etwa zwischen Warteschlangensystemen und Load Balancer sind lose gekoppelt

Amazon SQS-Warteschlangen und Elastic Load Balancers sind nur zwei Möglichkeiten, um eine Zwischenschicht für lose Kopplung hinzuzufügen. Ereignisgesteuerte Architekturen können auch in der AWS Cloud mithilfe von Amazon EventBridge erstellt werden, was Clients (Ereignisproduzenten) von den Services abstrahieren kann, auf die sie sich verlassen (Ereignisverbraucher). Amazon Simple Notification Service (Amazon SNS) ist eine effektive Lösung, wenn Sie Push-basiertes Many-to-Many-Messaging mit hohem Durchsatz benötigen. Mithilfe von Amazon SNS-Themen können Ihre Publisher-Systeme Nachrichten zur parallelen Verarbeitung an eine große Anzahl von Abonnenten-Endpunkten senden.

Warteschlangen bieten zwar mehrere Vorteile, doch Anfragen, die älter als ein Schwellenwert sind (oft Sekunden), sollten in den meisten harten Echtzeitsystemen als veraltet betrachtet (der Client hat aufgegeben und wartet nicht mehr auf eine Antwort) und nicht verarbeitet werden. Auf diese Weise können stattdessen neuere (und wahrscheinlich noch gültige Anfragen) verarbeitet werden.

Gewünschtes Ergebnis: Wenn Sie lose gekoppelte Abhängigkeiten implementieren, können Sie die Fehlerfläche auf Komponentenebene minimieren, was die Diagnose und Lösung von Problemen erleichtert. Außerdem vereinfacht es die Entwicklungszyklen, da die Teams Änderungen auf modularer Ebene implementieren können, ohne die Leistung anderer Komponenten, die davon abhängen, zu beeinträchtigen. Dieser Ansatz ermöglicht eine Aufskalierung auf Komponentenebene auf Grundlage des Ressourcenbedarfs sowie der Auslastung einer Komponente und trägt so zur Kosteneffizienz bei.

Typische Anti-Muster:

- Bereitstellen eines monolithischen Workloads.
- APIs werden zwischen Workload-Ebenen direkt aufgerufen, ohne Möglichkeit eines Failovers oder einer asynchronen Verarbeitung der Anfrage.
- Enge Verkoppelung mithilfe gemeinsam genutzter Daten. Lose gekoppelte Systeme sollten die gemeinsame Nutzung von Daten durch gemeinsam genutzte Datenbanken oder andere Formen der eng gekoppelten Datenspeicherung vermeiden, da dies wieder zu einer engen Verkoppelung führen und die Skalierbarkeit behindern kann.
- Gegendruck wird ignoriert. Ihr Workload sollte in der Lage sein, die eingehenden Daten zu verlangsamen oder zu stoppen, wenn eine Komponente sie nicht mit der gleichen Geschwindigkeit verarbeiten kann.

Vorteile der Nutzung dieser bewährten Methode: Eine lose Verkoppelung hilft dabei, das Verhalten einer Komponente von anderen Komponenten zu isolieren, die von ihr abhängen, wodurch die Resilienz und Agilität erhöht werden. Fehler in einer Komponente sind von anderen isoliert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Implementieren lose gekoppelter Abhängigkeiten Es gibt verschiedene Lösungen, mit denen Sie lose gekoppelte Anwendungen erstellen können. Dazu gehören u. a. Services für die Implementierung vollständig verwalteter Warteschlangen, automatisierter Workflows, die Reaktion auf Ereignisse und APIs, die dazu beitragen können, das Verhalten von Komponenten gegenüber anderen Komponenten zu isolieren und so die Ausfallsicherheit und Agilität zu erhöhen.

- Aufbau ereignisgesteuerter Architekturen: Amazon EventBridge hilft Ihnen beim Aufbau lose gekoppelter und verteilter ereignisgesteuerter Architekturen.

- Implementieren von Warteschlangen in verteilten Systemen: Sie können Amazon Simple Queue Service (Amazon SQS) verwenden, um verteilte Systeme zu integrieren und zu entkoppeln.
- Containerisieren Sie Komponenten als Microservices: Microservices ermöglichen es Teams, Anwendungen zu erstellen, die aus kleinen unabhängigen Komponenten bestehen, die über wohldefinierte APIs kommunizieren. Amazon Elastic Container Service (Amazon ECS) und Amazon Elastic Kubernetes Service (Amazon EKS) können Ihnen helfen, schneller mit Containern zu beginnen.
- Verwalten der Workflows mit Step Functions: Step Functions hilft Ihnen, mehrere AWS-Dienste in flexiblen Workflows zu koordinieren.
- Nutzen von Publish-Subscribe (Pub/Sub)-Messaging-Architekturen: Amazon Simple Notification Service(Amazon SNS) sorgt für die Zustellung von Nachrichten von Publishern an Abonnenten (auch als Produzenten und Verbraucher bezeichnet).

Implementierungsschritte

- Komponenten in einer ereignisgesteuerten Architektur werden durch Ereignisse ausgelöst. Ereignisse sind Aktionen, die in einem System stattfinden, z. B. wenn ein Benutzer einen Artikel in den Warenkorb legt. Wenn eine Aktion erfolgreich ist, wird ein Ereignis erzeugt, das die nächste Komponente des Systems auslöst.
  - Erstellen ereignisgesteuerter Anwendungen mit Amazon EventBridge
  - AWS re:Invent 2022 - Designing Event-Driven Integrations using Amazon EventBridge (AWS re:Invent 2022 – Entwurf ereignisgesteuerter Integrationen mit Amazon EventBridge)
- Verteilte Nachrichtensysteme haben drei Hauptbestandteile, die für eine warteschlangenbasierte Architektur implementiert werden müssen. Dazu gehören Komponenten des verteilten Systems, die Warteschlange, die für die Entkopplung verwendet wird (auf Amazon SQS-Servern verteilt), und die Nachrichten in der Warteschlange. Ein typisches System hat einen Produzenten, der die Nachricht in die Warteschlange einstellt, und einen Verbraucher, der die Nachricht aus der Warteschlange empfängt. Die Warteschlange speichert Nachrichten aus Redundanzgründen auf mehreren Amazon SQS-Servern.
  - Grundlegende Amazon SQS-Architektur
  - Senden von Nachrichten zwischen verteilten Anwendungen mit Amazon Simple Queue Service
- Wenn Microservices gut genutzt werden, verbessern sie die Wartbarkeit und die Skalierbarkeit, da lose gekoppelte Komponenten von unabhängigen Teams verwaltet werden. Sie ermöglichen zudem die Isolierung von Verhaltensweisen auf eine einzelne Komponente im Falle von Änderungen.

  - Implementieren von Microservices in AWS
  - Let's Architect! Architektur von Microservices mit Containern
- Mit AWS Step Functions können Sie unter anderem verteilte Anwendungen erstellen, Prozesse automatisieren und Microservices orchestrieren. Die Orchestrierung mehrerer Komponenten in einem automatisierten Workflow ermöglicht es Ihnen, Abhängigkeiten in Ihrer Anwendung zu entkoppeln.
  - Erstellen eines Serverless Workflows mit AWS Step Functions und AWS Lambda
  - Erste Schritte mit AWS Step Functions

Ressourcen

Zugehörige Dokumente:

- Amazon EC2: Idempotenz sicherstellen
- Die Amazon Builders' Library: Herausforderungen für verteilte Systeme
- Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee
- Was ist Amazon EventBridge?
- Was ist Amazon Simple Queue Service?
- Break up with your monolith (Teilen Sie den Monolithen auf)
- Orchestrate Queue-based Microservices with AWS Step Functions and Amazon SQS (Orchestrieren der warteschlangenbasierten Microservices mit AWS Step Functions und Amazon SQS)
- Grundlegende Amazon SQS-Architektur
- Warteschlangenbasierte Architektur

Zugehörige Videos:

- AWS New York Summit 2019: Intro to Event-driven Architectures and Amazon EventBridge (MAD205) (AWS New York Summit 2019: Einführung in ereignisgesteuerte Architekturen und Amazon EventBridge [MAD205])
- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 (includes loose coupling, constant work, static stability) (AWS re:Invent 2018: Close Loops und Opening Minds: Wie man die Kontrolle über große und kleine Systeme übernimmt ARC337 [umfasst lose Verkoppelung, konstante Ausführung, statische Stabilität])

- AWS re:Invent 2019: Moving to event-driven architectures (SVS308) (Umstieg auf ereignisgesteuerte Architekturen)
- AWS re:Invent 2019: Scalable serverless event-driven applications using Amazon SQS and Lambda (API304) (AWS re:Invent 2019: Skalierbare ereignisgesteuerte Serverless-Anwendungen, die Amazon SQS und Lambda nutzen [API304])
- AWS re:Invent 2019: Scalable serverless event-driven applications using Amazon SQS and Lambda (AWS re:Invent 2019: Skalierbare ereignisgesteuerte Serverless-Anwendungen, die Amazon SQS und Lambda nutzen)
- AWS re:Invent 2022 - Designing event-driven integrations using Amazon EventBridge (AWS re:Invent 2022 – Entwurf ereignisgesteuerter Integrationen mit Amazon EventBridge)
- AWS re:Invent 2017: Elastic Load Balancing Deep Dive and Best Practices (AWS re:Invent 2017: Elastic Load Balancing – Vertiefung und bewährte Praktiken)

REL04-BP03 Konstante Ausführung

Bei größeren, schnellen Lastveränderungen können Systeme ausfallen. Wenn Ihre Workload beispielsweise eine Zustandsprüfung ausführt, die den Zustand vieler tausend Server überwacht, sollte sie jedes Mal die gleiche Nutzlast senden (einen vollständigen Snapshot des aktuellen Status). Unabhängig davon, ob keine Server oder alle Server ausfallen, führt das System für die Zustandsprüfung die Aufgaben stetig und ohne große, schnelle Änderungen aus.

Wenn das Zustandsprüfungssystem beispielsweise 100 000 Server überwacht, ist die Last darauf angesichts der normalerweise geringen Serverausfallrate nominal. Wenn jedoch ein großes Ereignis die Hälfte dieser Server fehlerhaft macht, wäre das Zustandsprüfungssystem überfordert, wenn es versucht, Benachrichtigungssysteme zu aktualisieren und den Status an seine Clients zu kommunizieren. Stattdessen sollte das Zustandsprüfungssystem jedes Mal den vollständigen Snapshot des aktuellen Status senden. 100 000 Server-Zustände, die jeweils durch ein Bit dargestellt werden, entsprächen nur eine Nutzlast von 12,5 KB. Unabhängig davon, ob keine oder alle Server ausfallen – das System für die Zustandsprüfung erledigt seine Arbeit konstant und große, schnelle Änderungen stellen keine Bedrohung für die Systemstabilität dar. Auf diese Weise führt Amazon Route 53 Zustandsprüfungen für Endpunkte (wie z. B. IP-Adressen) durch, um zu ermitteln, wie Endbenutzer an diese weitergeleitet werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

- Führen Sie Aufgaben konstant aus, sodass auch bei großen, schnellen Lastveränderungen keine Fehler auf Systemen auftreten.
- Implementieren Sie lose gekoppelte Abhängigkeiten. Abhängigkeiten etwa zwischen Warteschlangensystemen, Streaming-Systemen, Workflows und Load Balancern sind lose gekoppelt. Eine lose Verkoppelung hilft, das Verhalten einer Komponente von anderen Komponenten zu isolieren, die von ihr abhängig sind. Dies verbessert Resilienz und Agilität.
  - Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee
  - AWS re:Invent 2018: Kreisläufe schließen & aufgeschlossen sein: Wie man die Kontrolle über große und kleine Systeme übernimmt ARC337 (umfasst konstante Ausführung)
    - Beispiel: Zustandsprüfungssystem, das 100.000 Server überwacht: Entwickeln Sie die Workloads so, dass die Nutzlastgrößen unabhängig von der Anzahl der Erfolge oder Ausfälle konstant bleiben.

Ressourcen

Ähnliche Dokumente:

- Amazon EC2: Idempotenz sicherstellen
- Die Amazon Builders' Library: Herausforderungen für verteilte Systeme
- Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee

Ähnliche Videos:

- AWS New York Summit 2019: Einführung in ereignisgesteuerte Architekturen und Amazon EventBridge (MAD205)
- AWS re:Invent 2018: Kreisläufe schließen & aufgeschlossen sein: Wie man die Kontrolle über große und kleine Systeme übernimmt ARC337 (umfasst konstante Ausführung)
- AWS re:Invent 2018: Kreisläufe schließen & aufgeschlossen sein: Wie man die Kontrolle über Systeme übernimmt – große und kleine ARC337 (umfasst lose Verkoppelung, konstante Ausführung, statische Stabilität)
- AWS re:Invent 2019: Moving to event-driven architectures (SVS308) (Umstieg auf ereignisgesteuerte Architekturen)

REL04-BP04 Festlegen aller Reaktionen als idempotent

Ein idempotenter Service garantiert, dass jede Anfrage genau einmal abgeschlossen wird. Das bedeutet, dass das Senden mehrerer identischer Anfragen den gleichen Effekt hat wie das Senden einer einzelnen Anfrage. Ein idempotenter Service erleichtert es einem Client, Wiederholungen zu implementieren. So muss nicht befürchtet werden, dass eine Anfrage fälschlicherweise mehrfach verarbeitet wird. Zu diesem Zweck können Clients API-Anfragen mit einem Idempotenz-Token ausgeben. Das gleiche Token wird verwendet, wenn die Anfrage wiederholt wird. Eine idempotente Service-API gibt mithilfe des Tokens eine Antwort zurück, die identisch mit der Antwort ist, die beim ersten Abschluss der Anfrage zurückgegeben wurde.

In einem verteilten System ist es einfach, eine Aktion höchstens einmal (der Client stellt nur eine Anforderung) oder mindestens einmal (Anforderung so lange, bis der Client erfolgreich ist) durchzuführen. Es ist jedoch schwer zu gewährleisten, dass eine Aktion idempotent ist, was bedeutet, dass sie genau einmal ausgeführt wird, sodass das Erstellen mehrerer identischer Anfragen den gleichen Effekt hat wie das Erstellen einer einzelnen Anfrage. Durch die Verwendung von idempotenten Tokens in APIs können Services einmal oder mehrmals eine sich verändernde Anfrage erhalten, ohne dass doppelte Datensätze erstellt werden oder sonstige Probleme entstehen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Legen Sie alle Reaktionen als idempotent fest. Ein idempotenter Service garantiert, dass jede Anfrage genau einmal abgeschlossen wird. Das bedeutet, dass das Senden mehrerer identischer Anfragen den gleichen Effekt hat wie das Senden einer einzelnen Anfrage.
  - Clients können API-Anfragen mit einem Idempotenz-Token ausgeben. Das gleiche Token wird bei einer Wiederholung der Anfrage verwendet. Eine idempotente Service-API gibt mithilfe des Tokens eine Antwort zurück, die identisch mit der Antwort ist, die beim ersten Abschluss der Anfrage zurückgegeben wurde.
    - Amazon EC2: Idempotenz sicherstellen

Ressourcen

Ähnliche Dokumente:

- Amazon EC2: Idempotenz sicherstellen
- Die Amazon Builders' Library: Herausforderungen bei verteilten Systemen

- Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee

Ähnliche Videos:

- AWS New York Summit 2019: Einführung in ereignisgesteuerte Architekturen und Amazon EventBridge (MAD205)
- AWS re:Invent 2018: Kreisläufe schließen & aufgeschlossen sein: Wie man die Kontrolle über Systeme übernimmt – große und kleine ARC337 (umfasst lose Verkoppelung, konstante Ausführung, statische Stabilität)
- AWS re:Invent 2019: Moving to event-driven architectures (SVS308) (Umstieg auf ereignisgesteuerte Architekturen)

REL 5. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle abgemildert oder bewältigt werden?

Verteilte Systeme nutzen Kommunikationsnetzwerke, um Komponenten (wie Server oder Services) miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder höherer Latenz in diesen Netzwerken zuverlässig ausgeführt werden. Komponenten des verteilten Systems müssen so funktionieren, dass sie keine negativen Auswirkungen auf andere Komponenten oder die Workload haben. Diese bewährten Methoden sorgen dafür, dass Workloads Belastungen oder Fehlern standhalten, sich schneller davon erholen und die Auswirkungen solcher Beeinträchtigungen abgeschwächt werden. Das Ergebnis ist eine verbesserte mittlere Reparaturzeit (MTTR).

Bewährte Methoden

- REL05-BP01 Implementieren einer ordnungsgemäßen Funktionsminderung, um harte Abhängigkeiten in weiche zu ändern
- REL05-BP02 Drosselung von Anfragen
- REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen
- REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen
- REL05-BP05 Festlegen von Client-Zeitüberschreitungen
- REL05-BP06 Erstellen zustandsloser Systeme
- REL05-BP07 Implementieren von Nothebeln

REL05-BP01 Implementieren einer ordnungsgemäßen Funktionsminderung, um harte Abhängigkeiten in weiche zu ändern

Anwendungskomponenten sollten weiterhin ihre Kernfunktion erfüllen, auch wenn Abhängigkeiten nicht mehr verfügbar sind. Sie liefern möglicherweise leicht veraltete Daten, alternative Daten oder sogar keine Daten. Dadurch wird sichergestellt, dass die Gesamtsystemfunktion nur minimal durch lokale Ausfälle beeinträchtigt wird, während gleichzeitig der zentrale Geschäftswert gewährleistet ist.

Gewünschtes Ergebnis: Wenn die Abhängigkeiten einer Komponente fehlerhaft sind, kann die Komponente selbst weiterhin funktionieren, wenn auch in eingeschränkter Weise. Komponentenausfälle sollten als normaler Geschäftsbetrieb betrachtet werden. Arbeitsabläufe sollten so konzipiert sein, dass solche Ausfälle nicht zu einem vollständigen Ausfall oder zumindest zu vorhersehbaren und wiederherstellbaren Zuständen führen.

Typische Anti-Muster:

- Die erforderlichen Kerngeschäftsfunktionen wurden nicht identifiziert. Es wird nicht getestet, ob die Komponenten auch bei Abhängigkeitsfehlern funktionsfähig sind.
- Es werden keine Daten zu Fehlern bereitgestellt oder wenn nur eine von mehreren Abhängigkeiten nicht verfügbar ist und Teilergebnisse dennoch zurückgegeben werden können.
- Es entsteht ein inkonsistenter Zustand, wenn eine Transaktion teilweise fehlschlägt.
- Es gibt keine alternative Möglichkeit, auf einen zentralen Parameterspeicher zuzugreifen.
- Lokale Zustände werden aufgrund einer fehlgeschlagenen Aktualisierung ungültig oder geleert, ohne die Konsequenzen zu berücksichtigen.

Vorteile der Nutzung dieser bewährten Methode: Eine schrittweise Degradation verbessert die Verfügbarkeit des gesamten Systems und gewährleistet die Funktionsfähigkeit der wichtigsten Funktionen auch bei Ausfällen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Die Implementierung einer schrittweisen Degradation trägt dazu bei, die Auswirkungen von Abhängigkeitsfehlern auf die Komponentenfunktion zu minimieren. Im Idealfall erkennt eine Komponente Abhängigkeitsfehler und umgeht sie so, dass sich dies nur minimal auf andere Komponenten oder Kunden auswirkt.

Eine Architektur, die auf eine schrittweise Degradation ausgerichtet ist, bedeutet, potenzielle Ausfallmodi beim Entwurf von Abhängigkeiten zu berücksichtigen. Sorgen Sie für jeden Ausfallmodus für eine Möglichkeit, aufrufenden Komponenten oder Kunden die meisten oder zumindest die wichtigsten Funktionen der Komponente bereitzustellen. Diese Überlegungen können zu zusätzlichen Anforderungen werden, die getestet und verifiziert werden können. Im Idealfall ist eine Komponente in der Lage, ihre Kernfunktion auf akzeptable Weise auszuführen, selbst wenn eine oder mehrere Abhängigkeiten ausfallen.

Dies ist sowohl eine geschäftliche als auch eine technische Diskussion. Alle Geschäftsanforderungen sind wichtig und sollten nach Möglichkeit erfüllt werden. Es ist jedoch immer noch sinnvoll, sich zu fragen, was passieren soll, wenn nicht alle erfüllt werden können. Ein System kann so konzipiert werden, dass es verfügbar und konsistent ist. Doch was davon ist wichtiger, wenn auf eines davon verzichtet werden muss? Bei der Zahlungsabwicklung könnte dies die Konsistenz sein. Bei einer Echtzeitanwendung ist es eher die Verfügbarkeit. Bei einer kundenseitigen Website kann die Antwort von den Kundenerwartungen abhängen.

Was das bedeutet, hängt von den Anforderungen der Komponente ab und davon, was als ihre Kernfunktion angesehen werden sollte. Zum Beispiel:

- Eine E-Commerce-Website kann Daten aus verschiedenen Systemen wie personalisierte Empfehlungen, bestbewertete Produkte und den Status von Kundenbestellungen auf der Startseite anzeigen. Wenn ein Upstream-System ausfällt, ist es immer noch sinnvoll, alles andere anzuzeigen, anstatt einem Kunden eine Fehlerseite anzuzeigen.
- Eine Komponente, die Batch-Schreibvorgänge durchführt, kann einen Stapel trotzdem weiterverarbeiten, wenn eine der einzelnen Operationen fehlschlägt. Es sollte einfach sein, einen Wiederholungsmechanismus zu implementieren. Geben Sie dazu Informationen dazu zurück, welche Operationen erfolgreich, welche fehlgeschlagen und warum sie fehlgeschlagen sind. Oder stellen Sie fehlgeschlagene Anfragen in eine Warteschlange für unzustellbare Nachrichten, um asynchrone Wiederholungsversuche zu implementieren. Informationen über fehlgeschlagene Operationen sollten ebenfalls protokolliert werden.
- Ein System, das Transaktionen verarbeitet, muss überprüfen, ob entweder alle oder keine einzelnen Aktualisierungen ausgeführt werden. Bei verteilten Transaktionen kann das Saga-Muster verwendet werden, um vorherige Operationen rückgängig zu machen, falls ein späterer Vorgang derselben Transaktion fehlschlägt. Hier besteht die Kernfunktion darin, die Konsistenz aufrechtzuerhalten.
- Zeitkritische Systeme sollten in der Lage sein, mit Abhängigkeiten umzugehen, die nicht rechtzeitig reagieren. In diesen Fällen kann das Unterbrechermuster verwendet werden. Wenn bei Antworten

aus einer Abhängigkeit eine Zeitüberschreitung auftritt, kann das System in einen geschlossenen Zustand wechseln, in dem keine weiteren Aufrufe getätigt werden.

- Eine Anwendung kann Parameter aus einem Parameterspeicher lesen. Es kann nützlich sein, Container-Images mit einem Satz von Standardparametern zu erstellen und diese zu verwenden, falls der Parameterspeicher nicht verfügbar ist.

Beachten Sie, dass die im Falle eines Komponentenausfalls eingeschlagenen Pfade getestet werden müssen und deutlich einfacher sein sollten als der primäre Pfad. Allgemein sollten Fallback-Strategien vermieden werden.

Implementierungsschritte

Identifizieren Sie externe und interne Abhängigkeiten. Überlegen Sie, welche Arten von Fehlern bei ihnen auftreten können. Überlegen Sie, wie Sie die negativen Auswirkungen dieser Ausfälle auf vor- und nachgeschaltete Systeme und Kunden minimieren können.

Im Folgenden finden Sie eine Liste von Abhängigkeiten und wie Sie sie schrittweise degradieren können, wenn sie ausfallen:

1. Teilweiser Ausfall von Abhängigkeiten: Eine Komponente kann mehrere Anfragen an nachgelagerte Systeme stellen, entweder in Form mehrerer Anfragen an ein System oder in Form einer Anfrage an jeweils mehrere Systeme. Je nach Unternehmenskontext können unterschiedliche Vorgehensweisen angemessen sein (weitere Einzelheiten finden Sie in den vorherigen Beispielen in den Implementierungsleitfäden).
2. Ein nachgelagertes System kann Anfragen aufgrund der hohen Auslastung nicht verarbeiten: Wenn Anfragen an ein nachgelagertes System immer wieder fehlschlagen, ist es nicht sinnvoll, es erneut zu versuchen. Dies kann ein bereits überlastetes System zusätzlich belasten und die Wiederherstellung erschweren. Hier kann das Unterbrechermuster verwendet werden, das fehlgeschlagene Aufrufe an ein nachgelagertes System überwacht. Wenn eine große Anzahl von Aufrufen fehlschlägt, werden keine weiteren Anfragen mehr an das nachgelagerte System gesendet und nur gelegentlich Aufrufe durchgelassen, um zu testen, ob das nachgelagerte System wieder verfügbar ist.
3. Ein Parameterspeicher ist nicht verfügbar: Um einen Parameterspeicher umzuwandeln, können Soft Dependency Caching oder vernünftige Standardwerte verwendet werden, die in Container-Images oder Machine Images enthalten sind. Beachten Sie, dass diese Standardwerte auf dem neuesten Stand gehalten und in die Testsuiten aufgenommen werden müssen.

4. Ein Überwachungsservice oder eine andere nicht funktionale Abhängigkeit ist nicht verfügbar: Wenn eine Komponente zeitweise nicht in der Lage ist, Protokolle, Metriken oder Spuren an einen zentralen Überwachungsservice zu senden, ist es oft am besten, Geschäftsfunktionen weiterhin wie gewohnt auszuführen. Es ist oft nicht akzeptabel, Metriken über einen längeren Zeitraum stillschweigend nicht zu protokollieren oder weiterzuleiten. In einigen Anwendungsfällen können auch vollständige Auditeinträge erforderlich sein, um die Compliance-Anforderungen zu erfüllen.
5. Eine primäre Instances einer relationalen Datenbank ist möglicherweise nicht verfügbar: Amazon Relational Database Service kann, wie fast alle relationalen Datenbanken, nur eine primäre Writer-Instance haben. Dies führt zu einem einzigen Fehlerpunkt für Schreib-Workloads und erschwert die Skalierung. Dies kann teilweise gemildert werden, indem eine Multi-AZ-Konfiguration für hohe Verfügbarkeit oder Amazon Aurora Serverless für eine bessere Skalierung verwendet wird. Bei sehr hohen Verfügbarkeitsanforderungen kann es sinnvoll sein, sich überhaupt nicht auf den primären Writer zu verlassen. Für Abfragen, die nur lesen, können Lesereplikate verwendet werden, die Redundanz und die Möglichkeit bieten, nicht nur hoch-, sondern auch aufzuskalieren. Schreibvorgänge können gepuffert werden, zum Beispiel in einer Amazon Simple Queue Service-Warteschlange, sodass Schreibanfragen von Kunden auch dann akzeptiert werden können, wenn das primäre Gerät vorübergehend nicht verfügbar ist.

Ressourcen

Zugehörige Dokumente:

- Amazon API Gateway: Throttle API Requests for Better Throughput (Amazon API Gateway: Drosseln von API-Anfragen für einen besseren Durchsatz)
- CircuitBreaker (Zusammenfassung des Circuit Breaker aus dem Buch „Release It!“)
- Error Retries and Exponential Backoff in AWS (Fehlerwiederholungen und exponentielles Backoff in AWS)
- Michael Nygard, „Release It!“ Design and Deploy Production-Ready Software“
- Die Amazon Builders' Library: Vermeiden von Fallback in verteilten Systemen
- Die Amazon Builders' Library: Vermeiden von nicht mehr aufholbaren Warteschlangen-Rückständen
- Die Amazon Builders' Library: Herausforderungen und Strategien für das Caching
- Die Amazon Builders' Library: Timeouts, Wiederholungen und Backoff mit Jitter

Zugehörige Videos:

- Wiederholung, Backoff und Jitter: AWS re:Invent 2019: Einführung in die Amazon Builders' Library (DOP328)

Zugehörige Beispiele:

- Well-Architected Lab: Level 300: Implementieren von Zustandsprüfungen und Verwalten von Abhängigkeiten zur Verbesserung der Zuverlässigkeit

REL05-BP02 Drosselung von Anfragen

Drosseln Sie Anfragen, um eine Ressourcenüberlastung aufgrund eines unerwarteten Nachfrageanstiegs zu verringern. Anfragen, die unter der Drosselungsrate liegen, werden verarbeitet, während Anfragen, die über dem definierten Limit liegen, abgelehnt werden. Es wird eine Meldung zurückgegeben, die besagt, dass die Anfrage gedrosselt wurde.

Gewünschtes Ergebnis: Stark ansteigendes Volumen, das entweder durch plötzliche Anstiege des Kundendatenverkehrs, Flooding-Angriffe oder Wiederholungsstürme verursacht wird, wird durch Anfragedrosselung abgeschwächt, sodass Workloads die normale Verarbeitung des unterstützten Anforderungsvolumens fortsetzen können.

Typische Anti-Muster:

- API-Endpunktdrosselungen sind nicht implementiert oder werden auf Standardwerten belassen, ohne die erwarteten Volumina zu berücksichtigen.
- API-Endpunkte werden nicht ausgelastet oder die Drosselungsgrenzwerte werden nicht getestet.
- Anforderungsraten werden ohne Berücksichtigung der Größe oder Komplexität der Anfrage gedrosselt.
- Es werden sowohl die maximalen Anforderungsraten als auch die maximale Anforderungsgröße getestet, aber nicht beides zusammen.
- Ressourcen werden nicht mit denselben Limits bereitgestellt, die beim Testen festgelegt wurden.
- Es wurden keine Nutzungspläne konfiguriert oder für A2A-API-Verbraucher in Betracht gezogen.
- Für Warteschlangenverbraucher, die horizontal skalieren, sind keine Einstellungen für maximale Parallelität konfiguriert.
- Eine Ratenbegrenzung pro IP-Adresse wurde nicht implementiert.

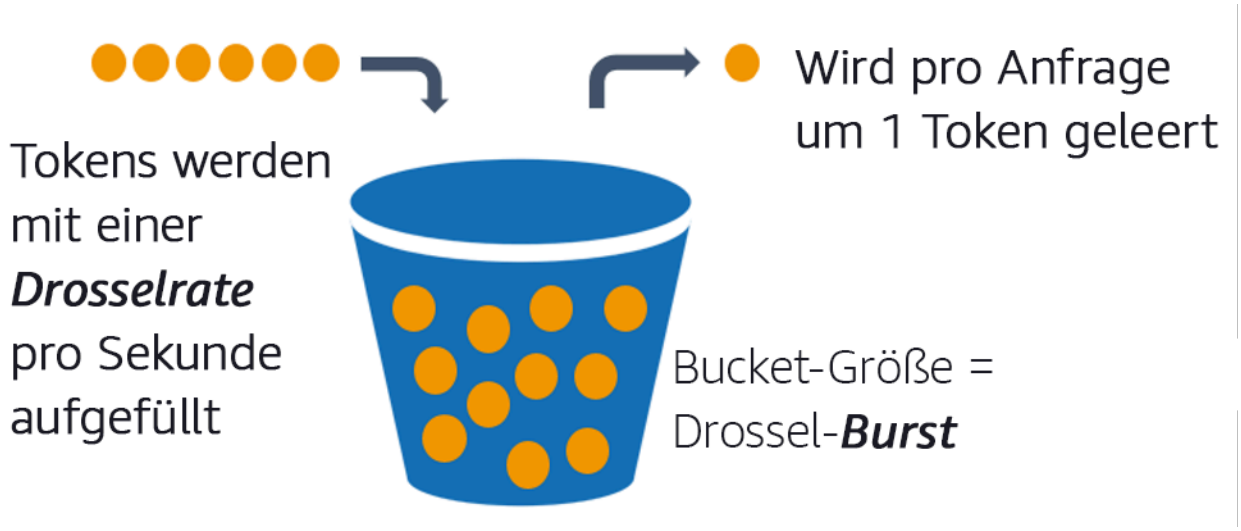
Vorteile der Nutzung dieser bewährten Methode: Workloads, die Drosselgrenzwerte festlegen, können normal arbeiten und akzeptierte Anfragen auch bei unerwarteten Volumenspitzen erfolgreich verarbeiten. Plötzliche oder anhaltende Spitzen von Anfragen an APIs und Warteschlangen werden gedrosselt und verbrauchen keine Ressourcen für die Anforderungsverarbeitung. Ratenbegrenzungen drosseln einzelne Anforderer, sodass ein hohes Datenverkehrsvolumen von einer einzelnen IP-Adresse oder einem API-Verbraucher keine Ressourcen verbraucht, die sich auf andere Verbraucher auswirken.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Services sollten so konzipiert sein, dass sie eine bekannte Kapazität von Anfragen verarbeiten. Diese Kapazität kann durch Auslastungstests ermittelt werden. Wenn die Anzahl der Anfragen die Grenzwerte überschreitet, signalisiert die entsprechende Antwort, dass eine Anfrage gedrosselt wurde. Dies ermöglicht es dem Verbraucher, den Fehler zu beheben und es später erneut zu versuchen.

Wenn für Ihren Service eine Drosselungsimplementierung erforderlich ist, sollten Sie die Implementierung des Token-Bucket-Algorithmus in Betracht ziehen, bei dem ein Token für eine Anfrage zählt. Tokens werden mit einer Drosselrate pro Sekunde aufgefüllt und asynchron um ein Token pro Anfrage geleert.



Der Token-Bucket-Algorithmus

Amazon API Gateway implementiert den Token-Bucket-Algorithmus entsprechend den Konto- und Regionslimits und kann pro Client mit Nutzungsplänen konfiguriert werden. Darüber hinaus können

Amazon Simple Queue Service (Amazon SQS) und Amazon Kinesis Anfragen zwischenspeichern, um die Anforderungsrate auszugleichen, und höhere Drosselungsraten für Anfragen ermöglichen, die bearbeitet werden können. Schließlich können Sie die Ratenbegrenzung mit AWS WAF implementieren, um bestimmte API-Verbraucher zu drosseln, die ungewöhnlich hohe Lasten erzeugen.

Implementierungsschritte

Sie können API Gateway mit Drosselungslimits für Ihre APIs konfigurieren und „`429 Too Many Requests`“ -Fehler zurückgeben, wenn Grenzwerte überschritten werden. Sie können AWS WAF zusammen mit Ihren AWS AppSync- und API Gateway-Endpunkten verwenden, um die Ratenbegrenzung pro IP-Adresse zu aktivieren. Wenn Ihr System asynchrone Verarbeitung toleriert, können Sie außerdem Nachrichten in eine Warteschlange oder einen Stream stellen, um die Antworten an Service-Clients zu beschleunigen und so höhere Drosselungsraten zu erreichen.

Wenn Sie Amazon SQS als Ereignisquelle für AWS Lambda konfiguriert haben, können Sie mit asynchroner Verarbeitung maximale Gleichzeitigkeit konfigurieren, um zu verhindern, dass hohe Ereignisraten die für andere Services in Ihrem Workload oder Konto benötigten Kontingente für gleichzeitige Ausführungen auf Kontoebene verbrauchen.

API Gateway bietet zwar eine verwaltete Implementierung des Token-Buckets, aber in Fällen, in denen Sie API Gateway nicht verwenden können, können Sie sprachspezifische Open-Source-Implementierungen (siehe entsprechende Beispiele unter Ressourcen) des Token-Buckets für Ihre Services nutzen.

- Verstehen und konfigurieren Sie API Gateway-Drosselungslimits auf Kontoebene pro Region, API pro Phase und API-Schlüssel pro Nutzungsplanebene.
- Wenden Sie die AWS WAF-Regeln zur Ratenbegrenzung auf API Gateway- und AWS AppSync-Endpunkte an, um sich vor Flooding zu schützen und schädliche IPs zu sperren. Regeln zur Ratenbegrenzung können auch für AWS AppSync-API-Schlüssel für A2A-Verbraucher konfiguriert werden.
- Überlegen Sie, ob Sie für AWS AppSync-APIs mehr Drosselungskontrolle als Ratenbegrenzung benötigen, und konfigurieren Sie in diesem Fall ein API Gateway vor Ihrem AWS AppSync-Endpunkt.
- Wenn Amazon SQS-Warteschlangen als Auslöser für Lambda-Warteschlangenverbraucher eingerichtet werden, legen Sie die maximale Gleichzeitigkeit auf einen Wert fest, mit dem genug verarbeitet wird, um Ihre Service-Level-Ziele zu erreichen, aber keine Gleichzeitigkeitsbeschränkungen ausnutzt werden, die sich auf andere Lambda-Funktionen

auswirken. Erwägen Sie, die reservierte Gleichzeitigkeit für andere Lambda-Funktionen in demselben Konto und derselben Region festzulegen, wenn Sie Warteschlangen mit Lambda verbrauchen.

- Verwenden Sie API Gateway mit nativen Serviceintegrationen in Amazon SQS oder Kinesis, um Anfragen zwischenzuspeichern.
- Wenn Sie API Gateway nicht verwenden können, nutzen Sie sprachspezifische Bibliotheken, um den Token-Bucket-Algorithmus für Ihren Workload zu implementieren. Sehen Sie sich den Abschnitt mit den Beispielen an und recherchieren Sie selbst, um eine geeignete Bibliothek zu finden.
- Testen Sie Grenzwerte, die Sie festlegen oder deren Erhöhung Sie zulassen möchten, und dokumentieren Sie die getesteten Grenzwerte.
- Erhöhen Sie die Grenzwerte nicht über das hinaus, was Sie beim Testen festgelegt haben. Wenn Sie einen Grenzwert erhöhen, stellen Sie sicher, dass die bereitgestellten Ressourcen bereits denen in Testszenarien entsprechen oder diese übertreffen, bevor Sie die Erhöhung anwenden.

Ressourcen

Zugehörige bewährte Methoden:

- REL04-BP03 Konstante Ausführung
- REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen

Zugehörige Dokumente:

- Amazon API Gateway: Throttle API Requests for Better Throughput (Amazon API Gateway: Drosseln von API-Anfragen für einen besseren Durchsatz)
- AWS WAF: Rate-based rule statement (AWS WAF: Ratenbasierte Regelaussage)
- Introducing maximum concurrency of AWS Lambda when using Amazon SQS as an event source (Einführung maximaler Gleichzeitigkeit von AWS Lambda bei Verwendung von Amazon SQS als Ereignisquelle)
- AWS Lambda: Maximum Concurrency (AWS Lambda: Maximale Gleichzeitigkeit)

Zugehörige Beispiele:

- The three most important AWS WAF rate-based rules (Die drei wichtigsten ratenbasierten Regeln in AWS WAF)
- Java Bucket4j
- Python Token-Bucket
- Node-Token-Bucket
- .NET System Threading Rate Limiting (Ratenbegrenzung für .NET-System-Threading)

Zugehörige Videos:

- Implementing GraphQL API security best practices with AWS AppSync (Implementierung von bewährten Sicherheitsmethoden für GraphQL API mit AWS AppSync)

Zugehörige Tools:

- Amazon API Gateway
- AWS AppSync
- Amazon SQS
- Amazon Kinesis
- AWS WAF

REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen

Verwenden Sie das exponentielle Backoff, um Anfragen in zunehmend längeren Intervallen zwischen den einzelnen Wiederholungsversuchen zu wiederholen. Führen Sie Jitter zwischen den Wiederholungen ein, um die Wiederholungsintervalle zufällig zu bestimmen. Beschränken Sie die maximale Anzahl an Wiederholungen.

Gewünschtes Ergebnis: Typische Komponenten in einem verteilten Softwaresystem sind Server, Load Balancer, Datenbanken und DNS-Server. Während des normalen Betriebs können diese Komponenten auf Anfragen mit temporären oder begrenzten Fehlern sowie mit Fehlern antworten, die unabhängig von Wiederholungsversuchen dauerhaft bleiben würden. Wenn Clients Anfragen an Services stellen, verbrauchen die Anfragen Ressourcen wie Speicher, Threads, Verbindungen, Ports oder andere begrenzte Ressourcen. Die Steuerung und Einschränkung von Wiederholungsversuchen ist eine Strategie zur Freigabe und Minimierung des Ressourcenverbrauchs, sodass beanspruchte Systemkomponenten nicht überlastet werden.

Wenn Client-Anfragen eine Zeitüberschreitung oder Fehlerantworten erhalten, sollten sie entscheiden, ob sie es erneut versuchen möchten oder nicht. Wenn sie es erneut versuchen, tun sie dies mit exponentiellem Backoff mit Jitter und einem maximalen Wiederholungswert. Dadurch werden Backend-Services und -Prozesse entlastet und erhalten Zeit, um sich selbst zu reparieren, was zu einer schnelleren Wiederherstellung und einer erfolgreichen Bearbeitung von Anfragen führt.

Typische Anti-Muster:

- Wiederholungsversuche werden ohne exponentielles Backoff, Jitter und maximale Wiederholungswerte implementiert. Backoff und Jitter helfen dabei, künstliche Datenverkehrsspitzen zu vermeiden, die durch ungewollt koordinierte Wiederholungsversuche in regelmäßigen Intervallen entstehen.
- Wiederholungsversuche werden implementiert, ohne ihre Auswirkungen zu testen, oder es wird davon ausgegangen, dass Wiederholungsversuche bereits in ein SDK integriert sind, ohne Wiederholungsszenarien zu testen.
- Veröffentlichte Fehlercodes aus Abhängigkeiten werden nicht richtig interpretiert, was dazu führt, dass bei allen Fehlern eine Wiederholung versucht wird, auch dann, wenn die Ursache auf eine fehlende Berechtigung, einen Konfigurationsfehler oder ein anderes Problem hindeutet, das vorhersehbar nicht ohne manuelles Eingreifen behoben werden kann.
- Beobachtbarkeits-Praktiken, einschließlich der Überwachung und Meldung von Warnmeldungen bei wiederholten Serviceausfällen, damit die zugrunde liegenden Probleme bekannt werden und behoben werden können, werden nicht beachtet.
- Es werden benutzerdefinierte Wiederholungsmechanismen entwickelt, wenn integrierte Wiederholungsfunktionen oder Wiederholungsfunktionen von Drittanbietern ausreichen.
- Es werden Wiederholungsversuche auf mehreren Ebenen eines Anwendungsstapels auf eine Weise ausgeführt, die Wiederholungsversuche verstärkt, was die Ressourcen durch einen Wiederholungssturm weiter verbraucht. Vergewissern Sie sich, dass Sie verstehen, wie sich diese Fehler auf Ihre Anwendung und die Abhängigkeiten auswirken, auf die Sie sich verlassen, und führen Sie dann Wiederholungsversuche nur auf einer Ebene durch.
- Nicht idempotente Serviceaufrufe werden erneut versucht, was zu unerwarteten Nebeneffekten wie doppelten Ergebnissen führt.

Vorteile der Nutzung dieser bewährten Methode: Wiederholungsversuche helfen Clients dabei, die gewünschten Ergebnisse zu erzielen, wenn Anfragen fehlschlagen, verbrauchen aber auch mehr Zeit auf dem Server, um die gewünschten erfolgreichen Antworten zu erhalten. Wenn Fehler selten oder vorübergehend auftreten, funktionieren Wiederholungsversuche gut. Wenn Fehler

durch Ressourcenüberlastung verursacht werden, können Wiederholungsversuche die Situation verschlimmern. Durch das Hinzufügen eines exponentiellen Backoffs mit Jitter zu den Client-Wiederholungsversuchen können Server sich erholen, wenn Ausfälle durch Ressourcenüberlastung verursacht werden. Jitter verhindert, dass Anfragen zu Datenverkehrsspitzen führen, und Backoff verringert die Lasteskalation, die durch das Hinzufügen von Wiederholungsversuchen zur normalen Anforderungslast verursacht wird. Schließlich ist es wichtig, eine maximale Anzahl von Wiederholungsversuchen oder die verstrichene Zeit zu konfigurieren, um zu vermeiden, dass Rückstände entstehen, die zu metastabilen Ausfällen führen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Steuern und begrenzen Sie Wiederholungsaufrufe. Verwenden Sie ein exponentielles Backoff, um Aufrufe nach zunehmend längeren Intervallen zu wiederholen. Nutzen Sie Jitter, um die Wiederholungsintervalle zu randomisieren, und legen Sie ein Limit für die Zahl der Wiederholungen fest.

Mit AWS SDKs werden Wiederholungen und exponentielles Backoff standardmäßig implementiert. Verwenden Sie diese integrierten AWS-Implementierungen, sofern dies in Ihrem Workload erforderlich ist. Implementieren Sie eine ähnliche Logik in Ihrem Workload, wenn Sie Services aufrufen, die idempotent sind und bei denen Wiederholungsversuche die Verfügbarkeit Ihrer Clients verbessern. Legen Sie entsprechend Ihrem Anwendungsfall Zeitüberschreitungen fest und geben Sie an, wann Wiederholversuche gestoppt werden sollen. Erstellen Sie Testszenarien für diese Wiederholungsfälle und führen Sie sie aus.

Implementierungsschritte

- Ermitteln Sie die optimale Ebene in Ihrem Anwendungsstack, um Wiederholungsversuche für die Services zu implementieren, auf die sich Ihre Anwendung stützt.
- Seien Sie sich der vorhandenen SDKs bewusst, die bewährte Wiederholungsstrategien mit exponentiellem Backoff und Jitter für die Sprache Ihrer Wahl implementieren, und nutzen Sie eher diese, anstatt eigene Wiederholungsimplementierungen zu schreiben.
- Überprüfen Sie, dass Services idempotent sind, bevor Sie Wiederholungen implementieren. Sobald Wiederholungsversuche implementiert wurden, stellen Sie sicher, dass sie sowohl getestet als auch regelmäßig in der Produktion ausgeführt werden.
- Verwenden Sie beim Aufrufen von AWS-Service-APIs die AWS SDKs und AWS CLI und machen Sie sich mit den Konfigurationsoptionen für Wiederholungsversuche vertraut. Finden Sie heraus, ob

die Standardeinstellungen für Ihren Anwendungsfall geeignet sind, testen Sie sie und passen Sie sie nach Bedarf an.

Ressourcen

Zugehörige bewährte Methoden:

- REL04-BP04 Festlegen aller Reaktionen als idempotent
- REL05-BP02 Drosselung von Anfragen
- REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen
- REL05-BP05 Festlegen von Client-Zeitüberschreitungen
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Zugehörige Dokumente:

- Error Retries and Exponential Backoff in AWS (Fehlerwiederholungen und exponentielles Backoff in AWS)
- Die Amazon Builders' Library: Timeouts, Wiederholungen und Backoff mit Jitter
- Exponentielles Backoff und Jitter
- Making retries safe with idempotent APIs (Sichere Wiederholungsversuche mit idempotenten APIs)

Zugehörige Beispiele:

- Spring Retry (Spring-Wiederholung)
- Resilience4j Retry (Resilience4j-Wiederholung)

Zugehörige Videos:

- Wiederholung, Backoff und Jitter: AWS re:Invent 2019: Einführung in die Amazon Builders' Library (DOP328)

Zugehörige Tools:

- AWS SDKs und Tools: Wiederholungsverhalten
- AWS Command Line Interface: AWS CLI-Wiederholungen

REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen

Wenn ein Service nicht in der Lage ist, erfolgreich auf eine Anfrage zu antworten, sollte er schnell scheitern. Dies ermöglicht die Freigabe von mit einer Anfrage verbundenen Ressourcen und damit die Wiederherstellung eines Services, falls dieser nicht mehr über genügend Ressourcen verfügt. Schnelles Scheitern ist ein etabliertes Softwaredesignmuster, das genutzt werden kann, um hochzuverlässige Workloads in der Cloud aufzubauen. Warteschlangen sind ebenfalls ein etabliertes Integrationsmuster für Unternehmen. Sie sorgen für eine ausgeglichene Auslastung und ermöglichen es den Clients, Ressourcen freizugeben, wenn eine asynchrone Verarbeitung toleriert wird. Wenn ein Service unter normalen Bedingungen erfolgreich antworten kann, aber fehlschlägt, wenn die Anforderungsrate zu hoch ist, verwenden Sie eine Warteschlange, um Anfragen zwischenzuspeichern. Lassen Sie jedoch keine langen Warteschlangen zu. Sie können dazu führen, dass veraltete Anfragen verarbeitet werden, die ein Client bereits aufgegeben hat.

Gewünschtes Ergebnis: Wenn bei Systemen Ressourcenknappheit, Timeouts, Ausnahmen oder Grauausfälle auftreten, die Service-Level-Ziele unerreichbar machen, ermöglichen Strategien für schnelles scheitern eine schnellere Systemwiederherstellung. Systeme, die Traffic-Spitzen absorbieren müssen und asynchrone Verarbeitung ermöglichen, können die Zuverlässigkeit verbessern, indem sie es Clients ermöglichen, Anfragen schnell freizugeben, indem sie Warteschlangen verwenden, um Anfragen an Back-End-Services zu puffern. Beim Puffern von Anfragen in Warteschlangen werden Strategien zur Warteschlangenverwaltung implementiert, um nicht mehr aufzuholende Rückstände zu vermeiden.

Typische Anti-Muster:

- Implementierung von Nachrichtenwarteschlangen, aber keine Konfiguration von Warteschlangen für unzustellbare Nachrichten (DLQ) oder Alarmen für volle DLQs, um zu erkennen, wenn ein System ausfällt.
- Nichterfassung des Alters von Nachrichten in einer Warteschlange, einem Indikator für Latenz, um zu verstehen, wann Warteschlangenverbraucher mit der Verarbeitung nicht mehr hinterher kommen oder Fehler machen, was zu erneuten Versuchen führt.
- Kein Löschen von aufgestauten Nachrichten aus einer Warteschlange, wenn es keinen Sinn macht, diese Nachrichten zu verarbeiten, da kein Geschäftsbedarf mehr besteht.
- Die Konfiguration von First-in-First-Out (FIFO)-Warteschlangen, wenn Last-In-First-Out (LIFO)-Warteschlangen den Client-Anforderungen besser gerecht werden würden. Dies ist beispielsweise dann der Fall, wenn keine strenge Reihenfolge erforderlich ist und die Backlog-Verarbeitung alle neuen und zeitkritischen Anfragen verzögert, was dazu führt, dass alle Clients die Service-Levels nicht einhalten.

- Bereitstellung interner Warteschlangen für Clients, anstatt APIs verfügbar zu machen, die den Arbeitseingang verwalten und Anfragen in internen Warteschlangen platzieren.
- Wenn zu viele Arbeitsanforderungstypen in einer einzigen Warteschlange zusammengefasst werden, kann dies die Backlog-Bedingungen verschärfen, da der Ressourcenbedarf auf die verschiedenen Anforderungstypen verteilt wird.
- Verarbeitung komplexer und einfacher Anfragen in derselben Warteschlange, obwohl unterschiedliche Überwachungs-, Timeout- und Ressourcenzuweisungen erforderlich sind.
- Keine Validierung von Eingaben oder Nutzung von Aussagen, um Mechanismen für schnelles Scheitern in Software zu implementieren, die Ausnahmen an übergeordnete Komponenten weiterleiten, die Fehler problemlos verarbeiten können.
- Keine Entfernung fehlerhafter Ressourcen aus der Anforderungsweiterleitung, insbesondere bei Ausfällen ohne erkennbare Ursache mit sowohl erfolgreicher als auch fehlgeschlagener Verarbeitung aufgrund von Abstürzen und Neustarts, zeitweise auftretenden Abhängigkeitsfehlern, verringerter Kapazität oder Verlust von Netzwerkpaketen.

Vorteile der Nutzung dieser bewährten Methode: Systeme, die schnelles Scheitern nutzen, lassen sich leichter debuggen und korrigieren und weisen häufig Probleme im Code und in der Konfiguration auf, bevor Releases für die Produktion veröffentlicht werden. Systeme, die effektive Warteschlangenstrategien beinhalten, sind widerstandsfähiger und zuverlässiger bei Traffic-Spitzen und zeitweiligen Systemstörungen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Strategien für schnelles Scheitern können sowohl in Softwarelösungen als auch in der Infrastruktur konfiguriert werden. Warteschlangen scheitern nicht nur schnell, sondern sind auch eine einfache und dennoch leistungsstarke Architekturtechnik zur Entkopplung von Systemkomponenten für eine ausgeglichene Auslastung. Amazon CloudWatch bietet Funktionen zur Überwachung von Ausfällen und zur Warnung bei Ausfällen. Sobald erkannt wird, dass ein System ausfällt, können Strategien zur Schadensbegrenzung umgesetzt werden, darunter auch der Wechsel weg von knapp werdenden Ressourcen. Wenn in Systemen Warteschlangen mit Amazon SQS und anderen Warteschlangentechnologien implementiert werden, um eine ausgeglichene Auslastung zu gewährleisten, muss berücksichtigt werden, wie Warteschlangenrückstände sowie Fehler beim Nachrichtenabruf verwaltet werden können.

Implementierungsschritte

- Implementieren Sie programmatische Aussagen oder spezifische Metriken in Ihrer Software und verwenden Sie diese, um explizit Alarme bei Systemproblemen auszulösen. Amazon CloudWatch hilft Ihnen bei der Erstellung von Metriken und Alarmen auf der Grundlage des Anwendungsprotokollmusters und der SDK-Instrumentierung.
- Verwenden Sie CloudWatch-Metriken und Alarme, um knappe Ressourcen zu erkennen, die die Latenz bei der Verarbeitung erhöhen oder Anfragen wiederholt nicht bearbeiten können.
- Nutzen Sie asynchrone Verarbeitung, indem Sie APIs entwerfen, die Anfragen annehmen und an interne Warteschlangen anhängen. Verwenden Sie dazu Amazon SQS und senden Sie dann eine Erfolgsmeldung an den Nachrichten-Client, sodass der Client Ressourcen freigeben und mit anderen Arbeiten fortfahren kann, während die Verbraucher der Backend-Warteschlangen Anfragen verarbeiten.
- Messen und überwachen Sie die Latenz bei der Verarbeitung von Warteschlangen, indem Sie jedes Mal, wenn Sie eine Nachricht aus einer Warteschlange nehmen, eine CloudWatch-Metrik erstellen, indem Sie die aktuelle Uhrzeit mit dem Nachrichtenzeitstempel vergleichen.
- Wenn Fehler eine erfolgreiche Nachrichtenverarbeitung verhindern oder der Datenverkehr so stark ansteigt, dass er im Rahmen der Service Level Agreements nicht verarbeitet werden kann, wird älterer oder überschüssiger Datenverkehr in eine Überlaufwarteschlange ausgelagert. So können vorrangig neuere Aufträge verarbeitet werden. Ältere Aufträge werden verarbeitet, sobald Kapazitäten frei werden. Diese Technik ist eine Annäherung an die LIFO-Verarbeitung und ermöglicht eine normale Systemverarbeitung für alle neuen Aufträge.
- Verwenden Sie Warteschlangen für unzustellbare Nachrichten oder Redrive-Warteschlangen, um Nachrichten, die nicht verarbeitet werden können, aus dem Backlog an einen Ort zu verschieben, der später geprüft und verarbeitet werden kann.
- Versuchen Sie es entweder erneut oder, sofern dies tolerierbar ist, löschen Sie alte Nachrichten, indem Sie die tatsächliche Zeit mit dem Nachrichtenzeitstempel vergleichen und Nachrichten verwerfen, die für den anfragenden Client nicht mehr relevant sind.

Ressourcen

Zugehörige bewährte Methoden:

- REL04-BP02 Implementieren lose gekoppelter Abhängigkeiten
- REL05-BP02 Drosselung von Anfragen
- REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen

- REL06-BP02 Definieren und Berechnen von Metriken (Aggregierung)
- REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System

Zugehörige Dokumente:

- Vermeiden von nicht mehr aufzuholenden Rückständen
- Schnell scheitern
- Wie kann ich einen zunehmenden Rückstand an Nachrichten in meiner Amazon SQS-Warteschlange verhindern?
- Elastic Load Balancing: Zonenverschiebung
- Amazon Route 53 Application Recovery Controller: Routingsteuerung für Traffic-Failover

Zugehörige Beispiele:

- Muster der Unternehmensintegration: Channel für unzustellbare Nachrichten

Zugehörige Videos:

- AWS re:Invent 2022 – Operating highly available Multi-AZ applications (AWS re:Invent 2022 – Betrieb hochverfügbarer Multi-AZ Anwendungen)

Zugehörige Tools:

- Amazon SQS
- Amazon MQ
- AWS IoT Core
- Amazon CloudWatch

REL05-BP05 Festlegen von Client-Zeitüberschreitungen

Legen Sie angemessene Zeitüberschreitungen für Verbindungen und Anfragen fest, überprüfen Sie sie systematisch und verlassen Sie sich nicht auf Standardwerte, da sie nicht Workload-spezifisch sind.

Gewünschtes Ergebnis: Client-Zeitüberschreitungen sollten die Kosten für Client, Server und Workload berücksichtigen, die mit dem Warten auf Anfragen verbunden sind, deren Bearbeitung ungewöhnlich lange dauert. Da es nicht möglich ist, die genaue Ursache einer Zeitüberschreitung zu ermitteln, müssen Clients ihr Wissen über Services nutzen, um Erwartungen hinsichtlich wahrscheinlicher Ursachen und geeigneter Zeitüberschreitungen zu entwickeln.

Bei Client-Verbindungen kommt es aufgrund der konfigurierten Werte zu einer Zeitüberschreitung. Nach einer Zeitüberschreitung entscheidet der Client entweder, die Anfrage abzubrechen und es erneut zu versuchen oder er öffnet einen Unterbrecher. Durch diese Muster wird vermieden, dass Anfragen gestellt werden, die einen zugrunde liegenden Fehlerzustand verschlimmern könnten.

Typische Anti-Muster:

- Systemzeitüberschreitungen oder standardmäßige Zeitüberschreitungen werden nicht beachtet.
- Normale Abschlusszeit für Anfragen ist nicht bekannt.
- Mögliche Ursachen, warum die Bearbeitung von Anfragen ungewöhnlich lange dauert, oder die Kosten für die Client-, Service- oder Workload-Leistung, die während des Wartens darauf, dass diese Anfragen abgeschlossen werden, anfallen, sind nicht bekannt.
- Die Wahrscheinlichkeit, dass ein gestörtes Netzwerk dazu führt, dass eine Anfrage erst dann fehlschlägt, wenn die Zeitüberschreitung erreicht ist, und die Kosten für die Client- und Workload-Leistung, die entstehen, wenn keine kürzere Zeitüberschreitung gewählt wird, sind nicht bekannt.
- Zeitüberschreitungsszenarien sowohl für Verbindungen als auch für Anfragen werden nicht getestet.
- Zu hohe Zeitüberschreitungen können zu langen Wartezeiten führen und die Ressourcenauslastung erhöhen.
- Zu niedrige Zeitüberschreitungen führen zu künstlichen Fehlschlägen.
- Muster zur Behandlung von Zeitüberschreitungsfehlern bei Remote-Aufrufen wie Unterbrecher und Wiederholungsversuchen werden übersehen.
- Die Überwachung der Fehlerraten bei Serviceaufrufen, der Service-Level-Ziele für die Latenz und der Latenzausreißer wird nicht in Betracht gezogen. Diese Metriken können Aufschluss über aggressive oder tolerante Zeitüberschreitungen geben.

Vorteile der Nutzung dieser bewährten Methode: Zeitüberschreitungen für Remote-Aufrufe sind konfiguriert und die Systeme sind so konzipiert, dass sie Zeitüberschreitungen ordnungsgemäß behandeln, sodass Ressourcen geschont werden, wenn Remote-Aufrufe ungewöhnlich langsam reagieren und Zeitüberschreitungsfehler von Service-Clients ordnungsgemäß behandelt werden.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Legen Sie eine Zeitüberschreitung für Verbindungen sowie Anfragen für alle Serviceabhängigkeitsaufrufe und generell für prozessübergreifende Aufrufe fest. Viele Frameworks bieten integrierte Zeitüberschreitungsfunktionen. Seien Sie jedoch vorsichtig, da einige Standardwerte unendlich oder höher als für Ihre Serviceziele akzeptabel sind. Ein zu hoher Wert reduziert die Nützlichkeit der Zeitbeschränkung, da Ressourcen weiterhin verbraucht werden, während der Client auf das Einsetzen der Zeitbeschränkung wartet. Ein zu niedriger Wert kann zu erhöhtem Datenverkehr im Backend und zu erhöhter Latenz führen, da zu viele Anfragen wiederholt werden. In einigen Fällen kann dies zu vollständigen Ausfällen führen, da alle Anfragen wiederholt werden.

Beachten Sie bei der Festlegung von Zeitüberschreitungsstrategien Folgendes:

- Die Bearbeitung von Anfragen kann aufgrund ihres Inhalts, Beeinträchtigungen eines Zieldienstes oder eines Ausfalls einer Netzwerkpartition länger als normal dauern.
- Anfragen mit ungewöhnlich aufwändigem Inhalt könnten unnötige Server- und Client-Ressourcen verbrauchen. In diesem Fall können Ressourcen geschont werden, wenn für diese Anfragen eine Zeitüberschreitung konfiguriert wird und es nicht erneut versucht wird. Services sollten sich auch durch Drosselungen und serverseitige Zeitüberschreitungen vor ungewöhnlich aufwändigen Inhalten schützen.
- Anfragen, die aufgrund einer Servicebeeinträchtigung ungewöhnlich lange dauern, können mit einer Zeitüberschreitung abgebrochen und erneut versucht werden. Die Servicekosten für die Anfrage und den erneuten Versuch sollten berücksichtigt werden. Wenn die Ursache jedoch eine lokale Beeinträchtigung ist, ist ein erneuter Versuch wahrscheinlich nicht teuer und reduziert den Ressourcenverbrauch des Clients. Die Zeitüberschreitung kann je nach Art der Beeinträchtigung auch Serverressourcen freisetzen.
- Anfragen, deren Bearbeitung lange dauert, weil die Anfrage oder Antwort nicht vom Netzwerk zugestellt wurde, können mit einer Zeitüberschreitung abgebrochen und erneut versucht werden. Da die Anfrage oder Antwort nicht zugestellt wurde, würde sie unabhängig von der Länge der Zeitüberschreitung fehlschlagen. Durch eine Zeitüberschreitung werden in diesem Fall keine Serverressourcen, aber Client-Ressourcen freigegeben und die Workload-Leistung wird verbessert.

Nutzen Sie bewährte Entwurfsmuster wie erneute Versuche und Unterbrecher, um Zeitüberschreitungen problemlos zu behandeln und Ansätze für schnelles Scheitern zu unterstützen.

AWS SDKs und AWS CLI ermöglichen die Konfiguration von Zeitüberschreitungen sowohl für Verbindungen als auch für Anfragen sowie für erneute Versuche mit exponentiellem Backoff und Jitter. AWS Lambda -Funktionen unterstützen die Konfiguration von Zeitüberschreitungen. Mit AWS Step Functionskönnen Sie Low-Code-Unterbrecher erstellen, die die Vorteile vorgefertigter Integrationen mit AWS-Services und SDKs nutzen. AWS App Mesh Envoy bietet Funktionen für Zeitüberschreitungen und Unterbrecher an.

Implementierungsschritte

- Konfigurieren Sie Zeitüberschreitungen für Remote-Serviceaufrufe und nutzen Sie die integrierten sprachspezifischen Zeitüberschreitungsfunktionen oder Open-Source-Bibliotheken für Zeitüberschreitungen.
- Wenn Ihr Workload Anrufe mit einem AWS SDK tätigt, finden Sie in der Dokumentation die sprachspezifische Zeitüberschreitungskonfiguration.
  - Python
  - PHP
  - .NET
  - Ruby
  - Java
  - Go
  - Node.js
  - C++
- Wenn Sie AWS SDKs oder AWS CLI-Befehle in Ihrem Workload verwenden, konfigurieren Sie die Standardwerte für Zeitüberschreitungen durch Festlegen der AWS -Standardeinstellungen für die Konfiguration für `connectTimeoutInMillis` und `tlsNegotiationTimeoutInMillis`.
- Wenden Sie die Befehlszeilenoptionen `cli-connect-timeout` und `cli-read-timeout` an, um einmalige AWS CLI-Befehle an AWS-Services zu steuern.
- Überwachen Sie Remote-Serviceanfragen auf Zeitüberschreitungen und richten Sie Alarme für anhaltende Fehler ein, sodass Sie proaktiv mit Fehlerszenarien umgehen können.
- Implementieren Sie CloudWatch-Metriken und CloudWatch-Erkennung von Unregelmäßigkeiten für Aufruffehlerraten, Service-Level-Ziele für Latenz und Latenzausreißer, um Einblicke in den Umgang mit zu aggressiven oder toleranten Zeitüberschreitungen zu erhalten.
- Konfigurieren Sie Zeitüberschreitungen für Lambda-Funktionen.

- API Gateway-Clients müssen bei der Verarbeitung von Zeitüberschreitungen eigene erneute Versuche implementieren. API Gateway unterstützt eine Integrationszeitüberschreitung zwischen 50 Millisekunden und 29 Sekunden für Downstream-Integrationen und versucht es nicht erneut, wenn bei Integrationsanfragen Zeitüberschreitungen auftreten.
- Implementieren Sie das Unterbrecher -Muster, um zu vermeiden, dass Remote-Aufrufe getätigt werden, wenn Zeitüberschreitungen auftreten. Öffnen Sie die Leitung, um fehlschlagende Aufrufe zu vermeiden, und schließen Sie die Leitung, wenn die Aufrufe normal reagieren.
- Für containerbasierte Workloads können Sie die Funktionen von App Mesh Envoy nutzen, um von den integrierten Zeitüberschreitungen und Unterbrechern zu profitieren.
- Verwenden Sie AWS Step Functions, um Low-Code-Unterbrecher für Remote-Serviceaufrufe zu erstellen, insbesondere beim Aufrufen nativer AWS SDKs und unterstützter Step Functions-Integrationen, um Ihren Workload zu vereinfachen.

Ressourcen

Zugehörige bewährte Methoden:

- REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen
- REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen
- REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System

Zugehörige Dokumente:

- AWS SDK: Wiederholungen und Zeitüberschreitungen
- Die Amazon Builders' Library: Timeouts, Wiederholungen und Backoff mit Jitter
- Amazon API Gateway-Kontingente und wichtige Hinweise
- AWS Command Line Interface: Befehlszeilenoptionen
- AWS SDK for Java 2.x: Konfigurieren von API-Timeouts
- AWS Botocore mit dem Konfigurationsobjekt und der Konfigurationsreferenz
- AWS SDK für .NET: Wiederholungen und Zeitüberschreitungen
- AWS Lambda: Konfigurieren von Lambda-Funktionsoptionen

Zugehörige Beispiele:

- Verwenden des Unterbrechermusters mit AWS Step Functions und Amazon DynamoDB

- Martin Fowler: CircuitBreaker

Zugehörige Tools:

- AWS SDKs
- AWS Lambda
- Amazon SQS
- AWS Step Functions
- AWS Command Line Interface

REL05-BP06 Erstellen zustandsloser Systeme

Systeme sollten entweder keinen Zustand erfordern oder ihn so auslagern, dass zwischen verschiedenen Client-Anfragen keine Abhängigkeit von lokal gespeicherten Daten auf der Festplatte und im Arbeitsspeicher besteht. Auf diese Weise können Server nach Belieben ersetzt werden, ohne dass dies Auswirkungen auf die Verfügbarkeit hat.

Wenn Benutzer oder Services mit einer Anwendung interagieren, führen sie häufig eine Reihe von Interaktionen aus, die eine Sitzung bilden. Bei einer Sitzung handelt es sich um eindeutige Daten für Benutzer, die zwischen Anfragen bestehen bleiben, während sie die Anwendung verwenden. Eine zustandslose Anwendung ist eine Anwendung, die keine Informationen zu früheren Interaktionen benötigt und keine Sitzungsinformationen speichert.

Bei Anwendungen mit zustandslosem Design können Sie Serverless-Computing-Services wie AWS Lambda oder AWS Fargate verwenden.

Neben dem Serverersatz besteht ein weiterer Vorteil zustandsloser Anwendungen darin, dass sie horizontal skaliert werden können, da alle verfügbaren Computing-Ressourcen (z. B. EC2 Instances und AWS Lambda-Funktionen) jede Anfrage bearbeiten können.

Vorteile der Einführung dieser bewährten Methode: Systeme mit zustandslosem Design lassen sich besser an die horizontale Skalierung anpassen, sodass Kapazitäten je nach vorhandenem Datenverkehr und bestehender Nachfrage hinzugefügt oder entfernt werden können. Sie sind auch inhärent widerstandsfähig gegenüber Ausfällen und bieten Flexibilität und Agilität bei der Anwendungsentwicklung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Erstellen Sie zustandslose Anwendungen. Zustandslose Anwendungen ermöglichen eine horizontale Skalierung und sind widerstandsfähig gegenüber Ausfällen einzelner Knoten. Analysieren Sie die Komponenten Ihrer Anwendung, die ihren Status innerhalb der Architektur beibehalten. Auf diese Weise können Sie die potenziellen Auswirkungen der Umstellung auf ein zustandsloses Design bewerten. Eine zustandslose Architektur entkoppelt Benutzerdaten und entlädt die Sitzungsdaten. Dies bietet die Flexibilität, jede Komponente unabhängig zu skalieren, um unterschiedlichen Workload-Anforderungen gerecht zu werden und die Ressourcenauslastung zu optimieren.

Implementierungsschritte

- Identifizieren und analysieren Sie die zustandsbehafteten Komponenten in Ihrer Anwendung.
- Entkoppeln Sie Daten, indem Sie Benutzerdaten von der Kernanwendungslogik trennen und verwalten.
  - Amazon Cognito kann Benutzerdaten mithilfe von Features wie Identitätspools, Benutzerpools und Amazon Cognito vom Anwendungscode entkoppeln.
  - Sie können AWS Secrets Manager verwenden, um Benutzerdaten zu entkoppeln, indem Sie Secrets an einem sicheren, zentralen Ort speichern. Das bedeutet, dass der Anwendungscode keine Secrets speichern muss, was seine Sicherheit erhöht.
  - Erwägen Sie die Verwendung von Amazon S3, um große, unstrukturierte Daten wie Bilder und Dokumente zu speichern. Ihre Anwendung kann diese Daten bei Bedarf abrufen, sodass sie nicht im Arbeitsspeicher gespeichert werden müssen.
  - Verwenden Sie Amazon DynamoDB, um Informationen wie Benutzerprofile zu speichern. Ihre Anwendung kann diese Daten nahezu in Echtzeit abfragen.
- Verlagern Sie Sitzungsdaten in eine Datenbank, einen Cache oder externe Dateien.
  - Amazon ElastiCache, Amazon DynamoDB, Amazon Elastic File System (Amazon EFS) und Amazon MemoryDB sind Beispiele für AWS-Services, mit denen Sie Sitzungsdaten auslagern können.
- Entwerfen Sie eine zustandslose Architektur, nachdem Sie festgelegt haben, welche Zustands- und Benutzerdaten in Ihrer bevorzugten Speicherlösung abgelegt werden müssen.

Ressourcen

Zugehörige bewährte Methoden:

- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen

Zugehörige Dokumente:

- Die Amazon Builders' Library: Vermeiden von Fallback in verteilten Systemen
- Die Amazon Builders' Library: Vermeiden von nicht mehr aufholbaren Warteschlangen-Rückständen
- Die Amazon Builders' Library: Herausforderungen und Strategien für das Caching
- Bewährte Methoden für die zustandslose Webebene auf AWS

REL05-BP07 Implementieren von Nothebeln

Nothebel sind schnelle Prozesse, die die Auswirkungen auf die Verfügbarkeit Ihres Workloads mindern können.

Nothebel bewirken, dass das Verhalten von Komponenten oder Abhängigkeiten mithilfe bekannter und getesteter Mechanismen deaktiviert, gedrosselt oder geändert wird. Dadurch können Beeinträchtigungen des Workloads, die durch die Erschöpfung von Ressourcen aufgrund unerwarteter Nachfragesteigerungen verursacht werden, gemildert und die Auswirkungen von Ausfällen bei nicht kritischen Komponenten innerhalb Ihres Workloads reduziert werden.

Gewünschtes Ergebnis: Durch die Implementierung von Nothebeln können Sie bewährte Prozesse einrichten, um die Verfügbarkeit kritischer Komponenten in Ihrem Workload aufrechtzuerhalten. Der Workload sollte sich problemlos reduzieren lassen und auch während der Aktivierung eines Nothebels weiterhin seine geschäftskritischen Funktionen ausführen. Weitere Informationen über die ordnungsgemäße Funktionsminderung finden Sie unter REL05-BP01 Implementieren einer ordnungsgemäßen Funktionsminderung, um harte Abhängigkeiten in weiche zu ändern.

Typische Anti-Muster:

- Der Ausfall von nicht kritischen Abhängigkeiten wirkt sich auf die Verfügbarkeit Ihres Kern-Workloads aus.
- Das Verhalten kritischer Komponenten wird während der Beeinträchtigung unkritischer Komponenten nicht getestet oder überprüft.
- Es sind keine klaren und deterministischen Kriterien für die Aktivierung oder Deaktivierung eines Nothebels definiert.

Vorteile der Nutzung dieser bewährten Methode: Die Implementierung von Nothebeln kann die Verfügbarkeit der kritischen Komponenten Ihres Workloads verbessern, indem Ihre Resolver mit

bewährten Prozessen ausgestattet werden, um auf unerwartete Nachfragespitzen oder Ausfälle von nicht kritischen Abhängigkeiten zu reagieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

- Ermitteln Sie die kritischen Komponenten in Ihrem Workload.
- Entwerfen und gestalten Sie die kritischen Komponenten Ihres Workloads so, dass sie Ausfällen von nicht kritischen Komponenten standhalten.
- Führen Sie Tests durch, um das Verhalten Ihrer kritischen Komponenten beim Ausfall von nicht kritischen Komponenten zu überprüfen.
- Definieren und überwachen Sie relevante Metriken oder Auslöser für die Einleitung von Nothebeln.
- Definieren Sie die Verfahren (manuell oder automatisiert), die Bestandteil des Nothebels sind.

Implementierungsschritte

- Ermitteln Sie die kritischen Komponenten in Ihrem Workload.
  - Jede technische Komponente Ihres Workloads sollte der entsprechenden Geschäftsfunktion zugeordnet und als kritisch oder nicht kritisch eingestuft werden. Beispiele für wichtige und unkritische Funktionen bei Amazon finden Sie unter Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second (Jeder Tag kann ein Prime Day sein: Wie die Amazon.com-Suche mit Hilfe von Chaos Engineering über 84.000 Anfragen pro Sekunde bewältigt).
  - Hierbei handelt es sich sowohl um eine technische als auch um eine geschäftliche Entscheidung, die je nach Organisation und Workload unterschiedlich ausfallen kann.
- Entwerfen und gestalten Sie die kritischen Komponenten Ihres Workloads so, dass sie Ausfällen von nicht kritischen Komponenten standhalten.
  - Berücksichtigen Sie bei der Abhängigkeitsanalyse alle potenziellen Fehlermodi und stellen Sie sicher, dass Ihre Notfallmechanismen die kritischen Funktionen an nachgelagerte Komponenten weitergeben.
- Führen Sie Tests durch, um das Verhalten Ihrer kritischen Komponenten bei der Aktivierung Ihrer Nothebel zu überprüfen.
  - Vermeiden Sie bimodales Verhalten. Weitere Informationen finden Sie unter REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität.

- Definieren und überwachen Sie relevante Metriken und lassen Sie gegebenenfalls einen Alarm auslösen, um einen Nothebel einzuleiten.
  - Die richtigen Metriken zur Überwachung zu finden, hängt von Ihrem Workload ab. Einige Beispielmetriken sind die Latenzzeit oder die Anzahl der fehlgeschlagenen Anfragen an eine Abhängigkeit.
- Definieren Sie die manuellen oder automatisierten Verfahren, die Bestandteil des Nothebels sind.
  - Dazu können Mechanismen wie Lastabwurf, Drosselung von Anfragen oder die Implementierung einer ordnungsgemäßen Funktionsminderung gehören.

Ressourcen

Zugehörige bewährte Methoden:

- REL05-BP01 Implementieren einer ordnungsgemäßen Funktionsminderung, um harte Abhängigkeiten in weiche zu ändern
- REL05-BP02 Drosselung von Anfragen
- REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität

Zugehörige Dokumente:

- Automating safe, hands-off deployments (Automatisierung sicherer, vollautomatischer Bereitstellungen)
- Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second (Jeder Tag kann ein Prime Day sein: Wie die Amazon.com-Suche mit Hilfe von Chaos Engineering über 84.000 Anfragen pro Sekunde bewältigt)

Zugehörige Videos:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability (AWS re:Invent 2020: Zuverlässlichkeit, Konsistenz und Vertrauen durch Unveränderlichkeit)

Änderungsverwaltung

Fragen

- REL 6. Was ist bei der Überwachung von Workload-Ressourcen zu beachten?

- REL 7 Wie lässt sich die Workload so gestalten, dass sie sich an Bedarfsänderungen anpasst?
- REL 8. Wie implementieren Sie Änderungen?

REL 6. Was ist bei der Überwachung von Workload-Ressourcen zu beachten?

Protokolle und Metriken sind wertvolle Tools, um einen Einblick in den Zustand Ihrer Workloads zu gewinnen. Sie können Ihre Workload so konfigurieren, dass Protokolle und Metriken überwacht und bei Über- oder Unterschreiten von Schwellenwerten oder wichtigen Ereignissen Benachrichtigungen gesendet werden. Dank der Überwachung kann die Workload erkennen, wenn Schwellenwerte für eine niedrige Leistung unterschritten werden oder Ausfälle auftreten, sodass als Reaktion drauf eine automatische Wiederherstellung erfolgen kann.

Bewährte Methoden

- REL06-BP01 Überwachen aller Komponenten der Workload (Generierung)
- REL06-BP02 Definieren und Berechnen von Metriken (Aggregierung)
- REL06-BP03 Senden von Benachrichtigungen (Verarbeitung und Benachrichtigung in Echtzeit)
- REL06-BP04 Automatisieren von Antworten (Verarbeitung und Benachrichtigung in Echtzeit)
- REL06-BP05 Analysen
- REL06-BP06 Regelmäßiges Durchführen von Prüfungen
- REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System

REL06-BP01 Überwachen aller Komponenten der Workload (Generierung)

Überwachen Sie die Komponenten der Workload mit Amazon CloudWatch oder Tools von Drittanbietern. Überwachen Sie AWS-Services mit dem AWS Health Dashboard.

Alle Komponenten Ihrer Workload sollten überwacht werden, einschließlich Frontend, Geschäftslogik und Speicherstufen. Definieren Sie Schlüsselmetriken, beschreiben Sie, wie Sie diese gegebenenfalls aus Protokollen extrahieren, und legen Sie Schwellenwerte für das Auslösen entsprechender Alarmereignisse fest. Stellen Sie sicher, dass die Metriken für die wichtigen Leistungskennzahlen (KPIs) Ihrer Workload relevant sind und verwenden Sie Metriken und Protokolle, um frühe Warnzeichen einer Serviceverschlechterung zu identifizieren. Beispielsweise kann eine mit Geschäftsergebnissen zusammenhängende Metrik wie etwa die Anzahl der pro Minute erfolgreich verarbeiteten Bestellungen schneller auf Workload-Probleme hinweisen als eine technische Metrik wie etwa die CPU-Auslastung. Verwenden Sie das AWS Health Dashboard für

eine personalisierte Ansicht der Leistung und Verfügbarkeit der AWS-Services, die Ihren AWS-Ressourcen zugrunde liegen.

Die Überwachung in der Cloud bietet neue Möglichkeiten. Die meisten Cloudanbieter haben anpassbare Hooks entwickelt und können Einblicke liefern, mit denen Sie mehrere Ebenen Ihrer Workload überwachen können. AWS-Services wie Amazon CloudWatch wenden statistische und Machine-Learning-Algorithmen an, um Metriken von Systemen und Anwendungen kontinuierlich zu analysieren, normale Basiswerte zu erkennen und Oberflächenanomalien anhand eines minimalen Benutzereingriffs aufzudecken. Algorithmen zur Erkennung von Anomalien berücksichtigen saisonale Schwankungen und Trendänderungen von Metriken.

AWS stellt zahlreiche Überwachungs- und Protokollinformationen bereit, die genutzt werden können, um workload-spezifische Metriken und Bedarfsänderungsprozesse zu definieren und Machine-Learning-Verfahren unabhängig von der ML-Erfahrung einzuführen.

Zudem können Sie auch all Ihre externen Endpunkte überwachen, um sicherzustellen, dass diese von Ihrer Basisimplementierung unabhängig sind. Diese aktive Überwachung kann anhand von synthetischen Transaktionen erfolgen (auch Benutzer-Canariesgenannt, jedoch nicht zu verwechseln mit Canary-Bereitstellungen). Diese führen regelmäßig eine Reihe gängiger Aufgaben aus, die mit Aktionen übereinstimmen, die von Clients der Workload durchgeführt werden. Diese Aufgaben sollten nicht zu lang sein und Sie sollten darauf achten, Ihre Workload beim Testen nicht zu überlasten. Mit Amazon CloudWatch Synthetics können Sie synthetische Canaries erstellen, um Ihre Endpunkte und APIs zu überwachen. Sie können die synthetischen Canary-Client-Knoten auch mit der AWS X-Ray-Konsole kombinieren, um zu bestimmen, bei welchen synthetischen Canaries im ausgewählten Zeitraum Probleme mit Fehlern, Störungen oder Drosselungsraten auftreten.

Gewünschtes Ergebnis:

Erfassen und Nutzen kritischer Metriken aus allen Komponenten der Workload, um die Workload-Zuverlässigkeit und eine optimale Benutzererfahrung sicherzustellen. Zu erkennen, dass eine Workload keine Geschäftsergebnisse erzielt, ermöglicht es Ihnen, schnell einen Systemausfall zu deklarieren und das System nach einem Vorfall wiederherzustellen.

Gängige Antimuster:

- Es werden nur externe Schnittstellen zur Workload überwacht.
- Es werden keine workload-spezifischen Metriken erzeugt und Sie verlassen sich nur auf Metriken, die Ihnen von den AWS-Services, die Ihre Workload verwendet, bereitgestellt werden.

- Es werden nur technische Metriken in Ihrer Workload verwendet und es werden keinerlei Metriken im Zusammenhang mit nicht-technischen KPIs, zu denen die Workload beiträgt, überwacht.
- Sie verlassen sich auf den Produktionsdatenverkehr und einfache Zustandsprüfungen für die Überwachung und Bewertung des Workload-Status.

Vorteile der Einführung dieser bewährten Methode: Durch die Überwachung aller Ebenen Ihrer Workload können Sie Probleme in den darin enthaltenen Komponenten schneller vorhersehen und beheben.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

1. Aktivieren Sie die Protokollierung, wann immer verfügbar. Von allen Workload-Komponenten sollten Überwachungsdaten erzielt werden. Aktivieren Sie eine zusätzliche Protokollierung, wie etwa S3 Access Logs, und ermöglichen Sie es Ihrer Workload, die workload-spezifischen Daten zu protokollieren. Erfassen Sie Metriken für die Durchschnittswerte zu CPU, Netzwerk-E/A und Laufwerk-E/A von Services wie Amazon ECS, Amazon EKS, Amazon EC2, Elastic Load Balancing, AWS Auto Scaling und Amazon EMR. Unter AWS-Services, die CloudWatch-Metriken veröffentlichen finden Sie eine Liste an AWS-Services, die Metriken in CloudWatch veröffentlichen.
2. Sehen Sie sich alle Standardmetriken an, um mehr über mögliche Datenerfassungslücken zu erfahren. Jeder Service generiert Standardmetriken. Durch die Erfassung von Standardmetriken erhalten Sie ein besseres Verständnis über die Abhängigkeiten zwischen Workload-Komponenten und darüber, wie die Komponentenzuverlässigkeit und -leistung die Workload beeinträchtigen. Sie können auch Ihre eigenen Metriken in CloudWatch unter Verwendung der AWS CLI oder einer API erstellen und veröffentlichen. Dies
3. Bewerten Sie alle Metriken, um zu entscheiden, für welche eine Warnmeldung für jeden AWS-Service in Ihrer Workload eingerichtet werden soll. Sie können eine Metriken-Untergruppe auswählen, die eine höhere Auswirkung auf die Workload-Zuverlässigkeit hat. Wenn Sie sich auf kritische Metriken und Schwellenwerte konzentrieren, können Sie die Anzahl an Warnmeldungen genauer definieren und so Falschmeldungen reduzieren.
4. Definieren Sie Warnungen und den Wiederherstellungsprozess für Ihre Workload nach dem Auslösen der Warnmeldung. Das Definieren von Warnmeldungen ermöglicht es Ihnen, schnell zu benachrichtigen, zu eskalieren und die für die Wiederherstellung nach einem Vorfall erforderlichen Schritte durchzuführen, um so Ihren festgelegten Recovery Time Objective (RTO) zu erfüllen. Sie können Amazon CloudWatch-Alarme für das Aufrufen von automatisierten Workflows und

die Initiierung von Wiederherstellungsverfahren basierend auf definierten Schwellenwerten verwenden.

5. Erfahren Sie mehr über die Verwendung von synthetischen Transaktionen für das Erfassen relevanter Daten zum Workload-Status. Die synthetische Überwachung folgt denselben Routen und führt dieselben Aktionen aus wie ein Kunde. Dadurch haben Sie die Möglichkeit, die Kundenerfahrung kontinuierlich zu überprüfen, selbst, wenn Sie keinen Kundendatenverkehr auf Ihren Workloads haben. Durch die Verwendung von synthetischen Transaktionenkönnen Sie Probleme erkennen, bevor Ihre Kunden dies tun.

Ressourcen

Relevante bewährte Methoden:

- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen

Relevante Dokumente:

- Getting started with your AWS Health Dashboard – Your account health (Erste Schritte mit Ihrem AWS Health-Dashboard – Der Zustand Ihres Kontos)
- AWS-Services, die CloudWatch-Metriken veröffentlichen
- Zugriffsprotokolle für Ihren Network Load Balancer
- Zugriffsprotokolle für Ihre Application Load Balancer
- Zugriff auf Amazon CloudWatch Logs für AWS Lambda
- Protokollierung von Amazon S3-Serverzugriffen
- Aktivieren Sie Zugriffsprotokolle für Ihren Classic Load Balancer.
- Exportieren von Protokolldaten zu Amazon S3
- Installieren des CloudWatch-Agenten
- Veröffentlichen benutzerdefinierter Metriken
- Verwenden von Amazon CloudWatch-Dashboards
- Verwenden von Amazon CloudWatch-Metriken
- Verwenden von Synthetic Monitoring
- Was sind Amazon CloudWatch Logs?

Benutzerhandbücher:

- Erstellen eines Trails
- Überwachen von Arbeitsspeicher- und Datenträgermetriken für Amazon EC2 Linux-Instances
- Verwenden von CloudWatch Logs mit Container-Instances
- VPC Flow Logs
- Was ist Amazon DevOps Guru?
- Was ist AWS X-Ray?

Ähnliche Blogs:

- Debugging mit Amazon CloudWatch Synthetics und AWS X-Ray

Ähnliche Beispiele und Workshops:

- AWS Well-Architected Labs: Operational Excellence - Dependency Monitoring (AWS Well-Architected Labs: Operative Exzellenz – Überwachung von Abhängigkeiten)
- Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen
- Workshop zur Beobachtbarkeit

REL06-BP02 Definieren und Berechnen von Metriken (Aggregierung)

Speichern Sie Protokolldaten und wenden Sie gegebenenfalls Filter an, um Metriken zu berechnen. Dazu gehören z. B. die Anzahl eines bestimmten Protokollereignisses oder die Latenz, die aus den Zeitstempeln des Protokollereignisses berechnet wird.

Amazon CloudWatch und Amazon S3 dienen als primäre Aggregierungs- und Speicherebenen. Bei einigen Services wie AWS Auto Scaling und Elastic Load Balancing werden Standardkennzahlen für die CPU-Last oder die durchschnittliche Anfragelatenz eines Clusters oder einer Instance bereitgestellt. Für Streaming-Services wie VPC Flow Logs und AWS CloudTrail werden Ereignisdaten an CloudWatch Logs weitergeleitet und Sie müssen Filter definieren und anwenden, um Metriken aus diesen Ereignisdaten zu extrahieren. Auf diese Weise erhalten Sie Zeitreihendaten, die als Eingaben für CloudWatch-Alarme dienen können, die Sie zum Auslösen von Warnungen definieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Definieren und berechnen Sie Metriken (Aggregierung). Speichern Sie Protokolldaten und wenden Sie gegebenenfalls Filter an, um Metriken zu berechnen. Dazu gehören z. B. die Anzahl eines bestimmten Protokollereignisses oder die Latenz, die aus den Zeitstempeln des Protokollereignisses berechnet wird.
  - Metrikfilter definieren die Begriffe und Muster, die in Protokolldaten zu suchen sind, wenn diese an CloudWatch Logs gesendet werden. CloudWatch Logs verwendet diese Metrikfilter, um Protokolldaten in numerische CloudWatch-Metriken umzuwandeln, die Sie grafisch darstellen oder für die Sie einen Alarm einrichten können.
    - Suchen und Filtern von Protokolldaten
  - Verwenden Sie einen vertrauenswürdigen Drittanbieter für die Protokollaggregierung.
    - Befolgen Sie die Anweisungen des Drittanbieters. Die meisten Produkte von Drittanbietern lassen sich in CloudWatch und Amazon S3 integrieren.
  - Einige AWS-Services können Protokolle direkt in Amazon S3 veröffentlichen. Wenn die Speicherung von Protokollen in Amazon S3 die wichtigste Anforderung ist, kann der Protokoll-Service die Protokolle direkt an Amazon S3 senden, ohne dass eine zusätzliche Infrastruktur eingerichtet werden muss.
    - Senden von Protokollen direkt an Amazon S3

Ressourcen

Relevante Dokumente:

- Amazon CloudWatch Logs Insights-Beispielabfragen
- Debugging mit Amazon CloudWatch Synthetics und AWS X-Ray
- Workshop zur Beobachtbarkeit
- Suchen und Filtern von Protokolldaten
- Senden von Protokollen direkt an Amazon S3
- Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen

REL06-BP03 Senden von Benachrichtigungen (Verarbeitung und Benachrichtigung in Echtzeit)

Wenn Organisationen potenzielle Probleme erkennen, senden sie Benachrichtigungen und Warnungen in Echtzeit an das entsprechende Personal und die entsprechenden Systeme, um schnell und effektiv auf diese Probleme reagieren zu können.

Gewünschtes Ergebnis: Durch die Konfiguration relevanter Alarme auf der Grundlage von Service- und Anwendungsmetriken ist eine schnelle Reaktion auf operative Ereignisse möglich. Bei Überschreitung der Alarmschwellen werden das entsprechende Personal und die entsprechenden Systeme benachrichtigt, damit sie die zugrunde liegenden Probleme beseitigen können.

Typische Anti-Muster:

- Sie konfigurieren Alarme mit einem übermäßig hohen Schwellenwert, was dazu führt, dass wichtige Benachrichtigungen nicht gesendet werden können.
- Sie konfigurieren Alarme mit einem zu niedrigen Schwellenwert, was dazu führt, dass bei wichtigen Warnungen aufgrund des Lärms übermäßiger Benachrichtigungen keine Aktion erfolgt.
- Sie aktualisieren keine Alarme und ihre Schwellenwerte, wenn sich die Nutzung ändert.
- Bei Alarmen, die am besten durch automatische Aktionen behoben werden, führt das Senden der Benachrichtigung an das Personal, anstatt die automatische Aktion zu generieren, dazu, dass übermäßig viele Benachrichtigungen gesendet werden.

Vorteile der Nutzung dieser bewährten Methode: Das Senden von Benachrichtigungen und Warnungen in Echtzeit an das entsprechende Personal und die entsprechenden Systeme ermöglicht eine frühzeitige Erkennung von Problemen und eine schnelle Reaktion auf betriebliche Vorfälle.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Workloads sollten mit der Verarbeitung und Benachrichtigung in Echtzeit ausgestattet sein, um die Erkennbarkeit von Problemen zu verbessern, die sich auf die Verfügbarkeit der Anwendung auswirken und als Auslöser für automatische Reaktionen dienen könnten. Organisationen können die Verarbeitung und Benachrichtigung in Echtzeit durchführen, indem sie Warnungen mit definierten Metriken erstellen, um Benachrichtigungen zu erhalten, wenn wichtige Ereignisse eintreten oder eine Metrik einen Schwellenwert überschreitet.

Amazon CloudWatch ermöglicht es Ihnen, Metrik-Alarme und zusammengesetzte Alarme mithilfe von CloudWatch-Alarmen zu erstellen, die auf statischen Schwellenwerten, der Erkennung von

Unregelmäßigkeiten und anderen Kriterien basieren. Weitere Informationen zu den Alarmtypen, die Sie mit CloudWatch konfigurieren können, finden Sie im Abschnitt über Alarme in der CloudWatch-Dokumentation.

Sie können benutzerdefinierte Ansichten von Metriken und Warnungen Ihrer AWS-Ressourcen für Ihre Teams erstellen, indem Sie CloudWatch-Dashboards nutzen. Die anpassbaren Startseiten in der CloudWatch-Konsole ermöglichen es Ihnen, Ihre Ressourcen in einer einzigen Ansicht über mehrere Regionen hinweg zu überwachen.

Alarme können mindestens eine Aktion ausführen, z. B. das Senden einer Benachrichtigung an ein Amazon SNS-Thema, das eine Amazon EC2- Aktion oder eine Amazon EC2 Auto Scaling- Aktion durchführt oder ein OpsItem-Element oder einen Vorfall in AWS Systems Manager erstellen.

Amazon CloudWatch verwendet Amazon SNS zum Senden von Benachrichtigungen, wenn sich der Status des Alarms ändert, und ermöglicht so die Nachrichtenzustellung von den Publishern (Produzenten) an die Subscriber (Verbraucher). Weitere Informationen zum Einrichten von Amazon SNS-Benachrichtigungen finden Sie unter Konfigurieren von Amazon SNS.

CloudWatch sendet EventBridge- Ereignisse, wenn ein CloudWatch-Alarm erstellt, aktualisiert oder gelöscht wird oder sich sein Status ändert. Sie können EventBridge mit diesen Ereignissen verwenden, um Regeln zu erstellen, die Aktionen ausführen, z. B. Sie benachrichtigen, wenn sich der Status eines Alarms ändert, oder automatisch Ereignisse in Ihrem Konto mit Systems Manager-Automatisierung auslösen.

Wann sollten Sie EventBridge im Vergleich zu Amazon SNS verwenden?

Sowohl EventBridge als auch Amazon SNS können zur Entwicklung ereignisgesteuerter Anwendungen verwendet werden. Ihre Wahl hängt von Ihren spezifischen Anforderungen ab.

Amazon EventBridge wird empfohlen, wenn Sie eine Anwendung erstellen möchten, die auf Ereignisse aus Ihren eigenen Anwendungen, SaaS-Anwendungen und AWS-Services reagiert. EventBridge ist der einzige ereignisbasierte Service, der direkt in SaaS-Partner von Drittanbietern integriert werden kann. EventBridge nimmt außerdem automatisch Ereignisse von über 200 AWS-Services auf, ohne dass Entwickler Ressourcen in ihrem Konto erstellen müssen.

EventBridge verwendet eine definierte JSON-basierte Struktur für Ereignisse und hilft Ihnen bei der Erstellung von Regeln, die auf den gesamten Ereignistext angewendet werden, um Ereignisse auszuwählen, die an ein Ziel weitergeleitet werden sollen. EventBridge unterstützt derzeit über 20 AWS-Services als Ziele, darunter AWS Lambda, Amazon SQS, Amazon SNS, Amazon Kinesis Data Streamsund Amazon Data Firehose.

Amazon SNS wird für Anwendungen empfohlen, die eine hohe Verteilung benötigen (Tausende oder Millionen von Endpunkten). Ein gängiges Muster, das wir beobachten, ist, dass Kunden Amazon SNS als Ziel für ihre Regel verwenden, um die Ereignisse zu filtern, die sie benötigen, und dann an mehrere Endpunkte zu verteilen.

Nachrichten sind unstrukturiert und können in jedem Format vorliegen. Amazon SNS unterstützt die Weiterleitung von Nachrichten an sechs verschiedene Zieltypen, darunter Lambda, Amazon SQS, HTTP/S-Endpunkte, SMS, mobile Push-Benachrichtigungen und E-Mail. Amazon SNS Die typische Latenz liegt unter 30 Millisekunden. Eine Vielzahl von AWS-Services sendet Amazon SNS-Nachrichten, indem sie den Service entsprechend konfigurieren (mehr als 30, einschließlich Amazon EC2, Amazon S3und Amazon RDS).

Implementierungsschritte

1. Erstellen Sie einen Alarm mithilfe von Amazon CloudWatch-Alarmen.
   a. Ein metrischer Alarm überwacht eine einzelne CloudWatch-Metrik oder einen Ausdruck, der von CloudWatch-Metriken abhängig ist. Der Alarm initiiert eine oder mehrere Aktionen auf der Grundlage des Werts der Metrik oder des Ausdrucks im Vergleich zu einem Schwellenwert über eine Reihe von Zeitintervallen. Die Aktion kann darin bestehen, eine Benachrichtigung an ein Amazon SNS-Themazu senden, das eine Amazon EC2- Aktion oder eine Amazon EC2 Auto Scaling- Aktion durchführt oder ein OpsItem-Element oder einen Vorfall in AWS Systems Manager zu erstellen.
   b. Ein zusammengesetzter Alarm besteht aus einem Regelausdruck, der die Alarmbedingungen anderer von Ihnen erstellter Alarme berücksichtigt. Der zusammengesetzte Alarm wechselt nur dann in den Alarmstatus, wenn alle Regelbedingungen erfüllt sind. Die im Regelausdruck eines zusammengesetzten Alarms angegebenen Alarme können metrische Alarme und zusätzliche zusammengesetzte Alarme enthalten. Zusammengesetzte Alarme können Amazon SNS-Benachrichtigungen senden, wenn sich ihr Status ändert, und sie können Systems Manager OpsItems-Elemente oder Vorfälle auslösen, wenn sie in den Alarmzustand wechseln, aber sie können weder Amazon EC2- noch Auto Scaling-Aktionen ausführen.
2. Richten Sie Amazon SNS-Benachrichtigungen ein. Wenn Sie einen CloudWatch-Alarm erstellen, können Sie ein Amazon SNS-Thema hinzufügen, um eine Benachrichtigung zu senden, wenn sich der Status des Alarms ändert.
3. Erstellen Sie Regeln in EventBridge, die bestimmten CloudWatch-Alarmen entsprechen. Jede Regel unterstützt mehrere Ziele, einschließlich Lambda-Funktionen. Sie können beispielsweise einen Alarm definieren, der initiiert wird, wenn der verfügbare Festplattenspeicher knapp wird, wodurch über eine EventBridge-Regel eine Lambda-Funktion ausgelöst wird, um den

Speicherplatz zu bereinigen. Weitere Informationen zu EventBridge-Zielen finden Sie unter EventBridge-Ziele.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- REL06-BP01 Überwachen aller Komponenten der Workload (Generierung)
- REL06-BP02 Definieren und Berechnen von Metriken (Aggregierung)
- REL12-BP01 Untersuchen von Fehlern mit Playbooks:

Zugehörige Dokumente:

- Amazon CloudWatch
- CloudWatch Logs-Erkenntnisse
- Verwenden von Amazon CloudWatch-Alarmen
- Verwenden von Amazon CloudWatch-Dashboards
- Using Amazon CloudWatch metrics (Verwenden von Amazon CloudWatch-Metriken)
- Einrichtung von Amazon SNS-Benachrichtigungen
- CloudWatch-Erkennung von Unregelmäßigkeiten
- CloudWatch Logs-Datenschutz
- Amazon EventBridge
- Amazon Simple Notification Service

Zugehörige Videos:

- re:Invent 2022 observability videos
- AWS re:Invent 2022 – Observability best practices at Amazon (AWS re:Invent 2022 – Bewährte Überwachungsmethoden bei Amazon)

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit

- Amazon EventBridge bis AWS Lambda mit Feedbacksteuerung durch Amazon CloudWatch-Alarme

REL06-BP04 Automatisieren von Antworten (Verarbeitung und Benachrichtigung in Echtzeit)

Automatisieren Sie bei Erkennung von Ereignissen die erforderlichen Maßnahmen, wie etwa den Austausch fehlerhafter Komponenten.

Die automatische Echtzeitverarbeitung von Alarmen ist implementiert, sodass die Systeme bei Auslösung von Alarmen schnell korrigierend eingreifen und versuchen können, Ausfälle oder Beeinträchtigungen des Services zu verhindern. Zu den automatisierten Reaktionen auf Alarme könnten der Austausch ausgefallener Komponenten, die Anpassung der Rechenkapazität, die Umleitung des Datenverkehrs auf fehlerfreie Hosts, Availability Zones oder andere Regionen sowie die Benachrichtigung der Betreiber gehören.

Gewünschtes Ergebnis: Echtzeitalarme werden ermittelt und die automatische Verarbeitung von Alarmen wird eingerichtet, um die entsprechenden Maßnahmen zur Einhaltung von Service-Level-Zielen und Service Level Agreements (SLAs) einzuleiten. Die Automatisierung kann von der Selbstreparatur einzelner Komponenten bis hin zum Failover eines ganzen Standorts reichen.

Typische Anti-Muster:

- Fehlen einer genauen Bestandsaufnahme oder eines Katalogs der wichtigsten Echtzeitalarme
- Keine automatischen Reaktionen auf kritische Alarme (z. B. automatische Skalierung, wenn die Rechenkapazität fast erschöpft ist)
- Widersprüchliche Alarmreaktionen
- Fehlen von Standard-Betriebsabläufen (SOPs), an die sich die Bediener halten müssen, wenn sie Alarmmeldungen erhalten
- Keine Überwachung von Konfigurationsänderungen, da unentdeckte Konfigurationsänderungen zu Ausfallzeiten bei Workloads führen können
- Keine Strategie, um unbeabsichtigte Konfigurationsänderungen rückgängig zu machen

Vorteile der Nutzung dieser bewährten Methode: Die Automatisierung der Alarmverarbeitung kann die Ausfallsicherheit des Systems verbessern. Das System ergreift automatisch Korrekturmaßnahmen und reduziert so manuelle Tätigkeiten, bei denen es zu einem menschlichen, fehleranfälligen Eingreifen kommen kann. Der Workload-Betrieb erfüllt die Verfügbarkeitsziele und reduziert Serviceunterbrechungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Zur wirksamen Verwaltung von Alarmen und zur Automatisierung ihrer Beantwortung kategorisieren Sie die Alarme nach ihrer Kritikalität und Auswirkung, dokumentieren die Reaktionsverfahren und planen die Reaktionen, bevor Sie die Aufgaben einordnen.

Ermitteln Sie Aufgaben, die bestimmte Aktionen erfordern (oft in Runbooks detailliert beschrieben), und untersuchen Sie alle Runbooks und Playbooks, um festzustellen, welche Aufgaben automatisiert werden können. Lassen sich Aktionen definieren, können sie oft auch automatisiert werden. Wenn Aktionen nicht automatisiert werden können, dokumentieren Sie die manuellen Schritte in einer SOP und schulen Sie die Mitarbeiter darin. Hinterfragen Sie kontinuierlich manuelle Prozesse und suchen Sie nach Möglichkeiten zur Automatisierung, um einen Plan für die Automatisierung von Alarmreaktionen zu erstellen und zu verwalten.

Implementierungsschritte

1. Erstellen eines Inventars von Alarmen: Um eine Liste aller Alarme zu erhalten, können Sie die AWS CLI mit dem Amazon CloudWatch-Befehl `describe-alarms` verwenden. Je nachdem, wie viele Alarme Sie eingerichtet haben, müssen Sie möglicherweise eine Paginierung verwenden, um eine Untergruppe von Alarmen für jeden Anruf aufzurufen. Alternativ können Sie das AWS-SDK verwenden, um die Alarme über einen API-Aufruf aufzurufen.
2. Dokumentieren aller Alarmaktionen: Aktualisieren Sie ein Runbook mit allen Alarmen und ihren Aktionen, unabhängig davon, ob sie manuell oder automatisiert sind. AWS Systems Manager bietet vordefinierte Runbooks. Ausführliche Informationen zum Anzeigen von Runbook-Inhalten finden Sie unter Mit Runbooks arbeiten. Ausführliche Informationen zum Anzeigen von Runbook-Inhalten finden Sie unter Runbook-Inhalt anzeigen.
3. Einrichten und Verwalten von Alarmaktionen: Für jeden der Alarme, die eine Aktion erfordern, geben Sie die automatische Aktion mithilfe des CloudWatch-SDK an. So können Sie beispielsweise den Zustand Ihrer Amazon EC2-Instances automatisch auf Grundlage eines CloudWatch-Alarms ändern, indem Sie Aktionen für einen Alarm erstellen und aktivieren oder Aktionen für einen Alarm deaktivieren.

   Sie können Amazon EventBridge auch verwenden, um automatisch auf Systemereignisse zu reagieren, z. B. auf Probleme mit der Anwendungsverfügbarkeit oder auf Ressourcenänderungen. Sie können Regeln erstellen, um anzugeben, an welchen Ereignissen Sie interessiert sind und welche Aktionen durchgeführt werden sollen, wenn ein Ereignis einer Regel entspricht.

Zu den Aktionen, die automatisch ausgelöst werden können, gehören der Aufruf einer AWS Lambda-Funktion, der Aufruf des Amazon EC2 `Run Command`, die Weiterleitung des Ereignisses an Amazon Kinesis Data Streams und die Anzeige von Automatisieren von Amazon EC2 mit EventBridge.

4. Standard-Betriebsabläufe (SOPs): Basierend auf den Komponenten Ihrer Anwendung empfiehlt AWS Resilience Hub mehrere SOP-Vorlagen. Sie können diese SOPs verwenden, um alle Prozesse zu dokumentieren, die ein Bediener im Falle eines Alarms befolgen sollte. Sie können auch eine SOP auf Grundlage von Resilience Hub-Empfehlungen erstellen, für die Sie eine Resilience Hub-Anwendung mit einer zugehörigen Resilienzrichtlinie sowie eine historische Resilienzbewertung für diese Anwendung benötigen. Die Empfehlungen für Ihre SOP ergeben sich aus der Resilienzbewertung.

   Resilience Hub arbeitet mit Systems Manager zusammen, um die einzelnen Schritte Ihrer SOPs zu automatisieren. Dazu erhalten Sie eine Reihe von SSM-Dokumenten, die Sie als Grundlage für diese SOPs verwenden können. So kann Resilience Hub zum Beispiel eine SOP für das Hinzufügen von Speicherplatz auf Grundlage eines bestehenden SSM-Automatisierungsdokuments empfehlen.
5. Durchführen automatisierter Aktionen mit Amazon DevOps Guru: Sie können Amazon DevOps Guru verwenden, um Anwendungsressourcen automatisch auf anomales Verhalten zu überwachen und gezielte Empfehlungen für eine schnellere Problemerkennung und -behebung zu geben. Mit DevOps Guru können Sie Ströme von Betriebsdaten aus verschiedenen Quellen wie Amazon CloudWatch-Metriken, AWS Config, AWS CloudFormation und AWS X-Ray nahezu in Echtzeit überwachen. Sie können DevOps Guru auch verwenden, um automatisch OpsItems in OpsCenter zu erstellen und Ereignisse an EventBridge zu senden, um eine zusätzliche Automatisierung zu erreichen.

Ressourcen

Zugehörige bewährte Methoden:

- REL06-BP01 Überwachen aller Komponenten der Workload (Generierung)
- REL06-BP02 Definieren und Berechnen von Metriken (Aggregierung)
- REL06-BP03 Senden von Benachrichtigungen (Verarbeitung und Benachrichtigung in Echtzeit)
- REL08-BP01 Verwenden von Runbooks für Standardaktivitäten wie die Bereitstellung

Zugehörige Dokumente:

- AWS Systems Manager-Automatisierung
- Erstellen einer EventBridge-Regel, die durch ein Ereignis aus einer AWS-Ressource ausgelöst wird
- Workshop zur Beobachtbarkeit
- Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen
- Was ist Amazon DevOps Guru?
- Arbeiten mit Automation-Dokumenten (Playbooks)

Zugehörige Videos:

- AWS re:Invent 2022 - Observability best practices at Amazon (AWS re:Invent 2022: Bewährte Überwachungsmethoden bei Amazon)
- AWS re:Invent 2020: Automate anything with AWS Systems Manager(AWS re:Invent 2020: Automatiserung mit AWS Systems Manager)
- Introduction to AWS Resilience Hub(Einführung in AWS Resilience Hub)
- Create Custom Ticket Systems for Amazon DevOps Guru Notifications (Benutzerdefinierte Ticketsysteme für x Benachrichtigungen erstellen Amazon DevOps Guru)
- Enable Multi-Account Insight Aggregation with Amazon DevOps Guru (Aktivieren der Erkenntnisaggregierung bei mehreren Konten mithilfe von Amazon DevOps Guru)

Zugehörige Beispiele:

- Workshops zur Zuverlässigkeit
- Amazon CloudWatch- und Systems Manager-Workshop

REL06-BP05 Analysen

Erfassen Sie Protokolldateien und Metrikverläufe und analysieren Sie diese, um allgemeine Trends zu erkennen und Workload-Einblicke zu erhalten.

Amazon CloudWatch Logs Insights unterstützt eine einfache und dennoch leistungsstarke Abfragesprache, mit der Sie Protokolldaten analysieren können. Amazon CloudWatch Logs unterstützt auch Abonnements, mit denen Daten nahtlos nach Amazon S3 fließen können, wo Sie sie nutzen oder Amazon Athena verwenden können, um die Daten abzufragen. Abfragen für eine große

Auswahl von Formaten werden ebenfalls unterstütz. Unter Unterstützte SerDes- und Datenformate im Amazon Athena-Benutzerhandbuch finden Sie weitere Informationen dazu. Für die Analyse riesiger Protokolldateisätze können Sie einen Amazon EMR-Cluster ausführen, um Analysen im Petabyte-Bereich auszuführen.

Es gibt es eine Reihe von Werkzeugen von AWS-Partnern und externen Anbietern, die Aggregierung, Verarbeitung, Speicherung und Analyse ermöglichen. Dazu gehören u. a. die Tools New Relic, Splunk, Loggly, Logstash, CloudHealth und Nagios. Die Generierung außerhalb von System- und Anwendungsprotokollen weicht jedoch bei jedem Cloud-Anbieter und häufig sogar bei den einzelnen Services ab.

Ein häufig übersehener Teil des Überwachungsprozesses ist die Datenverwaltung. Sie müssen Aufbewahrungsanforderungen für die Überwachung von Daten definieren und anschließend entsprechende Lebenszyklusrichtlinien anwenden. Amazon S3 unterstützt die Lebenszyklusverwaltung auf der Ebene von S3-Buckets. Diese Lebenszyklusverwaltung kann auf unterschiedliche Weise auf verschiedene Pfade im Bucket angewendet werden. Gegen Ende des Lebenszyklus können Sie die Daten zur Langzeitspeicherung an Amazon S3 Glacier weiterleiten und nach Ablauf der Aufbewahrungsperiode die Speicherung beenden. Die S3 Intelligent-Tiering-Speicherklasse wurde entwickelt, um die Kosten zu optimieren. Daten werden automatisch in die kostengünstigste Zugriffsstufe verschoben, ohne Auswirkungen auf die Leistung oder höheren Betriebsaufwand.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Mit CloudWatch Logs Insights können Sie Protokolldaten in Amazon CloudWatch Logs interaktiv durchsuchen und analysieren.
  - Analysieren von Protokolldaten mit CloudWatch Logs Insights
  - Amazon CloudWatch Logs Insights-Beispielabfragen
- Verwenden Sie Amazon CloudWatch Logs, um Protokolle an Amazon S3 zu senden, wo Sie sie nutzen oder Amazon Athena verwenden können, um die Abfrage der Daten nutzen können.
  - Wie analysiere ich meine Amazon S3-Serverzugriffsprotokolle mit Athena?
    - Erstellen Sie eine S3-Lebenszyklusrichtlinie für Ihren Bucket mit den Serverzugriffsprotokollen. Konfigurieren Sie die Richtlinie so, dass Protokolldateien regelmäßig entfernt werden. Dies reduziert die Datenmenge, die Athena für die einzelnen Abfragen analysiert.
      - Wie erstelle ich eine Lebenszyklusrichtlinie für einen S3-Bucket?

Ressourcen

Relevante Dokumente:

- Amazon CloudWatch Logs Insights-Beispielabfragen
- Analysieren von Protokolldaten mit CloudWatch Logs Insights
- Debugging mit Amazon CloudWatch Synthetics und AWS X-Ray
- Wie erstelle ich eine Lebenszyklusrichtlinie für einen S3-Bucket?
- Wie analysiere ich meine Amazon S3-Serverzugriffsprotokolle mit Athena?
- Workshop zur Beobachtbarkeit
- Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen

REL06-BP06 Regelmäßiges Durchführen von Prüfungen

Prüfen Sie regelmäßig, wie die Workload-Überwachung implementiert ist, und aktualisieren Sie sie auf Grundlage wichtiger Ereignisse und Änderungen.

Eine effektive Überwachung basiert auf wichtigen Geschäftsmetriken. Stellen Sie sicher, dass diese Metriken in Ihrer Workload berücksichtigt werden, wenn sich geschäftliche Prioritäten ändern.

Durch die Prüfung Ihrer Überwachung stellen Sie sicher, dass Sie wissen, wann eine Anwendung die eigenen Verfügbarkeitsziele erfüllt. Für die Durchführung von Ursachenanalysen ist es erforderlich, bei Ausfällen ermitteln zu können, was passiert ist. AWS bietet Services, mit denen Sie den Status Ihrer Services während eines Vorfalls nachverfolgen können.

- Amazon CloudWatch Logs: Sie können Ihre Protokolle in diesem Service speichern und die Inhalte überprüfen.
- Amazon CloudWatch Logs Insights: Ein vollständig verwalteter Service, mit dem Sie umfangreiche Protokolle innerhalb von Sekunden analysieren können. Es bietet Ihnen schnelle, interaktive Abfragen und Visualisierungen.
- AWS Config: Sie können sehen, welche AWS-Infrastruktur zu verschiedenen Zeitpunkten verwendet wurde.
- AWS CloudTrail: Mit diesem Service können Sie erkennen, welche AWS-APIs zu welchem Zeitpunkt und durch welchen Prinzipal aufgerufen wurden.

Bei AWS werden wöchentliche Meetings abgehalten, um die Produktionsleistung zu prüfen und Erkenntnisse mit anderen Teams zu teilen. Da es so viele Teams in AWS gibt, haben wir Das Rad entwickelt, um zufällig eine zu überprüfende Workload auszuwählen. Der Aufbau einer Struktur mit regelmäßigen Überprüfungen der betrieblichen Leistung und mit Wissensaustausch verbessert Ihre Fähigkeit, höhere Leistungen bei Ihren Betriebsteams zu erzielen.

Gängige Antimuster:

- Es werden nur Standardmetriken erfasst.
- Es wird eine Überwachungsstrategie festgelegt, aber nie überprüft.
- Bei Bereitstellung größerer Änderungen wird die Überwachung nicht erörtert.

Vorteile der Einführung dieser bewährten Methode: Durch die regelmäßige Prüfung der Überwachung können Sie mögliche Probleme vorhersehen, statt nur zu reagieren, wenn ein Problem tatsächlich auftritt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Erstellen Sie mehrere Dashboards für die Workload. Ein übergeordnetes Dashboard mit den wichtigsten Geschäftsmetriken ist unverzichtbar. Es sollte zudem die technischen Metriken enthalten, die Sie für den prognostizierten Zustand der Workload bei variabler Nutzung als die relevantesten eingestuft haben. Dashboards für verschiedene Anwendungsebenen und Abhängigkeiten, die untersucht werden können, sind ebenfalls empfehlenswert.
  - Verwenden von Amazon CloudWatch-Dashboards
- Planen und prüfen Sie die Workload-Dashboards regelmäßig. Führen Sie regelmäßige Untersuchungen der Dashboards durch. Was die Gründlichkeit der Untersuchungen angeht, sind unterschiedliche Intervalle denkbar.
  - Spüren Sie Trends in den Metriken auf. Vergleichen Sie die Metrikwerte mit Werten aus der Vergangenheit, um Trends zu erkennen, die darauf hinweisen könnten, dass etwas untersucht werden muss. Beispiele hierfür: ansteigende Latenz, Nachlassen der primären Geschäftsfunktion und zunehmende Anzahl von Reaktionen auf Fehler.
  - Spüren Sie Ausreißer/Anomalien in den Metriken auf. Ausreißer sind anhand von Durchschnitts- oder Mittelwerten oder Anomalien nicht unbedingt erkennbar. Sehen Sie sich die höchsten und niedrigsten Werte in einem bestimmten Zeitraum an und untersuchen Sie die Ursachen für die extremen Werte. Beseitigen Sie nach und nach die Ursachen und legen Sie dabei einen engeren

Maßstab für die Definition von Extremwerten an. So können Sie die Beständigkeit der Workload-Leistung weiter erhöhen.

- Spüren Sie plötzliche Änderungen im Verhalten auf. Eine plötzliche Veränderung in der Menge oder Richtung einer Metrik kann auf eine Änderung in der Anwendung hindeuten. Sie kann aber auch ein Hinweis auf externe Faktoren sein, für deren Verfolgung Sie möglicherweise weitere Metriken hinzufügen müssen.

Ressourcen

Ähnliche Dokumente:

- Amazon CloudWatch Logs Insights-Beispielabfragen
- Debugging mit Amazon CloudWatch Synthetics und AWS X-Ray
- Workshop zur Beobachtbarkeit
- Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen
- Verwenden von Amazon CloudWatch-Dashboards

REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System

Verfolgen Sie Anfragen während der Bearbeitung durch die Servicekomponenten, damit Produktteams Probleme einfacher analysieren und beheben und die Leistung verbessern können.

Gewünschtes Ergebnis: Workloads mit umfassender Nachverfolgung über alle Komponenten hinweg lassen sich leicht debuggen und verbessern so die durchschnittliche Zeit für die Behebung (MTTR) von Fehlern und Latenz durch eine vereinfachte Ursachenerkennung. Die durchgängige Nachverfolgung reduziert die Zeit, die benötigt wird, um betroffene Komponenten zu erkennen und die Ursachen von Fehlern oder Latenzen genau zu ermitteln.

Typische Anti-Muster:

- Nachverfolgung wird für einige Komponenten verwendet, aber nicht für alle. Ohne Nachverfolgung in AWS Lambda können Teams beispielsweise die durch Kaltstarts bei hohen Workloads verursachte Latenz nicht genau nachvollziehen.
- Synthetische Canaries oder Real-User Monitoring (RUM) sind nicht für Nachverfolgung konfiguriert. Ohne Canaries oder RUM wird die Telemetrie der Client-Interaktion in der Spurenanalyse ausgelassen, was zu einem unvollständigen Leistungsprofil führt.

- Hybride Workloads umfassen sowohl cloudnative Nachverfolgungs-Tools als auch Tools von Drittanbietern, es wurden jedoch keine Schritte unternommen, um eine einzige Nachverfolgungs-Lösung auszuwählen und vollständig zu integrieren. Basierend auf der gewählten Nachverfolgungs-Lösung sollten cloudnative Nachverfolgungs-SDKs verwendet werden, um Komponenten zu instrumentieren, die nicht cloudnativ sind. Oder Tools von Drittanbietern sollten so konfiguriert werden, dass sie cloudnative Nachverfolgungstelemetrie aufnehmen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Entwicklungsteams über Probleme informiert werden, können sie sich ein vollständiges Bild der Interaktionen zwischen den Systemkomponenten machen, einschließlich der Beziehung zwischen Komponenten, Protokollierung, Leistung und Ausfällen. Da die Nachverfolgung die visuelle Identifizierung der Ursachen erleichtert, können diese schneller untersucht werden. Teams, die die Interaktionen der Komponenten im Detail verstehen, treffen bessere und schnellere Entscheidungen bei der Lösung von Problemen. Entscheidungen, z. B. wann ein Notfallwiederherstellung (DR)-Failover eingeleitet werden sollte oder wo Strategien zur Selbstreparatur am besten implementiert werden sollten, können durch die Analyse von Systemprotokollen verbessert werden, was letztlich die Kundenzufriedenheit mit Ihren Services erhöht.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Teams, die verteilte Anwendungen betreiben, können mithilfe von Nachverfolgungs-Tools eine Korrelationskennung einrichten, Spuren von Anfragen erfassen und Service-Maps für verbundene Komponenten erstellen. Alle Anwendungskomponenten sollten in den Anforderungsspuren enthalten sein, einschließlich Service-Clients, Middleware-Gateways und Event Busse, Rechenkomponenten und Speicher, einschließlich Schlüssel-Wert-Speicher und -Datenbanken. Integrieren Sie synthetische Canaries und Real-User Monitoring in Ihre Konfiguration für die gesamte Nachverfolgung, um die Interaktionen und Latenz von Remote-Clients zu messen, sodass Sie die Leistung Ihres Systems anhand Ihrer Service Level Agreements und Ziele genau bewerten können.

Nutzen Sie Instrumentierungsservices wie AWS X-Ray und Amazon CloudWatch-Anwendungsüberwachung , um einen vollständigen Überblick über die Anfragen zu erhalten, die in Ihrer Anwendung verarbeitet werden. X-Ray erfasst Anwendungstelemetrie und ermöglicht es Ihnen, diese nach Payloads, Funktionen, Spuren, Services und APIs zu visualisieren und zu filtern. Sie kann für Systemkomponenten aktiviert werden, bei denen kein Code oder Low-Code verwendet wird. Die CloudWatch-Anwendungsüberwachung umfasst ServiceLens, um Ihre Spuren in Metriken, Protokollen und Alarmen zu integrieren. Die CloudWatch-Anwendungsüberwachung umfasst auch

synthetische Funktionen zur Überwachung Ihrer Endpunkte und APIs sowie Real-User Monitoring zur Instrumentierung Ihrer Webanwendungsclients.

Implementierungsschritte

- Verwenden Sie AWS X-Ray auf allen unterstützten nativen Services wie Amazon S3, AWS Lambda und Amazon API Gateway. Diese AWS-Services ermöglichen X-Ray mit Konfigurationsschaltern unter Verwendung von Infrastruktur als Code, AWS SDKs oder der AWS Management Console.
- Instrumentenanwendungen AWS Distro for OpenTelemetry und X-Ray oder Erfassungs-Agenten von Drittanbietern.
- Im AWS X-Ray-Entwicklerhandbuch finden Sie weitere Informationen für die programmiersprachenspezifische Implementierung. In diesen Dokumentationsabschnitten wird detailliert beschrieben, wie HTTP-Anfragen, SQL-Abfragen und andere Prozesse, die für Ihre Anwendungsprogrammiersprache spezifisch sind, instrumentiert werden.
- Verwenden Sie X-Ray-Nachverfolgung für Amazon CloudWatch synthetische Canaries und Amazon CloudWatch RUM, um den Anforderungspfad von Ihrem Endbenutzer-Client durch Ihre AWS-Downstream-Infrastruktur zu analysieren.
- Konfigurieren Sie CloudWatch-Metriken und -Alarme auf der Grundlage des Ressourcenzustands und der Canary-Telemetrie, sodass Teams schnell über Probleme informiert werden und dann mit ServiceLens Spuren und Servicemaps eingehend untersuchen können.
- Aktivieren Sie die X-Ray-Integration für Nachverfolgungs-Tools von Drittanbietern wie Datadog, New Relicoder Dynatrace, wenn Sie Tools von Drittanbietern als primäre Nachverfolgungslösung verwenden.

Ressourcen

Zugehörige bewährte Methoden:

- REL06-BP01 Überwachen aller Komponenten der Workload (Generierung)
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Zugehörige Dokumente:

- Was ist AWS X-Ray?
- Amazon CloudWatch: Anwendungsüberwachung

- Debugging mit Amazon CloudWatch Synthetics und AWS X-Ray
- Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen
- Integration von AWS X-Ray in andere AWS-Dienste
- AWS Distro for OpenTelemetry und AWS X-Ray
- Amazon CloudWatch: Synthetische Überwachung verwenden
- Amazon CloudWatch: CloudWatch RUM verwenden
- Amazon CloudWatch Synthetics Canary und Amazon CloudWatch-Alarm einrichten
- Verfügbarkeit und mehr: Verdeutlichung und Verbesserung der Ausfallsicherheit bei verteilten Systemen in AWS

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit

Zugehörige Videos:

- AWS re:Invent 2022: Kontenübergreifendes Überwachen Ihrer Anwendungen
- Überwachen Ihrer AWS-Anwendungen

Zugehörige Tools:

- AWS X-Ray
- Amazon CloudWatch
- Amazon Route 53

REL 7 Wie lässt sich die Workload so gestalten, dass sie sich an Bedarfsänderungen anpasst?

Eine skalierbare Workload bietet die Elastizität, Ressourcen automatisch entsprechend dem aktuellen Bedarf hinzuzufügen oder zu entfernen.

Bewährte Methoden

- REL07-BP01 Automatisches Abrufen und Skalieren von Ressourcen:

- REL07-BP02 Abrufen von Ressourcen bei Erkennen einer Beeinträchtigung einer Workload
- REL07-BP03 Abrufen von Ressourcen bei Feststellung, dass für eine Workload mehr Ressourcen benötigt werden
- REL07-BP04 Durchführen von Lasttests für die Workload

REL07-BP01 Automatisches Abrufen und Skalieren von Ressourcen:

Wenn Sie beeinträchtigte Ressourcen ersetzen oder Ihre Workload skalieren, können Sie den Prozess mithilfe von verwalteten AWS-Services wie Amazon S3 und AWS Auto Scaling automatisieren. Sie können die Skalierung auch mit Tools von Drittanbietern und AWS SDKs automatisieren.

Zu den verwalteten AWS-Services gehören Amazon S3, Amazon CloudFront, AWS Auto Scaling, AWS Lambda, Amazon DynamoDB, AWS Fargate und Amazon Route 53.

Mit AWS Auto Scaling können Sie beeinträchtigte Instances erkennen und ersetzen. Außerdem können Sie Skalierungspläne für Ressourcen erstellen, unter anderem für Amazon EC2 -Instances und Spot-Flotten, Amazon ECS -Aufgaben, Amazon DynamoDB -Tabellen und -Indizes sowie für Amazon Aurora -Replicas.

Bei der Skalierung von EC2-Instances sollten Sie mehrere Availability Zones nutzen (mindestens drei) und Kapazität hinzufügen oder entfernen, um ein Gleichgewicht über diese Availability Zones hinweg zu gewährleisten. ECS-Aufgaben oder Kubernetes-Pods (bei Verwendung von Amazon Elastic Kubernetes Service) sollten ebenfalls über mehrere Availability Zones hinweg verteilt werden.

Bei Verwendung von AWS Lambda werden Instances automatisch skaliert. Jedes Mal, wenn eine Ereignisbenachrichtigung für Ihre Funktion eingeht, ermittelt AWS Lambda schnell freie Kapazität innerhalb seiner Compute-Flotte und führt Ihren Code bis zur zugeteilten Gleichzeitigkeit aus. Sie müssen sicherstellen, dass die erforderliche Gleichzeitigkeit auf dem spezifischen Lambda und in Ihrem Service Quotas konfiguriert ist.

Amazon S3 wird automatisch skaliert, um hohe Anfrageraten zu verarbeiten. Beispielsweise kann Ihre Anwendung mindestens 3 500 PUT/COPY/POST/DELETE- oder 5 500 GET/HEAD-Anfragen pro Sekunde pro Präfix in einem Bucket erreichen. Für die Anzahl der Präfixe in einem Bucket gibt es keine Beschränkungen. Sie können Ihre Lese- oder Schreibleistung erhöhen, indem Sie Lesevorgänge parallelisieren. Wenn Sie beispielsweise 10 Präfixe in einem Amazon S3-Bucket erstellen, können Sie die Leseleistung auf 55 000 Leseanfragen pro Sekunde skalieren, um die Lesevorgänge zu parallelisieren.

Konfigurieren und nutzen Sie Amazon CloudFront oder ein vertrauenswürdiges Content Delivery Network (CDN). Ein CDN kann Antwortzeiten für Endbenutzer verkürzen und Anfragen für Inhalte aus dem Cache verarbeiten. Dadurch wird die Notwendigkeit zur Skalierung Ihrer Workload verringert.

Gängige Antimuster:

- Es werden Auto-Scaling-Gruppen für die automatisierte Reparatur implementiert, aber keine Elastizität.
- Als Reaktion auf stark ansteigenden Datenverkehr wird automatisch skaliert.
- Es werden hochgradig zustandsbehaftete Anwendungen bereitgestellt, wodurch die Option der Elastizität entfällt.

Vorteile der Einführung dieser bewährten Methode: Durch die Automatisierung entfällt die Gefahr manueller Fehler bei der Bereitstellung und Außerbetriebnahme von Ressourcen. Durch die Automatisierung entfällt das Risiko von Kostenüberschreitungen und Dienstverweigerungen (Denial of Service) aufgrund der langsamen Reaktion auf Bedürfnisse bezüglich der Bereitstellung oder Außerbetriebnahme von Ressourcen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Konfigurieren und nutzen Sie AWS Auto Scaling. Hiermit erfolgt eine Überwachung der Anwendungen und eine automatische Anpassung der Kapazität, um eine stabile, vorhersehbare Leistung zu möglichst niedrigen Kosten aufrechtzuerhalten. Mit AWS Auto Scaling lässt sich die Anwendungsskalierung für mehrere Ressourcen in mehreren Services einrichten.
  - Was ist AWS Auto Scaling?
    - Konfigurieren Sie Auto Scaling nach Bedarf in Ihren Amazon EC2-Instances und Spot-Flotten, Amazon ECS-Aufgaben, Amazon DynamoDB-Tabellen und -Indizes, Amazon Aurora-Replikaten und AWS Marketplace-Appliances.
      - Automatische Verwaltung der Durchsatzkapazität mit DynamoDB Auto Scaling
        - Legen Sie über die Service-API Alarme, Skalierungsrichtlinien sowie Aufwärm- und Abkühlungszeiten fest.
- Nutzen Sie Elastic Load Balancing. Load Balancer können die Last nach Pfaden oder Netzwerkkonnektivität verteilen.
  - Was ist Elastic Load Balancing?

- Application Load Balancers kann Lasten nach Pfaden verteilen.
  - Was ist ein Application Load Balancer?
    - Konfigurieren Sie einen Application Load Balancer, um Datenverkehr basierend auf dem Pfad unter dem Domänennamen auf verschiedene Workloads zu verteilen.
    - Mit Application Load Balancers können Sie Lasten entsprechend dem AWS Auto Scaling verteilen, um den Bedarf zu verwalten.
      - Nutzen eines Load Balancer mit einer Auto-Scaling-Gruppe
- Network Load Balancer können Lasten nach Verbindungen verteilen.
  - Was ist ein Network Load Balancer?
    - Konfigurieren Sie einen Network Load Balancer, um Datenverkehr auf verschiedene Workloads mit TCP zu verteilen oder einen konstanten Satz von IP-Adressen für die Workload festzulegen.
    - Mit Network Load Balancern können Sie Lasten entsprechend dem AWS Auto Scaling verteilen, um den Bedarf zu verwalten.

- Nutzen Sie einen hochverfügbaren DNS-Anbieter. Mithilfe von DNS-Namen können Ihre Benutzer anstelle von IP-Adressen Namen eingeben, um auf Ihre Workloads zuzugreifen. Diese Informationen werden innerhalb einer definierten Reichweite (meist weltweit) für Benutzer der Workload verteilt.
  - Nutzen Sie Amazon Route 53 oder einen vertrauenswürdigen DNS-Anbieter.
    - Was ist Amazon Route 53?
  - Mit Route 53 können Sie Ihre CloudFront-Verteilungen und Load Balancer verwalten.
    - Ermitteln Sie die zu verwaltenden Domänen und Subdomänen.
    - Erstellen Sie entsprechende Datensätze mithilfe von ALIAS- oder CNAME-Datensätzen.
      - Arbeiten mit Datensätzen
- Nutzen Sie das globale AWS-Netzwerk, um den Pfad von den Benutzern zu Ihren Anwendungen zu optimieren. AWS Global Accelerator überwacht kontinuierlich den Zustand der Anwendungsendpunkte und leitet den Datenverkehr in weniger als 30 Sekunden an fehlerfreie Endpunkte um.
  - Bei AWS Global Accelerator handelt es sich um einen Service, der die Verfügbarkeit und Leistung der Anwendungen bei lokalen oder weltweiten Benutzern verbessert. Er stellt statische IP-Adressen bereit, die als fester Einstiegspunkt zu den Anwendungsendpunkten in einer einzelnen oder in mehreren AWS-Regionen fungieren, z. B. Application Load Balancers, Network Load Balancer oder Amazon EC2-Instances.

    - Was ist AWS Global Accelerator?
- Konfigurieren und nutzen Sie Amazon CloudFront oder ein vertrauenswürdiges Content Delivery Network (CDN). Ein Content Delivery Network kann Antwortzeiten für Endbenutzer verkürzen und Anfragen für Inhalte verarbeiten, die zu einer unnötigen Skalierung Ihrer Workloads führen könnten.
  - Was ist Amazon CloudFront?
    - Konfigurieren Sie Amazon CloudFront-Verteilungen für Ihre Workloads oder verwenden Sie das CDN eines Drittanbieters.
      - Sie können festlegen, dass die Workloads nur über CloudFront zugänglich sind. Legen Sie hierfür die IP-Bereiche für CloudFront in den Sicherheitsgruppen oder Zugriffsrichtlinien der Endpunkte fest.

Ressourcen

Relevante Dokumente:

- APN-Partner: Partner, die Ihnen beim Erstellen automatisierter Datenverarbeitungslösungen helfen können
- AWS Auto Scaling: Funktionsweise von Skalierungsplänen
- AWS Marketplace: Für Auto Scaling geeignete Produkte
- Automatische Verwaltung der Durchsatzkapazität mit DynamoDB Auto Scaling
- Nutzen eines Load Balancer mit einer Auto-Scaling-Gruppe
- Was ist AWS Global Accelerator?
- Was ist Amazon EC2 Auto Scaling?
- Was ist AWS Auto Scaling?
- Was ist Amazon CloudFront?
- Was ist Amazon Route 53?
- Was ist Elastic Load Balancing?
- Was ist ein Network Load Balancer?
- Was ist ein Application Load Balancer?
- Arbeiten mit Datensätzen

REL07-BP02 Abrufen von Ressourcen bei Erkennen einer Beeinträchtigung einer Workload

Skalieren Sie Ressourcen bei Bedarf reaktiv, wenn die Verfügbarkeit beeinträchtigt ist, um die Verfügbarkeit der Workload wiederherzustellen.

Sie müssen zunächst Zustandsprüfungen und die Kriterien für diese Prüfungen konfigurieren, um anzugeben, wann die Verfügbarkeit durch fehlende Ressourcen beeinträchtigt wird. Benachrichtigen Sie anschließend entweder die zuständigen Mitarbeiter, um die Ressource manuell zu skalieren, oder starten Sie die Automatisierung, um sie automatisch zu skalieren.

Die Skalierung kann manuell an Ihre Workload angepasst werden, z. B. indem Sie die Anzahl der EC2-Instances in einer Auto Scaling-Gruppe ändern oder den Durchsatz einer DynamoDB-Tabelle über die AWS Management Console oder AWS CLI. Wann immer es möglich ist, sollte jedoch Automatisierung eingesetzt werden (siehe Automatisiertes Abrufen oder Skalieren von Ressourcen).

Gewünschtes Ergebnis: Skalierungsaktivitäten (entweder automatisch oder manuell) werden eingeleitet, um die Verfügbarkeit wiederherzustellen, sobald ein Ausfall oder eine Verschlechterung der Kundenerfahrung festgestellt wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Implementieren Sie Beobachtbarkeit und Überwachung für alle Komponenten Ihres Workloads, um die Kundenerfahrung zu überwachen und Fehler zu erkennen. Definieren Sie die manuellen oder automatischen Verfahren zur Skalierung der erforderlichen Ressourcen. o Weitere Informationen finden Sie unter REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler.

Implementierungsschritte

- Definieren Sie die manuellen oder automatisierten Verfahren, mit denen die erforderlichen Ressourcen skaliert werden.
  - Die Skalierungsverfahren hängen davon ab, wie die verschiedenen Komponenten innerhalb Ihres Workloads gestaltet sind.
  - Die Skalierungsverfahren variieren auch je nach der zugrunde liegenden Technologie, die verwendet wird.
    - Komponenten, die AWS Auto Scaling verwenden, können Skalierungspläne nutzen, um eine Reihe von Anweisungen für die Skalierung Ihrer Ressourcen zu konfigurieren. Wenn Sie mit AWS CloudFormation arbeiten oder AWS-Ressourcen Tags hinzufügen, können Sie pro Anwendung Skalierungspläne für verschiedene Ressourcengruppen einrichten. Auto

Scaling bietet Empfehlungen für Skalierungsstrategien, die auf die einzelnen Ressourcen zugeschnitten sind. Nachdem Sie einen Skalierungsplan erstellt haben, kombiniert Auto Scaling zur Unterstützung Ihrer Skalierungsstrategie Methoden für die dynamische und prädiktive Skalierung. Weitere Informationen finden Sie unter Funktionsweise von Skalierungsplänen.

- Mit Amazon EC2 Auto Scaling können Sie sicherstellen, dass Ihnen die richtige Anzahl von Amazon EC2-Instances zur Verfügung steht, um die Anwendungslast zu bewältigen. Sie erstellen Sammlungen von EC2-Instances, sogenannte Auto Scaling-Gruppen. In jeder Auto Scaling-Gruppe können Sie die Mindestanzahl von Instances angeben. Amazon EC2 Auto Scaling stellt dann sicher, dass die Gruppe diese Größe nie unter- oder überschreitet. Weitere Informationen finden Sie unter Was ist Amazon EC2 Auto Scaling?
- Bei der automatischen Skalierung von Amazon DynamoDB wird der Application Auto Scaling-Service genutzt, um die bereitgestellte Durchsatzkapazität in Ihrem Auftrag dynamisch an die Muster des tatsächlichen Datenverkehrs anzupassen. So kann eine Tabelle oder ein GSI die bereitgestellte Lese- und Schreibkapazität erhöhen, um einen plötzlichen Anstieg des Datenverkehrs ohne Drosselung zu bewältigen. Weitere Informationen finden Sie unter Automatische Verwaltung der Durchsatzkapazität mit DynamoDB-Auto-Scaling.

Ressourcen

Zugehörige bewährte Methoden:

- REL07-BP01 Automatisches Abrufen und Skalieren von Ressourcen
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Zugehörige Dokumente:

- AWS Auto Scaling: Funktionsweise von Skalierungsplänen
- Automatische Verwaltung der Durchsatzkapazität mit DynamoDB-Auto-Scaling
- Was ist Amazon EC2 Auto Scaling?

REL07-BP03 Abrufen von Ressourcen bei Feststellung, dass für eine Workload mehr Ressourcen benötigt werden

Skalieren Sie Ressourcen proaktiv, um den Bedarf zu erfüllen und Auswirkungen auf die Verfügbarkeit zu vermeiden.

Viele AWS-Services werden automatisch dem Bedarf entsprechend skaliert. Wenn Sie Amazon EC2-Instances oder Amazon ECS-Cluster verwenden, können Sie die automatische Skalierung dieser Instances auf der Grundlage von Nutzungsmetriken konfigurieren, die dem Bedarf Ihrer Workload entsprechen. Für Amazon EC2 können Sie die durchschnittliche CPU-Auslastung, die Anzahl der Load Balancer-Anfragen oder die Netzwerkbandbreite verwenden, um EC2-Instances zu skalieren. Für Amazon ECS können Sie die durchschnittliche CPU-Auslastung, die Anzahl der Load-Balancer-Anfragen und die Speichernutzung verwenden, um ECS-Aufgaben auf- oder abzuskalieren. Mit Target Auto Scaling auf AWS fungiert der Autoscaler wie ein Haushaltsthermostat, der Ressourcen hinzufügt oder entfernt, um den von Ihnen angegebenen Zielwert (z. B. 70 % CPU-Auslastung) beizubehalten.

AWS Auto Scaling kann auch Predictive Auto Scalingdurchführen. Dabei wird Machine Learning verwendet, um die bisherige Workload jeder Ressource zu analysieren und regelmäßig die zukünftige Last für die nächsten zwei Tage zu prognostizieren.

Das Gesetz von Little hilft beim Berechnen der Anzahl von Compute-Instances, die Sie benötigen (EC2-Instances, gleichzeitige Lambda-Funktionen usw.).

$L = \lambda W$

L = Anzahl der Instances (oder mittlere Gleichzeitigkeit im System)

λ = mittlere Rate des Eingangs von Anfragen (Anfrage/Sekunde)

W = mittlere Zeit, die jede Anfrage im System verbringt (Sekunden)

Wenn beispielsweise bei 100 RPS die Verarbeitung jeder Anfrage 0,5 Sekunden dauert, benötigen Sie 50 Instances, um mit dem Bedarf Schritt zu halten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Rufen Sie Ressourcen ab, wenn Sie feststellen, dass für eine Workload mehr Ressourcen benötigt werden. Skalieren Sie Ressourcen proaktiv, um den Bedarf zu erfüllen und Auswirkungen auf die Verfügbarkeit zu vermeiden.
  - Berechnen Sie, wie viele Rechenressourcen Sie benötigen (Gleichzeitigkeit der Datenverarbeitung), um eine bestimmte Anfragerate zu verarbeiten.
    - Berichte über das Gesetz von Little

- Wenn Sie über ein Verlaufsmuster für die Nutzung verfügen, richten Sie die geplante Skalierung für Amazon EC2 ein.
  - Geplante Skalierung für Amazon EC2 Auto Scaling
- Verwenden Sie die vorausschauende Skalierung von AWS.
  - Prädiktive Skalierung für EC2, unterstützt von Machine Learning

Ressourcen

Relevante Dokumente:

- AWS Auto Scaling: Funktionsweise von Skalierungsplänen
- AWS Marketplace: Für Auto Scaling geeignete Produkte
- Automatische Verwaltung der Durchsatzkapazität mit DynamoDB Auto Scaling
- Prädiktive Skalierung für EC2, unterstützt von Machine Learning
- Geplante Skalierung für Amazon EC2 Auto Scaling
- Berichte über das Gesetz von Little
- Was ist Amazon EC2 Auto Scaling?

REL07-BP04 Durchführen von Lasttests für die Workload

Messen Sie anhand von Lasttests, ob die Skalierung den Workload-Anforderungen gerecht wird.

Es ist wichtig, regelmäßige Lasttests durchzuführen. Mit diesen Tests können Sie die Belastungsgrenze Ihrer Workload ermitteln und deren Leistung prüfen. AWS erleichtert das Einrichten temporärer Testumgebungen, die den Umfang Ihrer Produktions-Workload modellieren. Sie können in der Cloud bei Bedarf eine Testumgebung in Produktionsgröße einrichten, Ihre Tests abschließen und die Ressourcen dann wieder stilllegen. Weil Sie für die Testumgebung nur dann zahlen, wenn sie genutzt wird, können Sie Ihre Live-Umgebung zu einem Bruchteil der Kosten testen, die Sie an einem On-Premises-Standort hätten.

Lasttests in der Produktion sollten auch im Rahmen von Ernstfallübungen durchgeführt werden, bei denen das Produktionssystem in einem Zeitraum mit geringer Kundennutzung stark belastet wird. Alle Mitarbeiter sollten an dieser Übung beteiligt sein, die Ergebnisse gemeinsam interpretieren und auftretende Probleme beheben.

Typische Anti-Muster:

- Es werden Lasttests für Bereitstellungen durchgeführt, die nicht mit der Konfiguration der Produktionsumgebung übereinstimmen.
- Lasttests werden nur für einzelne Teile, nicht aber für die gesamte Workload durchgeführt.
- Es werden Lasttests mit einer Teilmenge von Anfragen durchgeführt, aber nicht mit einer repräsentativen Gruppe tatsächlicher Anfragen.
- Es werden Lasttests mit einem kleinen Sicherheitsfaktor durchgeführt, der über der erwarteten Last liegt.

Vorteile der Nutzung dieser bewährten Methode: Sie wissen, welche Komponenten in der Architektur unter Last ausfallen, und können die zu überwachenden Metriken festlegen, die rechtzeitig auf die Annäherung an die Belastungsgrenze hinweisen, damit Sie das Problem beheben und entsprechende Auswirkungen vermeiden können.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

- Bestimmen Sie anhand von Lasttests, welcher Aspekt der Workload angeben soll, dass Kapazität hinzugefügt oder entfernt werden muss. Bei Lasttests sollte ein repräsentativer Datenverkehr zum Einsatz kommen, der dem in der Produktion ähnelt. Erhöhen Sie unter Beobachtung der instrumentierten Metriken die Last, um zu bestimmen, welche Metrik angibt, wann Ressourcen hinzugefügt oder entfernt werden müssen.
  - Verteilte Lasttests auf AWS: Simulation Tausender verbundener Benutzer
    - Ermitteln Sie die Zusammensetzung von Anfragen. Möglicherweise haben Sie unterschiedliche Zusammensetzungen von Anfragen. Daher sollten Sie sich bei der Ermittlung der Zusammensetzung des Datenverkehrs verschiedene Zeiträume ansehen.
    - Implementieren Sie einen Lasttreiber. Zum Implementieren eines Lasttreibers können Sie Software mit eigenem Code, Open-Source-Software oder kommerzielle Software verwenden.
    - Führen Sie Lasttests zunächst mit geringer Kapazität durch. Schon bei der Erhöhung der Last für eine Einheit mit geringerer Kapazität, etwa einer einzelnen Instance oder einem einzelnen Container, stellen Sie unmittelbare Auswirkungen fest.
    - Führen Sie Lasttests mit größerer Kapazität durch. Bei einer verteilten Last sehen die Auswirkungen anders aus. Daher müssen Sie bei Tests Bedingungen herstellen, die der Produktionsumgebung möglichst nahekommen.

Ressourcen

Zugehörige Dokumente:

- Verteilte Lasttests auf AWS: Simulation Tausender verbundener Benutzer
- Lasttestanwendungen

Zugehörige Videos:

- AWS Summit ANZ 2023: Mit AWS Distributed Load Testing zuversichtlich in die Zukunft starten

REL 8. Wie implementieren Sie Änderungen?

Kontrollierte Änderungen sind erforderlich, um neue Funktionen bereitzustellen und um sicherzustellen, dass die Workloads und die Betriebsumgebung bekannte Software ausführen und auf vorhersagbare Weise durch Patches aktualisiert oder ersetzt werden können. Wenn diese Änderungen nicht kontrolliert stattfinden, ist es schwierig, ihre Auswirkungen vorherzusagen oder daraus entstehende Probleme zu beheben.

Bewährte Methoden

- REL08-BP01 Verwenden von Runbooks für Standardaktivitäten wie die Bereitstellung
- REL08-BP02 Integrieren von Funktionstests in die Bereitstellung
- REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung
- REL08-BP04 Bereitstellung mit einer unveränderlichen Infrastruktur
- REL08-BP05 Automatisieren von Änderungen

REL08-BP01 Verwenden von Runbooks für Standardaktivitäten wie die Bereitstellung

Runbooks sind vordefinierte Verfahren, die ein bestimmtes Ergebnis verfolgen. Verwenden Sie Runbooks, um Standardaktivitäten manuell oder automatisch durchzuführen. Beispiele für solche Aktivitäten sind etwa die Bereitstellung und das Patchen einer Workload oder das Vornehmen von DNS-Änderungen.

Sie können z. B. Prozesse einrichten, um bei Bereitstellungen die Rollback-Sicherheit zu gewährleisten. Wenn Sie eine Bereitstellung ohne Unterbrechung für Ihre Kunden zurücksetzen können, steigert das die Zuverlässigkeit Ihres Service.

Für Runbook-Verfahren sollten Sie mit einem gültigen, effektiven manuellen Prozess beginnen, diesen in Code implementieren und ggf. die automatische Ausführung auslösen.

Selbst bei anspruchsvollen Workloads mit umfassender Automatisierung sind Runbooks nützlich, um Ernstfallübungen auszuführen oder strenge Berichterstellungs- und Auditing-Anforderungen zu erfüllen.

Playbooks werden als Reaktion auf bestimmte Vorfälle verwendet und mit Runbooks sollen bestimmte Ergebnisse erzielt werden. Häufig werden Runbooks für Routineaktivitäten genutzt, während Playbooks für die Reaktion auf außerplanmäßige Ereignisse verwendet werden.

Gängige Antimuster:

- Durchführen ungeplanter Änderungen an der Konfiguration in der Produktion.
- Überspringen von Schritten in Ihrem Plan, um schneller bereitzustellen, was dann jedoch zum Fehlschlagen der Bereitstellung führt.
- Vornehmen von Änderungen, ohne die Umkehrung der Änderung zu testen.

Vorteile der Einführung dieser bewährten Methode: Die effektive Änderungsplanung erhöht Ihre Fähigkeit, die Änderung erfolgreich auszuführen, da Sie sich über alle betroffenen Systeme bewusst sind. Die Validierung Ihrer Änderungen in Testumgebungen erhöht Ihre Sicherheit.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Unterstützen Sie konsistente und schnelle Reaktionen auf gut bekannte Ereignisse, indem Sie Verfahren in Runbooks dokumentieren.
  - AWS-Well-Architected-Framework: Konzepte: Runbook
- Verwenden Sie zur Definition Ihrer Infrastruktur den Grundsatz „Infrastructure as Code“. Wenn Sie Ihre Infrastruktur mit AWS CloudFormation oder dem vertrauenswürdigen Tool eines Drittanbieters definieren, können Sie Änderungen mithilfe einer Versionskontrollsoftware versionieren und nachverfolgen.
  - Nutzen Sie zur Definition Ihrer Infrastruktur AWS CloudFormation (oder das vertrauenswürdige Tool eines Drittanbieters).
    - Was ist AWS CloudFormation?
  - Erstellen Sie unter Anwendung guter Grundsätze für das Softwaredesign Vorlagen, die getrennt und entkoppelt sind.

- Ermitteln Sie die für die Implementierung erforderlichen Berechtigungen, Vorlagen und zuständigen Parteien.
  - Zugriffssteuerung mit AWS Identity and Access Management
- Verwenden Sie zur Versionskontrolle eine Quellkontrolle wie AWS CodeCommit oder das vertrauenswürdige Tool eines Drittanbieters.
  - Was ist AWS CodeCommit?

Ressourcen

Relevante Dokumente:

- APN-Partner: Partner, die Sie beim Erstellen automatisierter Bereitstellungslösungen unterstützen können
- AWS Marketplace: Produkte zur Automatisierung Ihrer Bereitstellungen
- AWS-Well-Architected-Framework: Konzepte: Runbook
- Was ist AWS CloudFormation?
- Was ist AWS CodeCommit?

Ähnliche Beispiele:

- Automating operations with Playbooks and Runbooks (Vorgänge mit Playbooks und Runbooks automatisieren)

REL08-BP02 Integrieren von Funktionstests in die Bereitstellung

Funktionstests werden im Rahmen der automatisierten Bereitstellung ausgeführt. Wenn die Erfolgskriterien nicht erfüllt sind, wird die Pipeline angehalten oder rückgängig gemacht. Diese Tests werden in einer Vorproduktionsumgebung ausgeführt, die vor der Produktion in der Pipeline bereitgestellt wird. Idealerweise erfolgt dies im Rahmen einer Bereitstellungspipeline.

Gewünschtes Ergebnis: Sie verwenden Automatisierung, um Funktionstests durchzuführen, und die zugehörigen Testdaten reduzieren die Testdauer und die Kosten und verbessern die Genauigkeit der Testergebnisse. Sie integrieren Funktionstests als Teil Ihres Bereitstellungsprozesses, was Ihnen hilft, Ihre Veröffentlichungspipelines für schnelle und zuverlässige Anwendungs- und Infrastrukturupdates zu automatisieren.

Typische Anti-Muster:

- Sie führen Tests außerhalb der Bereitstellungspipeline manuell durch.
- Sie überspringen Testschritte in Ihrer Automatisierung durch manuelle Notfall-Workflows.
- Sie folgen nicht Ihren etablierten Testplänen und Prozessen zugunsten beschleunigter Zeitpläne.

Vorteile der Einführung dieser bewährten Methode: Funktionstests bestätigen, dass das System gemäß den angegebenen Anforderungen funktioniert. Sie werden verwendet, um die beabsichtigte Funktionsreihenfolge von Komponenten wie Benutzeroberflächen, APIs, Datenbanken und Quellcode konsistent zu überprüfen. Wenn Sie diese Systemkomponenten untersuchen, stellen Funktionstests sicher, dass sich jedes Feature wie erwartet verhält, wodurch sowohl die Benutzererwartungen als auch die Integrität der Software geschützt werden. Integrieren Sie Funktionstests als Teil Ihrer regulären Bereitstellung und nutzen Sie die Automatisierung, um alle Änderungen umzusetzen, wodurch das Risiko menschlicher Fehler reduziert wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Integrieren Sie Funktionstests in Ihre Bereitstellung. Funktionstests werden im Rahmen der automatisierten Bereitstellung ausgeführt. Wenn die Erfolgskriterien nicht erfüllt werden, wird die Pipeline gestoppt oder rückgängig gemacht. AWS CodePipeline bietet eine Continuous-Delivery-Pipeline für automatisierte Tests, die es Testern ermöglicht, den gesamten Test- und Bereitstellungsprozess zu automatisieren. Es lässt sich in AWS-Services wie AWS CodeBuild und AWS CodeDeploy zur Automatisierung der Erstellungs-, Test- und Bereitstellungsphasen des Softwareentwicklungszyklus integrieren.

Implementierungsschritte

- Konfigurieren Sie Ihre Pipeline: Richten Sie Ihre Quell-, Build-, Test- und Bereitstellungsphasen mithilfe der AWS CodePipeline-Konsole oder AWS Command Line Interface (CLI) ein.
  - Definieren Sie Ihren Quellcode: Mit AWS CodePipeline können Sie automatisch Quellcode von Versionskontrollsystemen wie GitHub AWS CodeCommit oder Bitbucket abrufen, wodurch verifiziert wird, dass immer der neueste Code zum Testen verwendet wird.
  - Automatisieren Sie Builds und Tests: AWS CodeBuild kann Ihren Code automatisch erstellen und testen und Testberichte generieren. Es unterstützt beliebte Testframeworks wie JUnit, NUnit und TestNG.

- Stellen Sie Ihren Code bereit: Sobald der Code erstellt und getestet wurde, kann AWS CodeDeploy ihn in Ihrer Testumgebung bereitstellen, einschließlich Amazon EC2-Instances, AWS Lambda-Funktionen oder On-Premises-Servern.
- Überwachen Sie Pipelines: AWS CodePipeline kann den Fortschritt Ihrer Pipeline und den Status jeder Phase verfolgen. Sie können auch Qualitätsprüfungen verwenden, um die Pipeline gemäß dem Testausführungsstatus zu blockieren. Außerdem können Sie Benachrichtigungen über jeden Ausfall einer Pipeline-Phase oder den Abschluss einer Pipeline erhalten.

Ressourcen

Zugehörige Dokumente:

- Verwenden von AWS CodePipeline mit AWS CodeBuild zum Testen von Code und zum Ausführen von Builds
- Protokollieren und Überwachen von AWS CodeBuild
- Indikatoren für Funktionstests

REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung

Integrieren Sie Resilienztests, indem Sie bewusst Fehler in Ihr System einleiten, um dessen Leistungsfähigkeit im Falle von Störszenarien zu messen. Resilienztests unterscheiden sich von Geräte- und Funktionstests, die normalerweise in Bereitstellungszyklen integriert werden, da sie sich auf die Identifizierung unerwarteter Ausfälle in Ihrem System konzentrieren. Es ist zwar sicher, in der Vorproduktion mit der Integration von Resilienztests zu beginnen, aber setzen Sie sich das Ziel, diese Tests im Rahmen Ihrer GameDays in der Produktion zu implementieren.

Gewünschtes Ergebnis: Resilienztests tragen dazu bei, Vertrauen in die Fähigkeit des Systems aufzubauen, Beeinträchtigungen in der Produktion standzuhalten. Experimente identifizieren Schwachstellen, die zu Ausfällen führen könnten. Auf diese Weise können Sie Ihr System verbessern, um Ausfälle und Beeinträchtigungen automatisch und effizient zu beheben.

Typische Anti-Muster:

- Mangelnde Beobachtbarkeit und Überwachung in Bereitstellungsprozessen
- Verlass auf Menschen, um Systemausfälle zu beheben
- Analysemechanismen von schlechter Qualität

- Fokus auf bekannte Probleme in einem System und das Fehlen von Experimenten, um unbekannte Probleme zu identifizieren
- Identifizieren von Fehlern, aber keine Lösung
- Keine Dokumentation der Erkenntnisse und keine Runbooks

Vorteile der Einführung dieser bewährten Methode: Resilienztests, die in Ihre Bereitstellungen integriert sind, helfen dabei, unbekannte Probleme im System zu identifizieren, die andernfalls unbemerkt bleiben und zu Produktionsausfällen führen können. Die Identifizierung dieser unbekannten Probleme in einem System hilft Ihnen, Ergebnisse zu dokumentieren, Tests in Ihren CI/CD-Prozess zu integrieren und Runbooks zu erstellen, die die Schadensbegrenzung durch effiziente, wiederholbare Mechanismen vereinfachen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die gängigsten Formen von Resilienztests, die in die Bereitstellungen Ihres Systems integriert werden können, sind die Notfallwiederherstellung und Chaos-Engineering.

- Fügen Sie Aktualisierungen Ihrer Notfallwiederherstellungspläne und Standardarbeitsanweisungen (SOPs) bei jeder wichtigen Bereitstellung hinzu.
- Integrieren Sie Zuverlässigkeitstests in Ihre automatisierten Bereitstellungspipelines. Services wie AWS Resilience Hub können in Ihre CI/CD-Pipeline integriert werden, um kontinuierliche Resilienzbewertungen zu erstellen, die im Rahmen jeder Bereitstellung automatisch bewertet werden.
- Definieren Sie Ihre Anwendungen in AWS Resilience Hub. Resilienzanalysen generieren Codefragmente, die Sie bei der Erstellung von Wiederherstellungsprozeduren als AWS Systems Manager-Dokumente für Ihre Anwendungen unterstützen und eine Liste mit empfohlenen Amazon CloudWatch-Monitoren und -Alarmen enthalten.
- Sobald Ihre DR-Pläne und SOPs aktualisiert sind, führen Sie Notfallwiederherstellungstests durch, um sicherzustellen, dass sie wirksam sind. Mithilfe von Notfallwiederherstellungstests können Sie feststellen, ob Sie Ihr System nach einem Ereignis wiederherstellen und zum normalen Betrieb zurückkehren können. Sie können verschiedene Notfallwiederherstellungsstrategien simulieren und feststellen, ob Ihre Planung ausreicht, um Ihre Verfügbarkeitsanforderungen zu erfüllen. Zu den gängigen Notfallwiederherstellungsstrategien gehören Backup und Wiederherstellung, Pilot Light, Cold Standby, Warm Standby, Hot Standby und Aktiv-Aktiv. Sie alle unterscheiden sich

in Kosten und Komplexität. Vor dem Notfallwiederherstellungstest empfehlen wir, dass Sie Ihr Recovery Time Objective (RTO) und Ihr Recovery Point Objective (RPO) definieren, um die Wahl der zu simulierenden Strategie zu vereinfachen. AWS bietet Notfallwiederherstellungstools wie AWS Elastic Disaster Recovery, die Ihnen unter anderem den Einstieg in Ihre Planung und Tests erleichtern.

- Experimente im Bereich des Chaos-Engineering führen zu Störungen im System, wie z. B. Netzwerk- und Serviceausfällen. Durch die Simulation mit kontrollierten Ausfällen können Sie die Sicherheitsschwachstellen Ihres Systems erkennen und gleichzeitig die Auswirkungen der eingeführten Fehler eindämmen. Führen Sie wie bei den anderen Strategien kontrollierte Ausfallsimulationen in Umgebungen außerhalb der Produktion durch, indem Sie beispielsweise Services wie AWS Fault Injection Service nutzen, um vor dem Einsatz in der Produktion Vertrauen zu gewinnen.

Ressourcen

Zugehörige Dokumente:

- Experimente mit Misserfolgen durch Resilienztests, um die Wiederherstellungsbereitschaft zu verbessern
- Kontinuierliche Bewertung der Anwendungsresistenz mit AWS Resilience Hub und AWS CodePipeline
- Architektur für die Notfallwiederherstellung in AWS, Teil I: Strategien für die Wiederherstellung in der Cloud
- Überprüfen Sie die Belastbarkeit Ihrer Workloads mit Chaos-Engineering
- Grundlagen des Chaos-Engineering
- Workshop zum Chaos-Engineering

Zugehörige Videos:

- AWS re:Invent 2020: Resilienz mit Chaos-Engineering testen
- Verbessern Sie die Ausfallsicherheit von Anwendungen mit AWS Fault Injection Service
- Bereiten Sie Ihre Anwendungen vor und schützen Sie sie vor Störungen mit AWS Resilience Hub

REL08-BP04 Bereitstellung mit einer unveränderlichen Infrastruktur

Eine unveränderliche Infrastruktur sieht vor, dass Updates, Sicherheits-Patches oder Konfigurationsänderungen nicht direkt in Produktions-Workloads durchgeführt werden. Wenn eine Änderung erforderlich ist, wird die Architektur auf einer neuen Infrastruktur eingerichtet und für die Produktion bereitgestellt.

Verfolgen Sie eine Strategie zur Bereitstellung einer unveränderlichen Infrastruktur, um die Zuverlässigkeit, Konsistenz und Reproduzierbarkeit Ihrer Workload-Bereitstellungen zu erhöhen.

Gewünschtes Ergebnis: Bei einer unveränderlichen Infrastruktur sind keine direkten Änderungen an den Infrastrukturressourcen innerhalb eines Workloads erlaubt. Wenn eine Änderung erforderlich ist, wird stattdessen ein neuer Satz aktualisierter Infrastrukturressourcen, der alle erforderlichen Änderungen enthält, parallel zu Ihren vorhandenen Ressourcen bereitgestellt. Diese Bereitstellung wird automatisch validiert und bei Erfolg wird der Datenverkehr schrittweise auf die neuen Ressourcen verlagert.

Diese Bereitstellungsstrategie gilt unter anderem für Softwareupdates, Sicherheits-Patches, Infrastrukturänderungen, Konfigurationsupdates und Anwendungsupdates.

Typische Anti-Muster:

- Implementieren von Änderungen an laufenden Infrastruktur-Ressourcen vor Ort.

Vorteile der Nutzung dieser bewährten Methode:

- Erhöhte Konsistenz zwischen verschiedenen Umgebungen: Da es keine Unterschiede bei den Infrastrukturressourcen zwischen den Umgebungen gibt, wird die Konsistenz erhöht und das Testen vereinfacht.
- Verringerung der Konfigurationsabweichungen: Durch das Ersetzen von Infrastrukturressourcen durch eine bekannte und versionskontrollierte Konfiguration wird die Infrastruktur in einen bekannten, getesteten und vertrauenswürdigen Zustand versetzt, wodurch Konfigurationsabweichungen vermieden werden.
- Zuverlässige atomare Bereitstellungen: Entweder werden die Verteilungen erfolgreich abgeschlossen oder es ändert sich nichts, was die Konsistenz und Zuverlässigkeit des Verteilungsprozesses erhöht.
- Vereinfachte Bereitstellungen: Bereitstellungen werden vereinfacht, da sie keine Upgrades unterstützen müssen. Upgrades sind einfach neue Bereitstellungen.

- Sicherere Bereitstellungen mit schnellen Rollback- und Wiederherstellungsprozessen: Bereitstellungen sind sicherer, da die vorherige funktionierende Version nicht geändert wird. Sie können einen Rollback zur vorherigen Version durchführen, wenn Fehler erkannt werden.
- Verbesserte Sicherheitslage: Indem Sie keine Änderungen an der Infrastruktur zulassen, können Fernzugriffsmechanismen (wie SSH) deaktiviert werden. Dadurch wird der Angriffsvektor reduziert und die Sicherheitslage Ihrer Organisation verbessert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Automatisierung

Bei der Definition einer Strategie zur Bereitstellung einer unveränderlichen Infrastruktur empfiehlt es sich, die Automatisierung so weit wie möglich zu nutzen, um die Reproduzierbarkeit zu erhöhen und das Potenzial für menschliche Fehler zu minimieren. Weitere Informationen finden Sie unter REL08-BP05 Automatisieren von Änderungen und Automating safe, hands-off deployments (Automatisierung sicherer, vollautomatischer Bereitstellungen).

Mit Infrastructure as Code (IaC) werden Schritte zur Bereitstellung, Orchestrierung und Implementierung der Infrastruktur auf programmatische, beschreibende und deklarative Weise definiert und in einem Quellkontrollsystem gespeichert. Die Nutzung von Infrastructure as Code vereinfacht die Automatisierung der Infrastrukturbereitstellung und trägt zur Unveränderbarkeit der Infrastruktur bei.

Bereitstellungsmuster

Wenn eine Änderung des Workloads erforderlich ist, schreibt die Strategie der unveränderlichen Infrastrukturbereitstellung vor, dass ein neuer Satz von Infrastrukturressourcen bereitgestellt wird, einschließlich aller erforderlichen Änderungen. Es ist wichtig, dass diese neuen Ressourcen nach einem Muster eingeführt werden, das die Auswirkungen auf die Benutzer minimiert. Für diese Bereitstellung gibt es zwei Hauptstrategien:

Canary-Bereitstellung: Hierbei wird eine kleine Anzahl Ihrer Kunden auf die neue Version umgestellt, die in der Regel auf einer einzelnen Service-Instance (dem Canary) ausgeführt wird. Anschließend überprüfen Sie sämtliche Verhaltensänderungen oder Fehler, die generiert werden. Sie können Datenverkehr aus der Canary-Umgebung entfernen, wenn kritische Probleme auftreten, und die Benutzer auf die vorherige Version zurücksetzen. Wenn die Bereitstellung erfolgreich verläuft,

können Sie das gewünschte Tempo beibehalten und die Änderungen auf Fehler überwachen, bis der Bereitstellungsvorgang vollständig abgeschlossen ist. Sie können AWS CodeDeploy mit einer Bereitstellungskonfiguration konfigurieren, die eine Canary-Bereitstellung ermöglicht.

Blau/Grün-Bereitstellung: Verhält sich ähnlich wie die Canary-Bereitstellung, nur dass eine komplette Flotte der Anwendung parallel bereitgestellt wird. Sie können Ihre Bereitstellungen über die zwei Stacks (blau und grün) alternieren. Auch hier können Sie Datenverkehr an die neue Version senden und einen Failback auf die alte Version durchführen, wenn bei der Bereitstellung Probleme auftreten. Normalerweise wird der gesamte Datenverkehr auf einmal umgeschaltet. Sie können Ihren Datenverkehr aber auch auf die Versionen verteilen, um die Einführung der neuen Version mithilfe der gewichteten DNS-Routing-Funktionen von Amazon Route 53 durchzuführen. Sie können AWS CodeDeploy und AWS Elastic Beanstalk mit einer Bereitstellungskonfiguration konfigurieren, die eine Blau/Grün-Bereitstellung ermöglicht.

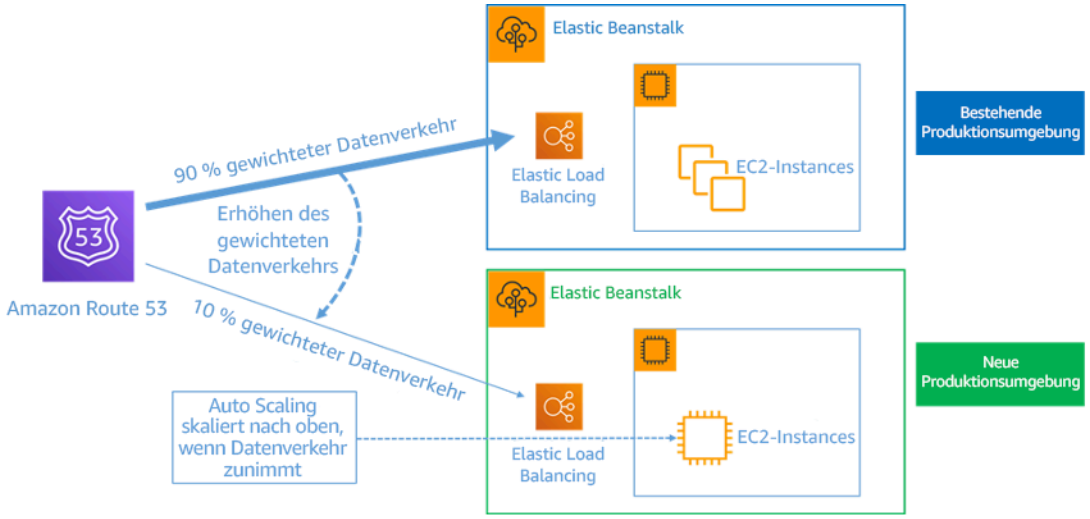


Abbildung 8: Blau/Grün-Bereitstellung mit AWS Elastic Beanstalk und Amazon Route 53

Abweichungserkennung

Als Abweichung wird jede Änderung bezeichnet, die dazu führt, dass eine Infrastrukturressource einen anderen Zustand oder eine andere Konfiguration aufweist als erwartet. Jede Art von nicht verwalteter Konfigurationsänderung widerspricht dem Konzept der unveränderlichen Infrastruktur und sollte erkannt und behoben werden, um eine erfolgreiche Implementierung der unveränderlichen Infrastruktur zu gewährleisten.

Implementierungsschritte

- Untersagen Sie die Änderung laufender Infrastruktur-Ressourcen an Ort und Stelle.

  - Sie können AWS Identity and Access Management (IAM) verwenden, um festzulegen, wer oder was auf Services und Ressourcen in AWS zugreifen darf, fein abgestufte Berechtigungen zentral verwalten und den Zugriff analysieren, um die Berechtigungen über AWS hinweg zu optimieren.
- Automatisieren Sie die Bereitstellung von Infrastrukturressourcen, um die Reproduzierbarkeit zu erhöhen und das Potenzial für menschliche Fehler zu minimieren.
  - Wie im Whitepaper Introduction to DevOps on AWS (Einführung in DevOps in AWS) beschrieben, ist die Automatisierung ein Eckpfeiler der AWS-Services und wird intern in allen Services, Features und Angeboten unterstützt.
  - Durch die Vorberechnung Ihres Amazon Machine Image (AMI) können Sie die Zeit bis zum Start verkürzen. EC2 Image Builder ist ein vollständig verwalteter AWS-Service, der Ihnen hilft, die Erstellung, Wartung, Validierung, Freigabe und Bereitstellung von benutzerdefinierten, sicheren und aktuellen Linux- oder Windows-AMIs zu automatisieren.
  - Zu den Services, die die Automatisierung unterstützen, gehören:
    - AWS Elastic Beanstalk ist ein Service zur schnellen Bereitstellung und Skalierung von Webanwendungen, die mit Java, .NET, PHP, Node.js, Python, Ruby, Go und Docker auf bekannten Servern wie Apache, NGINX, Passenger und IIS entwickelt wurden.
    - AWS Proton unterstützt Plattformteams dabei, all die verschiedenen Tools zu verbinden und zu koordinieren, die Ihre Entwicklungsteams für die Bereitstellung der Infrastruktur, die Bereitstellung von Code, die Überwachung und Updates benötigen. AWS Proton ermöglicht die automatisierte Bereitstellung von Infrastruktur als Code und die Bereitstellung von Serverless und containerbasierten Anwendungen.
  - Die Nutzung von Infrastructure as Code erleichtert die Automatisierung der Infrastrukturbereitstellung und trägt zur Unveränderbarkeit der Infrastruktur bei. AWS bietet Services, die die Erstellung, Bereitstellung und Wartung der Infrastruktur auf programmatische, beschreibende und deklarative Weise ermöglichen.
    - AWS CloudFormation hilft Entwicklern dabei, AWS-Ressourcen in einer geordneten und vorhersehbaren Weise zu erstellen. Ressourcen werden in Textdateien im JSON- oder YAML-Format geschrieben. Die Vorlagen erfordern eine bestimmte Syntax und Struktur, die von den Arten der zu erstellenden und zu verwaltenden Ressourcen abhängt. Sie verfassen Ihre Ressourcen in JSON oder YAML mit einem beliebigen Code-Editor wie AWS Cloud9, checken sie in ein Versionskontrollsystem ein und CloudFormation baut dann die angegebenen Services auf sichere, wiederholbare Weise auf.

    - AWS Serverless Application Model (AWS SAM) ist ein Open-Source-Framework, mit dem Sie Serverless-Anwendungen in AWS erstellen können. AWS SAM lässt sich mit anderen AWS-Services integrieren und ist eine Erweiterung von AWS CloudFormation.
    - AWS Cloud Development Kit (AWS CDK) ist ein Open-Source-Framework für die Softwareentwicklung zur Modellierung und Bereitstellung Ihrer Cloud-Anwendungsressourcen mit Hilfe gängiger Programmiersprachen. Sie können AWS CDK verwenden, um die Anwendungsinfrastruktur mit TypeScript, Python, Java und .NET zu modellieren. AWS CDK verwendet AWS CloudFormation im Hintergrund, um Ressourcen auf sichere, wiederholbare Weise bereitzustellen.
    - AWS -Cloud-Control- API führt einen gemeinsamen Satz von APIs zum Erstellen, Lesen, Aktualisieren, Löschen und Auflisten ein, mit denen Entwickler ihre Cloud-Infrastruktur auf einfache und konsistente Weise verwalten können. Die gemeinsamen Cloud Control API-APIs ermöglichen es Entwicklern, den Lebenszyklus von AWS- und Drittanbieter-Services einheitlich zu verwalten.
- Implementieren Sie Bereitstellungsmuster, die die Auswirkungen auf die Benutzer minimieren.
  - Canary-Bereitstellungen:
    - Einrichten einer API Gateway-Canary-Bereitstellung als Release
    - Erstellen Sie eine Pipeline mit Canary-Bereitstellungen für Amazon ECS mit AWS App Mesh
  - Blau/Grün-Bereitstellungen: Das Whitepaper Blau/Grün-Bereitstellungen in AWS beschreibt Beispieltechniken zur Umsetzung von Blau/Grün-Bereitstellungsstrategien.
- Erkennen Sie Konfigurations- oder Zustandsabweichungen. Weitere Informationen finden Sie unter Erkennen von nicht verwalteten Konfigurationsänderungen an Stacks und Ressourcen.

Ressourcen

Zugehörige bewährte Methoden:

- REL08-BP05 Automatisieren von Änderungen

Zugehörige Dokumente:

- Automating safe, hands-off deployments (Automatisierung sicherer, vollautomatischer Bereitstellungen)
- Nutzung von AWS CloudFormation zur Erstellung einer unveränderlichen Infrastruktur bei Nubank
- Infrastruktur als Code

- Implementieren eines Alarms zur automatischen Erkennung von Abweichungen in AWS CloudFormation-Stacks

Zugehörige Videos:

- AWS re:Invent 2020: Reliability, consistency, and confidence through immutability (AWS re:Invent 2020: Zuverlässlichkeit, Konsistenz und Vertrauen durch Unveränderlichkeit)

REL08-BP05 Automatisieren von Änderungen

Bereitstellungen und Patches werden automatisiert, um negative Auswirkungen zu vermeiden.

Änderungen an Produktionssystemen gehören in vielen Organisationen zu den größten Risikofaktoren. Neben den geschäftlichen Problemen, die durch die Software behoben werden, betrachten wir Bereitstellungen als vorrangiges Problem, das es zu lösen gilt. Heutzutage bedeutet das, wenn immer möglich und sinnvoll, Vorgänge zu automatisieren. Dazu gehören Tests und die Bereitstellung von Änderungen, das Hinzufügen oder Entfernen von Kapazität und das Migrieren von Daten.

Gewünschtes Ergebnis: Sie integrieren automatische Bereitstellungssicherheit in den Veröffentlichungsprozess mit umfangreichen Tests vor der Produktion, automatischen Rollbacks und gestaffelten Produktionsbereitstellungen. Diese Automatisierung minimiert die potenziellen Auswirkungen auf die Produktion, die durch fehlgeschlagene Bereitstellungen verursacht werden, und Entwickler müssen die Bereitstellungen nicht mehr aktiv bis zur Produktion beobachten.

Typische Anti-Muster:

- Sie führen manuelle Änderungen durch.
- Sie überspringen Schritte in Ihrer Automatisierung durch manuelle Notfall-Workflows.
- Sie folgen nicht Ihren etablierten Plänen und Prozessen zugunsten beschleunigter Zeitpläne.
- Sie führen schnelle Folgebereitstellungen durch, ohne entsprechende Bake-Zeit einzuräumen.

Vorteile der Einführung dieser bewährten Methode: Wenn Sie alle Änderungen mithilfe der Automatisierung implementieren, vermeiden Sie das Risiko menschlicher Fehler und bieten die Möglichkeit, Tests durchzuführen, bevor Sie die Produktion ändern. Wenn Sie diesen Vorgang vor dem Produktionsstart durchführen, wird sichergestellt, dass Ihre Pläne vollständig sind. Darüber

hinaus kann ein automatisches Rollback in Ihren Veröffentlichungsprozess Produktionsprobleme identifizieren und Ihren Workload wieder in den ursprünglichen Betriebszustand versetzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Automatisieren Sie Ihre Bereitstellungs-Pipeline. Mit Bereitstellungs-Pipelines können Sie Tests und die Entdeckung von Anomalien automatisieren und die Pipeline an einem bestimmten Schritt vor der Bereitstellung in der Produktion anhalten oder eine Änderung automatisch zurückführen. Ein integraler Bestandteil davon ist die Einführung einer Kultur der Continuous Integration und Continuous Delivery/Deployment (CI/CD), bei der ein Commit oder eine Codeänderung verschiedene automatische Stage-Gates von der Build- und Testphase bis zur Bereitstellung in Produktionsumgebungen durchläuft.

Obwohl es immer noch als sinnvoll erachtet wird, Personen bei den komplexesten betrieblichen Abläufen einzubinden, empfehlen wir, diese Abläufe wegen ihrer Komplexität zu automatisieren.

Implementierungsschritte

Sie können Bereitstellungen automatisieren, um manuelle Operationen zu vermeiden, indem Sie die folgenden Schritte ausführen:

- Einrichten eines Code-Repositorys zur Speicherung Ihres Codes: Verwenden Sie AWS CodeCommit, um ein sicheres Git-basiertes Repository zu erstellen.
- Konfigurieren eines Continuous-Integration-Services, um Ihren Quellcode zu kompilieren, Tests auszuführen und Bereitstellungsartefakte zu erstellen: Informationen zum Einrichten eines Build-Projekts für diesen Zweck finden Sie unter Einstieg in AWS CodeBuild mithilfe der Konsole.
- Einrichten eines Bereitstellungsservices, der Anwendungsbereitstellungen automatisiert und die Komplexität von Anwendungsupdates mindert, ohne auf fehleranfällige manuelle Bereitstellungen angewiesen zu sein: AWS CodeDeploy automatisiert Softwarebereitstellungen für eine Vielzahl von Computing-Services wie Amazon EC2, AWS Fargate, AWS Lambda und auf Ihren On-Premises-Servern. Informationen zur Konfiguration dieser Schritte finden Sie unter Einstieg in CodeDeploy.
- Konfigurieren eines Continuous-Integration-Services zur Automatisierung der Veröffentlichungspipelines für schnellere und zuverlässigere Anwendungs- und Infrastrukturupdates: Erwägen Sie die Verwendung von AWS CodePipeline, um die Automatisierung Ihrer Veröffentlichungspipelines zu unterstützen. Weitere Informationen finden Sie in den CodePipeline-Tutorials.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung
- OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung
- OPS06-BP02 Testbereitstellungen
- OPS06-BP04 Automatisieren von Tests und Rollback

Zugehörige Dokumente:

- Continuous Delivery von geschachtelten AWS CloudFormation-Stacks mit AWS CodePipeline
- Umfassende CI/CD mit AWS CodeCommit, AWS CodeBuild, AWS CodeDeploy und AWS CodePipeline
- APN-Partner: Partner, die Sie beim Erstellen automatisierter Bereitstellungslösungen unterstützen können
- AWS Marketplace: Produkte zur Automatisierung Ihrer Bereitstellungen
- Automatisieren von Chat-Nachrichten mit Webhooks
- Die Amazon Builders' Library: Rollback-Sicherheit bei Bereitstellungen gewährleisten
- Die Amazon Builders' Library: Schneller mit Continuous Delivery
- Was ist AWS CodePipeline?
- Was ist CodeDeploy?
- AWS Systems Manager Patch Manager
- Was ist Amazon SES?
- Was ist Amazon Simple Notification Service?

Zugehörige Videos:

- AWS Summit 2019: CI/CD auf AWS

Fehlerverwaltung

Fragen

- REL 9. Was ist bei der Sicherung von Daten zu beachten?
- REL 10. Wie schützen Sie Ihre Workload mithilfe der Fehlerisolierung?
- REL 11. Wie können Sie Workloads so gestalten, dass sie Komponentenausfällen gegenüber resilient sind?
- REL 12. Wie lässt sich die Zuverlässigkeit testen?
- REL 13. Was ist bei der Planung der Notfallwiederherstellung zu beachten?

REL 9. Was ist bei der Sicherung von Daten zu beachten?

Sichern Sie Daten, Anwendungen und Konfigurationen, um die Anforderungen im Hinblick auf das Recovery Time Objective (RTO, Wiederherstellungsdauer) und das Recovery Point Objective (RPO, Wiederherstellungszeitpunkt) zu erfüllen.

Bewährte Methoden

- REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen
- REL09-BP02 Schützen und Verschlüsseln von Backups
- REL09-BP03 Automatische Daten-Backups
- REL09-BP04 Verifizieren der Sicherungsintegrität und -verfahren durch regelmäßiges Wiederherstellen der Daten

REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen

Informieren Sie sich über die Backup-Funktionen der vom Workload genutzten Daten-Services und Ressourcen und nutzen Sie diese. Die meisten Services bieten Funktionen zur Sicherung von Workload-Daten.

Gewünschtes Ergebnis: Die Datenquellen wurden identifiziert und nach ihrer Bedeutung klassifiziert. Anschließend legen Sie eine auf dem RPO basierende Strategie für die Datenwiederherstellung fest. Diese Strategie involviert entweder die Sicherung dieser Datenquellen oder die Möglichkeit, Daten aus anderen Quellen zu reproduzieren. Im Falle eines Datenverlusts ermöglicht die implementierte Strategie die Wiederherstellung oder Reproduktion von Daten innerhalb der definierten RPO und RTO.

„Cloud-Reife“-Phase: Foundational

Typische Anti-Muster:

- Nicht alle Datenquellen für die Workload und deren Kritikalität sind bekannt.
- Es erfolgen keine Backups kritischer Datenquellen.
- Es erfolgen nur Backups von manchen Datenquellen ohne die Verwendung von Kritikalität als Kriterium.
- Es wurde kein RPO definiert oder die Backup-Häufigkeit kann den RPO nicht erfüllen.
- Es erfolgt keine Bewertung, ob ein Backup erforderlich ist oder ob Daten aus anderen Quellen reproduziert werden können.

Vorteile der Nutzung dieser bewährten Methode: Die Identifizierung der Stellen, an denen Backups erforderlich sind, und die Implementierung eines Mechanismus zur Erstellung von Backups oder die Möglichkeit, die Daten aus einer externen Quelle zu reproduzieren, verbessern die Fähigkeit zur Wiederherstellung und Wiederbeschaffung von Daten während eines Ausfalls.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Alle AWS-Datenspeicher bieten Backup-Möglichkeiten. Services wie Amazon RDS und Amazon DynamoDB unterstützen zusätzlich ein automatisiertes Backup, das eine zeitpunktbezogene Wiederherstellung (PITR) ermöglicht. So können Sie Backups zu einem beliebigen Zeitpunkt bis zu fünf Minuten oder weniger vor dem aktuellen Zeitpunkt wiederherstellen. Viele AWS-Services bieten die Möglichkeit, Backups in eine andere AWS-Region zu kopieren. AWS Backup ist ein Tool, das Ihnen die Möglichkeit gibt, den Schutz Ihrer Daten über AWS-Services hinweg zu zentralisieren und zu automatisieren. Mit AWS Elastic Disaster Recovery können Sie komplette Workloads von Servern kopieren und eine kontinuierliche Datensicherung von On-Premises-Ressourcen, AZ-übergreifenden Ressourcen oder Regionen hinweg aufrechterhalten. Das Recovery Point Objective (RPO) liegt dabei im Sekundenbereich.

Amazon S3 kann als Backup-Ziel für selbstverwaltete und AWS-verwaltete Datenquellen verwendet werden. AWS-Services wie Amazon EBS, Amazon RDS und Amazon DynamoDB bieten integrierte Möglichkeiten zur Backup-Erstellung. Sicherungssoftware von Drittanbietern kann ebenfalls eingesetzt werden.

On-Premises-Daten können mit AWS Storage Gateway oder AWS DataSync in der AWS Cloud gesichert werden. Mit Amazon S3-Buckets können Sie diese Daten auf speichern. Amazon S3 bietet

mehrere Speicherebenen wie Amazon S3 Glacier oder S3 Glacier Deep Archive, um die Kosten für den Datenspeicher zu senken.

Möglicherweise können Sie Ihre Datenwiederherstellungs-Anforderungen erfüllen, indem Sie Daten aus anderen Quellen reproduzieren. Zum Beispiel könnten Amazon ElastiCache-Replikat-Knoten oder Amazon RDS-Lesereplikate verwendet werden, um Daten zu reproduzieren, wenn der primäre Knoten verloren geht. In Fällen, in denen solche Quellen verwendet werden können, um Ihr Recovery Point Objective (RPO) und Recovery Time Objective (RTO) zu erfüllen, benötigen Sie möglicherweise kein Backup. Ein weiteres Beispiel: Wenn Sie mit Amazon EMR arbeiten, ist es möglicherweise nicht notwendig, ein Backup Ihres HDFS-Datenspeichers zu erstellen, solange Sie die Daten aus Amazon S3 in Amazon EMR wiederherstellen können.

Bei der Auswahl einer Backup-Strategie sollten Sie die für die Datenwiederherstellung benötigte Zeit berücksichtigen. Diese hängt von der Art des Backups (im Falle einer Backup-Strategie) oder von der Komplexität des Datenreproduktions-Mechanismus ab. Die benötigte Zeit sollte im RTO für die Workload liegen.

Implementierungsschritte

1. Identifizieren Sie alle Datenquellen für die Workload. Daten können über verschiedene Ressourcen wie Datenbanken, Volumes, Dateisysteme, Protokollierungssysteme und Objektspeicher gespeichert werden. Im Abschnitt Ressourcen finden Sie Verwandte Dokumente zu verschiedenen AWS-Services, mit denen Daten gespeichert werden, und zu den Backup-Möglichkeiten, die diese Services bieten.
2. Klassifizieren Sie Datenquellen basierend auf Kritikalität. Unterschiedliche Datensätze haben unterschiedliche Kritikalitäts-Niveaus für eine Workload und damit auch verschiedene Anforderungen an die Ausfallsicherheit. So können beispielsweise bestimmte kritische Daten einen RPO erfordern, der gegen Null geht, während bei anderen, weniger kritischen Daten, ein höherer RPO und somit ein gewisser Datenverlust toleriert werden kann. Ebenso können unterschiedliche Datensätze auch unterschiedliche RTO-Anforderungen haben.
3. Nutzen Sie AWS- oder Drittanbieter-Services, um Backups der Daten zu erstellen. AWS Backup ist ein verwalteter Service, der die Erstellung von Backups von verschiedenen Datenquellen auf AWS ermöglicht. https://aws.amazon.com/disaster-recovery/ übernimmt die automatisierte sekundengenaue Replikation von Daten in einer . Die meisten AWS-Services verfügen zusätzlich über native Funktionen zur Erstellung von Backups. Der AWS Marketplace umfasst zahlreiche Lösungen, die diese Funktionen ebenfalls bieten. In den unten aufgeführten Ressourcen finden Sie Informationen darüber, wie Sie Backups von Daten aus verschiedenen AWS-Services erstellen können.

4. Für Daten, die nicht gesichert werden, sollten Sie einen Datenreproduktions-Mechanismus festlegen. Es gibt verschiedene Gründe dafür, Daten, die aus anderen Quellen reproduziert werden können, nicht zu sichern. Möglicherweise ergibt sich die Situation, dass es günstiger ist, Daten bei Bedarf aus Quellen zu reproduzieren als ein Backup zu erstellen, da mit der Speicherung von Backups gewisse Kosten verbunden sind. Ein weiterer Grund wäre, wenn das Wiederherstellen aus einem Backup länger dauert als die Reproduktion der Daten aus anderen Quellen, was zu einer Nichteinhaltung des RTO führen würde. In solchen Situationen sollten Sie sich einen Kompromiss überlegen und einen gut definierten Prozess festlegen, wie Daten aus diesen Quellen reproduziert werden können, wenn eine Datenwiederherstellung erforderlich ist. Wenn Sie beispielsweise Daten zur Analyse aus Amazon S3 in ein Data Warehouse (wie Amazon Redshift) oder einen MapReduce-Cluster (wie Amazon EMR) geladen haben, kann es sich dabei z. B. um Daten handeln, die aus anderen Quellen reproduziert werden können. Solange die Ergebnisse dieser Analysen gespeichert werden oder reproduzierbar sind, besteht kein Risiko eines Datenverlusts durch einen Ausfall im Data Warehouse oder MapReduce-Cluster. Andere Daten, die aus Quellen reproduziert werden können, sind Cache-Inhalte (z. B. Amazon ElastiCache) oder RDS Read Replicas.
5. Legen Sie eine Kadenz für die Sicherung von Daten fest. Das Erstellen von Datenquellen ist ein periodischer Prozess und die Häufigkeit sollte vom RPO abhängen.

Grad des Aufwands für den Implementierungsplan: moderat.

Ressourcen

Zugehörige bewährte Methoden:

REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:

REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen

Zugehörige Dokumente:

- Was ist AWS Backup?
- Was ist AWS DataSync?
- Was ist Volume Gateway?
- APN-Partner: Partner, die Sie bei der Sicherung unterstützen können
- AWS Marketplace: Für die Sicherung geeignete Produkte

- Amazon EBS-Snapshots
- Backups von Amazon EFS
- Backups von Amazon FSx für Windows-Dateiserver
- Backup und Wiederherstellung für ElastiCache for Redis
- Erstellen eines DB-Cluster-Snapshots in Neptune
- Erstellen eines DB-Snapshots
- Erstellen einer EventBridge-Regel, die nach einem Zeitplan ausgelöst wird
- Regionsübergreifende Replikation mit Amazon S3
- EFS-zu-EFS AWS Backup
- Exportieren von Protokolldaten zu Amazon S3
- Verwaltung des Objektlebenszyklus
- On-Demand-Sicherung und Wiederherstellung in DynamoDB
- Zeitpunktbezogene Wiederherstellung für DynamoDB
- Mit Amazon OpenSearch Service Index-Snapshots arbeiten
- Was ist AWS Elastic Disaster Recovery?

Zugehörige Videos:

- AWS re: Ivent 2021 – Backup, disaster recovery, and ransomware protection with AWS (AWS re:Invent 2021 – Backup, Notfallwiederherstellung und Ransomware-Schutz mit AWS)
- AWS Backup Demo: Cross-Account and Cross-Region Backup (AWS Backup Demo: Konto- und regionsübergreifendes Backup)
- AWS re:Invent 2019: Ausführliche Beschreibung von AWS Backup, mit Rackspace (STG341)

Zugehörige Beispiele:

- Well-Architected Lab: Implementieren einer bidirektionalen Cross-Region Replication (CRR, regionsübergreifende Replikation) für Amazon S3
- Well-Architected Lab: Testen von Backup und Wiederherstellung von Daten
- Well-Architected Lab: Backup and Restore with Failback for Analytics Workload (Well-Architected Lab: Backups und Wiederherstellung mit Failback für Analytics-Workload)
- Well-Architected Lab: Notfallwiederherstellung – Backup und Wiederherstellung

REL09-BP02 Schützen und Verschlüsseln von Backups

Kontrollieren und erkennen Sie den Zugriff auf Backups durch eine Authentifizierung und Autorisierung. Vermeiden und erkennen Sie mittels Verschlüsselung Beeinträchtigungen der Datenintegrität von Backups.

Typische Anti-Muster:

- Derselbe Zugriff auf die Sicherungen und die automatisierte Wiederherstellung wie auf die Daten.
- Keine Verschlüsselung der Sicherungen.

Vorteile der Nutzung dieser bewährten Methode: Die Absicherung Ihrer Backups verhindert die Manipulation der Daten und die Verschlüsselung der Daten verhindert den Zugriff auf diese Daten, wenn sie versehentlich offengelegt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Steuern und erkennen Sie den Zugriff auf Backups durch Authentifizierung und Autorisierung wie z. B. mit AWS Identity and Access Management (IAM). Vermeiden und erkennen Sie mittels Verschlüsselung Beeinträchtigungen der Datenintegrität von Backups.

Amazon S3 unterstützt mehrere Verschlüsselungsmethoden für gespeicherte Daten. Mithilfe der serverseitigen Verschlüsselung akzeptiert Amazon S3 Ihre Objekte als unverschlüsselte Daten und sorgt für ihre Verschlüsselung bei der Speicherung. Bei der clientseitigen Verschlüsselung ist Ihre Workload-Anwendung für die Verschlüsselung der Daten verantwortlich, bevor sie an Amazon S3 gesendet werden. Beide Methoden ermöglichen Ihnen, zum Erstellen und Speichern des Datenschlüssels AWS Key Management Service (AWS KMS) zu verwenden oder einen eigenen Schlüssel bereitzustellen, für den Sie verantwortlich sind. Bei AWS KMS können Sie mithilfe von IAM festlegen, wer auf Ihre Datenschlüssel und entschlüsselten Daten zugreifen kann.

Wenn Sie bei Amazon RDS Ihre Datenbanken verschlüsseln, werden Ihre Sicherungsdaten ebenfalls verschlüsselt. DynamoDB-Sicherungen sind immer verschlüsselt. Bei Verwendung von AWS Elastic Disaster Recovery werden alle Daten während der Übertragung und im Ruhezustand verschlüsselt. Mit Elastic Disaster Recovery können Daten im Ruhezustand entweder mit dem standardmäßigen Amazon EBS-Volume-Verschlüsselungsschlüssel oder einem vom Kunden verwalteten Schlüssel verschlüsselt werden.

Implementierungsschritte

1. Verwenden Sie eine Verschlüsselung für jeden Datenspeicher. Wenn Ihre Quelldaten verschlüsselt sind, wird die Sicherung ebenfalls verschlüsselt.
   - Nutzen Sie die Verschlüsselung in Amazon RDS.. Beim Erstellen einer RDS-Instance können Sie die Verschlüsselung im Ruhezustand mit AWS Key Management Service konfigurieren.
   - Nutzen Sie die Verschlüsselung von Amazon EBS-Volumes.. Während der Erstellung von Volumes können Sie eine Standardverschlüsselung konfigurieren oder einen eindeutigen Schlüssel angeben.
   - Verwenden Sie die erforderliche Amazon DynamoDB-Verschlüsselung. DynamoDB verschlüsselt alle Daten im Ruhezustand. Sie können entweder einen AWS-eigenen AWS KMS-Schlüssel oder einen AWS-verwalteten KMS-Schlüssel verwenden und dabei einen Schlüssel angeben, der in Ihrem Konto gespeichert wird.
   - Verschlüsseln Sie Ihre in Amazon EFS gespeicherten Daten. Konfigurieren Sie die Verschlüsselung beim Erstellen des Dateisystems.
   - Konfigurieren Sie die Verschlüsselung in den Quell- und Zielregionen. Sie können die Verschlüsselung im Ruhezustand in Amazon S3 mit Schlüsseln konfigurieren, die in KMS gespeichert sind. Die Schlüssel sind jedoch regionsspezifisch. Sie können die Zielschlüssel angeben, während Sie die Replikation konfigurieren.
   - Entscheiden Sie sich für die Standardverschlüsselung oder die angepasste Amazon EBS-Verschlüsselung für Elastic Disaster Recovery. Mit dieser Option werden Ihre replizierten Daten im Ruhezustand auf den Staging-Area Subnetz-Datenträgern und den replizierten Datenträgern verschlüsselt.
2. Implementieren Sie Rechte mit geringsten Berechtigungen für den Zugriff auf Ihre Backups. Begrenzen Sie den Zugriff auf die Backups, Snapshots und Replikate anhand bewährter Methoden im Bereich Sicherheit.

Ressourcen

Zugehörige Dokumente:

- AWS Marketplace: Für die Sicherung geeignete Produkte
- Amazon EBS-Verschlüsselung.
- Amazon S3: Daten durch Verschlüsselung schützen
- Zusätzliche CRR-Konfiguration: Replizieren von Objekten, die mit serverseitiger Verschlüsselung (SSE) unter Verwendung von Verschlüsselungsschlüsseln erstellt wurden, die in AWS KMS gespeichert wurden.

- DynamoDB-Verschlüsselung im Ruhezustand
- Verschlüsseln von Amazon RDS-Ressourcen
- Encrypting Data and Metadata in Amazon EFS (Verschlüsseln von Daten und Metadaten in Amazon EFS)
- Verschlüsselung für Backups in AWS
- Verwalten verschlüsselter Tabellen
- Sicherheitssäule – AWS Well-Architected Framework
- Was ist AWS Elastic Disaster Recovery?

Zugehörige Beispiele:

- Well-Architected Lab: Implementieren einer bidirektionalen Cross-Region Replication (CRR, regionsübergreifende Replikation) für Amazon S3

REL09-BP03 Automatische Daten-Backups

Sie können die Backups so konfigurieren, dass sie automatisch nach Zeitplan, der auf dem Recovery Point Objective (RPO) basiert, oder bei Änderungen am Datensatz durchgeführt werden. Kritische Datasets, bei denen Datenverlust vermieden werden sollte, müssen regelmäßig automatisch gesichert werden, wohingegen weniger kritische Daten, bei denen ein gewisser Verlust akzeptabel ist, weniger häufig gesichert werden können.

Gewünschtes Ergebnis: Ein automatisierter Prozess, der Backups von Datenquellen in einem festgelegten Rhythmus erstellt.

Typische Anti-Muster:

- Sicherungen werden manuell durchgeführt.
- Es werden Ressourcen mit Sicherungsfunktionen verwendet, die Sicherung wird aber nicht in die Automatisierung einbezogen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Automatisierung von Backups wird sichergestellt, dass diese regelmäßig gemäß Ihrem RPO durchgeführt werden. Sie werden gewarnt, wenn sie nicht durchgeführt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

AWS Backup kann zum Erstellen von automatisierten Daten-Backups verschiedener AWS-Datenquellen genutzt werden. Amazon RDS-Instances können fast kontinuierlich alle fünf Minuten gesichert werden und Amazon S3-Objekte können praktisch durchgehend alle 15 Minuten gesichert werden, was eine zeitpunktbezogene Wiederherstellung (PITR) an einem bestimmten Zeitpunkt im Backup-Verlauf ermöglicht. Andere AWS-Datenquellen wie Amazon EBS-Volumes, Amazon DynamoDB-Tabellen oder Amazon FSx-Dateisysteme kann AWS Backup stündlich ein automatisiertes Backup ausführen. Diese Services bieten außerdem native Backup-Funktionen. Zu den AWS-Services, die ein automatisiertes Backup mit zeitpunktbezogener Wiederherstellung anbieten, gehören Amazon DynamoDB, Amazon RDS und Amazon Keyspaces (for Apache Cassandra). Diese können bis zu einem bestimmten Zeitpunkt innerhalb der Backup-Historie wiederhergestellt werden. Die meisten anderen AWS-Datenspeicher-Services bieten die Möglichkeit, stündliche periodische Backups einzuplanen.

Amazon RDS und Amazon DynamoDB bieten ein kontinuierliches Backup mit zeitpunktbezogener Wiederherstellung. Amazon S3 Sobald die Versionsverwaltung aktiviert ist, erfolgt sie automatisch. Mit Amazon Data Lifecycle Manager können Sie das Erstellen, Kopieren und Löschen von Amazon EBS-Snapshots automatisieren. Außerdem können damit das Erstellen, das Kopieren, die Außerbetriebnehmen und die Abmeldung von Amazon EBS-gestützten Amazon Machine Images (AMIs) und den zugrunde liegenden Amazon EBS-Snapshots automatisiert werden.

AWS Elastic Disaster Recovery bietet eine kontinuierliche Replikation auf Blockebene von der Quellumgebung (on-premises oder AWS) zur Ziel-Wiederherstellungsregion. Point-in-Time-Amazon EBS-Snapshots werden automatisch vom Service erstellt und verwaltet.

Für eine zentrale Ansicht Ihrer Sicherungsautomatisierung und des Verlaufs bietet AWS Backup eine vollständig verwaltete, richtlinienbasierte Sicherungslösung. Diese zentralisiert und automatisiert die Sicherung von Daten in mehreren AWS-Services in der Cloud sowie vor Ort mithilfe des AWS Storage Gateway.

Zusätzlich zum Versioning bietet Amazon S3 eine Replikationsfunktion. Der gesamte S3-Bucket kann automatisch in einen anderen Bucket in einer anderen AWS-Region repliziert werden.

Implementierungsschritte

1. Identifizieren Sie Datenquellen, die derzeit manuell gesichert werden. Weitere Details finden Sie unter REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen.

2. Bestimmen Sie das RPO für den Workload. Weitere Details finden Sie unter REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:.
3. Nutzen Sie eine automatisierte Backup-Lösung oder einen verwalteten Service. AWS Backup ist ein vollständig verwalteter Service, der die Zentralisierung und Automatisierung der Datensicherung über AWS-Services, in der Cloud und On-Premises erleichtert. Mithilfe von Backup-Plänen in AWS Backup erstellen Sie Regeln, die die zu sichernden Ressourcen und die Häufigkeit, mit der diese Backups erstellt werden sollen, festlegen. Diese Häufigkeit sollte auf dem in Schritt 2 festgelegten RPO basieren. Eine praktische Anleitung für die Erstellung automatisierter Backups mit AWS Backup finden Sie unter Testing Backup and Restore of Data (Testen von Backup und Wiederherstellung von Daten). Native Backup-Funktionen werden von den meisten AWS-Services, die Daten speichern, angeboten. So kann beispielsweise RDS für automatisierte Backups mit zeitpunktbezogener Wiederherstellung (PITR) genutzt werden.
4. Für Datenquellen, die nicht von einer automatisierten Backup-Lösung oder einem verwalteten Service unterstützt werden, wie z. B. On-Premises-Datenquellen oder Warteschlangen, sollten Sie eine zuverlässige Lösung eines Drittanbieters verwenden, um automatische Backups zu erstellen. Als Alternative können Sie die Automatisierung für diesen Vorgang mit der AWS CLI oder mit SDKs erstellen. Sie können AWS Lambda-Funktionen oder AWS Step Functions nutzen, um die Logik für die Erstellung eines Backups von Daten zu definieren und Amazon EventBridge einsetzen, um diese in einer Häufigkeit entsprechend Ihren RPOs auszuführen.

Grad des Aufwands für den Implementierungsplan: niedrig

Ressourcen

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Sicherung unterstützen können
- AWS Marketplace: Für die Sicherung geeignete Produkte
- Erstellen einer EventBridge-Regel, die nach einem Zeitplan ausgelöst wird
- Was ist AWS Backup?
- Was ist AWS Step Functions?
- Was ist AWS Elastic Disaster Recovery?

Zugehörige Videos:

- AWS re:Invent 2019: Ausführliche Beschreibung von AWS Backup, mit Rackspace (STG341)

Zugehörige Beispiele:

- Well-Architected Lab: Testen von Backup und Wiederherstellung von Daten

REL09-BP04 Verifizieren der Sicherungsintegrität und -verfahren durch regelmäßiges Wiederherstellen der Daten

Überprüfen Sie mit einem Wiederherstellungstest, ob sich mit Ihren Sicherungsverfahren das RTO und das RPO einhalten lassen.

Angestrebtes Ergebnis: Daten aus Backups werden regelmäßig mit genau definierten Mechanismen wiederhergestellt, um zu überprüfen, ob eine Wiederherstellung innerhalb des festgelegten Recovery Time Objectives (RTO) für den Workload möglich ist. Überprüfen Sie, dass die Wiederherstellung aus einem Backup in eine Ressource erfolgt, die die Originaldaten enthält und dass keine dieser Daten korrupt oder nicht zugänglich sind, sowie dass sich der Datenverlust im Rahmen des Recovery Point Objective (RPO) bewegt.

Typische Anti-Muster:

- Wiederherstellung eines Backups ohne Abfrage oder Abruf von Daten, um zu überprüfen, ob die Wiederherstellung funktionsfähig ist.
- Es wird angenommen, dass ein Backup existiert.
- Es wird angenommen, dass das Backup eines System voll funktionsfähig ist und Daten daraus wiederhergestellt werden können.
- Es wird angenommen, dass die Zeit für das Wiederherstellen von Daten aus einem Backup innerhalb des RTO für die Workload liegt.
- Es wird angenommen, dass die im Backup enthaltenen Daten in den RPO für die Workload fallen.
- Wiederherstellung bei Bedarf, ohne ein Runbook zu verwenden oder außerhalb eines etablierten automatisierten Verfahrens.

Vorteile der Nutzung dieser bewährten Methode: Das Testen der Wiederherstellung der Backups stellt sicher, dass die Daten bei Bedarf wiederhergestellt werden können, ohne dass Sie sich Sorgen um fehlende oder beschädigte Daten machen müssen, dass die Wiederherstellung innerhalb des RTOs für den Workload möglich ist und dass jeder Datenverlust innerhalb des RPOs für den Workload liegt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Das Testen der Sicherungs- und Wiederherstellungsfunktionen stärkt das Vertrauen in die Fähigkeit zur Durchführung dieser Aktionen während eines Ausfalls. Stellen Sie regelmäßig Backups an einem neuen Speicherort wieder her und führen Sie Tests aus, um die Datenintegrität zu überprüfen. Einige übliche Tests sind die Überprüfung, ob alle Daten verfügbar, nicht beschädigt und zugreifbar sind und ob ein Datenverlust innerhalb des RPO für den Workload liegt. Solche Tests können dabei helfen, zu ermitteln, ob die Wiederherstellungsmechanismen schnell genug sind, um dem RTO der Workload gerecht zu werden.

Mit AWS können Sie eine Testumgebung einrichten und Ihre Sicherungen wiederherstellen, um RTO- und RPO-Funktionen zu bewerten und Tests für Dateninhalte und Integrität durchzuführen.

Darüber hinaus ermöglichen Amazon RDS und Amazon DynamoDB eine Point-in-Time-Wiederherstellung. Durch die kontinuierliche Sicherung können Sie Ihren Datensatz in den Zustand zurücksetzen, in dem er sich an einem bestimmten Datum und zu einer bestimmten Uhrzeit befand.

Testen Sie, ob alle Daten verfügbar, nicht beschädigt und zugreifbar sind und ob ein Datenverlust innerhalb des RPOs für den Workload liegt. Solche Tests können dabei helfen, zu ermitteln, ob die Wiederherstellungsmechanismen schnell genug sind, um dem RTO der Workload gerecht zu werden.

AWS Elastic Disaster Recovery bietet eine kontinuierliche, zeitpunktbezogene Wiederherstellung von Snapshots von Amazon EBS-Volumes. Bei der Replikation von Quellservern werden die Point-in-Time-Zustände auf der Grundlage der konfigurierten Richtlinie im Laufe der Zeit aufgezeichnet. Elastic Disaster Recovery hilft Ihnen, die Integrität dieser Snapshots zu überprüfen, indem Sie Instances zu Test- und Übungszwecken starten, ohne den Datenverkehr weiterzuleiten.

Implementierungsschritte

1. Identifizieren Sie die Datenquellen, von denen derzeit ein Backup erstellt wird, und wo diese Backups gespeichert werden. Eine Anleitung zur Implementierung finden Sie unter REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen.
2. Etablieren von Kriterien zur Datenvalidierung für jede Datenquelle. Verschieden Datentypen können unterschiedliche Eigenschaften aufweisen und somit auch unterschiedliche Validierungsmechanismen erfordern. Überlegen Sie, wie diese Daten validiert werden können, bevor Sie sie in der Produktion einsetzen. Häufig werden für die Datenvalidierung Daten- und Sicherungseigenschaften wie Datentyp, Format, Prüfsumme, Größe oder eine Kombination dieser Eigenschaften mit einer benutzerdefinierten Validierungslogik verwendet. Ein Beispiel hierfür

wäre der Vergleich der Prüfsummenwerte zwischen der wiederhergestellten Ressource und der Datenquelle zum Zeitpunkt der Erstellung des Backups.

3. Etablieren des RTO und RPO für die Wiederherstellung der Daten basierend auf der Wichtigkeit der Daten. Eine Anleitung zur Implementierung finden Sie unter REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:.
4. Bewerten Sie die Funktion zur Datenwiederherstellung. Prüfen Sie Ihre Sicherungs- und Wiederherstellungsstrategie, um festzustellen, ob sie Ihre RTO und RPO erfüllen kann, und passen Sie die Strategie bei Bedarf an. Mit dem AWS Resilience Hub können Sie eine Bewertung Ihres Workloads vornehmen. Dabei wird Ihre Anwendungskonfiguration im Hinblick auf die Ausfallsicherheitsrichtlinien bewertet und Sie erfahren, ob Ihre RTO- und RPO-Ziele erfüllt werden können.
5. Führen Sie eine Testwiederherstellung durch, indem Sie die derzeit in der Produktion für die Wiederherstellung von Daten verwendeten Prozesse verwenden. Diese Prozesse hängen davon ab, wie die ursprüngliche Datenquelle gesichert wurde sowie vom Format und der Speicherung des Backups selbst oder davon, ob die Daten aus anderen Quellen reproduziert werden. Wenn Sie z. B. einen verwalteten Service wie AWS Backup verwenden, reicht es vielleicht aus, das Backup in einer neuen Ressource wiederherzustellen. Wenn Sie AWS Elastic Disaster Recovery verwendet haben, können Sie einen Recovery-Drill starten.
6. Validieren Sie die Datenwiederherstellung aus der wiederhergestellten Ressource anhand der Kriterien, die Sie zuvor für die Validierung der Daten festgelegt haben. Enthalten die wiederhergestellten Daten den neuesten Datensatz bzw. das neueste Element zum Zeitpunkt des Backups? Fallen diese Daten in das RPO für die Workload?
7. Messen Sie die benötigte Zeit für die Wiederherstellung und vergleichen Sie sie mit Ihrem festgelegten RTO. Ist dieser Prozess Teil des RTO für die Workload? Vergleichen Sie beispielsweise den Zeitstempel des Starts des Wiederherstellungsprozesses und des Abschlusses der Wiederherstellungsbewertung, um zu ermitteln, wie lange dieser Prozess dauert. Alle AWS-API-Aufrufe haben einen Zeitstempel. Sie finden diese Informationen in AWS CloudTrail. Während diese Informationen Details dazu liefern können, wann der Wiederherstellungsprozess gestartet wurde, sollte der End-Zeitstempel für den Abschluss der Validierung von der Validierungslogik aufgezeichnet werden. Wenn Sie einen automatisierten Prozess verwenden, können Sie Services wie Amazon DynamoDB nutzen, um diese Informationen zu speichern. Darüber hinaus können viele AWS-Services ein Ereignisprotokoll bereitstellen, das mit einem Zeitstempel versehene Informationen dazu enthält, wann bestimmte Aktionen aufgetreten sind. Innerhalb von AWS Backup werden Backup- und Wiederherstellungsaktionen als Jobs bezeichnet. Diese Jobs

enthalten als Teil ihrer Metadaten Zeitstempelinformationen, die zur Messung der für die Wiederherstellung benötigten Zeit verwendet werden können.

8. Benachrichtigen Sie die Stakeholder, wenn die Validierung der Daten fehlschlägt oder wenn die für die Wiederherstellung benötigte Zeit den festgelegten RTO für den Workload überschreitet. Bei der Implementierung einer entsprechenden Automatisierung, wie in dieser Übung, können Services wie Amazon Simple Notification Service (Amazon SNS) genutzt werden, um Push-Benachrichtigungen wie E-Mails oder SMS an Stakeholder zu senden. Diese Nachrichten können auch in Messaging-Anwendungen wie Amazon Chime, Slack oder Microsoft Teams veröffentlicht werden. Sie können zudem verwendet werden, um Aufgaben als OpsItems mit AWS Systems Manager OpsCenter zu erstellen.
9. Lassen Sie diesen Prozess regelmäßig automatisch ausführen. Sie können beispielsweise Services wie AWS Lambda oder einen Zustandsautomaten in AWS Step Functions nutzen, um die Wiederherstellungsprozesse zu automatisieren. Außerdem können Sie Amazon EventBridge verwenden, um diesen automatisierten Workflow regelmäßig auszulösen, wie im folgenden Architekturdiagramm abgebildet. Informieren Sie sich darüber, wie Sie die Validierung der Datenwiederherstellung mit AWS Backup automatisieren. Darüber hinaus bietet diese Well-Architected-Übung eine praxisorientierte Anleitung zur Automatisierung mehrerer der hier beschriebenen Schritte.

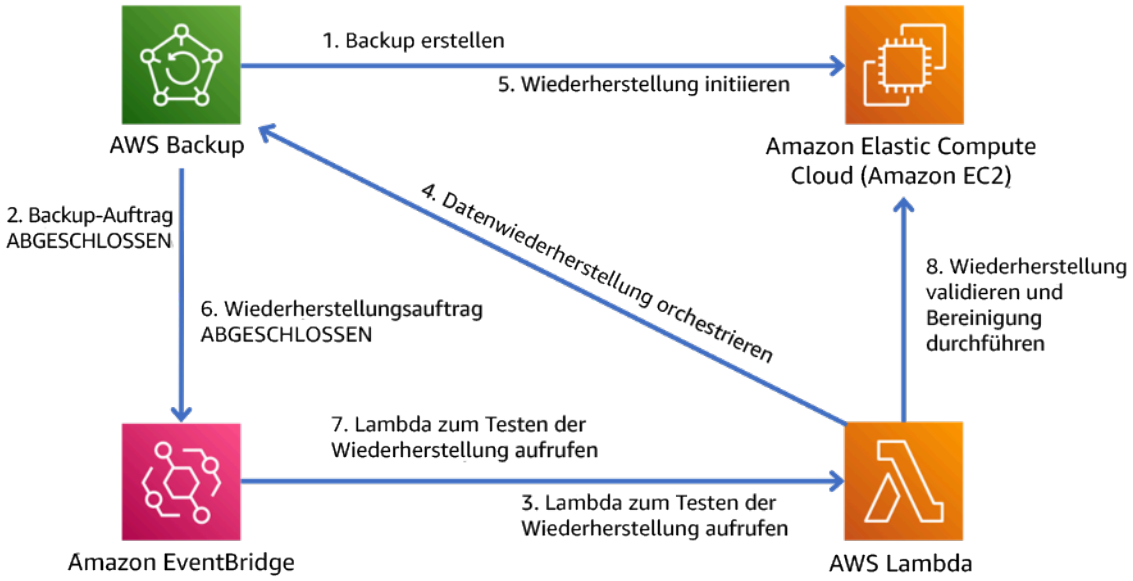


Abbildung 9. Ein automatisierter Sicherungs- und Wiederherstellungsprozess

Aufwandsniveau für den Implementierungsplan: Mäßig bis hoch, abhängig von der Komplexität der Validierungskriterien.

Ressourcen

Zugehörige Dokumente:

- Automatisieren der Datenwiederherstellung mit AWS Backup
- APN-Partner: Partner, die Sie bei der Sicherung unterstützen können
- AWS Marketplace: Für die Sicherung geeignete Produkte
- Erstellen einer EventBridge-Regel, die nach einem Zeitplan ausgelöst wird
- On-Demand-Sicherung und Wiederherstellung in DynamoDB
- Was ist AWS Backup?
- Was ist AWS Step Functions?
- Was ist AWS Elastic Disaster Recovery?
- AWS Elastic Disaster Recovery

Zugehörige Beispiele:

- Well-Architected Lab: Testen von Backup und Wiederherstellung von Daten

REL 10. Wie schützen Sie Ihre Workload mithilfe der Fehlerisolierung?

Fehlerisolierte Grenzen beschränken die Auswirkungen eines Ausfalls innerhalb eines Workloads auf eine begrenzte Anzahl von Komponenten. Komponenten außerhalb der Grenze sind vom Ausfall nicht betroffen. Wenn Sie mehrere fehlerisolierte Grenzen verwenden, können Sie die Auswirkungen auf Ihren Workload einschränken.

Bewährte Methoden

- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL10-BP02 Auswählen der geeigneten Standorte für Ihre Multi-Standort-Bereitstellung
- REL10-BP03 Automatisierte Wiederherstellung für Komponenten, die auf einen einzelnen Standort beschränkt sind
- REL10-BP04 Verwenden von Bulkhead-Architekturen, um den Umfang von Beeinträchtigungen zu begrenzen

REL10-BP01 Bereitstellen des Workloads an mehreren Standorten

Verteilen Sie die Workload-Daten und -Ressourcen über mehrere Availability Zones oder ggf. über mehrere AWS-Regionen. Die Standorte können so vielfältig wie nötig sein.

Eins der grundlegenden Prinzipien für das Servicedesign in AWS ist die Vermeidung von Single Points of Failure in der zugrunde liegenden physischen Infrastruktur. Dies treibt uns an, Software und Systeme zu entwickeln, die mehrere Availability Zones verwenden und Schutz beim Ausfall einer einzelnen Region bieten. Außerdem sollen Systeme gegen den Ausfall einzelner Compute-Knoten, einzelner Speicher-Volumes oder einzelner Instances einer Datenbank geschützt sein. Bei der Entwicklung eines Systems, das auf redundanten Komponenten basiert, muss gewährleistet sein, dass die Komponenten unabhängig voneinander betrieben werden und im Falle von AWS-Regionen autonom sind. Die Vorteile theoretischer Verfügbarkeitsberechnungen mit redundanten Komponenten sind nur anwendbar, wenn diese Voraussetzung erfüllt ist.

Availability Zones (AZs)

AWS-Regionen bestehen aus mehreren voneinander unabhängigen Availability Zones. Die einzelnen Availability Zones sind durch eine signifikante physische Distanz voneinander getrennt, um korrelierte Fehlerszenarios aufgrund von Umweltgefahren wie Feuer, Überflutungen und Tornados zu vermeiden. Jede Availability Zone verfügt außerdem über eine unabhängige physische Infrastruktur: eigene Verbindungen zur Stromversorgung, unabhängige Backup-Stromquellen, unabhängige mechanischen Services und unabhängige Netzwerkkonnektivität innerhalb der Availability Zone und darüber hinaus. Durch dieses Design bleiben Fehler in einem dieser Systeme auf die jeweils betroffene AZ beschränkt. Trotz ihrer geografischen Verteilung befinden sich Availability Zones in demselben regionalen Bereich, wodurch Netzwerke mit hohem Durchsatz und geringer Latenz ermöglicht werden. Die gesamte AWS-Region (über alle Availability Zones, die aus mehreren physisch unabhängigen Rechenzentren bestehen) kann wie ein logisches Bereitstellungsziel für Ihren Workload behandelt werden. Dies umfasst auch die Möglichkeit zum synchronen Replizieren von Daten (z. B. zwischen Datenbanken). So können Sie Availability Zones in einer Aktiv-Aktiv- oder einer Aktiv-Standby-Konfiguration nutzen.

Availability Zones sind voneinander unabhängig. Daher erhöht sich die Workload-Verfügbarkeit, wenn in der Architektur des Workloads mehrere Zonen verwendet werden. Einige AWS-Services (darunter auch die Amazon EC2-Instance-Datenebene) werden als strikte zonale Services bereitgestellt, die von denselben Fehlern betroffen sind wie die Availability Zone, in der sie sich befinden. Amazon EC2-Instances in den anderen AZs sind hingegen nicht betroffen und weiterhin funktionsfähig. Wenn entsprechend ein Fehler in einer Availability Zone zum Ausfall einer Amazon Aurora-Datenbank

führt, kann eine Auslese-Replikat-Aurora-Instance in einer nicht betroffenen AZ automatisch zur primären Instance hochgestuft werden. Regionale AWS-Services wie Amazon DynamoDB wiederum verwenden intern mehrere Availability Zones in einer Aktiv-Aktiv-Konfiguration, um die Verfügbarkeitsdesignziele für den jeweiligen Service zu erfüllen, ohne dass Sie die AZ-Platzierung konfigurieren müssen.

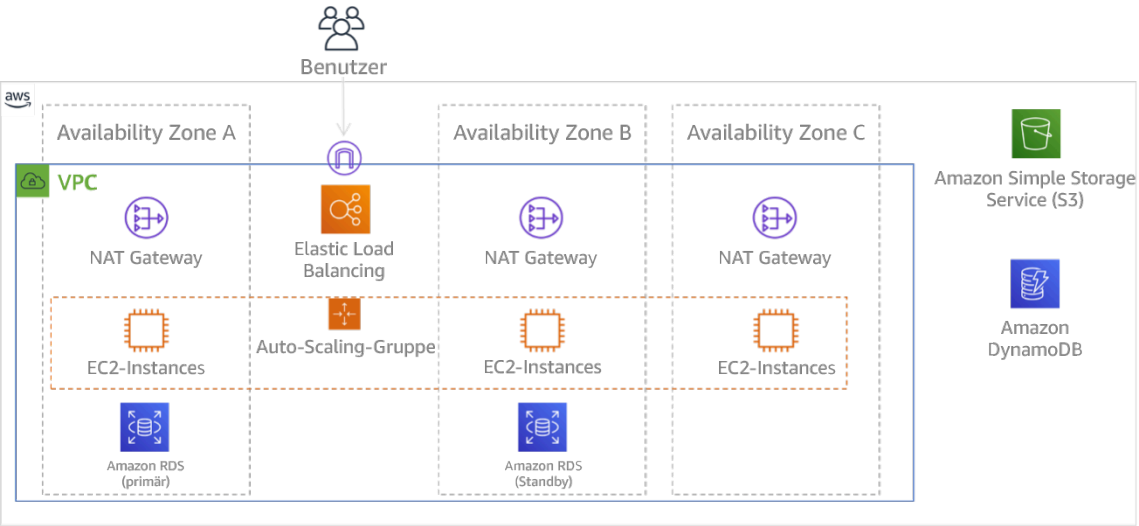


Abbildung 9: Mehrstufige Architektur, die in drei Availability Zones bereitgestellt wird. Amazon S3 und Amazon DynamoDB nutzen immer automatisch mehrere AZs. Auch der ELB wird in allen drei Zonen bereitgestellt.

Während Amazon EBS-Steuerebenen in der Regel die Möglichkeit bieten, Ressourcen innerhalb der gesamten Region (also in mehreren Availability Zones) zu verwalten, haben bestimmte Steuerebenen (wie AWS und Amazon EC2) die Fähigkeit, Ergebnisse in eine einzelne Availability Zone zu filtern. Wenn dies erledigt ist, wird die Anfrage nur in der angegebenen Availability Zone verarbeitet; dies reduziert die Wahrscheinlichkeit von Ausfällen in anderen Availability Zones. Dieses AWS CLI-Beispiel veranschaulicht das Abrufen von Amazon EC2-Instance-Informationen ausschließlich aus der Availability Zone „us-east-2c“:

```
AWS ec2 describe-instances --filters Name=availability-zone,Values=us-east-2c
```

AWS Local Zones

AWS Local Zones verhalten sich ähnlich wie Availability Zones innerhalb ihrer jeweiligen AWS-Region. Sie können als Platzierungsstandort für zonale AWS-Ressourcen wie Subnetze und EC2-Instances ausgewählt werden. Das Besondere daran ist, dass sie sich nicht in der zugehörigen AWS-Region befinden, sondern in der Nähe großer Ballungsräume, Industrie- und IT-Zentren, in denen

derzeit keine AWS-Region vorhanden ist. Sie sorgen dennoch für eine sichere Verbindung mit hoher Bandbreite zwischen lokalen Workloads in der lokalen Zone und Workloads in der AWS-Region. Sie sollten AWS Local Zones verwenden, um Workloads mit Anforderungen an eine geringe Latenz näher bei Ihren Benutzern bereitzustellen.

Amazon Global Edge Network

Amazon Global Edge Network besteht aus Edge-Standorten in Städten auf der ganzen Welt. Amazon CloudFront nutzt dieses Netzwerk, um Inhalte mit geringerer Latenz für Endbenutzer bereitzustellen. Mit AWS Global Accelerator können Sie Ihre Workload-Endpunkte an diesen Edge-Standorten erstellen, um ein Onboarding in das globale AWS-Netzwerk in der Nähe Ihrer Benutzer zu ermöglichen. Amazon API Gateway können Sie Edge-optimierte API-Endpunkte mithilfe einer CloudFront-Verteilung verwenden, um den Client-Zugriff über den nächstgelegenen Edge-Standort zu erleichtern.

AWS-Regionen

AWS-Regionen sind autonom konzipiert. Daher können Sie dedizierte Kopien von Services für jede Region bereitstellen, um einen multiregionalen Ansatz zu verwenden.

Ein multiregionaler Ansatz wird häufig für Strategien der Notfallwiederherstellung eingesetzt, um Wiederherstellungsziele zu erfüllen, falls einmalige Ereignisse mit großer Reichweite auftreten. Siehe Planung der Notfallwiederherstellung für weitere Informationen zu diesen Strategien. Hier liegt der Schwerpunkt allerdings auf der Verfügbarkeit, wobei versucht wird, ein mittleres Betriebszeitziel über einen längeren Zeitraum zu erreichen. Wenn eine hohe Verfügbarkeit angestrebt wird, ist eine multiregionale Architektur normalerweise Aktiv-Aktiv konzipiert. Dabei sind die einzelnen Servicekopien (in den jeweiligen Regionen) aktiv (und bearbeiten Anfragen).

> **Empfehlung**
>
> Die Verfügbarkeitsziele für die meisten Workloads können mithilfe einer Multi-AZ-Strategie innerhalb einer einzelnen AWS-Region erfüllt werden. Ziehen Sie multiregionale Architekturen nur in Betracht, wenn für Workloads extreme Verfügbarkeitsanforderungen gelten oder andere Unternehmensziele eine solche Architektur erforderlich machen.

AWS bietet Ihnen die Möglichkeit, Services regionsübergreifend zu betreiben. AWS stellt beispielsweise eine fortlaufende asynchrone Datenreplikation mit Amazon S3-Replikation (Amazon Simple Storage Service), Amazon RDS-Lesereplikaten (u. a. Aurora-Lesereplikaten) und globalen

Amazon DynamoDB-Tabellen bereit. Bei der fortlaufenden Replikation sind Versionen Ihrer Daten für die fast sofortige Nutzung in jeder aktiven Region verfügbar.

Mit AWS CloudFormation können Sie Ihre Infrastruktur definieren und einheitlich in AWS-Konten und AWS-Regionen bereitstellen. AWS CloudFormation StackSets erweitern diese Funktionen, indem Sie AWS CloudFormation-Stacks mit nur einem Vorgang in verschiedenen Konten und Regionen erstellen, aktualisieren oder löschen können. Bei Amazon EC2-Instance-Bereitstellungen wird ein AMI (Amazon Machine Image) verwendet, um Informationen wie die Hardwarekonfiguration und installierte Software bereitzustellen. Sie können eine Amazon EC2 Image Builder-Pipeline implementieren, die die benötigten AMIs erstellt, und diese in Ihre aktiven Regionen kopieren. Diese goldenen AMIs enthalten alles, was Sie zum Bereitstellen und Skalieren von Workloads in neuen Regionen benötigen.

Zum Weiterleiten von Datenverkehr ermöglichen sowohl Amazon Route 53 als auch AWS Global Accelerator das Definieren von Richtlinien, die angeben, welche Benutzer zu welchem aktiven regionalen Endpunkt geleitet werden. Mit Global Accelerator legen Sie für den Datenverkehr einen Prozentwert fest, der an die einzelnen Anwendungsendpunkte geleitet wird. Route 53 unterstützt diesen Ansatz mit Prozentwerten sowie eine Vielzahl weiterer Richtlinien, u. a. auf Grundlage der geografischen Nähe oder der Latenz. Global Accelerator nutzt automatisch das umfassende Netzwerk von AWS-Edge-Servern, um Datenverkehr an den Backbone des AWS-Netzwerks zu senden, sobald dies möglich ist. Dies führt zu einer geringeren Latenz bei Abfragen.

Alle diese Funktionen sind so konzipiert, dass die Autonomie der einzelnen Regionen erhalten wird. Es gibt nur sehr wenige Ausnahmen von diesem Ansatz, darunter unsere Services für eine weltweite Edge-Lieferung (z. B. Amazon CloudFront und Amazon Route 53) und die Steuerebene für den AWS Identity and Access Management-Service (IAM). Die meisten Services werden vollständig innerhalb einer einzigen Region betrieben.

On-Premises-Rechenzentrum

Für Workloads, die in einem On-Premises-Rechenzentrum ausgeführt werden, sollten Sie nach Möglichkeit eine hybride Umgebung erstellen. AWS Direct Connect bietet eine dedizierte Netzwerkverbindung zwischen Ihrem Standort und AWS, sodass eine Ausführung in beiden Umgebungen möglich ist.

Außerdem haben Sie die Möglichkeit, AWS-Infrastruktur und -Services mit AWS Outposts lokal auszuführen. AWS Outposts ist ein vollständig verwalteter Service, der die AWS-Infrastruktur, AWS-Services, APIs und Tools auf Ihr Rechenzentrum erweitert. Die gleiche Hardwareinfrastruktur, die in der AWS Cloud verwendet wird, wird dafür in Ihrem Rechenzentrum installiert. AWS Outposts werden

dann mit der nächstgelegenen AWS-Region verbunden. Anschließend können Sie AWS Outposts verwenden, um Workloads mit geringer Latenz oder lokalen Datenverarbeitungsanforderungen zu unterstützen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Verwenden Sie mehrere Availability Zones und AWS-Regionen. Verteilen Sie die Workload-Daten und -Ressourcen über mehrere Availability Zones oder ggf. über mehrere AWS-Regionen. Die Standorte können so vielfältig wie nötig sein.
  - Regionale Services werden von Haus aus in Availability Zones bereitgestellt.
    - Dazu gehören Amazon S3, Amazon DynamoDB und AWS Lambda (wenn keine VPC-Verbindung vorhanden ist).
  - Stellen Sie Ihre Container-, Instance- und funktionsbasierten Workloads in mehreren Availability Zones bereit. Verwenden Sie Multi-AZ-Datenspeicher, einschließlich Cache. Nutzen Sie EC2 Auto Scaling, die ECS-Aufgabenplatzierung, ElastiCache-Cluster sowie bei Ausführung in Ihrer VPC AWS Lambda-Funktionen.
    - Verwenden Sie für die Bereitstellung von Auto-Scaling-Gruppen Subnetze in getrennten Availability Zones.
      - Beispiel: Verteilen von Instances in Availability Zones
      - Strategien zur Aufgabenplatzierung mit Amazon ECS
      - Konfigurieren einer AWS Lambda-Funktion für den Zugriff auf Ressourcen in einer Amazon VPC
      - Auswählen von Regionen und Availability Zones
    - Verwenden Sie für die Bereitstellung von Auto-Scaling-Gruppen Subnetze in getrennten Availability Zones.
      - Beispiel: Verteilen von Instances in Availability Zones
    - Verwenden Sie ECS-Parameter für die Platzierung von Aufgaben unter Angabe von DB-Subnetzgruppen.
      - Strategien zur Aufgabenplatzierung mit Amazon ECS
    - Nutzen Sie Subnetze in mehreren Availability Zones, wenn Sie eine in Ihrem VPC auszuführende Funktion konfigurieren.
      - Konfigurieren einer AWS Lambda-Funktion für den Zugriff auf Ressourcen in einer Amazon VPC

    - Verwenden Sie mehrere Availability Zones mit ElastiCache-Clustern.
        - Auswählen von Regionen und Availability Zones
- Wenn Ihr Workload für mehrere Regionen bereitgestellt werden muss, sollten Sie sich für eine Strategie mit mehreren Regionen entscheiden. Die meisten Zuverlässigkeitsanforderungen können mithilfe einer Multi-Availability-Zone-Strategie innerhalb einer einzelnen AWS-Region erfüllt werden. Verwenden Sie eine Multi-Regionen-Strategie, wenn notwendig, um Ihre Geschäftsanforderungen zu erfüllen.
    - AWS re:Invent 2018: Architekturmuster für Aktiv-Aktiv-Anwendungen in mehreren Regionen (ARC209-R2)
        - Ein Backup in einer anderen AWS-Region kann zusätzliche Gewissheit bieten, dass Daten verfügbar sind, wenn sie benötigt werden.
        - Für einige Workloads gibt es gesetzliche Anforderungen, die eine Multi-Region-Strategie erfordern.
- Evaluieren Sie AWS Outposts für Ihren Workload. Wenn Ihre Workload eine niedrige Latenz für Ihr Rechenzentrum vor Ort erfordert oder lokale Datenverarbeitungsanforderungen hat. Führen Sie anschließend AWS-Infrastruktur und -Services On-Premises mit AWS Outposts aus.
    - Was ist AWS Outposts?
- Ermitteln Sie, ob AWS Local Zones Sie bei der Bereitstellung von Services für Ihre Benutzer unterstützt. Wenn Sie Anforderungen an eine geringe Latenz haben, prüfen Sie, ob sich AWS Local Zones in der Nähe Ihrer Benutzer befindet. Wenn dies der Fall ist, stellen Sie damit Workloads näher an diesen Benutzern bereit.
    - AWS Local Zones – häufig gestellte Fragen

Ressourcen

Ähnliche Dokumente:

- Globale AWS-Infrastruktur
- AWS Local Zones – häufig gestellte Fragen
- Strategien zur Aufgabenplatzierung mit Amazon ECS
- Auswählen von Regionen und Availability Zones
- Beispiel: Verteilen von Instances in Availability Zones
- Globale Tabellen: Multiregionale Replikation mit DynamoDB
- Verwenden von Amazon Aurora Global Databases

- Blog-Reihe: Creating a Multi-Region Application with AWS Services (Erstellen einer Multi-Region-Anwendung mit AWS-Services)
- Was ist AWS Outposts?

Relevante Videos:

- AWS re:Invent 2018: Architekturmuster für Aktiv-Aktiv-Anwendungen in mehreren Regionen (ARC209-R2)
- AWS re:Invent 2019: Innovation und Betrieb der globalen Netzwerkinfrastruktur von AWS (NET339)

REL10-BP02 Auswählen der geeigneten Standorte für Ihre Multi-Standort-Bereitstellung

Gewünschtes Ergebnis

Für eine hohe Verfügbarkeit stellen Sie Ihre Workload-Komponenten (falls möglich) immer in mehreren Availability Zone (AZ) bereit, wie in Abbildung 10 dargestellt. Überdenken Sie bei Workloads mit extremen Anforderungen an die Ausfallsicherheit die Optionen für eine Multi-Region-Architektur genau.

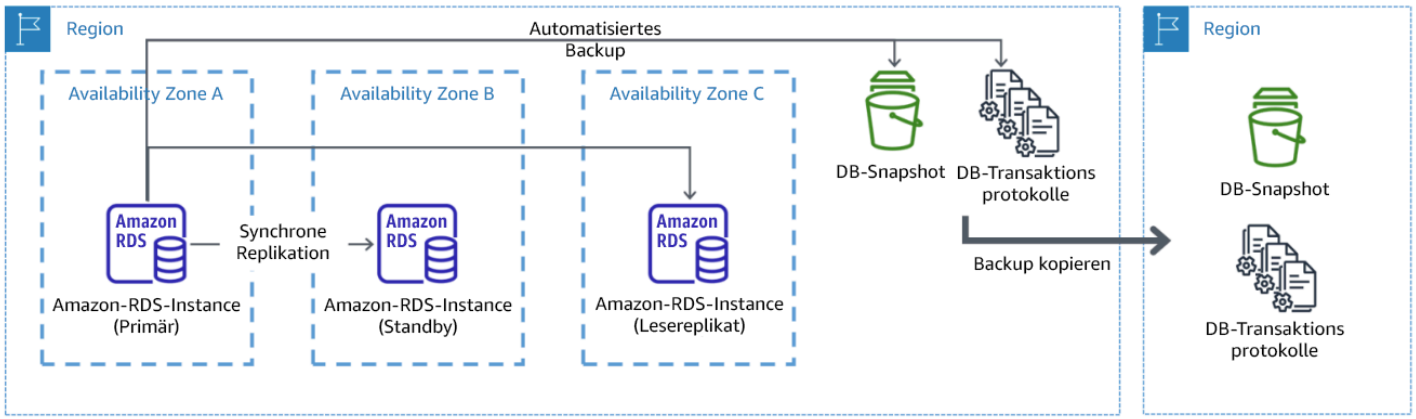


Abbildung 10: Resiliente Multi-AZ-Datenbankbereitstellung mit Backup in einer anderen AWS-Region

Gängige Antimuster

- Entscheidung für das Design einer Multi-Region-Architektur, wenn eine Multi-AZ-Architektur für die Anforderungen ausreichend wäre.

- Fehlende Berücksichtigung der Abhängigkeiten zwischen Anwendungskomponenten, wenn diese Komponenten unterschiedliche Anforderungen im Bezug auf Ausfallsicherheit und mehrere Standorte aufweisen.

Vorteile der Einführung dieser bewährten Methode:

Für die Ausfallsicherheit sollten Sie einen Ansatz wählen, bei dem verschiedene Verteidigungsebenen aufgebaut werden. Eine Ebene schützt vor kleineren, häufiger auftretenden Unterbrechungen, indem eine hochverfügbare Architektur mit mehreren AZs erstellt wird. Eine weitere Verteidigungsebene schützt vor seltenen Ereignissen wie Naturkatastrophen mit großer Reichweite und Unterbrechungen auf Regionsebene. Für diese zweite Ebene muss die Architektur Ihrer Anwendung mehrere AWS-Regionen umfassen.

- Der Unterschied zwischen einer Verfügbarkeit von 99,5 % und 99,99 % beträgt über 3,5 Stunden pro Monat. Die erwartete Verfügbarkeit eines Workloads kann nur „four nines“ (d. h. 99,99 %) erreichen, wenn er sich in mehreren AZs befindet.
- Indem Sie einen Workload in mehreren AZs ausführen, können Sie Fehler bei der Stromversorgung, Kühlung, im Netzwerk sowie die meisten Naturkatastrophen wie Feuer und Überflutung isolieren.
- Wenn Sie eine Multi-Region-Strategie für Ihren Workload implementieren, ist er vor weitreichenden Naturkatastrophen, die einen großen geografischen Bereich in einem Land betreffen, oder technischen Fehlern in einer ganzen Region geschützt. Beachten Sie dabei, dass das Implementieren einer Multi-Region-Architektur äußerst komplex sein kann und bei den meisten Workloads nicht erforderlich ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Bei einer Unterbrechung oder dem teilweisen Ausfall einer Availability Zone hilft die Implementierung eines hoch verfügbaren Workloads in mehreren Availability Zones innerhalb einer einzelnen AWS-Region, die Folgen von Naturkatastrophen oder technischen Problemen zu begrenzen. Jede AWS-Region besteht aus mehreren Availability Zones, die von Fehlern in den jeweils anderen Zonen isoliert sind und die eine deutliche Distanz aufweisen. In Bezug auf Notfallereignisse, bei denen das Risiko des Ausfalls mehrerer, voneinander weit entfernter Availability-Zone-Komponenten besteht, sollten Sie Optionen für die Notfallwiederherstellung implementieren. So können Sie Fehler eingrenzen, die sich auf eine ganze Region auswirken. Bei Workloads, für die eine extreme

Ausfallsicherheit erforderlich ist (kritische Infrastruktur, gesundheitsbezogene Anwendungen, Infrastruktur von Finanzsystemen usw.) wird möglicherweise eine Multi-Region-Strategie benötigt.

Implementierungsschritte

1. Analysieren Sie Ihren Workload und bestimmen Sie, ob die Anforderungen an die Ausfallsicherheit mit einem Multi-AZ-Ansatz erfüllt werden (eine AWS-Region) oder ob ein Multi-Region-Ansatz erforderlich ist. Das Implementieren einer Multi-Region-Architektur, um diese Anforderungen zu erfüllen, führt zu einer höheren Komplexität. Betrachten Sie daher Ihren Anwendungsfall und wägen Sie die Anforderungen sorgfältig ab. Die Anforderungen an die Ausfallsicherheit können fast immer auch mit einer AWS-Region erfüllt werden. Berücksichtigen Sie bei der Entscheidung, ob Sie mehrere Regionen verwenden möchten, die folgenden möglichen Anforderungen:
   a. Notfallwiederherstellung (Disaster Recovery, DR): Bei einer Unterbrechung oder dem teilweisen Ausfall einer Availability Zone hilft die Implementierung eines hoch verfügbaren Workloads in mehreren Availability Zones innerhalb einer einzelnen AWS-Region, die Folgen von Naturkatastrophen oder technischen Problemen zu begrenzen. In Bezug auf Notfallereignisse, bei denen das Risiko des Ausfalls mehrerer, voneinander weit entfernter Availability Zone-Komponenten besteht, sollten Sie eine Notfallwiederherstellung in mehreren Regionen implementieren. So können Sie die Risiken durch Naturkatastrophen oder technische Fehler eingrenzen, die sich auf eine ganze Region auswirken.
   b. Hohe Verfügbarkeit (High Availability, HA): Mit einer Multi-Region-Architektur (mit mehreren AZs in jeder Region) kann eine höhere Verfügbarkeit als „four 9's“ (> 99,99 %) erreicht werden.
   c. Stack-Lokalisierung: Beim Bereitstellen eines Workloads für Benutzer weltweit können Sie lokalisierte Stacks in verschiedenen AWS-Regionen bereitstellen, um die Benutzer in diesen Regionen zu versorgen. Die Lokalisierung kann Sprache, Währung und die gespeicherten Datentypen umfassen.
   d. Nähe zu den Benutzern: Wenn Sie einen Workload für Benutzer weltweit bereitstellen, können Sie die Latenz reduzieren, indem Sie Stacks in AWS-Regionen in der Nähe der Endbenutzer bereitstellen.
   e. Datenresidenz: Für einige Workloads gelten Anforderungen an die Datenresidenz, d. h. die Daten von bestimmten Nutzern müssen innerhalb der Grenzen eines bestimmten Landes gespeichert werden. Abhängig von der jeweiligen Regelung können Sie einen ganzen Stack oder nur die Daten in der AWS-Region innerhalb dieser Landesgrenzen bereitstellen.
2. Im Folgenden finden Sie einige Bespiele für Multi-AZ-Funktionen, die von AWS-Services bereitgestellt werden:

a. Um Workloads mit EC2 oder ECS zu schützen, stellen Sie einen Elastic Load Balancer vor den Datenverarbeitungsressourcen bereit. Elastic Load Balancing bietet so die Lösung, um Instances in fehlerhaften Zonen zu erkennen und den Datenverkehr zu fehlerfreien Zonen zu leiten.

   i. Erste Schritte mit Application Load Balancers

   ii. Erste Schritte mit Network Load Balancers

b. Bei EC2-Instances, auf denen kommerzielle Standardsoftware ohne Unterstützung für Load Balancing ausgeführt wird, können Sie eine gewisse Fehlertoleranz durch die Implementierung einer Methodologie für die Multi-AZ-Notfallwiederherstellung erreichen.

   i. the section called “REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen”

c. Stellen Sie für Amazon ECS-Aufgaben den Service gleichmäßig auf drei AZs verteilt bereit, um eine ausgeglichene Verteilung von Verfügbarkeit und Kosten zu erreichen.

   i. Bewährte Methoden für die Amazon ECS-Verfügbarkeit | Container

d. Wenn Sie nicht mit Aurora Amazon RDS arbeiten, können Sie Multi-AZ als Konfigurationsoption auswählen. Beim Ausfall der primären Datenbank-Instance stuft Amazon RDS automatisch eine Standby-Datenbank hoch, sodass sie Datenverkehr in einer anderen Availability Zone empfangen kann. Außerdem können Multi-Region-Lesereplikate erstellt werden, um die Ausfallsicherheit zu steigern.

   i. Amazon RDS-Multi-AZ-Bereitstellungen

   ii. Erstellen eines Lesereplikats in einer anderen AWS-Region

3. Im Folgenden finden Sie einige Bespiele für Multi-Region-Funktionen, die von AWS-Services bereitgestellt werden:

   a. Für Amazon S3-Workloads, bei denen Multi-AZ-Verfügbarkeit automatisch vom Service bereitgestellt wird, erwägen Sie Multi-Region-Zugriffspunkte, wenn eine Multi-Region-Bereitstellung benötigt wird.

      i. Multi-Region-Zugriffspunkte in Amazon S3

   b. Wenn bei DynamoDB-Tabellen Multi-AZ-Verfügbarkeit automatisch vom Service bereitgestellt wird, können Sie vorhandene Tabellen problemlos in globale Tabellen konvertieren, um mehrere Regionen nutzen zu können.

      i. Konvertieren von Amazon DynamoDB-Tabellen für eine Region in globale Tabellen

c. Wenn Ihr Workload hinter Application Load Balancers oder Network Load Balancers liegt, verwenden Sie AWS Global Accelerator, um die Verfügbarkeit Ihrer Anwendung zu verbessern, indem Sie Datenverkehr zu mehreren Regionen mit fehlerfreien Endpunkten leiten.
   i. Endpunkte für Standard-Accelerators in AWS Global Accelerator – AWS Global Accelerator (amazon.com)
d. Erwägen Sie bei Anwendungen, die AWS EventBridge nutzen, die Verwendung von regionsübergreifenden Buses, um Ereignisse an ausgewählte Regionen weiterzuleiten.
   i. Senden und Empfangen von Amazon EventBridge-Ereignissen zwischen AWS-Regionen
e. Ziehen Sie bei Amazon Aurora-Datenbanken globale Aurora-Datenbanken in Erwägungen, die mehrere AWS-Regionen umfassen können. Vorhandene Cluster können ebenfalls geändert werden, um neue Regionen hinzuzufügen.
   i. Erste Schritte mit globalen Amazon Aurora-Datenbanken
f. Wenn Ihr Workload AWS Key Management Service-Verschlüsselungsschlüssel (AWS KMS) umfasst, überlegen Sie, ob Multi-Region-Schlüssel für Ihre Anwendung geeignet sind.
   i. Multi-Region-Schlüssel in AWS KMS
g. Weitere Funktionen von AWS-Services finden Sie in dieser Blog-Reihe zum Erstellen einer Multi-Region-Anwendung mit AWS-Services

Grad des Aufwands für den Implementierungsplan: Mittel bis hoch

Ressourcen

Ähnliche Dokumente:

- Erstellen einer Multi-Region-Anwendung mit AWS-Services
- Disaster Recovery (DR) Architecture on AWS, Part IV: Multi-site Active/Active (Architektur für die Notfallwiederherstellung (Disaster Recovery, DR) in AWS, Teil IV: Multi-Site Aktiv-Aktiv)
- Globale AWS-Infrastruktur
- AWS Local Zones – häufig gestellte Fragen
- Architektur für die Notfallwiederherstellung in AWS, Teil I: Strategien für die Wiederherstellung in der Cloud
- Die Notfallwiederherstellung in der Cloud unterscheidet sich
- Globale Tabellen: Multiregionale Replikation mit DynamoDB

Relevante Videos:

- AWS re:Invent 2018: Architekturmuster für Aktiv-Aktiv-Anwendungen in mehreren Regionen (ARC209-R2)
- Auth0: multiregionale Architektur mit hoher Verfügbarkeit, die auf mehr als 1,5 Milliarden Anmeldungen pro Monat mit automatisiertem Failover skaliert werden kann.

Ähnliche Beispiele:

- Architektur für die Notfallwiederherstellung in AWS, Teil I: Strategien für die Wiederherstellung in der Cloud
- DTCC erzielt Resilienz weit über das hinaus, was On-Premises möglich wäre
- Expedia Group nutzt eine Architektur mit mehreren Regionen und Availability Zones und einem proprietären DNS-Service, um den Anwendungen Resilienz hinzuzufügen.
- Uber: Notfallwiederherstellung für multiregionales Kafka
- Netflix: Aktiv-Aktiv für multiregionale Resilienz
- Entwicklung von Data Residency für Atlassian Cloud
- Intuit TurboTax wird über zwei Regionen ausgeführt

REL10-BP03 Automatisierte Wiederherstellung für Komponenten, die auf einen einzelnen Standort beschränkt sind

Wenn Komponenten des Workloads nur in einer einzigen Availability Zone oder in einem On-Premises-Rechenzentrum ausgeführt werden können, implementieren Sie die Möglichkeit, den Workload innerhalb Ihrer definierten Wiederherstellungsziele komplett neu aufzusetzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wenn die bewährte Methode zur Bereitstellung des Workloads an mehreren Standorten aufgrund technologischer Einschränkungen nicht möglich ist, müssen Sie einen alternativen Pfad zur Ausfallsicherheit implementieren. Sie müssen die Möglichkeit automatisieren, die erforderliche Infrastruktur neu zu erstellen, Anwendungen neu bereitzustellen und die erforderlichen Daten für diese Fälle neu zu erstellen.

Amazon EMR startet beispielsweise alle Knoten für einen bestimmten Cluster in derselben Availability Zone, da die Ausführung eines Clusters in derselben Zone eine höhere Datenzugriffsrate bietet und dadurch eine höhere Leistung für die Aufgabenbearbeitung bereitstellt. Wenn diese Komponente für die Ausfallsicherheit von Workloads erforderlich ist, müssen Sie die Möglichkeit haben, den Cluster und seine Daten erneut bereitzustellen. Für Amazon EMR sollten Sie nicht nur Multi-AZs verwenden, um für Redundanz zu sorgen. Sie können mehrere Knoten bereitstellen. Mit EMR File System (EMRFS) können Daten in EMR in Amazon S3 gespeichert werden, das wiederum über mehrere Availability Zones oder AWS-Regionen repliziert werden kann.

Ähnlich wie bei Amazon Redshift wird Ihr Cluster standardmäßig in einer zufällig ausgewählten Availability Zone innerhalb der ausgewählten AWS-Region bereitgestellt. Alle Cluster-Knoten werden in derselben Zone bereitgestellt.

Für zustandsbehaftete serverbasierte Workloads, die in einem On-Premises-Rechenzentrum bereitgestellt werden, können Sie AWS Elastic Disaster Recovery verwenden, um Ihre Workloads in AWS zu schützen. Wenn Sie bereits in AWS gehostet sind, können Sie Elastic Disaster Recovery verwenden, um Ihren Workload in einer anderen Availability Zone oder Region zu schützen. Elastic Disaster Recovery verwendet eine kontinuierliche Replikation auf Block-Ebene in eine schlanke Staging-Area, um eine schnelle, zuverlässige Wiederherstellung von On-Premises-Anwendungen und cloudbasierten Anwendungen zu gewährleisten.

Implementierungsschritte

1. Implementieren Sie die Selbstreparatur. Stellen Sie Ihre Instances oder Container nach Möglichkeit mit automatischer Skalierung bereit. Wenn dies nicht möglich ist, nutzen Sie für EC2-Instances die automatische Wiederherstellung oder implementieren Sie eine automatische Selbstreparatur basierend auf Amazon EC2- oder ECS-Container-Lebenszyklusereignissen.
   - Verwenden Sie Amazon EC2 Auto Scaling-Gruppen für Instances und Container-Workloads, die keine Anforderungen an eine einzelne Instance-IP-Adresse, private IP-Adresse, elastische IP-Adresse und Instance-Metadaten stellen.
     - Die Benutzerdaten der Startvorlage können zur Implementierung einer Automatisierung verwendet werden, die die meisten Workloads automatisch reparieren kann.
   - Verwenden Sie die automatische Wiederherstellung von Amazon EC2-Instances für Workloads, die eine einzige Instance-IP-Adresse, eine private IP-Adresse, eine elastische IP-Adresse und Instance-Metadaten erfordern.
     - Automatic Recovery sendet Benachrichtigungen zum Wiederherstellungsstatus an ein SNS-Thema, wenn der Instance-Fehler erkannt wird.

- Verwenden Sie Lebenszyklusereignisse von Amazon EC2-Instances oder Amazon ECS-Ereignissen, um das Self-Healing zu automatisieren, wenn eine automatische Skalierung oder EC2-Wiederherstellung nicht verwendet werden kann.
  - Verwenden Sie die Ereignisse, um die Automatisierung der Reparatur der Komponente entsprechend der erforderlichen Prozesslogik aufzurufen.
- Schützen Sie zustandsbasierte Workloads, die auf einen einzigen Standort beschränkt sind, mit AWS Elastic Disaster Recovery.

Ressourcen

Zugehörige Dokumente:

- Amazon ECS-Ereignisse
- Amazon EC2 Auto Scaling-Lebenszyklus-Hooks
- Stellen Sie Ihre Instance wieder her.
- Automatische Skalierung von Services
- Was ist Amazon EC2 Auto Scaling?
- AWS Elastic Disaster Recovery

REL10-BP04 Verwenden von Bulkhead-Architekturen, um den Umfang von Beeinträchtigungen zu begrenzen

Implementieren Sie Bulkhead-Architekturen (zellenbasierte Architekturen), um die Auswirkungen von Fehlern innerhalb eines Workloads auf eine begrenzte Anzahl von Komponenten zu beschränken.

Gewünschtes Ergebnis: Eine zellenbasierte Architektur verwendet mehrere isolierte Instances eines Workloads, wobei jede Instance als Zelle bezeichnet wird. Jede Zelle ist unabhängig. Sie teilt ihren Status nicht mit anderen Zellen und bearbeitet eine Teilmenge der gesamten Workload-Anfragen. Dadurch werden die möglichen Auswirkungen eines Fehlers, z. B. eines fehlerhaften Software-Updates, auf eine einzelne Zelle und die von ihr verarbeiteten Anfragen reduziert. Wenn in einem Workload 10 Zellen für die Beantwortung von 100 Anfragen verwendet werden, sind bei einem Fehler 90 % der gesamten Anfragen nicht davon betroffen.

Typische Anti-Muster:

- Es wird ein unbegrenztes Wachstum der Zellen zugelassen.

- Code-Updates oder Bereitstellungen werden auf alle Zellen gleichzeitig angewandt.
- Status oder Komponenten werden von den Zellen geteilt (mit Ausnahme der Router-Schicht).
- Es werden komplexe Geschäfts- oder Routing-Logiken in die Routing-Schicht eingefügt.
- Es gibt keine Minimierung der zellenübergreifenden Interaktionen.

Vorteile der Nutzung dieser bewährten Methode: Bei zellenbasierten Architekturen treten viele häufige Fehlerarten innerhalb einer Zelle selbst auf, was eine zusätzliche Fehlerisolierung ermöglicht. Diese Fehlergrenzen bieten Schutz vor Fehlern, die sich sonst nur schwer eindämmen lassen, wie z. B. eine erfolglose Codebereitstellung oder Anfragen, die beschädigt sind oder einen bestimmten Fehlermodus auslösen (Poison Pill Requests).

Implementierungsleitfaden

Auf einem Schiff sorgen Schotten dafür, dass eine Beschädigung des Rumpfes auf einen Teil des Schiffes beschränkt bleibt. In komplexen Systemen wird dieses Muster oft kopiert, um eine Fehlerisolierung zu ermöglichen. Fehlerisolierte Grenzen beschränken die Auswirkungen eines Fehlers innerhalb eines Workloads auf eine begrenzte Anzahl von Komponenten. Komponenten außerhalb der Grenze sind vom Ausfall nicht betroffen. Wenn Sie mehrere fehlerisolierte Grenzen verwenden, können Sie die Auswirkungen auf Ihren Workload einschränken. Bei AWS können Kunden mehrere Availability Zones und Regionen verwenden, um eine Fehlerisolierung zu gewährleisten. Das Konzept der Fehlerisolierung lässt sich jedoch auch auf die Architektur Ihres Workloads ausweiten.

Der gesamte Workload wird durch einen Partitionsschlüssel in Zellen unterteilt. Dieser Schlüssel muss mit dem Grain des Service übereinstimmen, d. h. mit der logischen Art und Weise, in der der Workload eines Service mit minimalen zellenübergreifenden Interaktionen unterteilt werden kann. Beispiele für Partitionsschlüssel sind die ID des Kunden, die ID der Ressource oder jeder andere Parameter, der in den meisten API-Aufrufen leicht zugänglich ist. Eine Schicht für das Routing von Zellen verteilt Anfragen auf der Grundlage des Partitionsschlüssels an einzelne Zellen und präsentiert den Kunden einen einzigen Endpunkt.

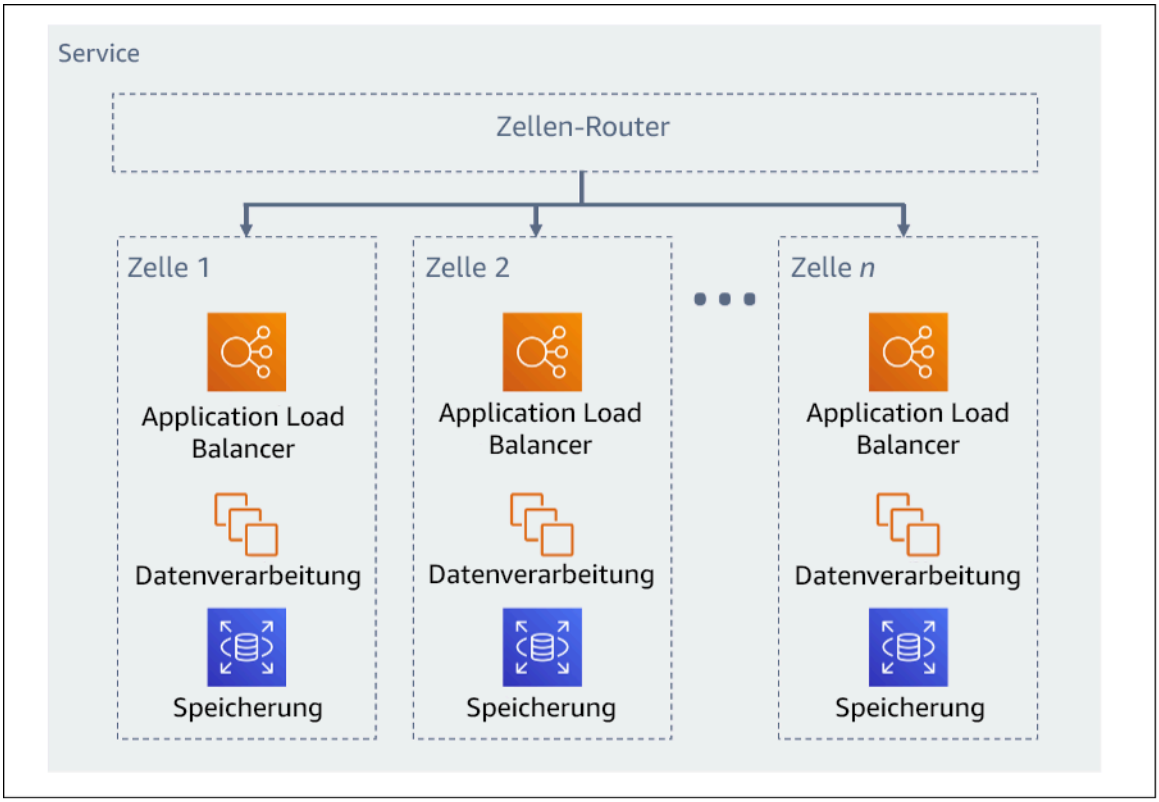


Abbildung 11: Zellenbasierte Architektur

Implementierungsschritte

Bei der Entwicklung einer zellenbasierten Architektur sind mehrere Designüberlegungen zu berücksichtigen:

1. Partitionsschlüssel: Bei der Wahl des Schlüssels für die Partitionierung sollten Sie besonders sorgfältig vorgehen.
   - Er sollte mit der Struktur des Service übereinstimmen oder mit der natürlichen Art und Weise, wie der Workload eines Service mit minimalen zellenübergreifenden Interaktionen unterteilt werden kann. Beispiele sind `Kunden-ID` oder `Ressourcen-ID`.
   - Der Partitionsschlüssel muss in allen Anfragen verfügbar sein – entweder direkt oder in einer Weise, die sich durch andere Parameter leicht deterministisch ableiten lässt.
2. Persistente Zellenzuordnung: Upstream-Services sollten während des Lebenszyklus ihrer Ressourcen nur mit einer einzigen Zelle interagieren.
   - Je nach Workload kann eine Strategie zur Migration von Zellen erforderlich sein, um Daten von einer Zelle in eine andere zu migrieren. Ein mögliches Szenario, in dem eine Migration von

Zellen erforderlich sein kann, ist, wenn ein bestimmter Benutzer oder eine bestimmte Ressource in Ihrem Workload zu groß wird und eine eigene Zelle benötigt.

- Zellen sollten keinen Status und keine Komponenten gemeinsam nutzen.
- Folglich sollten zellenübergreifende Interaktionen vermieden oder auf ein Minimum beschränkt werden, da diese Interaktionen Abhängigkeiten zwischen den Zellen schaffen und somit die Möglichkeiten zur Fehlerisolierung verringern.

3. Routing-Schicht: Die Routing-Schicht ist eine gemeinsame Komponente von Zellen und kann daher nicht dieselbe Strategie der Segmentierung wie bei Zellen nutzen.
   - Es wird empfohlen, dass die Routing-Schicht Anfragen auf einzelne Zellen verteilt, indem sie einen effizienten Algorithmus für die Zuordnung von Partitionen einsetzt – z. B. als die Kombination von kryptographischen Hash-Funktionen und einer modularen Arithmetik.
   - Um Auswirkungen auf mehrere Zellen zu vermeiden, muss die Routing-Schicht so einfach und horizontal skalierbar wie möglich bleiben, was den Verzicht auf eine komplexe Geschäftslogik innerhalb dieser Schicht erforderlich macht. Dies hat den zusätzlichen Nutzen, dass das erwartete Verhalten jederzeit leicht nachvollziehbar ist, was eine gründliche Testbarkeit ermöglicht. Wie Colm MacCárthaigh in Reliability, constant work, and a good cup of coffee (Zuverlässigkeit, konstante Arbeit und eine gute Tasse Kaffee) erläutert, führen einfache Designs und konstante Arbeitsmuster zu zuverlässigen Systemen und verringern die Antifragilität.
4. Zellengröße: Zellen sollten eine maximale Größe haben und nicht darüber hinaus wachsen dürfen.
   - Die maximale Größe sollte durch gründliche Tests ermittelt werden – bis Sollbruchstellen erreicht und sichere operative Margen etabliert sind. Weitere Details zur Implementierung von Testverfahren finden Sie unter REL07-BP04 Durchführen von Lasttests für die Workload
   - Der gesamte Workload sollte durch Hinzufügen zusätzlicher Zellen wachsen, sodass der Workload mit der steigenden Nachfrage skalieren kann.
5. Multi-AZ oder Multi-Region-Strategien: Es sollten mehrere Schichten zur Ausfallsicherheit genutzt werden, um sich gegen verschiedene Fehlerbereiche zu schützen.
   - Für die Ausfallsicherheit sollten Sie einen Ansatz wählen, bei dem verschiedene Verteidigungsebenen aufgebaut werden. Eine Ebene schützt vor kleineren, häufiger auftretenden Unterbrechungen, indem eine hochverfügbare Architektur mit mehreren AZs erstellt wird. Eine weitere Verteidigungsebene schützt vor seltenen Ereignissen wie Naturkatastrophen mit großer Reichweite und Unterbrechungen auf Regionsebene. Für diese zweite Ebene muss die Architektur Ihrer Anwendung mehrere AWS-Regionen umfassen. Wenn Sie eine Multi-Region-Strategie für Ihren Workload implementieren, ist er vor weitreichenden

Naturkatastrophen, die einen großen geografischen Bereich in einem Land betreffen, oder technischen Fehlern in einer ganzen Region geschützt. Beachten Sie dabei, dass das Implementieren einer Multi-Region-Architektur äußerst komplex sein kann und bei den meisten Workloads nicht erforderlich ist. Weitere Details finden Sie unter REL10-BP02 Auswählen der geeigneten Standorte für Ihre Multi-Standort-Bereitstellung.

6. Code-Bereitstellung: Eine gestaffelte Strategie für die Bereitstellung von Code sollte der gleichzeitigen Bereitstellung von Codeänderungen in allen Zellen vorgezogen werden.
   - Auf diese Weise werden mögliche Fehler in mehreren Zellen aufgrund einer fehlerhaften Bereitstellung oder menschlichen Versagens minimiert. Weitere Details finden Sie unter Automatisierung sicherer, vollautomatischer Bereitstellungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Ressourcen

Zugehörige bewährte Methoden:

- REL07-BP04 Durchführen von Lasttests für die Workload
- REL10-BP02 Auswählen der geeigneten Standorte für Ihre Multi-Standort-Bereitstellung

Zugehörige Dokumente:

- Reliability, constant work, and a good cup of coffee (Zuverlässigkeit, konstante Arbeit und ein ordentlicher Kaffee)
- AWS and Compartmentalization (Segmentierung mit AWS)
- Workload-Isolation mit Shuffle Sharding
- Automatisierung sicherer, vollautomatischer Bereitstellungen

Zugehörige Videos:

- AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small (AWS re:Invent 2018: Details und Strategien: Wie man die Kontrolle über große und kleine Systeme übernimmt)
- AWS re:Invent 2018: So minimiert AWS den Wirkungsradius von Fehlern (ARC338)
- Shuffle Sharding: AWS re:Invent 2019: Einführung in die Amazon Builders' Library (DOP328)

- AWS Summit ANZ 2021 – Everything fails, all the time: Designing for resilience (AWS Summit ANZ 2021 – Alles schlägt fehl, immer wieder: Design für Ausfallsicherheit)

Zugehörige Beispiele:

- Well-Architected Lab: Fehlerisolierung mit Shuffle Sharding

REL 11. Wie können Sie Workloads so gestalten, dass sie Komponentenausfällen gegenüber resilient sind?

Workloads, die eine hohe Verfügbarkeit und eine niedrige mittlere Wiederherstellungszeit (Mean Time To Recovery, MTTR) benötigen, müssen auf Resilienz ausgelegt sein.

Bewährte Methoden

- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP02 Failover zu fehlerfreien Ressourcen
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung
- REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität
- REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken
- REL11-BP07 Architektur Ihres Produkts zur Erfüllung von Verfügbarkeitszielen und Uptime-SLAs (Service Level Agreements)

REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Überwachen Sie den Zustand Ihres Workloads kontinuierlich, damit Sie und Ihre automatisierten Systeme auf Fehler oder Verschlechterungen aufmerksam werden, sobald diese auftreten. Überwachen Sie Key Performance Indicators (KPIs, wichtige Leistungskennzahlen) auf Grundlage des geschäftlichen Wertes.

Alle Wiederherstellungs- und Reparaturmechanismen müssen auf eine schnelle Erkennung von Problemen ausgelegt sein. Technische Fehler sollten zuerst erkannt werden, damit sie behoben werden können. Die Verfügbarkeit basiert jedoch auf der Fähigkeit Ihrer Workload, einen Unternehmenswert zu liefern. Daher müssen wichtige Leistungskennzahlen (KPIs), die dies messen, in Ihre Erkennungs- und Behebungsstrategie integriert sein.

Gewünschtes Ergebnis: Wesentliche Komponenten eines Workloads werden unabhängig überwacht, um Fehler zu erkennen und anzuzeigen, wann und wo sie auftreten.

Typische Anti-Muster:

- Es sind keine Alarme konfiguriert, sodass Ausfälle ohne Benachrichtigung auftreten.
- Alarme sind vorhanden, aber mit Schwellenwerten, die keine ausreichende Zeit für die Reaktion bieten.
- Metriken werden nicht häufig genug erfasst, um das Recovery Time Objective (RTO) zu erreichen.
- Nur die kundenorientierten Schnittstellen des Workloads werden aktiv überwacht.
- Es werden nur technische Metriken erfasst, keine Metriken für Geschäftsfunktionen.
- Es gibt keine Metriken, die die Benutzererfahrung der Workload messen.
- Es werden zu viele Überwachungen erstellt.

Vorteile der Nutzung dieser bewährten Methode: Mit einer angemessenen Überwachung auf allen Ebenen können Sie die Wiederherstellungszeit reduzieren, indem Sie die Zeit bis zur Erkennung verkürzen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Identifizieren Sie alle Workloads, die für die Überwachung überprüft werden sollen. Sobald Sie alle zu überwachenden Komponenten des Workloads identifiziert haben, müssen Sie das Überwachungsintervall festlegen. Das Überwachungsintervall wirkt sich direkt darauf aus, wie schnell eine Wiederherstellung eingeleitet werden kann (abhängig davon, wie lange die Erkennung eines Fehlers dauert). Die Mittlere Zeit bis zur Erkennung ist die Zeitspanne zwischen dem Auftreten eines Fehlers und dem Beginn der Reparaturarbeiten. Die Liste der Services sollte umfassend und vollständig sein.

Die Überwachung muss alle Ebenen des Anwendungs-Stacks (inklusive Anwendung, Plattform, Infrastruktur und Netzwerk) abdecken.

Ihre Überwachungsstrategie sollte außerdem die Auswirkungen von grauen Fehlernberücksichtigen. Weitere Details zu grauen Fehlern finden Sie unter Graue Fehler im Whitepaper „Advanced Multi-AZ Resilience Patterns“ (Erweiterte Multi-AZ Resilience-Muster).

Implementierungsschritte

- Überwachungsintervall hängt davon ab, wie schnell Wiederherstellungen durchgeführt werden müssen. Die Wiederherstellungszeit hängt davon ab, wie viel Zeit für eine Wiederherstellung benötigt wird. Daher müssen Sie die Häufigkeit der Erfassung bestimmen, indem Sie diese Zeit und das RTO einkalkulieren.
- Konfigurieren Sie eine detaillierte Überwachung für Komponenten und verwaltete Services.
  - Bestimmen Sie, ob eine detaillierte Überwachung für EC2-Instances und Auto Scaling notwendig ist. Eine detaillierte Überwachung liefert Metriken in einminütigen Intervallen, die Standardüberwachung liefert Metriken in fünfminütigen Intervallen.
  - Bestimmen Sie, ob eine erweiterte Überwachung für RDS erforderlich ist. Die erweiterte Überwachung verwendet einen Agenten auf RDS-Instances, um nützliche Informationen über verschiedene Prozesse oder Threads zu erhalten.
  - Bestimmen Sie die Anforderungen an die Überwachung von kritischen Serverless-Komponenten für Lambda, API Gateway, Amazon EKS, Amazon ECS, und alle Arten von Load Balancernberücksichtigen.
  - Ermitteln Sie die Überwachungsanforderungen von Speicherkomponenten für Amazon S3, Amazon FSx, Amazon EFSund Amazon EBS.
- Erstellen Sie benutzerdefinierte Metriken, um geschäftliche Key Performance Indicators (KPIs) zu messen. Workloads implementieren wichtige geschäftliche Funktionen, die als KPIs verwendet werden sollten, um zu erkennen, wann ein indirektes Problem auftritt.
- Überwachen Sie das Benutzererlebnis auf Fehler mithilfe von Benutzer-Canarys. Tests für synthetische Transaktionen (auch bekannt als Canary-Tests, aber nicht zu verwechseln mit Canary-Bereitstellungen), die das Kundenverhalten simulieren können, gehören zu den wichtigsten Testprozessen. Führen Sie diese Tests für Ihre Workload-Endpunkte konstant von verschiedenen Remote-Standorten aus.
- Erstellen Sie benutzerdefinierte Metriken, die das Benutzererlebnis nachverfolgen. Wenn Sie das Kundenerlebnis instrumentieren können, können Sie die Verschlechterung des Kundenerlebnisses feststellen.
- Legen Sie Alarme fest, um zu erkennen, wenn ein Teil Ihres Workloads nicht ordnungsgemäß funktioniert, und um anzuzeigen, wann die Ressourcen automatisch skaliert werden müssen. Alarme können visuell auf Dashboards angezeigt werden, Warnungen über Amazon SNS oder E-Mail versenden und mit Auto Scaling zusammenarbeiten, um Workload-Ressourcen hoch- oder herunterskalieren zu können.

- Erstellen Sie Dashboards, um Ihre Metriken zu visualisieren. Dashboards können verwendet werden, um Trends, Ausreißer und andere Indikatoren für potenzielle Probleme zu visualisieren, und auf Probleme hinweisen, die Sie untersuchen sollten.
- Erstellen Sie eine verteilte Tracing-Überwachung für Ihre Services. Mit der verteilten Überwachung können Sie nachvollziehen, wie Ihre Anwendung und die ihr zugrunde liegenden Services arbeiten, um die Ursache von Leistungsproblemen und Fehlern zu identifizieren und zu beheben.
- Erstellen Sie Überwachungssysteme (mit CloudWatch oder X-Ray) Dashboards und einer Datenerfassung in einer eigenen Region und einem eigenen Konto.
- Erstellen Sie eine Integration zur Amazon Health Aware Überwachung, um die Überwachung von AWS-Ressourcen zu ermöglichen, bei denen es zu Leistungseinbußen kommen könnte. Für geschäftskritische Workloads bietet diese Lösung Zugriff auf proaktive und Echtzeitbenachrichtigungen für AWS-Services.

Ressourcen

Zugehörige bewährte Methoden:

- Definition der Verfügbarkeit
- REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken

Zugehörige Dokumente:

- Amazon CloudWatch Synthetics unterstützt Sie bei der Erstellung von Benutzer-Canaries.
- Aktivieren oder deaktivieren Sie die detaillierte Überwachung für Ihre Instance
- Erweiterte Überwachung
- Überwachen ihrer Auto Scaling-Gruppe und Instances mit Amazon CloudWatch
- Veröffentlichen benutzerdefinierter Metriken
- Verwenden von Amazon CloudWatch-Alarmen
- Verwenden von CloudWatch-Dashboards
- Using Cross Region Cross Account CloudWatch Dashboards (Verwenden von konto- und regionenübergreifenden Amazon CloudWatch-Dashboards)
- Using Cross Region Cross Account X-Ray Tracing (Verwenden der konto- und regionenübergreifenden Amazon CloudWatch-Nachverfolgung)

- Verstehen der Verfügbarkeit
- Implementing Amazon Health Aware (AHA) (Implementierung von Amazon Health Aware (AHA))

Zugehörige Videos:

- Mitigating gray failures (Beheben von grauen Fehlern)

Zugehörige Beispiele:

- Well-Architected Lab: Level 300: Implementieren von Zustandsprüfungen und Verwalten von Abhängigkeiten zur Verbesserung der Zuverlässigkeit
- Workshop zur Beobachtbarkeit: X-Ray erkunden

Zugehörige Tools:

- CloudWatch
- CloudWatch X-Ray

REL11-BP02 Failover zu fehlerfreien Ressourcen

Wenn ein Fehler bei einer Ressource auftritt, sollten intakte Ressourcen weiterhin Anfragen bedienen. Stellen Sie sicher, dass Sie bei Standortbeeinträchtigungen (z. B. Availability Zone oder AWS-Region) über Systeme verfügen, die einen Failover auf intakte Ressourcen an nicht beeinträchtigten Standorten ermöglichen.

Wenn Sie einen Service entwerfen, verteilen Sie die Last auf Ressourcen, Availability Zones oder Regionen. So kann der Fehler einer einzelnen Ressource oder eine Beeinträchtigung durch die Verlagerung des Datenverkehrs auf die verbleibenden intakten Ressourcen aufgefangen werden. Überlegen Sie, wie Services im Falle eines Fehlers erkannt und geroutet werden.

Entwerfen Sie Ihre Services mit Blick auf die Fehlerbehebung. Bei AWS konzipieren wir Services mit dem Ziel, die Wiederherstellungszeit nach Ausfällen und die Auswirkungen auf Daten zu minimieren. Unsere Services verwenden primär Datenspeicher, die Anfragen erst akzeptieren, nachdem sie dauerhaft auf mehreren Replikaten in einer Region gespeichert wurden. Sie sind so aufgebaut, dass sie eine zellenbasierte Isolation und die Fehlerisolierung von Availability Zones nutzen. In unseren betrieblichen Abläufen setzen wir sehr stark auf Automatisierung. Außerdem optimieren wir unsere

Funktionalität für Ersetzungsvorgänge und Neustarts, um nach Unterbrechungen eine schnelle Wiederherstellung zu ermöglichen.

Die Muster und Entwürfe, die den Failover ermöglichen, variieren für jeden AWS-Plattform-Service. Viele native verwaltete Services von AWS nutzen von Haus aus mehrere Availability Zones (wie Lambda oder API Gateway). Andere AWS-Services (wie EC2 und EKS) erfordern spezielle bewährte Methoden, um einen Failover von Ressourcen oder Datenspeichern über AZs hinweg zu unterstützen.

Es sollte eine Überwachung eingerichtet werden, um zu überprüfen, ob die Failover-Ressource in Ordnung ist, den Fortschritt der Failover-Ressourcen zu verfolgen und die Wiederherstellung von Geschäftsprozessen zu überwachen.

Gewünschtes Ergebnis: Die Systeme sind in der Lage, automatisch oder manuell neue Ressourcen zu verwenden, um sich von Störungen zu erholen.

Typische Anti-Muster:

- Die Planung für Fehler ist nicht Teil der Planungs- und Designphase.
- RTO und RPO sind nicht festgelegt.
- Unzureichende Überwachung, um ausfallende Ressourcen zu erkennen.
- Ordnungsgemäße Isolierung von fehlerhaften Domänen.
- Multi-Region-Failover wird nicht berücksichtigt.
- Die Erkennung von Fehlern ist bei der Entscheidung für einen Failover zu empfindlich oder zu aggressiv.
- Failover-Design wird nicht getestet oder validiert.
- Durchführen automatischer Reparaturen ohne die Benachrichtigung, dass eine Reparatur erforderlich war.
- Fehlender Ausgleichszeitraum, um einen zu frühen Failover zu vermeiden.

Vorteile der Nutzung dieser bewährten Methode: Sie können widerstandsfähigere Systeme aufbauen, die auch bei Fehlern zuverlässig bleiben, indem sie ordnungsgemäß reduziert werden und sich schnell erholen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

AWS-Services, wie z. B. Elastic Load Balancing und Amazon EC2 Auto Scalinghelfen, die Last auf Ressourcen und Availability Zones zu verteilen. Daher können der Ausfall einer einzelnen Ressource (wie etwa einer EC2-Instance) oder die Beeinträchtigung einer Availability Zone gemindert werden, indem Datenverkehr verlagert wird, um Ressourcen fehlerfrei zu halten.

Bei Workloads, die mehrere Regionen umfassen, sind Designs etwas komplizierter. Mit regionenübergreifenden Lesereplikaten können Sie beispielsweise Ihre Daten für mehrere AWS-Regionen bereitstellen. Der Failover ist jedoch immer noch erforderlich, um das Lesereplikat zum primären Endpunkt zu machen und den Datenverkehr auf den neuen Endpunkt zu lenken. Amazon Route 53, Route 53 ARC, CloudFront und AWS Global Accelerator können beim Routing des Datenverkehrs über AWS-Regionen helfen.

AWS-Services wie Amazon S3, Lambda, API Gateway, Amazon SQS, Amazon SNS, Amazon SES, Amazon Pinpoint, Amazon ECR, AWS Certificate Manager, EventBridge oder Amazon DynamoDB werden von AWS automatisch in mehreren Availability Zones bereitgestellt. Im Falle eines Fehlers leiten diese AWS-Services den Datenverkehr automatisch an intakte Standorte um. Die Daten werden redundant in mehreren Availability Zones gespeichert und bleiben verfügbar.

Für Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon EKS oder Amazon ECS ist Multi-AZ eine Konfigurationsoption. AWS kann den Datenverkehr zur intakten Instance umleiten, wenn ein Failover eingeleitet wird. Diese Failover-Aktion kann von AWS oder auf Wunsch des Kunden durchgeführt werden.

Für Amazon EC2-Instances, Amazon Redshift, Amazon ECS-Aufgaben oder Amazon EKS-Pods wählen Sie aus, in welchen Availability Zones sie bereitgestellt werden sollen. Für einige Designs bietet Elastic Load Balancing die Lösung, Instances in fehlerhaften Zonen zu erkennen und den Datenverkehr in die intakten Zonen zu routen. Elastic Load Balancing kann den Datenverkehr auch zu Komponenten in Ihrem On-Premises-Rechenzentrum routen.

Für den Failover von Datenverkehr aus mehreren Regionen kann das Rerouting mit Amazon Route 53, ARC, AWS Global Accelerator, Route 53 Private DNS for VPCs oder CloudFront eine Möglichkeit bieten. Sie können Internetdomänen definieren und Routing-Richtlinien einschließlich Zustandsprüfungen zuweisen, um den Datenverkehr in intakte Regionen zu leiten. AWS Global Accelerator stellt statische IP-Adressen bereit, die als fester Einstiegspunkt für Ihre Anwendung fungieren und dann zu Endpunkten Ihrer Wahl in AWS-Regionen geroutet werden, wobei das globale Netzwerk von AWS anstelle des Internets für eine bessere Leistung und Zuverlässigkeit genutzt wird.

Implementierungsschritte

- Erstellen Sie Failover-Designs für alle entsprechenden Anwendungen und Services. Isolieren Sie jede Komponente der Architektur und erstellen Sie Failover-Designs, die das RTO und RPO für jede Komponente erfüllen.
- Konfigurieren Sie weniger anspruchsvolle Umgebungen (wie Entwicklungs- oder Testumgebungen) mit allen Services, die für einen Failover-Plan erforderlich sind. Stellen Sie die Lösungen mit Infrastructure as Code (IaC) bereit, um die Reproduzierbarkeit sicherzustellen.
- Konfigurieren Sie einen Wiederherstellungsstandort, z. B. eine zweite Region, um die Failover-Designs zu implementieren und zu testen. Falls erforderlich, können die Ressourcen für die Tests vorübergehend konfiguriert werden, um die zusätzlichen Kosten zu begrenzen.
- Bestimmen Sie, welche Failover-Pläne durch AWS automatisiert sind, welche durch einen DevOps-Prozess automatisiert werden können und welche möglicherweise manuell sind. Dokumentieren und messen Sie die RTO- und RPO-Zeiten der einzelnen Services.
- Erstellen Sie ein Failover-Playbook, das alle Schritte zum Failover jeder Ressource, Anwendung und jedes Services enthält.
- Erstellen Sie ein Failback-Playbook, das alle Schritte zum Failback (mit Zeitangabe) für jede Ressource, jede Anwendung und jeden Service enthält.
- Erstellen Sie einen Plan, um das Playbook zu initiieren und zu proben. Verwenden Sie Simulationen und Chaostests, um die Schritte des Playbooks und die Automatisierung zu testen.
- Stellen Sie sicher, dass Sie bei einer Beeinträchtigung des Standorts (z. B. Availability Zone oder AWS-Region) über Systeme verfügen, die einen Failover auf intakte Ressourcen an nicht beeinträchtigten Standorten ermöglichen. Überprüfen Sie Kontingente, die automatische Skalierung und laufende Ressourcen vor dem Failover-Test.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- REL13- Planen für DR
- REL10 – Nutzen der Fehlerisolierung zum Schutz Ihres Workloads

Zugehörige Dokumente:

- Einstellen von RTO- und RPO-Zielen

- Einrichten von ARC mit Application Load Balancers
- Failover mit gewichtetem Route 53-Routing
- DR mit ARC
- EC2 mit automatischer Skalierung
- EC2-Bereitstellungen – Multi-AZ
- ECS-Bereitstellungen – Multi-AZ
- Datenverkehr umleiten mit ARC
- Lambda mit einem Application Load Balancer und Failover
- ACM-Replikation und -Failover
- Parameter Store-Replikation und -Failover
- Regionsübergreifende ECR-Replikation und Failover
- Konfigurieren der regionsübergreifenden Replikation von Secrets Manager
- Aktivieren der regionsübergreifende Replikation für EFS und Failover
- Regionsübergreifende EFS-Replikation und Failover
- Netzwerk-Failover
- S3-Endpunkt-Failover mit MRAP
- Erstellen einer regionsübergreifenden Replikation für S3
- Failover-Region API Gateway mit ARC
- Failover mit Global Accelerator über mehrere Regionen
- Failover mit DRS
- Erstellen von Mechanismen für die Notfallwiederherstellung mit Amazon Route 53

Zugehörige Beispiele:

- Notfallwiederherstellung auf AWS
- Elastische Notfallwiederherstellung auf AWS

REL11-BP03 Automatisieren der Reparatur auf allen Ebenen

Verwenden Sie bei Erkennung eines Fehlers automatisierte Funktionen, um Maßnahmen zur Behebung durchzuführen. Beeinträchtigungen können automatisch durch interne Service-

Mechanismen behoben werden. Es kann aber auch erforderlich sein, Ressourcen neu zu starten oder Abhilfemaßnahmen durchzuführen.

Für selbstverwaltete Anwendungen und regionenübergreifende Korrekturen können Wiederherstellungskonzepte und automatisierte Korrekturprozesse aus bestehenden bewährten Methoden verwendet werden.

Die Möglichkeit, eine Ressource neu zu starten oder zu entfernen, ist ein wichtiges Instrument zur Behebung von Fehlern. Eine bewährte Methode besteht darin, Services nach Möglichkeit zustandslos zu betreiben. Dies verhindert den Datenverlust oder den Verlust der Verfügbarkeit bei einem Neustart der Ressource. In der Cloud können Sie (und sollten Sie üblicherweise) die gesamte Ressource (z. B. eine Computing-Instance oder eine Serverless-Funktion) im Rahmen des Neustarts ersetzen. Der Neustart selbst ist eine einfache und zuverlässige Methode zur Wiederherstellung nach einem Ausfall. Bei Workloads treten viele verschiedene Arten von Fehlern auf. Fehler können bei Hardware, Software, Kommunikation und Betrieb auftreten.

Der Neustart oder Wiederholungsversuch gilt auch für Netzwerkanfragen. Nutzen Sie denselben Wiederherstellungsansatz für eine Netzwerk-Zeitüberschreitung und einen Abhängigkeitsfehler, bei dem die Abhängigkeit einen Fehler ausgibt. Beide Ereignisse wirken sich in ähnlicher Weise auf das System aus. Statt also zu versuchen, aus den einzelnen Ereignissen einen „Sonderfall“ zu konstruieren, sollten Sie eine ähnliche Strategie anwenden und versuchen, einen exponentiellen Backoff mit Jitter durchzuführen. Die Fähigkeit zum Neustart ist eine Funktion, die in wiederherstellungsorientierten Computing- und Hochverfügbarkeits-Cluster-Architekturen empfohlen wird.

Gewünschtes Ergebnis: Automatisierte Aktionen werden durchgeführt, um die Erkennung eines Fehlers zu beheben.

Typische Anti-Muster:

- Bereitstellung von Ressourcen ohne automatische Skalierung.
- Einzelne Bereitstellung von Anwendungen in Instances oder Containern.
- Bereitstellen von Anwendungen, die nicht ohne automatische Wiederherstellung an mehreren Standorten bereitgestellt werden können.
- Manuelle Reparatur von Anwendungen, die sich mit Auto Scaling und einer automatischen Wiederherstellung nicht reparieren lassen.
- Keine Automatisierung beim Failover von Datenbanken.

- Keine automatisierten Methoden zur Umleitung des Datenverkehrs auf neue Endpunkte.
- Keine Speicherreplikation.

Vorteile der Nutzung dieser bewährten Methode: Eine automatisierte Korrektur kann die mittlere Zeit bis zur Wiederherstellung verkürzen und Ihre Verfügbarkeit verbessern.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Designs für Amazon EKS oder andere Kubernetes-Services sollten sowohl minimale und maximale Replikate oder zustandsbehaftete Sets als auch die minimale Größenanpassung von Clustern und Knotengruppen umfassen. Diese Mechanismen sorgen für ein Minimum an kontinuierlich verfügbaren Verarbeitungsressourcen und beheben gleichzeitig automatisch alle Fehler über die Steuerebene von Kubernetes.

Entwurfsmuster, auf die über einen Load Balancer mit Computing-Clustern zugegriffen wird, sollten Auto Scaling-Gruppen nutzen. Elastic Load Balancing (ELB) verteilt den eingehenden Datenverkehr von Anwendungen automatisch auf mehrere Ziele und virtuelle Appliances in einer oder mehreren Availability Zones (AZs).

Bei Cluster-Compute-Instances, die kein Load Balancing nutzen, sollte die Größe für den Verlust von mindestens einem Knoten ausgelegt sein. Auf diese Weise kann der Service mit möglicherweise reduzierter Kapazität weiterlaufen, während er einen neuen Knoten wiederherstellt. Beispiele für Services sind Mongo, DynamoDB Accelerator, Amazon Redshift, Amazon EMR, Cassandra, Kafka, MSK-EC2, Couchbase, ELK und Amazon OpenSearch Service. Viele dieser Services können mit zusätzlichen Funktionen zur Selbstheilung ausgestattet werden. Einige Cluster-Technologien müssen beim Verlust eines Knotens einen Alarm generieren, der einen automatisierten oder manuellen Workflow zur Wiederherstellung eines neuen Knotens auslöst. Dieser Workflow kann mit AWS Systems Manager automatisiert werden, um Probleme schnell zu beheben.

Mit Amazon EventBridge lassen sich Ereignisse wie CloudWatch-Alarme oder Statusänderungen in anderen AWS-Services überwachen und filtern. Auf der Grundlage von Ereignisinformationen kann er dann AWS Lambda, Systems Manager Automation oder andere Ziele aufrufen, um eine angepasste Abhilfelogik für Ihren Workload auszuführen. Amazon EC2 Auto Scaling kann so konfiguriert werden, dass der Status der EC2-Instance überprüft wird. Wenn sich die Instance nicht im ausgeführten Status befindet oder der Systemstatus beeinträchtigt ist, betrachtet Amazon EC2 Auto Scaling die Instance als fehlerhaft und startet eine Ersatz-Instance. Bei Large-Scale-Ersetzungen (z. B.

dem Verlust einer ganzen Availability Zone) ist für eine Hochverfügbarkeit die statische Stabilität vorzuziehen.

Implementierungsschritte

- Verwenden Sie Auto Scaling-Gruppen, um Tiers in einem Workload bereitzustellen. Auto Scaling kann zustandslose Anwendungen selbst reparieren und Kapazitäten hinzufügen oder entfernen.
- Für die bereits erwähnten Computing-Instances verwenden Sie Load Balancing und wählen Sie den entsprechenden von Load-Balancer-Typ aus.
- Erwägen Sie die Reparatur für Amazon RDS. Bei Standby-Instances konfigurieren Sie den automatischen Failover auf die Standby-Instance. Bei Amazon RDS-Lesereplikaten ist ein automatisierter Workflow erforderlich, um ein Lesereplikat zur primären Instance zu machen.
- Implementieren Sie die automatische Wiederherstellung auf EC2-Instances, die Anwendungen bereitstellen, die nicht an mehreren Standorten bereitgestellt werden können und die einen Neustart bei Fehlern tolerieren können. Mithilfe der automatischen Wiederherstellung kann ausgefallene Hardware ersetzt und die Instance neu gestartet werden, wenn die Anwendung sich nicht an mehreren Standorten bereitstellen lässt. Die Metadaten der Instance und die zugehörigen IP-Adressen werden beibehalten, ebenso wie die EBS-Volumes und Einbindungspunkte für Amazon Elastic File System oder Dateisysteme für Lustre und Windows. Mit AWS OpsWorkskönnen Sie die automatische Wiederherstellung von EC2-Instances auf Layer-Ebene konfigurieren.
- Implementieren Sie die automatische Wiederherstellung mit AWS Step Functions und AWS Lambda wenn Sie keine automatische Skalierung oder automatische Wiederherstellung verwenden können oder wenn die automatische Wiederherstellung fehlschlägt. Wenn Sie keine automatische Skalierung verwenden können und die automatische Wiederherstellung entweder nicht genutzt werden kann oder fehlschlägt, können Sie die Reparatur mithilfe von AWS Step Functions und AWS Lambda automatisieren.
- Amazon EventBridge kann verwendet werden, um Ereignisse zu überwachen und zu filtern, wie CloudWatch-Alarme oder Zustandsänderungen in anderen AWS-Services. Auf der Grundlage von Ereignisinformationen kann es dann AWS Lambda (oder andere Ziele) aufrufen, um eine angepasste Wiederherstellungslogik für Ihren Workload auszuführen.

Ressourcen

Zugehörige bewährte Methoden:

- Definition der Verfügbarkeit

- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Zugehörige Dokumente:

- So funktioniert AWS Auto Scaling
- Automatische Wiederherstellung mit Amazon EC2
- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Was ist Amazon FSx for Lustre?
- Was ist Amazon FSx for Windows File Server?
- AWS OpsWorks: Verwenden von Auto Healing zum Austausch fehlgeschlagener Instances
- Was ist AWS Step Functions?
- Was ist AWS Lambda?
- Was ist Amazon EventBridge?
- Verwenden von Amazon CloudWatch-Alarmen
- Amazon RDS-Failover
- SSM – Systems Manager-Automatisierung
- Bewährte Methoden für eine widerstandsfähige Architektur

Zugehörige Videos:

- Automatically Provision and Scale OpenSearch Service (OpenSearch automatisch Bereitstellen und skalieren)
- Automatischer Failover mit Amazon RDS

Zugehörige Beispiele:

- Workshop zu Auto Scaling
- Amazon RDS-Failover Workshop

Zugehörige Tools:

- CloudWatch

- CloudWatch X-Ray

REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung

Steuerebenen stellen die administrativen APIs zum Erstellen, Lesen und Schreiben, Aktualisieren, Löschen und Auflisten (CRUDL) von Ressourcen bereit, während Datenebenen den normalen Datenverkehr des Services abwickeln. Konzentrieren Sie sich bei der Implementierung von Wiederherstellungs- oder Abhilfemaßnahmen für Ereignisse, die sich möglicherweise auf die Ausfallsicherheit auswirken, auf eine minimale Anzahl von Operationen auf der Steuerebene, um den Service wiederherzustellen, zu skalieren, zu reparieren oder einen Failover durchzuführen. Aktionen auf der Datenebene sollten während dieser Beeinträchtigungen Vorrang vor allen anderen Aktivitäten haben.

Die folgenden Aktionen gehören beispielsweise alle zur Steuerebene: Starten einer neuen Computing-Instance, Erstellen von Block-Speicher und Beschreiben von Warteschlangen-Services. Wenn Sie Computing-Instances starten, muss die Steuerebene mehrere Aufgaben erfüllen, z. B. einen physischen Host mit Kapazität finden, Netzwerkschnittstellen zuweisen, lokale Block-Speicher-Volumes vorbereiten, Anmeldeinformationen generieren und Sicherheitsregeln hinzufügen. Steuerebenen neigen zu einer komplizierten Orchestrierung.

Gewünschtes Ergebnis: Wenn bei einer Ressource eine Störung auftritt, ist das System in der Lage, diese automatisch oder manuell zu beheben, indem es den Datenverkehr von gestörten auf intakte Ressourcen umleitet.

Typische Anti-Muster:

- Abhängigkeit von der Änderung von DNS-Einträgen, um den Datenverkehr umzuleiten.
- Abhängigkeit von Skalierungsoperationen auf Steuerebene, um beeinträchtigte Komponenten aufgrund einer unzureichenden Bereitstellung von Ressourcen zu ersetzen.
- Abhängigkeit von umfangreichen Aktionen auf der Steuerebene, in die mehrere Services und APIs involviert sind, um Störungen jeglicher Art zu beheben.

Vorteile der Nutzung dieser bewährten Methode: Eine höhere Erfolgsquote bei der automatisierten Behebung kann Ihre mittlere Zeit bis zur Wiederherstellung verkürzen und die Verfügbarkeit des Workloads verbessern.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel: Bei bestimmten Arten von Service-Störungen sind die Steuerebenen betroffen. Die Abhängigkeit von einer umfassenden Nutzung der

Steuerebene für die Behebung kann die Wiederherstellungszeit (RTO) und die mittlere Zeit bis zur Wiederherstellung (MTTR) erhöhen.

Anleitung zur Umsetzung

Um die Aktionen auf der Datenebene zu begrenzen, bewerten Sie für jeden Service, welche Aktionen zur Wiederherstellung des Services erforderlich sind.

Nutzen Sie Amazon Application Recovery Controller (ARC), um den DNS-Datenverkehr zu verlagern. Diese Funktionen überwachen kontinuierlich die Fähigkeit Ihrer Anwendung, nach Fehlern wiederhergestellt zu werden, und ermöglichen es Ihnen, die Wiederherstellung Ihrer Anwendung über mehrere AWS-Regionen, Availability Zones und On-Premises zu steuern.

Route 53-Routingrichtlinien verwenden die Steuerebene. Verlassen Sie sich also bei der Wiederherstellung nicht auf diese Ebene. Die Route 53-Datenebenen beantworten DNS-Abfragen und führen Zustandsprüfungen durch und werten diese aus. Sie sind global verteilt und für ein Service Level Agreement (SLA) mit einer Verfügbarkeit von 100 % entworfen worden.

Die Route 53-Verwaltungs-APIs und -Konsolen, über die Sie Route 53-Ressourcen erstellen, aktualisieren und löschen, arbeiten auf Steuerebenen, die so konzipiert sind, dass die starke Konsistenz und Stabilität, die Sie beim Verwalten von DNS benötigen, Priorität haben. Zu diesem Zweck befinden sich die Steuerebenen in einer einzelnen Region, USA Ost (Nord-Virginia). Beide Systeme sind zwar äußerst zuverlässig, aber die Steuerebenen sind nicht in der SLA enthalten. In seltenen Fällen kann es vorkommen, dass das ausfallsichere Design der Datenebene es ermöglicht, die Verfügbarkeit aufrechtzuerhalten, während die Steuerebene dies nicht tut. Verwenden Sie für die Notfallwiederherstellung und Failover-Mechanismen Datenebenen-Funktionen, um die bestmögliche Zuverlässigkeit bereitzustellen.

Verwenden Sie für Amazon EC2 Designs mit statischer Stabilität, um Aktionen auf der Steuerebene zu begrenzen. Zu den Aktionen auf der Steuerebene gehört das Hochskalieren von Ressourcen einzeln oder über Auto Scaling-Gruppen (ASG). Für ein Höchstmaß an Ausfallsicherheit stellen Sie ausreichende Kapazitäten in dem für den Failover verwendeten Cluster bereit. Wenn diese Kapazität begrenzt werden muss, legen Sie Drosselungen für das gesamte System fest, um den Gesamtdatenverkehr an die beschränkte Ressourcenmenge sicher zu begrenzen.

Bei Services wie Amazon DynamoDB, Amazon API Gateway, Load Balancern und AWS Lambda Serverless wird die Datenebene für diese Services genutzt. Die Erstellung neuer Funktionen, Load Balancers, API-Gateways oder DynamoDB-Tabellen ist jedoch eine Aktion auf der Steuerebene und sollte vor der Störung als Vorbereitung auf ein Ereignis und zum Üben von Failover-Aktionen

durchgeführt werden. Für Amazon RDS ermöglichen Aktionen auf der Datenebene den Zugriff auf Daten.

Weitere Informationen über Datenebenen, Steuerebenen und wie AWS Services aufbaut, um Hochverfügbarkeitsziele zu erfüllen, finden Sie im Dokument Statische Stabilität mithilfe von Availability Zones.

Erfahren Sie, welche Operationen auf der Datenebene und welche Operationen auf der Steuerebene ausgeführt werden.

Implementierungsschritte

Bewerten Sie für jeden Workload, der nach einem Störfall wiederhergestellt werden muss, das Failover-Runbook, das Hochverfügbarkeitsdesign, das Auto Healing Design oder den Plan zur Wiederherstellung von HA-Ressourcen. Identifizieren Sie jede Aktion, die als Aktion auf der Steuerebene in Frage kommt.

Ziehen Sie in Erwägung, eine Aktion auf der Steuerebene in eine Aktion auf der Datenebene umzuwandeln:

- Auto Scaling (Steuerebene) im Vergleich zu vorab skalierten Amazon EC2-Ressourcen (Datenebene)
- Migrieren Sie zu Lambda und seinen Skalierungsmethoden (Datenebene) oder Amazon EC2 und ASG (Steuerebene)
- Bewerten Sie alle Entwürfe unter Verwendung von Kubernetes und der Art der Aktionen auf der Steuerebene. Das Hinzufügen von Pods ist eine Aktion auf der Datenebene von Kubernetes. Aktionen sollten sich auf das Hinzufügen von Pods und nicht von Knoten beschränken. Mit der Verwendung von überdimensionierten Knoten ist die bevorzugte Methode zur Begrenzung von Aktionen auf der Steuerebene.

Ziehen Sie alternative Ansätze in Betracht, bei denen Aktionen auf der Datenebene dieselbe Maßnahme bewirken können.

- Route 53 Record change (control plane) or ARC (data plane) (Route 53-Datensatzänderung (Steuerebene) oder Route 53 ARC (Datenebene))
- Route 53 Health checks for more automated updates (Route 53-Zustandsprüfungen für weitere automatisierte Aktualisierungen)

Ziehen Sie einige Services in einer sekundären Region in Betracht, wenn der Service geschäftskritisch ist, um mehr Aktionen auf der Steuerebene und Datenebene in einer nicht betroffenen Region zu ermöglichen.

- Amazon EC2 Auto Scaling oder Amazon EKS in einer primären Region im Vergleich zu Amazon EC2 Auto Scaling oder Amazon EKS in einer sekundären Region und Routing des Datenverkehrs zur sekundären Region (Aktion auf Steuerebene)
- Ein Lesereplikat in der sekundären primären Region erstellen oder Versuchen derselben Aktion in der primären Region (Aktion auf der Steuerebene)

Ressourcen

Zugehörige bewährte Methoden:

- Definition der Verfügbarkeit
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Automatisierung der Fehlertoleranz unterstützen können
- AWS Marketplace: Zur Erzielung von Fehlertoleranz geeignete Produkte
- Amazon Builders' Library: Vermeiden von Überlastungen verteilter Systeme durch Übernahme der Steuerung durch den kleineren Service
- Amazon DynamoDB API (Steuerebene und Datenebene)
- AWS Lambda-Ausführungen (aufgeteilt in die Steuerebene und die Datenebene)
- AWS Elemental MediaStore-Datenebene
- Entwickeln hoch resilienter Anwendungen mit Amazon Route 53 Application Recovery Controller, Teil 1: Stack für eine einzelne Region
- Entwickeln hoch resilienter Anwendungen mit Amazon Route 53 Application Recovery Controller, Teil 2: Stack für mehrere Regionen
- Erstellen von Mechanismen für die Notfallwiederherstellung mit Amazon Route 53
- Was ist Route 53 Application Recovery Controller?
- Kubernetes-Steuerebene und -Datenebene

Zugehörige Videos:

- Back to Basics – Using Static Stability (Zurück zu den Basics – Verwendung statischer Stabilität)
- Building resilient multi-site workloads using AWS global services (Aufbau belastbarer Workloads an mehreren Standorten mit globalen AWS-Services)

Zugehörige Beispiele:

- Vorstellung von Amazon Route 53 Application Recovery Controller
- Amazon Builders' Library: Vermeiden von Überlastungen verteilter Systeme durch Übernahme der Steuerung durch den kleineren Service
- Entwickeln hoch resilienter Anwendungen mit Amazon Route 53 Application Recovery Controller, Teil 1: Stack für eine einzelne Region
- Entwickeln hoch resilienter Anwendungen mit Amazon Route 53 Application Recovery Controller, Teil 2: Stack für mehrere Regionen
- Statische Stabilität mithilfe von Availability Zones

Zugehörige Tools:

- Amazon CloudWatch
- AWS X-Ray

REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität

Workloads sollten statisch stabil sein und nur in einem einzigen Normalmodus ausgeführt werden. Bimodales Verhalten liegt vor, wenn sich der Workload im Normalmodus und im Fehlermodus unterschiedlich verhält.

Sie können beispielsweise versuchen, nach einem Ausfall der Availability Zone eine Wiederherstellung durchzuführen, indem Sie neue Instances in einer anderen Availability Zone starten. Dies kann zu einer bimodalen Reaktion während eines Ausfallmodus führen. Stattdessen sollten Sie Workloads erstellen, die statisch stabil sind und nur in einem Modus betrieben werden. In diesem Beispiel hätten diese Instances vor dem Ausfall in der zweiten Availability Zone bereitgestellt werden sollen. Dieses statische Stabilitätsdesign verifiziert, dass der Workload nur in einem einzigen Modus ausgeführt wird.

Gewünschtes Ergebnis: Workloads zeigen im Normalmodus und im Fehlermodus kein bimodales Verhalten.

Typische Anti-Muster:

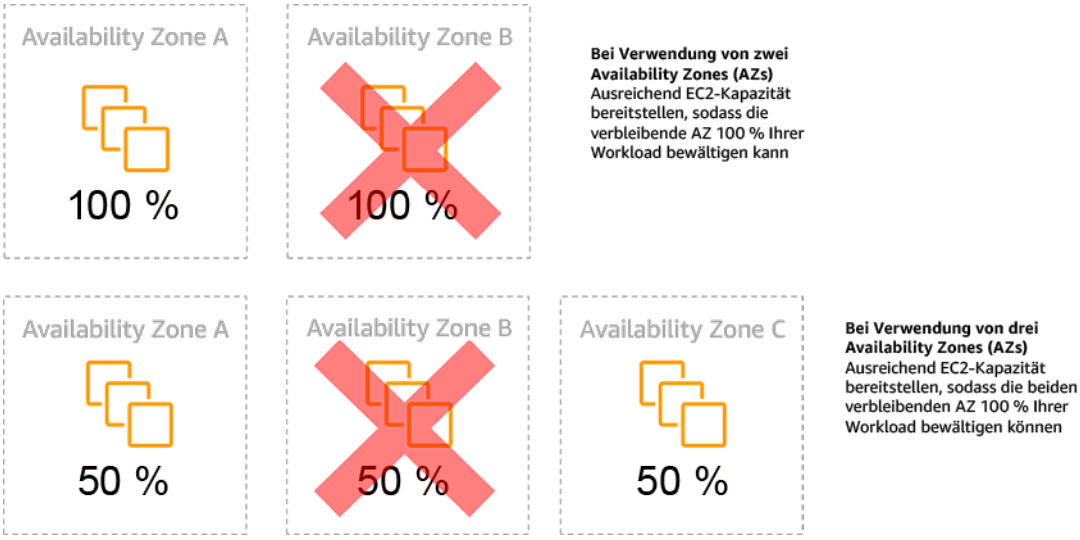
- Es wird davon ausgegangen, dass Ressourcen unabhängig vom Umfang des Fehlers immer bereitgestellt werden können.
- Während eines Fehlers wird versucht, dynamisch Ressourcen zu erwerben.
- Es werden keine ausreichenden Ressourcen für Zonen oder Regionen bereitgestellt, bis ein Fehler auftritt.
- Statische stabile Designs werden nur für Rechenressourcen in Erwägung gezogen.

Vorteile der Nutzung dieser bewährten Methode: Workloads, die mit statisch stabilen Designs ausgeführt werden, sind in der Lage, bei normalen Ereignissen und bei Ausfällen vorhersehbare Ergebnisse erzielen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Bimodales Verhalten bedeutet, dass ein Workload im normalen Modus und im Fehlermodus unterschiedliche Verhaltensweisen zeigt (z. B. Verlassen auf den Start neuer Instances bei Ausfall einer Availability Zone). Ein Beispiel für bimodales Verhalten ist, wenn stabile Elastic Load Balancing-Designs genügend Instances in jeder Availability Zone bereitstellen, so dass die Verarbeitung der Workload auch beim Entfernen einer Availability Zone gewährleistet ist. Anschließend sollten Sie die beeinträchtigten Instances mithilfe von Elastic Load Balancing oder Amazon Route 53-Zustandsprüfungen entlasten. Nachdem der Datenverkehr verlagert wurde, können Sie AWS Auto Scaling verwenden, um Instances in der ausgefallenen Zone asynchron zu ersetzen und sie in den fehlerfreien Zonen zu starten. Statische Stabilität für die Bereitstellung von Rechenleistung (z. B. EC2-Instances oder -Container) führt zu höchster Zuverlässigkeit.



Statische Stabilität von EC2-Instances in Availability Zones

Dies muss gegen die Kosten für dieses Modell und den geschäftlichen Nutzen der Aufrechterhaltung des Workloads in allen Ausfallsituationen abgewogen werden. Es ist kostengünstiger, weniger Rechenkapazität bereitzustellen und bei einem Ausfall neue Instances zu starten. Bei großen Ausfällen (z. B. bei Beeinträchtigung einer Availability Zone oder Region) ist dieser Ansatz jedoch weniger effektiv, da er sowohl auf einer Betriebsebene als auch auf der Verfügbarkeit ausreichender Ressourcen in den nicht betroffenen Zonen oder Regionen beruht.

Ihre Lösung sollte die Anforderungen an die Zuverlässigkeit und Kosten für Ihren Workload gegeneinander abwägen. Ansätze mit statischer Stabilität gelten für eine Vielzahl von Architekturen, darunter Computing-Instances, die über Availability Zones verteilt sind, Designs mit Lesereplikaten für Datenbanken, Kubernetes (Amazon EKS)-Clusterdesigns und Failover-Architekturen für mehrere Regionen.

Es ist auch möglich, ein statisch stabileres Design zu implementieren, indem mehr Ressourcen in jeder Zone verwendet werden. Wenn Sie eine größere Anzahl von Zonen hinzufügen, verringert sich die Menge der zusätzlichen Rechenleistung, die Sie für die statische Stabilität benötigen.

Ein weiteres Beispiel für bimodales Verhalten ist eine Netzwerk-Zeitüberschreitung, die dazu führen kann, dass ein System versucht, den Konfigurationsstatus des gesamten Systems zu aktualisieren. Dies kann zur unerwarteten Auslastung einer anderen Komponente führen, die daraufhin ausfallen könnte, was möglicherweise weitere unerwartete Konsequenzen nach sich zieht. Diese negative Feedback-Schleife wirkt sich auf die Verfügbarkeit Ihres Workloads aus. Deshalb sollten Sie stattdessen Systeme erstellen, die statisch stabil sind und nur in einem Modus betrieben werden. Ein statisch stabiles Design arbeitet konstant und aktualisiert den Konfigurationsstatus in regelmäßigen

Abständen. Wenn ein Aufruf fehlschlägt, verwendet der Workload den zuvor zwischengespeicherten Wert und löst einen Alarm aus.

Ein weiteres Beispiel für bimodales Verhalten: Sie lassen zu, dass Clients im Fehlerfall den Workload-Cache umgehen. Dies scheint eine Lösung zu sein, die Clientanforderungen erfüllt, sie kann aber die Belastung Ihres Workloads erheblich ändern und führt wahrscheinlich zu Fehlern.

Bewerten Sie kritische Workloads, um festzustellen, für welche Workloads diese Art von Resilienzentwurf erforderlich ist. Für diejenigen, die als kritisch eingestuft werden, muss jede Anwendungskomponente überprüft werden. Beispiele für Services, für die statische Stabilitätsbewertungen erforderlich sind:

- Datenverarbeitung: Amazon EC2, EKS-EC2, ECS-EC2, EMR-EC2
- Datenbanken: Amazon Redshift, Amazon RDS, Amazon Aurora
- 存储: Amazon S3 (eine Zone), Amazon EFS (Bereitstellungen), Amazon FSx (Bereitstellungen)
- Load Balancer: Unter bestimmten Designs

Implementierungsschritte

- Erstellen Sie Systeme, die statisch stabil sind und nur in einem einzigen Modus ausgeführt werden. Stellen Sie in diesem Fall in jeder Availability Zone oder Region genügend Instances bereit, um die Workload-Kapazität zu bewältigen, falls eine Availability Zone oder Region entfernt würde. Eine Vielzahl von Services kann für das Routing zu intakten Ressourcen verwendet werden, z. B.:
  - Regionsübergreifendes DNS-Routing
  - MRAP-Amazon S3-Routing mit mehreren Regionen
  - AWS Global Accelerator
  - Amazon Application Recovery Controller (ARC)
- Konfigurieren Sie Datenbank-Read-Replikate, um den Verlust einer einzelnen primären Instance oder einer Read Replica zu berücksichtigen. Wenn der Datenverkehr von Lesereplikaten bedient wird, sollte die Menge in jeder Availability Zone und jeder Region dem Gesamtbedarf im Fall eines Zonen- oder Regionsausfalls entsprechen.
- Konfigurieren Sie kritische Daten in einem Amazon S3-Speicher, der so konzipiert ist, dass er für die gespeicherten Daten beim Ausfall einer Availability Zone statisch stabil ist. Wenn Amazon S3 One Zone-IA Speicherklasse verwendet wird, sollte diese nicht als statisch stabil angesehen werden, da der Ausfall dieser Zone den Zugriff auf die zugehörigen gespeicherten Daten minimiert.

- Load Balancer sind manchmal falsch oder so konfiguriert, dass sie eine bestimmte Availability Zone bedienen. In diesem Fall könnte das statisch stabile Design darin bestehen, einen Workload über mehrere AZs in einem komplexeren Design zu verteilen. Das ursprüngliche Design kann aus Sicherheits-, Latenz- oder Kostengründen verwendet werden, um den Verkehr zwischen den Zonen zu reduzieren.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- Definition der Verfügbarkeit
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung

Zugehörige Dokumente:

- Minimierung der Abhängigkeiten bei der Planung der Notfallwiederherstellung
- Die Amazon Builders' Library: Statische Stabilität durch Availability Zones
- Fault Isolation Boundaries (Grenzen für die Fehlerisolierung)
- Statische Stabilität mithilfe von Availability Zones
- Multi-Zone RDS (RDS für mehrere Zonen)
- Minimierung der Abhängigkeiten bei der Planung der Notfallwiederherstellung
- Regionsübergreifendes DNS-Routing
- MRAP-Amazon S3-Routing mit mehreren Regionen
- AWS Global Accelerator
- ARC
- Amazon S3 mit einer Zone
- Cross Zone Load Balancing (Zonenübergreifender Lastenausgleich)

Zugehörige Videos:

- Static stability in AWS: AWS re:Invent 2019: Introducing The Amazon Builders' Library (DOP328) (Statische Stabilität in AWS: AWS re:Invent 2019: Einführung der Amazon Builders' Library (DOP328))

Zugehörige Beispiele:

- Die Amazon Builders' Library: Statische Stabilität durch Availability Zones

REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken

Benachrichtigungen werden nach Erkennung von Schwellenwertüberschreitungen gesendet, auch wenn das durch das Ereignis verursachte Problem automatisch behoben wurde.

Auto Healing sorgt dafür, dass Ihr Workload zuverlässig ist. Allerdings können dadurch auch zugrunde liegende Probleme verschleiert werden, die behoben werden müssen. Implementieren Sie geeignete Überwachungsfunktionen und Ereignisse, damit Sie Problemmuster erkennen können, einschließlich solcher, die durch Auto Healing behoben werden. Auf diese Weise können Sie die Fehlerursachen beheben.

Resiliente Systeme sind so konzipiert, dass Verschlechterungsereignisse sofort an die entsprechenden Teams gemeldet werden. Diese Benachrichtigungen sollten über einen oder mehrere Kommunikationskanäle gesendet werden.

Gewünschtes Ergebnis: Bei Überschreitung von Schwellenwerten wie Fehlerraten, Latenz oder anderen kritischen Leistungsindikatoren (KPIs) werden sofort Benachrichtigungen an die Betriebsteams gesendet, sodass diese Probleme so schnell wie möglich behoben und Auswirkungen auf die Benutzer vermieden oder minimiert werden.

Typische Anti-Muster:

- Es werden zu viele Alarme gesendet.
- Es werden Alarme gesendet, die keine Maßnahmen erfordern.
- Die Schwellenwerte für den Alarm sind zu hoch (überempfindlich) oder zu niedrig (nicht empfindlich genug).
- Es werden keine Alarme für externe Abhängigkeiten gesendet.
- Nicht berücksichtigt werden die grauen Fehler bei der Gestaltung von Überwachung und Alarmen.
- Es werden automatische Reparaturen ausgeführt, ohne das entsprechende Team darüber zu benachrichtigen, dass eine Reparatur erforderlich war.

Vorteile der Nutzung dieser bewährten Methode: Durch Benachrichtigungen über die Wiederherstellung werden Betriebs- und Geschäftsteams über Service-Einschränkungen informiert,

sodass sie sofort reagieren können, um sowohl die mittlere Zeit zur Erkennung (Mean Time to Detect, MTTD) als auch die mittlere Wiederherstellungszeit (Mean Time to Repair, MTTR) zu minimieren. Benachrichtigungen zu Wiederherstellungen stellen sicher, dass Sie selten auftretende Probleme nicht ignorieren.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel. Wenn keine geeigneten Überwachungsfunktionen und Mechanismen zur Benachrichtigung bei Ereignissen implementiert werden, kann dies dazu führen, dass Problemmuster nicht erkannt werden, einschließlich solcher, die durch Auto Healing behoben werden. Ein Team wird nur dann auf eine Verschlechterung des Systems aufmerksam gemacht, wenn Benutzer den Kundendienst kontaktieren oder der Fehler zufällig bemerkt wird.

Implementierungsleitfaden

Bei der Definition einer Überwachungsstrategie ist ein ausgelöster Alarm ein häufiges Ereignis. Dieses Ereignis würde wahrscheinlich eine Kennung für den Alarm enthalten, den Alarmstatus (z. B. `ALARM AKTIV` oder `OK`) und Einzelheiten darüber, was ihn ausgelöst hat. In vielen Fällen sollte ein Alarmereignis erkannt und eine E-Mail-Benachrichtigung gesendet werden. Dies ist ein Beispiel für eine Aktion bei einem Alarm. Die Alarmbenachrichtigung ist für die Beobachtbarkeit von entscheidender Bedeutung, da hiermit die richtigen Personen darüber informiert werden, dass ein Problem vorliegt. Wenn die Aktionen bei Ereignissen in Ihrer Lösung für die Beobachtbarkeit ausgereift sind, kann das Problem automatisch behoben werden, ohne dass menschliches Eingreifen erforderlich ist.

Sobald Alarme zur KPI-Überwachung eingerichtet wurden, sollten die entsprechenden Teams Warnmeldungen erhalten, wenn Schwellenwerte überschritten werden. Diese Warnungen können auch verwendet werden, um automatisierte Prozesse auszulösen, die versuchen, die Verschlechterung zu beheben.

Für eine komplexere Schwellenwertüberwachung sollten zusammengesetzte Alarme in Betracht gezogen werden. Zusammengesetzte Alarme verwenden eine Reihe von Alarmen zur KPI-Überwachung, um eine Warnung auf Grundlage der Geschäftslogik zu erstellen. CloudWatch-Alarme können so konfiguriert werden, dass E-Mails gesendet oder Vorfälle mithilfe der Amazon SNS-Integration oder Amazon EventBridge in Drittanbietersystemen zur Nachverfolgung von Vorfällen protokolliert werden.

Implementierungsschritte

Erstellen Sie verschiedene Arten von Alarmen, je nachdem, wie Workloads überwacht werden, z. B.:

- Anwendungsalarme werden verwendet, um zu erkennen, wenn ein Teil des Workloads nicht ordnungsgemäß funktioniert.
- Alarme für die Infrastruktur geben an, wann Ressourcen skaliert werden sollen. Alarme können visuell in Dashboards angezeigt werden, Warnungen per Amazon SNS oder E-Mail senden und mit Auto Scaling die Ressourcen für einen Workload hoch- oder herunterskalieren.
- Einfache statische Alarme können erstellt werden, um zu überwachen, wann eine Metrik für eine bestimmte Anzahl von Bewertungszeiträumen einen statischen Schwellenwert überschreitet.
- Zusammengesetzte Alarme, können komplexe Alarme aus mehreren Quellen berücksichtigen.
- Nachdem der Alarm erstellt wurde, erstellen Sie entsprechende Benachrichtigungsereignisse. Sie können direkt eine Amazon SNS-API aufrufen, um Benachrichtigungen zu senden und alle Automatisierungen zur Behebung oder Kommunikation zu verknüpfen.
- Setzen Sie Amazon Health Aware Überwachung, um die Überwachung von AWS-Ressourcen zu ermöglichen, bei denen es zu Leistungseinbußen kommen könnte. Für geschäftskritische Workloads bietet diese Lösung Zugriff auf proaktive und Echtzeitbenachrichtigungen für AWS-Services.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- Definition der Verfügbarkeit

Zugehörige Dokumente:

- Erstellen eines CloudWatch-Alarms auf der Basis eines statischen Schwellenwerts
- Was ist Amazon EventBridge?
- Was ist Amazon Simple Notification Service?
- Veröffentlichen benutzerdefinierter Metriken
- Using Amazon CloudWatch Alarms (Verwenden von Amazon CloudWatch-Alarmen)
- Amazon Health Aware (AHA)
- Einrichten von zusammengesetzten CloudWatch-Alarmen
- Neuheiten im Bereich AWS-Beobachtbarkeit bei der re:Invent 2022

Zugehörige Tools:

- CloudWatch
- CloudWatch X-Ray

REL11-BP07 Architektur Ihres Produkts zur Erfüllung von Verfügbarkeitszielen und Uptime-SLAs (Service Level Agreements)

Entwerfen Sie Ihr Produkt zur Erfüllung der Verfügbarkeitsziele und der Uptime-SLAs (Service Level Agreements). Wenn Sie Verfügbarkeitsziele oder Uptime-SLAs veröffentlichen oder privat vereinbaren, stellen Sie sicher, dass Ihre Architektur und Ihre operativen Prozesse so konzipiert sind, dass sie diese unterstützen.

Gewünschtes Ergebnis: Jede Anwendung hat ein definiertes Ziel für die Verfügbarkeit und eine SLA für Leistungsmetrik, die überwacht und aufrechterhalten werden können, um die Geschäftsziele zu erreichen.

Typische Anti-Muster:

- Entwurf und Bereitstellung von Workloads ohne Einstellung von SLAs.
- SLA-Metriken werden ohne Begründung oder geschäftliche Anforderungen zu hoch angesetzt.
- SLAs werden ohne Berücksichtigung von Abhängigkeiten und den ihnen zugrunde liegenden SLAs festgelegt.
- Anwendungsdesigns werden ohne Berücksichtigung des Modells der geteilten Verantwortung für die Ausfallsicherheit erstellt.

Vorteile der Nutzung dieser bewährten Methode: Die Entwicklung von Anwendungen auf der Grundlage von Schlüsselzielen für die Ausfallsicherheit hilft Ihnen, Geschäftsziele und Kundenerwartungen zu erfüllen. Diese Ziele sind die Grundlage für die Entwicklung von Anwendungen, bei der verschiedene Technologien bewertet und verschiedene Kompromisse in Betracht gezogen werden.

Implementierungsleitfaden

Bei der Entwicklung von Anwendungen müssen Sie eine Reihe von Anforderungen berücksichtigen, die sich aus geschäftlichen, operativen und finanziellen Zielen ergeben. Im Rahmen der operativen Anforderungen müssen für Workloads spezifische Metriken für die Ausfallsicherheit festgelegt werden, damit sie angemessen überwacht und unterstützt werden können. Die Metriken für die Ausfallsicherheit sollten nicht nach der Bereitstellung des Workloads festgelegt oder ermittelt

werden. Sie sollten in der Entwurfsphase festgelegt werden und als Leitlinien für verschiedene Entscheidungen und Abwägungen dienen.

- Jeder Workload sollte seine eigenen Metriken für die Ausfallsicherheit haben. Diese Metriken können sich von anderen geschäftlichen Anwendungen unterscheiden.
- Die Reduzierung von Abhängigkeiten kann sich positiv auf die Verfügbarkeit auswirken. Jeder Workload sollte seine Abhängigkeiten und deren SLAs berücksichtigen. Wählen Sie im Allgemeinen Abhängigkeiten mit Verfügbarkeitszielen aus, die den Zielen Ihres Workloads entsprechen oder höher sind.
- Ziehen Sie eine lose Kopplung in Betracht, damit Ihr Workload trotz der Beeinträchtigung durch Abhängigkeiten korrekt arbeiten kann, sofern dies möglich ist.
- Reduzieren Sie die Abhängigkeiten auf der Steuerebene, insbesondere während der Wiederherstellung oder einer Beeinträchtigung. Evaluieren Sie Designs, die für geschäftskritische Workloads statisch stabil sind. Nutzen Sie den sparsamen Umgang mit Ressourcen, um die Verfügbarkeit dieser Abhängigkeiten in einem Workload zu erhöhen.
- Die Überwachbarkeit und die Instrumentierung sind entscheidend für das Erreichen von SLAs. Sie reduzieren die Mean Time to Detection (MTTD) und die Mean Time to Repair (MTTR).
- Weniger häufige Störungen (längere MTBF), kürzere Fehlererkennungszeiten (kürzere MTTD) und kürzere Reparaturzeiten (kürzere MTTR) sind die drei Faktoren, die zur Verbesserung der Verfügbarkeit in verteilten Systemen eingesetzt werden.
- Das Festlegen und Einhalten von Metriken für die Ausfallsicherheit eines Workloads ist eine der Grundlagen für jedes effektive Design. Diese Entwürfe müssen Kompromisse in Bezug auf Designkomplexität, Service-Abhängigkeiten, Leistung, Skalierung und Kosten berücksichtigen.

Implementierungsschritte

- Überprüfen und dokumentieren Sie den Workload-Entwurf unter Berücksichtigung der folgenden Fragen:
  - Wo werden die Steuerebenen im Workload verwendet?
  - Wie implementiert der Workload die Ausfallsicherheit?
  - Wie sehen die Entwurfsmuster für die Skalierung, automatische Skalierung, Redundanz und hochverfügbare Komponenten aus?
  - Welche Anforderungen gibt es an die Datenkonsistenz und -verfügbarkeit?
  - Gibt es Überlegungen zur sparsamen Nutzung von Ressourcen oder zur statischen Stabilität von Ressourcen?

  - Welche Abhängigkeiten bestehen zwischen den Services?
- Definieren Sie in Zusammenarbeit mit den Stakeholdern SLA-Metriken auf der Grundlage der Workload-Architektur. Berücksichtigen Sie die SLAs aller Abhängigkeiten, die der Workload nutzt.
- Sobald das SLA-Ziel festgelegt ist, optimieren Sie die Architektur, um die SLA zu erfüllen.
- Sobald das Design festgelegt ist, das die SLA erfüllt, implementieren Sie operative Änderungen, Prozessautomatisierungen und Runbooks, die ebenfalls auf die Reduzierung von MTTD und MTTR ausgerichtet sind.
- Sobald die Bereitstellung erfolgt ist, überwachen Sie die SLA und erstatten Sie darüber Bericht.

Ressourcen

Zugehörige bewährte Methoden:

- REL03-BP01 Segmentierung Ihres Workloads
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler
- REL11-BP03 Automatisieren der Reparatur auf allen Ebenen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering
- REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:
- Grundlegendes zum Workload-Status

Zugehörige Dokumente:

- Availability with redundancy (Verfügbarkeit mit Redundanz)
- Zuverlässigkeitssäule – Verfügbarkeit
- Measuring availability (Messung der Verfügbarkeit)
- AWS Fault Isolation Boundaries (AWS-Grenzen für die Fehlerisolierung)
- Modell der geteilten Verantwortung für Ausfallsicherheit
- Statische Stabilität mithilfe von Availability Zones
- AWS Service Level Agreements (SLAs)
- Guidance for Cell-based Architecture on AWS (Leitfaden für eine zellenbasierte Architektur auf AWS)
- AWS-Infrastruktur

- Advanced Multi-AZ Resiliance Patterns whitepaper (Whitepaper: Fortschrittliche Multi-AZ-Resiliance-Muster)

Zugehörige Services:

- Amazon CloudWatch
- AWS Config
- AWS Trusted Advisor

REL 12. Wie lässt sich die Zuverlässigkeit testen?

Nachdem Sie Ihre Workload so konzipiert haben, dass sie den Belastungen der Produktion standhält, sind Tests die einzige Möglichkeit, sie auf die erwartete Funktionalität und Ausfallsicherheit hin zu testen.

Bewährte Methoden

- REL12-BP01 Untersuchen von Fehlern mit Playbooks:
- REL12-BP02 Durchführen von Analysen nach Vorfällen
- REL12-BP03 Testen funktionaler Anforderungen
- REL12-BP04 Testen von Skalierungs- und Leistungsanforderungen
- REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering
- REL12-BP06 Regelmäßiges Abhalten von Gamedays

REL12-BP01 Untersuchen von Fehlern mit Playbooks:

Ermöglichen Sie konsistente und schnelle Antworten auf noch unbekannte Fehlerszenarien, indem Sie den Untersuchungsprozess in Playbooks dokumentieren. Playbooks sind vordefinierte Abläufe zum Identifizieren der Faktoren, die zu einem Fehlerszenario beitragen. Die Ergebnisse aus jedem Prozessschritt sind die Grundlage für die nächsten Schritte. Nach diesem Muster wird vorgegangen, bis das Problem identifiziert oder eskaliert wird.

Das Playbook ist eine proaktive Planung, die für effektive Reaktionen erforderlich ist. Wenn nicht vom Playbook abgedeckte Fehlerszenarien in der Produktion auftreten, beheben Sie zunächst das Problem. Analysieren Sie danach die unternommenen Schritte und verwenden Sie diese, um einen neuen Eintrag im Playbook hinzuzufügen.

Beachten Sie, dass Playbooks als Reaktion auf bestimmte Vorfälle verwendet werden, während Runbooks verwendet werden, um bestimmte Ergebnisse zu erzielen. Häufig werden Runbooks für Routineaktivitäten verwendet, Playbooks hingegen, um auf außergewöhnliche Ereignisse zu reagieren.

Gängige Antimuster:

- Planen der Bereitstellung eines Workloads, ohne die Prozesse für die Diagnose von Problemen oder die Reaktion auf Vorfälle zu kennen.
- Ungeplante Entscheidungen darüber, in welchen Systemen bei der Untersuchung von Ereignissen Protokolle und Metriken erfasst werden sollen.
- Metriken und Ereignisse werden nicht lange genug aufbewahrt, um die Daten abrufen zu können.

Vorteile der Einführung dieser bewährten Methode: Durch das Erfassen von Playbooks wird sichergestellt, dass Prozesse konsistent befolgt werden können. Ihre Playbooks werden als Code festgehalten, um die Entstehung von Fehlern durch manuelle Aktivitäten zu reduzieren. Durch die Automatisierung von Playbooks kann schneller auf Ereignisse reagiert werden, weil Teammitglieder nicht eingreifen müssen oder ihnen vor dem Eingreifen zusätzliche Informationen zur Verfügung gestellt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Ermitteln von Probleme mit Playbooks. Playbooks sind dokumentierte Prozesse für die Untersuchung von Problemen. Durch die Dokumentation der Prozesse in Playbooks schaffen Sie die Voraussetzung für eine einheitliche und schnelle Reaktion auf Fehlerszenarien. Playbooks müssen die Informationen und Anleitungen enthalten, die eine entsprechend qualifizierte Person zum Zusammentragen sachdienlicher Informationen, zum Identifizieren möglicher Fehlerursachen, zum Isolieren von Fehlern und zum Bestimmen beitragender Faktoren (zum Analysieren nach einem Vorfall) benötigt.
  - Implementieren von Playbooks als Code. Führen Sie Ihre Operationen als Code aus, indem Sie Skripts für Ihre Playbooks erstellen, um Konsistenz sicherzustellen und Fehler zu reduzieren, die durch manuelle Prozesse verursacht werden. Playbooks können aus mehreren Skripts bestehen, die die verschiedenen Schritte darstellen, die erforderlich sein können, um die zu einem Problem beitragenden Faktoren zu identifizieren. Runbook-Aktivitäten können ausgelöst oder im Rahmen von Playbook-Aktivitäten ausgeführt werden. Sie können auch als Antwort auf identifizierte Ereignisse die Ausführung eines Playbooks auslösen.

- Automatisieren Sie Ihre operativen Playbooks mit AWS Systems Manager
- AWS Systems Manager Befehl ausführen
- AWS Systems Manager Automation
- Was ist AWS Lambda?
- Was ist Amazon EventBridge?
- Verwenden von Amazon CloudWatch Alarmen

Ressourcen

Zugehörige Dokumente:

- AWS Systems Manager Automation
- AWS Systems Manager Befehl ausführen
- Automatisieren Sie Ihre operativen Playbooks mit AWS Systems Manager
- Verwenden von Amazon CloudWatch Alarmen
- Verwenden von Canaries (Amazon CloudWatch Synthetics)
- Was ist Amazon EventBridge?
- Was ist AWS Lambda?

Ähnliche Beispiele:

- Automatisieren von Vorgängen mit Playbooks und Runbooks

REL12-BP02 Durchführen von Analysen nach Vorfällen

Überprüfen Sie die Ereignisse mit Auswirkungen auf Kunden und bestimmen Sie die beitragenden Faktoren und Präventivmaßnahmen. Entwickeln Sie anhand dieser Informationen Abhilfemaßnahmen, um ein wiederholtes Auftreten nach Möglichkeit zu verhindern. Entwickeln Sie Verfahren für schnelle und effektive Reaktionen. Informieren Sie nach Bedarf auf zielgruppengerechte Weise über beitragende Faktoren und Korrekturmaßnahmen. Legen Sie eine Kommunikationsmethode fest, um andere bei Bedarf über die Ursachen zu informieren.

Bewerten Sie, warum bestehende Tests das Problem nicht gefunden haben. Fügen Sie Tests für diesen Fall hinzu, wenn noch keine Tests vorhanden sind.

Gewünschtes Ergebnis: Ihre Teams verfolgen einen konsistenten und vereinbarten Ansatz für die Analyse nach einem Vorfall. Einer dieser Mechanismen ist der COE-Prozess (Correction of Error, Fehlerkorrektur). Der COE-Prozess hilft Ihren Teams, die Ursachen für Vorfälle zu identifizieren, zu verstehen und zu beseitigen. Gleichzeitig werden Mechanismen und Leitlinien entwickelt, um die Wahrscheinlichkeit zu verringern, dass sich ein solcher Vorfall wiederholt.

Typische Anti-Muster:

- Beitragende Faktoren werden ermittelt, es wird jedoch nicht weiter nach anderen potenziellen Problemen und Lösungsansätzen gesucht.
- Es werden nur menschliche Fehlerursachen ermittelt, es wird aber keine Schulung oder Automatisierung bereitgestellt, die menschliche Fehler verhindern könnte.
- Der Fokus liegt auf Schuldzuweisungen, anstatt die Ursache zu verstehen, wodurch eine Kultur der Angst entsteht und eine offene Kommunikation behindert wird.
- Es wird versäumt, Erkenntnisse weiterzugeben, wodurch die Ergebnisse der Ereignisanalyse in einer kleinen Gruppe bleiben und andere nicht von den gewonnenen Erkenntnissen profitieren können.
- Es gibt keine Mechanismen zur Erfassung des institutionellen Wissens, wodurch wertvolle Erkenntnisse verloren gehen, da die gewonnenen Erkenntnisse nicht in Form von aktualisierten bewährten Methoden festgehalten werden und es zu wiederholten Vorfällen mit derselben oder einer ähnlichen Ursache kommt.

Vorteile der Nutzung dieser bewährten Methode: Durch Analysen von Vorfällen und das Teilen von Ergebnissen können die Risiken für andere Workloads mit den gleichen beitragenden Faktoren verringert werden. Außerdem können Abhilfemaßnahmen oder automatisierte Wiederherstellungen implementiert werden, bevor es zu einem Vorfall kommt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Durch gute Analysen nach Vorfällen lassen sich allgemeine Lösungen für Probleme mit Architekturmustern ermitteln, die Sie bereits an anderer Stelle in den Systemen anwenden.

Ein Grundpfeiler des COE-Prozesses ist die Dokumentation und Behandlung von Problemen. Es wird empfohlen, ein standardisiertes Verfahren zur Dokumentation kritischer Ursachen festzulegen und sicherzustellen, dass diese überprüft und behoben werden. Weisen Sie die Verantwortung für den

Analyseprozess nach einem Vorfall eindeutig zu. Benennen Sie ein verantwortliches Team oder eine Person, die die Untersuchungen von Vorfällen und die Folgemaßnahmen beaufsichtigt.

Fördern Sie eine Kultur, die sich auf Lernen und Verbesserung konzentriert, anstatt Schuldzuweisungen vorzunehmen. Betonen Sie, dass das Ziel darin besteht, zukünftige Vorfälle zu verhindern, und nicht darin, Einzelpersonen zu strafen.

Entwickeln Sie klar definierte Verfahren für die Durchführung von Analysen nach einem Vorfall. Diese Verfahren sollten die zu ergreifenden Schritte, die zu sammelnden Informationen und die Schlüsselfragen, die während der Analyse zu behandeln sind, darlegen. Untersuchen Sie Vorfälle gründlich und gehen Sie dabei über die unmittelbaren Ursachen hinaus, um die Grundursachen und die beitragenden Faktoren zu ermitteln. Verwenden Sie Techniken wie die 5-Why-Methode, um sich eingehend mit den zugrundeliegenden Problemen zu befassen.

Führen Sie eine Sammlung von Erkenntnissen, die Sie aus der Analyse von Vorfällen gewonnen haben. Dieses institutionelle Wissen kann als Referenz für zukünftige Vorfälle und Präventionsmaßnahmen dienen. Tauschen Sie die Ergebnisse und Erkenntnisse aus den Analysen nach dem Vorfall aus und erwägen Sie, offene Besprechungen nach dem Vorfall abzuhalten, um die gewonnenen Erkenntnisse zu diskutieren.

Implementierungsschritte

- Achten Sie bei der Analyse nach einem Vorfall darauf, dass der Prozess frei von Schuldzuweisungen ist. Dies ermöglicht es den an dem Vorfall beteiligten Personen, die vorgeschlagenen Korrekturmaßnahmen sachlich zu beurteilen und fördert eine ehrliche Selbsteinschätzung und die Zusammenarbeit zwischen den Teams.
- Definieren Sie eine standardisierte Methode zur Dokumentation kritischer Probleme. Ein solches Dokument könnte beispielsweise folgendermaßen strukturiert sein:
  - Was ist passiert?
  - Welche Auswirkungen gab es auf Kunden und Ihr Unternehmen?
  - Was war die Ursache?
  - Welche Daten haben Sie, um dies zu unterstützten?
    - Zum Beispiel Metriken und Grafiken
  - Welches waren die kritischen Auswirkungen auf die Säulen, insbesondere in puncto Sicherheit?
    - Beim Entwerfen von Workloads sollten Sie je nach Geschäftskontext zwischen den einzelnen Säulen abwägen. Diese Geschäftsentscheidungen können Ihre technischen Prioritäten beeinflussen. Sie können optimieren, um Kosten zulasten der Zuverlässigkeit in

Entwicklungsumgebungen zu senken, oder Sie können bei unternehmenskritischen Lösungen die Zuverlässigkeit mit höheren Kosten optimieren. Sicherheit ist immer oberstes Gebot, da Sie Ihre Kunden schützen müssen.

- Welche Erkenntnisse haben Sie gewonnen?
- Welche Maßnahmen ergreifen Sie?
  - Aktionspunkte
  - Verwandte Artikel

- Erstellen Sie klar definierte Standardverfahren für die Durchführung von Analysen nach einem Vorfall.
- Richten Sie ein standardisiertes Verfahren zur Meldung von Vorfällen ein. Dokumentieren Sie alle Vorfälle ausführlich, einschließlich des ersten Vorfallberichts, der Protokolle, der Kommunikation und der während des Vorfalls getroffenen Maßnahmen.
- Denken Sie daran, dass ein Vorfall nicht unbedingt einen Ausfall zur Folge haben muss. Es könnte sich um einen Beinahe-Unfall handeln oder um ein System, das auf unerwartete Weise funktioniert und dennoch seine Geschäftsfunktion erfüllt.
- Verbessern Sie Ihren Analyseprozess nach einem Vorfall kontinuierlich auf Grundlage von Rückmeldungen und gewonnenen Erkenntnissen.
- Halten Sie die wichtigsten Erkenntnisse in einem Wissensmanagementsystem fest und überlegen Sie, welche Muster in Entwicklerhandbücher oder Checklisten vor der Bereitstellung aufgenommen werden sollten.

Ressourcen

Zugehörige Dokumente:

- Darum sollten Sie eine Fehlerkorrektur (COE) entwickeln

Zugehörige Videos:

- Amazon’s approach to failing successfully (Amazons Ansatz zum erfolgreichen Scheitern)
- AWS re:Invent 2021 - Amazon Builders’ Library: Operational Excellence at Amazon (Die Amazon Builders' Library: Operative Exzellenz von Amazon)

REL12-BP03 Testen funktionaler Anforderungen

Verwenden Sie Techniken wie Komponenten- und Integrationstests, mit denen die erforderliche Funktionalität validiert wird.

Im Idealfall sollten diese Tests automatisch als Teil von Build- und Bereitstellungsaktionen ausgeführt werden. Mit AWS CodePipeline übergeben Entwickler beispielsweise Änderungen an ein Quell-Repository, in dem CodePipeline die Änderungen automatisch erkennt. Diese Änderungen werden vorgenommen und Tests werden ausgeführt. Nachdem die Tests abgeschlossen sind, wird der erstellte Code für Tests auf Staging-Servern bereitgestellt. Auf dem Staging-Server führt CodePipeline weitere Tests aus, z. B. Integrations- oder Belastungstests. Nach dem erfolgreichen Abschluss dieser Tests stellt CodePipeline den getesteten und genehmigten Code für Produktions-Instances bereit.

Außerdem zeigen frühere Erfahrungen, dass synthetische Transaktionstests (auch bekannt als Canary-Tests, aber nicht zu verwechseln mit Canary-Bereitstellungen), die ausgeführt werden können und das Kundenverhalten simulieren, zu den wichtigsten Testprozessen gehören. Führen Sie diese Tests für Ihre Workload-Endpunkte konstant von verschiedenen Remote-Standorten aus. Mit Amazon CloudWatch Synthetics können Sie Canaries erstellen, um Ihre Endpunkte und APIs zu überwachen.

Risikostufe, falls diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Testen funktionaler Anforderungen: Dazu gehören Komponenten- und Integrationstests, mit denen die erforderliche Funktionalität validiert wird.
  - Verwenden von AWS CodeBuild mit CodePipeline zum Testen von Code und zum Ausführen von Builds
  - AWS CodePipeline Adds Support for Unit and Custom Integration Testing with AWS CodeBuild (AWS CodePipeline fügt Unterstützung für Komponententests und angepasste Integrationstests mit AWS CodeBuild hinzu)
  - Kontinuierliche Bereitstellung und kontinuierliche Integration
  - Using synthetic monitoring (Amazon CloudWatch Synthetics) (Verwenden von synthetischer Überwachung (Amazon CloudWatch Synthetics))
  - Automatisierung von Softwaretests

Ressourcen

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Implementierung einer Continuous Integration-Pipeline unterstützen können
- AWS CodePipeline Adds Support for Unit and Custom Integration Testing with AWS CodeBuild (AWS CodePipeline fügt Unterstützung für Komponententests und angepasste Integrationstests mit AWS CodeBuild hinzu)
- AWS Marketplace: Für die kontinuierliche Integration geeignete Produkte
- Kontinuierliche Bereitstellung und kontinuierliche Integration
- Automatisierung von Softwaretests
- Verwenden von AWS CodeBuild mit CodePipeline zum Testen von Code und zum Ausführen von Builds
- Using synthetic monitoring (Amazon CloudWatch Synthetics) (Verwenden von synthetischer Überwachung (Amazon CloudWatch Synthetics))

REL12-BP04 Testen von Skalierungs- und Leistungsanforderungen

Verwenden Sie Techniken wie Lasttests, um zu überprüfen, ob die Workload die Skalierungs- und Leistungsanforderungen erfüllt.

In der Cloud können Sie bei Bedarf eine Testumgebung für Ihren Workload in Produktionsumgebungen erstellen. Wenn Sie diese Tests auf einer herunterskalierten Infrastruktur ausführen, müssen Sie die Ergebnisse auf den Maßstab der Produktionsumgebung hochrechnen. Last- und Leistungstests können auch in der Produktion durchgeführt werden. Achten Sie dabei darauf, Benutzer nicht zu beeinträchtigen und Ihre Testdaten mit Tags zu versehen, sodass sie nicht mit Benutzerdaten vermischt werden und Nutzungsstatistiken oder Produktionsberichte verfälschen.

Stellen Sie mit Tests sicher, dass Ihre Basisressourcen, Skalierungseinstellungen, Servicekontingente und die Ausfallsicherheit unter Auslastung wie erwartet funktionieren.

Risikostufe, wenn diese Best Practice nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Testen Sie Skalierungs- und Leistungsanforderungen. Führen Sie Lasttests durch, um zu prüfen, ob der Workload die Skalierungs- und Leistungsanforderungen erfüllt.

- Verteilte Lasttests auf AWS: Simulation Tausender verbundener Benutzer
- Apache JMeter
  - Stellen Sie Ihre Anwendung in einer Umgebung bereit, die mit Ihrer Produktionsumgebung identisch ist, und führen Sie einen Lasttest durch.
    - Erstellen Sie auf Grundlage von "Infrastructure as Code"-Konzepten eine Umgebung, die Ihrer Produktionsumgebung möglichst ähnlich ist.

Ressourcen

Zugehörige Dokumente:

- Verteilte Lasttests auf AWS: Simulation Tausender verbundener Benutzer
- Apache JMeter

REL12-BP05 Testen der Ausfallsicherheit mit Chaos-Engineering

Führen Sie regelmäßig Chaos-Experimente in oder nahe an Produktionsumgebungen aus, um zu verstehen, wie Ihr System auf ungünstige Bedingungen reagiert.

Gewünschtes Ergebnis:

Die Ausfallsicherheit der Workload wird regelmäßig durch die Anwendung von Chaos-Engineering in Form von Fehlerinjektionsexperimenten oder einer Injektion unerwarteter Last überprüft. Dazu kommen Tests der Ausfallsicherheit, um das bekannte erwartete Verhalten der Workload während eines Ereignisses zu validieren. Kombinieren Sie Chaos-Engineering mit Tests der Ausfallsicherheit, um sicher zu sein, dass Ihre Workload Komponentenausfällen standhalten und sich von unerwarteten Unterbrechungen erholen kann – mit minimalen oder gar keinen Auswirkungen.

Typische Anti-Muster:

- Auslegung der Systeme auf Ausfallsicherheit, aber keine Überprüfung, wie die Workload als Ganzes funktioniert, wenn Fehler auftreten.
- Keine Experimente unter echten Bedingungen und der erwarteten Last.
- Keine Behandlung der Experimente als Code und fehlendes Aufrechterhalten während des Entwicklungszyklus.
- Keine Durchführung von Chaosexperimenten als Teil Ihrer CI/CD-Pipeline und außerhalb von Bereitstellungen.

- Keine Nutzung früherer Analysen nach Vorfällen bei der Entscheidung über die Fehler, mit denen experimentiert werden soll.

Vorteile der Nutzung dieser bewährten Methode: Durch die Injektion von Fehlern zur Überprüfung der Resilienz Ihres Workloads gewinnen Sie die nötige Zuversicht, dass die Wiederherstellungsverfahren Ihres resilienten Entwurfs im Fall eines realen Fehlers funktionieren.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Das Chaos-Engineering bietet Ihren Teams die nötigen Chancen, um auf kontrollierte Weise kontinuierlich reale Störungen (Simulationen) auf Serviceanbieter-, Infrastruktur-, Workload- und Komponentenebene zu injizieren – mit nur minimalen oder gar keinen Auswirkungen auf Ihre Kunden. Ihre Teams können so aus Fehlern lernen und die Resilienz Ihrer Workloads beobachten, messen und verbessern. Darüber hinaus können sie überprüfen, ob Warnungen ausgelöst werden und die Teams über Ereignisse benachrichtigt werden.

Bei kontinuierlicher Ausführung kann das Chaos-Engineering Mängel in Ihren Workloads aufzeigen, die sich negativ auf Verfügbarkeit und Ausführung auswirken könnten, wenn sie nicht behoben werden.

> **Note**
>
> Beim Chaos-Engineering geht es um das Experimentieren mit einem System, um sich davon zu überzeugen, dass das System in der Produktion auch außergewöhnlichen Bedingungen standhalten kann. – Grundlagen des Chaos-Engineering

Wenn ein System diesen Disruptionen standhalten kann, sollte das Chaos-Experiment weiter als automatisierter Regressionstest ausgeführt werden. In dieser Form sollten Chaos-Experimente als Teil Ihres Systementwicklungszyklus (Systems Development Lifecycle, SDLC) und Ihrer CI/CD-Pipeline ausgeführt werden.

Um sicherzustellen, dass Ihr Workload resilient gegenüber dem Ausfall von Komponenten ist, sollten Sie im Rahmen Ihrer Experimente Ereignisse aus der Praxis injizieren. Sie könnten beispielsweise mit dem Verlust von Amazon EC2-Instances oder einem Failover der primären Amazon RDS-Datenbank-Instance experimentieren und so verifizieren, dass Ihr Workload nicht beeinträchtigt wird

(oder nur minimal beeinträchtigt wird). Mit einer Kombination von Komponentenfehlern könnten Sie Ereignisse simulieren, die von einer Disruption in einer Availability Zone verursacht werden könnten.

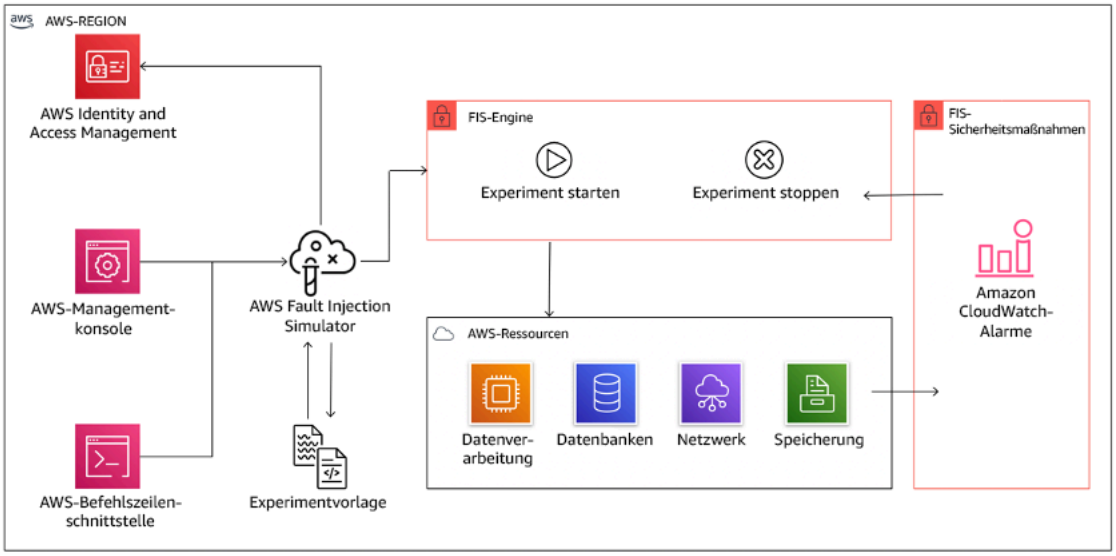
Hinsichtlich Fehlern auf Anwendungsebene (z. B. Abstürzen) könnten Sie mit Stressfaktoren wie Speicher- und CPU-Auslastung beginnen.

Zur Validierung von Fallback- oder Failover-Mechanismen für externe Abhängigkeiten, die bei zeitweisen Netzwerkdisruptionen ausgelöst werden, sollten Ihre Komponenten diese Ereignisse durch das Blockieren des Zugriffs auf externe Anbieter über einen bestimmten Zeitraum simulieren, der von wenigen Sekunden bis zu mehreren Stunden dauern kann.

Andere Degradierungsmodi führen möglicherweise zu einer reduzierten Funktionalität und zu verzögerten Reaktionen, was eine Disruption Ihrer Services verursachen kann. Bekannte Quellen für diese Degradierung sind eine erhöhte Latenz bei kritischen Services und eine unzuverlässige Netzwerkkommunikation (Verlust von Paketen). Experimente mit diesen Fehlern, darunter Netzwerkeffekten wie Latenz, Nachrichtenverlust und DNS-Ausfällen, könnten die fehlende Fähigkeit zur Auflösung eines Namens, zum Erreichen des DNS-Service oder zur Herstellung von Verbindungen zu abhängigen Services umfassen.

Chaos-Engineering-Tools:

AWS Fault Injection Service (AWS FIS) ist ein vollständig verwalteter Service für die Injektion von Fehlern, den Sie innerhalb oder außerhalb Ihrer CD-Pipeline verwenden können, um mit diesen Fehlern zu experimentieren. AWS FIS ist eine gute Wahl für Gamedays, die dem Chaos-Engineering gewidmet sind. Der Service unterstützt die gleichzeitige Injektion von Fehlern in verschiedene Arten von Ressourcen, darunter Amazon EC2, Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) und Amazon RDS. Zu diesen Fehlern gehören die Beendigung von Ressourcen, die Erzwingung von Failovern, die Auslastung von CPU oder Arbeitsspeicher, Drosselung, Latenz und Paketverluste. Da dieser Service in Amazon CloudWatch Alarms integriert ist, können Sie Stoppbedingungen als Integritätsschutz einrichten, um Experimente rückgängig zu machen, wenn sie unerwartete Auswirkungen haben.



Diagramm, das die Integration von AWS Fault Injection Service in AWS-Ressourcen zeigt, um Ihnen die Ausführung von Fehlerinjektionsexperimenten für Ihre Workloads zu ermöglichen.

Es gibt auch verschiedene Drittanbieteroptionen für Fehlerinjektionsexperimente. Dazu gehören Open-Source-Tools wie Chaos Toolkit, Chaos Meshund Litmus Chaossowie kommerzielle Optionen wie Gremlin. Zur Erweiterung der Art der Fehler, die in AWS injiziert werden können, kann AWS FIS in Chaos Mesh und Litmus Chaos integriert werden.So können Sie Fehlerinjektions-Workflows über verschiedene Tools hinweg koordinieren. Sie können beispielsweise einen Stresstest für die CPU eines Pods mit Chaos-Mesh- oder Litmus-Fehlern ausführen und gleichzeitig einen zufällig ausgewählten Prozentsatz von Cluster-Knoten mit AWS FIS-Fehleraktionen beenden.

Implementierungsschritte

- Ermitteln Sie die Fehler, mit denen experimentiert werden soll.

  Bewerten Sie das Design Ihres Workloads in Bezug auf die Resilienz. Diese Designs (anhand der Best Practices des Well-Architected Frameworkerstellt) berücksichtigen Risiken im Zusammenhang mit kritischen Abhängigkeiten, früheren Ereignissen, bekannten Problemen und Compliance-Anforderungen. Listen Sie die einzelnen Elemente des Designs auf, die Resilienz zeigen sollen, und die Fehler, denen es standhalten soll. Weitere Informationen zur Erstellung dieser Listen finden Sie im Whitepaper zur Überprüfung der betrieblichen Bereitschaft. Dieses Whitepaper führt Sie durch die Entwicklung eines Prozesses zur Verhinderung der Wiederholung früherer Vorfälle. Der Prozess für die Analyse von Fehlerarten und ihren Auswirkungen (Failure Modes and Effects Analysis, FMEA) stellt Ihnen ein Framework für Fehleranalysen auf Komponentenebene und die Analyse der Auswirkungen dieser Fehler auf Ihren Workload bereit. FMEA wird von Adrian

Cockcroft in Failure Modes and Continuous Resilience(Fehlerarten und kontinuierliche Resilienz) detaillierter beschrieben.

- Weisen Sie jedem Fehler eine Priorität zu.

  Beginnen Sie mit einer groben Kategorisierung wie hoch, mittel oder niedrig. Berücksichtigen Sie bei der Festlegung der Priorität die Häufigkeit des Fehlers und die Auswirkungen des Fehlers auf den Workload insgesamt.

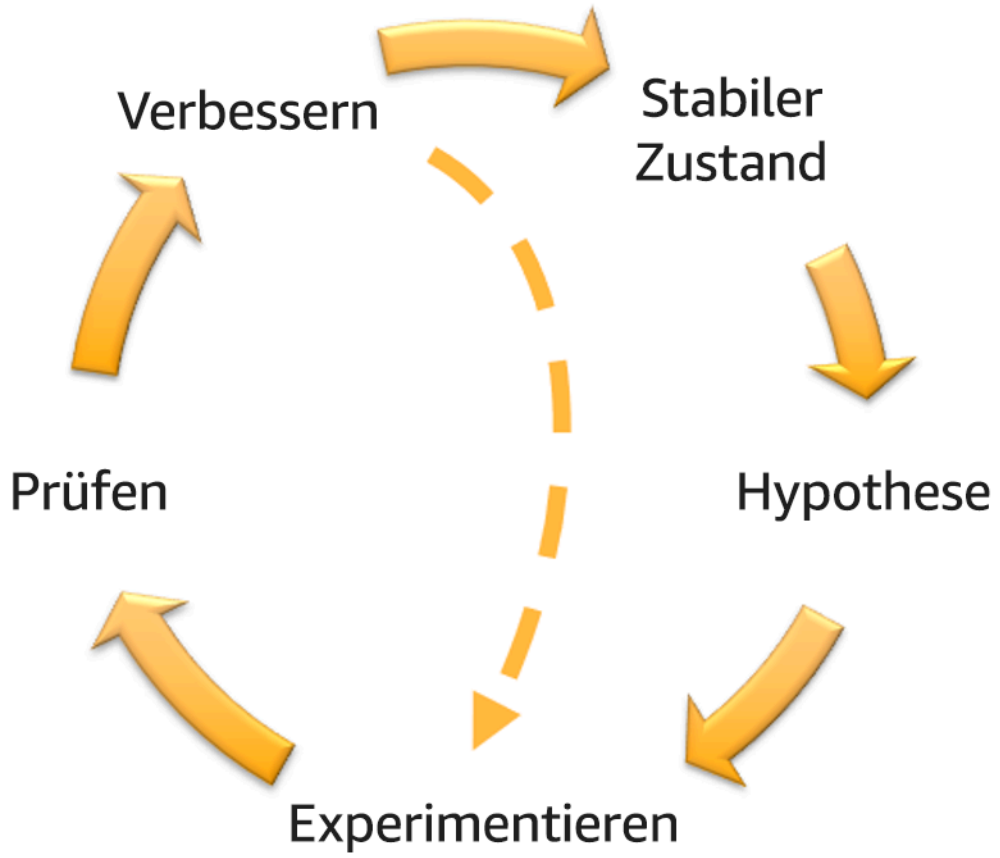
  Analysieren Sie hinsichtlich der Häufigkeit eines bestimmten Fehlers frühere Daten für den betreffenden Workload, wenn verfügbar. Wenn keine Daten verfügbar sind, verwenden Sie Daten zu anderen Workloads, die in einer ähnlichen Umgebung ausgeführt werden.

  Bei der Betrachtung der Auswirkungen eines bestimmten Fehlers gilt, dass die Auswirkungen im Allgemeinen umso größer sind, je größer der vom Fehler betroffene Bereich ist. Sie sollten auch das Design und den Zweck des Workloads berücksichtigen. Beispielsweise ist für einen Workload, der Daten transformiert und analysiert, der Zugriff auf die Quelldatenspeicher von kritischer Bedeutung. In diesem Fall würden Sie Experimente im Zusammenhang mit Zugriffsfehlern, Zugriffsdrosselungen und Latenzen priorisieren.

  Nach Vorfällen durchgeführte Analysen stellen eine gute Datenquelle dar, um Häufigkeit und Auswirkungen von Fehlerarten besser zu verstehen.

  Legen Sie anhand der zugewiesenen Priorität die Fehler fest, mit denen zuerst experimentiert werden soll, und die Reihenfolge, in der neue Fehlerinjektionsexperimente entwickelt werden sollen.

- Für jedes von Ihnen ausgeführte Experiment sollten Sie sich am Schwungrad für Chaos-Engineering und kontinuierliche Resilienz orientieren.



Schwungrad für Chaos-Engineering und kontinuierliche Resilienz unter Verwendung der wissenschaftlichen Methode von Adrian Hornsby.

- Definieren Sie den Steady-State als die messbare Ausgabe eines Workloads, der ein normales Verhalten zeigt.

  Ihr Workload befindet sich im Steady-State, wenn er zuverlässig und wie erwartet ausgeführt wird. Daher sollten Sie die Integrität Ihres Workloads überprüfen, bevor Sie den Steady-State definieren. Steady-State bedeutet nicht notwendigerweise, dass sich ein Fehler nicht auf den Workload auswirkt, da ein bestimmter Prozentsatz an Fehlern innerhalb akzeptabler Grenzen liegen könnte. Der Steady-State ist die Basislinie, die Sie während des Experiments beobachten. Diese wird Anomalien aufweisen, wenn Ihre Hypothese, die Sie im nächsten Schritt definieren, nicht die erwarteten Ergebnisse zeigt.

  Der Steady-State eines Zahlungssystems kann beispielsweise als die Verarbeitung von 300 TPS mit einer Erfolgsrate von 99 % und einer Roundtrip-Zeit von 500 ms definiert sein.

- Formulieren Sie eine Hypothese dazu, wie der Workload auf den Fehler reagieren wird.

  Eine gute Hypothese basiert darauf, wie der Workload den Fehler voraussichtlich bewältigt, um den Steady-State zu wahren. Die Hypothese besagt, dass bei einem Fehler eines spezifischen Typs das System oder der Workload weiter im Steady-State bleiben, da der Workload mit bestimmten Resilienzmerkmalen entworfen wurde. Der spezifische Fehlertyp und die Fehlerbewältigung sollten in der Hypothese angegeben werden.

  Sie können für die Hypothese die folgende Vorlage verwenden (andere Formulierungen sind jedoch auch akzeptabel):

  > **Note**
  >
  > Wenn *(spezifischer Fehler)* auftritt, wird der *Workload* (Name des Workloads) *(Maßnahmen zur Bewältigung beschreiben),* um die Auswirkungen auf *geschäftliche oder technische Metriken einzudämmen*.

  Beispiel:

  - Wenn 20 % der Knoten in der Amazon EKS-Knotengruppe ausfallen, wird die Transaction Create API das 99. Perzentil der Anforderungen weiter in weniger als 100 ms erfüllen (Steady-State). Die Amazon EKS-Knoten werden innerhalb von fünf Minuten wiederhergestellt und die Pods werden geplant und verarbeiten Traffic innerhalb von acht Minuten nach der Einleitung des Experiments. Warnungen werden innerhalb von drei Minuten ausgelöst.
  - Wenn eine einzelne Amazon EC2-Instance ausfällt, veranlasst die Elastic Load Balancing-Zustandsprüfung des Bestellsystems Elastic Load Balancing, Anforderungen ausschließlich an die noch intakten Instances zu senden, während Amazon EC2 Auto Scaling die ausgefallene Instance ersetzt. Dabei kommt es zu einer Steigerung der serverseitigen Fehler (5xx) um weniger als 0,01 % (Steady-State).
  - Wenn die primäre Amazon RDS-Datenbank-Instance ausfällt, führt der Workload für die Erfassung von Lieferkettendaten einen Failover aus und stellt eine Verbindung zur Amazon RDS-Standby-Datenbank-Instance her, sodass es für weniger als 1 Minute zu Lese- oder Schreibfehlern für die Datenbank kommt (Steady-State).

- Führen Sie das Experiment aus, indem Sie den Fehler injizieren.

  Ein Experiment sollte grundsätzlich nicht zu einem Ausfall führen und vom Workload toleriert werden. Wenn Sie wissen, dass der Workload ausfallen wird, sollten Sie das Experiment

nicht durchführen. Das Chaos-Engineering sollte verwendet werden, um bekannt-unbekannte oder unbekannt-unbekannte Ereignisse zu untersuchen. Bekannt-unbekannte Ereignisse sind Ereignisse, die Ihnen bekannt sind, die Sie jedoch nicht vollständig verstehen. Unbekannt-unbekannte Ereignisse sind Ereignisse, die Sie weder kennen noch vollständig verstehen. Wenn Sie Experimente für einen Workload ausführen, von dem Sie wissen, dass er fehlerhaft ist, werden Sie keine neuen Erkenntnisse gewinnen. Ihr Experiment sollte sorgfältig geplant sein, einen klaren Wirkungsumfang besitzen und einen Rollback-Mechanismus besitzen, der bei unerwarteten Störungen angewendet werden kann. Wenn eine sorgfältige Überprüfung zeigt, dass Ihr Workload das Experiment überstehen sollte, können Sie das Experiment starten. Für die Injektion von Fehlern gibt es verschiedene Optionen. Für AWS-Workloads stellt AWS FIS zahlreiche vordefinierte Fehlersimulationen bereit, die als Aktionenbezeichnet werden. Sie können auch angepasste Aktionen für AWS FIS definieren, die mithilfe von AWS Systems Manager-Dokumenten ausgeführt werden.

Wir raten davon ab, angepasste Skripts für Chaos-Experimente zu verwenden, es sei denn, die Skripts können den aktuellen Zustand des Workloads erkennen, können Protokolle ausgeben und stellen Rollback-Mechanismen und Stoppbedingungen bereit, soweit möglich.

Ein effektives Framework oder Toolset, das Chaos-Engineering unterstützt, sollte den aktuellen Status des Experiments nachverfolgen, Protokolle ausgeben und Rollback-Mechanismen bereitstellen, um eine kontrollierte Ausführung zu unterstützen. Beginnen Sie mit einem verbreitet verwendeten Service wie AWS FIS, der Ihnen die Ausführung von Experimenten mit einem klar definierten Umfang ermöglicht und Sicherheitsmechanismen bereitstellt, um ein Experiment rückgängig machen zu können, wenn es zu unerwarteten Störungen führt. Weitere Informationen zu Experimenten unter Verwendung von AWS FIS finden Sie im Resilient and Well-Architected Apps with Chaos Engineering Lab. Darüber hinaus analysiert AWS Resilience Hub Ihren Workload und erstellt Experimente, die Sie in AWS FIS implementieren und ausführen können.

> **Note**
>
> Sie sollten den Umfang und die Auswirkungen jedes Experiments genau verstehen. Wir empfehlen, Fehler zunächst in einer Nichtproduktionsumgebung zu simulieren, bevor sie in der Produktion ausgeführt werden.

Experimente sollten in der Produktion unter realen Bedingungen ausgeführt werden. Dabei sollten nach Möglichkeit Canary-Bereitstellungen verwendet werden, die sowohl ein

Kontrollsystem als auch ein Experimentsystem bereitstellen. Die Ausführung von Experimenten außerhalb von Spitzenzeiten stellt ein empfehlenswertes Verfahren dar, um potenzielle Auswirkungen zu reduzieren, wenn ein Experiment zum ersten Mal in der Produktion durchgeführt wird. Wenn die Verwendung von tatsächlichem Kunden-Traffic ein zu großes Risiko darstellt, können Sie unter Verwendung der Kontroll- und Experimentbereitstellungen Experimente mit synthetischem Traffic in der Produktionsinfrastruktur durchführen. Wenn ein Experiment nicht in der Produktion ausgeführt werden kann, führen Sie es in einer Präproduktionsumgebung aus, die der Produktionsumgebung so nahe wie möglich ist.

Sie müssen einen Integritätsschutz einrichten und überwachen, um sicherzustellen, dass sich das Experiment nicht jenseits akzeptabler Grenzen auf den Produktions-Traffic oder andere Systeme auswirkt. Richten Sie Stoppbedingungen ein, um ein Experiment anhalten zu können, wenn es in einer Integritätsschutz-Metrik einen von Ihnen definierten Schwellenwert erreicht. Diese Metriken sollten die Metrik für den Steady-State des Workloads und die Metrik für die Komponenten einschließen, in die Sie den Fehler injizieren. Die synthetische Überwachung (auch als Benutzer-Canary bezeichnet) gehört zu den Metriken, die Sie in der Regel als Benutzer-Proxy einschließen sollten. Stoppbedingungen für AWS FIS werden als Teil der Experimentvorlage unterstützt. Es sind bis zu fünf Stoppbedingungen pro Vorlage möglich.

Zu den Grundsätzen des Chaos-Engineering gehört die Minimierung von Umfang und Auswirkungen des Experiments:

Auch wenn einige kurzfristige negative Auswirkungen zulässig sein sollten, ist der Chaos-Engineer dafür verantwortlich, die Auswirkungen der Experimente zu minimieren und einzudämmen.

Eine Methode für die Überprüfung des Umfangs und der möglichen Auswirkungen besteht darin, das Experiment statt in der Produktionsumgebung zunächst in einer Nichtproduktionsumgebung durchzuführen. Dabei wird überprüft, ob die Schwellenwerte für Stoppbedingungen während des Experiments wie vorgesehen aktiviert werden und ob das Experiment beobachtet werden kann, um Ausnahmen abzufangen.

Wenn Sie Fehlerinjektionsexperimente durchführen, müssen alle verantwortlichen Beteiligten gut informiert sein. Teilen Sie den betroffenen Teams mit, wann die Experimente durchgeführt werden und was zu erwarten ist. Dies können Operations-Teams, die für die Servicezuverlässigkeit verantwortlichen Teams und der Kundensupport sein. Stellen Sie diesen Teams Kommunikationstools bereit, damit sie das Team, das das Experiment durchführt, über nachteilige Auswirkungen informieren können.

Sie müssen nach dem Experiment den Workload und die zugrunde liegenden Systeme wieder in den ursprünglichen, gut funktionierenden Zustand zurückversetzen. Häufig führt das resiliente Design des betreffenden Workloads eine Selbstreparatur durch. Einige Fehlerdesigns oder fehlgeschlagenen Experimente können Ihren Workload jedoch in einem nicht erwarteten Fehlerzustand zurücklassen. Nach dem Ende des Experiments müssen Sie dies erkennen und den Workload und die Systeme wiederherstellen können. Mit AWS FIS können Sie eine Rollback-Konfiguration innerhalb der Aktionsparameter einrichten (auch als „Post-Aktion“ bezeichnet). Eine Post-Aktion führt das Ziel in den Zustand zurück, in dem es sich vor Ausführung der Aktion befunden hat. Ob automatisiert (bei Verwendung von AWS FIS) oder manuell – diese Post-Aktionen sollten Teil eines Playbooks sein, das die Erkennung und Behandlung von Fehlern und Ausfällen beschreibt.

- Prüfen Sie die Hypothese.

  Grundlagen des Chaos-Engineering stellt die folgende Anleitung für die Verifizierung des Steady-State Ihres Workloads bereit:

  Konzentrieren Sie sich auf die messbare Ausgabe des Systems und nicht auf die internen Attribute des Systems. Messungen dieser Ausgabe über einen kurzen Zeitraum stellen einen Proxy für den Steady-State des Systems dar. Der Gesamtdurchsatz, die Fehlerraten und die Latenz-Perzentile des Systems könnten Metriken sein, die das Steady-State-Verhalten beschreiben. Durch die Konzentration auf die Verhaltensmuster des Systems während Experimenten überprüft das Chaos-Engineering, ob das System funktioniert, statt zu versuchen, die Art der Funktion zu validieren.

  In unseren beiden Beispielen oben verwenden wir die Steady-State-Metrik einer Erhöhung von weniger als 0,01 % bei serverseitigen Fehlern (5xx) und von weniger als einer Minute, in der Datenbankschreib- und Lesefehler auftreten.

  Die 5xx-Fehler stellen eine gute Metrik dar, da sie die Folge des Fehlermodus sind, dem ein Client des Workloads direkt unterliegen wird. Die Messung der Datenbankfehler ist als direkte Folge des Fehlers gut als Metrik geeignet, sollte jedoch durch eine Messung der Client-Auswirkungen ergänzt werden, beispielsweise in Form von fehlgeschlagenen Kundenanfragen oder Fehlern im Client. Zusätzlich sollten Sie für alle APIs oder URIs, auf die der Client Ihres Workloads direkt zugreift, eine synthetische Überwachung einrichten (auch als Benutzer-Canary bezeichnet).

- Verbessern Sie das Workload-Design hinsichtlich der Resilienz.

Wenn der Steady-State nicht bewahrt wurde, untersuchen Sie, wie das Workload-Design verbessert werden könnte, um den Fehler zu bewältigen. Wenden Sie dabei die Best Practices der AWS Well-Architected-Säule „Zuverlässigkeit“ an. Zusätzliche Anleitungen und Ressourcen finden Sie in der AWS Builder’s Library.Diese Bibliothek enthält Artikel zur Verbesserung von Zustandsprüfungen oder zur Nutzung von Wiederholungen mit Backoff im Anwendungscodeund mehr.

Führen Sie das Experiment nach der Implementierung dieser Änderungen erneut durch (angezeigt durch die gepunktete Linie im Flywheel für das Chaos-Engineering), um ihre Effektivität zu ermitteln. Wenn der Verifizierungsschritt zeigt, dass die Hypothese zutrifft, befindet sich der Workload im Steady-State und der Zyklus wird fortgesetzt.

- Führen Sie regelmäßig Experimente durch.

  Ein Chaos-Experiment ist ein Zyklus. Daher sollten Experimente regelmäßig als Teil des Chaos-Engineering durchgeführt werden. Wenn die Hypothese eines Experiments auf einen Workload zutrifft, sollte das Experiment automatisiert werden, um innerhalb Ihrer CI/CD-Pipeline kontinuierlich als Regression ausgeführt zu werden. Informationen hierzu finden Sie in diesem Blog, der die Ausführung von AWS FIS-Experimenten mit AWS CodePipelinebeschreibt. Dieses Lab für wiederholte AWS FIS-Experimente in einer CI/CD-Pipeline ermöglicht Ihnen de Sammlung praktischer Erfahrungen.

  Fehlerinjektionsexperimente sind auch Bestandteil von Gamedays (siehe REL12-BP06 Regelmäßiges Abhalten von Gamedays). Bei Gamedays wird ein Fehler oder Ereignis simuliert, um Systeme, Prozesse und die Reaktionen von Teams zu testen. Dabei sollen die auszuführenden Aktionen vom Team wie im Fall eines außergewöhnlichen Ereignisses tatsächlich ausgeführt werden.

- Erfassen und speichern Sie die Ergebnisse der Experimente.

  Die Ergebnisse von Fehlerinjektionsexperimenten müssen erfasst und gespeichert werden. Erfassen Sie dabei alle notwendigen Daten (wie Zeit, Workload und Bedingungen), um die Ergebnisse und Trends von Experimenten später analysieren zu können. Beispiele für erfasste Ergebnisse können Screenshots von Dashboards, CSV-Versionen der Metrikdatenbank oder manuell eingegebene Aufzeichnungen von Ereignissen und Beobachtungen während des Experiments sein. Die Protokollierung von Experimenten mit AWS FIS kann Bestandteil dieser Datenerfassung sein.

Ressourcen

Zugehörige bewährte Methoden:

- REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung
- REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung:

Zugehörige Dokumente:

- Was ist AWS Fault Injection Service?
- Was ist AWS Resilience Hub?
- Grundlagen des Chaos-Engineering
- Chaos-Engineering: Planung Ihres ersten Experiments
- Resilience Engineering: Aus Fehlern lernen
- Chaos-Engineering-Geschichten
- Vermeiden von Fallback in verteilten Systemen
- Canary-Bereitstellung für Chaos-Experimente

Zugehörige Videos:

- AWS re:Invent 2020: Testing resiliency using chaos engineering (ARC316)
- AWS re:Invent 2019: Improving resiliency with chaos engineering (DOP309-R1)
- AWS re:Invent 2019: Performing chaos engineering in a serverless world (CMY301)

Zugehörige Beispiele:

- Well-Architected Lab: Level 300: Testen auf Resilienz von Amazon EC2, Amazon RDS und Amazon S3
- Chaos Engineering in AWS (Lab)
- Resilient and Well-Architected Apps with Chaos Engineering Lab
- Serverless-Chaos (Lab)
- Messen und Verbessern der Resilienz Ihrer Anwendung mit AWS Resilience Hub (Lab)

Zugehörige Tools:

- AWS Fault Injection Service
- AWS Marketplace: Gremlin Chaos Engineering Platform
- Chaos Toolkit
- Chaos Mesh
- Litmus

REL12-BP06 Regelmäßiges Abhalten von Gamedays

Nutzen Sie Gamedays, um Ihre Verfahren für Reaktionen auf Ereignisse und Fehler unter möglichst produktionsnahen Bedingungen (einschließlich Produktionsumgebungen) regelmäßig mit den Personen zu testen, die auch in tatsächlichen Fehlerszenarien beteiligt sind. Bei Gamedays werden Vorkehrungen getroffen, die sicherstellen, das sich Produktionsereignisse nicht auf Benutzer auswirken.

Bei Gamedays wird ein Fehler oder Ereignis simuliert, um Systeme, Prozesse und die Reaktion von Teams zu testen. Dabei sollen die auszuführenden Aktionen vom Team wie im Fall eines außergewöhnlichen Ereignisses tatsächlich ausgeführt werden. So können Sie nachvollziehen, wo nachgebessert werden kann. Zudem üben Sie dabei ein, wie Ihre Organisation mit Ereignissen umgeht. Gamedays sollten regelmäßig ausgeführt werden, damit die Reaktion für Ihr Team zu einem Reflex wird.

Nachdem Sie Ihre Maßnahmen für Ausfallsicherheit implementiert und in Umgebungen abseits der Produktion getestet haben, können Sie an einem Gameday feststellen, ob in der Produktion alles wie geplant funktioniert. An einem Gameday, insbesondere am ersten, werden alle Entwickler und Betriebsteams miteinbezogen und über Zeitpunkt sowie Ablauf des Tests informiert. Die Runbooks müssen vorhanden sein. Simulierte Ereignisse, auch potenzielle Ausfallereignisse, werden wie vorgeschrieben in den Produktionssystemen ausgeführt und deren Auswirkungen werden bewertet. Wenn alle Systeme wie vorgesehen funktionieren, erfolgen Erkennung und Selbstreparatur mit minimalen oder gar keinen Auswirkungen. Wenn jedoch negative Auswirkungen festgestellt werden, wird ein Rollback des Tests durchgeführt und die Workload-Probleme werden bei Bedarf manuell behoben (gemäß Runbook). Da Gamedays oft in der Produktion stattfinden, sollten alle Vorkehrungen getroffen werden, um Kunden vor Beeinträchtigungen der Verfügbarkeit zu schützen.

Gängige Antimuster:

- Die eigenen Verfahren werden dokumentiert, jedoch nie trainiert.
- Entscheidungsträger werden bei den Tests außen vorgelassen.

Vorteile der Einführung dieser Best Practice: Die regelmäßige Durchführung von Gamedays sorgt dafür, dass bei einem tatsächlichen Vorfall alle Mitarbeiter die Richtlinien und Verfahren befolgen. Außerdem wird überprüft, ob diese Richtlinien und Verfahren geeignet sind.

Risikostufe, falls diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Planen Sie Gamedays, um Ihre Runbooks und Playbooks regelmäßig zu trainieren. An Gamedays sollten alle Mitarbeiter beteiligt werden, die von Produktionsunterbrechungen betroffen sein können: Geschäftsinhaber, Entwickler, Produktionsmitarbeiter und die Teams, die auf Vorfälle reagieren.
  - Führen Sie Ihre Last- oder Leistungstests durch und schleusen Sie anschließend Fehler ein.
  - Prüfen Sie die Runbooks auf Anomalien und suchen Sie nach Möglichkeiten zur Ausführung der Playbooks.
    - Optimieren Sie bei Abweichungen die Runbooks oder ändern Sie das Verhalten. Ermitteln Sie bei Ausführung eines Playbooks das Runbook, das hätte verwendet werden sollen, oder erstellen Sie ein neues.

Ressourcen

Zugehörige Dokumente:

- Was ist AWS GameDay?

Zugehörige Videos:

- AWS re:Invent 2019: Verbesserung der Ausfallsicherheit mit Chaos-Engineering (DOP309-R1)

  Zugehörige Beispiele:

- AWS Well-Architected Labs: Testen der Ausfallsicherheit

REL 13. Was ist bei der Planung der Notfallwiederherstellung zu beachten?

Backups und redundante Workload-Komponenten sind der Ausgangspunkt Ihrer Strategie für die Notfallwiederherstellung. RTO und RPO sind Ihre Ziele für die Wiederherstellung Ihres Workloads. Legen Sie diese entsprechend den geschäftlichen Anforderungen fest. Implementieren Sie eine

Strategie, um diese Ziele zu erreichen. Berücksichtigen Sie dabei Standorte und Funktionen von Workload-Ressourcen und -Daten. Die Wahrscheinlichkeit von Disruptionen und die Kosten von Wiederherstellungen sind ebenfalls wichtige Faktoren bei der Ermittlung des Unternehmenswerts, den Notfallwiederherstellungen von Workloads bieten.

Bewährte Methoden

- REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:
- REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen
- REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung:
- REL13-BP04 Verwalten der Konfigurationsabweichungen am Standort oder in der Region der Notfallwiederherstellung:
- REL13-BP05: Automatisieren der Wiederherstellung

REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:

Für die Workload gelten ein Recovery Time Objective (RTO, Wiederherstellungsdauer) und ein Recovery Point Objective (RPO, Wiederherstellungszeitpunkt).

Die Wiederherstellungsdauer ist die maximal akzeptable Verzögerung zwischen der Unterbrechung und der Wiederherstellung des Service. Damit wird festgelegt, was als akzeptables Zeitfenster gilt, wenn der Service nicht verfügbar ist.

Der Wiederherstellungszeitpunkt ist die maximal zulässige Zeitspanne seit dem letzten Wiederherstellungspunkt. Damit wird festgelegt, was als akzeptabler Datenverlust zwischen dem letzten Wiederherstellungspunkt und der Service-Unterbrechung gilt.

RTO- und RPO-Werte sind wichtige Überlegungen bei der Auswahl einer geeigneten Notfallwiederherstellungsstrategie (Disaster Recovery, DR) für Ihre Workload. Diese Ziele werden vom Unternehmen festgelegt und dann von den technischen Teams zur Auswahl und Umsetzung einer DR-Strategie verwendet.

Gewünschtes Ergebnis:

Jeder Workload sind ein RTO und ein RPO zugewiesen, die auf der Grundlage der geschäftlichen Auswirkungen definiert werden. Die Workload wird einer vordefinierten Stufe zugewiesen, die die Serviceverfügbarkeit und den akzeptablen Datenverlust mit einem entsprechenden RTO und

RPO definiert. Wenn eine solche Einstufung nicht möglich ist, kann die Zuweisung individuell pro Workload erfolgen, mit der Absicht, zu einem späteren Zeitpunkt Stufen zu erstellen. RTO und RPO werden als eine der Hauptüberlegungen für die Auswahl einer Notfallwiederherstellungsstrategie für die Workload verwendet. Weitere Überlegungen bei der Auswahl einer DR-Strategie sind Kostenbeschränkungen, Abhängigkeiten von der Workload und betriebliche Anforderungen.

Bei der RTO sind die Auswirkungen anhand der Dauer eines Ausfalls zu verstehen. Ist sie linear oder gibt es nichtlineare Auswirkungen? (Beispiel: Nach vier Stunden wird eine Fertigungsstraße bis zum Beginn der nächsten Schicht stillgelegt.)

Eine Matrix der Notfallwiederherstellung wie die folgende kann Ihnen helfen zu verstehen, wie die Kritikalität der Workload mit den Wiederherstellungszielen zusammenhängt. (Beachten Sie, dass die tatsächlichen Werte für die X- und Y-Achsen an die Bedürfnisse Ihres Unternehmens angepasst werden sollten.)

Matrix der Notfallwiederherstellung						
		Wiederherstellungszeitpunkt				
		< 1 Minute	< 1 Stunde	< 6 Stunden	< 1 Tag	+ 1 Tag
Wiederher-stellungsdauer	< 10 Minuten	Kritisch	Kritisch	Hoch	Mittel	Mittel
	< 2 Stunden	Kritisch	Hoch	Mittel	Mittel	Niedrig
	< 8 Stunden	Hoch	Mittel	Mittel	Niedrig	Niedrig
	< 24 Stunden	Mittel	Mittel	Niedrig	Niedrig	Niedrig
	24 + Stunden	Mittel	Niedrig	Niedrig	Niedrig	Niedrig

Abbildung 16: Matrix der Notfallwiederherstellung

Gängige Antimuster:

- Keine definierten Wiederherstellungsziele.
- Auswählen beliebiger Wiederherstellungsziele.
- Auswählen von Wiederherstellungszielen, die zu lasch sind und die Geschäftsziele nicht erfüllen.
- Kein Verständnis des Auswirkung von Ausfallzeiten und Datenverlust.
- Auswahl unrealistischer Wiederherstellungsziele, wie z. B. Null-Zeit bis zur Wiederherstellung und Null-Datenverlust, die für Ihre Workload-Konfiguration möglicherweise nicht erreicht werden können.

- Auswählen von Wiederherstellungszielen, die strikter sind als die tatsächlichen Geschäftsziele. Dies erzwingt Implementierungen für die Notfallwiederherstellung, die kostspieliger und komplizierter sind als die Anforderungen der Workload.
- Auswahl von Wiederherstellungszielen, die mit denen einer abhängigen Workloads unvereinbar sind.
- Ihre Wiederherstellungsziele berücksichtigen nicht die Einhaltung gesetzlicher Vorschriften.
- RTO und RPO sind für eine Workload definiert, aber nie getestet.

Vorteile der Einführung dieser bewährten Methode: Die Wiederherstellungsziele für Dauer und Datenverlust sind als Orientierungshilfe für die Implementierung der Notfallwiederherstellung erforderlich.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Bei der gegebenen Workload müssen Sie die Auswirkungen von Ausfallzeiten und Datenverlusten auf Ihr Unternehmen verstehen. Die Auswirkungen werden in der Regel mit zunehmender Ausfallzeit oder Datenverlust größer, aber die Form dieses Anstiegs kann je nach Art der Workload unterschiedlich sein. So können Sie z. B. Ausfallzeiten bis zu einer Stunde ohne größere Beeinträchtigung tolerieren, danach steigen die Auswirkungen jedoch schnell an. Die Auswirkungen auf das Unternehmen zeigen sich in vielen Formen, darunter monetäre Kosten (z. B. entgangene Einnahmen), Kundenvertrauen (und Auswirkungen auf den Ruf), betriebliche Probleme (z. B. fehlende Gehaltsabrechnungen oder verringerte Produktivität) und gesetzliche Risiken. Führen Sie die folgenden Schritte aus, um diese Auswirkungen zu verstehen und RTO und RPO für Ihre Workload festzulegen.

Implementierungsschritte

1. Bestimmen Sie die Interessengruppen Ihres Unternehmens für diese Workload und arbeiten Sie mit ihnen zusammen, um diese Schritte umzusetzen. Die Wiederherstellungsziele für eine Workload sind eine geschäftliche Entscheidung. Die technischen Teams arbeiten dann mit den Business-Stakeholdern zusammen, um anhand dieser Ziele eine DR-Strategie auszuwählen.

> **Note**
>
> Für die Schritte 2 und 3 können Sie Folgendes verwenden: the section called "Implementierungsarbeitsblatt".

2. Sammeln Sie die notwendigen Informationen, um eine Entscheidung zu treffen, indem Sie die folgenden Fragen beantworten.
3. Gibt es in Ihrem Unternehmen Kategorien oder Stufen der Kritikalität für die Auswirkungen von Workloads?
   a. Falls zutreffend, ordnen Sie diese Workload einer Kategorie zu.
   b. Falls nicht zutreffend, richten Sie diese Kategorien ein. Legen Sie fünf oder weniger Kategorien fest und verfeinern Sie die Spanne der angestrebten Wiederherstellungszeit für jede Kategorie. Zu den Beispielkategorien gehören: kritisch, hoch, mittel, niedrig. Um zu verstehen, wie sich Workloads den Kategorien zuordnen lassen, sollten Sie prüfen, ob die Workload unternehmenskritisch, geschäftswichtig oder nicht geschäftsrelevant ist.
   c. Legen Sie RTO und RPO für die Workload je nach Kategorie fest. Wählen Sie immer eine Kategorie, die strikter ist (niedrigere RTO- und RPO-Werte) als die bei der Eingabe dieses Schritts berechneten Rohwerte. Wenn dies zu einer unangemessen großen Veränderung des Wertes führt, sollten Sie eine neue Kategorie anlegen.
4. Weisen Sie auf der Grundlage dieser Antworten der Workload RTO- und RPO-Werte zu. Dies kann direkt geschehen oder durch Zuweisung der Workload zu einer vordefinierten Serviceebene.
5. Dokumentieren Sie den Notfallwiederherstellungsplan (Disaster Recovery Plan, DRP) für diese Workload, der Teil der Unternehmensstrategie ist. Betriebskontinuitätsplan (BCP)an einem Ort, der für das Workload-Team und die Stakeholder zugänglich ist
   a. Halten Sie die RTO- und RPO-Werte sowie die zur Ermittlung dieser Werte verwendeten Informationen fest. Geben Sie eine Strategie zur Bewertung der Auswirkungen der Workload auf das Unternehmen an.
   b. Erfassen Sie neben RTO und RPO auch andere Metriken, die Sie für Notfallwiederherstellungsziele verfolgen oder zu verfolgen planen
   c. Sie fügen diesem Plan Details zu Ihrer DR-Strategie und Ihrem Runbook hinzu, wenn Sie diese erstellen.
6. Indem Sie die Kritikalität der Workload in einer Matrix wie der in Abbildung 15 nachschlagen, können Sie damit beginnen, vordefinierte Serviceebenen für Ihr Unternehmen festzulegen.

7. Nachdem Sie eine DR-Strategie (oder einen Machbarkeitsnachweis für eine DR-Strategie) gemäß implementiert haben,the section called “REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen”testen Sie diese Strategie, um die tatsächliche RTC (Recovery Time Capability) und RPC (Recovery Point Capability) der Workload zu bestimmen. Wenn diese nicht den angestrebten Wiederherstellungszielen entsprechen, arbeiten Sie entweder mit Ihren Stakeholdern zusammen, um diese Ziele anzupassen, oder nehmen Sie Änderungen an der DR-Strategie vor, um die Zielvorgaben zu erreichen.

Primäre Fragen

1. Wie lange kann die Workload maximal ausfallen, bevor es zu schwerwiegenden Auswirkungen auf das Unternehmen kommt?
   a. Bestimmen Sie die monetären Kosten (direkte finanzielle Auswirkungen) für das Unternehmen pro Minute, wenn die Workload unterbrochen wird.
   b. Bedenken Sie, dass die Auswirkungen nicht immer linear sind. Die Auswirkungen können zunächst begrenzt sein und dann ab einem kritischen Zeitpunkt rasch zunehmen.
2. Wie groß ist die maximale Datenmenge, die verloren gehen kann, bevor es zu schwerwiegenden Auswirkungen auf das Unternehmen kommt?
   a. Berücksichtigen Sie diesen Wert für Ihren wichtigsten Datenspeicher. Identifizieren Sie die jeweilige Kritikalität für andere Datenspeicher.
   b. Können Workload-Daten bei Verlust wiederhergestellt werden? Wenn dies aus betrieblicher Sicht einfacher ist als Backup und Wiederherstellung, dann wählen Sie das RPO auf der Grundlage der Kritikalität der Ursprungsdaten, die zur Wiederherstellung der Workload-Daten verwendet werden.
3. Wie lauten die Wiederherstellungsziele und Verfügbarkeitserwartungen von Workloads, von denen dieser abhängt (Downstream), oder von Workloads, die von diesem abhängen (Upstream)?
   a. Wählen Sie Wiederherstellungsziele, die es dieser Workload ermöglichen, die Anforderungen der vorgelagerten Abhängigkeiten zu erfüllen
   b. Wählen Sie Wiederherstellungsziele, die angesichts der Wiederherstellungsmöglichkeiten der nachgelagerten Abhängigkeiten erreichbar sind. Unkritische nachgelagerte Abhängigkeiten (die Sie „umgehen“ können) können ausgeschlossen werden. Oder arbeiten Sie mit kritischen, nachgelagerten Abhängigkeiten zusammen, um deren Wiederherstellungsmöglichkeiten zu verbessern.

Weitere Fragen

Überlegen Sie sich, wie diese Fragen auf diese Workload zutreffen könnten:

4. Haben Sie unterschiedliche RTO und RPO je nach Art des Ausfalls (Region vs. Region)? AZ, etc.)?
5. Gibt es einen bestimmten Zeitpunkt (Saisonabhängigkeit, Verkaufsveranstaltungen, Produkteinführungen), zu dem sich Ihr RTO/RPO ändern kann? Wenn ja, was ist die unterschiedliche Messung und die zeitliche Begrenzung?
6. Wie viele Kunden sind von einer Unterbrechung der Workload betroffen?
7. Welche Auswirkungen hat es auf den Ruf, wenn die Workload unterbrochen wird?
8. Welche anderen betrieblichen Auswirkungen können auftreten, wenn die Workload unterbrochen wird? Zum Beispiel Auswirkungen auf die Produktivität der Mitarbeiter, wenn die E-Mail-Systeme nicht verfügbar sind oder wenn die Lohnbuchhaltungssysteme keine Transaktionen übermitteln können.
9. Wie stimmen RTO und RPO der Workload mit der DR-Strategie der Geschäftsbereiche und des Unternehmens überein?
10. Gibt es interne vertragliche Verpflichtungen für die Erbringung einer Dienstleistung? Gibt es Strafen für die Nichteinhaltung dieser Vorgaben?
11. Welche rechtlichen oder Compliance-Bedingungen gelten für die Daten?

Implementierungsarbeitsblatt

Sie können dieses Arbeitsblatt für die Implementierungsschritte 2 und 3 verwenden. Sie können dieses Arbeitsblatt an Ihre speziellen Bedürfnisse anpassen, indem Sie beispielsweise zusätzliche Fragen hinzufügen.

Schritt 2: primäre Fragen	Gilt für Workload?	Workload-RTO	Workload-RPO	RTO anpassen	RPO anpassen	Anleitungen
[1] Maximale Zeit, in der der Workload ausfallen kann						Gemessen Zeit seit Beginn des Ausfalls bis zur Wiederherstellung
[2] Maximale Datenmenge, die verloren gehen kann						Gemessen in Zeit seit dem letzten bekannten gut wiederherstellbaren Datensatz
[3a] Vorgelagerte Abhängigkeiten						Strengste nachgelagerte Wiederherstellungsziele eingeben
[3b] Nachgelagerte Abhängigkeiten						Am wenigsten strenge nachgelagerte Wiederherstellungsziele eingeben
[3a] Abgeglichene vorgelagerte Abhängigkeiten						Wenn der vorgelagerte Wert niedriger ist als aktuelle Werte und der nachgelagerte Wert größer ist,
[3b] Abgeglichene nachgelagerte Abhängigkeiten						arbeiten Sie mit Abhängigkeiten, um auszugleichen und hier ausgeglichene Werte einzugeben.
[3] Abhängigkeiten						Werte senken, um vorgelagerte Abhängigkeiten zu erfüllen oder die basierend auf nachgelagerten Abhängigkeitsfähigkeiten zu erhöhen
Schritt 2: zusätzliche Fragen						Geben Sie an, ob die Frage zutrifft. Falls nicht, überspringen Sie sie.
Basis-RTO/-RPO						Übertragen Sie die RTO- und RPO-Werte von oben nach hier unten.
[4] Art des Ausfalls	[ ]J/[ ]N					Geben Sie Wiederherstellungsziele für Ereignisarten mit strengsten Anforderungen ein.
[5] Spezifische zeitbasierte Ziele	[ ]J/[ ]N					Geben Sie Wiederherstellungsziele für Zeiten mit strengsten Anforderungen ein.
[6] Unterbrechungen bei Kunden	[ ]J/[ ]N					Grafische Darstellung der betroffenen Kunden in Abhängigkeit von der Ausfallzeit oder dem Datenverlust. Verwenden Sie dies, um das maximal zulässige RTO und RPO auf der Grundlage der Kundenauswirkungen einzugeben.
[7] Auswirkungen auf den Ruf	[ ]J/[ ]N					Mit dem Unternehmen arbeiten, um die maximale RTO und den maximalen RPO basierend auf der Auswirkung auf die Reputation zu bestimmen
[8] Betriebliche Auswirkungen	[ ]J/[ ]N					Geben Sie das maximale RTO und RPO auf der Grundlage der betrieblichen Auswirkungen ein.
[9] Organisatorische Ausrichtung	[ ]J/[ ]N					Geben Sie das maximale RTO und RPO für Workloads dieses Typs gemäß den LOB- und Organisationsanforderungen ein.
[10] Vertragliche Verpflichtungen	[ ]J/[ ]N					Geben Sie das maximale RTO und RPO auf der Grundlage der vertraglichen Verpflichtungen ein.
[11] Gesetzliche Vorschriften	[ ]J/[ ]N					Geben Sie das maximale RTO und RPO auf der Grundlage der geltenden gesetzlichen Bestimmungen ein.
Ziel basierend auf zusätzlichen Fragen						Nehmen Sie den Mindestwert (strengerer Wert) aus den Fragen 4–11 und geben Sie ihn hier ein.
Angepasstes Ziel						Wenn die Ziele in der obigen Zeile nicht erreicht werden können, arbeiten Sie mit den Beteiligten zusammen, um die Beschränkungen zu lockern, und geben Sie hier ein neues Minimum ein.
RTO/RPO angepasst						Geben Sie die Basis-RPO-/RTO-Werte oder das angepasste Ziel ein, je nachdem, welcher Wert niedriger ist.
Schritt 3						
Zuordnung zu vordefinierter Kategorie oder Stufe						Senken Sie beide Werte (machen Sie sie strenger), um sie an die nächstgelegene definierte Stufe anzupassen.

Arbeitsblatt

Grad des Aufwands für den Implementierungsplan: Niedrig

Ressourcen

Ähnliche bewährte Methoden:

- the section called “REL09-BP04 Verifizieren der Sicherungsintegrität und -verfahren durch regelmäßiges Wiederherstellen der Daten”
- the section called “REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen”
- the section called “REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung:”

Zugehörige Dokumente:

- AWS Architecture Blog: Notfallwiederherstellungsserie

- Notfallwiederherstellung von Workloads auf AWS: Wiederherstellung in der Cloud (AWS-Whitepaper)
- Verwalten von Ausfallsicherheit mit AWS Resilience Hub
- APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können
- AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte

Relevante Videos

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2) (Architekturmuster für Aktiv-Aktiv-Anwendungen in mehreren Regionen)
- Notfallwiederherstellung von Workloads auf AWS

REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen

Definieren Sie eine Notfallwiederherstellungsstrategie (Disaster Recovery, DR), die den Wiederherstellungszielen Ihrer Workloads entspricht. Wählen Sie eine Strategie aus, z. B. Backup und Wiederherstellung, Standby (aktiv/passiv) oder Aktiv/Aktiv.

Gewünschtes Ergebnis: Für jeden Workload gibt es eine definierte und implementierte Notfallwiederherstellungsstrategie, die dem Workload das Erreichen der Notfallwiederherstellungsziele ermöglicht. DR-Strategien zwischen Workloads nutzen wiederverwendbare Muster (wie die zuvor beschriebenen Strategien),

Typische Anti-Muster:

- Implementierung von inkonsistenten Wiederherstellungsprozeduren für Workloads mit ähnlichen DR-Zielen.
- Die DR-Strategie muss im Notfall Ad-hoc umgesetzt werden.
- Es gibt keinen Plan für die Notfallwiederherstellung.
- Abhängigkeit von Vorgängen auf der Steuerebene während der Wiederherstellung.

Vorteile der Nutzung dieser bewährten Methode:

- Durch die Nutzung definierter Wiederherstellungsstrategien können Sie verbreitet verwendete Tools und Testverfahren verwenden.

- Die Verwendung definierter Wiederherstellungsstrategien verbessert den Wissensaustausch zwischen den Teams und die Implementierung der Notfallwiederherstellung für ihre Workloads.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch. Ohne eine geplante, implementierte und getestete DR-Strategie ist es unwahrscheinlich, dass Sie Ihre Wiederherstellungsziele im Falle eines Notfalls erreichen.

Implementierungsleitfaden

Eine DR-Strategie beruht auf der Fähigkeit, Ihre Workload an einem Wiederherstellungsstandort bereitzustellen, wenn Ihr primärer Standort nicht mehr in der Lage ist, den Workload auszuführen. Die häufigsten Wiederherstellungsziele sind RTO und RPO, wie besprochen in REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:.

Eine DR-Strategie, die mehrere Availability Zones (AZs) innerhalb eines einzigen AWS-Region umfasst, kann Katastrophenereignisse wie Brände, Überschwemmungen und größere Stromausfälle abfedern. Wenn es erforderlich ist, einen Schutz gegen ein unwahrscheinliches Ereignis zu implementieren, das verhindert, dass Ihre Workload in einer bestimmten AWS-Region ausgeführt werden kann, können Sie eine DR-Strategie verwenden, die mehrere Regionen nutzt.

Wenn Sie eine DR-Strategie für mehrere Regionen entwickeln, sollten Sie eine der folgenden Strategien wählen. Sie werden nach zunehmenden Kosten und zunehmender Komplexität und abnehmender RTO und RPO aufgelistet. Die Wiederherstellungsregion verweist auf eine andere AWS-Region als die für Ihren Workload verwendete primäre Region.

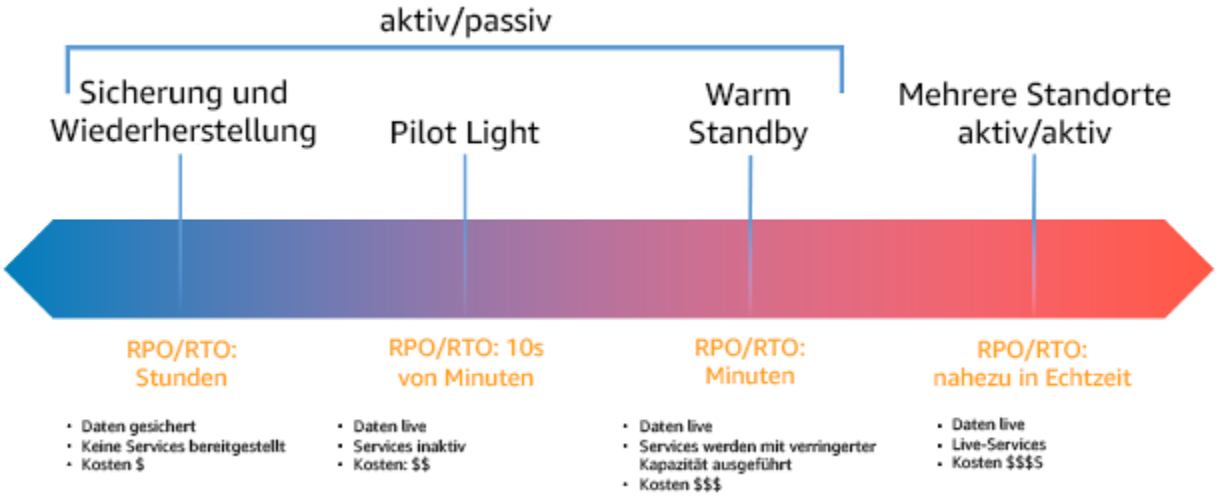


Abbildung 17: Notfallwiederherstellungsstrategien (DR)

- Backup und Wiederherstellung (RPO im Stundenbereich, RTO innerhalb von 24 Stunden oder weniger): Sichern Sie Ihre Daten und Anwendungen in der Wiederherstellungsregion. Die Verwendung automatisierter oder kontinuierlicher Backups ermöglicht eine zeitpunktgenaue Wiederherstellung, wodurch das RPO in einigen Fällen auf bis zu 5 Minuten gesenkt werden kann. Im Falle eines Notfalls stellen Sie Ihre Infrastruktur bereit (wobei Sie Infrastruktur als Code verwenden, um die RTO zu verkürzen), stellen Ihren Code bereit und stellen die gesicherten Daten wieder her, um eine Wiederherstellung nach einem Notfall in der Wiederherstellungsregion zu erfahren.
- Pilot-Light (RPO im Minutenbereich, RTO innerhalb von zehn Minuten): Bereitstellung einer Kopie Ihrer Core-Workload-Infrastruktur in der Wiederherstellungsregion. Replizieren Sie Ihre Daten in die Wiederherstellungsregion und erstellen Sie dort Sicherungskopien der Daten. Ressourcen, die zur Unterstützung der Datenreplikation und -sicherung erforderlich sind, wie Datenbanken und Objektspeicher, sind immer eingeschaltet. Andere Elemente wie Anwendungsserver oder Serverless Compute werden nicht bereitgestellt, sondern können bei Bedarf mit der erforderlichen Konfiguration und dem Anwendungscode erstellt werden.
- Warm-Standby (RPO im Sekundenbereich, RTO im Minutenbereich): Aufrechterhaltung einer herunterskalierten, aber voll funktionsfähigen Version Ihres Workloads, die immer in der Wiederherstellungsregion ausgeführt wird. Geschäftskritische Systeme sind vollständig dupliziert und ständig aktiv, aber mit herunterskalierter Infrastruktur. Die Daten werden repliziert und sind in der Wiederherstellungsregion live. Wenn eine Wiederherstellung erforderlich ist, wird das System zur Bewältigung der Produktionslast schnell hochskaliert. Je höher die Skalierung des Warm-Standby, desto geringer ist die Abhängigkeit von RTO und Steuerebene. Bei einer vollständigen Abdeckung spricht man von Hot-Standby.
- Multi-Region (Multi-Site) Aktiv/Aktiv (RPO nahe Null, RTO potenziell Null): Ihr Workload wird an mehreren AWS-Regionen-Standorten bereitgestellt und bedient aktiv den Datenverkehr von diesen. Bei dieser Strategie müssen Sie die Daten zwischen den Regionen synchronisieren. Mögliche Konflikte, die durch Schreibvorgänge auf denselben Datensatz in zwei verschiedenen regionalen Repliken verursacht werden, müssen vermieden oder behandelt werden, was sehr komplex sein kann. Die Datenreplikation ist nützlich für die Datensynchronisation und schützt Sie vor einigen Arten von Notfällen, aber sie schützt Sie nicht vor Datenbeschädigung oder -zerstörung, es sei denn, Ihre Lösung umfasst auch Optionen für eine zeitpunktgenaue Wiederherstellung.

> **Note**
>
> Der Unterschied zwischen Pilot-Light und Warm-Standby kann schwer zu überblicken sein. Beide beinhalten eine Umgebung in Ihrer Wiederherstellungsregion mit Kopien der Assets Ihrer Primärregion. Der Unterschied besteht darin, dass Pilot-Light keine Anfragen bearbeiten kann, ohne dass zuvor zusätzliche Maßnahmen ergriffen werden, während Warm-Standby den Datenverkehr (mit reduzierter Kapazität) sofort bearbeiten kann. Bei Pilot-Light müssen Sie die Server einschalten, möglicherweise zusätzliche (nicht zum Kerngeschäft gehörende) Infrastruktur bereitstellen und die Leistung hochskalieren, während Sie bei Warm-Standby nur die Leistung hochskalieren müssen (alles ist bereits bereitgestellt und läuft). Wählen Sie je nach RTO- und RPO-Anforderungen zwischen diesen Varianten.
> Wenn die Kosten eine Rolle spielen und Sie ähnliche RPO- und RTO-Ziele wie bei der Warm-Standby-Strategie erreichen möchten, könnten Sie cloud-native Lösungen wie AWS Elastic Disaster Recovery in Betracht ziehen, die den Pilot-Light-Ansatz verfolgen und bessere RPO- und RTO-Ziele bieten.

Implementierungsschritte

1. Bestimmen Sie eine DR-Strategie, die die Wiederherstellungsanforderungen für diese Workload erfüllt.

Die Wahl einer DR-Strategie ist eine Abwägung zwischen der Reduzierung von Ausfallzeiten und Datenverlusten (RTO und RPO) und den Kosten und der Komplexität der Implementierung der Strategie. Sie sollten vermeiden, eine Strategie zu verfolgen, die strikter ist als nötig, da dies unnötige Kosten verursacht.

Im folgenden Diagramm hat das Unternehmen beispielsweise seine maximal zulässige RTO sowie die Grenze der Ausgaben für seine Strategie zur Wiederherstellung von Diensten festgelegt. In Anbetracht der Ziele des Unternehmens erfüllen die DR-Strategien Pilot-Light oder Warm-Standby sowohl die RTO- als auch die Kostenkriterien.

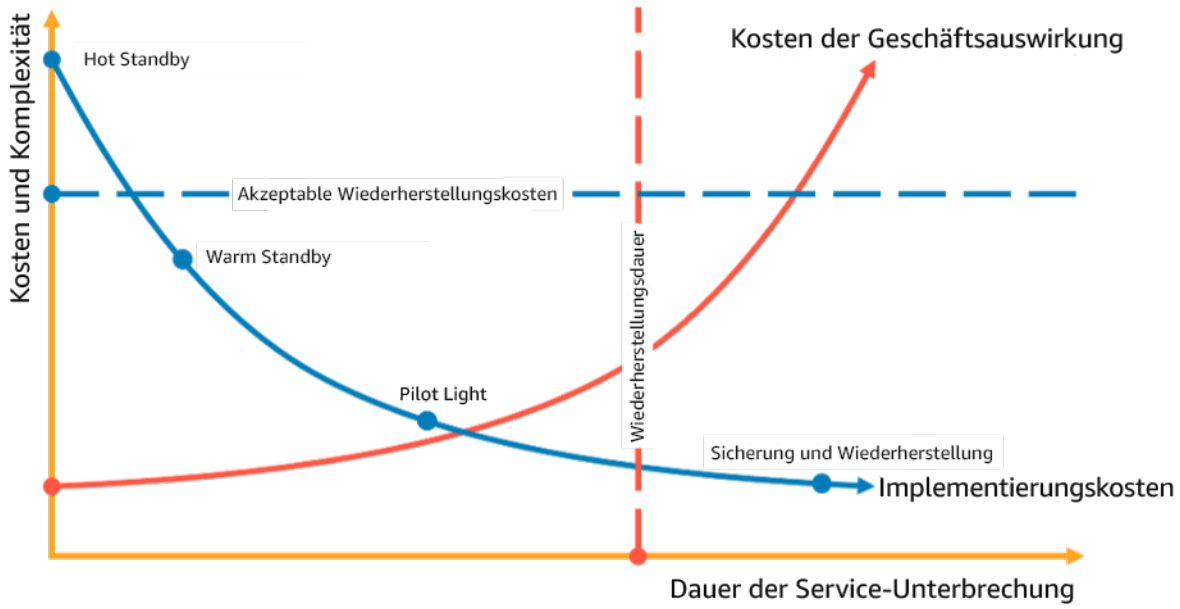


Abbildung 18: Auswahl einer DR-Strategie auf der Grundlage von RTO und Kosten

Weitere Informationen finden Sie unter Business Continuity Plan (BCP).

2. Überprüfen Sie die Muster, wie die ausgewählte DR-Strategie umgesetzt werden kann.

In diesem Schritt geht es darum, zu verstehen, wie Sie die gewählte Strategie umsetzen wollen. Die Strategien werden durch die Verwendung von AWS-Regionen als primäre und Wiederherstellungsstandort erläutert. Sie können jedoch auch Verfügbarkeitszonen innerhalb einer einzigen Region als DR-Strategie verwenden, die Elemente mehrerer dieser Strategien nutzt.

In den folgenden Schritten können Sie die Strategie auf Ihren spezifischen Workload anwenden.

Sicherung und Wiederherstellung

Backup und Wiederherstellung ist die am einfachsten zu implementierende Strategie, erfordert jedoch mehr Zeit und Aufwand für die Wiederherstellung des Workloads, was zu einem höheren RTO und RPO führt. Es ist eine gute Vorgehensweise, immer Sicherungskopien Ihrer Daten zu erstellen und diese auf einen anderen Standort (z. B. einen anderen AWS-Region) zu kopieren.

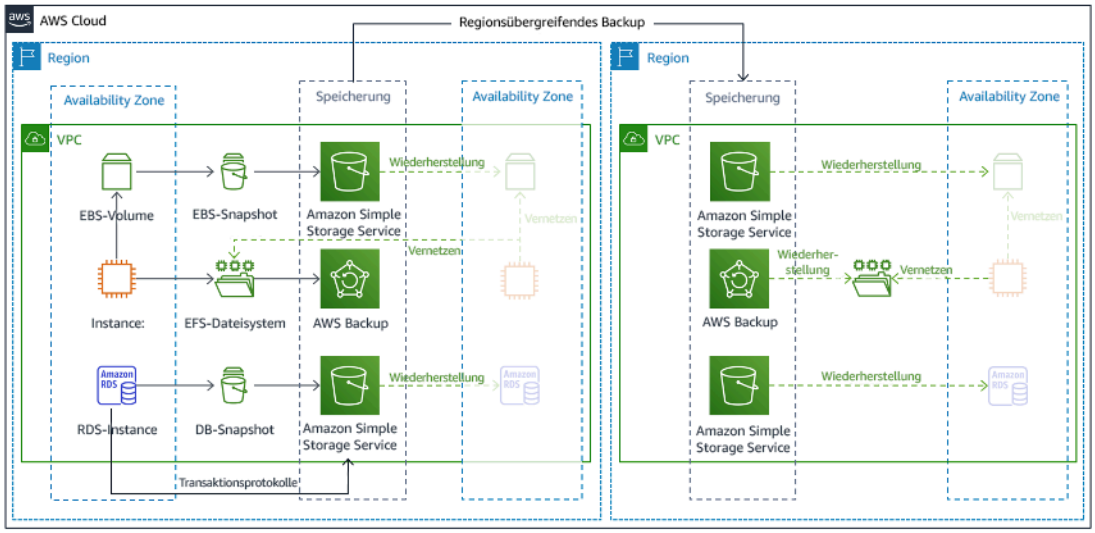


Abbildung 19: Sicherungs- und Wiederherstellungsarchitektur

Weitere Details zu dieser Strategie finden Sie unter Disaster Recovery (DR) Architecture on AWS, Part II: Backup and Restore with Rapid Recovery (Architektur zur Notfallwiederherstellung (DR) auf AWS, Teil II: Backup und Wiederherstellung mit schneller Wiederherstellung).

Pilot Light

Mit dem Pilot-Light-Ansatz replizieren Sie Ihre Daten von Ihrer primären Region auf Ihre Recovery Region. Die Kernressourcen, die für die Workload-Infrastruktur verwendet werden, werden in der Wiederherstellungsregion bereitgestellt, jedoch werden noch zusätzliche Ressourcen und Abhängigkeiten benötigt, um diesen Stack funktionsfähig zu machen. In Abbildung 20 werden zum Beispiel keine Compute-Instances bereitgestellt.

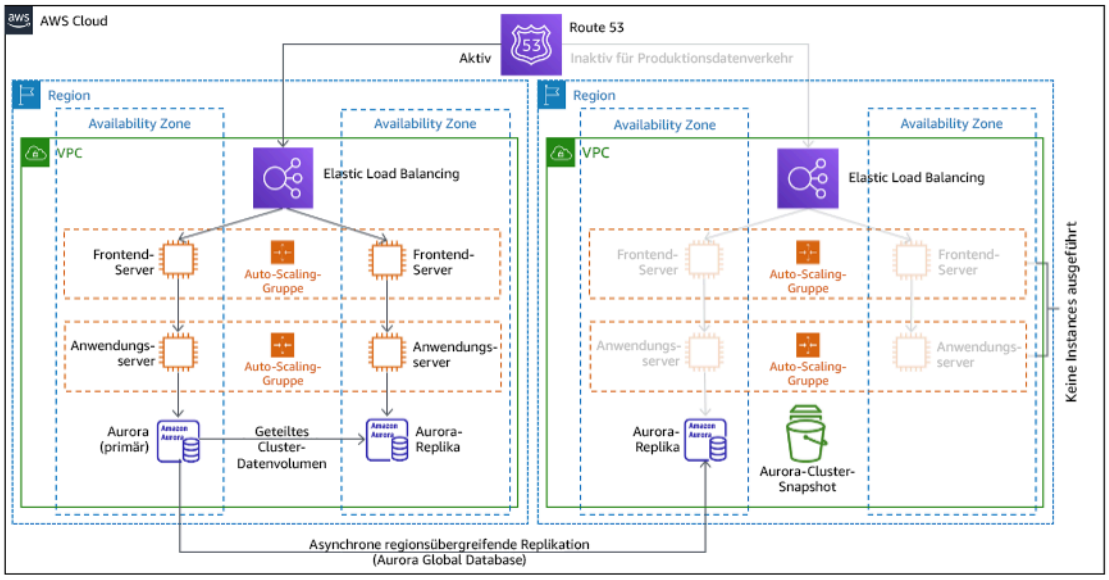


Abbildung 20: Pilot-Light-Architektur

Weitere Details zu dieser Strategie finden Sie unter Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby (Architektur zur Notfallwiederherstellung (DR) auf AWS, Teil III: Pilot-Light und Warm-Standby).

Warm Standby

Der Warm-Standby-Ansatz besteht darin, dass eine herunterskalierte, aber voll funktionsfähige Kopie Ihrer Produktionsumgebung in einer anderen Region vorhanden ist. Dieser Ansatz erweitert das Konzept des Pilot-Light und verkürzt die Zeit bis zur Wiederherstellung, da die Workload in einer anderen Region ständig präsent ist. Wenn die Wiederherstellungsregion mit voller Kapazität bereitgestellt wird, wird dies als Hot-Standby bezeichnet.

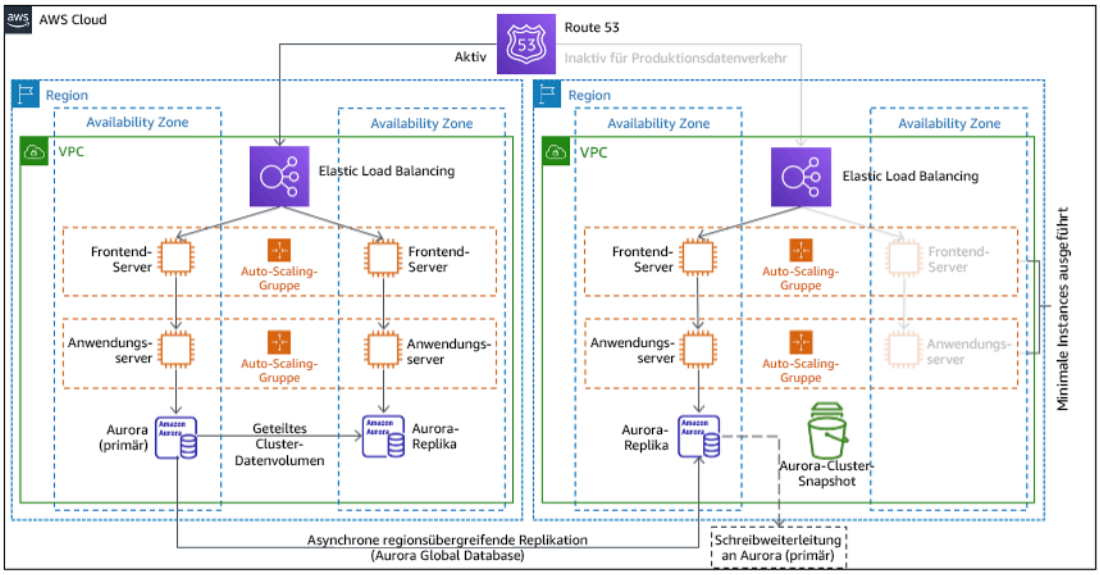


Abbildung 21: Warm-Standby-Architektur

Der Einsatz von Warm-Standby oder Pilot-Light erfordert ein Hochskalieren der Ressourcen in der Wiederherstellungsregion. Um sicherzustellen, dass Kapazität bei Bedarf verfügbar ist, sollten Sie die Verwendung von Kapazitätsreservierungen für EC2-Instances in Betracht ziehen. Wenn Sie AWS Lambda verwenden, können Sie mit provisioned concurrency Ausführungsumgebungen bereitstellen, damit diese sofort auf die Aufrufe Ihrer Funktion reagieren können.

Weitere Details zu dieser Strategie finden Sie unter Disaster Recovery (DR) Architecture on AWS, Part III: Pilot Light and Warm Standby (Architektur zur Notfallwiederherstellung (DR) auf AWS, Teil III: Pilot-Light und Warm-Standby).

Mehrere Standorte aktiv/aktiv

Sie können Ihren Workload gleichzeitig in mehreren Regionen als Teil einer Multi-Site Aktiv/Aktiv-Strategie ausführen. Multi-Site Aktiv/Aktiv bedient den Datenverkehr aus allen Regionen, in denen es eingesetzt wird. Kunden können diese Strategie aus anderen Gründen als DR wählen. Sie kann zur Erhöhung der Verfügbarkeit oder bei der Bereitstellung einer Workload für eine globale Zielgruppe verwendet werden (um den Endpunkt näher an die Benutzer zu bringen und/oder um Stacks bereitzustellen, die für die Zielgruppe in dieser Region lokalisiert sind). Wenn der Workload in einer der AWS-Regionen, in denen er bereitgestellt wird, nicht unterstützt werden kann, wird diese Region evakuiert und die verbleibenden Regionen werden zur Aufrechterhaltung der Verfügbarkeit genutzt. Multi-Site Aktiv/Aktiv ist die betrieblich komplexeste der DR-Strategien und sollte nur dann gewählt werden, wenn die Geschäftsanforderungen dies erfordern.

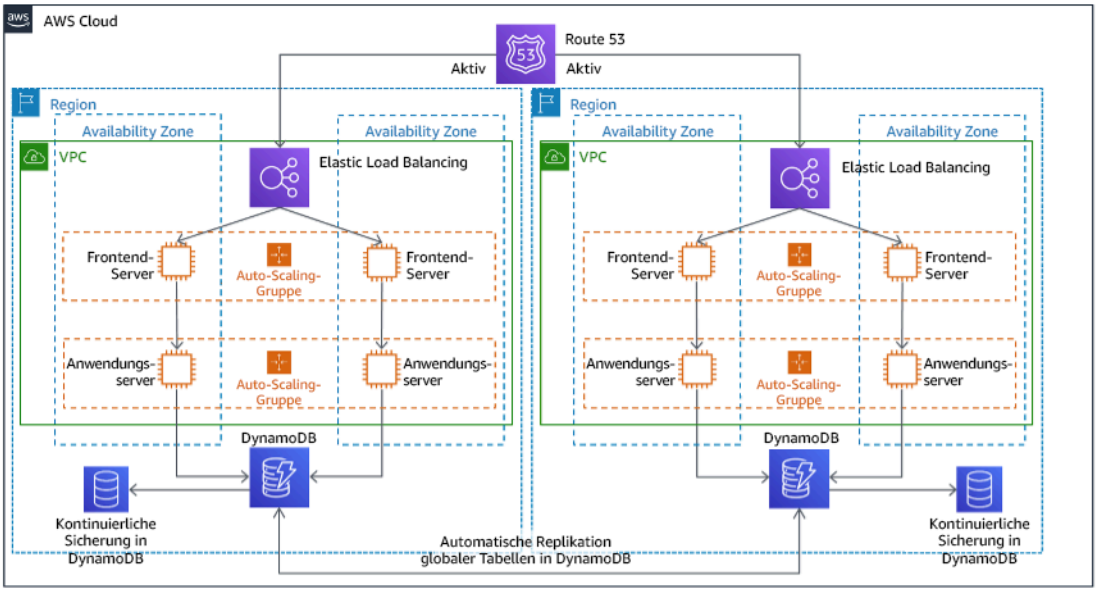


Abbildung 22: Multi-Site Aktiv/Aktiv Architektur

Weitere Details zu dieser Strategie finden Sie unter Disaster Recovery (DR) Architecture on AWS, Part IV: Multi-site Active/Active (Architektur zur Notfallwiederherstellung (DR) auf AWS, Teil IV: Multi-Site Aktiv/Aktiv).

AWS Elastic Disaster Recovery

Wenn Sie für die Notfallwiederherstellung die Pilot-Light- oder die Warm-Standby-Strategie in Betracht ziehen, könnte AWS Elastic Disaster Recovery einen alternativen Ansatz mit verbesserten Vorteilen bieten. Elastic Disaster Recovery kann ein ähnliches RPO- und RTO-Ziel wie Warm-Standby bieten, behält aber den kostengünstigen Ansatz von Pilot-Light bei. Elastic Disaster Recovery repliziert Ihre Daten von Ihrer primären Region auf Ihre Wiederherstellungsregion und nutzt dabei die kontinuierliche Datensicherung, um ein RPO im Sekundenbereich und ein RTO im Minutenbereich zu erreichen. In der Wiederherstellungsregion werden nur die für die Replikation der Daten erforderlichen Ressourcen bereitgestellt, was die Kosten ähnlich wie bei der Pilot-Light-Strategie niedrig hält. Bei Verwendung von Elastic Disaster Recovery koordiniert und orchestriert der Service die Wiederherstellung von Computing-Ressourcen, wenn diese als Teil eines Failover oder Drills initiiert wird.

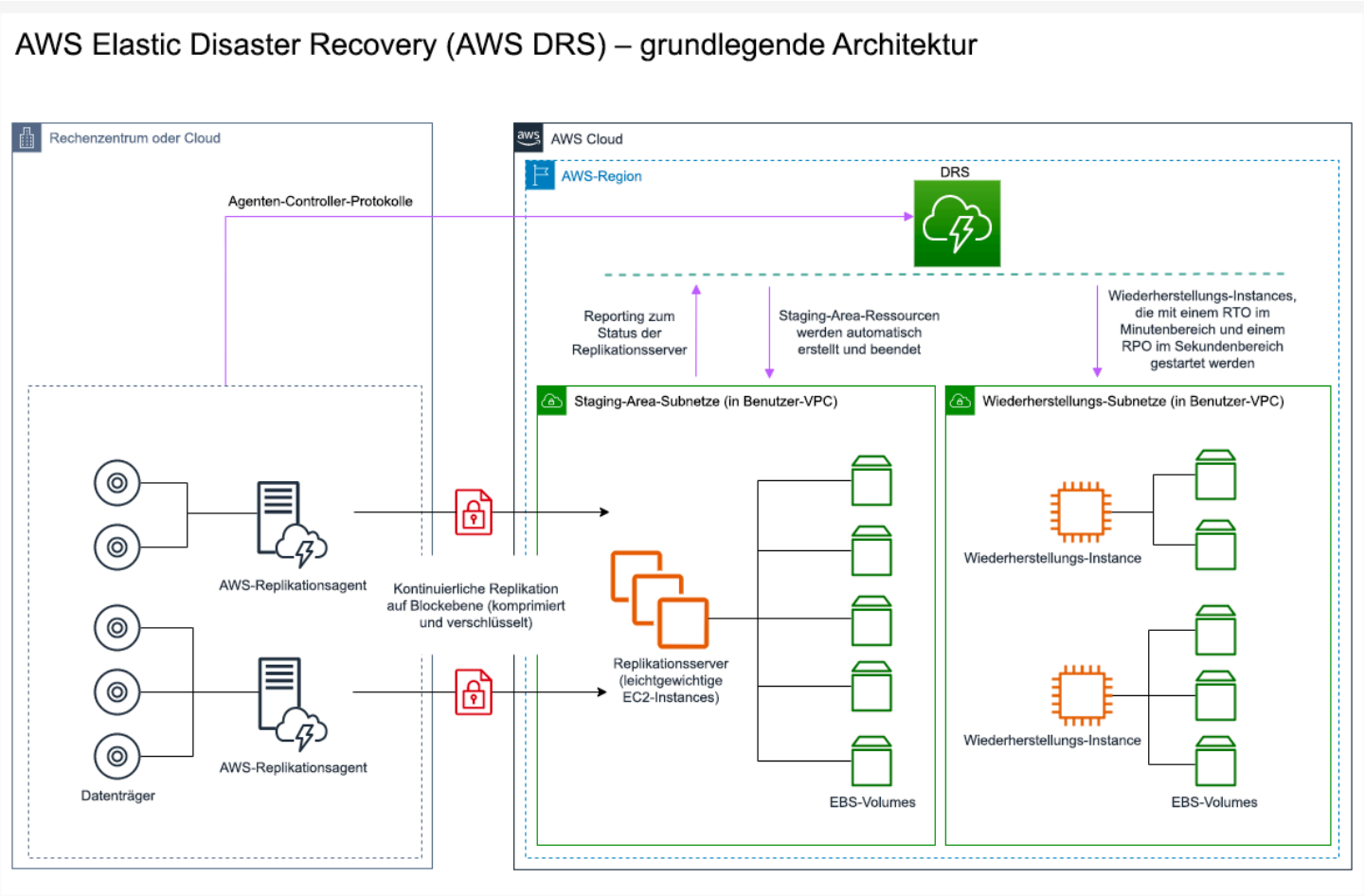


Abbildung 23: AWS Elastic Disaster Recovery-Architektur

Zusätzliche Praktiken zum Schutz von Daten

Bei allen Strategien müssen Sie sich auch gegen einen Datennotfall wappnen. Kontinuierliche Datenreplikation schützt Sie vor einigen Arten von Notfällen, aber sie schützt Sie möglicherweise nicht vor Datenbeschädigung oder -zerstörung, es sei denn, Ihre Strategie umfasst auch die Versionsverwaltung gespeicherter Daten oder Optionen für eine zeitpunktgenaue Wiederherstellung. Sie müssen auch die replizierten Daten in der Wiederherstellungssite sichern, um zusätzlich zu den Replikaten zeitpunktgenaue Sicherungen zu erstellen.

Verwendung von mehreren Availability Zones (AZs) innerhalb einer einzigen AWS-Region

Wenn Sie mehrere AZs in einer einzigen Region verwenden, nutzt Ihre DR-Implementierung mehrere Elemente der oben genannten Strategien. Zunächst müssen Sie eine Hochverfügbarkeitsarchitektur (High Availability, HA) mit mehreren AZs erstellen, wie in Abbildung 23 dargestellt. Diese Architektur

nutzt einen Aktiv/Aktiv-Ansatz für mehrere Standorte, da die Amazon EC2-Instance und der Elastic-Load-Balancer über Ressourcen verfügen, die in mehreren AZs bereitgestellt werden und aktiv Anfragen weiterleiten. Die Architektur demonstriert auch Hot-Standby, d. h. wenn die primäre Amazon RDS-Instance ausfällt (oder die AZ selbst), wird die Standby-Instance zur primären Instance befördert.

Abbildung 24: Multi-AZ-Architektur

Zusätzlich zu dieser HA-Architektur müssen Sie Backups aller Daten hinzufügen, die für die Ausführung Ihrer Workloads erforderlich sind. Dies ist besonders bei Daten wichtig, die auf eine einzige Zone beschränkt sind – wie Amazon EBS-Volumes oder Amazon Redshift-Cluster. Wenn eine AZ ausfällt, müssen Sie diese Daten in einer anderen AZ wiederherstellen. Wenn möglich, sollten Sie auch Datensicherungen auf einen anderen AWS-Region kopieren, um eine zusätzliche Sicherheit zu gewährleisten.

Ein weniger verbreiteter alternativer Ansatz für eine Single-Region, Multi-AZ-Notfallwiederherstellung wird im Blogbeitrag Building highly resilient applications using Amazon Route 53 Application Recovery Controller, Part 1: Single-Region stack (Erstellen hoch belastbarer Anwendungen mit Amazon Route 53 Application Recovery Controller, Teil 1: Single-Region-Stack) beschrieben. Hier besteht die Strategie darin, so viel Isolation wie möglich zwischen den AZs aufrechtzuerhalten, ähnlich wie bei den Regionen. Bei dieser alternativen Strategie können Sie sich für einen Aktiv/Aktiv- oder Aktiv/Passiv-Ansatz entscheiden.

> **Note**
>
> Für einige Workloads gibt es gesetzliche Vorschriften zur Aufbewahrung von Daten. Wenn dies auf Ihre Workload in einer Region zutrifft, in der es derzeit nur eine AWS-Region gibt, dann ist die Multi-Region für Ihre geschäftlichen Anforderungen nicht geeignet. Multi-AZ-Strategien bieten einen guten Schutz gegen die meisten Notfälle.

3. Beurteilen Sie die Ressourcen Ihrer Workloads und deren Konfiguration in der Wiederherstellungsregion vor dem Failover (während des normalen Betriebs).

Für die Infrastruktur und AWS-Ressourcen verwenden Sie Infrastructure-as-Code-Angebote wie AWS CloudFormation oder Drittanbieter-Tools wie Hashicorp Terraform. Um mehrere Konten und Regionen über einen einzelnen Vorgang bereitzustellen, können Sie AWS CloudFormation StackSets nutzen. Bei Multi-Site-Aktiv/Aktiv- und Hot Standby-Strategien verfügt die in Ihrer Wiederherstellungsregion bereitgestellte Infrastruktur über dieselben Ressourcen wie Ihre Primärregion. Bei den Strategien Pilot-Light und Warm-Standby sind zusätzliche Maßnahmen erforderlich, um die Infrastruktur produktionsreif zu machen. Mit CloudFormation-Parametern und bedingter Logik können Sie mit einer einzigen Vorlage steuern, ob ein bereitgestellter Stack aktiv oder standby ist. Wenn Sie Elastic Disaster Recovery verwenden, repliziert und orchestriert der Service die Wiederherstellung von Anwendungskonfigurationen und Computing-Ressourcen.

Alle Notfallwiederherstellungsstrategien setzen voraus, dass die Datenquellen innerhalb der AWS-Region gesichert werden und diese Backups dann in die Wiederherstellungsregion kopiert werden. AWS Backup bietet eine zentrale Anzeige, in der Sie Backups für diese Ressourcen konfigurieren, planen und überwachen können. Bei Pilot-Light, Warm-Standby und Multi-Site Aktiv/Aktiv sollten Sie außerdem Daten aus der primären Region auf Datenressourcen in der Wiederherstellungsregion replizieren (z. B. Amazon Relational Database Service (Amazon RDS)-DB-Instances oder Amazon DynamoDB-Tabellen. Diese Datenressourcen sind daher aktiv und bereit, Anfragen in der Wiederherstellungsregion zu bedienen.

Weitere Informationen darüber, wie AWS-Services über Regionen hinweg arbeiten, finden Sie in der Blogserie über die Erstellung einer multiregionalen Anwendung mit AWS-Services.

4. Legen Sie fest, wie Sie Ihre Wiederherstellungsregion bei Bedarf (während eines Notfallereignisses) für einen Failover bereit machen wollen, und setzen Sie diese um.

Bei Multi-Site Aktiv/Aktiv bedeutet Failover, dass eine Region evakuiert wird und die verbleibenden aktiven Regionen genutzt werden. Im Allgemeinen sind diese Regionen bereit, Datenverkehr aufzunehmen. Bei den Strategien Pilot-Light und Warm-Standby müssen Ihre Wiederherstellungsmaßnahmen die fehlenden Ressourcen bereitstellen, z. B. die EC2-Instances in Abbildung 20, sowie alle anderen fehlenden Ressourcen.

Bei allen oben genannten Strategien müssen Sie möglicherweise schreibgeschützte Instances von Datenbanken zur primären Lese-/Schreib-Instance machen.

Bei der Sicherung und Wiederherstellung werden durch die Wiederherstellung von Daten aus der Sicherung Ressourcen für diese Daten wie EBS-Volumes, RDS-DB-Instances und DynamoDB-Tabellen erstellt. Außerdem müssen Sie die Infrastruktur wiederherstellen und Code bereitstellen. Sie können AWS Backup nutzen, um Daten in der Wiederherstellungsregion wiederherzustellen. Unter REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen finden Sie weitere Informationen. Zum Wiederaufbau der Infrastruktur gehört auch die Erstellung von Ressourcen wie EC2-Instances, zusätzlich zu den Amazon Virtual Private Cloud (Amazon VPC), Subnetzen und Sicherheitsgruppen. Sie können einen Großteil des Wiederherstellungsprozesses automatisieren. Wie das geht, erfahren Sie in diesem Blogbeitrag.

5. Legen Sie fest und implementieren Sie, wie Sie den Datenverkehr bei Bedarf (im Notfall) zum Failover umleiten werden.

Dieser Failover-Vorgang kann entweder automatisch oder manuell eingeleitet werden. Ein automatisch eingeleiteter Failover auf der Grundlage von Zustandsprüfungen oder Alarmen ist mit Vorsicht zu genießen, da ein unnötiger Failover (Fehlalarm) Kosten wie Nichtverfügbarkeit und Datenverlust verursacht. Daher wird häufig ein manuell initiierter Failover verwendet. In diesem Fall sollten Sie die Schritte für den Failover dennoch automatisieren, sodass die manuelle Auslösung wie ein Knopfdruck wirkt.

Bei der Inanspruchnahme von AWS-Services gibt es mehrere Optionen für die Verwaltung des Datenverkehrs zu berücksichtigen. Eine Möglichkeit ist die Verwendung von Amazon Route 53. Mit Amazon Route 53 können Sie mehrere IP-Endpunkte in einem oder mehreren AWS-Regionen mit einem Route-53-Domänennamen verknüpfen. Um einen manuell initiierten Failover zu implementieren, können Sie Amazon Route 53 Application Recovery Controller verwenden. Dieser Service bietet eine hochverfügbare API für die Datenebene, um den Datenverkehr in die Wiederherstellungsregion umzuleiten. Verwenden Sie bei der Implementierung von Failover Vorgänge auf der Datenebene und vermeiden Sie solche auf der Steuerebene, wie beschrieben in REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung.

Weitere Informationen zu dieser und anderen Optionen finden Sie in diesem Abschnitt des Whitepapers zur Notfallwiederherstellung.

6. Entwerfen Sie einen Plan für den Failback Ihres Workloads.

Failback bedeutet, dass Sie den Workload-Betrieb in der primären Region wieder aufnehmen, nachdem ein Notfallereignis abgeklungen ist. Die Bereitstellung von Infrastruktur und Code für die primäre Region erfolgt im Allgemeinen in denselben Schritten wie ursprünglich, wobei Infrastruktur als Code und Code-Bereitstellungspipelines verwendet werden. Die Herausforderung beim Failback ist die Wiederherstellung von Datenspeichern und die Sicherstellung ihrer Konsistenz mit der in Betrieb befindlichen Wiederherstellungsregion.

Im ausgefallenen Zustand sind die Datenbanken in der Wiederherstellungsregion aktiv und verfügen über die aktuellen Daten. Ziel ist es dann, eine erneute Synchronisierung von der Wiederherstellungsregion mit der primären Region vorzunehmen, um sicherzustellen, dass diese auf dem neuesten Stand ist.

Einige AWS-Services werden das automatisch tun. Wenn Sie globale Amazon DynamoDB-Tabellen verwenden, führt DynamoDB die Weiterleitung aller ausstehenden Schreibvorgänge durch, sobald sie wieder online ist (selbst wenn die Tabelle in der primären Region nicht mehr verfügbar ist). Wenn Sie Amazon Aurora Global Database und einen verwalteten, geplanten Failover verwenden, dann wird Aurora die bestehende Replikationstopologie der globalen Datenbank beibehalten. Daher wird die ehemalige Lese-/Schreib-Instance in der primären Region zu einem Replikat und erhält Aktualisierungen von der Wiederherstellungsregion.

In Fällen, in denen dies nicht automatisch geschieht, müssen Sie die Datenbank in der primären Region als Replikat der Datenbank in der Wiederherstellungsregion neu einrichten. In vielen Fällen bedeutet dies, dass die alte primäre Datenbank gelöscht und neue Replikate erstellt werden müssen. Ein Beispiel für eine Anleitung, wie Sie dies mit Amazon Aurora Global Database unter der Annahme eines ungeplanten Failovers umsetzen, finden Sie in dieser Übung: Fail Back a Global Database (Failback einer globalen Datenbank).

Wenn Sie nach einem Failover in Ihrer Wiederherstellungsregion weiterarbeiten können, sollten Sie diese zur neuen Primärregion machen. Sie würden trotzdem alle oben genannten Schritte durchführen, um die ehemalige Primärregion in eine Wiederherstellungsregion zu verwandeln. Einige Unternehmen führen eine planmäßige Rotation durch und tauschen ihre Primär- und Wiederherstellungsregionen in regelmäßigen Abständen aus (z. B. alle drei Monate).

Alle für Failover und Failback erforderlichen Schritte sollten in einem Playbook festgehalten werden, das allen Teammitgliedern zur Verfügung steht und regelmäßig überprüft wird.

Wenn Sie Elastic Disaster Recovery verwenden, hilft der Service bei der Orchestrierung und Automatisierung des Failback-Prozesses. Weitere Details finden Sie unter Performing a failback (Durchführen eines Failbacks).

Grad des Aufwands für den Implementierungsplan: hoch

Ressourcen

Zugehörige bewährte Methoden:

- the section called “REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen”
- the section called “REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung”
- the section called “REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten:”

Zugehörige Dokumente:

- AWS Architecture Blog: Notfallwiederherstellungsserie
- Notfallwiederherstellung von Workloads auf AWS: Wiederherstellung in der Cloud (AWS-Whitepaper)
- Optionen für die Notfallwiederherstellung in der Cloud
- Entwickeln Sie eine Multi-Region-Serverless-Backend-Lösung, die aktiv/aktiv ist.
- Multi-Region-Serverless-Backend – neu aufgelegt
- RDS: Regionsübergreifendes Replizieren von Lesereplikaten
- Route 53: Konfigurieren von DNS-Failover
- S3: Regionsübergreifende Replikation
- Was ist AWS Backup?
- Was ist Route 53 Application Recovery Controller?
- AWS Elastic Disaster Recovery
- HashiCorp Terraform: Get Started – AWS (Erste Schritte)
- APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können

- AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte

Zugehörige Videos:

- Notfallwiederherstellung von Workloads auf AWS
- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2) (Architekturmuster für Aktiv/Aktiv-Anwendungen in mehreren Regionen)
- Erste Schritte mit AWS Elastic Disaster Recovery | Amazon Web Services

Zugehörige Beispiele:

- Well-Architected Lab – Disaster Recovery (Well-Architected Lab – Notfallwiederherstellung) – Eine Reihe von Workshops zur Veranschaulichung von Notfallwiederherstellungsstrategien

REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung:

Testen Sie regelmäßig den Failover zu Ihrem Wiederherstellungsstandort, um zu überprüfen, ob er ordnungsgemäß funktioniert und ob das RTO und RPO eingehalten werden.

Typische Anti-Muster:

- Failover sollten nie in der Produktion getestet werden.

Vorteile der Nutzung dieser bewährten Methode: Das regelmäßige Testen Ihres Plans zur Notfallwiederherstellung stellt sicher, dass er funktioniert, wenn er benötigt wird, und dass Ihr Team weiß, wie die Strategie auszuführen ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Vom Erstellen selten durchgeführter Wiederherstellungspfade wird abgeraten. So könnten Sie beispielsweise einen zweiten Datenspeicher unterhalten, der nur für Leseabfragen verwendet wird. Wenn Sie Daten in einen Datenspeicher schreiben und der primäre Datenspeicher einen Fehler ausgibt, können Sie einen Failover auf den zweiten Datenspeicher durchführen. Wenn Sie diesen Failover nicht regelmäßig testen, werden Sie möglicherweise feststellen, dass Ihre Annahmen zu den Möglichkeiten des sekundären Datenspeichers unzutreffend sind. Die Kapazität des zweiten Datenspeichers, die bei den letzten Tests möglicherweise noch ausreichend war,

genügt möglicherweise nicht mehr den Anforderungen dieses Szenarios. Unsere Erfahrungen haben gezeigt, dass bei einer Wiederherstellung nach einem Fehler nur der Pfad funktioniert, den Sie regelmäßig testen. Daher ist es ratsam, mehrere Wiederherstellungspfade zu pflegen. Sie können Wiederherstellungsmuster erstellen und diese regelmäßig testen. Auch komplexe oder kritische Wiederherstellungspfade müssen regelmäßig mittels Fehlersimulationen in der Produktion durchgeführt werden, um sicherzustellen, dass sie funktionieren. In dem gerade besprochenen Beispiel sollten Sie regelmäßig und unabhängig von der Erfordernis einen Failover auf die Standby-Ressourcen durchführen.

Implementierungsschritte

1. Workloads für die Wiederherstellung auslegen. Regelmäßige Tests der Wiederherstellungspfade Das Recovery-orientierte Computing identifiziert die Merkmale von Systemen, die die Wiederherstellung verbessern: Isolierung und Redundanz, systemweite Fähigkeit zur Rücknahme von Änderungen, Fähigkeit zur Überwachung und Bestimmung des Zustands, Fähigkeit zur Diagnose, automatisierte Wiederherstellung, modularer Aufbau und Fähigkeit zum Neustart. Testen Sie den Wiederherstellungspfad, um zu überprüfen, ob Sie die Wiederherstellung in der angegebenen Zeit und in dem angegebenen Zustand durchführen können. Dokumentieren Sie während dieser Wiederherstellung auftretende Probleme in Ihren Runbooks und suchen Sie vor dem nächsten Test nach Lösungen.
2. Für Amazon EC2-basierte Workloads verwenden Sie AWS Elastic Disaster Recovery, um Drill-Instances für Ihre Notfallwiederherstellungsstrategie zu implementieren und zu starten. AWS Elastic Disaster Recovery bietet die Möglichkeit, Drills effizient auszuführen, was Ihnen bei der Vorbereitung auf ein Failover-Ereignis hilft. Sie können Ihre Instances mit Elastic Disaster Recovery außerdem regelmäßig zu Test- und Übungszwecken starten, ohne den Datenverkehr weiterleiten zu müssen.

Ressourcen

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können
- AWS Architecture Blog: Notfallwiederherstellungsserie
- AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte
- AWS Elastic Disaster Recovery
- Notfallwiederherstellung von Workloads auf AWS: Wiederherstellung in der Cloud (AWS-Whitepaper)

- AWS Elastic Disaster Recovery – Vorbereitungen auf einen Failover
- The Berkeley/Stanford Recovery-Oriented Computing (ROC) Project
- Was ist AWS Fault Injection Simulator?

Zugehörige Videos:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (AWS re:Invent 2018: Architekturmuster für Multi-Region Aktiv/Aktive-Anwendungen)
- AWS re:Invent 2019: Backup-and-restore and disaster-recovery solutions with AWS (AWS re:Invent 2019: Backup-and-Wiederherstellung und Notfallwiederherstellungs-Lösungen mit AWS)

Zugehörige Beispiele:

- Well-Architected Lab – Testing for Resiliency (Well-Architected Lab – Testen auf Ausfallsicherheit)

REL13-BP04 Verwalten der Konfigurationsabweichungen am Standort oder in der Region der Notfallwiederherstellung:

Stellen Sie sicher, dass die Infrastruktur, die Daten und die Konfiguration am Standort oder in der Region der Notfallwiederherstellung den Anforderungen entsprechen. Sie sollten beispielsweise prüfen, ob AMIs und Service Quotas auf dem neuesten Stand sind.

AWS Config überwacht und zeichnet Ihre AWS-Ressourcenkonfigurationen kontinuierlich auf. Es kann Abweichungen erkennen und als Auslöser für AWS Systems Manager Automation dienen, um diese zu beheben und Warnmeldungen zu senden. AWS CloudFormation kann zusätzlich Abweichungen in bereitgestellten Stacks erkennen.

Gängige Antimuster:

- Versäumnis, Aktualisierungen an Ihren Wiederherstellungsstandorten vorzunehmen, wenn Sie Konfigurations- oder Infrastrukturänderungen an Ihren Hauptstandorten vornehmen.
- Mögliche Einschränkungen (z. B. Serviceunterschiede) an Ihren primären Standorten und den Standorten für die Notfallwiederherstellung werden nicht berücksichtigt.

Vorteile der Einführung dieser bewährten Methode: Wenn Ihre Umgebung für die Notfallwiederherstellung mit der vorhandenen Umgebung konsistent ist, gewährleisten dies eine vollständige Wiederherstellung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Sicherstellen, dass die Bereitstellung an Haupt- und Sicherungsstandorte erfolgt Pipelines für die Bereitstellung von Anwendungen in der Produktion müssen die Anwendungen an alle Standorte verteilen, die in der Strategie für die Notfallwiederherstellung angegeben sind. Dazu gehören auch Entwicklungs- und Testumgebungen.
- Aktivieren von AWS Config zum Verfolgen von Standorten mit möglichen Abweichungen. Erstellen Sie mithilfe von AWS Config Regeln Systeme, die Ihre Strategien für die Notfallwiederherstellung durchsetzen und bei Erkennung von Abweichungen Warnungen generieren.
  - Korrigieren von nicht konformen AWS-Ressourcen mit AWS-Config-Regeln
  - AWS Systems Manager Automation
- Verwenden Sie AWS CloudFormation zur Bereitstellung Ihrer Infrastruktur. AWS CloudFormation kann Abweichungen zwischen den Angaben in den CloudFormation-Vorlagen und der tatsächlichen Bereitstellung erkennen.
  - AWS CloudFormation: Ermitteln von Abweichungen im gesamten CloudFormation-Stack

Ressourcen

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können
- AWS Architecture Blog: Notfallwiederherstellungsserie
- AWS CloudFormation: Ermitteln von Abweichungen im gesamten CloudFormation-Stack
- AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte
- AWS Systems Manager Automation
- Notfallwiederherstellung von Workloads auf AWS: Wiederherstellung in der Cloud (AWS-Whitepaper)
- Wie implementiere ich eine Lösung für die Verwaltung der Infrastrukturkonfiguration in AWS?
- Korrigieren von nicht konformen AWS-Ressourcen mit AWS-Config-Regeln

Relevante Videos:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2) (Architekturmuster für Aktiv-Aktiv-Anwendungen in mehreren Regionen)

REL13-BP05: Automatisieren der Wiederherstellung

Automatisieren Sie mit Tools von AWS oder Drittanbietern die Systemwiederherstellung und leiten Sie Datenverkehr zum Standort oder zur Region der Notfallwiederherstellung weiter.

Basierend auf konfigurierten Zustandsprüfungen können AWS-Services wie Elastic Load Balancing und AWS Auto Scaling die Last auf fehlerfreie Availability Zones verteilen während Services wie z. B, Amazon Route 53 und AWS Global Accelerator, die Last an fehlerfreie AWS-Regionen leiten können. Amazon Route 53 Application Recovery Controller hilft Ihnen, mithilfe von Bereitschaftsprüfungen und Routing-Steuerungsfunktionen Failover-Vorgänge zu verwalten und zu koordinieren. Diese Funktionen überwachen kontinuierlich die Fähigkeit Ihrer Anwendung, eine Wiederherstellung nach Fehlern durchzuführen, so dass Sie die Wiederherstellung der Anwendung über mehrere AWS-Regionen, Availability Zones und On-Premises kontrollieren können.

Für Workloads in bestehenden physischen oder virtuellen Rechenzentren oder privaten Clouds, AWS Elastic Disaster Recovery,verfügbar durch AWS Marketplace, ermöglicht es Unternehmen, eine automatisierte Notfallwiederherstellungsstrategie auf AWS einzurichten. CloudEndure unterstützt auch die regions- bzw. AZ-übergreifende Notfallwiederherstellung in AWS.

Gängige Antimuster:

- Die Implementierung von identischem automatisiertem Failover und Failback kann bei einem Fehler zu Flapping führen.

Vorteile der Einführung dieser bewährten Methode: Die automatisierte Wiederherstellung verkürzt die Wiederherstellungszeit, da manuelle Fehler nicht mehr möglich sind.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Automatisieren von Wiederherstellungspfaden. Wenn in Szenarien mit hoher Verfügbarkeit kurze Wiederherstellungszeiten erforderlich sind, sind menschliche Beurteilungen und Aktionen zu langsam. Das System sollte in jeder Situation in der Lage sein, eine Wiederherstellung durchzuführen.

- Verwenden Sie CloudEndure Disaster Recovery für automatisiertes Failover und Failback. CloudEndure Disaster Recovery repliziert Ihre Computer (einschließlich Betriebssystem, Systemstatuskonfiguration, Datenbanken, Anwendungen und Dateien) kontinuierlich in einen kostengünstigen Staging-Bereich in Ihrem AWS-Konto-Zielkonto und in Ihrer bevorzugten Region. Bei einem Notfall können Sie CloudEndure Disaster Recovery anweisen, innerhalb weniger Minuten automatisch Tausende Ihrer virtuellen Maschinen vollständig bereitgestellt zu starten.
  - Ausführen von Failover und Failback bei Notfallwiederherstellungen
  - CloudEndure Disaster Recovery

Ressourcen

Zugehörige Dokumente:

- APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können
- AWS Architecture Blog: Notfallwiederherstellungsserie
- AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte
- AWS Systems Manager Automation
- CloudEndure Disaster Recovery auf AWS
- Notfallwiederherstellung von Workloads auf AWS: Wiederherstellung in der Cloud (AWS-Whitepaper)

Relevante Videos:

- AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications (ARC209-R2) (Architekturmuster für Aktiv-Aktiv-Anwendungen in mehreren Regionen)

Leistungseffizienz

Die Säule „Leistungseffizienz“ umfasst die Fähigkeit, Rechenressourcen effizient entsprechend den Systemanforderungen zu nutzen und diese Effizienz aufrechtzuerhalten, während sich die Nachfrage ändert und die Technologie weiterentwickelt. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule der Leistungseffizienz.

Bereiche für bewährte Methoden

- Auswahl der Architektur
- Computer und Hardware
- Datenverwaltung
- Networking und Bereitstellung von Inhalten
- Prozess und Kultur

Auswahl der Architektur

Fragen

- LEIST 1. Wie wählen Sie geeignete Cloud-Ressourcen und -Architekturen für Ihren Workload aus?

LEIST 1. Wie wählen Sie geeignete Cloud-Ressourcen und -Architekturen für Ihren Workload aus?

Die optimale Lösung für einen bestimmten Workload variiert und Lösungen bestehen häufig aus einer Kombination mehrerer Ansätze. Well-Architected-Workloads nutzen mehrere Lösungen und ermöglichen verschiedene Funktionen zur Verbesserung der Leistung.

Bewährte Methoden

- PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features
- PERF01-BP02 Einholen von Rat beim Cloud-Anbieter oder einem geeigneten Partner, um mehr über Architekturmuster und bewährte Methoden zu erfahren
- PERF01-BP03 Berücksichtigen der Kosten bei architektonischen Entscheidungen
- PERF01-BP04 Evaluieren, wie sich Kompromisse auf Kunden und Architektureffizienz auswirken
- PERF01-BP05 Verwenden von Richtlinien und Referenzarchitekturen
- PERF01-BP06 Verwenden von Benchmarking, um architektonische Entscheidungen zu treffen
- PERF01-BP07 Verwenden eines datengesteuerten Ansatzes für architektonische Entscheidungen

PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features

Informieren Sie sich kontinuierlich über verfügbare Services und Konfigurationen, die Ihnen helfen, bessere architektonische Entscheidungen zu treffen und die Leistungseffizienz Ihrer Workload-Architektur zu verbessern.

Typische Anti-Muster:

- Sie verwenden die Cloud als gemeinsam genutztes Rechenzentrum.
- Sie modernisieren die Anwendung nach der Migration in die Cloud nicht.
- Sie verwenden nur einen Speichertyp für alle Objekte, die gespeichert werden müssen.
- Sie verwenden Instance-Typen, die am besten zu Ihren aktuellen Standards passen, bei Bedarf jedoch größer sind.
- Von Ihnen werden Technologien bereitgestellt und verwaltet, die als verwaltete Services verfügbar sind.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie neue Services und Konfigurationen in Betracht ziehen, können Sie möglicherweise die Leistung erheblich verbessern, die Kosten senken und den Aufwand für die Aufrechterhaltung des Workloads optimieren. Es kann Ihnen auch dabei helfen, die Wertschöpfung für Cloud-fähige Produkte zu beschleunigen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS veröffentlicht kontinuierlich neue Services und Features, mit denen die Leistung verbessert und die Kosten von Cloud-Workloads gesenkt werden können. Es ist entscheidend, mit diesen neuen Services und Features auf dem Laufenden zu bleiben, um die Leistungseffizienz in der Cloud aufrechtzuerhalten. Die Modernisierung der Workload-Architektur hilft Ihnen auch dabei, die Produktivität zu beschleunigen, Innovationen voranzutreiben und mehr Wachstumschancen zu erschließen.

Implementierungsschritte

- Inventarisieren Sie die Workload-Software und -Architektur für verwandte Services. Entscheiden Sie, über welche Produktkategorie Sie mehr erfahren möchten.
- Erkunden Sie die AWS-Angebote, um die relevanten Services und Konfigurationsoptionen zu identifizieren und kennenzulernen, mit denen Sie die Leistung verbessern und die Kosten und die betriebliche Komplexität reduzieren können.
  - Amazon Web Services Cloud
  - AWS Academy
  - Neuerungen bei AWS
  - AWS-Blog

  - AWS Skill Builder
  - AWS-Veranstaltungen und -Webinare
  - AWS Training und -Zertifizierungen
  - YouTube-Kanal: AWS
  - AWS-Workshops
  - AWS-Communitys
- Verwenden Sie Sandbox- bzw. Nicht-Produktionsumgebungen, um neue Services zu erlernen und mit ihnen zu experimentieren, ohne dass zusätzliche Kosten anfallen.
- Informieren Sie sich kontinuierlich über neue Cloud-Services und -Features.

Ressourcen

Zugehörige Dokumente:

- Übersicht über Amazon Web Services
- Amazon EC2-Features
- Lernen Sie Schritt für Schritt mit einem Lehrplan für AWS-Partner
- AWS Training and Certification
- Mein Lernpfad zum AWS Solutions Architect
- AWS-Architekturzentrum
- AWS Partner Network
- Die AWS-Lösungsbibliothek
- AWS Knowledge Center
- Moderne Anwendungen in AWS entwickeln

Zugehörige Videos:

- AWS re:Invent 2023 – Neuerungen bei Amazon EC2
- AWS re:Invent 2022 – Betriebs- und Infrastrukturkosten mit Amazon ECS senken
- AWS re:Invent 2023 – Mit AWS die Effizienz, Agilität und Innovation der Cloud nutzen
- AWS re:Invent 2022 – ML-Modelle für Inferenz mit hoher Leistung und niedrigen Kosten bereitstellen

- This is My Architecture

Zugehörige Beispiele:

- AWS Samples
- AWS-SDK-Beispiele

PERF01-BP02 Einholen von Rat beim Cloud-Anbieter oder einem geeigneten Partner, um mehr über Architekturmuster und bewährte Methoden zu erfahren

Greifen Sie bei Ihren architektonischen Entscheidungen auf die Ressourcen von Cloud-Unternehmen, wie etwa Dokumentation, Lösungsarchitekten, professionelle Services oder einen geeigneten Partner zurück. Diese Ressourcen helfen Ihnen dabei, Ihre Architektur zu überprüfen und zu verbessern, um so die Leistung zu optimieren.

Typische Anti-Muster:

- Sie verwenden AWS als gängigen Cloud-Anbieter.
- Sie verwenden AWS-Services auf eine Weise, für die sie nicht konzipiert wurden.
- Sie befolgen alle Anweisungen, ohne Ihren Geschäftskontext zu berücksichtigen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie sich Rat bei einem Cloud-Anbieter oder einem geeigneten Partner einholen, können Sie die richtige Architektur für den Workload wählen und Entscheidungen mit größerer Zuversicht treffen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

AWS bietet eine breite Palette an Anleitungen, Dokumentation und Ressourcen, die Sie bei der Entwicklung und Verwaltung effizienter Cloud-Workloads unterstützen können. Die AWS-Dokumentation enthält Codebeispiele, Tutorials und detaillierte Serviceerklärungen. Zusätzlich zur Dokumentation bietet AWS Trainings- und Zertifizierungsprogramme, Lösungsarchitekten und professionelle Services, die Kunden dabei helfen können, verschiedene Aspekte von Cloud-Services zu entdecken und eine effiziente Cloud-Architektur in AWS zu implementieren.

Nutzen Sie diese Ressourcen, um Einblicke in wertvolles Wissen und bewährte Methoden zu gewinnen, Zeit zu sparen und bessere Ergebnisse in der AWS Cloud zu erzielen.

Implementierungsschritte

- Lesen Sie die AWS-Dokumentation und -Anleitungen und befolgen Sie die bewährten Methoden. Diese Ressourcen können Ihnen helfen, Services effektiv auszuwählen und zu konfigurieren und eine bessere Leistung zu erzielen.
  - AWS-Dokumentation (wie Benutzerhandbücher und Whitepapers)
  - AWS-Blog
  - AWS Training und -Zertifizierungen
  - YouTube-Kanal: AWS
- Nehmen Sie an AWS-Partnerveranstaltungen (wie AWS Global Summits, AWS re:Invent, Benutzergruppen und Workshops) teil, um von AWS-Experten mehr über bewährte Methoden für die Nutzung von AWS-Services zu lernen.
  - Lernen Sie Schritt für Schritt mit einem Lehrplan für AWS-Partner
  - AWS-Veranstaltungen und -Webinare
  - AWS-Workshops
  - AWS-Communitys
- Wenden Sie sich an AWS, wenn Sie zusätzliche Anleitungen oder Produktinformationen benötigen. AWS Solutions Architects und AWS Professional Services liefern Ratschläge für die Implementierung von Lösungen. AWS-Partner bieten AWS-Fachwissen, damit Sie in Ihrem Unternehmen flexibel agieren und Innovationen nutzen können.
- Verwenden Sie Support, wenn Sie technischen Support benötigen, um einen Service effektiv nutzen zu können. Unsere Support-Pläne bieten Ihnen die richtige Kombination aus Tools und Zugang zu Fachwissen, um die Grundlagen für Ihren Erfolg mit AWS zu legen, ohne dabei Themen wie Leistungsoptimierung, Risikomanagement und Kostenkontrolle zu vernachlässigen.

Ressourcen

Zugehörige Dokumente:

- AWS-Architekturzentrum
- AWS Partner Network
- AWS-Lösungsbibliothek
- AWS Knowledge Center
- AWS Enterprise Support

Zugehörige Videos:

- This is My Architecture
- AWS re:Invent 2023 – Fortgeschrittene ereignisgesteuerte Muster mit Amazon EventBridge
- AWS re:Invent 2023 – Implementierung verteilter Designmuster in AWS
- AWS re:Invent 2023 – Anwendungsarchitektur als Code

Zugehörige Beispiele:

- AWS Samples
- AWS-SDK-Beispiele
- AWS Analytics-Referenzarchitektur

PERF01-BP03 Berücksichtigen der Kosten bei architektonischen Entscheidungen

Berücksichtigen Sie die Kosten bei Ihren architektonischen Entscheidungen, um die Ressourcennutzung und Leistungseffizienz der Cloud-Workloads zu verbessern. Wenn Sie sich der Kostenauswirkungen des Cloud-Workloads bewusst sind, ist es wahrscheinlicher, dass Sie effiziente Ressourcen nutzen und verschwenderische Methoden reduzieren.

Typische Anti-Muster:

- Sie verwenden nur eine Instance-Familie.
- Sie bewerten keine lizenzierten Lösungen verglichen mit Open-Source-Lösungen.
- Sie definieren keine Speicher-Lebenszyklusrichtlinien.
- Sie prüfen keine neuen Services und Features der AWS Cloud.
- Sie nutzen nur Blockspeicher.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie die Kosten bei Ihrer Entscheidungsfindung berücksichtigen, können Sie effizientere Ressourcen einsetzen und andere Investitionen in Betracht ziehen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die Kostenoptimierung von Workloads kann die Ressourcennutzung verbessern und Verschwendung bei einem Cloud-Workload vermeiden. Die Berücksichtigung der Kosten bei architektonischen Entscheidungen beinhaltet in der Regel die richtige Dimensionierung der Workload-Komponenten und die Schaffung von Elastizität. Dies führt zu einer verbesserten Leistungseffizienz von Cloud-Workloads.

Implementierungsschritte

- Legen Sie Kostenziele wie Budgetlimits für den Cloud-Workload fest.
- Identifizieren Sie die wesentlichen Komponenten (wie Instances und Speicher), die die Kosten des Workloads erhöhen. Nutzen Sie Instrumentierungsservices wie AWS -Preisrechner und AWS Cost Explorer, um die wichtigsten Kostentreiber in Ihrem Workload zu identifizieren.
- Verstehen Sie die Preismodelle in der Cloud, z. B. On-Demand, Reserved Instances, Savings Plans und Spot Instances.
- Verwenden Sie Best Practices zur Kostenoptimierung bei Well-Architected-Technologien, um diese Schlüsselkomponenten aus Kostengründen zu optimieren.
- Überwachen und analysieren Sie kontinuierlich die Kosten, um Möglichkeiten zur Kostenoptimierung im Workload zu identifizieren.
  - Verwenden Sie AWS-Budgets, um bei nicht akzeptablen Kosten Warnungmeldungen zu erhalten.
  - Verwenden Sie AWS Compute Optimizer oder AWS Trusted Advisor , um Empfehlungen zur Kostenoptimierung zu erhalten.
  - Verwenden Sie AWS Cost Anomaly Detection, um das automatisierte Erkennen von Kostenanomalien mit Ursachenanalyse zu erhalten.

Ressourcen

Zugehörige Dokumente:

- Was ist AWS-Fakturierung und Kostenmanagement?
- Kostenoptimierung mit AWS
- Auswahl einer AWS-Kostenverwaltungsstrategie
- Ein Leitfaden für Anfänger zur AWS-Kostenverwaltung
- Eine detaillierte Übersicht über das Cost Intelligence Dashboard

- AWS-Architekturzentrum
- Die AWS-Lösungsbibliothek
- AWS Knowledge Center

Zugehörige Videos:

- This is My Architecture
- AWS re:Invent 2023 – Neuerungen bei der AWS-Kostenoptimierung
- AWS re:Invent 2023 – Optimieren der Kosten und Leistung sowie Verfolgen der Fortschritte bei der Schadensbegrenzung
- AWS re:Invent 2023 – Bewährte Methoden zur Kostenoptimierung für AWS-Speicher
- AWS re:Invent 2023 – Optimieren der Kosten in Ihren Umgebungen mit mehreren Konten

Zugehörige Beispiele:

- AWS Compute Optimizer-Demo-Code
- Workshop zur Kostenoptimierung
- Technische Playbooks zur Implementierung von Cloud Financial Management
- Startoptimierung: Optimierung der Anwendungsleistung für maximale Effizienz
- Workshop zur Serverless-Optimierung (Leistung und Kosten)
- Skalierung kostengünstiger Architekturen

PERF01-BP04 Evaluieren, wie sich Kompromisse auf Kunden und Architektureffizienz auswirken

Ermitteln Sie beim Evaluieren von leistungsbezogenen Verbesserungen, welche gewählten Optionen sich auf Ihre Kunden und die Effizienz der Workloads auswirken. Wenn sich die Systemleistung beispielsweise bei Verwendung eines Schlüssel-Wert-Datenspeichers erhöht, sollten Sie unbedingt ermitteln, welche Auswirkungen sich bei einem dauerhaften Einsatz für die Kunden ergeben würden.

Typische Anti-Muster:

- Sie gehen davon aus, dass alle Leistungsgewinne implementiert werden sollten, auch wenn es Kompromisse für die Implementierung gibt.
- Änderungen an Workloads werden nur dann ausgewertet, wenn ein Leistungsproblem einen kritischen Punkt erreicht hat.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie potenzielle leistungsbezogene Verbesserungen bewerten, müssen Sie entscheiden, ob die Kompromisse für die Änderungen angesichts der Workload-Anforderungen akzeptabel sind. In einigen Fällen müssen Sie möglicherweise zusätzliche Kontrollen implementieren, um Kompromisse zu kompensieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Identifizieren Sie kritische Bereiche in der Architektur in Bezug auf Leistung und Kundenauswirkung. Stellen Sie fest, welche Verbesserungen möglich und welche Kompromisse damit verbunden sind und wie sich diese auf das System und das Benutzererlebnis auswirken. So lässt sich beispielsweise durch Caching von Daten die Leistung deutlich steigern. Es ist aber eine eindeutige Strategie erforderlich, mit der festgelegt wird, wie und wann Cache-Daten aktualisiert oder ungültig werden, um unerwünschtes Systemverhalten zu verhindern.

Implementierungsschritte

- Verstehen Sie Ihre Workload-Anforderungen und SLAs.
- Definieren Sie klare Bewertungsfaktoren. Faktoren können sich auf Kosten, Zuverlässigkeit, Sicherheit und Leistung des Workloads beziehen.
- Wählen Sie die Architektur und Services, die Ihren Anforderungen entsprechen.
- Führen Sie Experimente und Machbarkeitsstudien (POCs) durch, um Kompromissfaktoren und Auswirkungen auf Kunden und Architektureffizienz zu bewerten. In der Regel verbrauchen hochverfügbare, leistungsstarke und sichere Workloads mehr Cloud-Ressourcen und bieten gleichzeitig ein besseres Kundenerlebnis. Machen Sie sich ein Bild von den Kompromissen in Bezug auf Komplexität, Leistung und Kosten Ihrer Workloads. In der Regel geht die Priorisierung von zwei der Faktoren auf Kosten des dritten.

Ressourcen

Zugehörige Dokumente:

- Amazon Builders' Library
- QuickSight-KPIs
- Amazon CloudWatch RUM
- X-Ray-Dokumentation

- Resilienzmuster und Kompromisse verstehen, um eine effiziente Architektur in der Cloud zu entwickeln

Zugehörige Videos:

- Optimieren von Anwendungen mithilfe von Amazon CloudWatch RUM
- AWS re:Invent 2023 – Kapazität, Verfügbarkeit, Kosteneffizienz: Wählen Sie drei Optionen aus
- AWS re:Invent 2023 – Erweiterte Integrationsmuster und Kompromisse für lose gekoppelte Systeme

Zugehörige Beispiele:

- Messen der Seitenladezeit mit Amazon CloudWatch Synthetics
- Amazon CloudWatch RUM Web Client

PERF01-BP05 Verwenden von Richtlinien und Referenzarchitekturen

Verwenden Sie interne Richtlinien und vorhandene Referenzarchitekturen bei der Auswahl von Services und Konfigurationen, um den Workload effizienter zu gestalten und zu implementieren.

Typische Anti-Muster:

- Sie erlauben eine Vielzahl von Technologien, was sich auf den Verwaltungsaufwand Ihres Unternehmens auswirken kann.

Vorteile der Nutzung dieser bewährten Methode: Durch Festlegung einer Richtlinie für die Architektur-, Technologie und Anbieterauswahl können Entscheidungen schnell getroffen werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Interne Richtlinien bei der Auswahl von Ressourcen und Architektur bieten Standards und Leitlinien, die bei Architekturentscheidungen zu beachten sind. Diese Richtlinien vereinfachen den Entscheidungsprozess bei der Auswahl des richtigen Cloud-Service und können zur Verbesserung der Leistungseffizienz beitragen. Stellen Sie den Workload mithilfe von Richtlinien oder Referenzarchitekturen bereit. Integrieren Sie die Services in Ihre Cloud-Bereitstellung. Überprüfen Sie

anschließend anhand von Leistungstests, dass Sie die eigenen Leistungsanforderungen weiterhin erfüllen können.

Implementierungsschritte

- Verstehen Sie die Anforderungen des Cloud-Workloads genau.
- Überprüfen Sie die internen und externen Richtlinien, um die relevantesten zu ermitteln.
- Verwenden Sie die entsprechenden Referenzarchitekturen, die von AWS bereitgestellt werden, oder die branchenweit anerkannten bewährten Methoden.
- Schaffen Sie ein Kontinuum, das aus Richtlinien, Standards, Referenzarchitekturen und präskriptiven Richtlinien für häufig auftretende Situationen besteht. Auf diese Weise können Ihre Teams schneller vorankommen. Passen Sie die Komponenten gegebenenfalls an die Branche an.
- Prüfen Sie diese Richtlinien und Referenzarchitekturen für den Workload in Sandbox-Umgebungen.
- Bleiben Sie über Industriestandards und AWS-Updates auf dem Laufenden, um sicherzustellen, dass die Richtlinien und Referenzarchitekturen zur Optimierung des Cloud-Workloads beitragen.

Ressourcen

Zugehörige Dokumente:

- AWS-Architekturzentrum
- AWS Partner Network
- Die AWS-Lösungsbibliothek
- AWS Knowledge Center
- AWS Architecture Blog

Zugehörige Videos:

- This is My Architecture
- AWS re:Invent 2022 – Beschleunigen der Wertschöpfung für Ihr Unternehmen mit SAP und der AWS-Referenzarchitektur

Zugehörige Beispiele:

- AWS Samples

- AWS-SDK-Beispiele

PERF01-BP06 Verwenden von Benchmarking, um architektonische Entscheidungen zu treffen

Führen Sie einen Benchmark-Vergleich für einen vorhandenen Workload durch, um sich ein Bild über dessen Leistung in der Cloud zu verschaffen, und treffen Sie architektonische Entscheidungen auf der Grundlage dieser Daten.

Typische Anti-Muster:

- Sie verlassen sich auf gängige Benchmarks, die für die Workload-Merkmale nicht aufschlussreich sind.
- Sie verlassen sich auf Kundenfeedback und Kundenwahrnehmung als einzige Benchmark.

Vorteile der Nutzung dieser bewährten Methode: Durch das Benchmarking der aktuellen Implementierung können Sie Leistungsverbesserungen messen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Kombinieren Sie Benchmarking mit synthetischen Tests, um die Leistung Ihrer Workload-Komponenten zu bewerten. Benchmarking lässt sich in der Regel schneller als Lasttests einrichten und dient zur Bewertung der Technologie einer bestimmten Komponente. Ein Benchmarking wird oft zu Beginn eines neuen Projekts durchgeführt, wenn Sie noch keine vollständige Lösung für einen Lasttest haben.

Sie können wahlweise eigene Benchmark-Tests erstellen oder einen branchenüblichen Standardtest wie etwa TPC-DS für das Benchmarking der Workloads verwenden. Branchen-Benchmarks sind zum Vergleich von Umgebungen nützlich. Benutzerdefinierte Benchmarks eignen sich zum Prüfen spezieller Arten von Vorgängen, die Sie in der Architektur ausführen möchten.

Beim Benchmarking ist es wichtig, die Testumgebung entsprechend vorzubereiten, um aussagekräftige Ergebnisse zu erzielen. Führen Sie denselben Benchmark-Test mehrmals aus, um sicherzustellen, dass alle Varianzen im Laufe der Zeit ermittelt wurden.

Da sich Benchmarks in der Regel schneller als Lasttests ausführen lassen, können Sie früher in der Bereitstellungspipeline eingesetzt werden und schneller Feedback zu Leistungsabweichungen

liefern. Wenn Sie eine wesentliche Veränderung einer Komponente oder eines Services bewerten, können Sie schnell ermitteln, ob der Aufwand für die Korrektur gerechtfertigt ist. Die Verwendung von Benchmarking in Verbindung mit Lasttests ist wichtig, da letztere Auskunft über die Leistung der Workload in der Produktion geben.

Implementierungsschritte

- Planen und Definieren:
  - Definieren Sie die Ziele, Baselines, Testszenarien, Metriken (wie CPU-Auslastung, Latenz oder Durchsatz) und KPIs für Ihren Benchmark.
  - Konzentrieren Sie sich auf die Benutzeranforderungen in Bezug auf die Benutzererlebnis und Faktoren wie Reaktionszeit und Barrierefreiheit.
  - Identifizieren Sie ein Benchmarking-Tool, das für Ihren Workload geeignet ist. Sie können AWS-Services wie Amazon CloudWatch oder ein Drittanbieter-Tool verwenden, das mit Ihrem Workload kompatibel ist.
- Konfiguration und Verwendung:
  - Richten Sie Ihre Umgebung ein und konfigurieren Sie Ihre Ressourcen.
  - Implementieren Sie Überwachungs- und Protokollierungsfunktionen, um Testergebnisse zu erfassen.
- Benchmarking und Überwachung:
  - Führen Sie die Benchmark-Tests durch und überwachen Sie die Metriken während des Tests.
- Analyse und Dokumentation:
  - Dokumentieren Sie Ihren Benchmarking-Prozess und die entsprechende Erkenntnisse.
  - Analysieren Sie die Ergebnisse, um Engpässe, Trends und Verbesserungsmöglichkeiten zu identifizieren.
  - Verwenden Sie die Testergebnisse, um die Architektur betreffende Entscheidungen zu fällen und das Workload anzupassen. Dies kann die Änderung von Services oder die Einführung neuer Funktionen beinhalten.
- Optimierung und Wiederholung:
  - Passen Sie die Ressourcenkonfigurationen und -zuweisungen auf der Grundlage Ihrer Benchmarks an.
  - Testen Sie Ihr Workload nach der Anpassung erneut, um Ihre Verbesserungen zu überprüfen.
  - Dokumentieren Sie Ihre Erkenntnisse und wiederholen Sie den Prozess, um weitere Verbesserungsmöglichkeiten zu identifizieren.

Ressourcen

Zugehörige Dokumente:

- AWS-Architekturzentrum
- AWS Partner Network
- AWS-Lösungsbibliothek
- AWS Knowledge Center
- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Genomik-Workflows, Teil 5: automatisiertes Benchmarking
- Benchmarking und Optimierung der Endpunktbereitstellung in Amazon SageMaker AI JumpStart

Zugehörige Videos:

- AWS re:Invent 2023 — Benchmarking von AWS Lambda-Kaltstarts
- Benchmarking von zustandsbehafteten Services in der Cloud
- This is My Architecture: Expedia
- Optimieren von Anwendungen mithilfe von Amazon CloudWatch RUM
- Demo von Amazon CloudWatch Synthetics

Zugehörige Beispiele:

- AWS-Beispiele
- AWS-SDK-Beispiele
- Verteilte Belastungstests
- Messen der Seitenladezeit mit Amazon CloudWatch Synthetics
- Amazon CloudWatch RUM Web Client

PERF01-BP07 Verwenden eines datengesteuerten Ansatzes für architektonische Entscheidungen

Definieren Sie einen klaren, datengesteuerten Ansatz für architektonische Entscheidungen, um sicherzustellen, dass die richtigen Cloud-Services und -Konfigurationen verwendet werden, um Ihre spezifischen Geschäftsanforderungen zu erfüllen.

Typische Anti-Muster:

- Sie gehen davon aus, dass die aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert werden sollte.
- Ihre architektonischen Entscheidungen basieren auf Vermutungen und Annahmen.
- Sie führen im Laufe der Zeit Änderungen an der Architektur ein, ohne sie zu begründen.

Vorteile der Nutzung dieser bewährten Methode: Durch einen klar definierten Ansatz für architektonische Entscheidungen verwenden Sie Daten, um das Workload-Design zu beeinflussen und im Laufe der Zeit fundierte Entscheidungen zu treffen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nutzen Sie interne Erfahrungen und Kenntnisse im Zusammenhang mit der Cloud oder ziehen Sie externe Ressourcen heran, wie etwa veröffentlichte Anwendungsbeispiele oder Whitepapers, um Ressourcen und Services in der Architektur auszuwählen. Sie sollten über einen klar definierten Prozess verfügen, der das Experimentieren und Benchmarking mit den Services fördert, die im Workload verwendet werden könnten.

Backlogs für kritische Workloads sollten nicht nur aus Benutzerszenarien bestehen, die für das Unternehmen und die Benutzer relevante Funktionen bereitstellen, sondern auch aus technischen Szenarien, die ein architektonisches System für den Workload bilden. Dieses System stützt sich auf neue technologische Fortschritte sowie neue Services und nimmt diese auf der Grundlage von Daten und entsprechender Begründung an. Dies stellt sicher, dass die Architektur zukunftssicher bleibt und nicht stagniert.

Implementierungsschritte

- Arbeiten Sie mit wichtigen Interessenvertretern zusammen, um die Workload-Anforderungen zu definieren, einschließlich Überlegungen zu Leistung, Verfügbarkeit und Kosten. Berücksichtigen Sie Faktoren wie die Anzahl der Benutzer und das Nutzungsmuster für den Workload.
- Erstellen Sie ein Architektursystem oder einen Technologie-Backlog, der zusammen mit dem funktionalen Backlog priorisiert wird.
- Bewerten und beurteilen Sie verschiedene Cloud-Services (weitere Informationen finden Sie unter PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features).

- Erkunden Sie verschiedene Architekturmuster wie Microservices oder Serverless, die Ihren Leistungsanforderungen entsprechen (weitere Informationen finden Sie unter PERF01-BP02 Einholen von Rat beim Cloud-Anbieter oder einem geeigneten Partner, um mehr über Architekturmuster und bewährte Methoden zu erfahren).

- Konsultieren Sie andere Teams, Architekturdiagramme und Ressourcen wie AWS Solution Architects, AWS-Architekturzentrumund AWS Partner Network,um Ihnen bei der Auswahl der richtigen Architektur für Ihren Workload zu helfen.

- Definieren Sie Leistungsmetriken wie Durchsatz und Reaktionszeit, anhand derer Sie die Leistung des Workloads bewerten können.
- Experimentieren Sie und verwenden Sie definierte Metriken, um die Leistung der ausgewählten Architektur zu validieren.
- Überwachen Sie kontinuierlich und nehmen Sie bei Bedarf Anpassungen vor, um die optimale Leistung der Architektur aufrechtzuerhalten.
- Dokumentieren Sie Ihre gewählte Architektur und Entscheidungen als Referenz für zukünftige Updates und Erkenntnisse.
- Überprüfen und aktualisieren Sie den Ansatz zur Architekturauswahl kontinuierlich auf der Grundlage von Erkenntnissen, neuen Technologien und Metriken, die auf eine notwendige Änderung oder ein Problem im aktuellen Ansatz hinweisen.

Ressourcen

Zugehörige Dokumente:

- Die AWS-Lösungsbibliothek
- AWS Knowledge Center
- Architekturmodelle für die Erstellung von datengesteuerten End-to-End-Anwendungen in AWS

Zugehörige Videos:

- This is My Architecture
- AWS re:Invent 2021 – Das datengesteuerte Unternehmen: Von der Vision zum Mehrwert
- AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen

- AWS re:Invent 2023 – Optimieren der Kosten und Leistung sowie Verfolgen der Fortschritte bei der Schadensbegrenzung
- AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse

Zugehörige Beispiele:

- AWS Samples
- AWS-SDK-Beispiele

Computer und Hardware

LEIST 2. Wie wählen und nutzen Sie Computing-Ressourcen für Ihren Workload?

Die optimale Datenverarbeitungsoption für einen bestimmten Workload kann sich je nach Anwendungsdesign, Nutzungsmustern und Konfigurationseinstellungen unterscheiden. Architekturen können verschiedene Computing-Optionen für verschiedene Komponenten verwenden und verschiedene Funktionen zur Verbesserung der Leistung bieten. Die Wahl der falschen Datenverarbeitungslösung für eine Architektur kann die Leistungseffizienz schmälern.

Bewährte Methoden

- PERF02-BP01 Auswählen der besten Datenverarbeitungsoptionen für den Workload
- PERF02-BP02 Verstehen verfügbarer Konfigurationen und Features für die Datenverarbeitung
- PERF02-BP03 Erfassen von Datenverarbeitungsmetriken
- PERF02-BP04 Konfigurieren und richtiges Dimensionieren von Datenverarbeitungsressourcen
- PERF02-BP05 Dynamisches Skalieren von Datenverarbeitungsressourcen
- PERF02-BP06 Verwenden von optimierten hardwarebasierten Datenverarbeitungsbeschleunigern

PERF02-BP01 Auswählen der besten Datenverarbeitungsoptionen für den Workload

Wenn Sie die für den Workload am besten geeignete Computing-Option auswählen, können Sie die Leistung verbessern, unnötige Infrastrukturkosten reduzieren und den Betriebsaufwand für die Aufrechterhaltung des Workloads senken.

Typische Anti-Muster:

- Sie verwenden dieselbe Option für die Datenverarbeitung, die on-premises verwendet wurde.

- Ihnen fehlt es an Bewusstsein für Cloud-Datenverarbeitungsoptionen, -funktionen und -lösungen und wie diese Lösungen die Datenverarbeitungsleistung verbessern können.
- Sie stellen eine bestehende Datenverarbeitungsoption zu viel bereit, um Skalierungs- oder Leistungsanforderungen zu erfüllen, wenn eine alternative Datenverarbeitungsoption den Workload-Merkmalen besser entsprechen würde.

Vorteile der Nutzung dieser bewährten Methode: Durch die Ermittlung der Anforderungen an die Datenverarbeitung und deren Bewertung anhand der verfügbaren Optionen können Sie den Workload ressourceneffizienter gestalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Zur Optimierung der Cloud-Workloads im Hinblick auf Leistungseffizienz ist es wichtig, die am besten geeigneten Datenverarbeitungsoptionen für Ihren Anwendungsfall und Ihre Leistungsanforderungen auszuwählen. AWS bietet eine Vielzahl von Datenverarbeitungsoptionen, die auf unterschiedliche Workloads in der Cloud zugeschnitten sind. Sie können beispielsweise Amazon EC2 verwenden, um virtuelle Server zu starten und zu verwalten, AWS Lambda, um Code auszuführen, ohne Server bereitstellen oder verwalten zu müssen, Amazon ECS oder Amazon EKS, um Container auszuführen und zu verwalten, oder AWS Batch, um große Datenmengen parallel zu verarbeiten. Basierend auf Ihren Skalierungs- und Datenverarbeitungsanforderungen sollten Sie die optimale Datenverarbeitungslösung für Ihre Situation auswählen und konfigurieren. Sie können auch erwägen, mehrere Arten von Datenverarbeitungslösungen in einem einzigen Workload zu verwenden, da jede ihre eigenen Vor- und Nachteile hat.

Die folgenden Schritte führen Sie durch die Auswahl der richtigen Datenverarbeitungsoptionen, die Ihren Workload-Eigenschaften und Leistungsanforderungen entsprechen.

Implementierungsschritte

- Verstehen Sie Ihre Workload-Datenverarbeitungsanforderungen. Die zu berücksichtigenden wesentlichen Anforderungen umfassen Anforderungen an Datenverarbeitung, Datenverkehrsmuster, Datenzugriffsmuster, Skalierung und Latenz.
- Erfahren Sie mehr über die verschiedenen Datenverarbeitungsoptionen, die für Ihren Workload in AWS verfügbar sind (wie unter PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features beschrieben). Hier finden Sie einige wichtige AWS-Datenverarbeitungsoptionen, ihre Eigenschaften und gängige Anwendungsfälle:

AWS service	Key characteristics	Common use cases
Amazon Elastic Compute Cloud (Amazon EC2)	Has dedicated option for hardware, license requirements, large selection of different instance families, processor types and compute accelerators	Lift and shift migrations, monolithic application, hybrid environments, enterprise applications
Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS)	Easy deployment, consistent environments, scalable	Microservices, hybrid environments
AWS Lambda	Serverlose Datenverarbeitung service that runs code in response to events and automatically manages the underlying compute resources.	Microservices, event-driven applications
AWS Batch	Efficiently and dynamically provisions and scales Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS), and AWS Fargate compute resources, with an option to use On-Demand or Spot Instances based on your job requirements	HPC, train ML models

AWS service	Key characteristics	Common use cases
Amazon Lightsail	Preconfigured Linux and Windows application for running small workloads	Simple web applications, custom website

- Bewerten Sie die Kosten (wie stündliche Gebühr oder Datenübertragung) und den Verwaltungsaufwand (wie Patching und Skalierung), die mit jeder Datenverarbeitungsoption verbunden sind.
- Führen Sie Experimente und Benchmarking in einer Nicht-Produktionsumgebung durch, um herauszufinden, welche Datenverarbeitungsoption Ihre Workload-Anforderungen am besten erfüllt.
- Nachdem Sie experimentiert und die neue Datenverarbeitungslösung ermittelt haben, planen Sie die Migration und überprüfen Sie die Leistungsmetriken.
- Verwenden Sie AWS-Überwachungstools wie Amazon CloudWatch und Optimierungsservices wie AWS Compute Optimizer, um die Computing-Ressourcen kontinuierlich auf der Grundlage realer Nutzungsmuster zu optimieren.

Ressourcen

Zugehörige Dokumente:

- Cloud Computing mit AWS
- Amazon EC2-Instance-Typen
- Amazon EKS-Container: Amazon EKS-Worker-Knoten
- Amazon ECS-Container: Amazon ECS-Container-Instances
- Funktionen: Lambda-Funktionskonfiguration
- Präskriptive Anleitung für Container
- Präskriptive Anleitung für Serverless

Zugehörige Videos:

- AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads
- AWS re:Invent 2023 – Neue generative KI-Funktionen vonAmazon Elastic Compute Cloud in AMS

- AWS re:Invent 2023 – Neuerungen bei Amazon Elastic Compute Cloud
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon Elastic Compute Cloud-Strategien zur Kostenoptimierung
- AWS re:Invent 2021 – Amazon Elastic Compute Cloud der nächsten Generation: Ausführliche Beschreibung des Nitro System
- AWS re:Invent 2019 – Optimieren von Leistung und Kosten für Ihr AWS-Compuntung
- AWS re:Invent 2019 – Amazon Elastic Compute Cloud-Grundlagen
- AWS re:Invent 2022 – ML-Modelle für Inferenz mit hoher Leistung und niedrigen Kosten bereitstellen
- AWS re:Invent 2019 – Optimieren von Leistung und Kosten für Ihr AWS-Computing
- Amazon EC2-Grundlagen
- ML-Modelle für Inferenz mit hoher Leistung und niedrigen Kosten bereitstellen

Zugehörige Beispiele:

- Migration der Webanwendung zu Containern
- Ausführen eines Serverless-„Hello World“
- Amazon EKS-Workshop
- Amazon EC2-Workshop
- Effiziente und belastbare Workloads mit Amazon Elastic Compute Cloud Auto Scaling
- Migration zu AWS Graviton mit Container Services

PERF02-BP02 Verstehen verfügbarer Konfigurationen und Features für die Datenverarbeitung

Informieren Sie sich über die verfügbaren Konfigurationsoptionen und Features für den Datenverarbeitungsservice, damit Sie die richtige Menge an Ressourcen bereitstellen und die Leistungseffizienz verbessern können.

Typische Anti-Muster:

- Sie bewerten keine Datenverarbeitungsoptionen oder verfügbaren Instance-Familien anhand der Workload-Merkmale.
- Sie stellen zu viele Datenverarbeitungsressourcen bereit, um Anforderungen von Nachfragespitzen zu erfüllen.

Vorteile der Nutzung dieser bewährten Methode: Machen Sie sich mit den AWS-Features und -Konfigurationen für die Datenverarbeitung vertraut, sodass Sie eine Datenverarbeitungslösung verwenden können, die für die Workload-Merkmale und -Anforderungen optimiert ist.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Jede Datenverarbeitungslösung verfügt über einzigartige Konfigurationen und Features, um unterschiedliche Workload-Merkmale und -Anforderungen zu unterstützen. Erfahren Sie, wie diese Optionen den Workload ergänzen, und finden Sie heraus, welche Konfigurationsoptionen am besten für Ihre Anwendung geeignet sind. Beispiele für diese Optionen sind Instance-Familien, Größen, Features (GPU, E/A), Bursting, Zeitüberschreitungen, Funktionsgrößen, Container-Instances und Gleichzeitigkeit. Wenn Ihre Workload die gleiche Rechenoption für mehr als vier Wochen verwendet hat und sie davon ausgehen, dass die Eigenschaften in Zukunft gleich bleiben, können Sie mithilfe von AWS Compute Optimizer herausfinden, ob Ihre aktuelle Datenverarbeitungsoption aus CPU- und Speicherebene für die Workloads geeignet ist.

Implementierungsschritte

1. Verstehen Sie die Workload-Anforderungen (wie CPU-Bedarf, Arbeitsspeicher und Latenz).
2. Lesen Sie die AWS-Dokumentation und die bewährten Methoden, um mehr über empfohlene Konfigurationsoptionen zu erfahren, mit denen Sie die Rechenleistung verbessern können. Hier finden Sie einige wichtige Konfigurationsoptionen, die Sie in Betracht ziehen sollten:

Konfigurationsoption	Beispiele
Instance-Typ	• Für die Datenverarbeitung optimiert e Instances eignen sich ideal für Workloads, die ein hohes vCPU-/Arbeitsspeicherverhältnis erfordern. • Arbeitsspeicheroptimierte Instances bieten große Mengen an Arbeitsspeicher, um arbeitsspeicherintensive Workloads zu unterstützen. • Speicheroptimierte Instances wurden für Workloads entworfen, die hohen, sequenzie

Konfigurationsoption	Beispiele
	llen Lese- und Schreibzugriff (IOPS) auf lokalen Speicher erfordern.
Preismodell	• On-Demand-Instances können Sie die Datenverarbeitungskapazität nach Sekunde oder Stunde ohne langfristige Verpflich tungen verwenden. Diese Instances eignen sich für Bursting über die Leistungsbasis hinaus. • Savings Plans bieten erhebliche Einsparun gen gegenüber On-Demand-Instances im Austausch gegen die Verpflichtung, eine bestimmte Menge an Rechenleistung für einen Zeitraum von ein oder drei Jahren zu nutzen. • Spot Instances ermöglichen es Ihnen, ungenutzte Instance-Kapazitäten mit einem Rabatt für Ihre zustandslosen, fehlertol eranten Workloads zu nutzen.
Auto Scaling	Nutzen Sie Auto Scaling Konfiguration zur Anpassung der Datenverarbeitungsressource n an die Datenverkehrsmuster.
Dimensionierung	• Verwenden Sie Compute Optimizer zum Erhalt von Machine-Learning-gestützten Empfehlungen dazu, welche Datenvera rbeitungskonfiguration am besten Ihren Datenverarbeitungsmerkmalen entspricht. • Verwenden Sie AWS Lambda Power Tuning können Sie die beste Konfiguration für Ihre Lambda-Funktion auswählen.

Konfigurationsoption	Beispiele
Hardwarebasierte Computing-Beschleuniger	• Beschleunigte Computing-Instances führen Funktionen wie die Grafikverarbeitung oder Datenmusterzuordnung effizienter aus als CPU-basierte Alternativen. • Nutzen Sie für Machine-Learning-W orkloads spezielle Hardware, die auf Ihren Workload abgestimmt ist, z. B. AWS Trainium, AWS Inferentiaund Amazon EC2 DL1

Ressourcen

Zugehörige Dokumente:

- Cloud Computing mit AWS
- Amazon EC2-Instance-Typen
- Steuerung des Prozessorzustands für Ihre Amazon EC2-Instance
- Amazon EKS-Container: Amazon EKS-Worker-Knoten
- Amazon ECS-Container: Amazon ECS-Container-Instances
- Funktionen: Lambda-Funktionskonfiguration

Zugehörige Videos:

- AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads
- AWS re:Invent 2023 – Neue generative KI-Funktionen von Amazon EC2 in AWS Management Console
- AWS re:Invent 2023 – Neuerungen bei Amazon EC2
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon EC2-Strategien zur Kostenoptimierung
- AWS re:Invent 2021 – Amazon EC2 der neuesten Generation: Ausführliche Beschreibung des Nitro System
- AWS re:Invent 2019 – Amazon EC2-Grundlagen

- AWS re:INVENT 2022 – https://www.youtube.com/watch?v=5B4-s_ivn1o

Zugehörige Beispiele:

- Compute Optimizer-Demo-Code
- Workshop zu Amazon EC2 Spot Instances
- Effiziente und belastbare Workloads mit Amazon EC2 AWS Auto Scaling
- Workshop für Graviton-Entwickler
- AWS für Microsoft-Workloads Immersion Day
- AWS für Linux-Workloads Immersion Day
- AWS Compute Optimizer-Demo-Code
- Amazon EKS-Workshop

PERF02-BP03 Erfassen von Datenverarbeitungsmetriken

Erfassen und verfolgen Sie Datenverarbeitungsmetriken, um die Leistung der Rechenressourcen besser zu verstehen und deren Leistung und Auslastung zu verbessern.

Typische Anti-Muster:

- Sie suchen ausschließlich manuell mithilfe von Protokolldateien nach Metriken.
- Sie verwenden nur die Standardmetriken, die von der Überwachungssoftware aufgezeichnet wurden.
- Sie überprüfen Metriken nur dann, wenn ein Problem vorliegt.

Vorteile der Nutzung dieser bewährten Methode: Die Erfassung von Leistungsmetriken hilft Ihnen dabei, die Anwendungsleistung an den Geschäftsanforderungen auszurichten, um sicherzustellen, dass Sie Ihre Workload-Anforderungen erfüllen. Es kann Ihnen auch dabei helfen, die Ressourcenleistung und -nutzung im Workload kontinuierlich zu verbessern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Cloud-Workloads können große Mengen an Daten generieren, wie Metriken, Protokolle und Ereignisse. In der AWS Cloud ist die Erfassung von Metriken ein entscheidender Schritt zur

Verbesserung von Sicherheit, Kosteneffizienz, Leistung und Nachhaltigkeit. AWS stellt eine Vielzahl von Leistungsmetriken bereit und nutzt dazu Überwachungsservices wie Amazon CloudWatch, um Ihnen wertvolle Einblicke zu bieten. Metriken wie CPU-Nutzung, Arbeitsspeicherauslastung, Datenträger-E/A sowie eingehender und ausgehender Netzwerkverkehr können Einblick in die Nutzung bzw. in Leistungsengpässe bieten. Nutzen Sie diese Metriken im Rahmen eines datengestützten Ansatzes, der Ihnen die aktive Feinabstimmung und Optimierung der vom Workload genutzten Ressourcen ermöglicht. Im Idealfall sollten Sie alle Metriken zu Ihren Datenverarbeitungsressourcen auf einer einzigen Plattform erfassen und Aufbewahrungsrichtlinien implementieren, um Kosten- und Betriebsziele zu unterstützen.

Implementierungsschritte

1. Identifizieren Sie, welche Leistungsmetriken für den Workload relevant sind. Sie sollten Metriken zur Ressourcennutzung und zum Betrieb des Cloud-Workloads (wie Reaktionszeit und Durchsatz) erfassen.
   a. Amazon EC2-Standardmetriken
   b. Amazon ECS-Standardmetriken
   c. Amazon EKS-Standardmetriken
   d. Lambda-Standardmetriken
   e. Amazon EC2-Arbeitsspeicher- und -Datenträgermetriken
2. Wählen Sie die richtige Protokollierungs- und Überwachungslösung für den Workload aus und richten Sie sie ein.
   a. AWS-native Beobachtbarkeit
   b. AWS Distro for OpenTelemetry
   c. Amazon Managed Service for Prometheus
3. Definieren Sie den erforderlichen Filter und die erforderliche Aggregation für die Metriken auf der Grundlage Ihrer Workload-Anforderungen.
   a. Quantifizieren benutzerdefinierter Anwendungsmetriken mit Amazon CloudWatch Logs und Metrikfiltern
   b. Erfassen benutzerdefinierter Metriken mit Amazon CloudWatch und strategischer Markierung
4. Konfigurieren Sie Richtlinien zur Datenaufbewahrung für Ihre Metriken so, dass sie Ihren Sicherheits- und Betriebszielen entsprechen.
   a. Standard-Datenaufbewahrung für CloudWatch-Metriken
   b. Standard-Datenaufbewahrung für CloudWatch Logs

5. Erstellen Sie bei Bedarf Alarme und Benachrichtigungen für Ihre Metriken, damit Sie proaktiv auf leistungsbezogene Probleme reagieren können.
   a. Alarme für benutzerdefinierte Metriken mit der Amazon CloudWatch-Erkennung von Unregelmäßigkeiten erstellen
   b. Metriken und Alarmen für bestimmte Webseiten mit Amazon CloudWatch RUM erstellen
6. Verwenden Sie die Automatisierung, um die Kundendienstmitarbeiter für die Metrik- und Protokollaggregation einzusetzen.
   a. AWS Systems Manager-Automatisierung
   b. OpenTelemetry Collector

Ressourcen

Zugehörige Dokumente:

- Überwachung und Beobachtbarkeit
- Bewährte Methoden: Implementierung der Beobachtbarkeit mit AWS
- Amazon CloudWatch-Dokumentation
- Erfassen von Metriken und Protokollen aus Amazon EC2-Instances und On-Premises-Servern mit dem CloudWatch Agent
- Zugriff auf Amazon CloudWatch Logs für AWS Lambda
- Verwenden von CloudWatch Logs mit Container-Instances
- Veröffentlichen von benutzerdefinierten Metriken
- AWS Answers: Zentralisierte Protokollierung
- CloudWatch-Services, die AWS-Metriken veröffentlichen
- Überwachen von Amazon EKS auf AWS Fargate

Zugehörige Videos:

- AWS re:Invent 2023 – [LAUNCH] Anwendungsüberwachung für moderne Workloads
- AWS re:Invent 2023 – Implementierung der Anwendungsbeobachtbarkeit
- AWS re:Invent 2023 – Aufbau einer effektiven Beobachtbarkeitsstrategie
- AWS re:Invent 2023 – Nahtlose Beobachtbarkeit mit AWS Distro for OpenTelemetry
- Verwaltung der Anwendungsleistung in AWS

Zugehörige Beispiele:

- AWS für Linux-Workloads Immersion Day – Amazon CloudWatch
- Überwachung von Clustern und Containern in Amazon ECS
- Überwachung mit Amazon CloudWatch-Dashboards
- Amazon EKS-Workshop

PERF02-BP04 Konfigurieren und richtiges Dimensionieren von Datenverarbeitungsressourcen

Konfigurieren und passen Sie die Größe der Datenverarbeitungsressourcen so an, dass sie den Leistungsanforderungen des Workloads entsprechen, und vermeiden Sie zu wenig oder zu stark ausgelastete Ressourcen.

Typische Anti-Muster:

- Sie ignorieren Ihre Workload-Leistungsanforderungen, was zu über- oder unterdimensionierten Datenverarbeitungsressourcen führt.
- Sie wählen nur die größte oder kleinste verfügbare Instance für alle Workloads aus.
- Sie verwenden nur eine Instance-Familie, um die Verwaltung zu vereinfachen.
- Sie ignorieren Empfehlungen von AWS Cost Explorer oder Compute Optimizer zur richtigen Dimensionierung.
- Sie bewerten den Workload nicht erneut auf die Eignung neuer Instance-Typen.
- Sie zertifizieren nur eine kleine Anzahl von Instance-Konfigurationen für Ihre Organisation.

Vorteile der Nutzung dieser bewährten Methode: Die richtige Dimensionierung der Datenverarbeitungsressourcen gewährleistet einen optimalen Betrieb in der Cloud, indem eine Über- und Unterdimensionierung von Ressourcen vermieden wird. Die richtige Dimensionierung der Datenverarbeitungsressourcen führt in der Regel zu einer besseren Leistung und einem besseren Kundenerlebnis bei gleichzeitiger Senkung der Kosten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die richtige Dimensionierung ermöglicht es Organisationen, ihre Cloud-Infrastruktur effizient und kostengünstig zu betreiben und gleichzeitig ihre Geschäftsanforderungen zu erfüllen. Eine zu hohe Bereitstellung von Cloud-Ressourcen kann zu zusätzlichen Kosten führen,

während eine unzureichende Bereitstellung zu einer schlechten Leistung und einem negativen Kundenerlebnis führen kann. AWS bietet Tools wie AWS Compute Optimizer und AWS Trusted Advisor, die historische Daten verwenden, um Empfehlungen zur richtigen Dimensionierung Ihrer Rechenressourcen abzugeben.

Implementierungsschritte

- Wählen Sie eine Instance, die am besten zu Ihren Anforderungen passt:
  - Wie wähle ich einen geeigneten Amazon EC2-Instance-Typ für meinen Workload aus?
  - Attributbasierte Auswahl des Instance-Typs für die Amazon EC2 Flotte
  - Erstellen Sie eine Auto Scaling-Gruppe unter Verwendung einer attributbasierten Auswahl des Instance-Typs.
  - Optimieren Ihrer Kubernetes-Datenverarbeitungskosten mit der Karpenter-Konsolidierung
- Analysieren Sie die verschiedenen Leistungsmerkmale Ihrer Workload und bewerten Sie, wie sich diese auf Arbeitsspeicher, Netzwerk und CPU-Auslastung auswirken. Wählen Sie anhand dieser Daten die für das Profil und die Leistungsziele des Workloads am besten geeigneten Ressourcen aus.
- Überwachen Sie Ihren Ressourcenverbrauch mithilfe von AWS-Überwachungstools wie Amazon CloudWatch.
- Wählen Sie die richtige Konfiguration für die Datenverarbeitungsressource aus.
  - Prüfen Sie für kurz andauernde Workloads Amazon CloudWatch-Instance-Metriken wie die `CPUUtilization` um festzustellen, ob die Instance zu wenig oder zu stark ausgelastet ist.
  - Prüfen Sie für stabile Workloads in regelmäßigen Intervallen AWS-Dimensionierungstools wie etwa AWS Compute Optimizer und AWS Trusted Advisor, um Möglichkeiten zur Optimierung und zur korrekten Dimensionierung der Datenverarbeitungsressource zu erkennen.
- Testen Sie Konfigurationsänderungen in einer Nicht-Produktionsumgebung, bevor Sie sie in einer Live-Umgebung implementieren.
- Bewerten Sie neue Datenverarbeitungsangebote und vergleichen Sie sie mit den Anforderungen Ihres Workloads.

Ressourcen

Zugehörige Dokumente:

- Cloud Computing mit AWS

- Amazon EC2-Instance-Typen
- Amazon ECS-Container: Amazon ECS-Container-Instances
- Amazon EKS-Container: Amazon EKS-Worker-Knoten
- Funktionen: Lambda-Funktionskonfiguration
- Steuerung des Prozessorzustands für Ihre Amazon EC2-Instance

Zugehörige Videos:

- Amazon EC2-Grundlagen
- AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads
- AWS re:Invent 2023 – Neue generative KI-Funktionen von Amazon EC2 in AWS Management Console
- AWS re:Invent 2023 – Neuerungen bei Amazon EC2
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon EC2-Strategien zur Kostenoptimierung
- AWS re:Invent 2021 – Amazon EC2 der neuesten Generation: Ausführliche Beschreibung des Nitro System
- AWS re:Invent 2019 – Amazon EC2-Grundlagen

Zugehörige Beispiele:

- AWS Compute Optimizer-Demo-Code
- Amazon EKS-Workshop
- Empfehlungen zur Dimensionierung

PERF02-BP05 Dynamisches Skalieren von Datenverarbeitungsressourcen

Nutzen Sie die Elastizität der Cloud, um die Datenverarbeitungsressourcen dynamisch nach oben oder unten zu skalieren, um Ihren Bedürfnissen zu entsprechen und eine Über- oder Unterdimensionierung von Kapazitäten für den Workload zu vermeiden.

Typische Anti-Muster:

- Sie reagieren auf Alarme, indem Sie die Kapazität manuell erhöhen.

- Sie verwenden dieselben Dimensionierungsrichtlinien (in der Regel statische Infrastruktur) wie bei On-Premises.
- Sie belassen die erhöhte Kapazität nach dem Hochskalieren, anstatt wieder herunterzuskalieren.

Vorteile der Nutzung dieser bewährten Methode: Durch das Konfigurieren und Testen der Elastizität von Rechenressourcen können Sie Geld sparen, Leistungsbenchmarks einhalten und die Zuverlässigkeit verbessern, wenn sich der Datenverkehr ändert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS bietet Ihnen die Flexibilität, Ressourcen dynamisch durch verschiedene Skalierungsmechanismen nach oben oder unten zu skalieren, um Bedarfsänderungen gerecht zu werden. In Kombination mit Datenverarbeitungsmetriken ermöglicht eine dynamische Skalierung Workloads, automatisch auf Änderungen zu reagieren und die optimalen Datenverarbeitungsressourcen zu nutzen, um die Zielvorgabe zu erreichen.

Sie können verschiedene Ansätze nutzen, um das Angebot an Ressourcen auf die Nachfrage abzustimmen.

- Ansatz zur Zielverfolgung: Überwachen Sie Ihre Skalierungsmetriken und erhöhen oder verringern Sie die Kapazität automatisch Ihrem Bedarf entsprechend.
- Vorausschauende Skalierung: Skalieren Sie in Erwartung täglicher und wöchentlicher Trends.
- Zeitplanbasierter Ansatz: Legen Sie Ihren eigenen Skalierungszeitplan entsprechend vorhersehbaren Laständerungen fest.
- Skalierung von Services: Wählen Sie Services (wie Serverless), die auf automatische Skalierung ausgelegt sind.

Sie müssen sicherstellen, dass Workload-Bereitstellungen sowohl Hoch- als auch Herunterskalierungsereignisse verarbeiten können.

Implementierungsschritte

- Datenverarbeitungs-Instances, Container und Funktionen bieten Mechanismen für Elastizität, sei es in Kombination mit AutoScaling oder als Merkmal des Service. Hier finden Sie einige Beispiele für automatische Skalierungsmechanismen:

Autoscaling-Mechanismus	Aktion
Amazon EC2 Auto Scaling	Zur Sicherstellung, dass die richtige Anzahl verfügbarer Amazon EC2 -Instances vorhanden ist, um die Benutzerlast für Ihre Anwendung zu bewältigen.
Application Auto Scaling	Zur automatischen Skalierung der Ressourcen für einzelne AWS-Services über Amazon EC2 hinaus, wie AWS Lambda -Funktionen oder Amazon Elastic Container Service (Amazon ECS) -Services.
Kubernetes Cluster Autoscaler/Karpenter	Zur automatischen Skalierung von Kubernete s-Clustern.

- Das Skalieren wird häufig im Zusammenhang mit Datenverarbeitungsservices wie Amazon EC2-Instances oder AWS Lambda-Funktionen genannt. Denken Sie auch daran, die Konfiguration von nicht Daten verarbeitenden Services in Betracht zu ziehen, z. B. AWS Glue, um die Nachfrage zu decken.
- Stellen Sie sicher, dass die Metriken für die Skalierung den Merkmalen des bereitgestellten Workloads entsprechen. Wenn Sie eine Anwendung zur Video-Transkodierung bereitstellen, wird eine CPU-Auslastung von 100 % erwartet, weshalb dies nicht die Hauptmetrik sein sollte. Verwenden Sie stattdessen die Tiefe der Aufgabenwarteschlange für die Transkodierung. Sie können eine benutzerdefinierte Metrik für Ihre Skalierungsrichtlinie verwenden, falls erforderlich. Beachten Sie zur Auswahl der geeigneten Metriken die folgenden Hinweise zu Amazon EC2:
  - Es sollte sich um eine gültige Nutzungsmetrik handeln, die beschreibt, wie stark eine Instance genutzt wird.
  - Der Metrikwert muss proportional zur Anzahl der Instances in der Auto Scaling-Gruppe steigen oder sinken.
- Vergewissern Sie sich, dass Sie dynamische Skalierung anstelle von manueller Skalierung für Ihre Auto Scaling-Gruppe verwenden. Weiterhin empfehlen wir, dass Sie Zielverfolgungs-Skalierungsrichtlinien für Ihre dynamische Skalierung verwenden.

- Prüfen Sie, ob Workload-Bereitstellungen mit beiden Skalierungen (nach oben und unten) umgehen können. Beispielsweise können Sie den Aktivitätsverlauf verwenden, um eine Skalierungsaktivität für eine Auto Scaling-Gruppe zu verifizieren.
- Evaluieren Sie Ihren Workload auf vorhersagbare Muster und skalieren Sie proaktiv, wenn Sie vorhergesagte und geplante Änderungen der Nachfrage erwarten. Mit der prädiktiven Skalierung können Sie die Notwendigkeit einer Überbereitstellung von Kapazität vermeiden. Weitere Details finden Sie unter Vorausschauende Skalierung mit Amazon EC2 Auto Scaling.

Ressourcen

Zugehörige Dokumente:

- Cloud Computing mit AWS
- Amazon EC2-Instance-Typen
- Amazon ECS-Container: Amazon ECS-Container-Instances
- Amazon EKS-Container: Amazon EKS-Worker-Knoten
- Funktionen: Lambda-Funktionskonfiguration
- Steuerung des Prozessorzustands für Ihre Amazon EC2-Instance
- Ausführliche Beschreibung von Amazon ECS Cluster Auto Scaling
- Vorstellung von Karpenter – Open-Source-Kubernetes-Cluster-Autoscaler mit hoher Leistung

Zugehörige Videos:

- AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads
- AWS re:Invent 2023 – Neue generative KI-Funktionen von Amazon EC2 in der AWS-Managementkonsole
- AWS re:Invent 2023 – Neuerungen bei Amazon EC2
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon EC2-Strategien zur Kostenoptimierung
- AWS re:Invent 2021 – Amazon EC2 der neuesten Generation: Ausführliche Beschreibung des Nitro System
- AWS re:Invent 2019 – Amazon EC2-Grundlagen

Zugehörige Beispiele:

- Amazon EC2 Auto Scaling-Gruppenbeispiele
- Amazon EKS-Workshop
- Skalieren von Amazon EKS-Workloads, durch die Ausführung auf IPv6

PERF02-BP06 Verwenden von optimierten hardwarebasierten Datenverarbeitungsbeschleunigern

Verwenden Sie Hardwarebeschleuniger, um bestimmte Funktionen effizienter auszuführen als CPU-basierte Alternativen.

Typische Anti-Muster:

- Sie haben im Workload keine Benchmark einer universellen Instance verglichen mit einer speziell entwickelten Instance durchgeführt, die eine höhere Leistung und niedrigere Kosten bieten kann.
- Sie verwenden hardwarebasierte Datenverarbeitungsbeschleuniger für Aufgaben, die mithilfe von CPU-basierten Alternativen effizienter sein können.
- Sie überwachen die GPU-Nutzung nicht.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung hardwarebasierter Beschleuniger wie Grafikprozessoren (GPUs) und Field Programmable Gate Arrays (FPGAs) können Sie bestimmte Verarbeitungsfunktionen effizienter ausführen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Beschleunigte Computing-Instances bieten Zugriff auf hardwarebasierte Datenverarbeitungsbeschleuniger wie GPUs und FPGAs. Diese Hardwarebeschleuniger führen bestimmte Funktionen wie die Grafikverarbeitung oder Datenmusterzuordnung effizienter aus als CPU-basierte Alternativen. Viele beschleunigte Workloads, wie Rendering, Transcodierung und Machine Learning, sind sehr variabel im Bezug auf die Ressourcennutzung. Betreiben Sie diese Hardware nur so lange wie nötig und nehmen Sie sie automatisch außer Betrieb, wenn sie nicht mehr benötigt wird, um die allgemeine Leistungseffizienz zu verbessern.

Implementierungsschritte

- Ermitteln Sie, welche beschleunigten Computing-Instances für Ihre Anforderungen geeignet sind.
- Nutzen Sie für Machine-Learning-Workloads spezielle Hardware, die auf Ihren Workload abgestimmt ist, z. B. AWS Trainium, AWS Inferentia oder Amazon EC2 DL1. AWS-Inferentia-

Instances wie Inf2-Instances bieten eine um bis zu 50 % bessere Leistung/Watt als vergleichbare Amazon EC2-Instances.

- Erfassen Sie Nutzungsmetriken für Ihre beschleunigten Computing-Instances. Sie können z. B. den CloudWatch Agent verwenden, um Metriken wie `utilization_gpu` und `utilization_memory` für Ihre GPUs zu erfassen. Dies wird im Artikel zum Erfassen von NVIDIA GPU-Metriken mit Amazon CloudWatch genauer beschrieben.
- Optimieren Sie Code, Netzwerkbetrieb und die Einstellungen von Hardwarebeschleunigern, um sicherzustellen, dass die zugrunde liegende Hardware optimal genutzt wird.
  - Optimieren der GPU-Einstellungen
  - GPU-Überwachung und -Optimierung
  - Optimieren von E/A für die GPU-Leistungsoptimierung von Deep Learning-Training in Amazon SageMaker AI
- Verwenden Sie die aktuellen leistungsstarken Bibliotheken und GPU-Treiber.
- Automatisieren Sie die Freigabe nicht genutzter GPU-Instances.

Ressourcen

Zugehörige Dokumente:

- Arbeiten mit GPUs in Amazon Elastic Container Service
- GPU-Instances
- Instances mit AWS Trainium
- Instances mit AWS Inferentia
- Let’s Architect! Erstellen von Architekturen mit benutzerdefinierten Chips und Beschleunigern

- Accelerated Computing
- Amazon EC2 VT1-Instances
- Wie wähle ich einen geeigneten Amazon EC2 Instance-Typ für meinen Workload aus?
- Auswählen des besten KI-Beschleunigers und der Modellkompilierung für Computer Vision Inference mit Amazon SageMaker AI

Zugehörige Videos:

- AWS re:Invent 2021 – Auswählen von Amazon Elastic Compute Cloud-GPU-Instances für Deep Learning
- AWS re:Invent 2022 – [NEUER LAUNCH] Einführung von AWS-Inferentia2-basierten Amazon EC2-Inf2-Instances
- AWS re:Invent 2022 – Beschleunigung von Deep Learning und schnellere Innovationen mit AWS Trainium
- AWS re:Invent 2022 – Deep Learning in AWS mit NVIDIA: Vom Training bis zur Bereitstellung

Zugehörige Beispiele:

- Amazon SageMaker AI und NVIDIA GPU Cloud (NGC)
- Verwendung von SageMaker AI mit Trainium und Inferentia für optimierte Deep-Learning-Trainings- und Inferenz-Workloads
- Optimierung von NLP-Modellen mit Amazon Elastic Compute Cloud-Inf1-Instances in Amazon SageMaker AI

Datenverwaltung

LEIST 3. Wie speichern und verwalten Sie die Daten in Ihrem Workload und wie greifen Sie darauf zu?

Die optimale Datenverwaltungslösung für ein bestimmtes System hängt vom Datentyp (Block, Datei oder Objekt), den Zugriffsmustern (zufällig oder sequenziell), dem erforderlichen Durchsatz, der Zugriffshäufigkeit (online, offline, Archiv), der Aktualisierungshäufigkeit (WORM, dynamisch) sowie den Verfügbarkeits- und Lebensdaueranforderungen ab. Well-Architected-Workloads verwenden zweckgebundene Daten-Stores, die verschiedene Funktionen zur Verbesserung der Leistung ermöglichen.

Bewährte Methoden

- PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt
- PERF03-BP02 Bewerten verfügbarer Konfigurationsoptionen für den Datenspeicher
- PERF03-BP03 Erfassen und Aufzeichnen von Metriken zur Datenspeicherleistung
- PERF03-BP04 Implementieren von Strategien zur Verbesserung der Abfrageleistung im Datenspeicher

- PERF03-BP05 Implementieren von Datenzugriffsmustern, die Caching nutzen

PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt

Machen Sie sich mit Datenmerkmalen (wie Freigabe, Größe, Cache-Größe, Zugriffsmuster, Latenz, Durchsatz und Persistenz von Daten) vertraut, um die richtigen, speziell entwickelten Datenspeicher (Speicher oder Datenbank) für den Workload auszuwählen.

Typische Anti-Muster:

- Sie halten an einem Datenspeicher fest, da es interne Erfahrungen und Wissen über eine bestimmte Datenbanklösung gibt.
- Sie gehen davon aus, dass für alle Workloads ähnliche Datenspeicher- und Zugriffsanforderungen gelten.
- Sie haben keinen Datenkatalog zur Inventarisierung Ihrer Datenbestände eingeführt.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie die Datenmerkmale und -anforderungen unterbewerten, können Sie die effizienteste und leistungsfähigste Speichertechnologie ermitteln, die für Ihre Workload-Anforderungen geeignet ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Stellen Sie bei der Auswahl und Implementierung von Datenspeicher sicher, dass die Abfrage-, Skalierungs- und Speichermerkmale die Workload-Datenanforderungen unterstützen. AWS bietet zahlreiche Datenspeicher- und Datenbanktechnologien, darunter Blockspeicher, Objektspeicher, Streaming-Speicher, Dateisystem-, relationale, Schlüsselwert-, Dokument-, In-Memory-, Graph-, Zeitreihen- und Ledger-Datenbanken. Jede Datenverwaltungslösung hat verfügbare Optionen und Konfigurationen, um Ihre Anwendungsfälle und Datenmodelle zu unterstützen. Wenn Sie die Merkmale und Anforderungen der Daten verstehen, können Sie sich von monolithischer Speichertechnologie und restriktiven Einheitsansätzen lösen und sich auf eine angemessene Datenverwaltung konzentrieren.

Implementierungsschritte

- Führen Sie eine Bestandsaufnahme der verschiedenen Datentypen durch, die in Ihrem Workload vorhanden sind.

- Verstehen und dokumentieren Sie Datenmerkmale und -anforderungen, einschließlich:
  - Datentyp (strukturiert, semistrukturiert, relational)
  - Datenvolumen und -wachstum
  - Lebensdauer von Daten: anhaltend, flüchtig, vorübergehend
  - Anforderungen an AKID (Atomarität, Konsistenz, Isolation, Dauerhaftigkeit)
  - Datenzugriffsmuster (leseintensiv oder schreibintensiv)
  - Latenz
  - Durchsatz
  - IOPS (Eingabe-/Ausgabevorgänge pro Sekunde)
  - Aufbewahrungsfrist der Daten
- Erfahren Sie mehr über die verschiedenen Datenspeicher (Speicher- und Datenbank-Services), die für Ihren Workload in AWS verfügbar sind und Ihre Datenmerkmale erfüllen können (wie beschrieben unter PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features. Einige Beispiele für AWS-Speichertechnologien und ihre Schlüsselmerkmale sind:

Typ	AWS-Services	Schlüsselmerkmale
Object storage	Amazon S3	Unlimited scalability, high availability, and multiple options for accessibility. Transferring and accessing objects in and out of Amazon S3 can use a service, such as Transfer Accelerat ion or Zugriffspunkte, to support your location, security needs, and access patterns.
Archiving storage	Amazon S3 Glacier	Built for data archiving.
Streaming storage	Amazon Kinesis Amazon Managed Streaming for Apache Kafka (Amazon MSK)	Efficient ingestion and storage of streaming data.

Typ	AWS-Services	Schlüsselmerkmale
Shared file system	Amazon Elastic File System (Amazon EFS)	Bereitstellbares Dateisystem, auf das mehrere Arten von Datenverarbeitungslösungen zugreifen können.
Shared file system	Amazon FSx	Built on the latest AWS compute solutions to support four commonly used file systems: NetApp ONTAP, OpenZFS, Windows File Server, and Lustre. Amazon FSx variieren vary per file system and should be considered when selecting the right file system for your workload needs.
Block storage	Amazon Elastic Block Store (Amazon EBS)	Scalable, high-performance block-storage service designed for Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS includes SSD-backed storage for transactional, IOPS-intensive workloads and HDD-backed storage for throughput-intensi ve workloads.

Typ	AWS-Services	Schlüsselmerkmale
Relational database	Amazon Aurora, Amazon RDS, Amazon Redshift.	Designed to support ACID (atomicity, consistency, isolation, durability) transactions, and maintain referential integrity and strong data consistency. Many traditional applications, enterprise resource planning (ERP), customer relationship management (CRM), and ecommerce use relational databases to store their data.
Key-value database	Amazon DynamoDB	Optimized for common access patterns, typically to store and retrieve large volumes of data. High-traffic web apps, ecommerce systems, and gaming applications are typical use-cases for key-value databases.
Document database	Amazon DocumentDB	Designed to store semi-structured data as JSON-like documents. These databases help developers build and update applications such as content management, catalogs, and user profiles quickly.

Typ	AWS-Services	Schlüsselmerkmale
In-memory database	Amazon ElastiCache , Amazon MemoryDB für Redis	Used for applications that require real-time access to data, lowest latency and highest throughput. You may use in-memory databases for application caching, session management, gaming leaderboards, low latency ML feature store, microserv ices messaging system, and a high-throughput streaming mechanism
Graph database	Amazon Neptune	Used for applications that must navigate and query millions of relationships between highly connected graph datasets with milliseco nd latency at large scale. Many companies use graph databases for fraud detection , social networking, and recommendation engines.
Time Series database	Amazon Timestream	Used to efficiently collect, synthesize, and derive insights from data that changes over time. IoT applications, DevOps, and industrial telemetry can utilize time-series databases.

Typ	AWS-Services	Schlüsselmerkmale
Wide column	Amazon Keyspaces (für Apache Cassandra)	Uses tables, rows, and columns, but unlike a relational database, the names and format of the columns can vary from row to row in the same table. You typically see a wide column store in high scale industrial apps for equipment maintenance, fleet management, and route optimization.
Ledger	Amazon Quantum Ledger Database (Amazon QLDB)	Provides a centralized and trusted authority to maintain a scalable, immutable, and cryptographically verifiabl e record of transactions for every application. We see ledger databases used for systems of record, supply chain, registrations, and even banking transactions.

- Wenn Sie eine Datenplattform aufbauen, nutzen Sie moderne Datenarchitektur in AWS, um Ihren Data Lake, Ihr Data Warehouse und Ihre speziell entwickelten Datenspeicher zu integrieren.
- Die wichtigsten Fragen, die Sie bei der Auswahl eines Datenspeichers für Ihren Workload berücksichtigen müssen, lauten wie folgt:

Question	Things to consider
How is the data structured?	• Wenn die Daten nicht strukturiert sind, erwägen Sie einen Objektspeicher wie Amazon S3 oder eine NoSQL-Datenbank wie Amazon DocumentDB.

Question	Things to consider
	• Erwägen Sie für Schlüssel-Werte-Daten DynamoDB, Amazon ElastiCache (Redis OSS) oder Amazon MemoryDB
What level of referential integrity is required?	• Bei Fremdschlüsseleinschränkungen können relationale Datenbanken wie Amazon RDS und Aurora diese Integritä tsebene bieten. • Üblicherweise würden Sie innerhalb eines NoSQL-Datenmodells Ihre Daten in ein einzelnes Dokument oder eine Sammlung von Dokumenten denormalisieren, die in einer einzelnen Anfrage abgerufen werden können, anstatt Daten in Dokumenten oder Tabellen zusammenzufügen.
Is ACID (atomicity, consistency, isolation, durability) compliance required?	• Wenn mit relationalen Datenbanken zusammenhängende AKID-Eigenschaften erforderlich sind, erwägen Sie eine relationa le Datenbank wie Amazon RDS und Aurora. • Wenn strikte Konsistenz für eine NoSQL-Datenbank erforderlich ist, können Sie strikt konsistente Lesevorgänge mithilfe von DynamoDB verwenden.
How will the storage requirements change over time? How does this impact scalability?	• Serverless-Datenbanken wie DynamoDB und Amazon Quantum Ledger Database (Amazon QLDB) skalieren dynamisch. • Relationale Datenbanken haben oftmals Obergrenzen bei bereitgestelltem Speicher und müssen mithilfe von Mechanismen wie Sharding horizontal partitioniert werden, sobald sie diese Grenzen erreicht haben.

Question	Things to consider
What is the proportion of read queries in relation to write queries? Would caching be likely to improve performance?	• Leseintensive Workloads können von einer Caching-Ebene wie ElastiCache oder DAX profitieren, wenn es sich bei der Datenbank um DynamoDB handelt. • Lesevorgänge können auch zu Read Replicas mit relationalen Datenbanken wie Amazon RDS ausgelagert werden.
Does storage and modification (OLTP - Online Transaction Processing) or retrieval and reporting (OLAP - Online Analytical Processing) have a higher priority?	• Erwägen Sie für Read-as-is-Transaktionsverarbeitung mit hohem Durchsatz eine NoSQL-Datenbank wie DynamoDB. • Verwenden Sie Amazon RDS für hohen Durchsatz und komplexe Lesemuster (wie Join) mit Konsistenz. • Erwägen Sie für analytische Abfragen eine spaltenbasierte Datenbank wie Amazon Redshift oder das Exportieren von Daten zu Amazon S3 und das Durchführen von Analysen mithilfe von Athena oder Amazon QuickSight.

Question	Things to consider
What level of durability does the data require?	• Aurora repliziert Ihre Daten automatisch in drei Availability Zones innerhalb von einer Region, was bedeutet, dass Ihre Daten hochbeständig sind und eine geringere Wahrscheinlichkeit von Datenverlust besteht. • DynamoDB wird automatisch in mehreren Availability Zones repliziert und bietet hohe Verfügbarkeit und Datenstabilität. • Amazon S3 bietet eine Langlebigkeit mit 11 Neunen. Viele Datenbankservices wie Amazon RDS und DynamoDB unterstützen das Exportieren von Daten zu Amazon S3 für Langzeitaufbewahrung und Archivierung.
Is there a desire to move away from commercial database engines or licensing costs?	• Ziehen Sie Open-Source-Engines wie PostgreSQL und MySQL auf Amazon RDS oder Aurora in Erwägung. • Nutzen Sie AWS Database Migration Service und AWS Schema Conversion Tool zum Migrieren von kommerziellen Datenbank-Engines zu Open-Source-Lösungen.
What is the operational expectation for the database? Is moving to managed services a primary concern?	• Das Verwenden von Amazon RDS anstatt von Amazon EC2 und DynamoDB oder Amazon DocumentDB anstatt eine NoSQL-Datenbank selbst zu hosten, kann den Betriebsaufwand verringern.

Question	Things to consider
How is the database currently accessed? Is it only application access, or are there business intelligence (BI) users and other connected off-the-shelf applications?	• Wenn Sie von externen Tools abhängig sind, müssen Sie möglicherweise mit der Datenbank, die unterstützt wird, die Kompatibilität aufrecht erhalten. Amazon RDS ist vollständig kompatibel mit den unterschiedlichen Engine-Versionen, die unterstützt werden, einschließlich Microsoft SQL Server, Oracle, MySQL und PostgreSQL.

- Führen Sie Experimente und Benchmarking in einer Nicht-Produktionsumgebung durch, um herauszufinden, welcher Datenspeicher Ihre Workload-Anforderungen erfüllen kann.

Ressourcen

Zugehörige Dokumente:

- Amazon EBS-Volume-Typen
- Amazon EC2-Speicher
- Amazon EFS: Leistung von Amazon EFS
- Leistung von Amazon FSx for Lustre
- Leistung von Amazon FSx for Windows File Server
- Amazon S3 Glacier: S3 Glacier-Dokumentation
- Amazon S3: Überlegungen zu Anfragerate und Leistung
- Cloud-Speicher mit AWS
- Amazon EBS-E/A-Merkmale
- Cloud-Datenbanken mit AWS
- AWS-Datenbank-Caching
- DynamoDB Accelerator
- Bewährte Methoden für Amazon Aurora
- Leistung von Amazon Redshift
- Die besten 10 Leistungstipps für Amazon Athena

- Bewährte Methoden für Amazon Redshift Spectrum
- Bewährte Methoden für Amazon DynamoDB
- Wählen Sie zwischen Amazon EC2 und Amazon RDS
- Bewährte Methoden für die Implementierung von Amazon ElastiCache

Zugehörige Videos:

- AWS re:Invent 2023 – Steigerung der Effizienz von Amazon Elastic Block Store und der allgemeinen Kosteneffizienz
- AWS re:Invent 2023 – Optimierung der Speicherkosten und der Leistung mit Amazon Simple Storage Service
- AWS re:Invent 2023 – Aufbau und Optimierung eines Data Lake in Amazon Simple Storage Service
- AWS re:Invent 2022: Erstellen von modernen Datenarchitekturen in AWS
- AWS re:Invent 2022 – Aufbau von Data-Mesh-Architekturen in AWS
- AWS re:Invent 2023 – Vertiefung in Amazon Aurora und seine Innovationen
- AWS re:Invent 2023: Fortschrittliche Datenmodellierung mit Amazon DynamoDB
- AWS re:Invent 2022: Modernisierung von Apps mit speziell entwickelten Datenbanken
- Vertiefung in Amazon DynamoDB: Fortschrittliche Entwurfsmuster

Zugehörige Beispiele:

- AWS-Workshop „Speziell entwickelte Datenbanken“
- Datenbanken für Entwickler
- AWS Immersion Day in moderne Datenarchitekturen
- Erstellung eines Data Mesh in AWS
- Amazon S3-Beispiele
- Optimierung von Datenmustern mithilfe von Amazon Redshift Data Sharing
- Datenbankmigrationen
- MS SQL Server – AWS Database Migration Service (AWS DMS)-Replikationsdemo
- Praktischer Workshop für die Datenbankmodernisierung
- Amazon Neptune-Beispiele

PERF03-BP02 Bewerten verfügbarer Konfigurationsoptionen für den Datenspeicher

Machen Sie sich mit den verschiedenen Funktionen und Konfigurationsoptionen vertraut, die für Ihre Datenspeicher verfügbar sind, und bewerten Sie sie, um Speicherplatz und Leistung für Ihren Workload zu optimieren.

Typische Anti-Muster:

- Sie verwenden nur einen Speichertyp, z. B. Amazon EBS, für alle Workloads.
- Sie verwenden bereitgestellte IOPS für alle Workloads, ohne reale Tests auf allen Speicherebenen durchzuführen.
- Ihnen fehlt das Bewusstsein für die Wahl der Konfigurationsoptionen der Datenverwaltungslösung.
- Sie verlassen sich ausschließlich auf das Vergrößern der Instance-Größe, ohne andere verfügbare Konfigurationsoptionen in Betracht zu ziehen.
- Sie testen die Skalierungsoptionen Ihres Datenspeichers nicht.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie Datenspeicherkonfigurationen erkunden und mit ihnen experimentieren, können Sie möglicherweise Infrastrukturkosten senken, die Leistung verbessern und den Aufwand zur Verwaltung Ihrer Workloads verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Für einen Workload können je nach Datenspeicher- und Zugriffsanforderungen ein oder mehrere Datenspeicher verwendet werden. Zur Optimierung der Leistungseffizienz und Kosten müssen Sie Datenzugriffsmuster auswerten, um die entsprechenden Datenspeicherkonfigurationen zu bestimmen. Während Sie die Datenspeicheroptionen erkunden, sollten Sie unterschiedliche Aspekte in Betracht ziehen. Dazu zählen Speicheroptionen, Arbeitsspeicher, Rechenvorgänge, Read Replica, Konsistenzanforderungen, Verbindungs-Pooling und Caching-Optionen. Experimentieren Sie mit diesen unterschiedlichen Konfigurationsoptionen, um Metriken zur Leistungseffizienz zu verbessern.

Implementierungsschritte

- Verstehen Sie die aktuellen Konfigurationen (wie Instance-Typ, Speichergröße oder Version der Datenbank-Engine) des Datenspeichers.
- Lesen Sie die AWS-Dokumentation und die bewährten Methoden, um mehr über empfohlene Konfigurationsoptionen zu erfahren, mit denen Sie die Leistung für den Datenspeicher verbessern

können. Die wichtigsten Datenspeicheroptionen, die Sie in Betracht ziehen sollten, sind die folgenden:

Configuration option	Examples
Offloading reads (like read replicas and caching)	• Bei DynamoDB-Tabellen können Sie Lesevorgänge mithilfe von DAX für Caching auslagern. • Sie können einen Amazon ElastiCache (Redis OSS)-Cluster erstellen und Ihre Anwendung so konfigurieren, dass sie zuerst aus dem Cache liest und dann auf die Datenbank zurückfällt, wenn das angeforderte Element nicht vorhanden ist. • Relationale Datenbanken wie Amazon RDS und Aurora sowie bereitgestellte NoSQL-Datenbanken wie Neptune und Amazon DocumentDB unterstützen alle das Hinzufügen von Read Replicas, um die Lesevorgänge des Workloads auszulagern. • Serverless-Datenbanken wie DynamoDB skalieren automatisch. Stellen Sie sicher, dass Sie ausreichend Read Capacity Units (RCU) bereitstellen, um den Workload zu verarbeiten.

Configuration option	Examples
Scaling writes (like partition key sharding or introducing a queue)	• Bei relationalen Datenbanken können Sie die Größe der Instance erhöhen, um einen erhöhten Workload zu bewältigen, oder die bereitgestellten IOPs erhöhen, um einen erhöhten Durchsatz in den zugrunde liegenden Speicher zu ermöglichen. • Sie können vor Ihrer Datenbank auch eine Warteschlange einrichten, anstatt direkt in die Datenbank zu schreiben. Mithilfe dieses Musters können Sie die Datenerfassung von der Datenbank entkoppeln und die Flow-Rate steuern, sodass die Datenbank nicht überwältigt wird. • Das Batching Ihrer Schreibanforderungen, anstatt mehrere kurzlebige Transaktionen zu erstellen, kann Ihnen dabei helfen, den Durchsatz bei relationalen Datenbanken mit hohem Schreibvolumen zu verbessern. • Serverless-Datenbanken wie DynamoDB können den Schreibdurchsatz automatisch skalieren oder indem die bereitgestellten Kapazitätseinheiten für Schreibvorgänge (Write Capacity Units, WCU) abhängig vom Kapazitätsmodus angepasst werden. • Es können immer noch Probleme mit heißen Partitionen auftreten, wenn Sie die Durchsatzgrenzen für einen bestimmten Partitionsschlüssel erreichen. Dies kann verhindert werden, indem Sie einen Partition sschlüssel auswählen, der gleichmäßiger verteilt ist, oder indem Sie die Schreibvo rgänge des Partitionsschlüssels in Shards aufteilen.

Configuration option	Examples
Policies to manage the lifecycle of your datasets	• Mit Amazon S3-Lebenszyklen können Sie Ihre Objekte während ihres gesamten Lebenszyklus verwalten. Wenn die Zugriffsmuster unbekannt oder nicht prognostizierbar sind oder sich ändern, können Sie Amazon S3 Intelligent-Tiering verwenden. Hiermit werden Zugriffsmuster überwacht und Objekte, auf die nicht zugegriffen wurde, automatisch in kostengünstigere Zugriffsebenen verschoben. Anhand von Amazon S3-Storage-Lens-Metriken können Sie Optimierungsmöglichkeiten und Lücken im Lebenszyklusmanagement ermitteln. • Das Amazon EFS-Lebenszyklusmanagement verwaltet den Dateispeicher für Ihre Dateisysteme automatisch.
Connection management and pooling	• Amazon RDS Proxy kann mit Amazon RDS und Aurora verwendet werden, um Verbindungen mit der Datenbank zu verwalten. • Serverless-Datenbanken wie DynamoDB haben keine ihnen zugewiesenen Verbindungen, aber ziehen Sie die bereitgestellte Kapazität sowie automatische Skalierungsrichtlinien in Betracht, um Datenverkehrsspitzen zu bewältigen.

- Führen Sie Experimente und Benchmarking in einer Nicht-Produktionsumgebung durch, um herauszufinden, welche Konfigurationsoption Ihre Workload-Anforderungen erfüllen kann.
- Nachdem Sie experimentiert haben, planen Sie die Migration und überprüfen Sie die Leistungsmetriken.

- Verwenden Sie AWS-Tools zur Überwachung (wie Amazon CloudWatch) und Optimierung (wie Amazon S3 Storage Lens), um den Datenspeicher kontinuierlich anhand realer Nutzungsmuster zu optimieren.

Ressourcen

Zugehörige Dokumente:

- Cloud-Speicher mit AWS
- Amazon EBS-Volume-Typen
- Amazon EC2-Speicher
- Amazon EFS: Leistung von Amazon EFS
- Leistung von Amazon FSx for Lustre
- Leistung von Amazon FSx for Windows File Server
- Amazon S3 Glacier: S3 Glacier-Dokumentation
- Amazon S3: Überlegungen zu Anfragerate und Leistung
- Amazon EBS-E/A-Merkmale
- Cloud-Datenbanken mit AWS
- AWS-Datenbank-Caching
- DynamoDB Accelerator
- Bewährte Methoden für Amazon Aurora
- Leistung von Amazon Redshift
- Die besten 10 Leistungstipps für Amazon Athena
- Bewährte Methoden für Amazon Redshift Spectrum
- Bewährte Methoden für Amazon DynamoDB

Zugehörige Videos:

- AWS re:Invent 2023 – Steigerung der Effizienz von Amazon Elastic Block Store und der allgemeinen Kosteneffizienz
- AWS re:Invent 2023 – Optimierung der Speicherkosten und der Leistung mit Amazon Simple Storage Service

- AWS re:Invent 2023 – Aufbau und Optimierung eines Data Lake in Amazon Simple Storage Service
- AWS re:Invent 2023 – Neuerungen bei AWS-Dateispeicher
- AWS re:Invent 2023 – Vertiefung in Amazon DynamoDB

Zugehörige Beispiele:

- AWS-Workshop „Speziell entwickelte Datenbanken“
- Datenbanken für Entwickler
- AWS Immersion Day in moderne Datenarchitekturen
- Amazon EBS – automatische Skalierung
- Amazon S3-Beispiele
- Amazon DynamoDB-Beispiele
- Beispiele von AWS-Datenbankmigration
- Workshop für die Datenbankmodernisierung
- Arbeiten mit Parametern auf Ihrem Amazon RDS für Postgress DB

PERF03-BP03 Erfassen und Aufzeichnen von Metriken zur Datenspeicherleistung

Verfolgen und zeichnen Sie relevante Leistungsmetriken für Ihren Datenspeicher auf, um zu verstehen, wie Ihre Datenverwaltungslösungen funktionieren. Mithilfe dieser Metriken können Sie Ihren Datenspeicher optimieren, überprüfen, ob Ihre Workload-Anforderungen erfüllt werden, und sich einen klaren Überblick über die Workload-Leistung verschaffen.

Typische Anti-Muster:

- Sie suchen ausschließlich manuell mithilfe von Protokolldateien nach Metriken.
- Sie veröffentlichen Metriken nur in internen Tools, die von Ihrem Team verwendet werden, und Sie haben kein umfassendes Bild Ihres Workloads.
- Sie verwenden nur die Standardmetriken, die von der Überwachungssoftware Ihrer Wahl aufgezeichnet wurden.
- Sie überprüfen Metriken nur dann, wenn ein Problem vorliegt.
- Sie überwachen Metriken nur auf Systemebene und erfassen keine Datenzugriffs- und Nutzungsmetriken.

Vorteile der Nutzung dieser bewährten Methode: Das Einrichten einer Leistungsbasislinie hilft Ihnen dabei, das normale Verhalten und die Anforderungen von Workloads zu verstehen. Abnorme Muster können schneller identifiziert und behoben werden, was die Leistung und Zuverlässigkeit des Datenspeichers erhöht.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um die Leistung der Datenspeicher zu überwachen, müssen Sie mehrere Leistungsmetriken über einen bestimmten Zeitraum aufzeichnen. Auf diese Weise können Sie Anomalien erkennen und die Leistung anhand von Geschäftsmetriken messen, um sicherzustellen, dass Sie die Anforderungen Ihres Workloads erfüllen.

Metriken sollten das zugrunde liegende System, das den Datenspeicher unterstützt, sowie die Datenbankmetriken enthalten. Die Metriken des zugrunde liegenden Systems können die CPU-Auslastung, den Arbeitsspeicher, den verfügbaren Festplattenspeicher, Festplatten-E/A, das Cache-Trefferverhältnis und Metriken zum eingehenden und ausgehenden Netzwerkdatenverkehr umfassen, während die Datenspeichermetriken die Transaktionen pro Sekunde, die häufigsten Abfragen, die durchschnittlichen Abfrageraten, Antwortzeiten, die Indexauslastung, Tabellenschlösser, Abfragezeitüberschreitungen und die Anzahl offener Verbindungen enthält. Diese Daten sind von entscheidender Bedeutung, um festzustellen, wie leistungsfähig der Workload ist und wie die Datenverwaltungslösung genutzt wird. Nutzen Sie diese Metriken im Rahmen eines datengestützten Ansatzes, der Ihnen die Feinabstimmung und Optimierung der vom Workload genutzten Ressourcen ermöglicht.

Nutzen Sie Tools, Bibliotheken und Systeme zum Aufzeichnen von Messungen zur Datenbankleistung.

Implementierungsschritte

1. Identifizieren Sie die wichtigsten Leistungsmetriken, die der Datenspeicher verfolgen soll.
   a. Metriken und Dimensionen von Amazon S3
   b. Überwachungsmetriken für innerhalb einer Amazon RDS-Instance
   c. Überwachen der DB-Last mit Performance Insights auf Amazon RDS
   d. Überblick über Erweiterte Überwachung
   e. Metriken und Dimensionen von DynamoDB
   f. Überwachen von DynamoDB Accelerator

   g. Überwachen von Amazon MemoryDB mit Amazon CloudWatch
   h. Welche Metriken sollte ich überwachen?
   i. Überwachen der Amazon Redshift-Cluster-Leistung
   j. Metriken und Dimensionen von Timestream
   k. Amazon CloudWatch-Metriken für Amazon Aurora
   l. Protokollieren und Überwachen von Amazon Keyspaces (for Apache Cassandra)
   m. Überwachen von Amazon Neptune-Ressourcen
2. Verwenden Sie eine zugelassene Protokollierungs- und Überwachungslösung, um diese Metriken zu erfassen. Amazon CloudWatch lassen sich Metriken aus sämtlichen Ressourcen Ihrer Architektur erfassen. Sie können auch benutzerdefinierte Metriken erfassen und in Oberflächen-, Geschäfts- oder abgeleiteten Metriken veröffentlichen. Richten Sie mit CloudWatch oder mit Lösungen von Drittanbietern Alarme ein, die auf das Überschreiten von Schwellenwerten hinweisen.
3. Prüfen Sie, ob die Datenspeicherüberwachung von einer Machine-Learning-Lösung profitieren kann, die Leistungsanomalien erkennt.
   a. Amazon DevOps Guru für Amazon RDS ermöglicht einen Einblick in Leistungsprobleme und bietet Empfehlungen für Korrekturmaßnahmen.
4. Konfigurieren Sie die Datenaufbewahrung in Ihrer Überwachungs- und Protokollierungslösung so, dass sie Ihren Sicherheits- und Betriebszielen entspricht.
   a. Standard-Datenaufbewahrung für CloudWatch-Metriken
   b. Standard-Datenaufbewahrung für CloudWatch Logs

Ressourcen

Zugehörige Dokumente:

- AWS-Datenbank-Caching
- Die besten 10 Leistungstipps für Amazon Athena
- Bewährte Methoden für Amazon Aurora
- DynamoDB Accelerator
- Bewährte Methoden für Amazon DynamoDB
- Bewährte Methoden für Amazon Redshift Spectrum
- Amazon Redshift-Leistung

- Cloud-Datenbanken mit AWS
- Amazon RDS Performance Insights

Zugehörige Videos:

- AWS re:Invent 2022 – Leistungsüberwachung mit Amazon RDS und Aurora, mit Autodesk
- Überwachung und Optimierung der Datenbankleistung mit Amazon DevOps Guru für Amazon RDS
- AWS re:Invent 2023 – Neuerungen bei AWS-Dateispeicher
- AWS re:Invent 2023 – Ausführliche Beschreibung von Amazon DynamoDB
- AWS re:Invent 2023 – Aufbau und Optimierung eines Data Lake in Amazon S3
- AWS re:Invent 2023 – Neuerungen bei AWS-Dateispeicher
- AWS re:Invent 2023 – Ausführliche Beschreibung von Amazon DynamoDB
- Bewährte Methoden für die Überwachung von Redis-Workloads auf Amazon ElastiCache

Zugehörige Beispiele:

- Framework zur AWS-Datensatzerfassung und Sammlung von Metriken
- Workshop zur Überwachung von Amazon RDS
- AWS-Workshop „Speziell entwickelte Datenbanken“

PERF03-BP04 Implementieren von Strategien zur Verbesserung der Abfrageleistung im Datenspeicher

Implementieren Sie Strategien zur Datenoptimierung und Verbesserung der Datenabfrage, um mehr Skalierbarkeit und eine effizientere Leistung für Ihre Workloads zu erzielen.

Typische Anti-Muster:

- Sie partitionieren keine Daten in Ihrem Datenspeicher.
- Sie speichern Daten in nur einem Dateiformat in Ihrem Datenspeicher.
- Sie verwenden keine Indizes in Ihrem Datenspeicher.

Vorteile der Nutzung dieser bewährten Methode: Die Optimierung der Daten- und Abfrageleistung führt zu mehr Effizienz, niedrigeren Kosten und einer verbesserten Benutzererfahrung.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Daten- und Abfrageoptimierung sind wichtige Aspekte der Leistungseffizienz in einem Datenspeicher, da sie sich auf die Leistung und Reaktionsfähigkeit des gesamten Cloud-Workloads auswirken. Nicht optimierte Abfragen können zu einem höheren Ressourcenverbrauch und Engpässen führen, wodurch die Gesamteffizienz eines Datenspeichers beeinträchtigt wird.

Die Datenoptimierung umfasst mehrere Techniken, um eine effiziente Datenspeicherung und einen effizienten Datenzugriff zu gewährleisten. Dies trägt auch dazu bei, die Abfrageleistung in einem Datenspeicher zu verbessern. Zu den wichtigsten Strategien gehören Datenpartitionierung, Datenkomprimierung und Datendenormalisierung, mit denen Daten sowohl für die Speicherung als auch für den Zugriff optimiert werden können.

Implementierungsschritte

- Verstehen und analysieren Sie die kritischen Datenabfragen, die in Ihrem Datenspeicher durchgeführt werden.
- Identifizieren Sie die langsamen Abfragen in Ihrem Datenspeicher und verwenden Sie Abfragepläne, um den aktuellen Status zu verstehen.
  - Analysieren des Abfrageplans in Amazon Redshift
  - Verwenden von EXPLAIN und EXPLAIN ANALYZE in Athena
- Implementieren Sie Strategien zur Verbesserung der Abfrageleistung. Einige der wichtigsten Strategien sind:
  - Verwenden eines spaltenförmigen Dateiformats (wie Parquet oder ORC).
  - Komprimieren von Daten im Datenspeicher, um Speicherplatz und E/A-Betrieb zu reduzieren.
  - Datenpartitionierung zur Aufteilung von Daten in kleinere Teile und zur Reduzierung der Zeit für das Scannen von Daten.
    - Partitionierung von Daten in Athena
    - Partitionen und Datenverteilung
  - Datenindizierung für die gemeinsamen Spalten in der Abfrage.
  - Verwenden Sie materialisierte Ansichten für häufige Abfragen.
    - Verstehen von materialisierten Ansichten
    - Erstellen von materialisierten Ansichten in Amazon Redshift

  - Wählen Sie den richtigen Verknüpfungsvorgang für die Abfrage aus. Wenn Sie zwei Tabellen verknüpfen, geben Sie die größere Tabelle auf der linken Seite der Verknüpfung und die kleinere Tabelle auf der rechten Seite der Verknüpfung an.
  - Verteilte Caching-Lösung zur Verbesserung der Latenz und zur Reduzierung der Anzahl von Datenbank-E/A-Vorgängen.
  - Regelmäßige Wartung wie das Ausführen von Statistiken.
- Experimentieren und testen Sie Strategien in einer Nicht-Produktionsumgebung.

Ressourcen

Zugehörige Dokumente:

- Bewährte Methoden für Amazon Aurora
- Amazon Redshift-Leistung
- Die besten 10 Leistungstipps für Amazon Athena
- AWS-Datenbank-Caching
- Bewährte Methoden für die Implementierung von Amazon ElastiCache
- Partitionierung von Daten in Athena

Zugehörige Videos:

- AWS re:Invent 2023 – Bewährte Methoden zur Kostenoptimierung für AWS-Speicher
- AWS re:Invent 2022 – Leistungsüberwachung mit Amazon RDS und Aurora, mit Autodesk
- Optimieren von Amazon Athena-Abfragen mit neuen Tools zur Abfrageanalyse

Zugehörige Beispiele:

- Amazon S3 Select – Abfragen von Daten ohne Server oder Datenbanken
- AWS-Workshop „Speziell entwickelte Datenbanken“

PERF03-BP05 Implementieren von Datenzugriffsmustern, die Caching nutzen

Implementieren Sie Zugriffsmuster, die vom Daten-Cashing profitieren, damit häufig aufgerufene Daten schnell abgerufen werden können.

Typische Anti-Muster:

- Sie speichern Daten, die sich häufig ändern.
- Sie verlassen sich auf zwischengespeicherte Daten, als ob sie dauerhaft gespeichert und immer verfügbar wären.
- Sie berücksichtigen nicht die Konsistenz Ihrer zwischengespeicherten Daten.
- Sie überwachen die Effizienz Ihrer Caching-Implementierung nicht.

Vorteile der Nutzung dieser bewährten Methode: Das Speichern von Daten in einem Cache kann die Leselatenz, den Lesedurchsatz, die Benutzererfahrung und die Gesamteffizienz verbessern sowie die Kosten senken.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Ein Cache ist eine Software- oder Hardwarekomponente zum Speichern von Daten, damit zukünftige Abfragen derselben Daten schneller oder effizienter verarbeitet werden können. Die in einem Cache gespeicherten Daten können bei Verlust rekonstruiert werden, indem eine frühere Berechnung wiederholt wird oder die Daten aus einem anderen Datenspeicher abgerufen werden.

Das Cashing von Daten kann eine der effektivsten Strategien sein, um die allgemeine Anwendungsleistung zu verbessern und die Belastung Ihrer zugrunde liegenden primären Datenquellen zu verringern. Daten können auf mehreren Ebenen in der Anwendung zwischengespeichert werden, z. B. innerhalb der Anwendung, die Remoteanrufe tätigt (als clientseitiges Caching bezeichnet) oder indem Sie einen schnellen sekundären Service zum Speichern der Daten verwenden (Remote-Caching).

Clientseitiges Caching

Beim clientseitigen Caching kann jeder Client (eine Anwendung oder ein Service, die bzw. der den Backend-Datenspeicher abfragt) die Ergebnisse seiner eindeutigen Abfragen lokal für einen bestimmten Zeitraum speichern. So kann die Anzahl der Anfragen an einen Datenspeicher im Netzwerk reduziert werden, da zuerst der lokale Client-Cache überprüft wird. Wenn die Ergebnisse nicht vorhanden sind, kann die Anwendung den Datenspeicher abfragen und diese Ergebnisse lokal speichern. Dieses Muster ermöglicht es jedem Client, Daten am nächstgelegenen Ort (dem Client selbst) zu speichern, was zur geringstmöglichen Latenz führt. Clients können auch weiterhin einige Abfragen bearbeiten, wenn der Backend-Datenspeicher nicht verfügbar ist, wodurch die Verfügbarkeit des Gesamtsystems erhöht wird.

Ein Nachteil dieses Ansatzes besteht darin, dass bei Beteiligung mehrerer Clients diese möglicherweise dieselben zwischengespeicherten Daten lokal speichern. Dies führt sowohl zu doppelten Speichervorgängen als auch zu Dateninkonsistenzen zwischen diesen Clients. So kann z. B. ein Client die Ergebnisse einer Abfrage zwischenspeichern und eine Minute später führt ein anderer Client dieselbe Abfrage aus und erhält ein anderes Ergebnis.

Remote-Caching

Zum Lösen des Problems doppelter Daten zwischen Clients kann ein schneller externer Service oder Remote-Cache verwendet werden, um die abgefragten Daten zu speichern. Anstatt einen lokalen Datenspeicher zu überprüfen, prüft jeder Client den Remote-Cache, bevor er den Backend-Datenspeicher abfragt. Diese Strategie ermöglicht konsistentere Antworten zwischen den Clients, eine bessere Effizienz der gespeicherten Daten und ein höheres Volumen an zwischengespeicherten Daten, da der Speicherplatz unabhängig von den Clients skaliert wird.

Der Nachteil eines Remote-Caches besteht darin, dass das Gesamtsystem möglicherweise eine höhere Latenz aufweist, da ein zusätzlicher Netzwerk-Hop erforderlich ist, um den Remote-Cache zu überprüfen. Das clientseitige Caching kann in Kombination mit dem Remote-Caching verwendet werden, um ein mehrstufiges Caching zu implementieren und die Latenz zu verbessern.

Implementierungsschritte

1. Identifizieren Sie Datenbanken, APIs und Netzwerkservices, die vom Caching profitieren könnten. Services, die hohe Lese-Workloads oder ein hohes Lese-Schreib-Verhältnis aufweisen oder deren Skalierung teuer ist, kommen für das Caching in Frage.
   - Datenbank-Caching
   - Aktivieren des API-Cachings zur Verbesserung der Reaktionsfähigkeit
2. Identifizieren Sie die geeignete Caching-Strategie, die am besten zu Ihrem Zugriffsmuster passt.
   - Strategien für Zwischenspeicher
   - AWS-Caching-Lösungen
3. Folgen Sie den bewährten Methoden für das Caching für Ihren Datenspeicher.
4. Konfigurieren Sie eine Cache-Invalidierungsstrategie, z. B. eine Time-to-Live (TTL), für alle Daten, die ein Gleichgewicht zwischen der Aktualität der Daten und der Verringerung der Auslastung des Backend-Datenspeichers herstellt.
5. Aktivieren Sie Features wie automatische Verbindungswiederholungen, exponentielles Backoff, clientseitige Timeouts und Verbindungspooling beim Client, sofern verfügbar, um die Leistung und Zuverlässigkeit zu verbessern.

- Bewährte Methoden: Redis-Clients und Amazon ElastiCache (Redis OSS)

6. Überwachen Sie die Cache-Trefferrate mit einem Ziel von mindestens 80 %. Niedrigere Werte können auf eine unzureichende Cache-Größe oder ein Zugriffsmuster hinweisen, das nicht vom Caching profitiert.
   - Welche Metriken sollte ich überwachen?
   - Bewährte Methoden für die Überwachung von Redis-Workloads auf Amazon ElastiCache
   - Bewährte Methoden für die Überwachung mit Amazon ElastiCache (Redis OSS) unter Verwendung von Amazon CloudWatch
7. Implementieren Sie die Datenreplikation, um Lesevorgänge auf mehrere Instances auszulagern und die Leseleistung und Verfügbarkeit von Daten zu verbessern.

Ressourcen

Zugehörige Dokumente:

- Verwenden von Amazon ElastiCache Well-Architected Lense
- Bewährte Methoden für die Überwachung mit Amazon ElastiCache (Redis OSS) unter Verwendung von Amazon CloudWatch
- Welche Metriken sollte ich überwachen?
- Whitepaper „Skalierbare Leistung mit Amazon ElastiCache“
- Caching-Herausforderungen und -Strategien

Zugehörige Videos:

- Lernpfad zu Amazon ElastiCache
- Erfolgreiches Design mit bewährten Methoden für Amazon ElastiCache
- AWS re:Invent 2020 – Erfolgreiches Design mit bewährten Methoden für Amazon ElastiCache
- AWS re:Invent 2023 – [LAUNCH] Einführung von Amazon ElastiCache Serverless
- AWS re:Invent 2022 – 5 hervorragende Methoden, um die Datenebene mit Redis neu zu gestalten
- AWS re:Invent 2021 – Ausführliche Beschreibung von Amazon ElastiCache (Redis OSS)

Zugehörige Beispiele:

- Boosting MySQL database performance with Amazon ElastiCache (Redis OSS) (Steigern der MySQL-Datenbankleistung mit Amazon ElastiCache for Redis)

Networking und Bereitstellung von Inhalten

LEIST 4. Wie wählen und konfigurieren Sie Netzwerkressourcen in Ihrem Workload?

Die effektivste Datenbanklösung für ein System variiert je nach den Anforderungen an die Verfügbarkeit, Konsistenz, Partitionstoleranz, Latenz, Lebensdauer, Skalierbarkeit und Abfragefähigkeit. Viele Systeme verwenden unterschiedliche Datenbanklösungen für verschiedene Subsysteme und nutzen verschiedene Funktionen, um die Leistung zu verbessern. Die Wahl der falschen Datenbanklösung und -funktionen kann die Leistungseffizienz eines Systems schmälern.

Bewährte Methoden

- PERF04-BP01 Verstehen der Auswirkungen des Netzwerks auf die Leistung
- PERF04-BP02 Evaluieren verfügbarer Netzwerk-Features
- PERF04-BP03 Auswählen von entsprechend dedizierter Konnektivität oder VPN für Ihre Workload
- PERF04-BP04 Lastausgleich verwenden, um den Datenverkehr auf mehrere Ressourcen zu verteilen
- PERF04-BP05 Auswählen leistungsfördernder Netzwerkprotokolle
- PERF04-BP06 Auswählen des Workload-Standortes entsprechend den Netzwerkanforderungen
- PERF04-BP07 Optimieren der Netzwerkkonfiguration basierend auf Metriken

PERF04-BP01 Verstehen der Auswirkungen des Netzwerks auf die Leistung

Analysieren und verstehen Sie, wie sich netzwerkbezogene Entscheidungen auf Ihre Workload auswirken, sodass Sie eine effiziente Leistung und ein verbessertes Benutzererlebnis erzielen können.

Typische Anti-Muster:

- Der gesamte Datenverkehr fließt durch Ihre bestehenden Rechenzentren.
- Sie leiten den gesamten Datenverkehr durch zentrale Firewalls, anstatt cloudnative Netzwerksicherheitstools zu verwenden.
- Sie stellen AWS Direct Connect-Verbindungen bereit, ohne die tatsächlichen Anforderungen der Benutzer zu verstehen.

- Sie berücksichtigen beim Definieren Ihrer Netzwerklösungen die Workload-Eigenschaften und den Verschlüsselungsaufwand nicht.
- Sie verwenden On-Premises-Konzepte und -Strategien für Netzwerklösungen in der Cloud.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie verstehen, wie das Netzwerk die Workload-Leistung beeinflusst, können Sie potenzielle Engpässe erkennen, die Benutzererfahrung verbessern, die Zuverlässigkeit erhöhen und den Betriebsaufwand verringern, während sich die Workload verändert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Das Netzwerk ist für die Verbindung zwischen Anwendungskomponenten, Cloud-Services, Edge-Netzwerken und On-Premises-Daten verantwortlich und kann daher die Workload-Leistung wesentlich beeinflussen. Die Benutzererfahrung kann nicht nur durch die Workload-Leistung, sondern auch durch Netzwerklatenz, Bandbreite, Protokolle, Standort, Netzwerküberlastungen, Jitter, Durchsatz und Routing-Regeln beeinträchtigt werden.

Sie haben eine dokumentierte Liste an Netzwerkanforderungen der Workload, einschließlich Latenz, Paketgröße, Routingregeln, Protokolle und unterstützender Datenverkehrsmuster. Sie überprüfen alle verfügbaren Netzwerklösungen und identifizieren, welcher Service den Netzwerkmerkmalen Ihrer Workload entspricht. Da cloudbasierte Netzwerke schnell geändert werden können, müssen Sie Ihre Netzwerkarchitektur im Laufe der Zeit weiterentwickeln, um die effiziente Leistung zu verbessern.

Implementierungsschritte:

1. Definieren und dokumentieren Sie die Anforderungen an die Netzwerkleistung, einschließlich Metriken wie Netzwerklatenz, Bandbreite, Protokolle, Standorte, Datenverkehrsmuster (Spitzen und Frequenz), Durchsatz, Verschlüsselung, Überprüfung und Routing-Regeln.
2. Erfahren Sie mehr über wichtige AWS-Netzwerkservices wie VPCs, AWS Direct Connect, Elastic Load Balancing (ELB)und Amazon Route 53.
3. Erfassen Sie die folgenden wichtigen Netzwerkmerkmale:

Merkmale	Tools und Metriken
Grundlegende Netzwerkmerkmale	• VPC Flow Logs • AWS Transit Gateway-Flow-Protokolle

Merkmale	Tools und Metriken
	• AWS Transit Gateway-Metriken • AWS PrivateLink-Metriken
Merkmale von Anwendungsnetzwerken	• Elastic Fabric Adapter • AWS App Mesh-Metriken • Amazon API Gateway-Metriken
Merkmale von Edge-Netzwerken	• Amazon CloudFront-Metriken • Amazon Route 53-Metriken • AWS Global Accelerator-Metriken
Merkmale hybrider Netzwerke	• AWS Direct Connect-Metriken • AWS Site-to-Site VPN-Metriken • AWS Client VPN-Metriken • AWS Cloud-WAN-Metriken
Merkmale von Sicherheitsnetzwerken	• Metriken von AWS Shield, AWS WAF und AWS Network Firewall
Nachverfolgungsmerkmale	• AWS X-Ray • VPC Reachability Analyzer • Network Access Analyzer • Amazon Inspector • Amazon CloudWatch RUM

4. Benchmarks für die Netzwerkleistung festlegen und testen:
   a. Benchmark- Netzwerkdurchsatz, da einige Faktoren die Amazon EC2-Netzwerkleistung beeinflussen können, wenn sich Instances in derselben VPC befinden. Messen Sie die Netzwerkbandbreite zwischen Amazon EC2-Linux-Instances in der gleichen VPC.
   b. Führen Sie Lasttests durch, um mit Netzwerklösungen und -optionen zu experimentieren.

Ressourcen

Zugehörige Dokumente:

- Application Load Balancer
- EC2: Enhanced Networking unter Linux
- EC2: Enhanced Networking unter Windows
- EC2: Platzierungsgruppen
- Aktivieren von Enhanced Networking-Funktionen mit dem Elastic Network Adapter (ENA) in Linux-Instances
- Network Load Balancer
- Netzwerkprodukte mit AWS
- Transit Gateway
- Umstellung auf latenzbasiertes Routing in Amazon Route 53
- VPC-Endpunkte

Zugehörige Videos:

- AWS re:Invent 2023 – AWS-Netzwerkgrundlagen
- AWS re:Invent 2023 – Welche Vorteile bietet die Vernetzung für Ihre Anwendung?
- AWS re:Invent 2023 – Erweiterte VPC-Designs und neue Funktionen
- AWS re:Invent 2023 – Leitfaden für Entwickler zu Cloud-Netzwerken
- AWS re:Invent 2019 – Konnektivität mit AWS und hybriden AWS-Netzwerkarchitekturen
- AWS re:Invent 2019 – Optimieren der Netzwerkleistung für Amazon EC2-Instances
- AWS Summit Online – Verbessern der Leistung von globalen Netzwerken für Anwendungen
- AWS re:Invent 2020 – Bewährte Methoden für Netzwerke und Tipps für das Well-Architected Framework
- AWS re:Invent 2020 – Bewährte Methoden für AWS-Netzwerke in umfangreichen Migrationen

Zugehörige Beispiele:

- AWS Transit Gateway und skalierbare Sicherheitslösungen
- Workshops zu AWS-Netzwerken
- Praktischer Workshop zur Netzwerk-Firewall
- Beobachten und Diagnostizieren Ihres Netzwerks in AWS
- Finden und Beheben von Netzwerkfehlkonfigurationen in AWS

PERF04-BP02 Evaluieren verfügbarer Netzwerk-Features

Prüfen Sie die Netzwerk-Features in der Cloud, mit denen die Leistung unter Umständen verbessert werden kann. Messen Sie die Auswirkungen der Features anhand von Tests, Metriken und Analysen. Nutzen Sie beispielsweise die verfügbaren Features auf Netzwerkebene, um die Latenz, die Netzwerkentfernung oder den Jitter zu reduzieren.

Typische Anti-Muster:

- Sie bleiben innerhalb einer Region, da sich Ihre Firmenzentrale dort befindet.
- Sie verwenden Firewalls anstelle von Sicherheitsgruppen, um den Datenverkehr zu filtern.
- Sie unterbrechen TLS für die Überprüfung des Datenverkehrs, anstatt sich auf Sicherheitsgruppen, Endpunktrichtlinien und andere cloudnative Funktionen zu verlassen.
- Sie nutzen nur eine subnetzbasierte Segmentierung anstelle von Sicherheitsgruppen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie alle Service-Features und Optionen evaluieren, kann dies die Workload-Leistung verbessern, die Infrastrukturkosten senken, den Verwaltungsaufwand für die Workload reduzieren und die allgemeine Sicherheit erhöhen. Dank der weltweiten Abdeckung von AWS können Sie Ihren Kunden stets das bestmögliche Netzwerkerlebnis bieten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS bietet Services wie AWS Global Accelerator und Amazon CloudFront, die zur Verbesserung der Netzwerkleistung beitragen können, während die meisten AWS-Services über Produkt-Features verfügen (wie das Amazon S3 Transfer Acceleration -Feature) zur Optimierung des Netzwerkverkehrs.

Sehen Sie sich die verfügbaren Konfigurationsoptionen für das Netzwerk an und finden Sie heraus, wie sich diese auf Ihre Workload auswirken. Die Leistungsoptimierung hängt davon ab, wie diese Optionen mit Ihrer Architektur interagieren und welche Auswirkungen sie auf die gemessene Leistung und auf die Benutzererfahrung haben.

Implementierungsschritte

- Erstellen Sie eine Liste der Workload-Komponenten.

- Erwägen Sie die Verwendung von AWS Cloud WAN, um das Netzwerk Ihrer Organisation aufzubauen, zu verwalten und zu überwachen, wenn Sie ein einheitliches globales Netzwerk aufbauen.
- Überwachen Sie Ihre globalen Netzwerke und Kernnetzwerke mit Amazon CloudWatch Logs-Metriken. Nutzen Sie Amazon CloudWatch RUM, das Erkenntnisse bietet, die dazu beitragen, das digitale Erlebnis der Benutzer zu identifizieren, zu verstehen und zu verbessern.
- Zeigen Sie die aggregierte Netzwerklatenz zwischen AWS-Regionen und Availability Zones sowie innerhalb jeder Availability Zone an, indem Sie AWS Network Manager verwenden, um Erkenntnisse bezüglich des Zusammenhangs zwischen der Leistung Ihrer Anwendung und der Leistung des zugrunde liegenden AWS-Netzwerks zu erhalten.
- Verwenden Sie ein vorhandenes Konfigurationsmanagementdatenbank-Tool (CMDB-Tool) oder einen Service wie AWS Config um eine Bestandsaufnahme Ihrer Workload und deren Konfiguration zu erstellen.

- Wenn es sich um einen bestehenden Workload handelt, ermitteln und dokumentieren Sie die Benchmark für Ihre Leistungsmetriken. Konzentrieren Sie sich dabei auf Engpässe und Bereiche mit Verbesserungspotenzial. Leistungsbezogene Netzwerkmetriken werden je nach geschäftlichen Anforderungen und Workload-Merkmalen für die einzelnen Workloads unterschiedlich sein. Für den Anfang könnte die Prüfung folgender Metriken für Ihre Workload wichtig sein: Bandbreite, Latenz, Paketverlust, Jitter und erneute Übertragungen.
- Bei einer neuen Workload sollten Sie Lasttests durchführen, um Leistungsengpässe zu identifizieren.
- Prüfen Sie für die ermittelten Leistungsengpässe die Konfigurationsoptionen Ihrer Lösungen, um Möglichkeiten zur Leistungsverbesserung zu finden. Informieren Sie sich über die folgenden wichtigen Netzwerkoptionen und -Features:

Verbesserungsmöglichkeit	Lösung
Netzwerkpfad oder -routen	Mithilfe des Network Access Analyzer Pfade oder Routen identifizieren.
Netzwerkprotokolle	Siehe PERF04-BP05 Auswählen leistungs fördernder Netzwerkprotokolle
Netzwerktopologie	Bewerten Sie Ihre betrieblichen und leistungs bezogenen Kompromisse zwischen VPC Peering und AWS Transit Gateway beim

Verbesserungsmöglichkeit	Lösung
	Verbinden mehrerer Konten. AWS Transit Gateway vereinfacht die Art und Weise, wie Sie all Ihre VPCs miteinander verbinden, die sich über Tausende von AWS-Konten-Netzwerken bis hin zu On-Premises-Netzwerken erstrecken können. Teilen Sie Ihr AWS Transit Gateway zwischen mehreren Konten mit AWS Resource Access Manager. Siehe PERF04-BP03 Auswählen von entsprechend dedizierter Konnektivität oder VPN für Ihre Workload

Verbesserungsmöglichkeit	Lösung
Netzwerkservices	AWS Global Accelerator ist ein Netzwerks ervice, der die Leistung des Benutzerd atenverkehrs unter Verwendung der globalen Netzwerkinfrastruktur von AWS um bis zu 60 % verbessert. Amazon CloudFront kann die Leistung Ihrer Workload-Inhaltsbereitstellung und Latenz weltweit verbessern. Nutzen Sie Lambda@edge, um Funktionen auszuführen, die den Inhalt, den CloudFront bereitstellt, besser an die Benutzer anpassen, die Latenz reduzieren und die Leistung erhöhen. Amazon Route 53 bietet Optionen für latenzbasiertes Routing, Geolocation-Routin g, Routing auf der Grundlage der geografis chen Näheund IP-basiertes Routing und trägt damit zur Leistungsverbesserung der Workload für eine globale Zielgruppe bei. Ermitteln Sie, welche Routing-Option Ihre Workload-Leistung optimieren würde. Prüfen Sie dazu Ihren Workload-Datenverkehr und den Benutzerstandort bei der globalen Verteilung Ihrer Workload.

Verbesserungsmöglichkeit	Lösung
Speicherressourcen-Features	Amazon S3 Transfer Acceleration ist ein Feature, mit dessen Hilfe externe Benutzer beim Hochladen von Daten in Amazon S3 von den Netzwerkoptimierungen von CloudFront profitieren können. Dies erleichtert die Übertragung großer Datenmengen von Remote-Standorten ohne spezielle Konnektivität zur AWS Cloud. Multi-Region-Zugriffspunkte in Amazon S3 replizieren Inhalte in mehreren Regionen und vereinfachen die Workload durch die Bereitstellung eines Zugriffspunkts. Bei Verwendung eines Multi-Region-Zugriffspunkts können Sie Daten anfordern oder in Amazon S3 schreiben, wobei der Service den Bucket mit der geringsten Latenz ermittelt.

Verbesserungsmöglichkeit	Lösung
Compute-Ressourcen-Features	Elastic Network Interfaces (ENA, Elastic-Network-Schnittstellen), die von Amazon EC2-Instances, Containern und Lambda-Funktionen verwendet werden, sind pro Fluss begrenzt. Prüfen Sie Ihre Platzierungsgruppen, um Ihren EC2-Netzwerkdurchsatz zu optimieren.. Um Engpässe auf Pro-Fluss-Basis zu vermeiden, sollten Sie Ihre Anwendung so gestalten, dass mehrere Flüsse verwendet werden. Um Ihre datenverarbeitungsbezogenen Netzwerkmetriken zu überwachen und Einblicke in diese Metriken zu erhalten, verwenden Sie CloudWatch-Metriken und ethtool. Der `ethtool` -Befehl ist im ENA-Treiber enthalten und stellt zusätzliche netzwerkbezogene Metriken zur Verfügung, die als benutzerdefinierte Metriken in CloudWatch veröffentlicht werden können. Amazon Elastic Network Adapters (ENA) ermöglichen eine weitere Optimierung, da sie einen besseren Durchsatz für Ihre Instances innerhalb einer Cluster-Placement-Gruppe bieten. Elastic Fabric Adapter (EFA) ist eine Netzwerkschnittstelle für Amazon EC2-Instances, mit der Sie Workloads, die ein hohes Maß an Kommunikation zwischen Knoten erfordern, in AWS bedarfsgesteuert ausführen können. Amazon EBS-optimierte Instances verwenden einen optimierten Konfigurations-Stack und

Verbesserungsmöglichkeit	Lösung
	stellen zusätzliche dedizierte Kapazität zur Erhöhung der Amazon EBS-I/O bereit.

Ressourcen

Zugehörige Dokumente:

- Application Load Balancer
- EC2: Enhanced Networking unter Linux
- EC2: Enhanced Networking unter Windows
- EC2: Platzierungsgruppen
- Aktivieren von Enhanced Networking-Funktionen mit dem Elastic Network Adapter (ENA) in Linux-Instances
- Network Load Balancer
- Netzwerkprodukte mit AWS
- Umstellung auf latenzbasiertes Routing in Amazon Route 53
- VPC-Endpunkte
- VPC Flow Logs

Zugehörige Videos:

- AWS re:Invent 2023 – Sind Sie bereit für Neues? Gestaltung von Netzwerken für Wachstum und Flexibilität
- AWS re:Invent 2023 – Erweiterte VPC-Designs und neue Funktionen
- AWS re:Invent 2023 – Leitfaden für Entwickler von Cloud-Netzwerken
- AWS re:Invent 2022 – Ausführliche Beschreibung der AWS-Netzwerkinfrastruktur
- AWS re:Invent 2019 – Konnektivität mit AWS und hybriden AWS-Netzwerkarchitekturen
- AWS re:Invent 2018 – Optimieren der Netzwerkleistung für Amazon EC2-Instances
- AWS Global Accelerator

Zugehörige Beispiele:

- AWS Transit Gateway und skalierbare Sicherheitslösungen
- Workshops zu AWS-Netzwerken
- Überwachung und Diagnose Ihres Netzwerks
- Finden und Beheben von Netzwerkfehlkonfigurationen in AWS

PERF04-BP03 Auswählen von entsprechend dedizierter Konnektivität oder VPN für Ihre Workload

Wenn Hybrid-Konnektivität für die Verbindung von On-Premises- und Cloud-Ressourcen erforderlich ist, stellen Sie ausreichend Bandbreite bereit, um Ihre Leistungsanforderungen zu erfüllen. Schätzen Sie die Anforderungen an Bandbreite und Latenz für Ihren hybriden Workload ab. Diese Zahlen dienen als Grundlage für die Größenanpassung.

Typische Anti-Muster:

- Sie evaluieren nur VPN-Lösungen für Ihre Netzwerk-Verschlüsselungsanforderungen.
- Sie bewerten keine Optionen für Sicherung oder redundante Verbindungen.
- Sie identifizieren nicht alle Workload-Anforderungen (Verschlüsselung, Protokoll, Bandbreite und Traffic-Bedarf).

Vorteile der Nutzung dieser bewährten Methode: Durch die Auswahl und Konfiguration geeigneter Konnektivitätslösungen wird die Zuverlässigkeit Ihrer Workloads erhöht und die Leistung maximiert. Indem Sie die Workload-Anforderungen identifizieren, im Voraus planen und hybride Lösungen evaluieren, verringern Sie teure physische Netzwerkänderungen sowie den Betriebsaufwand und verkürzen die Amortisationszeit.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Entwickeln Sie eine hybride Netzwerkarchitektur entsprechend den Bandbreitenanforderungen. AWS Direct Connect ermöglicht es Ihnen, Ihr On-Premises-Netzwerk privat mit AWS zu verbinden. Sie ist geeignet, wenn Sie eine hohe Bandbreite und eine geringe Latenz bei gleichbleibender Leistung benötigen. Eine VPN-Verbindung stellt eine sichere Verbindung über das Internet her. Sie wird verwendet, wenn lediglich eine temporäre Verbindung erforderlich ist, wenn die Kosten eine Rolle spielen, oder wenn bei der Verwendung von AWS Direct Connect darauf gewartet wird, dass eine resiliente physische Netzwerkkonnektivität hergestellt wird.

Wenn Ihre Bandbreitenanforderungen hoch sind, könnten Sie mehrere AWS Direct Connect oder VPN-Services in Betracht ziehen. Der Lastausgleich für den Datenverkehr kann über die Services hinweg erfolgen. Allerdings empfehlen wir aufgrund der Latenz- und Bandbreitenunterschiede keinen Lastausgleich zwischen AWS Direct Connect und VPN.

Implementierungsschritte

1. Schätzen Sie die Anforderungen an Bandbreite und Latenz für Ihre bestehenden Anwendungen ab.
   a. Für bestehende Workloads , die auf AWS umgestellt werden, nutzen Sie die Daten aus Ihren internen Systemen zur Überwachung des Netzwerks.
   b. Bei neuen oder bestehenden Workloads, für die Sie keine Monitoring-Daten haben, beraten Sie sich mit den Besitzern der Produkte, um angemessene Metriken für die Leistung zu bestimmen und ein gutes Benutzererlebnis zu gewährleisten.
2. Wählen Sie eine dedizierte Verbindung oder ein VPN als Konnektivitätsoption aus. Je nach den Anforderungen des Workloads (Verschlüsselung, Bandbreite und Traffic-Bedarf) können Sie entweder AWS Direct Connect oder AWS VPN auswählen (oder beides). Das folgende Diagramm kann Ihnen bei der Wahl der geeigneten Verbindungsart helfen.
   a. AWS Direct Connect liefert dedizierte Konnektivität für die AWS-Umgebung, von 50 Mbit/s bis zu 100 Gbit/s, entweder über dedizierte Verbindungen oder über gehostete Verbindungen. So erhalten Sie eine verwaltete und kontrollierte Latenz und bereitgestellte Bandbreite, damit sich Ihr Workload effizient mit anderen Umgebungen verbinden kann. Mit einem AWS Direct Connect-Partner können Sie eine End-to-End-Konnektivität aus mehreren Umgebungen nutzen und so ein erweitertes Netzwerk mit konsistenter Leistung bereitstellen. AWS bietet eine Skalierung der Bandbreite für Direct Connect-Verbindungen entweder über native 100 Gbit/s, Link Aggregation Group (LAG) oder BGP Equal-Cost Multipath (ECMP).
   b. Das AWS Site-to-Site VPN bietet einen verwalteten VPN-Service, der das IPsec (Internet Protocol Security) unterstützt. Wenn eine VPN-Verbindung erstellt wird, besteht die VPN-Verbindung aus zwei Tunneln, um eine hohe Verfügbarkeit zu gewährleisten.
3. Folgen Sie der AWS-Dokumentation, um eine geeignete Verbindungsoption auszuwählen:
   a. Wenn Sie sich für die Verwendung von AWS Direct Connect entscheiden, wählen Sie die entsprechende Bandbreite für Ihre Konnektivität aus.
   b. Wenn Sie ein AWS Site-to-Site VPN über mehrere Standorte hinweg nutzen, um eine Verbindung zu einer AWS-Region herzustellen, sollten Sie eine beschleunigte Site-to-Site VPN-Verbindung verwenden, um die Netzwerkleistung verbessern zu können.

c. Wenn Ihr Netzwerkdesign aus einer IPSec-VPN-Verbindung über AWS Direct Connectbesteht, sollten Sie erwägen, Private IP VPN zu verwenden, um die Sicherheit zu verbessern und eine Segmentierung zu erzielen. AWS Site-to-Site Private IP VPN wird auf der virtuellen Transitschnittstelle bereitgestellt.

d. AWS Direct Connect SiteLink ermöglicht die Schaffung redundanter Verbindungen mit niedriger Latenz zwischen Ihren Rechenzentren weltweit, indem Daten über den schnellsten Weg zwischen AWS Direct Connect-Standorten, unter Umgehung von AWS-Regionen, gesendet werden.

4. Überprüfen Sie Ihr Konnektivitäts-Setup, bevor Sie es in der Produktion einsetzen. Führen Sie Sicherheits- und Leistungstests durch, um sicherzustellen, dass das Setup Ihre Anforderungen an Bandbreite, Zuverlässigkeit, Latenz und Compliance erfüllt.
5. Überwachen Sie regelmäßig die Leistung und Nutzung Ihrer Konnektivität und optimieren Sie sie bei Bedarf.

Flussdiagramm zur deterministischen Leistung

Ressourcen

Zugehörige Dokumente:

- Netzwerkprodukte mit AWS
- AWS Transit Gateway
- VPC-Endpunkte
- Erstellen einer skalierbaren und sicheren Multi-VPC-AWS-Netzwerkinfrastruktur
- Client-VPN

Zugehörige Videos:

- AWS re:Invent 2023 – Aufbau einer hybriden Netzwerkkonnektivität mit AWS
- AWS re:Invent 2023 – Sichere Remote-Verbindung zu AWS
- AWS re:Invent 2022 – Optimierung der Leistung mit Amazon CloudFront
- AWS re:Invent 2019 – Konnektivität mit AWS und hybriden AWS-Netzwerkarchitekturen

- AWS re:Invent 2020 Connect – AWS Transit Gateway Connect

Zugehörige Beispiele:

- AWS Transit Gateway und skalierbare Sicherheitslösungen
- Workshops zu AWS-Netzwerken

PERF04-BP04 Lastausgleich verwenden, um den Datenverkehr auf mehrere Ressourcen zu verteilen

Verteilen Sie den Datenverkehr auf mehrere Ressourcen oder Services, um von der Elastizität der Cloud zu profitieren. Sie können den Lastausgleich auch nutzen, um die Terminierung von Verschlüsselung auszulagern. So lässt sich die Leistung und Zuverlässigkeit optimieren und der Datenverkehr effektiv verwalten und weiterleiten.

Typische Anti-Muster:

- Sie berücksichtigen bei der Wahl des Load-Balancer-Typs nicht die Anforderungen Ihres Workloads.
- Sie nutzen die Funktionen des Load Balancers nicht zur Optimierung der Leistung.
- Der Workload ist direkt mit dem Internet verbunden, ohne dass ein Load Balancer zum Einsatz kommt.
- Sie leiten den gesamten Internetverkehr über vorhandene Load Balancer weiter.
- Sie nutzen einen generischen TCP-Lastausgleich und lassen die SSL-Verschlüsselung von den einzelnen Rechenknoten verarbeiten.

Vorteile der Nutzung dieser bewährten Methode: Ein Load Balancer verarbeitet die variierende Last des Anwendungsdatenverkehrs in einer einzigen oder in mehreren Availability Zones und ermöglicht eine hohe Verfügbarkeit, automatische Skalierung sowie eine bessere Nutzung für Ihre Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Load Balancer fungieren als Eingangspunkt für Ihren Workload und verteilen den Datenverkehr von dort aus auf Ihre Backend-Ziele – wie Computing-Instances oder Container –, um die Nutzung zu verbessern.

Die Wahl des richtigen Load Balancer-Typs ist der erste Schritt zur Optimierung Ihrer Architektur. Starten Sie mit einer Auflistung Ihrer Workload-Merkmale wie Protokoll (z. B. TCP, HTTP, TLS oder WebSockets), Zieltyp (z. B. Instances, Container oder Serverless), Anwendungsanforderungen (z. B. langfristige Verbindungen, Benutzerauthentifizierung oder Stickiness) und Platzierung (z. B. Region, lokale Zone, Outposts oder Zonenisolierung).

AWS stellt für Ihre Anwendungen mehrere Modelle zur Verwendung der Lastenverteilung bereit. Application Load Balancer eignet sich optimal für die Lastenverteilung von HTTP- und HTTPS-Datenverkehr. Sie profitieren hierbei von einem fortschrittlichen Routing von Anforderungen, die es Ihnen ermöglicht, moderne Anwendungsarchitekturen mit Microservices und Containern bereitzustellen.

Network Load Balancer eignet sich optimal für die Lastenverteilung von TCP-Datenverkehr, wenn eine hohe Leistung erforderlich ist. Hiermit lassen sich mit konstant geringer Latenz Millionen Anforderungen pro Sekunde und plötzliche Datenverkehrsspitzen oder schwankende Datenverkehrsmuster verarbeiten.

Elastic Load Balancing ermöglicht die integrierte Zertifikatverwaltung und SSL/TLS-Entschlüsselung. Auf diese Weise können Sie die SSL-Einstellungen des Load Balancers flexibel zentral verwalten und CPU-intensive Arbeitsschritte für Ihren Workload auslagern.

Nachdem Sie sich für den richtigen Load Balancer entschieden haben, können Sie damit beginnen, seine Features zu nutzen, um die Belastung Ihres Backends durch den Datenverkehr zu verringern.

So können Sie beispielsweise sowohl mit Application Load Balancer (ALB) als auch mit Network Load Balancer (NLB) die SSL/TLS-Verschlüsselung auslagern, was die Möglichkeit bietet, den CPU-intensiven TLS-Handshake bei Ihren Zielen zu vermeiden und die Verwaltung der Zertifikate zu verbessern.

Wenn Sie SSL/TLS-Offloading in Ihrem Load Balancer konfigurieren, übernimmt dieser die Verschlüsselung des Datenverkehrs von und zu den Clients. Er leitet den Datenverkehr dann unverschlüsselt an Ihre Backends weiter, wodurch Ihre Backend-Ressourcen entlastet werden und die Reaktionszeit für die Clients verbessert wird.

Application Load Balancer kann außerdem HTTP/2-Datenverkehr ausliefern, ohne dass Sie ihn auf Ihren Zielen unterstützen müssen. Diese einfache Entscheidung kann die Reaktionszeit Ihrer Anwendung verbessern, da HTTP/2 TCP-Verbindungen effizienter nutzt.

Bei der Definition der Architektur sollten Sie die Anforderungen an die Latenz Ihres Workloads berücksichtigen. Wenn Sie beispielsweise eine latenzempfindliche Anwendung haben, können Sie

sich für Network Load Balancer mit einer extrem niedrigen Latenz entscheiden. Alternativ können Sie Ihr Workload auch näher an Ihre Kunden heranbringen, indem Sie Application Load Balancer in AWS Local Zones oder sogar AWS Outposts einsetzen.

Eine weitere Überlegung für latenzempfindliche Workloads ist das zonenübergreifende Load-Balancing. Beim zonenübergreifenden Lastausgleich nimmt jeder Load Balancer-Knoten eine Verteilung des Datenverkehrs auf die registrierten Ziele in allen zulässigen Availability Zones vor.

Verwenden Sie die Auto Scaling-Integration für Ihren Load Balancer. Einer der Schlüssel für ein leistungsfähiges System ist die richtige Größenanpassung Ihrer Backend-Ressourcen. Zu diesem Zweck können Sie Load Balancer-Integrationen für Backend-Zielressourcen nutzen. Mithilfe der Load Balancer-Integration mit Auto Scaling-Gruppen werden Ziele je nach Bedarf als Reaktion auf den eingehenden Datenverkehr zum Load-Balancer hinzugefügt oder aus ihm entfernt. Load Balancers können für containerisierte Workloads außerdem mit Amazon ECS und Amazon EKS integriert werden.

- Amazon ECS – Service-Lastenverteilung
- Anwendungslastenverteilung auf Amazon EKS
- Netzwerklastenverteilung auf Amazon EKS

Implementierungsschritte

- Definieren Sie Ihre Anforderungen an die Lastenverteilung, einschließlich Datenverkehrsvolumen, Verfügbarkeit und Anwendungsskalierbarkeit.
- Wählen Sie den richtigen Load Balancer-Typ für Ihre Anwendung.
  - Verwenden Sie Application Load Balancer für HTTP/HTTPS Workloads.
  - Verwenden Sie Network Load Balancer für Nicht-HTTP-Workloads, die TCP oder UDP nutzen.
  - Verwenden Sie eine Kombination aus beiden (ALB als Ziel von NLB) aus, wenn Sie die Funktionen beider Produkte nutzen möchten. Dies ist zum Beispiel möglich, wenn Sie die statischen IP-Adressen von NLB zusammen mit dem HTTP-Header-basierten Routing von ALB verwenden möchten oder wenn Sie Ihren HTTP-Workload mit AWS PrivateLink teilen möchten.
  - Einen vollständigen Vergleich von Load Balancers finden Sie im ELB-Produktvergleich.
- Verwenden Sie nach Möglichkeit SSL/TLS-Offloading.
  - Konfigurieren Sie HTTPS/TLS-Listener, bei denen Application Load Balancer und Network Load Balancer mit AWS Certificate Manager integriert sind.

  - Beachten Sie, dass einige Workloads aus Compliance-Gründen eine Ende-zu-Ende-Verschlüsselung benötigen können. In diesem Fall ist es erforderlich, die Verschlüsselung an den Zielen zuzulassen.
  - Bewährte Methoden für die Sicherheit finden Sie unter SEC09-BP02 Erzwingen einer Verschlüsselung bei der Übertragung.
- Wählen Sie den richtigen Routing-Algorithmus (nur ALB) aus.
  - Der Routing-Algorithmus kann einen entscheidenden Einfluss darauf haben, wie gut Ihre Backend-Ziele ausgelastet sind und wie sie die Leistung beeinflussen. ALB bietet zum Beispiel zwei Optionen für Routing-Algorithmen:
  - Am wenigsten ausstehende Anfragen: Verwenden Sie diese Option, um eine bessere Verteilung der Last auf Ihre Backend-Ziele zu erreichen, wenn die Anfragen für Ihre Anwendung unterschiedlich komplex sind oder Ihre Ziele unterschiedliche Kapazitäten für die Verarbeitung haben.
  - Round Robin: Verwenden Sie diese Option, wenn die Anfragen und Ziele ähnlich sind oder wenn Sie die Anfragen gleichmäßig auf die Ziele verteilen müssen.
- Ziehen Sie eine zonenübergreifende Verarbeitung oder Zonenisolierung in Betracht.
  - Verwenden Sie die deaktivierte zonenübergreifende Isolierung (Zonenisolierung), um die Latenz zu verbessern und Domänen mit Zonenfehlern zu vermeiden. In NLB ist dies standardmäßig deaktiviert. In ALB können Sie die Option pro Gruppe deaktivieren.
  - Verwenden Sie die aktivierte zonenübergreifende Verarbeitung für eine höhere Verfügbarkeit und Flexibilität. Standardmäßig ist die zonenübergreifende Verarbeitung für ALB aktiviert. In NLB können Sie sie pro Gruppe aktivieren.
- Aktivieren Sie HTTP-Keep-Alives für Ihre HTTP-Workloads (nur ALB). Mit diesem Feature kann der Load Balancer Backend-Verbindungen wiederverwenden, bis die Keep-Alive-Zeit abgelaufen ist, wodurch sich Ihre HTTP-Anfrage- und Reaktionszeiten verbessern und die Auslastung der Ressourcen auf Ihren Backend-Zielen reduziert wird. Details zu dieser Funktion für Apache und Nginx finden Sie unter Was sind die optimalen Einstellungen für die Verwendung von Apache oder NGINX als Backend-Server für ELB?.
- Aktivieren Sie die Überwachung für Ihren Load Balancer.
  - Aktivieren Sie die Zugriffsprotokolle für Ihren Application Load Balancer und Network Load Balancer.
  - Die wichtigsten zu berücksichtigenden Elemente für ALB sind `request_processing_time`, `target_processing_time` und `response_processing_time`.

- Die wichtigsten zu berücksichtigenden Elemente für ALB sind `connection_time` und `tls_handshake_time`.
- Bereiten Sie sich darauf vor, die Protokolle bei Bedarf abfragen zu können. Sie können Amazon Athena verwenden, um sowohl ALB-Protokolle als auch NLB-Protokolle abzufragen.
- Erstellen Sie Alarme für leistungsbezogene Metriken wie `TargetResponseTime` für ALB.

Ressourcen

Zugehörige Dokumente:

- ELB-Produktvergleich
- Globale AWS-Infrastruktur
- Verbesserung der Leistung und Senkung der Kosten durch Availability Zone-Affinität
- Schritt für Schritt zur Protokollanalyse mit Amazon Athena
- Abfragen von Application Load Balancer-Protokollen
- Überwachen Ihrer Application Load Balancers
- Überwachen Ihrer Network Load Balancer
- Verwenden Sie Elastic Load Balancing, um den Datenverkehr über die Instances in Ihrer Auto Scaling-Gruppe zu verteilen.

Zugehörige Videos:

- AWS re:Invent 2023 – Welche Vorteile bietet die Vernetzung für Ihre Anwendung?
- AWS re:Inforce 20 – So verbessern Sie mit Elastic Load Balancing Ihren Sicherheitsstatus im großen Umfang
- AWS re:Invent 2018 – Elastic Load Balancing: Ein ausführliche Beschreibung und bewährte Methoden
- AWS re:Invent 2021 – So wählen Sie den richtigen Load Balancer für Ihre AWS-Workloads aus
- AWS re:Invent 2019 – Holen Sie das Beste aus Elastic Load Balancing für verschiedene Workloads heraus

Zugehörige Beispiele:

- Gateway Load Balancer

- CDK und AWS CloudFormation-Beispiele für die Protokollanalyse mit Amazon Athena

PERF04-BP05 Auswählen leistungsfördernder Netzwerkprotokolle

Treffen Sie Entscheidungen über Protokolle für die Kommunikation zwischen Systemen und Netzwerken auf Grundlage der Auswirkungen, die sich für die Leistung der Workload ergeben.

In Bezug auf die Erzielung eines höheren Durchsatzes besteht eine Beziehung zwischen der Latenz und der Bandbreite. Wenn Ihre Dateiübertragung über TCP (Transmission Control Protocol) erfolgt, verringern höhere Latenzen höchstwahrscheinlich den gesamten Durchsatz. Es gibt verschiedene Ansätze, dies mit der TCP-Optimierung und optimierten Übertragungsprotokollen zu lösen. Eine Lösung besteht jedoch in der Verwendung des User Datagram Protocol (UDP).

Typische Anti-Muster:

- Sie verwenden TCP unabhängig von den Leistungsanforderungen für alle Workloads.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie sicherstellen, dass ein geeignetes Protokoll für die Kommunikation zwischen Benutzern und Workload-Komponenten verwendet wird, können Sie das Benutzererlebnis für Ihre Anwendungen insgesamt verbessern. Das verbindungslose UDP ermöglicht zwar beispielsweise eine hohe Geschwindigkeit, bietet aber weder eine erneute Übertragung noch hohe Zuverlässigkeit. TCP ist ein Protokoll mit vollem Funktionsumfang, bringt jedoch einen größeren Overhead für die Verarbeitung der Pakete mit sich.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Wenn Sie in der Lage sind, verschiedene Protokolle für Ihre Anwendung auszuwählen, und Sie über Fachwissen in diesem Bereich verfügen, optimieren Sie Ihre Anwendungs- und Endbenutzererfahrung, indem Sie ein anderes Protokoll verwenden. Beachten Sie, dass dieser Ansatz mit erheblichen Schwierigkeiten verbunden ist und nur versucht werden sollte, wenn Sie Ihre Anwendung zuvor auf andere Weise optimiert haben.

Um die Leistung Ihres Workloads zu verbessern, sollten Sie in erster Linie die Anforderungen an die Latenz und den Durchsatz kennen und dann Netzwerkprotokolle auswählen, die die Leistung optimieren.

Wann sollten Sie TCP verwenden

TCP bietet eine zuverlässige Zustellung von Daten und kann für die Kommunikation zwischen Workload-Komponenten verwendet werden, bei denen die Zuverlässigkeit und die garantierte Zustellung von Daten wichtig sind. Viele webbasierte Anwendungen verlassen sich auf TCP-basierte Protokolle wie HTTP und HTTPS, um TCP-Sockets für die Kommunikation zwischen Anwendungskomponenten zu öffnen. E-Mail- und Dateidatenübertragung sind gängige Anwendungen, die auch TCP verwenden, da es sich um einen einfachen und zuverlässigen Übertragungsmechanismus zwischen Anwendungskomponenten handelt. Die Verwendung von TLS mit TCP kann zu einem gewissen Overhead bei der Kommunikation führen, was eine erhöhte Latenz und einen verringerten Durchsatz zur Folge haben kann. Sie bietet jedoch den Vorteil der Sicherheit. Der Overhead entsteht vor allem durch den zusätzlichen Aufwand des Handshake-Prozesses, der mehrere Roundtrips in Anspruch nehmen kann. Sobald der Handshake abgeschlossen ist, ist der Overhead für die Ver- und Entschlüsselung der Daten relativ gering.

Wann sollten Sie UDP verwenden

UDP ist ein verbindungsloses Protokoll und eignet sich daher für Anwendungen, die eine schnelle, effiziente Übertragung benötigen, wie z. B. die Protokollierung, die Überwachung und VoIP-Daten. Ziehen Sie die Verwendung von UDP auch in Betracht, wenn Sie Workload-Komponenten haben, die auf kleine Abfragen von einer großen Anzahl von Clients reagieren, um eine optimale Leistung des Workloads zu gewährleisten. Datagram Transport Layer Security (DTLS) ist die UDP-Entsprechung von Transport Layer Security (TLS). Bei der Verwendung von DTLS mit UDP entsteht der Overhead durch die Verschlüsselung und Entschlüsselung der Daten, da der Handshake-Prozess vereinfacht ist. DTLS fügt den UDP-Paketen außerdem einen geringen Overhead hinzu, da es zusätzliche Felder zur Angabe der Sicherheitsparameter und zur Erkennung von Manipulationen umfasst.

Wann sollten Sie SRD verwenden

Scalable Reliable Datagram (SRD) ist ein Netzwerktransportprotokoll, das für Workloads mit hohem Durchsatz optimiert ist, da es in der Lage ist, den Datenverkehr über mehrere Pfade zu verteilen und sich schnell von Paketverlusten oder Verbindungsfehlern zu erholen. SRD eignet sich daher am besten für HPC-Workloads (High Performance Computing), die einen hohen Durchsatz und eine geringe Latenz bei der Kommunikation zwischen Computing-Knoten erfordern. Dazu gehören z. B. parallele Verarbeitungsaufgaben wie Simulationen, Modellierung und Datenanalyse, bei denen eine große Menge an Daten zwischen den Knoten übertragen werden muss.

Implementierungsschritte

1. Verwenden Sie die AWS Global Accelerator- und AWS Transfer Family- Services, um den Durchsatz Ihrer Anwendungen für die Onlineübertragung von Dateien zu verbessern. Der AWS

Global Accelerator-Service hilft Ihnen, die Latenz zwischen Ihren Client-Geräten und Ihrem Workload auf AWS zu verringern. Mit AWS Transfer Family können Sie TCP-basierte Protokolle wie Secure Shell File Transfer Protocol (SFTP) und File Transfer Protocol over SSL (FTPS) verwenden, um Ihre Dateiübertragungen zu AWS-Speicherservices sicher zu skalieren und zu verwalten.

2. Bestimmen Sie anhand der Netzwerklatenz, ob TCP für die Kommunikation zwischen Workload-Komponenten geeignet ist. Wenn die Netzwerklatenz zwischen Ihrer Client-Anwendung und dem Server hoch ist, kann der TCP-Drei-Wege-Handshake einige Zeit in Anspruch nehmen, was sich auf die Reaktionsfähigkeit Ihrer Anwendung auswirkt. Metriken wie Time to First Byte (TTFB) und Round-Trip Time (RTT) können zur Messung der Netzwerklatenz verwendet werden. Wenn Ihr Workload dynamische Inhalte für Benutzer bereitstellt, sollten Sie die Verwendung von Amazon CloudFrontin Betracht ziehen. So wird eine dauerhafte Verbindung zu jeder Quelle für dynamische Inhalte hergestellt, um die Zeit für den Verbindungsaufbau zu vermeiden, die sonst jede Client-Anfrage verlangsamen würde.
3. Die Verwendung von TLS mit TCP oder UDP kann aufgrund der Auswirkungen der Ver- und Entschlüsselung zu einer erhöhten Latenz und einem reduzierten Durchsatz für Ihren Workload führen. Ziehen Sie für solche Workloads das SSL/TLS-Offloading auf Elastic Load Balancing in Betracht, um die Leistung des Workloads zu verbessern, indem Sie den Load Balancer die SSL/TLS-Verschlüsselung und -Entschlüsselung übernehmen lassen, anstatt dies den Backend-Instances zu überlassen. Dies kann dazu beitragen, die CPU-Auslastung der Backend-Instances zu reduzieren, was die Leistung verbessern und die Kapazität erhöhen kann.
4. Verwenden Sie den Network Load Balancer (NLB), um Services bereitzustellen, die auf dem UDP-Protokoll basieren (wie die Authentifizierung und Autorisierung, die Protokollierung, DNS, IoT und das Streamen von Medien), um die Leistung und Zuverlässigkeit Ihres Workloads zu verbessern. Der NLB verteilt den eingehenden UDP-Datenverkehr auf mehrere Ziele, sodass Sie Ihren Workload horizontal skalieren, die Kapazität erhöhen und den Overhead eines einzelnen Ziels reduzieren können.
5. Für Ihre HPC-Workloads (High Performance Computing) sollten Sie die Elastic Network Adapter (ENA) Express-Funktionalität in Betracht ziehen, die das SRD-Protokoll nutzt, um die Leistung des Netzwerks zu verbessern, indem sie eine höhere Bandbreite für einen einzelnen Datenfluss (25 Gbit/s) und eine niedrigere Latenz (25 Perzentil) für den Netzwerkverkehr zwischen EC2-Instances bietet.
6. Verwenden Sie den Application Load Balancer (ALB), um Ihren gRPC-Datenverkehr (Remote Procedure Calls) zwischen Workload-Komponenten oder zwischen gRPC-Clients und -Services zu routen und ein Load-Balancing durchzuführen. gRPC verwendet das TCP-basierte HTTP/2-

Protokoll für den Transport und bietet Vorteile in Bezug auf die Leistung, wie z. B. einen geringeren Netzwerk-Footprint, Komprimierung, effiziente binäre Serialisierung, Unterstützung zahlreicher Sprachen und bidirektionales Streaming.

Ressourcen

Zugehörige Dokumente:

- Routing von UDP-Verkehr nach Kubernetes
- Application Load Balancer
- EC2: Enhanced Networking unter Linux
- EC2: Enhanced Networking unter Windows
- EC2: Platzierungsgruppen
- Aktivieren von Enhanced Networking-Funktionen mit dem Elastic Network Adapter (ENA) in Linux-Instances
- Network Load Balancer
- Netzwerkprodukte mit AWS
- Umstellung auf latenzbasiertes Routing in Amazon Route 53
- VPC-Endpunkte

Zugehörige Videos:

- AWS re:Invent 2022 – Skalierung der Netzwerkleistung auf Amazon Elastic Compute Cloud-Instances der nächsten Generation
- AWS re:Invent 2022 – Grundlagen der Anwendungsnetzwerke

Zugehörige Beispiele:

- AWS Transit Gateway und skalierbare Sicherheitslösungen
- Workshops zu AWS-Netzwerken

PERF04-BP06 Auswählen des Workload-Standortes entsprechend den Netzwerkanforderungen

Evaluieren Sie Optionen für die Platzierung von Ressourcen, um die Latenz im Netzwerk zu verringern und den Durchsatz zu verbessern und so ein optimales Benutzererlebnis durch kürzere Seitenlade- und Datentransferzeiten zu gewährleisten.

Typische Anti-Muster:

- Sie konsolidieren alle Workload-Ressourcen an einem geografischen Standort.
- Sie haben sich für die Region entschieden, die Ihrem Standort, aber nicht dem Workload-Endbenutzer, am nächsten liegt.

Vorteile der Nutzung dieser bewährten Methode: Das Benutzererlebnis wird stark von der Latenz zwischen dem/der Benutzer:in und Ihrer Anwendung beeinflusst. Durch die Verwendung geeigneter AWS-Regionen und des privaten globalen AWS-Netzwerks können Sie die Latenz reduzieren und Remote-Benutzern ein besseres Erlebnis bieten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Ressourcen wie Amazon EC2-Instances werden in Availability Zones innerhalb von AWS-Regionen, AWS Local Zones, AWS Outposts oder AWS Wavelength-Zonen platziert. Die Auswahl dieses Standorts beeinflusst die Latenz des Netzwerks und den Durchsatz vom Standort des Benutzers aus. Edge-Services wie Amazon CloudFront und AWS Global Accelerator können ebenfalls zur Verbesserung der Netzwerkleistung eingesetzt werden, indem sie entweder Inhalte an Edge-Standorten zwischenspeichern oder den Benutzer:innen einen optimalen Pfad zum Workload durch das globale Netzwerk von AWS bereitstellen.

Amazon EC2 verfügt über Platzierungsgruppen für das Netzwerk. Eine Platzierungsgruppe ist eine logische Gruppierung von Instances, um die Latenz zu verringern. Die Verwendung von Platzierungsgruppen mit unterstützten Instance-Typen und einem Elastic Network Adapter (ENA) ermöglicht die Verarbeitung von Workloads in einem Netzwerk mit 25 Gbit/s, reduziertem Jitter und geringer Latenz. Platzierungsgruppen werden für Workloads empfohlen, für die eine niedrige Netzwerklatenz bzw. ein hoher Durchsatz von Vorteil sind.

Latenzempfindliche Dienste werden an Edge-Standorten über ein globales AWS-Netzwerk bereitgestellt, z. B. Amazon CloudFront. Diese Edge-Standorte verfügen in der Regel über Services wie ein Content Delivery Network (CDN) und Domain Name System (DNS). Durch die Platzierung am Edge können die Workloads mit geringer Latenz auf Anforderungen zu Inhalten

oder zur DNS-Auflösung reagieren. Es sind auch geografische Services wie das Geo-Targeting von Inhalten (Bereitstellung unterschiedlicher Inhalte gemäß dem Standort von Endbenutzern) oder die latenzbasierte Weiterleitung von Endbenutzern zur nächsten Region (minimale Latenz) verfügbar.

Verwenden Sie Edge-Services, um die Latenz zu reduzieren und das Caching von Inhalten zu ermöglichen. Konfigurieren Sie die Cache-Steuerung für DNS und HTTP/HTTPS richtig, um aus diesen Ansätzen den größtmöglichen Nutzen zu ziehen.

Implementierungsschritte

- Erfassen Sie Informationen über den an den Netzwerkschnittstellen ein- und ausgehenden IP-Datenverkehr.
  - Protokollierung von IP-Datenverkehr mithilfe von VPC Flow Logs
  - Wie Sie die Client-IP-Adresse in AWS Global Accelerator beibehalten
- Analysieren Sie die Netzwerkzugriffsmuster in Ihrem Workload, um zu ermitteln, wie die Benutzer Ihre Anwendung verwenden.
  - Verwenden Sie Überwachungstools wie Amazon CloudWatch und AWS CloudTrail, um Daten zu Netzwerkaktivitäten zu erfassen.
  - Analysen Sie die Daten, um das Netzwerkzugriffsmuster zu identifizieren.
- Wählen Sie Regionen für Ihre Workload-Bereitstellung auf der Grundlage der folgenden zentralen Elemente aus:
  - Dem Speicherort Ihrer Daten: Für datenintensive Anwendungen (wie Big Data oder Machine Learning) sollte der Anwendungscode so nahe wie möglich zu den Daten ausgeführt werden.
  - Den Standorten Ihrer Benutzer:innen: Wählen Sie bei nutzerorientierten Anwendungen eine Region (oder Regionen) möglichst nahe an den Benutzer:innen Ihres Workloads.
  - Anderen einschränkenden Faktoren: Denken Sie dabei etwa an Kosten und Compliance, wie in Überlegungen bei der Auswahl einer Region für Ihren Workload beschrieben.
- Verwenden Sie AWS Local Zones für Workloads wie Video-Rendering. Mit Local Zones können Sie von allen Vorteilen profitieren, die sich durch die Platzierung der Datenverarbeitungs- und Speicherressourcen in der Nähe Ihrer Endbenutzer ergeben.
- Verwenden Sie AWS Outposts für Workloads, die On-Premises verarbeitet werden müssen und die Sie nahtlos mit Ihren restlichen Workloads in AWS ausführen möchten.
- Anwendungen wie hochauflösendes Live-Video-Streaming, High-Fidelity-Audio und Augmented Reality oder Virtual Reality (AR/VR) erfordern extrem niedrige Latenzen für 5G-Geräte. Ziehen Sie für solche Anwendungen AWS Wavelength in Betracht. AWS Wavelength bettet AWS-Computing-

und Speicher-Services in 5G-Netzwerke ein und bietet eine mobile Edge-Computing-Infrastruktur für die Entwicklung, Bereitstellung und Skalierung von Anwendungen mit extrem niedriger Latenz.

- Verwenden Sie lokale Zwischenspeicherung oder AWS-Zwischenspeicherungslösungen für häufig genutzte Ressourcen zur Verbesserung der Leistung, zur Verringerung von Datenverschiebungen und zur Reduzierung der Umweltauswirkungen.

Service	When to use
Amazon CloudFront	Verwenden Sie dies für die Zwischens peicherung statischer Inhalte wie Bilder, Skripts und Videos sowie dynamischer Inhalte wie API-Antworten oder Webanwendungen.
Amazon ElastiCache	Verwenden Sie dies für die Zwischens peicherung von Inhalten für Webanwend ungen.
DynamoDB Accelerator	Verwenden Sie dies für die Add-in-Speicher-Beschleunigung für Ihre DynamoDB-Tabellen.

- Nutzen Sie Services, die Ihnen dabei helfen können, Code näher an den Nutzern Ihres Workloads auszuführen:

Service	When to use
Lambda@edge	Verwenden Sie dies für rechenintensive Anwendungen, die initiiert werden, wenn sich Objekte nicht im Zwischenspeicher befinden.
Amazon CloudFront-Funktionen	Verwenden Sie diese für einfache Anwendung sfälle wie HTTP(s)-Anfragen oder Antwortma nipulationen, die von kurzlebigen Funktionen initiiert werden können.
AWS IoT Greengrass	Verwenden Sie dies für die Ausführung lokaler Rechenoperationen, Messaging sowie die

Service	When to use
	Datenzwischenspeicherung für verbundene Geräte.

- Einige Anwendungen benötigen feste Zugangspunkte oder eine höhere Leistung. Bei diesen müssen First-Byte-Latenz der Jitter verringert und der Durchsatz erhöht werden. Diese Anwendungen können von Netzwerk-Services profitieren, die statische Anycast-IP-Adressen und eine TCP-Terminierung an Edge-Standorten bieten. AWS Global Accelerator kann die Leistung Ihrer Anwendungen um bis zu 60 % verbessern und bietet ein schnelles Failover für Architekturen mit mehreren Regionen. AWS Global Accelerator stellt Ihnen statische Anycast-IP-Adressen zur Verfügung, die als fester Zugangspunkt für Ihre Anwendungen dienen, die in einer oder mehreren AWS-Regionen gehostet werden. Diese IP-Adressen sorgen dafür, dass Datenverkehr so nah wie möglich an Ihren Benutzern in das globale AWS-Netzwerk eingebunden wird. AWS Global Accelerator reduziert die Zeit für den anfänglichen Verbindungsaufbau, indem eine TCP-Verbindung zwischen dem Client und dem AWS-Edge-Standort hergestellt wird, der dem Client am nächsten liegt. Prüfen Sie die Verwendung von AWS Global Accelerator, um die Leistung Ihrer TCP/UDP-Workloads zu verbessern und einen schnellen Failover für Architekturen mit mehreren Regionen zu ermöglichen.

Ressourcen

Zugehörige bewährte Methoden:

- COST07-BP02 Implementieren von Regionen auf Basis der Kosten
- COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten
- REL10-BP01 Bereitstellen des Workloads an mehreren Standorten
- REL10-BP02 Auswählen der geeigneten Standorte für Ihre Multi-Standort-Bereitstellung
- SUS01-BP01 Auswählen der Region auf Grundlage von Unternehmensanforderungen und Nachhaltigkeitszielen
- SUS02-BP04 Optimieren der geografischen Platzierung von Workloads auf der Grundlage ihrer Netzwerkanforderungen
- SUS04-BP07 Minimieren von Datenübertragungen zwischen Netzwerken

Zugehörige Dokumente:

- Globale AWS-Infrastruktur
- AWS Local Zones und AWS Outposts Outposts: Die Auswahl der richtigen Technologie für Ihren Edge-Workload
- Platzierungsgruppen
- AWS Local Zones
- AWS Outposts
- AWS Wavelength
- Amazon CloudFront
- AWS Global Accelerator
- AWS Direct Connect
- AWS Site-to-Site VPN
- Amazon Route 53

Zugehörige Videos:

- Erklärungsvideo zu AWS Local Zones
- AWS Outposts: Übersicht und Funktionsweise
- AWS re:Invent 2023 – Eine Migrationsstrategie für Edge- und On-Premises-Workloads
- AWS re:Invent 2021 – AWS Outposts: Das AWS AWS Erlebnis on-premises
- AWS re:Invent 2020: AWS Wavelength: Apps mit ultraniedriger Latenz am 5G-Edge ausführen
- AWS re:Invent 2022 – AWS Local Zones: Entwickeln von Anwendungen für einen verteilten Edge
- AWS re:Invent 2021 – Entwicklung von Websites mit niedriger Latenz mit Amazon CloudFront
- AWS re:Invent 2022 – Verbessern der Leistung und Verfügbarkeit mit AWS Global Accelerator
- AWS re:Invent 2022 – Aufbau Ihres globalen Wide Area Networks mit AWS
- AWS re:Invent 2020: Globales Datenverkehrsmanagement mit Amazon Route 53

Zugehörige Beispiele:

- AWS Global Accelerator Workshop für benutzerdefiniertes Routing
- Verarbeitung von Rewrites und Redirects mit Edge-Funktionen

PERF04-BP07 Optimieren der Netzwerkkonfiguration basierend auf Metriken

Treffen Sie anhand der erfassten und analysierten Daten fundierte Entscheidungen zum Optimieren Ihrer Netzwerkkonfiguration.

Typische Anti-Muster:

- Sie gehen davon aus, dass alle leistungsbezogenen Probleme auf Anwendungen zurückzuführen sind.
- Sie testen die Netzwerkleistung ausschließlich an einem Standort nahe der Stelle, an der Sie die Workload bereitgestellt haben.
- Sie verwenden Standardkonfigurationen für alle Netzwerk-Services.
- Sie führen eine Überdimensionierung der Netzwerkressourcen durch, um eine ausreichende Kapazität zu gewährleisten.

Vorteile der Nutzung dieser bewährten Methode: Das Sammeln der erforderlichen Metriken Ihres AWS-Netzwerks und die Implementierung von Tools zur Überwachung des Netzwerks bieten Ihnen die Möglichkeit, die Leistung des Netzwerks zu ermitteln und die Netzwerkkonfigurationen zu optimieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Die Überwachung des Datenverkehrs von und zu VPCs, Subnetzen oder Netzwerkschnittstellen ist für das Verständnis der Nutzung von AWS-Netzwerkressourcen und zur Optimierung von Netzwerkkonfigurationen entscheidend. Mit den folgenden AWS-Networking-Tools können Sie Informationen über die Nutzung des Datenverkehrs, den Netzwerkzugriff und die Protokolle genauer untersuchen.

Implementierungsschritte

- Identifizieren Sie die wichtigsten Leistungsmetriken wie Latenz oder Paketverlust, die erfasst werden müssen. AWS bietet mehrere Tools, die Ihnen bei der Erfassung dieser Messwerte helfen können. Mit den folgenden Tools können Sie Informationen über die Nutzung des Datenverkehrs, den Netzwerkzugriff und die Protokolle genauer untersuchen:

AWS-Tool	Aktion
Amazon VPC IP Address Manager.	Nutzen Sie IPAM, um Ihre IP-Adressen für Ihre AWS- und On-Premises-Workloads zu planen, nachzuverfolgen und zu überwachen. Dies ist eine bewährte Methode zur Optimierung der Nutzung und Zuweisung von IP-Adressen.
VPC Flow Logs	Nutzen Sie VPC Flow Logs, um detaillierte Informationen über den Datenverkehr zu und von den Netzwerkschnittstellen in Ihren VPCs zu protokollieren. Mit VPC Flow Logs können Sie zu restriktive oder zu freizügige Regeln für Sicherheitsgruppen diagnostizieren und die Richtung des Datenverkehrs zu und von den Netzwerkschnittstellen ermitteln.
AWS Transit Gateway Flow Logs	Nutzen Sie AWS Transit Gateway Flow Logs, um Informationen über den IP-Datenverkehr zu und von Ihren Transit-Gateways zu erfassen.
DNS-Abfrageprotokollierung	Protokollieren Sie Informationen über von Route 53 empfangene öffentliche oder private DNS-Abfragen. Mit DNS-Protokollen können Sie DNS-Konfigurationen optimieren, indem Sie die angefragte Domäne oder Subdomäne bzw. die Route 53-Edge-Standorte, die auf DNS-Abfragen geantwortet haben, nachvollziehen.

AWS-Tool	Aktion
Reachability Analyzer	Reachability Analyzer hilft Ihnen, die Erreichbarkeit des Netzwerks zu analysieren und zu debuggen. Reachability Analyzer ist ein Konfigurationsanalyse-Tool, mit dem Sie die Konnektivität zwischen einer Quelle und einer Zielressource in Ihren VPCs testen können. Mit diesem Tool können Sie überprüfen, ob Ihre Netzwerkkonfiguration der geplanten Konnektivität entspricht.
Network Access Analyzer	Network Access Analyzer hilft Ihnen, den Netzwerkzugriff auf Ihre Ressourcen zu verstehen. Mit Network Access Analyzer können Sie Ihre Anforderungen an den Netzwerkzugriff spezifizieren und potenzielle Netzwerkpfade identifizieren, die Ihren Anforderungen nicht entsprechen. Indem Sie Ihre entsprechende Netzwerkkonfiguration optimieren, können Sie den Zustand Ihres Netzwerks nachvollziehen und überprüfen und belegen, dass Ihr AWS-Netzwerk Ihre Compliance-Anforderungen erfüllt.

AWS-Tool	Aktion
Amazon CloudWatch	Nutzen Sie Amazon CloudWatch und aktivieren Sie geeignete Metriken für Netzwerkoptionen. Stellen Sie sicher, dass Sie die richtige Netzwerk-Metrik für Ihren Workload auswählen. Sie können zum Beispiel Metriken für die VPC-Netzwerkadressennutzung, VPC-NAT-Gateways, AWS Transit Gateway, VPN-Tunnel, AWS Network Firewall, Elastic Load Balancing und AWS Direct Connect aktivieren. Die kontinuierliche Überwachung von Metriken ist eine gute Vorgehensweise, um den Status und die Nutzung Ihres Netzwerks zu beobachten und nachzuvollziehen. Sie hilft Ihnen, die Netzwerkkonfiguration auf der Basis Ihrer Beobachtungen zu optimieren.
AWS Network Manager	Mithilfe von AWS Network Manager können Sie die Echtzeit- und vergangene Leistung des AWS Global Network für betriebliche und planerische Zwecke überwachen. Network Manager bietet aggregierte Netzwerklatenz zwischen AWS-Regionen und Availability Zones sowie innerhalb jeder Availability Zone, wodurch Sie besser verstehen können, wie Ihre Anwendungsleistung mit der Leistung des zugrunde liegenden AWS-Netzwerks zusammenhängt.
Amazon CloudWatch RUM	Verwenden Sie Amazon CloudWatch RUM, um die Metriken zu erfassen, die Ihnen die Erkenntnisse liefern, mit denen Sie die Benutzererfahrung identifizieren, verstehen und verbessern können.

- Identifizieren Sie mithilfe von VPC und AWS Transit Gateway Flow Logs Top-Talker und Muster des Anwendungsdatenverkehrs.
- Beurteilen und optimieren Sie Ihre aktuelle Netzwerkarchitektur, einschließlich VPCs, Subnetze und Routing. Sie können beispielsweise bewerten, wie unterschiedliches VPC-Peering oder AWS Transit Gateway Ihnen helfen können, das Netzwerk in Ihrer Architektur zu verbessern.
- Untersuchen Sie die Routingpfade in Ihrem Netzwerk, um sicherzustellen, dass immer der kürzeste Pfad zwischen Zielen verwendet wird. Network Access Analyzer kann Ihnen dabei helfen.

Ressourcen

Zugehörige Dokumente:

- Öffentliche DNS-Abfrageprotokollierung
- Was ist IPAM?
- Was ist Reachability Analyzer?
- Was ist Network Access Analyzer?
- CloudWatch-Metriken für Ihre VPCs
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format (Optimieren der Leistung und Reduzieren der Kosten für die Netzwerk-Analytik mit VPC Flow Logs im Apache Parquet-Format)
- Monitoring your global and core networks with Amazon CloudWatch metrics (Überwachen von globalen und Kernnetzwerken mit Amazon CloudWatch-Metriken)
- Continuously monitor network traffic and resources (Kontinuierliches Überwachen von Netzwerkdatenverkehr und -ressourcen)

Zugehörige Videos:

- AWS re:Invent 2023 – Leitfaden für Entwickler von Cloud-Netzwerken
- AWS re:Invent 2023 – Sind Sie bereit für Neues? Gestaltung von Netzwerken für Wachstum und Flexibilität
- AWS re:Invent 2023 – Erweiterte VPC-Designs und neue Funktionen
- AWS re:Invent 2022 – Ausführliche Beschreibung der AWS-Netzwerkinfrastruktur
- AWS re:Invent 2020 – Bewährte Methoden für Netzwerke und Tipps für das AWS Well-Architected Framework

- AWS re:Invent 2020 – Überwachen des Netzwerkdatenverkehrs und Fehlerbehebung

Zugehörige Beispiele:

- Workshops zu AWS-Netzwerken
- Überwachung des AWS-Netzwerks
- Beobachten und Diagnostizieren Ihres Netzwerks in AWS
- Finden und Beheben von Netzwerkfehlkonfigurationen in AWS

Prozess und Kultur

LEIST 5. Wie tragen Ihre Unternehmenspraktiken und Ihre Unternehmenskultur zur Leistungseffizienz Ihres Workloads bei?

Bei der Architektur von Workloads gibt es Prinzipien und Praktiken, die Sie übernehmen können, um effiziente und leistungsstarke Cloud-Workloads besser zu betreiben. Um eine Kultur zu schaffen, die die Leistungseffizienz von Cloud-Workloads fördert, sollten Sie diese Schlüsselprinzipien und -praktiken berücksichtigen:

Bewährte Methoden

- PERF05-BP01 Festlegen wichtiger Leistungskennzahlen (KPIs) zum Messen des Zustands und der Leistung des Workloads
- PERF05-BP02 Verwenden von Überwachungslösungen, um Bereiche mit kritischem Leistungsbedarf zu identifizieren
- PERF05-BP03 Definieren eines Prozesses zum Verbessern der Workload-Leistung
- PERF05-BP04 Durchführen von Lasttests für den Workload
- PERF05-BP05 Verwenden von Automatisierung zur proaktiven Behebung leistungsbezogener Probleme
- PERF05-BP06 Konstantes Aktualisieren des Workloads und der Services
- PERF05-BP07 Regelmäßiges Überprüfen von Metriken

PERF05-BP01 Festlegen wichtiger Leistungskennzahlen (KPIs) zum Messen des Zustands und der Leistung des Workloads

Identifizieren Sie die KPIs, die die Workload-Leistung quantitativ und qualitativ messen. Mithilfe von KPIs können Sie den Zustand und die Leistung eines Workloads im Zusammenhang mit einem Geschäftsziel messen.

Typische Anti-Muster:

- Sie überwachen nur Metriken auf Systemebene, um Erkenntnisse über Ihren Workload zu gewinnen, und verstehen den geschäftlichen Einfluss dieser Metriken nicht.
- Sie gehen davon aus, dass Ihre KPIs bereits als standardmäßige Metrikdaten veröffentlicht und geteilt werden.
- Sie definieren keinen quantitativen, messbaren KPI.
- Sie richten KPIs nicht an Geschäftszielen oder -strategien aus.

Vorteile der Nutzung dieser bewährten Methode: Die Identifizierung spezifischer KPIs, die den Zustand und die Leistung des Workloads widerspiegeln, hilft Teams dabei, ihre Prioritäten auszurichten und erfolgreiche Geschäftsergebnisse zu definieren. Das Teilen dieser Metriken mit allen Abteilungen bietet Sichtbarkeit und die Ausrichtung an Grenzwerten, Erwartungen und Geschäftsauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

KPIs helfen Business- und Entwicklungsteams, das Messen von Zielen und Strategien abzustimmen und festzustellen, wie diese Faktoren gemeinsam zu Geschäftsergebnissen beitragen. Beispielsweise könnte ein Website-Workload die Ladezeit der Seite als Indikator für die Gesamtleitstung heranziehen. Diese Metrik wäre einer von mehreren Datenpunkten, mit denen das Benutzererlebnis gemessen wird. Zusätzlich zum Ermitteln der Grenzwerte für Seitenladezeiten sollten Sie das gewünschte Resultat dokumentieren bzw. das Geschäftsrisiko, wenn die ideale Leistung nicht erreicht wird. Die lange Ladezeit einer Seite betrifft Ihre Endbenutzer direkt, verringert die Bewertung ihres Benutzererlebnisses und kann zu einem Verlust von Kunden führen. Kombinieren Sie beim Definieren Ihrer KPI-Grenzwerte die Benchmarks der Branche und die Erwartungen Ihrer Endbenutzer. Beispielsweise, wenn die aktuelle Benchmark der Branche das Laden einer Webseite innerhalb von zwei Sekunden ist, Ihre Endbenutzer aber erwarten, dass eine Webseite innerhalb

von einer Sekunde geladen wird, sollten Sie beim Einrichten des KPI beide Datenpunkte in Betracht ziehen.

Ihr Team muss Ihre Workload-KPIs mithilfe von detaillierten Echtzeitdaten und historischen Daten als Referenz evaluieren und Dashboards erstellen, die Metrikberechnungen für Ihre KPI-Daten durchführen, um Einblicke in Betrieb und Auslastung zu erhalten. KPIs sollten dokumentiert werden und Grenzwerte enthalten, die Geschäftsziele und -strategien unterstützen, und sie sollten den Metriken zugeordnet sein, die überwacht werden. KPIs sollten erneut aufgegriffen werden, wenn sich Geschäftsziele, Strategien oder Anforderungen von Endbenutzern ändern.

Implementierungsschritte

- Stakeholder identifizieren: Identifizieren und dokumentieren Sie wichtige Stakeholder im Unternehmen, einschließlich Entwicklungs- und Betriebsteams.
- Ziele definieren: Arbeiten Sie mit diesen Stakeholdern zusammen, um die Ziele Ihres Workloads zu definieren und zu dokumentieren. Berücksichtigen Sie die kritischen Leistungsaspekte Ihrer Workloads, wie Durchsatz, Reaktionszeit und Kosten, sowie Geschäftsziele wie die Benutzerzufriedenheit.
- Bewährte Methoden der Branche überprüfen: Sehen Sie sich in der Branche bewährte Methoden an, um relevante KPIs zu identifizieren, die auf Ihre Workload-Ziele abgestimmt sind.
- Metriken Identifizieren: Identifizieren Sie Metriken, die mit Ihren Workload-Zielen übereinstimmen und Ihnen helfen können, Leistung und Geschäftsziele zu messen. Richten Sie KPIs basierend auf diesen Metriken ein. Beispiele für Metriken sind Messungen wie die durchschnittliche Reaktionszeit oder die Anzahl gleichzeitiger Benutzer:innen.
- KPIs definieren und dokumentieren: Verwenden Sie in der Branche bewährte Methoden und Ihre Workload-Ziele, um Ziele für Ihren Workload-KPI festzulegen. Verwenden Sie diese Informationen, um KPI-Schwellenwerte für Schweregrad oder Alarmebene festzulegen. Identifizieren und dokumentieren Sie das Risiko und die Auswirkungen, wenn ein KPI nicht erreicht wird.
- Überwachung implementieren: Verwenden Sie Überwachungstools wie Amazon CloudWatch oder AWS Config, um Metriken zu erfassen und KPIs zu messen.
- KPIs visuell kommunizieren: Verwenden Sie Dashboard-Tools wie Amazon QuickSight, um KPIs anzuzeigen und Stakeholdern mitzuteilen.
- Analysieren und optimieren: Überprüfen und analysieren Sie regelmäßig KPIs, um Bereiche Ihres Workloads zu identifizieren, die verbessert werden müssen. Arbeiten Sie mit den Stakeholdern zusammen, um diese Verbesserungen umzusetzen.

- Wiederaufgreifen und nachbessern: Überprüfen Sie regelmäßig Metriken und KPIs, um ihre Effektivität zu bewerten, insbesondere wenn sich die Geschäftsziele oder die Workload-Leistung ändern.

Ressourcen

Zugehörige Dokumente:

- CloudWatch-Dokumentation
- Überwachung, Protokollierung und Leistung von AWS Partners
- AWS-Tools zur Beobachtbarkeit
- Die Bedeutung von Key Performance Indicators (KPIs) für groß angelegte Cloud-Migrationen
- Wie Sie mit dem KPI-Dashboard Ihre KPIs zur Kostenoptimierung nachverfolgen
- X-Ray-Dokumentation
- Verwendung von Amazon CloudWatch-Dashboards
- QuickSight-KPIs

Zugehörige Videos:

- AWS re:Invent 2023 – Kosten- und Leistungsoptimierung sowie Fortschrittsverfolgung bei der Schadensbegrenzung
- AWS re:Invent 2023 – Verwaltung von Ereignissen im Ressourcenlebenszyklus im großen Maßstab mit AWS Health
- AWS re:Invent 2023 – Leistung und Effizienz bei Pinterest: Optimierung der neuer Instances
- AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse
- AWS re:Invent 2023 – Aufbau einer effektiven Beobachtbarkeitsstrategie
- AWS Summit SF 2022 – Full-Stack-Beobachtbarkeit und -Überwachung von Anwendungen mit AWS
- AWS re:Invent 2023 – Skalierung in AWS für die ersten 10 Millionen Benutzer:innen
- AWS re:Invent 2022 – Wie Amazon bessere Metriken für eine höhere Website-Leistung verwendet
- Erstellung einer effektiven Metrikenstrategie für Ihr Unternehmen | AWS-Ereignisse

Zugehörige Beispiele:

- Erstellen eines Dashboards mit QuickSight

PERF05-BP02 Verwenden von Überwachungslösungen, um Bereiche mit kritischem Leistungsbedarf zu identifizieren

Ermitteln Sie die Bereiche, in denen sich durch Steigern der Workload-Leistung positive Auswirkungen auf die Effizienz oder den Kundenkomfort realisieren lassen. Beispiel: Eine Website mit zahlreichen Kundeninteraktionen kann von der Nutzung von Edge-Services profitieren, indem Inhalte näher bei den Kunden bereitgestellt werden.

Typische Anti-Muster:

- Sie gehen davon aus, dass standardmäßige Datenverarbeitungsmetriken wie CPU-Auslastung oder Arbeitsspeicherdruck ausreichen, um Leistungsprobleme zu erfassen.
- Sie verwenden nur die Standardmetriken, die von der Überwachungssoftware Ihrer Wahl aufgezeichnet wurden.
- Sie überprüfen Metriken nur dann, wenn ein Problem vorliegt.

Vorteile der Nutzung dieser bewährten Methode: Das eingehende Verständnis kritischer Bereiche hilft Workload-Eigentümern dabei, KPIs zu überwachen und Verbesserungen mit größeren Auswirkungen zu priorisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Richten Sie durchgehende Nachverfolgung ein, um Datenverkehrsmuster, Latenz und kritische Leistungsbereiche zu identifizieren. Überwachen Sie Ihre Datenzugriffsmuster auf langsame Abfragen oder schlecht fragmentierte und partitionierte Daten. Identifizieren Sie problematische Workload-Bereiche mithilfe von Lasttests oder -überwachung.

Erhöhen Sie die Leistungseffizienz durch eingehendes Verständnis Ihrer Architektur, der Datenverkehrs- und der Datenzugriffmuster und identifizieren Sie Ihre Latenz- und Verarbeitungszeiten. Identifizieren Sie potenzielle Engpässe, die sich bei zunehmenden Workloads auf den Kundenkomfort auswirken könnten. Nachdem Sie diese Bereiche untersucht haben, sollten Sie prüfen, welche Lösung Sie nutzen können, um diese Leistungsprobleme zu beseitigen.

Implementierungsschritte

- Richten Sie durchgehende Überwachung ein, um alle Workload-Komponenten und -Metriken zu erfassen. Hier finden Sie Beispiele für Überwachungslösungen in AWS.

Service	Where to use
Amazon CloudWatch Real-User Monitoring (RUM)	To capture application performance metrics from real user client-side and frontend sessions.
AWS X-Ray	To trace traffic through the application layers and identify latency between components and dependencies. Use X-Ray service maps to see relationships and latency between workload components.
Amazon Relational Database Service Performance Insights	To view database performance metrics and identify performance improvements.
Amazon RDS Erweiterte Überwachung	To view database OS performance metrics.
Amazon DevOps Guru	To detect abnormal operating patterns so you can identify operational issues before they impact your customers.

- Führen Sie Tests durch, um Metriken zu generieren sowie Datenverkehrsmuster, Engpässe und kritische Leistungsbereiche zu identifizieren. Hier finden Sie einige Beispiele zum Durchführen von Tests:
  - Richten Sie CloudWatch Synthetic Canaries ein, um browserbasierte Benutzeraktivitäten programmgesteuert mit Linux-Cron-Aufträgen oder Ratenausdrücken nachzuahmen und im Zeitverlauf konsistente Metriken zu erhalten.
  - Verwenden Sie die Lösung AWS Distributed Load Testing, um Spitzendatenverkehr zu generieren oder Workloads mit der erwarteten Wachstumsrate zu testen.
- Evaluieren Sie die Metriken und die Telemetriedaten, um Ihre kritischen Leistungsbereiche zu identifizieren. Prüfen Sie diese Bereiche zusammen mit Ihrem Team und besprechen Sie Überwachung und Lösung zur Vermeidung von Engpässen.

- Experimentieren Sie mit Leistungsverbesserungen und messen Sie diese Änderungen anhand von Daten. Beispielsweise können Sie CloudWatch Evidently verwenden, um neue Verbesserungen und Leistungsauswirkungen auf Ihren Workload zu testen.

Ressourcen

Zugehörige Dokumente:

- Neuheiten im Bereich AWS-Beobachtbarkeit bei der re:Invent 2023
- Amazon Builders' Library
- X-Ray-Dokumentation
- Amazon CloudWatch RUM
- Amazon DevOps Guru

Zugehörige Videos:

- AWS re:Invent 2023 – [LAUNCH] Anwendungsüberwachung für moderne Workloads
- AWS re:Invent 2023 – Implementierung der Anwendungsbeobachtbarkeit
- AWS re:Invent 2023 – Aufbau einer effektiven Beobachtbarkeitsstrategie
- AWS Summit SF 2022 – Full-Stack-Beobachtbarkeit und -Überwachung von Anwendungen mit AWS
- AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse
- AWS re:Invent 2022 – Die Amazon Builders' Library: 25 Jahre operative Exzellenz von Amazon
- AWS re:Invent 2022 – Wie Amazon bessere Metriken für eine höhere Website-Leistung verwendet
- Visuelle Überwachung von Anwendungen mit Amazon CloudWatch Synthetics

Zugehörige Beispiele:

- Messen der Seitenladezeit mit Amazon CloudWatch Synthetics
- Amazon CloudWatch RUM Web Client
- X-Ray SDK for Python
- Verteilte Lasttests in AWS

PERF05-BP03 Definieren eines Prozesses zum Verbessern der Workload-Leistung

Definieren Sie einen Prozess, mit dem sich neu verfügbare Services, Designmuster, Ressourcentypen und Konfigurationen bewerten lassen. Führen Sie beispielsweise vorhandene Leistungstests für neue Instance-Angebote durch, um zu ermitteln, welche Verbesserungen sich für Ihre Workload ergeben.

Typische Anti-Muster:

- Sie gehen davon aus, dass Ihre aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert wird.
- Sie führen im Laufe der Zeit Änderungen an der Architektur ein, ohne sie begründen.

Vorteile der Nutzung dieser bewährten Methode: Durch einen definierten Prozess zum Ändern der Architektur können Sie die gesammelten Daten langfristig in die Gestaltung Ihres Workloads einfließen lassen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Für Ihren Workload gibt es einige wesentliche Einschränkungen. Dokumentieren Sie diese, damit Sie besser einschätzen können, durch welche Art von Innovation die Leistung Ihres Workloads gesteigert werden könnte. Ziehen Sie diese Informationen heran, wenn Sie von neuen verfügbaren Services oder Technologien erfahren, um Möglichkeiten zur Beseitigung von Einschränkungen oder Engpässen zu identifizieren.

Identifizieren Sie wesentliche Leistungseinschränkungen für Ihren Workload. Dokumentieren Sie die Leistungseinschränkungen Ihrer Workload, damit Sie besser einschätzen können, durch welche Art von Innovation die Leistung Ihrer Workload ggf. gesteigert werden kann.

Implementierungsschritte

- KPIs identifizieren: Identifizieren Sie Ihre Workload-Leistungs-KPIs, wie unter PERF05-BP01 Festlegen wichtiger Leistungskennzahlen (KPIs) zum Messen des Zustands und der Leistung des Workloads beschrieben, um eine Baseline für Ihren Workload zu erstellen.
- Überwachung implementieren: Verwenden Sie AWS-Tools zur Beobachtbarkeit, um Leistungsmetriken zu erfassen und KPIs zu messen.

- Analysen durchführen: Führen Sie eine eingehende Analyse durch, um leistungsschwache Bereiche (wie Konfiguration und Anwendungscode) in Ihrem Workload zu identifizieren, wie beschrieben unter PERF05-BP02 Verwenden von Überwachungslösungen, um Bereiche mit kritischem Leistungsbedarf zu identifizieren. Verwenden Sie Analyse- und Leistungs-Tools, um die Strategie zur Leistungsverbesserung zu identifizieren.
- Verbesserungen bestätigen: Verwenden Sie Sandbox- oder Vorproduktionsumgebungen, um die Wirksamkeit von Verbesserungsstrategien zu überprüfen.
- Änderungen implementieren: Implementieren Sie die Änderungen in der Produktion und überwachen Sie kontinuierlich die Leistung des Workloads. Dokumentieren Sie die Verbesserungen und teilen Sie die Änderungen den Stakeholdern mit.
- Wiederaufgreifen und Verfeinern: Überprüfen Sie regelmäßig Ihren Leistungsverbesserungsprozess, um Verbesserungsmöglichkeiten zu identifizieren.

Ressourcen

Zugehörige Dokumente:

- AWS-Blog
- Neuerungen bei AWS
- AWS Skill Builder

Zugehörige Videos:

- AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen
- AWS re:Invent 2023 – Kosten- und Leistungsoptimierung sowie Fortschrittsverfolgung bei der Schadensbegrenzung
- AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse
- AWS re:Invent 2022 — Optimierung Ihrer AWS-Workloads mit Anleitungen für bewährte Methoden

Zugehörige Beispiele:

- AWS Github

PERF05-BP04 Durchführen von Lasttests für den Workload

Führen Sie für den Workload Lasttests durch, um sicherzustellen, dass er die Produktionslast bewältigen kann, und identifizieren Sie Leistungsengpässe.

Typische Anti-Muster:

- Sie führen Lasttests für einzelne Teile der Workload durch, aber nicht für die gesamte Workload.
- Sie führen Lasttests in einer Infrastruktur durch, die sich von Ihrer Produktionsumgebung unterscheidet.
- Sie führen Lasttests nur für die erwartete Last durch und nicht für noch größere Lasten, um mögliche künftige Probleme besser vorherzusehen.
- Sie führen Lasttests durch, ohne die Amazon EC2-Testrichtlinien zu lesen oder ein Formular für die Einreichung simulierter Ereignisse abzusenden. Dies führt dazu, dass Ihr Test nicht ausgeführt werden kann, da er wie ein Denial-of-Service-Ereignis aussieht.

Vorteile der Nutzung dieser bewährten Methode: Die Messung der Leistung im Rahmen eines Lasttests gibt Aufschluss darüber, wo bei zunehmender Last mit Auswirkungen zu rechnen ist. Auf diese Weise können Sie erforderliche Änderungen vorhersehen, bevor sie sich auf Ihre Workload auswirken.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Lasttests in der Cloud sind ein Prozess zur Messung der Leistung eines Cloud-Workloads unter realistischen Bedingungen mit erwarteter Benutzerlast. Dieser Prozess beinhaltet die Bereitstellung einer produktionsähnlichen Cloud-Umgebung, die Verwendung von Lasttest-Tools zur Lastgenerierung und die Analyse von Metriken, um die Fähigkeit Ihres Workloads zu bewerten, mit einer realistischen Last umzugehen. Verwenden Sie für Lasttests synthetische oder bereinigte Daten und entfernen Sie sensible oder personenbezogene Informationen. Führen Sie automatisch Lasttests als Teil Ihrer Bereitstellungs-Pipeline durch und vergleichen Sie die Ergebnisse mit vordefinierten KPIs und Schwellenwerten. Dieser Prozess hilft Ihnen dabei, die erforderliche Leistung weiterhin zu erreichen.

Implementierungsschritte

- Testziele definieren: Identifizieren Sie die Leistungsaspekte Ihres Workloads, das Sie bewerten möchten, z. B. Durchsatz und Reaktionszeit.

- Testtool auswählen: Wählen und konfigurieren Sie das Lasttest-Tool, das zu Ihrem Workload passt.
- Umgebung einrichten: Richten Sie die Testumgebung basierend auf Ihrer Produktionsumgebung ein. Mithilfe von AWS-Services können Sie Umgebungen im Produktionsmaßstab ausführen und damit Ihre Architektur testen.
- Überwachung implementieren: Verwenden Sie Überwachungstools wie Amazon CloudWatch, um Metriken für alle Ressourcen in Ihrer Architektur zu erfassen. Sie können auch benutzerdefinierte Metriken erfassen und veröffentlichen.
- Szenarien definieren: Definieren Sie die Szenarien und Parameter der Lasttests (wie Testdauer und Anzahl der Benutzer:innen).
- Lasttests durchführen: Führen Sie Testszenarien in großem Maßstab durch. Testen Sie Ihren Workload mithilfe der AWS Cloud, um zu ermitteln, an welcher Stelle er nicht skalierbar ist oder ob die Skalierung nichtlinear erfolgt. Nutzen Sie beispielsweise Spot Instances, um kostengünstig Lasten zu erzeugen und Engpässe zu identifizieren, bevor diese in der Produktionsumgebung auftreten.
- Testergebnisse analyieren: Analysieren Sie die Ergebnisse, um Leistungsengpässe und Verbesserungsmöglichkeiten zu identifizieren.
- Erkenntnisse dokumentieren und teilen: Erkenntnisse und Empfehlungen dokumentieren und kommunizieren. Teilen Sie diese Informationen mit Stakeholdern, um ihnen zu helfen, fundierte Entscheidungen über Strategien zur Leistungsoptimierung zu treffen.
- Kontinuierlich iterieren: Lasttests sollten in regelmäßigen Abständen durchgeführt werden, insbesondere nach einem Systemwechsel oder Update.

Ressourcen

Zugehörige Dokumente:

- Amazon CloudWatch RUM
- Amazon CloudWatch Synthetics
- Verteilte Lasttests in AWS

Zugehörige Videos:

- AWS Summit ANZ 2023: Mit AWS Distributed Load Testing zuversichtlich in die Zukunft starten
- AWS re:Invent 2022: Skalierung in Scaling AWS für Ihre ersten 10 Millionen Benutzer:innen

- Lösen mit AWS-Solutions: Verteilte Lasttests

- AWS re:Invent 2021 – Optimierung von Anwendungen durch Endbenutzereinblicke mit Amazon CloudWatch RUM
- Demo von Amazon CloudWatch Synthetics

Zugehörige Beispiele:

- Verteilte Lasttests in AWS

PERF05-BP05 Verwenden von Automatisierung zur proaktiven Behebung leistungsbezogener Probleme

Verwenden Sie wichtige Leistungskennzahlen (KPIs) in Kombination mit Überwachungs- und Warnsystemen, um eine proaktive Behandlung leistungsbezogener Probleme zu ermöglichen.

Typische Anti-Muster:

- Sie geben dem Betriebspersonal nur die Möglichkeit, betriebliche Änderungen an der Workload vorzunehmen.
- Sie lassen alle Alarme ohne proaktive Behebung zum Betriebsteam filtern.

Vorteile der Nutzung dieser bewährten Methode: Die proaktive Behebung von Alarmaktionen ermöglicht es dem Support-Personal, sich auf die Elemente zu konzentrieren, die nicht automatisch umsetzbar sind. Dies hilft dem Betriebspersonal, alle Alarme zu bewältigen, ohne überfordert zu werden, und sich stattdessen auf die kritischen Alarme zu konzentrieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Verwenden Sie Alarme, um automatisierte Aktionen auszulösen und auf diese Weise Probleme nach Möglichkeit zu beheben. Leiten Sie den Alarm an die Personen weiter, die die richtigen Maßnahmen einleiten können, falls keine automatisierte Reaktion möglich ist. Beispielsweise können Sie ein System nutzen, das erwartete Werte wichtiger Leistungskennzahlen (KPIs) prognostiziert und bei Überschreiten bestimmter Schwellenwerte einen Alarm ausgibt. Denkbar ist auch ein Tool, das Bereitstellungen automatisch anhält oder zurücksetzt, wenn sich KPIs außerhalb der erwarteten Werte befinden.

Implementieren Sie Prozesse, die Ihnen Einblick in die Leistung gewähren, während Ihr Workload ausgeführt wird. Entwickeln Sie Dashboards für die Überwachung und legen Sie Leistungsnormen in Form von Grundwerten fest, um zu bestimmen, ob die Workload optimal funktioniert.

Implementierungsschritte

- Mängelbeseitigungsworkflow identifizieren: Identifizieren und verstehen Sie das Leistungsproblem, das automatisch behoben werden kann. Verwenden Sie AWS-Überwachungslösungen wie Amazon CloudWatch oder AWS X-Ray, um die Ursache des Problems besser zu verstehen.
- Automatisierungsprozess definieren: Erstellen Sie einen schrittweisen Prozess zur Mängelbeseitigung, mit dem das Problem automatisch behoben werden kann.
- Initiationsereignis konfigurieren: Konfigurieren Sie das Ereignis so, dass der Prozess zur Mängelbeseitigung automatisch eingeleitet wird. Sie können beispielsweise einen Auslöser definieren, der eine Instance automatisch neu startet, wenn sie einen bestimmten Schwellenwert für die CPU-Auslastung erreicht.
- Mängelbeseitigung automatisieren: Verwenden Sie AWS-Services und Technologien, um den Mängelbeseitigungsprozess zu automatisieren. AWS Systems Manager Automation bietet beispielsweise eine sichere und skalierbare Möglichkeit, den Prozess zur Mängelbeseitigung zu automatisieren. Achten Sie darauf, die Selbstheilungslogik zu verwenden, um Änderungen rückgängig zu machen, wenn das Problem nicht gelöst wurde.
- Workflow testen: Testen Sie den automatisierten Prozess zur Mängelbeseitigung in einer Vorproduktionsumgebung.
- Workflow implementieren: Implementieren Sie die automatisierten Prozess zur Mängelbeseitigung in der Produktionsumgebung.
- Playbook entwickeln: Entwickeln und dokumentieren Sie ein Playbook, in dem die Schritte für den Mängelbeseitigungsplan beschrieben werden, einschließlich der Initiierungsereignisse, der Mängelbeseitigungslogik und der ergriffenen Maßnahmen. Stellen Sie sicher, dass alle Stakeholder entsprechend geschult werden, damit sie effektiv auf automatisierte Mängelbeseitigungsereignisse reagieren können.
- Überprüfen und verfeinern: Beurteilen Sie regelmäßig die Effektivität des automatisierten Mängelbeseitiungs-Workflows. Passen Sie bei Bedarf die Initiierungsereignisse und die Mängelbeseitigungslogik an.

Ressourcen

Zugehörige Dokumente:

- CloudWatch-Dokumentation
- Überwachung, Protokollierung und Leistung von AWS Partner Network-Partnern
- X-Ray-Dokumentation
- Verwendung von Alarmen und Alarmaktionen in CloudWatch
- Aufbau einer Cloud-Automatisierungspraxis für Operational Excellence: Bewährte Methoden von AWS Managed Services
- Automatisieren Ihrer Amazon Redshift-Leistungsoptimierung mit automatischer Tabellenoptimierung

Zugehörige Videos:

- AWS re:Invent 2023 – Strategien für die automatisierte Skalierung, Mängelbeseitigung und intelligente Selbstreparatur
- AWS re:Invent 2023 – [LAUNCH] Anwendungsüberwachung für moderne Workloads
- AWS re:Invent 2023 – Implementierung der Anwendungsbeobachtbarkeit
- AWS re:Invent 2021 – Intelligente Automatisierung des Cloud-Betriebs
- AWS re:Invent 2022 – Einrichtung skalierbarer Kontrollen in Ihrer AWS-Umgebung
- AWS re:Invent 2022 – Automatisierung der Patch-Verwaltung und -Compliance mit AWS
- AWS re:Invent 2022 – Wie Amazon bessere Metriken für eine höhere Website-Leistung verwendet
- AWS re:Invent 2023 – Ballast abwerfen: Diagnose und Lösung von Leistungsproblemen mit Amazon RDS
- AWS re:Invent 2021 – {Neuer Launch} Automatische Erkennung und Behebung von Problemen mit Amazon DevOps Guru
- AWS re:Invent 2023 – Zentralisierung Ihrer Abläufe

Zugehörige Beispiele:

- CloudWatch Logs Konfigurieren von Alarmen

PERF05-BP06 Konstantes Aktualisieren des Workloads und der Services

Erhalten Sie aktuelle Informationen zu neuen Cloud-Services und -Funktionen, um effiziente Funktionen zu übernehmen, Probleme zu beseitigen und die allgemeine Leistungseffizienz des Workloads zu verbessern.

Typische Anti-Muster:

- Sie gehen davon aus, dass Ihre aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert wird.
- Sie haben keine Systeme oder regelmäßigen Besprechungen zur Prüfung, ob aktualisierte Software und Pakete mit Ihrem Workload kompatibel sind.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie einen Prozess einrichten, um aktuelle Informationen zu neuen Services und Angeboten zu erhalten, können Sie neue Funktionen und Kapazitäten nutzen, Probleme lösen und die Workload-Leistung verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Evaluieren Sie Möglichkeiten zur Verbesserung der Leistung, wenn neue Services, Entwurfsmuster und Produktfunktionen verfügbar sind. Ermitteln Sie anhand von Bewertungen, internen Diskussionen oder externen Analysen, wie sich diese neuen Optionen positiv auf die Leistung oder Effizienz der Workload auswirken können. Definieren Sie einen Prozess zum Bewerten von Updates, neuen Funktionen und Services, die für Ihren Workload relevant sind. Erstellen Sie beispielsweise Machbarkeitsstudien, die auf neuen Technologien aufbauen, oder beraten Sie sich mit einer internen Gruppe. Führen Sie beim Ausprobieren neuer Ideen oder Services Leistungstests durch, um die Auswirkungen auf die Leistung des Workloads zu messen.

Implementierungsschritte

- Inventarisierung Ihrer Workload: Inventarisieren Sie Ihre Workload-Software und -Architektur und identifizieren Sie Komponenten, die aktualisiert werden müssen.
- Identifizierung Ihrer Aktualisierungsquellen: Identifizieren Sie Nachrichten und Aktualisierungsquellen im Zusammenhang mit Ihren Workload-Komponenten. Sie können beispielsweise den Neuigkeiten im AWS-Blog für die Produkte abonnieren, die zu Ihrer Workload-Komponente passen. Sie können den RSS-Feed abonnieren oder Ihre E-Mail-Abonnements verwalten.
- Definition eines Aktualisierungszeitplans: Definieren Sie einen Zeitplan, um neue Services und Features für Ihr Workload zu bewerten.
  - Sie können AWS Systems Manager Inventory verwenden, um Betriebssystem (BS)-, Anwendungs- und Instance-Metadaten von Ihren Amazon EC2-Instances zu sammeln und so

schnell zu erfassen, welche Instances die Software und die Konfigurationen ausführen, die Ihre Softwarerichtlinie erfordert, und welche Instances aktualisiert werden müssen.

- Bewertung der neuen Aktualisierung: Erfahren Sie, wie die Komponenten Ihres Workloads aktualisiert werden. Nutzen Sie die Agilität in der Cloud, um schnell zu testen, wie neue Features Ihr Workload verbessern und so die Leistungseffizienz steigern können.
- Verwendung von Automatisierung: Verwenden Sie Automatisierung für den Aktualisierungsvorgang, um den Aufwand für die Bereitstellung neuer Features zu reduzieren und Fehler zu begrenzen, die durch manuelle Prozesse verursacht werden.
  - Sie können CI/CD verwenden, um AMIs, Container-Images und andere Artefakte im Zusammenhang mit Ihrer Cloud-Anwendung automatisch zu aktualisieren.
  - Sie können Tools wie den AWS Systems Manager Patch Manager verwenden, um den Systemaktualisierungsprozess zu automatisieren und die Aktivitäten mit AWS Systems Manager Maintenance Windows zu planen.
- Dokumentation des Prozesses: Dokumentieren Sie Ihren Prozess zur Bewertung von Updates und neuen Services. Geben Sie Ihren Eigentümern ausreichend Zeit und Raum zum Forschen, Testen, Experimentieren und zur Validierung von Aktualisierungen und neuen Services. Nutzen Sie die dokumentierten geschäftlichen Anforderungen und KPIs, um zu ermitteln, welche Aktualisierungen positive geschäftliche Auswirkungen haben werden.

Ressourcen

Zugehörige Dokumente:

- AWS-Blog
- Neuerungen bei AWS
- Implementierung aktueller Images mit automatisierten EC2 Image Builder Pipelines

Zugehörige Videos:

- AWS re:Inforce 2022 – Automatisierung der Patch-Verwaltung und -Compliance mit AWS
- All Things Patch: AWS Systems Manager | AWS-Veranstaltungen

Zugehörige Beispiele:

- Bestands- und Patch-Verwaltung

- Workshop zur Beobachtbarkeit

PERF05-BP07 Regelmäßiges Überprüfen von Metriken

Überprüfen Sie im Rahmen der routinemäßigen Wartungsmaßnahme oder als Reaktion auf Ereignisse oder Vorfälle, welche Metriken erfasst werden. Ermitteln Sie anhand dieser Überprüfung, welche Metriken für die Behebung von Problemen wesentlich waren und welche zusätzlichen Kennzahlen, sofern nachverfolgt, helfen könnten, Probleme zu identifizieren, zu beheben oder zu verhindern.

Typische Anti-Muster:

- Sie lassen zu, dass Metriken für einen längeren Zeitraum im Alarmstatus bleiben.
- Sie erstellen Alarme, die von einem Automatisierungssystem nicht umsetzbar sind.

Vorteile der Nutzung dieser bewährten Methode: Überprüfen Sie kontinuierlich Metriken, die erfasst werden, um sicherzustellen, dass sie Probleme ordnungsgemäß identifizieren, beheben oder verhindern. Metriken können auch veralten, wenn sie für einen längeren Zeitraum im Alarmstatus bleiben.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Verbessern Sie kontinuierlich die Erfassung und Überwachung von Metriken. Bewerten Sie beim Reagieren auf Vorfälle oder Ereignisse diejenigen Kennzahlen, die hilfreich für die Behebung des Problems waren, und überlegen Sie, welche derzeit noch nicht verfolgten Kennzahlen förderlich sein könnten. Verbessern Sie auf diese Weise die Qualität der erfassten Metriken, damit Sie zukünftige Probleme verhindern oder schneller beheben können.

Bewerten Sie beim Reagieren auf Vorfälle oder Ereignisse diejenigen Kennzahlen, die hilfreich für die Behebung des Problems waren, und überlegen Sie, welche derzeit noch nicht verfolgten Kennzahlen förderlich sein könnten. Verbessern Sie auf diese Weise die Qualität der erfassten Metriken, damit Sie zukünftige Probleme verhindern oder schneller beheben können.

Implementierungsschritte

- Metriken definieren: Definieren Sie wichtige Leistungsmetriken zur Überwachung, die auf Ihr Workload-Ziel abgestimmt sind, einschließlich Metriken wie Reaktionszeit und Ressourcenauslastung.

- Ausgangswert festlegen: Legen Sie für jede Metrik einen Ausgangswert und einen Zielwert fest. Der Ausgangswert sollte Referenzpunkte zur Identifizierung von Abweichungen oder Anomalien enthalten.
- Takt festlegen: Legen Sie einen Takt zur Überprüfung wichtiger Metriken fest (z. B. wöchentlich oder monatlich).
- Leistungsprobleme identifizieren: Beurteilen Sie bei jeder Überprüfung Trends und Abweichungen von den Ausgangswerten. Suchen Sie nach Leistungsengpässen oder Anomalien. Führen Sie bei identifizierten Problemen eine eingehende Ursachenanalyse durch, um den Hauptgrund für das Problem zu ermitteln.
- Korrekturmaßnahmen identifizieren: Identifizieren Sie Korrekturmaßnahmen mithilfe Ihrer Analysen. Dies kann die Parameteroptimierung, das Beheben von Fehlern und das Skalieren von Ressourcen beinhalten.
- Ergebnisse dokumentieren: Dokumentieren Sie Ihre Erkenntnisse, einschließlich identifizierter Probleme, Ursachen und Korrekturmaßnahmen.
- Iterieren und verbessern: Beurteilen und verbessern Sie kontinuierlich den Prozess zur Überprüfung der Metriken. Nutzen Sie die Erkenntnisse aus der vorherigen Überprüfung, um den Prozess im Laufe der Zeit zu verbessern.

Ressourcen

Zugehörige Dokumente:

- CloudWatch-Dokumentation
- Erfassen von Metriken und Protokollen aus Amazon EC2-Instances und On-Premises-Servern mit dem CloudWatch Agent
- Metrikabfrage mit CloudWatch Metrics Insights
- Überwachung, Protokollierung und Leistung von AWS Partner Network-Partnern
- X-Ray-Dokumentation

Zugehörige Videos:

- AWS re:Invent 2022 – Einrichtung skalierbarer Kontrollen in Ihrer AWS-Umgebung
- AWS re:Invent 2022 – Wie Amazon bessere Metriken für eine höhere Website-Leistung verwendet
- AWS re:Invent 2023 – Aufbau einer effektiven Beobachtbarkeitsstrategie

- AWS Summit SF 2022 – Full-Stack-Beobachtbarkeit und -Überwachung von Anwendungen mit AWS
- AWS re:Invent 2023 – Ballast abwerfen: Diagnose und Lösung von Leistungsproblemen mit Amazon RDS

Zugehörige Beispiele:

- Erstellen eines Dashboards mit QuickSight
- CloudWatch-Dashboards

Kostenoptimierung

Die Säule Kostenoptimierung umfasst die Fähigkeit, Systeme so auszuführen, dass sie geschäftlichen Wert bei geringstmöglichen Kosten liefern. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule der Kostenoptimierung.

Bereiche für bewährte Methoden

- Praxis für Cloud-Finanzmanagement
- Ausgabenerkennung und Nutzungsbewusstsein
- Kostengünstige Ressourcen
- Verwaltung von Nachfrage und Bereitstellung von Ressourcen
- Optimierung im Laufe der Zeit

Praxis für Cloud-Finanzmanagement

Frage

- KOSTEN 1. Wie implementieren Sie das Cloud Financial Management?

KOSTEN 1. Wie implementieren Sie das Cloud Financial Management?

Die Implementierung eines Cloud Financial Managements hilft Organisationen, geschäftliche Mehrwerte und einen wirtschaftlichen Erfolg zu erzielen, während sie die Kosten und die Nutzung optimieren und auf AWS skalieren.

Bewährte Methoden

- COST01-BP01 Definieren der Zuständigkeit für die Kostenoptimierung
- COST01-BP02 Einrichten einer Partnerschaft zwischen Finanzen und Technologie
- COST01-BP03 Erstellen von Cloud-Budgets und -Prognosen
- COST01-BP04 Implementieren von Kostenbewusstsein in Ihre Organisationsprozesse
- COST01-BP05 Berichte und Benachrichtigungen zur Kostenoptimierung
- COST01-BP06 Proaktive Überwachung der Kosten
- COST01-BP07 Verfolgen neuer Serviceversionen
- COST01-BP08 Schaffen einer kostenbewussten Kultur
- COST01-BP09 Quantifizieren des Geschäftswerts von Kostenoptimierungen

COST01-BP01 Definieren der Zuständigkeit für die Kostenoptimierung

Stellen Sie ein Team zusammen (Cloud Business Office, Cloud Center of Excellence oder FinOps-Team), das für die Entwicklung und Aufrechterhaltung des Kostenbewusstseins in Ihrer gesamten Organisation verantwortlich ist. Für die Kostenoptimierung kann eine Einzelperson oder ein Team zuständig sein (mit Mitarbeitern aus dem Finanz-, Technologie- und Geschäftsbereich), Voraussetzung ist eine Übersicht über die gesamte Organisation und die Finanzierung der Cloud.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Dies ist die Einführung einer Funktion oder eines Teams für Cloud Business Office (CBO) oder ein Cloud-Kompetenzzentrum (CCoE), das für die Entwicklung und Wahrung einer Kultur des Kostenbewusstseins im Bereich Cloud-Computing verantwortlich ist. Bei dieser Funktion kann es sich um eine bereits im Unternehmen hierfür zuständige Person, ein Team innerhalb Ihrer Organisation oder um ein neues Team handeln, das sich aus den wichtigsten Finanz-, Technologie- und Organisationsbeteiligten aus der gesamten Organisation zusammensetzt.

Die Funktion (Einzelperson oder Team) priorisiert und verbraucht den erforderlichen Prozentsatz ihrer Zeit für Kostenmanagement- und Kostenoptimierungsaktivitäten. Bei kleinen Unternehmen kann die Funktion einen geringeren Prozentsatz der Zeit im Vergleich zu einer Vollzeitfunktion für ein größeres Unternehmen aufwenden.

Diese Funktion (Einzelperson oder Team) priorisiert und nutzt den erforderlichen Prozentsatz ihrer Arbeitszeit für Kostenmanagement- und Kostenoptimierungsaktivitäten. In einer kleinen Organisation

benötigt die Funktion möglicherweise einen geringeren Zeitanteil für Kostenmanagement- und Optimierungsaktivitäten als in einer Vollzeitfunktion in einem größeren Unternehmen.

Die Funktion erfordert einen multidisziplinären Ansatz, der Kompetenzen in den Bereichen Projektmanagement, Datenwissenschaft, Finanzanalyse und Software- oder Infrastrukturentwicklung voraussetzt. Die Mitarbeiter können die Effizienz von Workloads durch Kostenoptimierungen auf drei unterschiedlichen Verantwortlichkeitsebenen verbessern:

- Zentralisiert: Mit designierten Teams, beispielsweise FinOps, Cloud Financial Management (CFM), Cloud Business Office (CBO) oder einem Cloud-Kompetenzzentrum (CCoE), können Kunden Governance-Mechanismen entwerfen und implementieren sowie unternehmensweit bewährte Methoden fördern.
- Dezentralisiert: Hierbei werden Technologieteams mit der Durchführung von Kostenoptimierungen beauftragt.
- Hybrid: Zentralisierte und dezentralisierte Teams arbeiten gemeinsam an der Umsetzung von Kostenoptimierungen.

Die Funktion kann anhand ihrer Fähigkeit zur Durchführung und Implementierung im Hinblick auf Kostenoptimierungsziele gemessen werden (z. B. durch Workload-Effizienzmetriken).

Sie müssen sicherstellen, dass Führungskräfte diese Funktion als Sponsoren/Förderer unterstützen. Dies ist ein entscheidender Erfolgsfaktor. Der entsprechende Sponsor befürwortet eine kosteneffiziente Cloud-Nutzung und bietet Eskalationsunterstützung für das Team, um sicherzustellen, dass die Aktivitäten zur Kostenoptimierung mit der vom Unternehmen definierten Priorität behandelt werden. Andernfalls können Anweisungen nicht beachtet und Möglichkeiten für Kosteneinsparungen nicht priorisiert werden. Gemeinsam helfen der Sponsor und das Team Ihrer Organisation dabei, die Cloud effizient zu nutzen und Werte für das Unternehmen zu schaffen.

Wenn Sie über den Business, Enterprise-On-Ramp- oder Enterprise- Supportplan verfügen und Hilfe beim Aufbau dieses Teams oder dieser Funktion benötigen, wenden Sie sich über Ihr Account-Team an Ihre Experten für Cloud Financial Management (CFM).

Implementierungsschritte

- Definieren wichtiger Mitglieder: Alle relevanten Bereiche Ihres Unternehmens müssen ihren Beitrag leisten und ein Interesse an der Kostenverwaltung haben. Häufig handelt es sich hierbei um Teams mit Verantwortung für Finanzen, Anwendungen oder Produkte, das Management und technische Teams (DevOps). Einige Teams setzen ihre ganze Arbeitszeit hierfür ein (Finanz-.

und Technikbereich), während andere nach Bedarf eingebunden werden. Die mit CFM befassten Personen oder Teams benötigen Kompetenzen in den folgenden Bereichen:

- Softwareentwicklung: für den Fall, dass Skripte und Automatisierungen erstellt werden.
- Infrastrukturtechnik: um Skripts bereitzustellen, Prozesse zu automatisieren und zu verstehen, wie Services oder Ressourcen bereitgestellt werden.
- Operatives Wissen: CFM stellt durch Messung, Überwachung, Änderung, Planung und Skalierung eine effiziente Nutzung der Cloud sicher.

- Definieren von Zielen und Metriken: Die Funktion muss der Organisation auf verschiedene Weise Mehrwert bieten. Diese Ziele werden definiert und mit der Entwicklung der Organisation kontinuierlich weiterentwickelt. Häufige Aktivitäten sind das Erstellen und Durchführen von Schulungsprogrammen zur Kostenoptimierung in der gesamten Organisation, Entwickeln von unternehmensweiten Standards wie Überwachung und Berichterstellung zur Kostenoptimierung sowie Festlegen der Workload-Ziele bei der Optimierung. Außerdem muss diese Funktion der Organisation regelmäßig über ihre Möglichkeiten zur Kostenoptimierung Bericht erstatten.

  Sie können wert- oder kostenbasierte Leistungsindikatoren (Key Performance Indicators, KPIs) definieren. Wenn Sie KPIs definieren, können Sie die erwarteten Kosten in Bezug auf Effizienz und erwartete geschäftliche Ergebnisse berechnen. Wertbasierte KPIs verbinden Kosten- und Nutzungsmetriken mit Geschäftswertfaktoren und helfen, Änderungen bei AWS-Ausgaben zu begründen. Der erste Schritt bei der Formulierung wertbasierter KPIs besteht in der organisationsweiten Zusammenarbeit, um einen Standardsatz von KPIs auszuwählen und zu vereinbaren.

- Festlegen einer regulären Kadenz: Die Gruppe (Teams aus den Bereichen Finanzen, Technologie und Geschäft) sollte sich regelmäßig treffen, um Ziele und Metriken zu überprüfen. Dazu gehört in der Regel die Überprüfung des Status der Organisation, der aktuell ausgeführten Programme und der gesamten Finanz- und Optimierungsmetriken. Anschließend werden detaillierte Berichte zu wichtigen Workloads erstellt.

  Bei diesen regelmäßigen Überprüfungen können Sie die Workload-Effizienz (Kosten) und die geschäftlichen Ergebnisse bewerten. Eine Kostensteigerung von 20 % für einen Workload könnte beispielsweise mit einer erhöhten Nutzung durch Kunden zusammenhängen. In einem solchen Fall kann die Kostensteigerung von 20 % als Investition betrachtet werden. Solche regelmäßigen Besprechungen können Teams helfen, wertbasierte KPIs zu identifizieren, die für die gesamte Organisation sinnvoll sind.

Ressourcen

Zugehörige Dokumente:

- AWS CCOE-Blog
- Einrichtung von Cloud Business Office
- CCOE – Cloud Center of Excellence

Zugehörige Videos:

- Vanguard CCOE, eine Erfolgsgeschichte

Zugehörige Beispiele:

- Nutzung eines Cloud-Kompetenzzentrums (Center of Excellence, CCOE) zur Transformation des gesamten Unternehmens
- Einrichtung eines CCOE zur Transformation des gesamten Unternehmens
- 7 Fehler, die Sie bei der Einrichtung eines CCOE vermeiden sollten

COST01-BP02 Einrichten einer Partnerschaft zwischen Finanzen und Technologie

Beziehen Sie Finanz- und Technologieteams in Kosten- und Nutzungsgespräche in allen Phasen Ihrer Cloud-Reise mit ein. Teams treffen sich regelmäßig, um Themen wie Unternehmensziele, aktuellen Kosten- und Nutzungsstatus sowie Finanz- und Buchhaltungsmethoden zu besprechen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Technologieteams können in der Cloud dank verkürzter Genehmigungs-, Beschaffungs- und Infrastrukturbereitstellungszyklen schneller Innovationen vorantreiben. Dies kann eine Anpassung für Finanzunternehmen sein, die zuvor an die Ausführung zeitaufwändiger und ressourcenintensiver Prozesse zur Beschaffung und Bereitstellung von Kapital in Rechenzentrums- und lokalen Umgebungen und die Kostenzuordnung nur nach Projektgenehmigung gewöhnt waren.

Was die Finanz- und Beschaffungsabteilungen betrifft, wurden die Prozesse in den Bereichen Budgetierung, Kapitalbedarf, Genehmigung, Beschaffung und Installation der physischen Infrastruktur über Jahrzehnte hinweg weiterentwickelt und standardisiert.

- In der Regel fordern die Entwicklungs- oder IT-Teams die Geldmittel an.
- Die Finanzteams genehmigen und beschaffen die Geldmittel.
- Die operativen Teams stellen die Infrastruktur zusammen, sodass sie direkt eingesetzt werden kann.

Mit der Einführung der Cloud werden Beschaffung und Nutzung der Infrastruktur nicht mehr als Kette von Abhängigkeiten betrachtet. Im Cloud-Modell entwickeln Technologie- und Produktteams ihre Produkte nicht nur, sondern führen sie auch selbst aus und sind für sie verantwortlich. Dabei führen sie die meisten Aktivitäten aus, die bisher als Domäne der Finanz- und operativen Teams betrachtet wurden, einschließlich Beschaffung und Bereitstellung.

Zur Bereitstellung von Cloud-Ressourcen werden lediglich ein Benutzerkonto und der richtige Satz von Berechtigungen benötigt. Dies reduziert auch die Risiken in den Bereichen IT und Finanzen, da die Teams stets nur einige Klicks oder API-Aufrufe von der Einstellung nicht genutzter oder nicht notwendiger Cloud-Ressourcen entfernt sind. Technologieteams können so auch schneller Innovationen einführen und erhalten die nötige Agilität und Flexibilität, um Experimente zu starten und zu beenden. Auch wenn sich die variable Natur der Cloud-Nutzung auf die Planbarkeit der Budgetierung und die Genauigkeit von Prognosen auswirken kann, bietet sie Organisationen jedoch auch die Möglichkeit, sowohl die Kosten für Überbereitstellungen als auch die Opportunitätskosten für konservative Unterbereitstellungen zu reduzieren.

Bauen Sie eine Partnerschaft zwischen wichtigen Beteiligten aus dem Finanzwesen und der Technologie auf, um ein gemeinsames Verständnis der organisatorischen Ziele zu schaffen und Mechanismen zu entwickeln, um im variablen Ausgabenmodell von Cloud Computing einen finanziellen Erfolg zu erzielen. Relevante Teams innerhalb Ihres Unternehmens müssen an Kosten- und Nutzungsdiskussionen in allen Phasen Ihrer Cloud-Reise beteiligt sein, einschließlich:

- Verantwortliche im Finanzbereich: CFOs, Finanzkontrolleure, Finanzplaner, Geschäftsanalysten, Beschaffung und Kreditorenbuchhaltung müssen das Cloud-Modell des Verbrauchs, Kaufoptionen und den monatlichen Rechnungsprozess verstehen. Die Teams in den Bereichen Finanzen und Technologie müssen zusammenarbeiten, um die IT-Wertschöpfung zu entwickeln und darzustellen, damit die geschäftlichen Teams die Verbindung zwischen Technologieausgaben und Geschäftsergebnissen verstehen können. Auf diese Weise werden Technologieaufwendungen nicht als Kosten angesehen, sondern als Investitionen. Aufgrund der grundlegenden Unterschiede zwischen der Cloud (z. B. Änderungsrate der Nutzung, Pay-as-you-go-Preisgestaltung, gestaffelte Preise, Preismodelle und detaillierte Abrechnungs- und Nutzungsinformationen) im Vergleich zum Betrieb vor Ort ist es für die Finanzorganisation von entscheidender Bedeutung, dass sie versteht, wie sich die Nutzung der Cloud auf geschäftliche Aspekte wie Beschaffungsprozesse, Anreizverfolgung, Kostenzuordnung und Finanzberichte auswirken kann.
- Verantwortliche im Technologiebereich: Technologieverantwortliche (einschließlich Produkt- und Anwendungsbesitzer) müssen die finanziellen Anforderungen (z. B. Budgeteinschränkungen) sowie die geschäftlichen Anforderungen (z. B. Service Level Agreements) kennen. Damit kann das System implementiert werden, um die gewünschten Ziele des Unternehmens zu erreichen.

Die Partnerschaft zwischen Finanzen und Technologie bietet folgende Vorteile:

- Finanz- und Technologieteams haben nahezu in Echtzeit Einblicke in Kosten und Nutzung.
- Finanz- und Technologieteams legen ein standardmäßiges Betriebsverfahren für die Bewältigung von Ausgabeunterschieden in der Cloud fest.
- Stakeholder im Bereich Finanzen handeln als strategische Berater bei der Nutzung von Kapital für den Kauf rabattierter Programme (z. B. Reserved Instances oder AWS Savings Plans) und der Nutzung der Cloud zur Förderung des Wachstums der Organisation.
- Vorhandene Kreditorenbuchhaltungs- und Beschaffungsprozesse werden mit der Cloud verwendet.
- Die Finanz- und Technologieteams prognostizieren gemeinsam die Kosten und die Nutzung von AWS in der Zukunft, um die Budgets der Organisation entsprechend auszurichten und zu entwickeln.
- Bessere unternehmensübergreifende Kommunikation durch eine gemeinsame Sprache und ein gemeinsames Verständnis von Finanzkonzepten.

Weitere Beteiligte innerhalb Ihres Unternehmens, die an Kosten- und Nutzungsdiskussionen beteiligt sein sollten, sind:

- Besitzer von Geschäftseinheiten: Besitzer von Geschäftseinheiten müssen sich mit dem Cloud-Geschäftsmodell vertraut machen, sodass sie den Geschäftseinheiten und dem gesamten Unternehmen die Richtung weisen können. Dieses Cloud-Wissen ist wichtig, wenn es erforderlich ist, das Wachstum und die Systemnutzung zu prognostizieren oder verschiedene Kaufoptionen zu bewerten, z. B. Reserved Instances oder Savings Plans.
- Entwicklungsteam: Eine Partnerschaft zwischen Finanz- und Technologieteams hat kritische Bedeutung für die Entwicklung einer kostenbewussten Kultur, die Entwickler motiviert, im Bereich Cloud Financial Management (CFM) aktiv zu werden. Ein häufiges Problem von CFM- und Finanzteams besteht darin, Entwicklern ein Verständnis des Geschäfts in der Cloud zu vermitteln und sie zur Umsetzung von Best Practices und empfohlenen Aktionen zu motivieren.
- Dritte: Wenn Ihr Unternehmen mit Dritten arbeitet (z. B. Berater oder Tools), dann stellen Sie sicher, dass diese an Ihren finanziellen Zielen ausgerichtet sind und sowohl die Ausrichtung durch ihre Engagement-Modelle als auch einen ROI (Return on Investment) nachweisen können. In der Regel beteiligen sich Dritte an der Berichterstellung und Analyse der von ihnen verwalteten Systeme, und sie stellen Kostenanalysen für die von ihnen konzipierten Workloads bereit.

Eine erfolgreiche CFM-Implementierung erfordert die Zusammenarbeit von Teams in den Bereichen Finanzen, Technologie und Geschäft sowie eine veränderte Kommunikation und Evaluierung in Bezug auf die Cloud-Ausgaben der Organisation. Beziehen Sie die Entwicklungsteams in alle Phasen der Diskussion über Kosten- und Nutzung ein und motivieren Sie sie zur Befolgung von Best Practices und zur Umsetzung vereinbarter Aktionen.

Implementierungsschritte

- Definieren wichtiger Mitglieder: Stellen Sie sicher, dass sich alle relevanten Mitglieder Ihrer Finanz- und Technologieteams aktiv an der Partnerschaft beteiligen. Relevante Mitglieder im Bereich Finanzen sind Personen, die mit Cloud-Ausgaben interagieren. Dies sind in der Regel CFOs, Finanzcontroller, Finanzplaner, Geschäftsanalysten und Mitarbeiter in Beschaffung und Einkauf. Technologiemitarbeiter sind in der Regel Produkt- und Anwendungsbesitzer, technische Manager und Vertreter aller Teams, die in der Cloud aktiv sind. Weitere Mitglieder können Geschäftsbereiche mit Einfluss auf die Nutzung von Produkten sein, zum Beispiel das Marketing, und Dritte wie Berater, die Sie bei der Ausrichtung an Ihren Zielen und Mechanismen und bei Berichten unterstützen.
- Definieren von Diskussionsthemen: Definieren Sie die Themen, die in den Teams häufig auftreten, oder ein gemeinsames Verständnis erfordern. Verfolgen Sie die Kosten ab dem Zeitpunkt, an dem sie generiert werden, bis zur Bezahlung der Rechnung. Beachten Sie alle beteiligten Mitglieder

und organisatorischen Prozesse, die angewendet werden müssen. Informieren Sie sich über jeden einzelnen Schritt oder Prozess, den sie durchlaufen, sowie die zugehörigen Informationen, wie z. B. verfügbare Preismodelle, gestaffelte Preise, Rabattmodelle, Budgetplanung und finanzielle Anforderungen.

- Festlegen einer regulären Kadenz: Richten Sie eine regelmäßige Kommunikationskadenz ein, um Finanz- und Technologieteams aneinander auszurichten und eine Partnerschaft zu unterstützen. Die Gruppe muss regelmäßig im Hinblick auf ihre Ziele und Metriken zusammenkommen. Dazu gehört in der Regel die Überprüfung des Status der Organisation, der aktuell ausgeführten Programme und der gesamten Finanz- und Optimierungsmetriken. Anschließend werden detaillierte Berichte zu wichtigen Workloads erstellt.

Ressourcen

Zugehörige Dokumente:

- AWS News-Blog

COST01-BP03 Erstellen von Cloud-Budgets und -Prognosen

Passen Sie vorhandene Budgetierungs- und Prognoseprozesse so an, dass sie mit der stark variablen Natur der Cloud-Kosten und -Nutzung kompatibel sind. Prozesse müssen dynamisch sein und Algorithmen anwenden, die auf Trends oder Geschäftsfaktoren oder einer Kombination aus beiden basieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Bei herkömmlichen On-Premises-IT-Setups stehen Kunden oft vor der Herausforderung, Fixkosten zu planen, die sich nur gelegentlich ändern, typischerweise beim Kauf neuer IT-Geräte und -Services, um die Spitzennachfrage zu decken. Im Gegensatz dazu verfolgt AWS Cloud einen anderen Ansatz, bei dem Kunden nur für die Ressourcen bezahlen, die sie nutzen, und zwar entsprechend ihren tatsächlichen IT- und Geschäftsanforderungen. In der Cloud-Umgebung kann die Nachfrage monatlich, täglich oder sogar stündlich schwanken.

Die Nutzung der Cloud bringt Effizienz, Geschwindigkeit und Agilität, damit allerdings auch ein stark variables Kosten- und Nutzungsmuster. Die Kosten können als Reaktion auf eine höhere Workload-Effizienz oder die Bereitstellung neuer Workloads und Features sinken oder manchmal eben auch

steigen. Wenn Workloads skaliert werden, um einen wachsenden Kundenstamm zu bedienen, steigen parallel dazu die Cloud-Nutzung und -Kosten aufgrund der besseren Verfügbarkeit von Ressourcen. Diese Flexibilität bei Cloud-Services erstreckt sich auch auf die Kosten und Prognosen, was zu einer gewissen Elastizität führt.

Es ist wichtig, sich eng an diesen sich ändernden Geschäftsanforderungen und Nachfragetreibern auszurichten und eine möglichst genaue Planung anzustreben. Traditionelle Budgetprozesse in Organisationen müssen angepasst werden, um dieser Variabilität Rechnung zu tragen.

Ziehen Sie bei der Prognose der Kosten für neue Workloads eine Kostenmodellierung in Betracht. Durch die Kostenmodellierung erhalten Sie ein grundlegendes Verständnis der erwarteten Cloud-Kosten, das Ihnen hilft, Gesamtbetriebskosten (TCO), Kapitalrendite (ROI) und andere Finanzanalysen durchzuführen, Ziele und Erwartungen mit Stakeholdern festzulegen und Möglichkeiten zur Kostenoptimierung zu identifizieren.

Ihre Organisation muss die Kostendefinitionen und akzeptierten Gruppierungen kennen. Der Detaillierungsgrad, mit dem Sie Prognosen erstellen, kann je nach Struktur und internen Workflows Ihrer Organisation variieren. Wählen Sie eine Granularität, die Ihren spezifischen Anforderungen und Ihrer Organisationsstruktur entspricht. Es ist wichtig zu verstehen, auf welcher Ebene die Prognose durchgeführt wird:

- Verwaltungskonto oder AWS Organizations-Ebene: Das Verwaltungskonto ist das Konto, das Sie zum Erstellen von AWS Organizations verwenden. Organizations haben standardmäßig ein Verwaltungskonto.
- Verbundenes oder Mitgliedskonto: Ein Konto in Organizations ist ein Standard-AWS-Konto, das Ihre AWS-Ressourcen und die Identitäten enthält, die auf diese Ressourcen zugreifen können.
- Umgebung: Eine Umgebung ist eine Sammlung von AWS-Ressourcen, auf denen eine Anwendungsversion ausgeführt wird. Eine Umgebung kann mit mehreren verknüpften oder Mitgliedskonten erstellt werden.
- Projekt: Ein Projekt ist eine Kombination aus festgelegten Zielen oder Aufgaben, die innerhalb eines bestimmten Zeitraums zu erfüllen sind. Es ist wichtig, den Projektlebenszyklus bei Ihrer Prognose zu berücksichtigen.
- AWS-Services: Gruppen oder Kategorien wie Computing- oder Speicherservices, in denen Sie AWS-Services für Ihre Prognose gruppieren können.
- Benutzerdefinierte Gruppierung: Sie können benutzerdefinierte Gruppen erstellen, die auf den Anforderungen Ihrer Organisation basieren, z. B. Geschäftseinheiten, Kostenstellen, Teams, Kostenzuordnungs-Tags, Kostenkategorien, verknüpfte Konten oder eine Kombination davon.

Identifizieren Sie die Geschäftsfaktoren, die sich auf Ihre Nutzungskosten auswirken können, und erstellen Sie für jeden dieser Faktoren separate Prognosen, um die erwartete Nutzung im Voraus zu berechnen. Einige der Faktoren fallen in den Verantwortungsbereich von IT- und Produktteams innerhalb der Organisation. Andere Geschäftsfaktoren, wie Marketingveranstaltungen, Werbeaktionen, geografische Expansionen, Fusionen und Übernahmen, sind den Führungskräften in Vertrieb und Marketing und der Geschäftsleitung bekannt. Es ist wichtig, zusammenzuarbeiten und auch all diese Nachfragetreiber zu berücksichtigen.

Mit AWS Cost Explorer können Sie Kosten für einen definierten zukünftigen Zeitraum basierend auf Trends und Ihren bisherigen Ausgaben prognostizieren. Die Prognose-Engine von AWS Cost Explorer segmentiert Ihre historischen Daten auf Grundlage von Gebührentypen (z. B. Reserved Instances) und verwendet eine Kombination aus Machine Learning und regelbasierten Modellen, um die Ausgaben für alle Gebührentypen individuell zu prognostizieren.

Sobald Sie Ihren Prognoseprozess eingerichtet und Modelle erstellt haben, können Sie mit AWS Budgets angepasste, detaillierte Budgets festlegen, indem Sie den Zeitraum, die Wiederholungen oder den Betrag (fest oder variabel) angeben und Filter wie Service, AWS-Region und Tags hinzufügen. Das Budget wird in der Regel für ein Jahr geplant und bleibt unverändert, sodass alle Stakeholder sich strikt daran halten müssen. Im Gegensatz dazu sind Prognosen flexibler, da sie erneute Anpassungen im Laufe des Jahres ermöglichen und dynamische Prognosen über einen Zeitraum von einem, zwei oder drei Jahren liefern. Sowohl die Budgetierung als auch Prognosen spielen eine entscheidende Rolle bei der Definition der Finanzerwartungen verschiedener Stakeholder aus dem technischen und geschäftlichen Bereich. Genaue Prognosen und deren Umsetzung sorgen zudem dafür, dass die Stakeholder, die direkt für die Bereitstellungskosten verantwortlich sind, zur Rechenschaft gezogen werden. Außerdem wird auf diese Weise das allgemeine Kostenbewusstsein gestärkt.

Um über die Leistung Ihrer bestehenden Budgets auf dem Laufenden zu bleiben, können Sie AWS Budgets-Berichte erstellen und planen, die Sie und Ihre Stakeholder in regelmäßigen Abständen per E-Mail erhalten. Sie können auch AWS Budgets-Warnmeldungen basierend auf tatsächlichen Kosten erstellen, also einen reaktiven Prozess. Budgetwarnungen zu prognostizierten Kosten geben Ihnen Zeit, Abhilfemaßnahmen gegen potenzielle Kostenüberschreitungen zu implementieren. Sie können sich benachrichtigen lassen, wenn Ihre Kosten oder Ihre Nutzung ein bestimmtes Niveau übersteigen oder in der Zukunft den budgetierten Betrag möglicherweise überschreiten werden.

Gestalten Sie vorhandene Budget- und Prognoseprozesse dynamischer. Hierzu können Sie trendbasierte Algorithmen (mit historischen Kosten als Eingabe) und auf Geschäftsfaktoren basierende Algorithmen verwenden (z. B. auf der Einführung neuer Produkte, auf einer regionalen

Expansion oder neuen Umgebungen für Workloads), die besonders für Umgebungen mit dynamischen und variablen Ausgaben geeignet sind. Sobald Sie Ihre trendbasierte Prognose mithilfe von Cost Explorer oder anderen Tools ermittelt haben, können Sie mit AWS -Preisrechner Ihren AWS-Anwendungsfall und die zukünftigen Kosten auf Grundlage der erwarteten Nutzung abschätzen (Datenverkehr, Anfragen pro Sekunde oder erforderliche Amazon EC2-Instances).

Überprüfen Sie die Genauigkeit dieser Prognose, da Budgets auf Grundlage dieser Prognoseberechnungen und -schätzungen festgelegt werden sollten. Überwachen Sie die Genauigkeit und Effektivität der integrierten Cloud-Kostenprognosen. Überprüfen Sie regelmäßig die tatsächlichen Ausgaben im Vergleich zur Prognose und passen Sie sie bei Bedarf an, um die Prognosepräzision zu verbessern. Verfolgen Sie die Prognoseabweichung und führen Sie eine Ursachenanalyse der berichteten Abweichungen durch, um zu reagieren und die Prognosen anzupassen.

Wie in COST01-BP02 Einrichten einer Partnerschaft zwischen Finanzen und Technologie erwähnt, ist es wichtig, eine Partnerschaft mit regelmäßigen Konsultationen zwischen IT, Finanzabteilung und anderen Stakeholdern zu schaffen, um zu bestätigen, dass alle in konsistenter Weise die gleichen Tools oder Prozesse anwenden. Wenn Budgets geändert werden müssen, führen Sie häufigere Besprechungen durch, um schneller darauf zu reagieren.

Implementierungsschritte

- Definieren Sie die Kostensprache innerhalb der Organisation: Schaffen Sie eine gemeinsame AWS-Kostensprache innerhalb der Organisation mit mehreren Dimensionen und Gruppierungen. Stellen Sie sicher, dass die Stakeholder die Granularität der Prognosen, die Preismodelle und das Niveau Ihrer Kostenprognosen verstehen.
- Analysieren Sie trendbasierte Prognosen: Verwenden Sie trendbasierte Prognosetools wie AWS Cost Explorer und Amazon Forecast. Analysieren Sie Ihre Nutzungskosten anhand verschiedener Dimensionen wie Service, Konto, Tags und Kostenkategorien. Wenn fortschrittliche Prognosen benötigt werden, importieren Sie Ihre AWS-Kosten- und Nutzungsdaten (CUR, Cost and Usage Report) in Amazon Forecast. Hier wird lineare Regression als eine Form des Machine Learning auf Prognosen angewendet.
- Analysieren Sie faktorbasierte Prognosen: Identifizieren Sie die Auswirkungen geschäftlicher Faktoren auf Ihre Cloud-Nutzung und erstellen Sie für jeden Faktor eine separate Prognose, um die erwarteten Nutzungskosten im Voraus zu berechnen. Arbeiten Sie eng mit Verantwortlichen von Geschäftseinheiten und Stakeholdern zusammen, um die Auswirkungen auf neue Faktoren zu verstehen und die erwarteten Kostenänderungen zu berechnen. So können Sie genaue Budgets definieren.

- Aktualisieren Sie die bestehenden Prognose- und Budgetprozesse: Definieren Sie Ihre Prozesse für die Prognose und Budgetierung auf Grundlage von bewährten Prognosemethoden, z. B. trendbasiert, geschäftsfaktorenbasiert oder einer Kombination aus beiden Ansätzen. Budgets sollten kalkuliert werden, realistisch sein und auf Ihren Prognosen basieren.
- Konfigurieren Sie Warnmeldungen und Benachrichtigungen: Verwenden Sie AWS Budgets-Warnmeldungen und die Erkennung von Kostenanomalien, um Warnmeldungen und Benachrichtigungen zu erhalten.
- Führen Sie regelmäßige Prüfungen zusammen mit wichtigen Stakeholdern durch: Einigen Sie sich mit Stakeholdern in den Bereichen IT, Finanzen, Plattform usw. auf Änderungen der Unternehmensausrichtung und der Nutzung.

Ressourcen

Zugehörige Dokumente:

- AWS Cost Explorer
- AWS Cost and Usage Report
- Prognosen mit Cost Explorer
- QuickSight-Prognosen
- Amazon Forecast
- AWS Budgets

Zugehörige Videos:

- Wie kann ich AWS Budgets verwenden, um meine Ausgaben und Nutzung zu verfolgen?
- AWS-Serie zur Kostenoptimierung: AWS Budgets

Zugehörige Beispiele:

- Understand and build driver-based forecasting (Faktorbasierte Prognosen verstehen und erstellen)
- How to establish and drive a forecasting culture (Eine Prognosekultur schaffen und fördern)
- How to improve your cloud cost forecasting (Prognosen für Cloud-Kosten optimieren)
- Using the right tools for your cloud cost forecasting (Die richtigen Tools für Cloud-Kostenprognosen verwenden)

COST01-BP04 Implementieren von Kostenbewusstsein in Ihre Organisationsprozesse

Implementieren Sie Kostenbewusstsein und sorgen Sie für Transparenz und Verantwortlichkeit bei neuen oder bestehenden Prozessen, die sich auf die Nutzung auswirken, und greifen Sie auf vorhandene Prozesse zur Steigerung des Kostenbewusstseins zurück. Implementieren Sie Kostenbewusstsein in die Mitarbeiterschulung.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Das Kostenbewusstsein muss in neuen und vorhandenen Organisationsprozessen implementiert werden. Dies ist eine der absoluten Grundlagen für weitere bewährte Methoden. Es wird empfohlen, vorhandene Prozesse nach Möglichkeit wiederzuverwenden und zu ändern. Dadurch werden die Auswirkungen auf Agilität und Geschwindigkeit minimiert. Informieren Sie die Technologieteams und die Entscheidungsträger in den Geschäfts- und Finanzteams über die Cloud-Kosten, um das Kostenbewusstsein zu verbessern, und richten Sie KPIs zur Effizienz für Beteiligte aus dem Finanz- und Geschäftsbereich ein. Die folgenden Empfehlungen helfen Ihnen bei der Implementierung der Kostenerkennung in Ihrem Workload:

- Stellen Sie sicher, dass das Änderungsmanagement eine Kostenmessung umfasst, um die finanziellen Auswirkungen Ihrer Änderungen zu quantifizieren. Auf diese Weise können Sie kostenbezogene Probleme proaktiv lösen und Kosteneinsparungen hervorheben.
- Stellen Sie sicher, dass die Kostenoptimierung eine zentrale Komponente Ihrer Betriebsfunktionen ist. Sie können beispielsweise vorhandene Vorfallmanagementprozesse nutzen, um die Ursache für Kosten- und Nutzungsanomalien (Kostenüberschreitungen) zu ermitteln und zu identifizieren.
- Beschleunigen Sie die Kosteneinsparungen und die Wertschöpfung des Unternehmens durch Automatisierung oder Tools. Wenn Sie über die Kosten der Implementierung nachdenken, sollten Sie das Gespräch so gestalten, dass es eine ROI-Komponente enthält, um die Investition von Zeit oder Geld zu rechtfertigen.
- Weisen Sie Cloud-Kosten zu, indem Sie Showbacks oder Chargebacks für Cloud-Aufwendungen implementieren, einschließlich Aufwendungen für verpflichtungsbasierte Kaufoptionen, gemeinsam genutzte Services und Markt-Einkäufe, um die Cloudnutzung in möglichst kostenbewusster Weise zu gestalten.
- Erweitern Sie vorhandene Schulungs- und Entwicklungsprogramme, um Schulungen zum Kostenbewusstsein in Ihrem gesamten Unternehmen einzubeziehen. Es wird empfohlen, dass dies fortlaufende Schulungen und Zertifizierungen umfasst. Dadurch entsteht ein Unternehmen, das Kosten und Nutzung selbst verwalten kann.

- Nutzen Sie kostenlose, native AWS-Tools, wie etwa AWS Cost Anomaly Detection, AWS Budgetsund AWS Budgets-Berichte.

Wenn Unternehmen Cloud Financial Management (CFM)-Praktiken in konsistenter Weise einsetzen, werden die entsprechenden Verhaltensweisen bald echte Bestandteile der Arbeitsweise und der Entscheidungsfindung. Das führt zu einer kostenbewussteren Kultur, in der Entwickler eine neue, in der Cloud entwickelte Anwendung bauen und Finanzmanager den ROI dieser neuen Cloud-Investitionen analysieren.

Implementierungsschritte

- Bestimmen relevanter organisatorischer Prozesse: Jede Organisationseinheit überprüft ihre Prozesse und identifiziert Prozesse, die sich auf Kosten und Nutzung auswirken. Alle Prozesse, die zur Erstellung oder Beendigung einer Ressource führen, müssen zur Überprüfung einbezogen werden. Suchen Sie auch nach Prozessen, die das Kostenbewusstsein in Ihrem Unternehmen unterstützen können, wie z. B. Vorfallmanagement und Schulungen.
- Schaffen einer sich selbst erhaltenden Kostenbewusstseinskultur: Sorgen Sie dafür, dass alle relevanten Beteiligten die Ursachen für Veränderungen und die damit verbundenen Kosten gut verstehen. So kann Ihr Unternehmen eine sich selbst erhaltende, kostenbewusste Innovationskultur entwickeln.
- Aktualisieren von Prozessen mit Kostenbewusstsein: Jeder Prozess wird so geändert, dass er kostenbewusst wird. Der Prozess erfordert möglicherweise zusätzliche Vorabprüfungen, z. B. die Bewertung der Auswirkungen von Kosten oder nachträgliche Prüfungen, die bestätigen, dass die erwarteten Kosten- und Nutzungsänderungen stattgefunden haben. Unterstützungsprozesse wie Schulungs- und Vorfallmanagement können auf Kosten- und Nutzungselemente erweitert werden.

Wenden Sie sich für Unterstützung über Ihr Account-Team an CFM-Sachverständige oder erkunden Sie die nachfolgend aufgeführten Ressourcen und Dokumente.

Ressourcen

Zugehörige Dokumente:

- AWS Cloud Financial Management

Zugehörige Beispiele:

- Strategie für effizientes Cloud-Kostenmanagement
- Blog-Serie zum Thema Kostenkontrolle Nr. 3: Umgang mit Kostenschocks
- AWS Cost Management für Anfänger

COST01-BP05 Berichte und Benachrichtigungen zur Kostenoptimierung

Richten Sie Cloud-Budgets ein und konfigurieren Sie Mechanismen zur Erkennung von Anomalien bei der Nutzung. Konfigurieren Sie zugehörige Tools für Kosten- und Nutzungswarnungen für vordefinierte Ziele und lassen Sie sich benachrichtigen, wenn eine Nutzung diese Ziele überschreitet. Halten Sie regelmäßig Treffen ab, um die Kosteneffektivität Ihrer Workloads zu analysieren und das Kostenbewusstsein zu stärken.

Risikostufe bei fehlender Befolgung dieser Best Practice: Niedrig

Implementierungsleitfaden

Sie müssen regelmäßig Kosten- und Nutzungsoptimierungen in Ihrem Unternehmen melden. Sie können dedizierte Sitzungen implementieren, um die Kosteneffizienz zu besprechen, oder die Kostenoptimierung in Ihre regulären operativen Berichtszyklen für Ihre Workloads einschließen. Nutzen Sie Services und Tools, um die Kosteneffizienz regelmäßig zu überwachen und Möglichkeiten zur Kosteneinsparung zu nutzen.

Zeigen Sie Ihre Kosten und Nutzung mit mehreren Filtern und Granularität an, indem Sie AWS Cost Explorerverwenden, das Dashboards und Berichte wie Kosten pro Service oder Konto, Tageskosten oder Marktplatzkosten bereitstellt. Sie können Ihren Fortschritt bei Kosten und Nutzung anhand konfigurierter Budgets verfolgen, mit AWS Budgets-Berichte.

Mit AWS Budgets können Sie angepasste Budgets einrichten, um Kosten und Nutzung nachzuverfolgen und schnell auf Warnungen zu reagieren, die Sie per E-Mail oder in Form von Amazon Simple Notification Service (Amazon SNS)-Benachrichtigungen erhalten, wenn Sie Ihren Schwellenwert überschreiten. Sie können den bevorzugten Budgetzeitraum auf täglich, monatlich, vierteljährlich oder jährlich festlegen und spezifische Budgetlimits einrichten, um zu sehen, wie sich die tatsächlichen oder prognostizierten Kosten in Bezug auf Ihren Budgetschwellenwert entwickeln. Sie können auch eine automatische Ausführung von Warnungen und Aktionen oder einen Genehmigungsprozess für den Fall einrichten, dass ein Budgetziel überschritten wird.

Darüber hinaus können Sie mit Benachrichtigungen zu Kosten und Nutzung schnell auf unerwartete Änderungen bei Kosten und Nutzung reagieren. AWS Cost Anomaly Detection ermöglicht Ihnen

die Reduzierung von Überraschungen bei den Kosten und die Verbesserung der Kontrolle, ohne die Innovationsfähigkeit zu beeinträchtigen. AWS Cost Anomaly Detection identifiziert anomale Ausgaben und ihre Ursachen, was Ihnen hilft, das Risiko für Überraschungen bei Abrechnungen zu reduzieren. In drei einfachen Schritten können Sie Ihre eigene kontextorientierte Überwachung einrichten und Benachrichtigungen erhalten, wenn anomale Ausgaben entdeckt werden.

Sie können QuickSight mit AWS Cost and Usage Report (CUR)-Daten verwenden, um hoch angepasste Berichte mit detaillierteren Daten zu erstellen. QuickSight ermöglicht Ihnen die Planung von Berichten und den Erhalt regelmäßiger E-Mails mit Berichten zu historischen Kosten und zur Nutzung oder zu Möglichkeiten für Kosteneinsparungen. Sehen Sie sich unsere Cost Intelligence Dashboard (CID)-Lösung an, die in QuickSight integriert ist und Ihnen erweiterte Transparenz bietet.

Verwenden Sie AWS Trusted Advisorerhalten Sie Anleitungen, mit denen Sie überprüfen können, ob bereitgestellte Ressourcen Best Practices für AWS zur Kostenoptimierung befolgen.

Vergleichen Sie Ihre Savings Plans-Empfehlungen anhand detaillierter grafischer Darstellungen zu Ihren Kosten und der Nutzung. Nach Stunden unterteilte Grafiken zeigen die On-Demand-Ausgaben zusammen mit den empfohlenen Savings Plans-Verpflichtungen und geben Aufschluss über die geschätzten Einsparungen, die Savings Plans-Abdeckung und Savings Plans-Nutzung. Auf diese Weise können Unternehmen nachvollziehen, wie ihre Savings Plans auf jede aufgewendete Stunde angewendet werden, ohne Zeit und Ressourcen in die Erstellung von Modellen zur Analyse ihrer Ausgaben investieren zu müssen.

Sie können regelmäßige Berichte erstellen, die Savings Plans, Reserved Instances und Amazon EC2-Empfehlungen aus AWS Cost Explorer für Anpassungen enthalten, um die Kosten für Steady-State-Workloads sowie nicht genutzte und nicht vollständig genutzte Ressourcen zu reduzieren. Identifizieren Sie unnötige Cloud-Ausgaben, die mit bereitgestellten Ressourcen verbunden sind, und gewinnen Sie diese zurück. Unnötige Cloud-Ausgaben entstehen, wenn Ressourcen mit der falschen Größe erstellt werden oder wenn andere als die erwarteten Nutzungsmuster beobachtet werden. Folgen Sie den Best Practices von AWS, um Ihren Abfall zu reduzieren, oder bitten Sie Ihr Account-Team und Ihren Partner, Ihnen dabei zu helfen, Ihre Cloud-Kosten zu optimieren und zu sparen.

Generieren Sie regelmäßig Berichte zu besseren Kaufoptionen für Ihre Ressourcen, um die Kosten pro Einheit für Ihre Workloads zu senken. Kaufoptionen wie Savings Plans, Reserved Instances oder Amazon EC2 Spot Instances bieten die umfassendsten Kosteneinsparungen für fehlertolerante Workloads. Stakeholder (Geschäftsleitung, Finanz- und Technologieteams) können sich an den Diskussionen zu den damit verbundenen Verpflichtungen beteiligen.

Teilen Sie die Berichte, die Einsparmöglichkeiten beschreiben, oder Ankündigungen neuer Versionen, um die Gesamtbetriebskosten (TCO) der Cloud zu reduzieren. Führen Sie neue Services, Regionen, Funktionen, Lösungen oder neue Möglichkeiten für weitere Kostenreduzierungen ein.

Implementierungsschritte

- Konfigurieren Sie AWS Budgets: Konfigurieren Sie AWS Budgets für alle Konten Ihres Workloads. Legen Sie ein Budget für die Gesamtkontoausgaben und ein Budget für den Workload mithilfe von Tags fest.
  - Well-Architected Labs: Kosten und Steuerung der Nutzung
- Bericht zur Kostenoptimierung: Richten Sie einen regelmäßigen Zyklus ein, um die Effizienz des Workloads zu besprechen und zu analysieren. Melden Sie anhand der eingerichteten Metriken die erreichten Metriken und die Kosten für deren Erreichung. Identifizieren und beheben Sie negative Trends und suchen Sie nach positiven Trends, die Sie in der gesamten Organisation fördern können. Bei der Berichterstellung sollten Vertreter der Anwendungsteams und -Verantwortlichen, Finanzverantwortliche und wichtige Entscheidungsträger in Bezug auf Cloud-Ausgaben einbezogen werden.

Ressourcen

Zugehörige Dokumente:

- AWS Cost Explorer
- AWS Trusted Advisor
- AWS Budgets
- AWS Cost and Usage Report
- Bewährte Methoden für AWS Budgets
- Amazon S3-Analysen

Zugehörige Beispiele:

- Well-Architected Labs: Kosten und Steuerung der Nutzung
- Zentrale Methoden für die Optimierung Ihrer AWS-Cloud-Kosten

COST01-BP06 Proaktive Überwachung der Kosten

Implementieren Sie Tools und Dashboards, um die Kosten proaktiv für den Workload zu überwachen. Überprüfen Sie regelmäßig die Kosten mithilfe konfigurierter oder vorab erstellter Tools. Untersuchen Sie Kosten und Kategorien nicht erst, wenn Sie Benachrichtigungen erhalten. Die proaktive Überwachung und Analyse der Kosten hilft Ihnen, positive Trends zu identifizieren und diese in der gesamten Organisation zu unterstützen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Es wird empfohlen, die Kosten und die Nutzung innerhalb Ihres Unternehmens proaktiv zu überwachen, nicht nur, wenn Ausnahmen oder Anomalien vorliegen. Hoch sichtbare Dashboards in Ihrem Büro oder Ihrer Arbeitsumgebung stellen sicher, dass relevante Mitarbeiter Zugriff auf benötigte Informationen haben, und signalisieren den Fokus des Unternehmens auf Kostenoptimierungen. Mit gut sichtbaren Dashboards können Sie den Erfolg aktiv unterstützen und und positive Ergebnisse in der gesamten Organisation implementieren.

Entwickeln Sie eine tägliche oder häufig ausgeführte Routine für die Verwendung von AWS Cost Explorer oder anderen Dashboards wie Amazon QuickSight, um die Kosten darzustellen und proaktiv zu analysieren. Analysieren Sie mithilfe von Gruppierung und Filterung Kosten und Nutzung von AWS-Services auf der Ebene von AWS-Konten, Workloads oder spezifischen AWS-Services und überprüfen Sie, ob es sich um erwartete oder unerwartete Ergebnisse handelt. Nutzen Sie die Granularität und die Tags auf Stunden- und Ressourcenbasis, um für die wichtigsten Ressourcen wiederkehrende Kosten herauszufiltern und zu identifizieren. Sie können auch über das Cost Intelligence Dashboardeigene Berichte erstellen. Dabei handelt es sich um eine Amazon QuickSight-Lösung, die von AWS Solution Architects entwickelt wurde. Sie ermöglicht Ihnen den Vergleich Ihrer Budgets mit den tatsächlichen Kosten und der tatsächlichen Nutzung.

Implementierungsschritte

- Bericht zur Kostenoptimierung: Richten Sie einen regelmäßigen Zyklus ein, um die Effizienz des Workloads zu besprechen und zu analysieren. Melden Sie anhand der eingerichteten Metriken die erreichten Metriken und die Kosten für deren Erreichung. Identifizieren und beheben Sie negative Trends und suchen Sie nach positiven Trends, um diese in der gesamten Organisation zu unterstützen. Bei der Berichterstellung sollten Vertreter der Anwendungsteams und Besitzer, Finanz- und Geschäftsleitung einbezogen werden.

- Erstellen und aktivieren Sie tägliche, detaillierte AWS Budgets für Kosten und Nutzung, um rechtzeitig Maßnahmen gegen potenzielle Kostenüberschreitungen ergreifen zu können. Mit AWS Budgets können Sie Warnungen konfigurieren, um stets zu wissen, ob ein Budgettyp außerhalb der vorab konfigurierten Schwellenwerte liegt. Die beste Art, AWS Budgets zu nutzen, besteht in der Einrichtung der erwarteten Kosten und der erwarteten Nutzung als Grenzwerte. So können alle Budgetüberschreitungen identifiziert werden.
- Erstellen Sie AWS Cost Anomaly Detection zur Kostenüberwachung: AWS Cost Anomaly Detection verwendet eine erweiterte Machine-Learning-Technologie, um anomale Ausgaben und ihre Ursachen schnell zu identifizieren, damit Sie schnell Maßnahmen ergreifen können. Sie können auf diese Weise Tools für die Überwachung der Kosten von Ausgabensegmenten konfigurieren, die Sie überwachen möchten (z. B. einzelne AWS-Services, Mitgliederkonten, Kostenzuweisungs-Tags und Kostenkategorien). Sie können auch festlegen, wann, wo und wie Sie Warnungen erhalten. Jedem Überwachungstool können Sie mehrere Warnungsabonnements für Geschäftsbereichsleiter und Technologieteams anfügen, einschließlich Name, Kostenschwellenwert und Häufigkeit (einzelne Warnungen, tägliche Zusammenfassung, wöchentliche Zusammenfassung) für die einzelnen Abonnements.
- Verwenden Sie AWS Cost Explorer oder integrieren Sie Ihre AWS Cost and Usage Report (CUR)-Daten in Amazon QuickSight-Dashboards, um die Kosten Ihrer Organisation zu visualisieren: AWS Cost Explorer besitzt eine benutzerfreundliche Oberfläche, in der Sie AWS-Kosten und -Nutzung über die Zeit visualisieren, verstehen und verwalten können. Das Cost Intelligence Dashboard ist ein anpassbares und zugängliches Dashboard, mit dem Sie die Grundlagen für Ihr eigenes Tool für Kostenmanagement und Optimierung legen können.

Ressourcen

Zugehörige Dokumente:

- AWS Budgets
- AWS Cost Explorer
- Tägliche Kosten und Nutzungsbudgets
- AWS Cost Anomaly Detection

Zugehörige Beispiele:

- Well-Architected Labs: Visualisierung
- Well-Architected Labs: Erweiterte Visualisierung

- Well-Architected Labs: Cloud Intelligence Dashboards
- Well-Architected Labs: Kostenvisualisierung
- AWS Cost Anomaly Detection-Warnung mit Slack

COST01-BP07 Verfolgen neuer Serviceversionen

Konsultieren Sie regelmäßig Experten oder AWS-Partner, um zu prüfen, welche Services und Features kostengünstiger sind. Lesen Sie AWS-Blogs und sonstige Informationsquellen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

AWS fügt ständig neue Funktionen hinzu, so dass Ihnen die neuesten Technologien zur Verfügung stehen, damit Sie experimentieren und Innovationen schneller einführen können. Sie können möglicherweise neue AWS-Services und -Features implementieren, um die Kosteneffizienz Ihres Workloads zu erhöhen. Lesen Sie regelmäßig AWS Cost Management, den AWS News-Blog, den AWS Cost Management-Blogund Neuerungen bei AWS, um Informationen zur Veröffentlichung neuer Services und Features zu erhalten. Die Posts in „Neuerungen“ bieten eine kurze Übersicht über alle Ankündigungen für AWS-Services, -Features und -Regionserweiterungen bei Veröffentlichung.

Implementierungsschritte

- Abonnieren Sie Blogs: Rufen Sie die Seiten für AWS-Blogs auf und abonnieren Sie den Blog „Neuerungen“ und andere relevante Blogs. Sie können sich auf der Seite für die Kommunikationseinstellungen mit Ihrer E-Mail-Adresse registrieren.
- Abonnieren Sie AWS-Nachrichten: Lesen Sie regelmäßig den AWS News-Blog und Neuerungen bei AWS, um Informationen zur Veröffentlichung neuer Services und Features zu erhalten. Abonnieren Sie den RSS-Feed oder registrieren Sie sich über Ihre E-Mail-Adresse, um Ankündigungen und Veröffentlichungen zu folgen.
- Verfolgen Sie AWSPreisreduzierungen: Wir geben die wirtschaftliche Effizienz, die wir aufgrund unserer Skalierbarkeit erzielen, mit regelmäßigen Preissenkungen für alle unsere Services als AWS-Standardverfahren an unsere Kunden weiter. Bis zum Jahr 2024 hat AWS die Preise seit der Einführung im Jahr 2006 115 Mal gesenkt. Wenn geschäftliche Entscheidungen aufgrund von Preisbedenken ausstehen, können Sie die Preise nach der Reduzierung und der Integration neuer Services erneut prüfen. Informationen zu früheren Preissenkungen, einschließlich Preissenkungen für Amazon Elastic Compute Cloud (Amazon EC2)-Instances, finden Sie in der Kategorie „Preissenkungen“ im AWS News-Blog.

- AWS-Veranstaltungen und -Treffen: Nehmen Sie am lokalen AWS-Summit und weiteren lokalen Treffen mit anderen Organisationen aus Ihrer Region teil. Wenn eine persönliche Teilnahme nicht möglich ist, können Sie in virtuellen Veranstaltungen mehr von AWS-Experten und über die Business Cases anderer Kunden erfahren.
- Treffen Sie sich mit Ihrem Account-Team: Planen Sie regelmäßige Treffen mit Ihrem Account-Team, um über Branchentrends und AWS-Services zu sprechen. Sprechen Sie mit Ihrem Account Manager, Solutions Architekt und Support-Team.

Ressourcen

Zugehörige Dokumente:

- AWS Cost Management
- Neuerungen bei AWS,
- AWS News-Blog

Zugehörige Beispiele:

- Amazon EC2 – 15 Years of Optimizing and Saving Your IT Costs
- AWS News-Blog – Preisreduzierung

COST01-BP08 Schaffen einer kostenbewussten Kultur

Implementieren Sie Änderungen oder Programme in Ihrem gesamten Unternehmen, um eine kostenbewusste Kultur zu schaffen. Es wird empfohlen, klein zu beginnen. Wenn Ihre Kompetenz und die Nutzung der Cloud in Ihrem Unternehmen zunehmen, implementieren Sie große und umfangreiche Programme.

Risikostufe bei fehlender Befolgung dieser Best Practice: Niedrig

Implementierungsleitfaden

Eine kostenbewusste Kultur ermöglicht Ihnen die Skalierung von Kostenoptimierung und Cloud-Finanzmanagement (operative Abläufe, Cloud-Kompetenzzentrum, Cloud Operations Teams usw.) mithilfe von Best Practices, die in der gesamten Organisation auf organische und dezentralisierte Weise angewendet werden. Wenn Sie ein Kostenbewusstsein entwickeln, können Sie im Vergleich zu einem zentralisierten Top-Down-Approach in der gesamten Organisation mit minimalem Aufwand einen hohen Grad an Kompetenz erzielen.

Die Entwicklung eines Kostenbewusstseins im Bereich Cloud-Computing, insbesondere bei primären Kostenfaktoren, ermöglicht Teams, die voraussichtlichen Ergebnisse von Änderungen aus Kostensicht zu verstehen. Teams, die auf Cloud-Umgebungen zugreifen, sollten die Preismodelle kennen und den Unterschied zwischen herkömmlichen On-Premises-Rechenzentren und Cloud-Computing verstehen.

Der Hauptvorteil einer Kultur des Kostenbewusstseins besteht darin, dass Technologieteams die Kosten proaktiv und kontinuierlich optimieren, statt bedarfsbasiert reaktive Kostenoptimierungen durchzuführen. (Die Kosten werden beispielsweise als eine nicht funktionale Anforderung betrachtet, wenn neue Workloads entwickelt oder vorhandene Workloads geändert werden.)

Kleine Veränderungen in der Kultur können große Auswirkungen auf die Effizienz Ihrer aktuellen und zukünftigen Workloads haben. Beispiele hierfür sind:

- Transparenz und Schaffung eines Bewusstseins bei Entwicklungsteams, damit diese verstehen, was sie tun und wie sich dies auf die Kosten auswirkt.
- Gamifizierung von Kosten und Nutzung in Ihrem gesamten Unternehmen. Dies kann über ein öffentliches Dashboard oder einen Bericht erfolgen, der Kosten und Nutzung normalisiert und teamübergreifend vergleicht (z. B. Kosten pro Workload und Kosten pro Transaktion).
- Kosteneffizienz erkennen. Belohnen Sie freiwillige oder unaufgeforderte Kostenoptimierungsleistungen öffentlich oder privat und lernen Sie aus Fehlern, um eine Wiederholung in Zukunft zu vermeiden.
- Erstellen Sie Top-Down-Organisationsanforderungen für die Ausführung von Workloads mit vordefinierten Budgets.
- Hinterfragen Sie die geschäftlichen Anforderungen in Bezug auf Änderungen und die Kostenauswirkungen von Änderungsanforderungen für die Architekturinfrastruktur oder die Workload-Konfiguration, um sicherzustellen, dass Sie nur für das bezahlen, was Sie benötigen.
- Stellen Sie sicher, dass sich Änderungsplaner voraussichtlicher Änderungen mit Auswirkungen auf die Kosten bewusst sind und dass diese Änderungen von den Stakeholdern genehmigt werden, um geschäftliche Ergebnisse auf kosteneffektive Weise zu erzielen.

Implementierungsschritte

- Informieren Sie die Technologieteams über die Cloud-Kosten: So erhöhen Sie das Kostenbewusstsein und können Effizienz-KPIs für Stakeholder in den Bereichen Finanzen und Geschäft einrichten.

- Informieren Sie Stakeholder oder Teammitglieder über geplante Änderungen: Erstellen Sie einen Tagesordnungspunkt zur Erörterung geplanter Änderungen und der Kosten-Nutzen-Auswirkungen auf die Arbeitsbelastung während der wöchentlichen Änderungsbesprechungen.
- Treffen Sie sich mit Ihrem Account-Team: Richten Sie regelmäßige Treffen mit Ihrem Account-Team ein, um über Branchentrends und AWS-Services zu sprechen. Sprechen Sie mit Ihrem Account Manager, Solutions Architect und Support-Team.
- Teilen Sie Erfolgsgeschichten: Teilen Sie Erfolgsgeschichten zu Kostensenkungen für einen Workload, ein AWS-Konto oder eine Abteilung, um eine positive Einstellung zu generieren und zu Kostensenkungen zu motivieren.
- Schulungen: Stellen Sie sicher, dass Technologieteams oder Teammitglieder in Bezug auf die Ressourcenkosten in AWS Cloud geschult sind.
- AWS-Veranstaltungen und -Treffen: Nehmen Sie an lokalen AWS-Summits und weiteren lokalen Treffen mit anderen Organisationen aus Ihrer Region teil.
- Abonnieren Sie Blogs: Rufen Sie die AWS-Blogs-Seiten auf und abonnieren Sie den Blog „Neuerungen“ und weitere relevante Blogs, um bei neuen Veröffentlichungen, Implementierungen, Beispielen und Änderungen auf dem Laufenden zu bleiben, die von AWS geteilt werden.

Ressourcen

Zugehörige Dokumente:

- AWS-Blog
- AWS Cost Management
- AWS News-Blog

Zugehörige Beispiele:

- AWS Cloud Financial Management
- AWS Well-Architected Labs: Cloud Financial Management

COST01-BP09 Quantifizieren des Geschäftswerts von Kostenoptimierungen

Durch die Quantifizierung des Geschäftswerts von Kostenoptimierungen können Sie die gesamten Vorteile für Ihr Unternehmen verstehen. Da die Kostenoptimierung eine notwendige Investition ist, können Sie durch die Quantifizierung des Geschäftswerts den Beteiligten den ROI erklären. Die

Quantifizierung des Geschäftswerts kann Ihnen helfen, mehr Unterstützung von Beteiligten für zukünftige Investitionen zur Kostenoptimierung zu gewinnen, und bietet einen Rahmen, um die Ergebnisse für die Kostenoptimierung Ihres Unternehmens zu messen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Quantifizierung des Geschäftswerts bedeutet, dass der Nutzen gemessen wird, den Unternehmen aus ihren Maßnahmen und Entscheidungen ziehen. Bei dem Geschäftswert kann es sich um einen materiellen Wert (z. B. geringere Ausgaben oder höhere Gewinne) oder einen immateriellen Wert (z. B. ein besseres Markenimage oder eine höhere Kundenzufriedenheit) handeln.

Die Quantifizierung des geschäftlichen Nutzens der Kostenoptimierung bedeutet, dass Sie feststellen müssen, wie viel Wert oder Nutzen Sie aus Ihren Bemühungen um effizientere Ausgaben ziehen. Wenn ein Unternehmen beispielsweise 100.000 USD für die Bereitstellung eines Workloads in AWS ausgibt und diesen später optimiert, betragen die neuen Kosten nur noch 80.000 USD, ohne dass die Qualität oder Ausgabe darunter leiden. In diesem Szenario würde der quantifizierte Geschäftswert aus der Kostenoptimierung eine Einsparung von 20.000 USD bedeuten. Aber über die reinen Einsparungen hinaus kann das Unternehmen den Wert auch in Form von kürzeren Lieferzeiten, verbesserter Kundenzufriedenheit oder anderen Kennzahlen, die sich aus den Kostenoptimierungsbemühungen ergeben, quantifizieren. Die Beteiligten müssen Entscheidungen über den potenziellen Wert der Kostenoptimierung, die Kosten für die Optimierung des Workloads und den Ertragswert treffen.

Zusätzlich zu den Einsparungen durch Kostenoptimierung wird empfohlen, den zusätzlichen Wert zu quantifizieren. Die Vorteile der Kostenoptimierung werden in der Regel in Bezug auf niedrigere Kosten pro Geschäftsergebnis quantifiziert. Sie können z. B. Amazon Elastic Compute Cloud (Amazon EC2) Kosteneinsparungen beziffern, wenn Sie Savings Plans kaufen, die die Kosten senken und das Niveau der Workload-Ausgabe beibehalten. Sie können die Kostensenkungen bei den AWS-Ausgaben quantifizieren, wenn ungenutzte Amazon EC2-Instances entfernt oder unverbundene Amazon Elastic Block Store (Amazon EBS) Volumes gelöscht werden.

Die Vorteile der Kostenoptimierung gehen jedoch über die Kostensenkung oder -vermeidung hinaus. Ziehen Sie in Betracht, zusätzliche Daten zu erfassen, um Effizienzsteigerungen und Geschäftswert zu messen.

Implementierungsschritte

- Bewertung der Geschäftsvorteile: Bei diesem Prozess werden die AWS Cloud-Kosten so analysiert und angepasst, dass der Nutzen für jeden ausgegebenen Dollar maximiert wird. Anstatt sich auf Kostensenkungen ohne geschäftlichen Nutzen zu konzentrieren, sollten Sie die Geschäftsvorteile und die Kapitalrendite für die Kostenoptimierung in Betracht ziehen, da diese einen größeren Nutzen aus den von Ihnen ausgegebenen Mitteln ziehen können. Dabei geht es darum, Ausgaben umsichtig vorzunehmen und Investitionen und Ausgaben in Bereichen zu tätigen, die den besten Ertrag bringen.
- Analysieren der Vorhersage der AWS-Kosten: Prognosen helfen den Finanzverantwortlichen dabei, die Erwartungen anderer interner und externer Stakeholder der Organisation festzulegen und die Finanzplanung Ihrer Organisation zu verbessern.AWS Cost Explorer kann für die Durchführung Ihrer Kosten- und Nutzungsprognosen verwendet werden.

Ressourcen

Zugehörige Dokumente:

- AWS Cloud Economics
- AWS-Blog
- AWS-Kostenmanagement
- AWS News-Blog
- Well-Architected Whitepaper zur Säule "Zuverlässigkeit"
- AWS Cost Explorer

Zugehörige Videos:

- Unlock Business Value with Windows on AWS(Erschließen des Unternehmenswertes mit Windows in AWS)

Zugehörige Beispiele:

- Den Geschäftswert von Customer 360 bestimmen und optimieren
- The Business Value of Adopting Amazon Web Services Managed Databases (Der geschäftliche Nutzen durch die Einführung von durch Amazon Web Services verwalteten Datenbanken)

- The Business Value of Amazon Web Services for Independent Software Vendors (Der Unternehmenswert von Amazon Web Services für unabhängige Softwareanbieter)
- Der Unternehmenswert der Cloud-Modernisierung
- Der geschäftliche Nutzen der Migration zu Amazon Web Services

Ausgabenerkennung und Nutzungsbewusstsein

Fragen

- KOSTEN 2. Wie können Sie die Nutzung steuern?
- KOSTEN 3. Wie überwachen Sie Ihre Kosten und die Nutzung?
- KOSTEN 4. Wie können Sie Ressourcen außer Betrieb nehmen?

KOSTEN 2. Wie können Sie die Nutzung steuern?

Legen Sie Richtlinien und Mechanismen fest, um zu überprüfen, ob angemessene Kosten anfallen und die Ziele erreicht werden. Durch den Einsatz eines Kontrollsystems können Sie Innovationen vorantreiben, ohne das Budget zu überschreiten.

Bewährte Methoden

- COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen
- COST02-BP02 Implementieren von Zielen und Ergebnissen
- COST02-BP03 Implementieren einer Kontenstruktur
- COST02-BP04 Implementieren von Gruppen und Rollen
- COST02-BP05 Implementieren von Kostenkontrollen
- COST02-BP06 Verfolgen des Projektlebenszyklus

COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen

Entwickeln Sie Richtlinien, die definieren, wie Ressourcen von Ihrem Unternehmen verwaltet werden, und überprüfen Sie sie regelmäßig. Die Richtlinien sollten sich auch mit den Kostenaspekten der Ressourcen und Workloads befassen, einschließlich Erstellung, Änderung und Außerbetriebnahme während der gesamten Lebensdauer der Ressourcen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Die Kenntnis der Kostentreiber in Ihrem Unternehmen ist für die effektive Verwaltung Ihrer Ausgaben und Nutzung und die Identifizierung von Kostenreduzierungsmöglichkeiten von entscheidender Bedeutung. Unternehmen betreiben in der Regel mehrere Workloads, die von mehreren Teams ausgeführt werden. Diese Teams können sich in verschiedenen Organisationseinheiten befinden, die jeweils über eigene Einnahmequellen verfügen. Die Möglichkeit, die Ressourcenkosten den Workloads, der jeweiligen Organisation oder den Produkteigentümern zuzuordnen, fördert ein effizientes Nutzungsverhalten und hilft, die Verschwendung von Ressourcen einzudämmen. Eine genaue Kosten- und Nutzungsüberwachung hilft Ihnen zu verstehen, wie optimiert ein Workload ist und wie profitabel Geschäftsbereiche und Produkte sind. Mit diesem Wissen können Sie fundiertere Entscheidungen dazu treffen, wo Ressourcen in Ihrem Unternehmen eingesetzt werden sollen. Das Bewusstsein der Nutzung auf allen Unternehmensebenen ist entscheidend für Veränderungen, da eine Änderung der Nutzung zu Kostenänderungen führt. Überlegen Sie sich, beim Ermitteln von Nutzungsmustern und Ausgaben einen mehrschichtigen Ansatz zu nutzen.

Der erste Schritt bei der Implementierung von Governance besteht darin, Richtlinien für die Cloud-Nutzung anhand der Anforderungen Ihres Unternehmens zu entwickeln. Diese Richtlinien definieren, wie Ihr Unternehmen die Cloud verwendet und wie Ressourcen verwaltet werden. Richtlinien sollten alle Aspekte von Ressourcen und Workloads abdecken, die sich auf Kosten oder Nutzung beziehen, einschließlich Erstellung, Änderung und Außerbetriebnahme über die Lebensdauer der Ressource. Überprüfen Sie, ob die Richtlinien und Verfahren bei jeder Änderung in einer Cloud-Umgebung eingehalten und umgesetzt werden. Stellen Sie bei Ihren IT-Änderungsmanagement-Meetings Fragen zu den Kostenauswirkungen geplanter Änderungen, ob die Kosten steigen oder sinken, zur geschäftlichen Rechtfertigung und zum erwarteten Ergebnis.

Richtlinien sollten einfach sein, damit sie leicht verständlich sind und im gesamten Unternehmen effektiv implementiert werden können. Richtlinien müssen außerdem leicht zu befolgen und zu interpretieren sein (damit sie angewendet werden können) sowie spezifisch sein (keine Fehlinterpretationen zwischen den Teams). Darüber hinaus müssen sie (wie unsere Mechanismen) regelmäßig überprüft und aktualisiert werden, wenn sich die Geschäftsbedingungen oder Prioritäten der Kunden ändern, wodurch die Richtlinie veraltet wäre.

Beginnen Sie mit umfangreichen allgemeinen Richtlinien, z. B. welche geografische Region verwendet werden soll oder zu welchen Tageszeiten Ressourcen ausgeführt werden sollen. Verfeinern Sie schrittweise die Richtlinien für die verschiedenen Organisationseinheiten und Workloads. Zu den allgemeinen Richtlinien gehört, welche Services und Funktionen verwendet werden können (z. B. Speicher mit niedrigerer Leistung in Test- und Entwicklungsumgebungen),

welche Ressourcentypen von verschiedenen Gruppen verwendet werden können (z. B. ist die größte Ressource in einem Entwicklungskonto mittelgroß) und wie lange diese Ressourcen verwendet werden (ob vorübergehend, kurzfristig oder für einen bestimmten Zeitraum).

Richtlinien-Beispiel

Im Folgenden finden Sie eine Beispielrichtlinie, die Sie überprüfen können, um Ihre eigenen Cloud-Governance-Richtlinien zur Kostenoptimierung zu erstellen. Stellen Sie sicher, dass Sie die Richtlinien an die Anforderungen Ihres Unternehmens und die Anforderungen Ihrer Interessenvertreter anpassen.

- Name der Richtlinie: Definieren Sie einen eindeutigen Namen für die Richtlinie, z. B. Richtlinie zur Ressourcenoptimierung und Kostenreduzierung.
- Zweck: Erläutern Sie, warum diese Richtlinie angewendet werden sollte und was das erwartete Ergebnis ist. Mit dieser Richtlinie soll überprüft werden, ob für die Bereitstellung und Ausführung des gewünschten Workloads Mindestkosten anfallen, um die Geschäftsanforderungen zu erfüllen.
- Umfang: Definieren Sie klar, wer diese Richtlinie verwenden soll und wann sie verwendet werden soll, z. B. DevOps X-Team für Kunden im Osten der USA für Umgebung X (Produktion oder Nicht-Produktion).

Grundsatzerklärung

1. Wählen Sie basierend auf der Umgebung Ihres Workloads und den Geschäftsanforderungen (Entwicklung, Benutzerakzeptanztests, Vorproduktion oder Produktion) entweder us-east-1 oder mehrere us-east-Regionen aus.
2. Planen Sie die Ausführung von Amazon EC2- und Amazon RDS-Instances zwischen sechs Uhr morgens und acht Uhr abends (Eastern Standard Time (EST)).
3. Stoppen Sie alle ungenutzten Amazon EC2-Instances nach acht Stunden und nicht genutzte Amazon RDS-Instances nach 24 Stunden Inaktivität.
4. Beenden Sie alle ungenutzten Amazon EC2-Instances nach 24 Stunden Inaktivität in Nicht-Produktionsumgebungen. Erinnern Sie den Amazon EC2-Instance-Besitzer (anhand von Tags) daran, seine gestoppten Amazon EC2-Instances in der Produktion zu überprüfen, und teilen Sie ihm mit, dass seine Amazon EC2-Instances innerhalb von 72 Stunden beendet werden, wenn sie nicht verwendet werden.
5. Verwenden Sie eine generische Instance-Familie und -größe wie m5.large und passen Sie dann die Größe der Instance anhand der CPU- und Speicherauslastung mithilfe von AWS Compute Optimizer an.

6. Priorisieren Sie mithilfe von Auto Scaling, um die Anzahl der ausgeführten Instances je nach Datenverkehr dynamisch anzupassen.
7. Verwenden Sie Spot-Instances für unkritische Workloads.
8. Prüfen Sie die Kapazitätsanforderungen, um Speicherpläne oder Reserved-Instances für vorhersehbare Workloads festzulegen, und informieren Sie das Cloud-Financial-Management-Team.
9. Verwenden Sie Amazon S3-Lebenszyklusrichtlinien, um Daten, auf die selten zugegriffen wird, auf günstigere Speicherebenen zu verschieben. Wenn keine Aufbewahrungsrichtlinie definiert ist, verwenden Sie Amazon S3 Intelligent Tiering, um Objekte automatisch auf die Archivebene zu verschieben.
10. Überwachen Sie die Ressourcenauslastung und richten Sie mithilfe von Amazon CloudWatch Alarme ein, um Skalierungsereignisse auszulösen.
11. Verwenden Sie für jedes AWS-Konto AWS Budgets, um die Kosten- und Nutzungsbudgets für Ihr Konto basierend auf Kostenstelle und Geschäftsbereichen festzulegen.
12. Indem Sie für Ihr Konto mithilfe von AWS Budgets Kosten- und Nutzungsbudgets festlegen, behalten Sie die Ausgaben im Blick und vermeiden unerwartete Rechnungen, was Ihnen eine bessere Kostenkontrolle ermöglicht.

Verfahren: Richten Sie detaillierte Verfahren für die Umsetzung dieser Richtlinie ein oder verweisen Sie auf andere Dokumente, in denen beschrieben wird, wie die einzelnen Grundsatzerklärungen umgesetzt werden. Dieser Abschnitt sollte schrittweise Anweisungen zur Erfüllung der Richtlinienanforderungen enthalten.

Zur Umsetzung dieser Richtlinie können Sie verschiedene Tools von Drittanbietern oder AWS Config-Regeln verwenden, um die Einhaltung der Richtlinienerklärung zu überprüfen und mithilfe von AWS Lambda-Funktionen automatische Abhilfemaßnahmen auszulösen. Sie können auch AWS Organizations verwenden, um die Richtlinie durchzusetzen. Darüber hinaus sollten Sie Ihre Ressourcennutzung regelmäßig überprüfen und die Richtlinie bei Bedarf anpassen, um sicherzustellen, dass sie weiterhin Ihren Geschäftsanforderungen entspricht.

Implementierungsschritte

- Treffen mit Interessenvertretern: Um Richtlinien zu entwickeln, bitten Sie die Interessenvertreter (Cloud-Geschäftsstellen, Techniker oder funktionale Entscheidungsträger für die Durchsetzung von Richtlinien) innerhalb Ihres Unternehmens, ihre Anforderungen festzulegen und zu dokumentieren. Führen Sie einen iterativen Ansatz aus, indem Sie bei jedem Schritt umfassend beginnen und

kontinuierlich auf die kleinsten Einheiten verfeinern. Zu den Teammitgliedern gehören Personen mit direktem Interesse am Workload, z. B. Organisationseinheiten oder Anwendungsbesitzer sowie unterstützende Gruppen wie Sicherheits- und Finanzteams.

- Bestätigung einholen: Vergewissern Sie sich, dass sich diejenigen Teams auf Richtlinien einigen, die auf die AWS Cloud Zugriff haben und darin Bereitstellungen vornehmen können. Sorgen Sie dafür, dass sie die Richtlinien Ihres Unternehmens befolgen und stellen Sie sicher, dass ihre Ressourcenerstellung mit den vereinbarten Richtlinien und Verfahren übereinstimmt.
- Onboarding-Trainings veranstalten: Fordern Sie neue Unternehmensmitarbeiter auf, Onboarding-Trainings zu absolvieren, um ein Kostenbewusstsein und ein Verständnis für die Unternehmensanforderungen zu schaffen. Möglicherweise gehen neue Unternehmensmitarbeiter aufgrund ihrer bisherigen Erfahrungen von anderen Richtlinien aus oder denken überhaupt nicht daran.
- Festlegen der Speicherorte für Ihren Workload: Definieren Sie, wo Ihr Workload ausgeführt wird, einschließlich des Landes und der Region innerhalb des Landes. Diese Informationen werden für die Zuweisung zu AWS-Regionen und Availability Zones verwendet.
- Definieren und Gruppieren von Services und Ressourcen: Definieren Sie die Services, die für die Workloads erforderlich sind. Geben Sie für jeden Service die Typen, den Umfang und die Anzahl der erforderlichen Ressourcen an. Definieren Sie Gruppen für die Ressourcen nach Funktion, z. B. Anwendungsserver oder Datenbankspeicher. Ressourcen können mehreren Gruppen angehören.
- Definieren und Gruppieren der Benutzer nach Funktion: Definieren Sie die Benutzer, die mit dem Workload interagieren, und konzentrieren Sie sich darauf, was sie tun und wie sie den Workload verwenden, nicht auf die Benutzer oder ihre Position in der Organisation. Fassen Sie ähnliche Benutzer oder Funktionen in einer Gruppe zusammen. Sie können die von AWS verwalteten Richtlinien als Leitfaden verwenden.
- Definieren der Aktionen: Definieren Sie mithilfe der zuvor identifizierten Standorte, Ressourcen und Benutzer die Aktionen, die von jedem benötigt werden, um die Workload-Ergebnisse über die Lebensdauer (Entwicklung, Betrieb und Außerbetriebnahme) zu erzielen. Identifizieren Sie die Aktionen an jedem Standort basierend auf den Gruppen, nicht auf den einzelnen Elementen in den Gruppen. Beginnen Sie umfassend mit Lese- oder Schreibvorgängen und verfeinern Sie dann auf bestimmte Aktionen für jeden Service.
- Definieren des Überprüfungszeitraums: Workloads und Organisationsanforderungen können sich im Laufe der Zeit ändern. Definieren Sie den Zeitplan für die Überprüfung des Workloads, um sicherzustellen, dass er mit den Prioritäten der Organisation übereinstimmt.

- Dokumentieren der Richtlinien: Stellen Sie sicher, dass auf die definierten Richtlinien zugegriffen werden kann, wie von Ihrer Organisation gefordert. Diese Richtlinien werden verwendet, um den Zugriff auf Ihre Umgebungen zu implementieren, zu verwalten und zu prüfen.

Ressourcen

Zugehörige Dokumente:

- Änderungsmanagement in der Cloud
- Von AWS verwaltete Richtlinien für Auftragsfunktionen
- AWS-Fakturierungsstrategie mit mehreren Konten
- Aktionen, Ressourcen und Bedingungsschlüssel für AWS-Services
- AWS Management und Governance
- Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien
- Globale Infrastruktur-Regionen und -AZs

Zugehörige Videos:

- AWS-Management and Governance in großem Umfang

Zugehörige Beispiele:

- VMware – was sind Cloud-Richtlinien?

COST02-BP02 Implementieren von Zielen und Ergebnissen

Implementieren Sie Kosten- und Nutzungsziele sowie Vorgaben für Ihren Workload. Ziele geben Ihrem Unternehmen die Richtung für die erwarteten Ergebnisse vor, und Vorgaben geben spezifische, messbare Ergebnisse vor, die für Ihre Workloads erreicht werden sollen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Entwickeln Sie Kosten- und Nutzungsziele sowie Vorgaben für Ihr Unternehmen. Als wachsende Organisation mit AWS ist es für Sie wichtig, Ziele zur Kostenoptimierung zu setzen und diese zu verfolgen. Diese Ziele oder Key Performance Indicators (KPIs) können Dinge wie den Prozentsatz

der Bedarfsausgaben oder die Einführung bestimmter optimierter Services wie AWS Graviton-Instances oder gp3-EBS-Volumetypen umfassen. Legen Sie messbare und erreichbare Ziele fest, anhand derer Sie Effizienzverbesserungen bewerten können, was für den Geschäftsbetrieb wichtig ist. Ziele bieten Ihrer Organisation richtungsweisende Anleitungen hinsichtlich der erwarteten Ergebnisse.

Vorgaben bieten spezifische messbare Ergebnisse, die erreicht werden müssen. Kurz gesagt: Ein Ziel ist die Richtung, in die Sie gehen wollen, und die Vorgabe ist, wie weit Sie in diese Richtung gehen und wann dieses Ziel erreicht werden soll (verwenden Sie die SMART-Orientierungshilfe (Specific, Measurable, Assignable, Realistic, and Timely), d. h. spezifische, messbare, zuweisbare, realistische und zeitgerechte Ziele). Ein Beispiel für ein Ziel ist, dass die Nutzung der Plattform deutlich steigen soll, wobei die Kosten nur geringfügig (nicht linear) steigen sollen. Ein Beispiel für eine Vorgabe ist eine Steigerung der Plattformnutzung um 20 % bei einem Kostenanstieg von weniger als fünf Prozent. Ein weiteres häufiges Ziel ist, dass Workloads alle sechs Monate effizienter werden müssen. Die damit verbundene Vorgabe wäre, dass die Metriken für die Kosten pro Unternehmen alle 6 Monate um 5 Prozent sinken müssen. Verwenden Sie die richtigen Metriken und legen Sie berechnete KPIs für Ihre Organisation fest. Sie können mit grundlegenden KPIs beginnen und diese später je nach Geschäftsanforderungen weiterentwickeln.

Ein Ziel der Kostenoptimierung besteht darin, die Workload-Effizienz zu erhöhen, also die Kosten pro Geschäftsergebnis des Workloads im Laufe der Zeit zu senken. Implementieren Sie dieses Ziel für alle Workloads und legen Sie als Vorgabe beispielsweise eine 5-prozentige Steigerung der Effizienz alle 6 Monate bis zu 1 Jahr fest. In der Cloud können Sie dies durch den Aufbau von Funktionen zur Kostenoptimierung sowie durch neue Service- und Feature-Releases erreichen.

Vorgaben sind die quantifizierbaren Benchmarks, die Sie anstreben, um Ihre Ziele zu erreichen, und mithilfe von Benchmarks werden die tatsächlichen Ergebnisse mit einer Vorgabe verglichen. Erstellen Sie Benchmarks mit KPIs für die Kosten pro Einheit von Computing-Services (wie Spot-Einführung, Graviton-Einführung, neueste Instance-Typen und On-Demand-Abdeckung), Speicherdiensten (wie EBS GP3-Einführung, überholte EBS-Snapshots und Amazon S3-Standardspeicher) oder die Nutzung von Datenbank-Services (wie RDS-Open-Source-Engines, Graviton-Einführung und On-Demand-Abdeckung). Mithilfe dieser Benchmarks und KPIs können Sie überprüfen, ob Sie AWS-Dienste auf die kostengünstigste Weise nutzen.

Die folgende Tabelle enthält eine Liste von AWS-Standardmetriken als Referenz. Jede Organisation kann unterschiedliche Zielwerte für diese KPIs haben.

Category	KPI (%)	Description
Compute	EC2 usage Coverage	EC2 instances (in cost or hours) using SP+RI+Spot compared to total (in cost or hours) of EC2 instances
Compute	Compute SP/RI utilization	Utilized SP or RI hours compared to total available SP or RI hours
Compute	EC2/Hour cost	EC2 cost divided by the number of EC2 instances running in that hour
Compute	vCPU cost	Cost per vCPU for all instances
Compute	Latest Instance Generation	Percentage of instances on Graviton (or other modern generation instance types)
Database	RDS coverage	RDS instances (in cost or hours) using RI compared to total (in cost or hours) of RDS instances
Database	RDS utilization	Utilized RI hours compared to total available RI hours
Database	RDS uptime	RDS cost divided by the number of RDS instances running in that hour
Database	Latest Instance Generation	Percentage of instances on Graviton (or other modern instance types)

Category	KPI (%)	Description
Storage	Storage utilization	Optimized storage cost (for example Glacier, deep archive, or Infrequent Access) divided by total storage cost
Tagging	Untagged resources	Cost Explorer: 1. Filtern Sie Gutschriften, Rabatte, Steuern, Rückerstattungen und Marketplace heraus und kopieren Sie die aktuellen Monatskosten. 2. Wählen Sie Nur Ressourcen ohne Tag anzeigen in Cost Explorer aus. 3. Teilen Sie den Betrag in Ressourcen ohne Tag durch Ihre Monatskosten.

Geben Sie anhand dieser Tabelle die Ziel- oder Benchmarkwerte an, die auf der Grundlage Ihrer Organisationsziele berechnet werden sollten. Sie müssen bestimmte Metriken für Ihr Unternehmen messen und die Geschäftsergebnisse für diese Workload verstehen, um genaue und realistische KPIs definieren zu können. Unterscheiden Sie bei der Bewertung von Leistungsmetriken innerhalb einer Organisation zwischen verschiedenen Arten von Metriken, die unterschiedlichen Zwecken dienen. Mit diesen Metriken werden in erster Linie die Leistung und Effizienz der technischen Infrastruktur und nicht direkt die allgemeinen Auswirkungen auf das Geschäft gemessen. Es können beispielsweise die Reaktionszeiten des Servers, die Netzwerklatenz oder die Systemverfügbarkeit verfolgt werden. Diese Metriken sind entscheidend, um zu bewerten, wie gut die Infrastruktur den technischen Betrieb der Organisation unterstützt. Sie bieten jedoch keinen direkten Einblick in umfassendere Geschäftsziele wie Kundenzufriedenheit, Umsatzwachstum oder Marktanteil. Um ein umfassendes Verständnis der Unternehmensleistung zu erhalten, ergänzen Sie diese Effizienzmetriken durch strategische Geschäftsmetriken, die direkt mit den Geschäftsergebnissen korrelieren.

Verschaffen Sie sich einen Überblick nahezu in Echtzeit über Ihre KPIs und die damit verbundenen Einsparmöglichkeiten und verfolgen Sie Ihre Fortschritte im Laufe der Zeit. Um mit der Definition und Verfolgung von KPI-Zielen zu beginnen, empfehlen wir das KPI-Dashboard von Cloud Intelligence Dashboards (CID). Basierend auf den Daten aus dem Kosten- und Nutzungsbericht (CUR) bietet das KPI-Dashboard eine Reihe von empfohlenen KPIs zur Kostenoptimierung an. Außerdem können Sie benutzerdefinierte Ziele festlegen und den Fortschritt im Laufe der Zeit verfolgen.

Wenn Sie die KPI-Ziele mit anderen Lösungen festlegen und verfolgen, achten Sie darauf, dass diese Methoden von allen Stakeholdern im Cloud-Finanzmanagement in Ihrer Organisation übernommen werden.

Implementierungsschritte

- Definieren Sie die erwarteten Nutzungsgrade: Konzentrieren Sie sich zu Beginn auf die Nutzungsgrade. Sprechen Sie mit den Anwendungsbesitzern, Marketing und größeren Geschäftsteams, um zu verstehen, wie die erwartete Nutzung für den Workload aussieht. Wie könnte sich die Kundennachfrage im Laufe der Zeit ändern und was kann sich aufgrund saisonaler Anstiege oder Marketingkampagnen ändern?
- Definieren von Ressourcen und Kosten für Workloads: Mit den definierten Nutzungsgraden quantifizieren Sie die Änderungen der Workload-Ressourcen, die erforderlich sind, um diese Nutzungsgrade zu erfüllen. Möglicherweise müssen Sie den Umfang oder die Anzahl der Ressourcen für eine Workload-Komponente und die Datenübertragung erhöhen oder Workload-Komponenten in einen anderen Service auf einer bestimmten Ebene ändern. Geben Sie die Kosten an jedem dieser Hauptpunkte an und prognostizieren Sie die Kostenänderung, wenn sich die Nutzung ändert.
- Definieren Sie Geschäftsziele: Nehmen Sie die Ergebnisse zu den erwarteten Änderungen bei Nutzung und Kosten, kombinieren Sie sie mit den erwarteten Änderungen in der Technologie oder sonstigen Programmen, die Sie ausführen, und entwickeln Sie Ziele für den Workload. Die Ziele müssen Nutzung und Kosten sowie das Verhältnis zwischen beiden berücksichtigen. Die Ziele müssen einfach und allgemein gehalten sein und den Mitarbeitern helfen zu verstehen, was das Unternehmen an Ergebnissen erwartet (z. B. sicherzustellen, dass ungenutzte Ressourcen unter einem bestimmten Kostenniveau gehalten werden). Sie müssen nicht für jeden ungenutzten Ressourcentyp Ziele definieren oder Kosten festlegen, die Verluste für Ziele und Vorgaben verursachen können. Überprüfen Sie, ob es organisatorische Programme gibt (z. B. Kompetenzaufbau wie Schulungen und Fortbildungen), wenn Kostenänderungen ohne veränderte Nutzung zu erwarten sind.

- Definieren der Ergebnisse: Geben Sie für jedes der definierten Ziele ein messbares Ergebnis an. Wenn das Ziel darin besteht, die Effizienz des Workloads zu erhöhen, sollte mit der Vorgabe der Umfang der Verbesserung (in der Regel in Form von Geschäftsergebnissen für jeden ausgegebenen Dollar) und der Zeitpunkt der Erreichung dieses Ziels angegeben werden. Sie könnten sich zum Beispiel das Ziel setzen, Verschwendung aufgrund einer Überversorgung zu minimieren. Bei diesem Ziel kann Ihre Vorgabe darin bestehen, dass die Verschwendung aufgrund einer zu hohen Rechenleistung in der ersten Stufe der Produktionsworkloads 10 Prozent der Computing-Kosten auf dieser Stufe nicht überschreiten sollte. Darüber hinaus könnte eine zweite Vorgabe darin bestehen, dass die Verschwendung aufgrund einer zu hohen Rechenleistung in der zweiten Stufe der Produktionsworkloads 5 Prozent der Rechenkosten auf dieser Stufe nicht überschreiten sollte.

Ressourcen

Zugehörige Dokumente:

- Von AWS verwaltete Richtlinien für Auftragsfunktionen
- AWS-Fakturierungsstrategie mit mehreren Konten
- Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien
- S.M.A.R.T. -Ziele
- Wie Sie mit dem CID-KPI-Dashboard Ihre KPIs zur Kostenoptimierung nachverfolgen

Zugehörige Videos:

- Well-Architected Labs: Ziele und Vorgaben (Stufe 100)

Zugehörige Beispiele:

- Was ist eine Einheitsmetrik?
- Auswahl einer Einheitsmetrik zur Unterstützung Ihres Unternehmens
- Einheitsmetriken in der Praxis – gewonnene Erkenntnisse
- Wie Einheitsmetriken dazu beitragen, Geschäftsfunktionen aufeinander abzustimmen
- Well-Architected Labs: Außerbetriebnahme von Ressourcen (Ziele und Vorgaben)
- Well-Architected Labs: Ressourcentyp, Größe und Anzahl (Ziele und Vorgaben)

COST02-BP03 Implementieren einer Kontenstruktur

Implementieren Sie eine Kontenstruktur, die für Ihre Organisation geeignet ist. Dadurch werden die Zuweisung und Verwaltung der Kosten in der gesamten Organisation erleichtert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Mit AWS Organizations können Sie mehrere AWS-Konten erstellen und so Ihre Umgebung zentral verwalten, wenn Sie Workloads in AWS skalieren. Sie können Ihre Organisationshierarchie modellieren, indem Sie AWS-Konten in einer Struktur von Organisationseinheiten (OEs) gruppieren und mehrere AWS-Konten in jeder OE erstellen. Um eine Kontostruktur zu erstellen, müssen Sie zuerst entscheiden, welches Ihrer AWS-Konten das Verwaltungskonto sein soll. Danach können Sie auf Grundlage der geplanten Kontostruktur neue AWS-Konten erstellen oder vorhandene Konten als Mitgliedskonten auswählen. Beachten Sie dabei bewährte Methoden für Verwaltungskonten und für Mitgliedskonten.

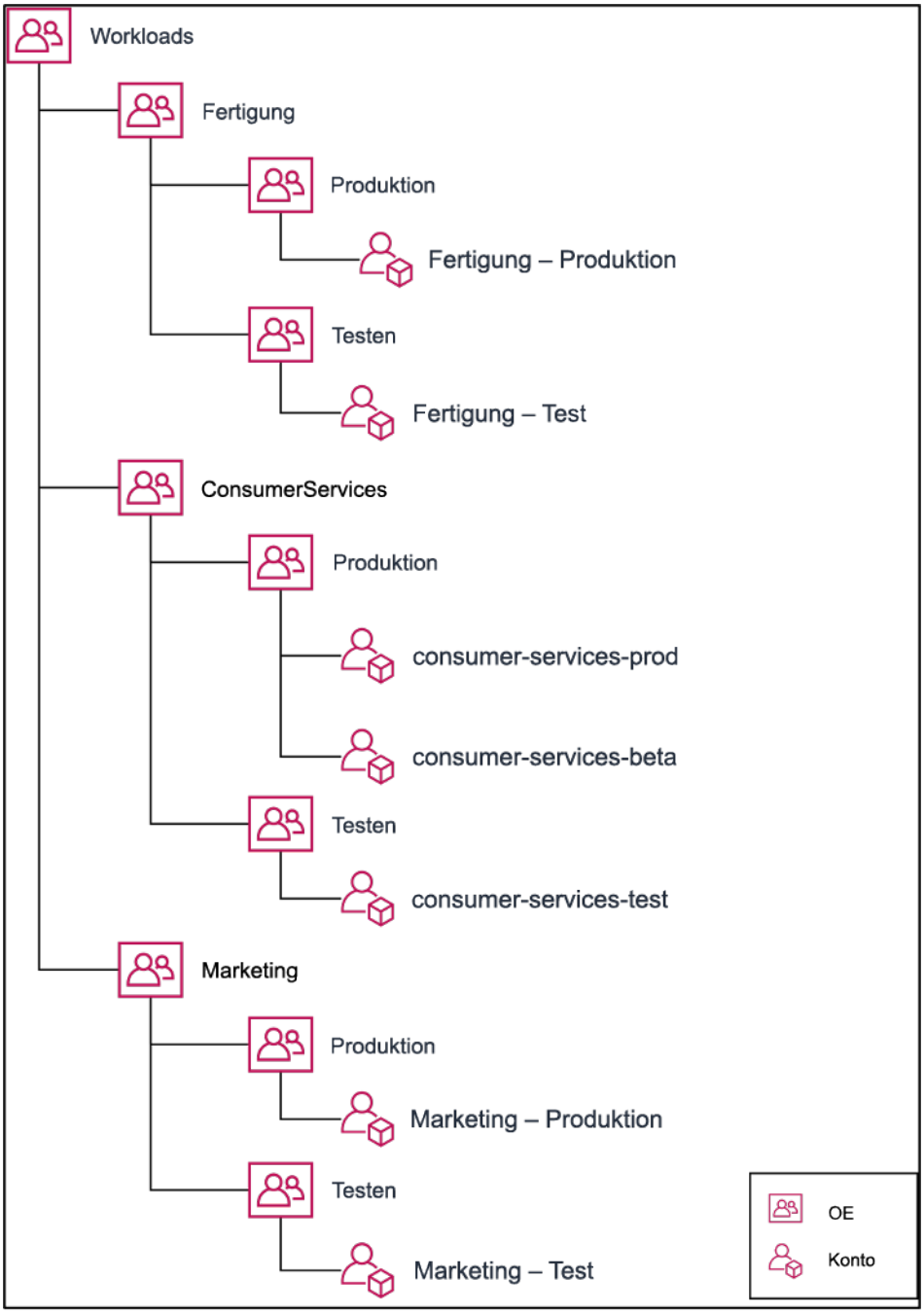
Sie sollten immer mindestens ein Verwaltungs- mit einem verknüpften Mitgliedskonto haben, unabhängig von der Unternehmensgröße oder Nutzung. Alle Workload-Ressourcen sollten sich nur in Mitgliedskonten befinden. In Verwaltungskonten sollten keine Ressourcen erstellt werden. Es gibt keine einheitliche Antwort dazu, über wie viele AWS-Konten Sie verfügen sollten. Zunächst sollten Sie Ihre aktuellen und künftigen Betriebs- und Kostenmodelle bewerten, um sicherzustellen, dass die Struktur Ihrer AWS-Konten die Ziele Ihres Unternehmens widerspiegelt. Einige Unternehmen erstellen aus geschäftlichen Gründen mehrere AWS-Konten, z. B.:

- Es ist eine administrative oder fiskale und fakturierungsbezogene Abgrenzung zwischen Organisationseinheiten, Kostenstellen oder spezifischen Workloads erforderlich.
- AWS-Service-Limits wurden für bestimmte Workloads definiert.
- Es besteht eine Anforderung für Isolierung und Trennung zwischen Workloads und Ressourcen.

Innerhalb von AWS Organizations erstellt die konsolidierte Fakturierung das Konstrukt zwischen einem oder mehreren Mitgliedskonten und dem Verwaltungskonto. Mit Mitgliedskonten können Sie Ihre Kosten und Nutzung nach Gruppen isolieren und unterscheiden. In diesem Kontext hat es sich bewährt, separate Mitgliedskonten für jede Organisationseinheit (z. B. Finanzen, Marketing und Vertrieb) oder für jeden Umgebungslebenszyklus (z. B. Entwicklung, Tests und Produktion) oder für jeden einzelnen Workload (Workload a, b und c) zu erstellen und diese verknüpften Konten dann über die konsolidierte Fakturierung zu aggregieren.

Mit der konsolidierten Fakturierung können Sie die Zahlung für mehrere AWS-Konten unter einem einzelnen Verwaltungskonto konsolidieren und dabei weiterhin die Sichtbarkeit für die Aktivitäten jedes verknüpften Kontos bereitstellen. Da Kosten und Nutzung im Verwaltungskonto aggregiert werden, können Sie sowohl Ihre Service-Volumenrabatte als auch die Nutzung Ihrer an feste Kapazität gebundene Rabatte (Savings Plans und Reserved Instances) maximieren und so die höchsten Vergünstigungen erzielen.

Im folgenden Diagramm wird gezeigt, wie Sie AWS Organizations mit Organisationseinheiten (OEs) verwenden können, um mehrere Konten zu gruppieren und mehrere AWS-Konten unter jeder OE zu platzieren. Sie sollten OEs für unterschiedliche Anwendungsfälle und Workloads verwenden, die Muster für die Organisation von Konten vorgeben.



Beispiel zum Gruppieren mehrerer AWS-Konten unter Organisationseinheiten.

AWS Control Tower kann schnell mehrere AWS-Konten einrichten und konfigurieren, um sicherzustellen, dass sowohl Governance als auch die Anforderungen Ihres Unternehmens erfüllt werden.

Implementierungsschritte

- Definieren von Trennungsanforderungen: Die Trennungsanforderungen sind eine Kombination aus mehreren Faktoren, darunter fallen Sicherheit, Zuverlässigkeit und finanzielle Konstrukte. Arbeiten Sie die einzelnen Faktoren in der richtigen Reihenfolge durch und geben Sie an, ob der Workload oder die Workload-Umgebung von anderen Workloads getrennt sein sollte. Bei der Sicherheit steht die Einhaltung der Anforderungen an Zugriff und Daten im Vordergrund. Zuverlässigkeit bezieht sich auf die Verwaltung von Limits, sodass Umgebungen und Workloads keine Auswirkungen auf andere Elemente haben. Gehen Sie die Säulen Sicherheit und Zuverlässigkeit des Well-Architected Framework regelmäßig durch und halten Sie sich an die angegebenen bewährten Methoden. Finanzielle Konstrukte schaffen eine strikte Trennung im Bereich der Finanzen (verschiedene Kostenstellen, Verantwortlichkeiten für die Workloads und Rechenschaftspflicht). Häufige Beispiele für die Trennung sind Produktions- und Test-Workloads, die in separaten Konten ausgeführt werden, oder die Verwendung eines separaten Kontos, sodass die Rechnungs- und Fakturierungsdaten den verschiedenen Unternehmenseinheiten oder Abteilungen in der Organisation oder dem Stakeholder bereitgestellt werden können, dem das Konto gehört.
- Definieren von Gruppenanforderungen: Die Anforderungen für die Gruppierung überschreiben die Trennungsanforderungen nicht, sondern unterstützen die Verwaltung. Gruppieren Sie ähnliche Umgebungen oder Workloads, die keine Trennung erfordern. Ein Beispiel hierfür ist die Gruppierung mehrerer Test- oder Entwicklungsumgebungen aus einem oder mehreren Workloads.
- Definieren der Kontenstruktur: Geben Sie mit diesen Trennungen und Gruppierungen ein Konto für jede Gruppe an und stellen Sie sicher, dass die Trennungsanforderungen erfüllt werden. Diese Konten sind Ihre Mitgliedskonten oder verknüpfte Konten. Indem Sie diese Mitgliedskonten unter einem einzigen Verwaltungs-/Zahlungskonto gruppieren, kombinieren Sie die Nutzung. Dies ermöglicht höhere Volumenrabatte für alle Konten und Sie erhalten eine gemeinsame Rechnung für alle Konten. Es ist möglich, Fakturierungsdaten zu trennen und jedem Mitgliedskonto eine individuelle Ansicht ihrer Fakturierungsdaten bereitzustellen. Definieren Sie mehrere Verwaltungs-/Zahlungskonten, wenn die Nutzungs- oder Fakturierungsdaten eines Mitgliedskontos für kein anderes Konto sichtbar sein dürfen oder wenn eine separate Rechnung von AWS erforderlich ist. In diesem Fall hat jedes Mitgliedskonto ein eigenes Verwaltungs-/Zahlungskonto. Ressourcen sollten immer in Mitgliedskonten oder verknüpften Konten platziert werden. Die Verwaltungs-/Zahlungskonten sollten nur für die Verwaltung verwendet werden.

Ressourcen

Zugehörige Dokumente:

- Verwenden von Kostenzuordnungs-Tags

- Von AWS verwaltete Richtlinien für Auftragsfunktionen
- AWS-Fakturierungsstrategie mit mehreren Konten
- Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien
- AWS Control Tower
- AWS Organizations
- Bewährte Methoden für Verwaltungskonten und Mitgliedskonten
- Organizing Your AWS Environment Using Multiple Accounts (Organisieren der AWS-Umgebung mithilfe mehrerer Konten)
- Turning on shared reserved instances and Savings Plans discounts (Aktivieren von geteilten reservierten Instances und Savings Plan-Rabatten)
- Konsolidierte Fakturierung
- Konsolidierte Fakturierung

Zugehörige Beispiele:

- Teilen des CUR und Freigabe des Zugangs

Zugehörige Videos:

- Introducing AWS Organizations (Einführung in AWS Organizations)
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations (Einrichten einer AWS-Multi-Konto-Umgebung, in der bewährte Methoden für AWS Organizations verwendet werden)

Zugehörige Beispiele:

- Well-Architected Labs: Create an AWS Organization (Level 100) (Well-Architected Labs: Erstellen einer AWS-Organisation (Stufe 100)
- Splitting the AWS Cost and Usage Report and Sharing Access (Teilen des CUR und Freigabe des Zugangs)
- Defining an AWS Multi-Account Strategy for telecommunications companies (Definieren einer AWS-Multi-Konto-Strategie für Telekommunikationsunternehmen)
- Best Practices for Optimizing AWS-Konten (Bewährte Methoden für das Optimieren von AWS-Konten)

- Bewährte Vorgehensweisen für Organisationseinheiten mit AWS Organizations

COST02-BP04 Implementieren von Gruppen und Rollen

Implementieren Sie Gruppen und Rollen, die Ihren Richtlinien entsprechen, und steuern Sie, wer Instances und Ressourcen in jeder Gruppe erstellen, ändern oder außer Betrieb nehmen kann. Implementieren Sie beispielsweise Entwicklungs-, Test- und Produktionsgruppen. Dies gilt sowohl für AWS-Services als auch für Lösungen anderer Anbieter.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Benutzerrollen und -gruppen sind grundlegende Bausteine bei der Entwicklung und Implementierung sicherer und effizienter Systeme. Rollen und Gruppen helfen Organisationen dabei, den Bedarf an Kontrolle mit den Anforderungen an Flexibilität und Produktivität in Einklang zu bringen, um letztlich die Unternehmensziele und die Bedürfnisse der Benutzer zu unterstützen. Wie im Abschnitt Identity and Access Management der Säule für die Sicherheit des AWS-Well-Architected-Framework empfohlen, benötigen Sie eine robuste Identitätsverwaltung und Berechtigungen, um den richtigen Personen unter den richtigen Bedingungen Zugriff auf die richtigen Ressourcen zu gewähren. Die Benutzer erhalten nur den Zugriff, den sie zur Erfüllung ihrer Aufgaben benötigen. Auf diese Weise wird das Risiko eines nicht autorisierten Zugriffs oder Missbrauchs minimiert.

Nachdem Sie Richtlinien entwickelt haben, können Sie logische Gruppen und Rollen von Benutzern innerhalb Ihrer Organisation erstellen. Auf diese Weise können Sie Berechtigungen zuweisen, die Nutzung kontrollieren und robuste Zugriffskontrollmechanismen implementieren, die den nicht autorisierten Zugriff auf sensible Informationen verhindern. Beginnen Sie mit allgemeinen Personengruppen. Dies entspricht in der Regel den Organisationseinheiten und beruflichen Rollen (z. B. ein Systemadministrator in der IT-Abteilung, ein Financial Controller oder ein Geschäftsanalytiker). Den Gruppen treten Personen bei, die ähnliche Aufgaben ausführen und ähnlichen Zugriff benötigen. Rollen definieren, was eine Gruppe tun muss. Es ist einfacher, Berechtigungen für Gruppen und Rollen zu verwalten als für einzelne Benutzer. Rollen und Gruppen weisen allen Benutzern konsistent und systematisch Berechtigungen zu und verhindern so Fehler und Inkonsistenzen.

Wenn sich die Rolle eines Benutzers ändert, können Administratoren den Zugriff auf Rollen- oder Gruppenebene anpassen, anstatt einzelne Benutzerkonten neu zu konfigurieren. Beispielsweise benötigt ein Systemadministrator in der IT Zugriff, um alle Ressourcen zu erstellen, aber ein Analyseteammitglied muss nur Analyseressourcen erstellen.

Implementierungsschritte

- Implementieren von Gruppen: Implementieren Sie bei Bedarf die entsprechenden Gruppen mithilfe der in Ihren Organisationsrichtlinien definierten Benutzergruppen. Bewährte Verfahren für Benutzer, Gruppen und Authentifizierung finden Sie unter der Säule für die Sicherheit des AWS-Well-Architected Framework.
- Implementieren von Rollen und Richtlinien: Erstellen Sie mithilfe der Aktionen, die in Ihren Organisationsrichtlinien definiert sind, die erforderlichen Rollen und Zugriffsrichtlinien. Bewährte Methoden zu Rollen und Richtlinien finden Sie unter der Säule für die Sicherheit des AWS-Well-Architected Framework.

Ressourcen

Zugehörige Dokumente:

- Von AWS verwaltete Richtlinien für Auftragsfunktionen
- AWS-Fakturierungsstrategie mit mehreren Konten
- Säule für Sicherheit des AWS-Well-Architected-Framework
- AWS Identity and Access Management (IAM)
- AWS Identity and Access Management-Richtlinien

Zugehörige Videos:

- Wozu dient das Identity and Access Management?

Zugehörige Beispiele:

- Well-Architected Lab Basic Identity and Access (Grundlegende Identität und Zugriff)
- Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien
- Die ersten Schritte mit Cloud Financial Management: Betriebskosten für die Cloud

COST02-BP05 Implementieren von Kostenkontrollen

Implementieren Sie Kontrollmechanismen, die auf den Organisationsrichtlinien sowie auf definierten Gruppen und Rollen basieren. Damit wird sichergestellt, dass nur Kosten im Rahmen der

festgelegten Organisationsanforderungen anfallen, z. B. durch Steuerung des Zugriffs auf Regionen oder Ressourcentypen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Ein häufiger erster Schritt bei der Implementierung von Kostenkontrollen ist die Einrichtung von Benachrichtigungen, wenn im Zusammenhang mit Kosten oder Nutzung Ereignisse auftreten, die den Richtlinien nicht entsprechen. Auf diese Weise können Sie schnell agieren und überprüfen, ob Korrekturmaßnahmen erforderlich sind, ohne dass Workloads oder neue Aktivitäten eingeschränkt oder beeinträchtigt werden. Nachdem Sie die Limits für Workloads und Umgebung kennen, können Sie die Governance erzwingen. Mit AWS Budgets lassen sich Benachrichtigungen festlegen und monatliche Budgets für AWS-Kosten, Nutzung und an feste Kapazität gebundene Rabatte definieren (Savings Plans und Reserved Instances). Sie können Budgets auf aggregierter Kostenebene (z. B. alle Kosten) oder auf einer detaillierteren Ebene erstellen, in der Sie nur bestimmte Dimensionen wie verknüpfte Konten, Services, Tags oder Availability Zones einschließen.

Wenn Sie die Budgetlimits mit AWS Budgets eingerichtet haben, können Sie mit AWS Cost Anomaly Detection unerwartete Kosten reduzieren. AWS Cost Anomaly Detection ist ein Kostenmanagementservice, der mithilfe von Machine Learning Ihre Kosten und Nutzung ständig überwacht, um ungewöhnliche Ausgaben zu erkennen. So können Sie untypische Ausgaben und ihre Ursachen schnell identifizieren und so schnell Maßnahmen ergreifen. Erstellen Sie zuerst eine Kostenüberwachung in AWS Cost Anomaly Detection und wählen Sie dann aus, wann Sie gewarnt werden möchten. Hierzu richten Sie einen Schwellenwert in Dollar ein und können sich z. B. bei Unregelmäßigkeiten benachrichtigen lassen, deren Auswirkungen 1.000 $ überschreiten. Wenn Sie Warnungen erhalten, können Sie die Ursachen hinter den Unregelmäßigkeiten und deren wirtschaftliche Auswirkungen analysieren. Sie können Unregelmäßigkeiten in AWS Cost Explorer auch selbst überwachen und analysieren.

Sie können Governance-Richtlinien in AWS durch AWS Identity and Access Management und AWS OrganizationsService-Kontrollrichtlinien (Service Control Policies, SCP) erzwingen. Mit IAM lässt sich der Zugriff auf AWS-Services und -Ressourcen sicher verwalten. Mit IAM können Sie steuern, wer AWS-Ressourcen erstellen und verwalten kann, welche Art von Ressourcen erstellt werden kann und wo sie erstellt werden können. So wird die Möglichkeit eingeschränkt, Ressourcen außerhalb der definierten Richtlinie zu erstellen. Verwenden Sie die zuvor erstellten Rollen und Gruppen und weisen Sie IAM-Richtlinien zu, um die korrekte Nutzung zu erzwingen. SCP bietet eine zentrale Kontrolle über die maximal verfügbaren Berechtigungen für alle Konten in Ihrer Organisation, damit Ihre

Konten die Vorgaben Ihrer Zugriffskontrollrichtlinien erfüllen. SCPs sind nur in einem Unternehmen verfügbar, für das alle Funktionen aktiviert sind, und Sie können die SCPs so konfigurieren, dass sie Aktionen für Mitgliedskonten standardmäßig verweigern oder zulassen. Weitere Informationen zur Implementierung des Zugriffsmanagements finden Sie im Whitepaper zur Well-Architected-Säule „Sicherheit“.

Über die Verwaltung von AWS Service Quotas können Sie ebenfalls Governance implementieren. Indem Sie sicherstellen, dass Service Quotas mit minimalem Overhead definiert und ordnungsgemäß verwaltet werden, können Sie die Ressourcenerstellung über die Geschäftsanforderungen hinaus minimieren. Dazu müssen Sie nachvollziehen, wie schnell sich Ihre Anforderungen ändern können, Sie müssen die derzeit ausgeführten Projekte kennen – in Bezug auf die Erstellung und die Deaktivierung von Ressourcen – und berücksichtigen, wie schnell Kontingentänderungen implementiert werden können. Service Quotas können bei Bedarf eingesetzt werden, um Ihre Kontingente zu erhöhen.

Implementierungsschritte

- Implementieren von Benachrichtigungen zu Ausgaben: Erstellen Sie mithilfe Ihrer definierten Organisationsrichtlinien AWS Budgets, um Benachrichtigungen zu erhalten, wenn Ausgaben außerhalb Ihrer Richtlinien liegen. Konfigurieren Sie mehrere Kostenbudgets, eines für jedes Konto, um über die allgemeinen Kontoausgaben informiert zu werden. Konfigurieren Sie zusätzliche Kostenbudgets innerhalb jedes Kontos für kleinere Einheiten innerhalb des Kontos. Diese Einheiten variieren je nach Kontenstruktur. Einige gängige Beispiele sind AWS-Regionen, Workloads (mithilfe von Tags) oder AWS-Services. Konfigurieren Sie eine E-Mail-Verteilerliste als Empfänger für Benachrichtigungen, nicht das E-Mail-Konto einer Person. Sie können ein tatsächliches Budget für den Fall konfigurieren, dass ein Betrag überschritten wird, oder ein prognostiziertes Budget zur Benachrichtigung über die prognostizierte Nutzung verwenden. Sie können auch AWS-Budgetaktionen vorkonfigurieren, die bestimmte IAM- oder SCP-Richtlinien erzwingen, oder Amazon EC2- oder Amazon RDS-Ziel-Instances beenden. Budgetaktionen werden entweder automatisch ausgeführt oder erfordern eine Workflow-Genehmigung.
- Implementieren von Benachrichtigungen zu ungewöhnlichen Ausgaben: Mit AWS Cost Anomaly Detection können Sie unerwartete Kosten in Ihrer Organisation reduzieren und die Ursachen potenzieller ungewöhnlicher Ausgaben analysieren. Wenn Sie eine Kostenüberwachung zum Identifizieren ungewöhnlicher Ausgaben mit der angegebenen Granularität erstellen und Benachrichtigungen in AWS Cost Anomaly Detection konfigurieren, erhalten Sie eine Warnung, wenn eine ungewöhnliche Ausgabe erkannt wird. So können Sie die Ursache der Unregelmäßigkeit analysieren und erhalten Informationen zu den Auswirkungen auf Ihre Kosten. Verwenden Sie AWS Cost Categories beim Konfigurieren von AWS Cost Anomaly Detection, um zu ermitteln,

welches Projekt- oder Geschäftseinheitsteam die Ursache der unerwartete Kosten analysieren und zeitnah die erforderlichen Maßnahmen ergreifen kann.

- Implementieren von Nutzungskontrollen: Implementieren Sie mithilfe Ihrer definierten Organisationsrichtlinien IAM-Richtlinien und -Rollen, um anzugeben, welche Aktionen Benutzer ausführen dürfen und welche nicht. In einer AWS-Richtlinie können mehrere Organisationsrichtlinien enthalten sein. Gehen Sie auf die gleiche Art und Weise vor, wie Sie Richtlinien definiert haben. Beginnen Sie umfassend und wenden dann bei jedem Schritt detailliertere Kontrollen an. Service Limits sind auch eine effektive Kontrolle der Nutzung. Implementieren Sie die richtigen Service Limits für alle Ihre Konten.

Ressourcen

Zugehörige Dokumente:

- Von AWS verwaltete Richtlinien für Auftragsfunktionen
- AWS-Fakturierungsstrategie mit mehreren Konten
- Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien
- AWS Budgets
- AWS Cost Anomaly Detection
- Control Your AWS Costs (Kontrollieren der AWS-Kosten)

Zugehörige Videos:

- Wie kann ich AWS Budgets verwenden, um meine Ausgaben und Nutzung zu verfolgen?

Zugehörige Beispiele:

- Example IAM access management policies (IAM-Beispielrichtlinien für die Zugriffsverwaltung)
- Beispiel-Service-Kontrollrichtlinien
- AWS Budgets Actions (AWS-Budget-Aktionen)
- Create IAM Policy to control access to Amazon EC2 resources using Tags (Erstellen von IAM-Richtlinien zum Steuern des Zugriffs auf EC2-Ressourcen mithilfe von Tags)
- Restrict the access of IAM Identity to specific Amazon EC2 resources (Einschränken des Zugriffs von IAM-Identitäten auf bestimmte EC2-Ressourcen)

- Create an IAM Policy to restrict Amazon EC2 usage by family (Erstellen einer IAM-Richtlinie zum Einschränken des EC2-Zugriffs durch Familien)
- Well-Architected Labs: Steuerung der Kosten und Nutzung (Stufe 100)
- Well-Architected Labs: Steuerung der Kosten und Nutzung (Stufe 200)
- Slack integrations for Cost Anomaly Detection using Amazon Q Developer in chat applications (Slack-Integrationen für AWS Cost Anomaly Detection mit AWS Chatbot)

COST02-BP06 Verfolgen des Projektlebenszyklus

Verfolgen, bewerten und überprüfen Sie den Lebenszyklus von Projekten, Teams und Umgebungen, damit Sie keine unnötigen Ressourcen nutzen, für die Sie zahlen müssen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Durch eine effektive Nachverfolgung des Projektlebenszyklus können Organisationen durch verbesserte Planung, Verwaltung und Ressourcenoptimierung eine bessere Kostenkontrolle erreichen. Die durch die Nachverfolgung gewonnenen Erkenntnisse sind von unschätzbarem Wert, um fundierte Entscheidungen zu treffen, die zur Kosteneffizienz und zum Gesamterfolg des Projekts beitragen.

Die Verfolgung des gesamten Lebenszyklus des Workloads hilft Ihnen zu verstehen, wann Workloads oder Workload-Komponenten nicht mehr benötigt werden. Die vorhandenen Workloads und Komponenten werden möglicherweise noch als in Gebrauch angezeigt, können jedoch bei der Veröffentlichung neuer Services oder Features durch AWS außer Betrieb genommen bzw. übernommen werden. Prüfen Sie die vorherigen Phasen der Workloads. Nachdem ein Workload in Betrieb genommen wurde, können frühere Umgebungen außer Betrieb genommen oder in ihrer Kapazität stark reduziert werden, bis sie wieder erforderlich sind.

Sie können Ressourcen mit einem Zeitrahmen oder einer Erinnerung versehen, um zu markieren, wann der Workload überprüft wurde. Wenn die Entwicklungsumgebung beispielsweise zuletzt vor Monaten überprüft wurde, könnte es ein guter Zeitpunkt sein, sie erneut zu überprüfen, um festzustellen, ob neue Dienste eingeführt werden können oder ob die Umgebung verwendet wird. Sie können Ihre Anwendungen mit myApplications auf AWS gruppieren und taggen, um Metadaten wie Kritikalität, Umgebung, letzte Überprüfung und Kostenstelle zu verwalten und zu verfolgen. Sie können sowohl den Lebenszyklus Ihres Workloads verfolgen als auch die Kosten, den Zustand, den Sicherheitsstatus und die Leistung Ihrer Anwendungen überwachen und verwalten.

AWS bietet verschiedene Management- und Governance-Services, die Sie für die Verfolgung von Entitätslebenszyklen verwenden können. Sie können AWS Config oder AWS Systems Manager verwenden, um eine detaillierte Bestandsaufnahme Ihrer AWS-Ressourcen und -Konfiguration zu erstellen. Es wird empfohlen, dass Sie diese mit Ihren vorhandenen Projekt- bzw. Komponentenverwaltungssystemen integrieren, um aktive Projekte und Produkte in Ihrer Organisation zu verfolgen. Durch die Kombination Ihres aktuellen Systems mit den umfangreichen Ereignissen und Metriken von AWS können Sie sich einen Überblick über wichtige Ereignisse im Lebenszyklus verschaffen und Ressourcen proaktiv verwalten, um unnötige Kosten zu reduzieren.

Ähnlich wie beim Application Lifecycle Management (ALM) sollte die Verfolgung des Projektlebenszyklus mehrere Prozesse, Tools und Teams umfassen, die zusammenarbeiten, z. B. Design und Entwicklung, Tests, Produktion, Support und Workload-Redundanz.

Durch die sorgfältige Überwachung jeder Phase des Lebenszyklus eines Projekts erhalten Organisationen entscheidende Einblicke und eine bessere Kontrolle, was die erfolgreiche Planung, Implementierung und den Abschluss von Projekten erleichtert. Mit dieser sorgfältigen Überwachung wird sichergestellt, dass die Projekte nicht nur den Qualitätsstandards entsprechen, sondern auch pünktlich und innerhalb des Budgets fertiggestellt werden, was die Kosteneffizienz insgesamt fördert.

Weitere Informationen zur Implementierung der Verfolgung des Lebenszyklus von Entitäten finden Sie im Whitepaper Säule „Operative Exzellenz“ – AWS-Well-Architected-Framework.

Implementierungsschritte

- Richten Sie einen Prozess zur Überwachung des Projektlebenszyklus ein: Das Cloud Center of Excellence-Team hat die Aufgabe, einen Prozess für die Überwachung des Projektlebenszyklus einzurichten. Entwickeln Sie einen strukturierten und systematischen Ansatz zur Überwachung der Workloads, um die Kontrolle, die Sichtbarkeit und die Leistung der Projekte zu verbessern. Sorgen Sie dafür, dass der Überwachungsprozess transparent und kooperativ ist und sich auf kontinuierliche Verbesserungen konzentriert, um seine Effektivität und seinen Wert zu maximieren.
- Führen Sie Workload-Überprüfungen durch: Richten Sie, wie in Ihren Organisationsrichtlinien festgelegt, einen regelmäßigen Rhythmus ein, um Ihre bestehenden Projekte zu überprüfen und Workload-Reviews durchzuführen. Der Aufwand für die Prüfung sollte proportional zum ungefähren Risiko, dem Wert oder den Kosten für die Organisation sein. Wichtige Bereiche, die in die Prüfung aufgenommen werden sollen, sind das Risiko eines Vorfalls oder eines Ausfalls, der Wert oder Beitrag für die Organisation (gemessen am Umsatz oder Ruf der Marke), die Kosten des Workloads (gemessen als Gesamtkosten für Ressourcen und Betriebskosten) und die Nutzung des Workloads (gemessen an der Anzahl der Ergebnisse der Organisation pro Zeiteinheit). Wenn sich

diese Bereiche im Laufe des Lebenszyklus ändern, sind Anpassungen des Workloads erforderlich, z. B. die vollständige oder teilweise Außerbetriebnahme.

Ressourcen

Zugehörige Dokumente:

- Leitfaden zum Tagging in AWS
- Was ist ALM (Application Lifecycle Management)?
- Von AWS verwaltete Richtlinien für Auftragsfunktionen

Zugehörige Beispiele:

- Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien

Zugehörige Tools:

- AWS Config
- AWS Systems Manager
- AWS Budgets
- AWS Organizations
- AWS CloudFormation

KOSTEN 3. Wie überwachen Sie Ihre Kosten und die Nutzung?

Definieren Sie Richtlinien und Verfahren, um Ihre Kosten überwachen und richtig zuordnen zu können. So können Sie die Kosteneffizienz eines Workloads messen und verbessern.

Bewährte Methoden

- COST03-BP01 Konfigurieren detaillierter Informationsquellen
- COST03-BP02 Hinzufügen von Unternehmensinformationen zu Kosten und Nutzung
- COST03-BP03 Identifizieren von Kostenzuordnungskategorien
- COST03-BP04 Definieren von Organisationsmetriken
- COST03-BP05 Konfigurieren von Tools für die Fakturierung und Kostenverwaltung
- COST03-BP06 Zuweisen von Kosten basierend auf Workload-Metriken

COST03-BP01 Konfigurieren detaillierter Informationsquellen

Richten Sie Kostenmanagement- und Berichtstools ein, um die Analyse und Transparenz von Kosten- und Nutzungsdaten zu verbessern. Konfigurieren Sie Ihr Workload so, dass Protokolleinträge erstellt werden, die die Nachverfolgung und Segmentierung von Kosten und Nutzung erleichtern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Detaillierte Abrechnungsinformationen, z. B. durch Aufschlüsselung nach Stunden in Kostenmanagement-Tools, ermöglichen es Unternehmen, ihre Ressourcennutzung anhand weiterer Details zu verfolgen und einige der Gründe für den Kostenanstieg zu identifizieren. Diese Datenquellen bieten die genaueste Ansicht der Kosten und Nutzung in Ihrem gesamten Unternehmen.

Sie können AWS Data Exports verwenden, um Exporte von AWS Cost and Usage Report (CUR) 2.0 zu erstellen. Dies ist die neue und empfohlene Methode, um Ihre detaillierten Kosten- und Nutzungsdaten von AWS abzurufen. Sie bietet tägliche oder stündliche Nutzungsaufschlüsselung, Tarife, Kosten und Nutzungsattribute für alle kostenpflichtigen AWS-Services (dieselben Informationen wie CUR), zusammen mit einigen Verbesserungen. Alle möglichen Dimensionen befinden sich im CUR, z. B. Tagging, Speicherort, Ressourcenattribute und Konto-IDs.

Je nach Art des Exports, den Sie erstellen möchten, gibt es drei Exporttypen: einen Standarddatenexport, einen Export in ein Kosten- und Nutzungs-Dashboard mit QuickSight-Integration und einen Legacy-Datenexport.

- Standarddatenexport: Ein benutzerdefinierter Export einer Tabelle, der regelmäßig an Amazon S3 gesendet wird.
- Kosten- und Nutzungs-Dashboard: Ein Export und eine Integration in QuickSight zur Bereitstellung eines vorgefertigten Kosten- und Nutzungs-Dashboards.
- Legacy-Datenexport: Ein Export des Legacy-AWS Cost and Usage Report (CUR).

Sie können Datenexporte mit den folgenden Anpassungen erstellen:

- Ressourcen-IDs einschließen
- Daten zur Zuordnung geteilter Kosten
- Stündliche Granularität
- Versioning

- Komprimierungstyp und Dateiformat

Aktivieren Sie für Ihre Workloads, die Container auf Amazon ECS oder Amazon EKS ausführen, Daten zur Aufteilung geteilter Kosten, sodass Sie Ihre Containerkosten einzelnen Geschäftseinheiten und Teams zuordnen können, je nachdem, wie Ihre Container-Workloads gemeinsam genutzte Computing- und Speicherressourcen verbrauchen. Mit Daten zur Aufteilung geteilter Kosten werden Kosten- und Nutzungsdaten für neue Ressourcen auf Containerebene in den AWS Cost and Usage Report aufgenommen. Die Daten zur Aufteilung geteilter Kosten werden berechnet, indem die Kosten der einzelnen ECS-Services und Aufgaben berechnet werden, die auf dem Cluster ausgeführt werden.

Ein Kosten- und Nutzungs-Dashboard exportiert die Kosten- und Nutzungs-Dashboard-Tabelle regelmäßig in einen S3-Bucket und stellt ein vorgefertigtes Kosten- und Nutzungs-Dashboard in QuickSight bereit. Verwenden Sie diese Option, wenn Sie schnell ein Dashboard mit Ihren Kosten- und Nutzungsdaten bereitstellen möchten. Mit dieser Option sind keine Anpassungen möglich.

Falls gewünscht, können Sie CUR weiterhin im Legacy-Modus exportieren, in den Sie andere Verarbeitungsservices integrieren können, z. B. AWS Glue, um die Daten für die Analyse vorzubereiten und Datenanalysen mit Amazon Athena durchzuführen, indem Sie SQL für die Datenabfrage verwenden.

Implementierungsschritte

- Datenexporte erstellen: Erstellen Sie benutzerdefinierte Exporte mit den gewünschten Daten und steuern Sie Ihr Exportschema. Erstellen Sie Fakturierungs- und Kostenmanagementdatenexporte mit einfachem SQL und zeigen Sie Ihre Abrechnungs- und Kostenmanagementdaten durch die Integration mit QuickSight an. Sie können Ihre Daten auch im Standardmodus exportieren, um Ihre Daten mit anderen Verarbeitungstools wie Amazon Athena zu analysieren.
- Konfiguration des Kosten- und Nutzungsberichts: Konfigurieren Sie über die Fakturierungskonsole mindestens einen Kosten- und Nutzungsbericht. Konfigurieren Sie einen Bericht mit stündlicher Granularität, der alle IDs und Ressourcen-IDs enthält. Sie können auch andere Berichte mit unterschiedlichen Granularitäten erstellen, um zusammenfassende Informationen bereitzustellen.
- Stündliche Granularität in Cost Explorer konfigurieren: Aktivieren Sie in der Fakturierungskonsole Daten auf Stundenbasis und auf Ressourcenebene, um auf Kosten- und Nutzungsdaten der letzten 14 Tage mit stündlicher Granularität zuzugreifen.
- Konfigurieren der Anwendungsprotokollierung: Überprüfen Sie, dass Ihre Anwendung jedes Geschäftsergebnis protokolliert, das sie liefert, sodass es nachverfolgt und gemessen werden

kann. Stellen Sie sicher, dass die Granularität dieser Daten mindestens stündlich ist, damit sie mit der Aufschlüsselung der Kosten- und Nutzungsdaten übereinstimmt. Weitere Informationen zur Protokollierung und Überwachung finden Sie unter Well-Architected: Säule „operative Exzellenz“.

Ressourcen

Zugehörige Dokumente:

- AWS Data Exports
- AWS Glue
- QuickSight
- AWS-Kostenmanagement – Preise
- Markieren von AWS-Ressourcen
- Analysieren Ihrer Kosten mit Cost Explorer
- Verwaltung von AWS Cost and Usage Report
- Well-Architected: Säule „operative Exzellenz“

Zugehörige Beispiele:

- AWS-Kontoeinrichtung
- Datenexporte für das AWS-Fakturierungs- und Kostenmanagement
- AWS Cost Explorer Häufige Anwendungsfälle

COST03-BP02 Hinzufügen von Unternehmensinformationen zu Kosten und Nutzung

Definieren Sie ein auf Ihrem Unternehmen basierendes Markierungsschema, Workload-Attribute und Kostenzuordnungskategorien, damit Sie nach Ressourcen filtern und suchen oder die Kosten und Nutzung in Kostenverwaltungstools überwachen können. Implementieren Sie ein einheitliches Markieren aller Ressourcen, wenn möglich nach Zweck, Team, Umgebung oder anderen für Ihr Unternehmen relevanten Kriterien.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Implementieren Sie das Markieren in AWS, um Unternehmensinformationen zu Ihren Ressourcen hinzuzufügen, die dann zu Ihren Kosten- und Nutzungsinformationen hinzugefügt werden. Ein

Tag (eine Markierung) ist ein Schlüssel-Wert-Paar – der Schlüssel ist definiert und muss innerhalb Ihres Unternehmens eindeutig sein und der Wert ist für eine Gruppe von Ressourcen eindeutig. Ein Beispiel für ein Schlüssel-Wert-Paar ist der Schlüssel `Umgebung` mit dem Wert `Produktion`. Alle Ressourcen in der Produktionsumgebung verfügen über dieses Schlüssel-Wert-Paar. Mit dem Markieren können Sie Ihre Kosten mit aussagekräftigen, relevanten Unternehmensinformationen kategorisieren und nachverfolgen. Sie können Tags anwenden, die Unternehmenskategorien (z. B. Kostenstellen, Anwendungsnamen, Projekte oder Besitzer) darstellen und Workloads und Merkmale von Workloads (z. B. Test oder Produktion) identifizieren, um Ihre Kosten und Nutzung in Ihrem gesamten Unternehmen zuzuordnen.

Wenn Sie Tags auf Ihre AWS-Ressourcen anwenden (z. B. Amazon Elastic Compute Cloud-Instances oder Amazon Simple Storage Service-Buckets) und die Tags aktivieren, fügt AWS diese Informationen zu Ihren Kosten- und Nutzungsberichten hinzu. Sie können Berichte ausführen und Analysen für markierte und nicht markierte Ressourcen durchführen, um eine größere Compliance mit internen Kostenverwaltungsrichtlinien zu ermöglichen und eine genaue Zuordnung zu gewährleisten.

Mit der Erstellung und Implementierung eines AWS-Markierungsstandards für alle Konten in Ihrem Unternehmen können Sie Ihre AWS-Umgebungen auf konsistente und einheitliche Weise verwalten und steuern. Verwenden Sie Tag-Richtlinien in AWS Organizations, um Regeln für die Verwendung von Tags für AWS-Ressourcen in Ihren Konten in AWS Organizations zu definieren. Mit Tag-Richtlinien können Sie problemlos einen standardisierten Ansatz für das Taggen von AWS-Ressourcen anwenden.

Mit dem AWS Tag Editor können Sie Tags für mehrere Ressourcen hinzufügen, löschen und verwalten. Mit Tag Editor suchen Sie nach den Ressourcen, die Sie taggen möchten, und verwalten dann die Tags für die Ressourcen in Ihren Suchergebnissen.

Mit AWSCost Categories können Sie Ihren Kosten eine Unternehmensbedeutung zuweisen, ohne dass Tags für Ressourcen erforderlich sind. Sie können Ihre Kosten- und Nutzungsinformationen eindeutigen internen Unternehmensstrukturen zuordnen. Sie definieren Kategorieregeln, um Kosten mithilfe von Fakturierungsdimensionen wie Konten und Tags zuzuordnen und zu kategorisieren. Dies bietet zusätzlich zum Tagging eine weitere Ebene der Verwaltungsfunktionen. Sie können auch bestimmte Konten und Tags mehreren Projekten zuordnen.

Implementierungsschritte

- Definieren eines Markierungsschemas: Versammeln Sie alle Beteiligten aus Ihrem gesamten Unternehmen, um ein Schema zu definieren. Dies umfasst in der Regel Mitarbeiter in technischen,

finanziellen und leitenden Funktionen. Definieren Sie eine Liste der Tags, die alle Ressourcen haben müssen, sowie eine Liste der Tags, die Ressourcen haben sollten. Stellen Sie sicher, dass die Tag-Namen und -Werte in Ihrer Organisation konsistent sind.

- Tag-Ressourcen: Platzieren Sie mithilfe Ihrer definierten Kostenzuordnungskategorien Tags für alle Ressourcen in Ihren Workloads entsprechend den Kategorien. Verwenden Sie Tools wie CLI, Tag Editor oder AWS Systems Manager, um die Effizienz zu steigern.
- Implementieren von AWS Cost Categories: Sie können Kostenkategorien erstellen, ohne das Markieren zu implementieren. Kostenkategorien verwenden die vorhandenen Kosten- und Nutzungsdimensionen. Erstellen Sie Kategorieregeln aus Ihrem Schema und implementieren Sie diese in Kostenkategorien.
- Automatisiertes Markieren: Automatisieren Sie das Markieren, um sicherzustellen, dass Sie ein hohes Maß an Markierungen für alle Ressourcen aufrechterhalten, damit Ressourcen automatisch bei ihrer Erstellung markiert werden. Nutzen Sie Services wie AWS CloudFormation, um zu überprüfen, ob die Ressourcen bei der Erstellung mit Markierungen versehen wurden. Sie können auch eine benutzerdefinierte Lösung für das automatische Markieren mithilfe von Lambda-Funktionen erstellen oder einen Microservice verwenden, der den Workload regelmäßig überprüft und alle nicht markierten Ressourcen entfernt, was ideal für Test- und Entwicklungsumgebungen ist.
- Überwachung von und Berichterstellung zu Tags: Um sicherzustellen, dass Sie in Ihrer Organisation ein hohes Maß an Markierungen aufrechterhalten, melden und überwachen Sie die Tags in Ihren Workloads. Sie können AWS Cost Explorer verwenden, um die Kosten für markierte und nicht markierte Ressourcen anzuzeigen. Alternativ können Sie auch Services wie Tag Editor verwenden. Überprüfen Sie regelmäßig die Anzahl der nicht markierten Ressourcen und ergreifen Sie Maßnahmen, um Tags hinzuzufügen, bis Sie die gewünschte Markierungsstufe erreichen.

Ressourcen

Zugehörige Dokumente:

- Bewährte Methoden für Tags
- AWS CloudFormation-Ressourcen-Tag
- AWS Cost Categories
- Markieren von AWS-Ressourcen
- Analysieren Ihrer Kosten mit AWS Budgets
- Analysieren Ihrer Kosten mit Cost Explorer

- Verwalten von AWS-Kosten- und -Nutzungsberichten

Zugehörige Videos:

- Wie kann ich meine AWS-Ressourcen markieren, um meine Rechnung nach Kostenstelle oder Projekt aufzuteilen?
- Markieren von AWS-Ressourcen

Zugehörige Beispiele:

- Automatisches Markieren von neuen AWS-Ressourcen basierend auf der Identität oder Position

COST03-BP03 Identifizieren von Kostenzuordnungskategorien

Identifizieren Sie Organisationskategorien wie Geschäftsbereiche, Abteilungen oder Projekte, anhand derer die Kosten innerhalb Ihres Unternehmens den internen Verbrauchern zugewiesen werden können. Verwenden Sie diese Kategorien, um ein Gefühl der Verantwortung für Ausgaben zu fördern, Bewusstsein für Kosten zu schaffen und ein effektives Nutzungsverhalten zu unterstützen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Der Prozess der Kostenkategorisierung ist für Budgetierung, Buchhaltung, Finanzberichterstattung, Entscheidungsfindung, Benchmarking und Projektmanagement von entscheidender Bedeutung. Durch die Klassifizierung und Kategorisierung von Ausgaben können Teams die Arten von Kosten besser nachvollziehen, die auf dem Weg in die Cloud entstehen werden. So können sie fundierte Entscheidungen treffen und Budgets effektiv verwalten.

Die Rechenschaftspflicht bei den Cloud-Ausgaben ist ein starker Anreiz für ein diszipliniertes Nachfrage- und Kostenmanagement. Das Ergebnis sind deutlich höhere Cloud-Kosteneinsparungen für Unternehmen, die den größten Teil ihrer Cloud-Ausgaben für verbrauchende Geschäftsbereiche oder Teams aufwenden. Darüber hinaus hilft die Zuweisung von Cloud-Ausgaben Unternehmen dabei, mehr bewährte Methoden für die zentralisierte Cloud-Governance einzuführen.

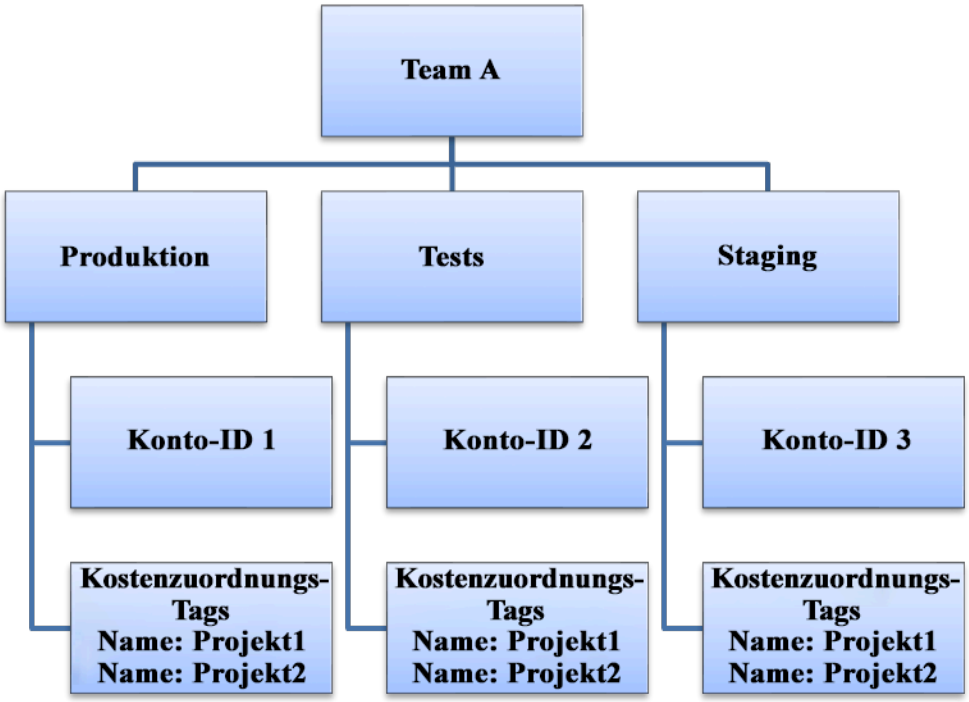
Arbeiten Sie in regelmäßigen Besprechungen mit Ihrem Finanzteam und anderen relevanten Stakeholdern zusammen, um zu verstehen, wie die Kosten innerhalb Ihres Unternehmens zugeordnet werden müssen. Workload-Kosten müssen über den gesamten Lebenszyklus hinweg

zugeordnet werden, einschließlich Entwicklung, Tests, Produktion und Außerbetriebnahme. Analysieren Sie, welche Kosten durch Schulungen, Personalentwicklung und Ideenentwicklung im Unternehmen entstehen. Dies kann hilfreich sein, um Konten, die zu diesem Zweck verwendet werden, korrekt den Schulungs- und Entwicklungsbudgets zuzuordnen, anstatt allgemeinen IT-Kostenbudgets.

Nachdem Sie Ihre Kostenzuordnungskategorien mit Stakeholdern in Ihrer Organisation definiert haben, können Sie mit AWS Cost Categories Ihre Kosten- und Nutzungsinformationen in aussagekräftige Kategorien in der AWS Cloud gruppieren, z. B. Kosten für ein bestimmtes Projekt oder AWS-Konten für Abteilungen oder Geschäftsbereiche. Sie können benutzerdefinierte Kategorien erstellen und Ihre Kosten- und Nutzungsinformationen diesen Kategorien zuordnen, und zwar basierend auf Regeln, die Sie anhand verschiedener Dimensionen wie Konto, Tag, Service, oder Kostenart definieren. Sobald die Kostenkategorien eingerichtet sind, können Sie Ihre Kosten- und Nutzungsinformationen nach diesen Kategorien aufgeschlüsselt anzeigen, sodass Ihr Unternehmen bessere Strategie- und Kaufentscheidungen treffen kann. Diese Kategorien sind auch in AWS Cost Explorer, AWS Budgets und AWS Cost and Usage Report sichtbar.

Erstellen Sie beispielsweise Kostenkategorien für Ihre Geschäftseinheiten (DevOps-Team) und erstellen Sie unter jeder Kategorie mehrere Regeln (Regeln für jede Unterkategorie) mit mehreren Dimensionen (AWS-Konten, Kostenzuordnungs-Tags, Services oder Kostenart) basierend auf den von Ihnen definierten Gruppierungen. Mit den Kostenkategorien können Sie Ihre Kosten mithilfe einer regelbasierten Engine organisieren. Die von Ihnen konfigurierten Regeln organisieren Ihre Kosten in Kategorien. Innerhalb dieser Regeln können Sie mithilfe mehrerer Dimensionen für jede Kategorie filtern, z. B. nach bestimmten AWS-Konten, bestimmten AWS-Services oder bestimmten Kostenarten. Sie können diese Kategorien dann für mehrere Produkte in der AWS Fakturierung und Kostenmanagement-Kostenmanagement Konsole verwenden. Dazu gehören AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report und AWS Cost Anomaly Detection.

Das folgende Diagramm zeigt Ihnen beispielsweise, wie Ihre Kosten- und Nutzungsinformationen in Ihrem Unternehmen gruppiert werden können, z. B. mit mehreren Teams (Kostenkategorie) und mehreren Umgebungen (Regeln), wobei jede Umgebung mehrere Ressourcen oder Assets (Dimensionen) aufweist.



Organigramm für Kosten und Nutzung

Sie können mithilfe von Kostenkategorien auch Kostengruppierungen erstellen. Nachdem Sie die Kostenkategorien erstellt haben (es kann nach dem Erstellen einer Kostenkategorie bis zu 24 Stunden dauern, bis die Werte in Ihren Nutzungsdatensätzen aktualisiert sind), erscheinen sie in AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Reportund AWS Cost Anomaly Detection. In AWS Cost Explorer und AWS Budgets erscheint eine Kostenkategorie als zusätzliche Fakturierungsdimension. Damit können Sie nach einem bestimmten Kostenkategoriewert filtern oder nach der Kostenkategorie gruppieren.

Implementierungsschritte

- Definieren der Organisationskategorien: Treffen Sie sich mit internen Stakeholdern und Vertretern aus verschiedenen Unternehmensbereichen, um Kategorien zu definieren, die die Struktur und Anforderungen Ihres Unternehmens widerspiegeln. Diese werden direkt der Struktur vorhandener Finanzkategorien zugeordnet, z. B. Geschäftsbereich, Budget, Kostenstelle oder Abteilung. Sehen Sie sich die Ergebnisse an, die die Cloud für Ihr Unternehmen liefert, z. B. Schulungen oder Fortbildungen, da es sich auch um Organisationskategorien handelt.
- Definieren der funktionalen Kategorien: Treffen Sie sich mit internen Stakeholdern und Vertretern aus verschiedenen Unternehmensbereichen, um Kategorien zu definieren, die die

Funktionen in Ihrem Unternehmen widerspiegeln. Dabei kann es sich um den Workload- oder Anwendungsnamen und die Art der Umgebung handeln, z. B. Produktion, Test oder Entwicklung.

- Definieren von AWS Cost Categories: Erstellen Sie Kostenkategorien, um Ihre Kosten- und Nutzungsinformationen zu organisieren, indem Sie AWS Cost Categories verwenden und Sie Ihre AWS-Kosten und -Nutzung aussagekräftigen Kategorien zuordnen. Einer Ressource können mehrere Kategorien zugewiesen werden. Eine Ressource kann sich in mehreren verschiedenen Kategorien befinden. Definieren Sie daher so viele Kategorien wie nötig, um Ihre Kosten innerhalb der kategorisierten Struktur mithilfe von AWS Cost Categories zu verwalten.

Ressourcen

Zugehörige Dokumente:

- Markieren von AWS-Ressourcen
- Verwenden von Kostenzuordnungs-Tags
- Analysieren Ihrer Kosten mit AWS Budgets
- Analysieren Ihrer Kosten mit Cost Explorer
- Verwalten von AWS Cost and Usage Reports
- AWS Cost Categories
- Verwalten Ihrer Kosten mit AWS Cost Categories
- Erstellen von Kostenkategorien
- Markieren von Kostenkategorien
- Aufteilen von Kosten innerhalb von Kostenkategorien
- Funktionen in AWS Cost Categories

Zugehörige Beispiele:

- Organisieren von Kosten- und Nutzungsdaten mit AWS Cost Categories
- Verwalten Ihrer Kosten mit AWS Cost Categories
- Well-Architected Labs: Visualisierung der Kosten und Nutzung
- Well-Architected Labs: Cost Categories

COST03-BP04 Definieren von Organisationsmetriken

Definieren Sie die Organisationsmetriken, die für diesen Workload erforderlich sind. Beispiele für Metriken eines Workloads sind erstellte Kundenberichte oder Webseiten, die den Kunden angezeigt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Entwickeln Sie ein Verständnis dafür, wie die Ausgabe Ihres Workloads im Vergleich zum Geschäftserfolg gemessen wird. Jeder Workload verfügt in der Regel über einen kleinen Satz von Hauptausgaben, die auf die Leistung hinweisen. Wenn Sie einen komplexen Workload mit vielen Komponenten haben, können Sie die Liste priorisieren oder Metriken für jede Komponente definieren und nachverfolgen. Arbeiten Sie mit Ihren Teams zusammen, um zu verstehen, welche Metriken verwendet werden sollen. Diese Einheit wird verwendet, um die Effizienz des Workloads oder die Kosten für die einzelnen Geschäftsausgaben zu verstehen.

Implementierungsschritte

- Definieren von Workload-Ergebnissen: Treffen Sie sich mit den Beteiligten im Unternehmen und definieren Sie die Ergebnisse für den Workload. Hierbei handelt es sich um eine primäre Maßnahme für die Kundennutzung. Es müssen Geschäftsmetriken und keine technischen Metriken gemessen werden. Es sollte eine kleine Anzahl von High-Level-Metriken (weniger als fünf) pro Workload geben. Wenn der Workload mehrere Ergebnisse für verschiedene Anwendungsfälle erzeugt, gruppieren Sie sie in einer einzigen Metrik.
- Definieren der Ergebnisse von Workload-Komponenten: Wenn Sie einen großen und komplexen Workload haben oder Ihren Workload problemlos in Komponenten (z. B. Microservices) mit gut definierten Ein- und Ausgaben aufteilen können, definieren Sie optional Metriken für jede Komponente. Der Aufwand sollte den Wert und die Kosten der Komponente widerspiegeln. Beginnen Sie mit den größten Komponenten und arbeiten Sie sich zu den kleineren Komponenten vor.

Ressourcen

Zugehörige Dokumente:

- Markieren von AWS-Ressourcen
- Analysieren Ihrer Kosten mit AWS Budgets

- Analysieren Ihrer Kosten mit Cost Explorer
- Verwalten von AWS-Kosten- und -Nutzungsberichten

COST03-BP05 Konfigurieren von Tools für die Fakturierung und Kostenverwaltung

Konfigurieren Sie Kostenverwaltungstools in Übereinstimmung mit den Richtlinien Ihrer Organisation, um die Cloud-Ausgaben zu verwalten und zu optimieren. Dazu gehören Services, Tools und Ressourcen zur Organisation und Nachverfolgung von Kosten- und Nutzungsdaten, zur Verbesserung der Kontrolle durch konsolidierte Fakturierung und Zugriffsberechtigungen, zur Verbesserung der Planung durch Budgetierung und Prognosen, zum Erhalt von Benachrichtigungen oder Warnmeldungen und zur Kostensenkung durch Ressourcen- und Preisoptimierungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Um eine starke Rechenschaftspflicht zu gewährleisten, erörtern Sie im Rahmen Ihrer Kostenzuordnungsstrategie zunächst Ihre Kontostrategie. Wenn Sie das richtig machen, reicht das möglicherweise schon aus. Andernfalls fehlen wichtige Informationen und es kann zu weiteren Problemen kommen.

Um die Rechenschaftspflicht für Cloud-Ausgaben zu fördern, gewähren Sie Benutzern Zugriff auf Tools, die einen Überblick über ihre Kosten und Nutzung bieten. AWS empfiehlt, dass Sie alle Workloads und Teams für die folgenden Zwecke konfigurieren:

- Organisation: Legen Sie Ihre Basis für die Kostenzuordnung und Governance mit Ihrer eigenen Tagging-Strategie und Kategorisierung fest. Erstellen Sie mehrere AWS-Konten mit Tools wie AWS Control Tower oder AWS Organization. Taggen Sie die unterstützten AWS-Ressourcen und kategorisieren Sie sie anhand Ihrer Organisationsstruktur (Geschäftseinheiten, Abteilungen oder Projekte) aussagekräftig. Taggen Sie Kontonamen für bestimmte Kostenstellen und ordnen Sie sie AWS Kostenkategorien zu, um Konten für Geschäftseinheiten für ihre Kostenstellen zu gruppieren, sodass der Eigentümer der Geschäftseinheit den Verbrauch mehrerer Konten an einem Ort sehen kann.
- Zugriff: Verfolgen Sie organisationsweite Fakturierungsinformationen durch konsolidierte Abrechnungen. Stellen Sie sicher, dass die richtigen Stakeholder und Geschäftsinhaber Zugriff darauf haben.
- Kontrolle: Entwickeln Sie effektive Governance-Mechanismen mit dem richtigen Integritätsschutz, um unerwartete Szenarien bei der Verwendung von Service-Kontrollrichtlinien (Service Control

Policies, SCP), Tag-Richtlinien, IAM-Richtlinien und Budgetwarnmeldungen zu verhindern. Beispielsweise können Sie Teams erlauben, bestimmte Ressourcen nur in bevorzugten Regionen zu erstellen, indem Sie effektive Kontrollmechanismen verwenden und verhindern, dass Ressourcen ohne bestimmte Tags (z. B. Kostenstelle) erstellt werden.

- Aktueller Status: Konfigurieren Sie ein Dashboard mit aktuellen Kosten- und Nutzungsraten. Das Dashboard sollte an einem gut sichtbaren Ort innerhalb der Arbeitsumgebung verfügbar sein (wie bei einem Betriebs-Dashboard). Sie können Daten exportieren und für eine gute Sichtbarkeit das Kosten- und Nutzungs-Dashboard aus dem AWS Cost Optimization Hub oder einem beliebigen unterstützten Produkt verwenden. Möglicherweise müssen Sie verschiedene Dashboards für verschiedene Personengruppen erstellen. Beispielsweise kann sich das Manager-Dashboard von einem Engineering-Dashboard unterscheiden.
- Benachrichtigungen: Stellen Sie mit AWS Budgets oder AWS Cost Anomaly Detection Benachrichtigungen bereit, wenn Kosten oder Nutzungen definierte Grenzwerte überschreiten und Anomalien auftreten.
- Berichte: Fassen Sie alle Kosten- und Nutzungsinformationen zusammen. Erhöhen Sie das Bewusstsein und die Verantwortlichkeit für Ihre Cloud-Ausgaben mit detaillierten, zuordnungsfähigen Kostendaten. Erstellen Sie Berichte, die für das Team, das sie bearbeitet, relevant sind und Empfehlungen enthalten.
- Nachverfolgung: Zeigen Sie die aktuellen Kosten und die aktuelle Nutzung in Bezug zu konfigurierten Zielen oder Vorgaben an.
- Analyse: Ermöglichen Sie Teammitgliedern die Durchführung benutzerdefinierter und detaillierter Analysen mit stündlicher, täglicher oder monatlicher Granularität und verschiedenen Filtern (Ressource, Konto, Tag usw.).
- Prüfung: Bleiben Sie hinsichtlich Ihrer Ressourcenbereitstellung und Ihrer Möglichkeiten zur Kostenoptimierung auf dem Laufenden. Erhalten Sie Benachrichtigungen mithilfe von Amazon CloudWatch, Amazon SNS oder Amazon SES für Ressourcenbereitstellungen auf Organisationsebene. Überprüfen Sie die Empfehlungen zur Kostenoptimierung mit AWS Trusted Advisor oder AWS Compute Optimizer.
- Trendberichte: Zeigen Sie die Variabilität von Kosten und Nutzung über den erforderlichen Zeitraum mit der erforderlichen Aufschlüsselung an.
- Prognosen: Zeigen Sie geschätzte zukünftige Kosten und schätzen Sie Ihre Ressourcennutzung und Ihre Ausgaben mit von Ihnen erstellten Prognose-Dashboards.

Sie können den AWS Cost Optimization Hub verwenden, um potenzielle Kostenoptimierungsmöglichkeiten zu ermitteln, die von einem zentralen Standort aus konsolidiert

werden, und Datenexporte für die Integration mit Amazon Athena erstellen. Mit dem AWS Cost Optimization Hub können Sie außerdem das Kosten- und Nutzungs-Dashboard bereitstellen, das QuickSight für interaktive Kostenanalysen und den sicheren Austausch von Kosteninformationen verwendet.

Wenn Sie in Ihrer Organisation nicht über die notwendigen Kenntnisse oder die erforderliche Bandbreite verfügen, können Sie mit AWS ProServ, AWS Managed Services (AMS) oder AWS-Partnern arbeiten. Sie können auch Tools von Drittanbietern verwenden. Stellen Sie jedoch sicher, dass Sie das Wertversprechen validieren.

Implementierungsschritte

- Teambasierten Zugriff auf Tools ermöglichen: Konfigurieren Sie Ihre Konten und erstellen Sie Gruppen, die Zugriff auf die erforderlichen Kosten- und Nutzungsberichte für ihre Verbräuche haben, und verwenden Sie AWS Identity and Access Management, um den Zugriff auf die Tools wie AWS Cost Explorer zu kontrollieren. Diese Gruppen müssen Vertreter aller Teams umfassen, die für eine Anwendung zuständig sind oder diese verwalten. Auf diese Weise wird sichergestellt, dass jedes Team Zugriff auf seine Kosten- und Nutzungsinformationen hat, um seinen Verbrauch nachzuverfolgen.
- Organisieren Sie Kosten-Tags und -Kategorien: Organisieren Sie Ihre Kosten nach Teams, Geschäftseinheiten, Anwendungen, Umgebungen und Projekten. Verwenden Sie Ressourcen-Tags, um Kosten nach Kostenzuordnungs-Tags zu organisieren. Erstellen Sie Kostenkategorien auf der Grundlage der Dimensionen, indem Sie anhand von Tags, Konten, Diensten usw. Ihre Kosten abbilden.
- Konfigurieren Sie AWS-Budgets: Konfigurieren Sie AWS-Budgets auf allen Konten für Ihre Workloads. Legen Sie mithilfe von Tags und Kostenkategorien Budgets für die Gesamtkontoausgaben und Budgets für die Workloads fest. Konfigurieren Sie Benachrichtigungen in AWS-Budgets, um Warnmeldungen zu erhalten, wenn Sie Ihre budgetierten Beträge überschreiten, oder wenn Ihre geschätzten Kosten Ihre Budgets übersteigen.
- Konfigurieren Sie die AWS Cost Anomaly Detection: Verwenden Sie die AWS Cost Anomaly Detection für Ihre Konten, Kernservices oder von Ihnen erstellte Kostenkategorien, um Ihre Kosten und Nutzung zu überwachen und ungewöhnliche Ausgaben zu erkennen. Sie können Warnmeldungen einzeln in aggregierten Berichten, in einer E-Mail oder einem Amazon SNS-Thema erhalten. Dies ermöglicht es Ihnen, die Ursache der Anomalie zu analysieren und zu bestimmen und den Faktor zu identifizieren, der die Kostensteigerung verursacht.
- Verwenden Sie Kostenanalysetools: Konfigurieren Sie AWS Cost Explorer für Ihren Workload und Ihre Konten, um Ihre Kostendaten für die weitere Analyse zu visualisieren. Erstellen Sie ein

Dashboard für den Workload, das die Gesamtausgaben, die wichtigsten Nutzungskennzahlen für den Workload und die Prognose künftiger Kosten auf der Grundlage Ihrer historischen Kostendaten nachverfolgt.

- Verwenden Sie Analysetools zur Kostenoptimierung: Verwenden Sie den AWS Cost Optimization Hub, um Einsparmöglichkeiten mit maßgeschneiderten Empfehlungen zu identifizieren, darunter das Löschen ungenutzter Ressourcen, die richtige Dimensionierung, Savings Plans, Reservierungen und Empfehlungen von Computing-Optimierern.
- Konfigurieren Sie fortschrittliche Tools: Sie können optional Grafiken erstellen, um die interaktive Analyse und den Austausch von Kosteninformationen zu erleichtern. Mit Datenexporten auf dem AWS Cost Optimization Hub können Sie ein Kosten- und Nutzungs-Dashboard erstellen, das von QuickSight für Ihre Organisation bereitgestellt wird und zusätzliche Details und Granularität bietet. Sie können auch fortschrittliche Analysefunktionen implementieren, indem Sie Datenexporte in Amazon Athena für erweiterte Abfragen verwenden, und Dashboards auf QuickSight erstellen. Arbeiten Sie mit AWS-Partnern zusammen, um Cloud-Management-Lösungen für die konsolidierte Überwachung und Optimierung von Cloud-Rechnungen einzuführen.

Ressourcen

Zugehörige Dokumente:

- Was ist AWS Fakturierung und Kostenmanagement und Kostenmanagement?
- Einrichten Ihrer AWS-Umgebung mit bewährten Methoden
- Bewährte Methoden für das Tagging von AWS-Ressourcen
- Tagging Ihrer AWS-Ressourcen
- AWS-Kostenkategorien
- Analysieren Ihrer Kosten mit AWS Budgets
- Analysieren Ihrer Kosten mit AWS Cost Explorer
- Was sind AWS-Datenexporte?

Zugehörige Videos:

- Bereitstellen von Cloud Intelligence Dashboards
- Erhalten von Warnmeldungen zu jeder FinOps- oder Kostenoptimierungsmetrik oder KPI

Zugehörige Beispiele:

- Kosten- und Nutzungs-Dashboard unterstützt von QuickSight
- Workshop zur AWS-Steuerung der Kosten und Nutzen

COST03-BP06 Zuweisen von Kosten basierend auf Workload-Metriken

Ordnen Sie die Kosten des betreffenden Workloads anhand von Nutzungsmetriken oder Geschäftsergebnissen zu, um die Kosteneffizienz des Workloads zu bewerten. Implementieren Sie einen Prozess zur Analyse der Kosten- und Nutzungsdaten mithilfe von Analysediensten, um von genaueren Einblicken und Rückbelastungsmöglichkeiten zu profitieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Die Kostenoptimierung ermöglicht Geschäftsergebnisse zum niedrigsten Preis, was nur durch Zuweisung von Workload-Kosten nach Workload-Metriken (gemessen nach Workload-Effizienz) erreicht werden kann. Überwachen Sie die definierten Workload-Metriken durch Protokolldateien oder andere Anwendungsüberwachung. Kombinieren Sie diese Daten mit den Workload-Kosten, die Sie erhalten können, indem Sie Kosten mit einem bestimmten Tag-Wert oder einer Konto-ID betrachten. Führen Sie diese Analyse stündlich durch. Ihre Effizienz ändert sich in der Regel, wenn Sie statische Kostenkomponenten haben (z. B. eine Backend-Datenbank, die dauerhaft ausgeführt wird) mit einer variierenden Anfragerate (z. B. Nutzungsspitzen von 9 bis 17 Uhr, mit wenigen Anfragen in der Nacht). Wenn Sie die Beziehung zwischen den statischen und variablen Kosten verstehen, können Sie Ihre Optimierungsaktivitäten besser fokussieren.

Das Erstellen von Workload-Metriken für gemeinsam genutzte Ressourcen kann im Vergleich zu Ressourcen wie containerisierten Anwendungen auf Amazon Elastic Container Service (Amazon ECS) und Amazon API Gateway eine Herausforderung sein. Es gibt jedoch bestimmte Möglichkeiten, die Nutzung zu kategorisieren und die Kosten zu verfolgen. Wenn Sie gemeinsam genutzte Ressourcen von Amazon ECS und AWS Batch verfolgen müssen, können Sie die Zuordnung geteilter Kosten in AWS Cost Explorer aktivieren. Mithilfe von Daten zur Aufteilung der Kosten können Sie die Kosten und die Nutzung Ihrer containerisierten Anwendungen nachvollziehen und optimieren und die Anwendungskosten auf Grundlage des Verbrauchs der gemeinsam genutzten Rechen- und Speicherressourcen einzelnen Geschäftsbereichen zuweisen.

Implementierungsschritte

- Kosten Workload-Metriken zuordnen: Erstellen Sie mit den definierten Metriken und konfigurierten Markierungen eine Metrik, die die Workload-Ausgabe und die Workload-Kosten kombiniert.

Verwenden Sie Analyse-Services wie Amazon Athena und Amazon QuickSight, um ein Effizienz-Dashboard für den gesamten Workload und alle Komponenten zu erstellen.

Ressourcen

Zugehörige Dokumente:

- Markieren von AWS-Ressourcen
- Analysieren Ihrer Kosten mit AWS Budgets
- Analysieren Ihrer Kosten mit Cost Explorer
- Verwalten von AWS-Kosten- und -Nutzungsberichten

Zugehörige Beispiele:

- Verbesserte Kostentransparenz von Amazon ECS und AWS Batch mit AWS-Daten zur Zuordnung geteilter Kosten

KOSTEN 4. Wie können Sie Ressourcen außer Betrieb nehmen?

Implementieren Sie vom Beginn bis zum Abschluss eines Projekts eine Änderungskontrolle und Ressourcenverwaltung. So stellen Sie sicher, dass Sie ungenutzte Ressourcen abschalten oder beenden, um Verschwendung zu vermeiden.

Bewährte Methoden

- COST04-BP01 Nachverfolgen von Ressourcen über ihre Lebensdauer
- COST04-BP02 Implementieren eines Prozesses für die Außerbetriebnahme
- COST04-BP03 Außerbetriebnahme von Ressourcen
- COST04-BP04 Automatische Stilllegung von Ressourcen
- COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung

COST04-BP01 Nachverfolgen von Ressourcen über ihre Lebensdauer

Definieren und implementieren Sie eine Methode zur Verfolgung von Ressourcen und deren Verknüpfungen mit Systemen über ihre gesamte Lebensdauer hinweg. Mit einer entsprechenden Markierung können Sie den Workload oder die Funktion der Ressource identifizieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Nicht mehr benötigte Workload-Ressourcen werden außer Betrieb genommen. Ein gängiges Beispiel sind Ressourcen, die zum Testen verwendet werden. Nach Abschluss des Tests können die Ressourcen entfernt werden. Das Nachverfolgen von Ressourcen mit Tags (und Ausführen von Berichten zu diesen Tags) kann Ihnen helfen, Komponenten zu identifizieren, die außer Betrieb genommen werden können, weil sie nicht genutzt werden oder ihre Lizenz abläuft. Die Verwendung von Tags ist eine effektive Möglichkeit, Ressourcen zu verfolgen, indem die Ressource mit ihrer Funktion oder einem bekannten Datum, an dem sie außer Betrieb genommen werden kann, gekennzeichnet wird. Berichte können dann zu diesen Tags ausgeführt werden. Ein Beispielwert für das Markieren von Funktionen ist `Feature-X-Test`, um den Zweck der Ressource in Bezug auf den Workload-Lebenszyklus anzugeben. Eine andere Möglichkeit ist die Verwendung von `LifeSpan` oder `TTL` für die Ressourcen, z. B. ein Tag-Schlüssel und -Wert für zu löschende Ressourcen, um den Zeitraum oder einen bestimmten Zeitpunkt für die Außerbetriebnahme zu definieren.

Implementierungsschritte

- Implementieren eines Markierungsschemas: Implementieren Sie ein Markierungsschema, das den Workload identifiziert, zu dem die Ressource gehört, und stellen Sie sicher, dass alle Ressourcen innerhalb des Workloads entsprechend markiert sind. Durch das Markieren können Sie Ressourcen nach Zweck, Team, Umgebung oder anderen, für Ihr Unternehmen relevanten Kriterien kategorisieren. Detaillierte Informationen zu Anwendungsfällen, Strategien und Verfahren zum Markieren finden Sie in den bewährten Methoden beim Tagging in AWS.
- Implementieren des Workload-Durchsatzes oder der Ausgabekontrolle: Implementieren Sie die Überwachung des Workload-Durchsatzes oder die Ausgabe von Alarmsignalen, die entweder bei der Eingabe oder Ausgabe ausgelöst werden. Konfigurieren Sie die Überwachung so, dass Benachrichtigungen erstellt werden, wenn Workload-Anforderungen oder -Ausgaben auf Null fallen. Dies bedeutet, dass die Workload-Ressourcen nicht mehr verwendet werden. Integrieren Sie einen Zeitfaktor, wenn der Workload unter normalen Bedingungen regelmäßig auf Null fällt. Weitere Informationen zu ungenutzten oder selten genutzten Ressourcen finden Sie im Artikel zu Checks für die Kostenoptimierung mit AWS Trusted Advisor.
- Gruppieren von AWS-Ressourcen: Erstellen Sie Gruppen für AWS-Ressourcen. Mit AWS Resource Groups können Sie Ihre AWS-Ressourcen organisieren und verwalten, die sich in derselben AWS-Region befinden. Den meisten Ressourcen lassen sich Tags hinzufügen, um sie innerhalb der Organisation zu identifizieren und zu sortieren. Mit dem Tag Editor können Sie mehreren unterstützten Ressourcen gleichzeitig Tags hinzufügen. Ziehen Sie die Verwendung

von AWS Service Catalog in Erwägung, um Portfolios mit genehmigten Produkten zu erstellen, zu verwalten und an Endnutzer zu verteilen und um den Produktlebenszyklus zu verwalten.

Ressourcen

Zugehörige Dokumente:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS Trusted Advisor Cost Optimization Checks (Checks für die Kostenoptimierung mit AWS Trusted Advisor)
- Markieren von AWS-Ressourcen
- Veröffentlichen benutzerdefinierter Metriken

Zugehörige Videos:

- How to optimize costs using AWS Trusted Advisor (Kostenoptimierung mit AWS Trusted Advisor)

Zugehörige Beispiele:

- Organisieren von AWS-Ressourcen
- Optimize cost using AWS Trusted Advisor (Kostenoptimierung mit AWS Trusted Advisor)

COST04-BP02 Implementieren eines Prozesses für die Außerbetriebnahme

Implementieren Sie einen Prozess für die Identifizierung und Außerbetriebnahme nicht genutzter Ressourcen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Implementieren Sie einen standardisierten Prozess in Ihrem gesamten Unternehmen, um ungenutzte Ressourcen zu identifizieren und zu entfernen. Der Prozess sollte definieren, wie häufig Suchvorgänge durchgeführt werden, und die Prozesse zum Entfernen der Ressource festlegen, um sicherzustellen, dass alle Unternehmensanforderungen erfüllt sind.

Implementierungsschritte

- Erstellen und Implementieren eines Prozesses für die Außerbetriebnahme: Erstellen Sie in Zusammenarbeit mit den Workload-Entwicklern und -Besitzern einen Prozess zur Außerbetriebnahme des Workloads und seiner Ressourcen. Der Prozess sollte die Methode abdecken, um zu überprüfen, ob der Workload verwendet wird, und auch, ob jede der Workload-Ressourcen verwendet wird. Definieren Sie die erforderlichen Schritte, um die Ressource außer Betrieb zu nehmen und gleichzeitig die Einhaltung gesetzlicher Anforderungen sicherzustellen. Alle zugeordneten Ressourcen sollten dabei eingeschlossen werden, z. B. Lizenzen oder zugehöriger Speicher. Informieren Sie die Besitzer des Workloads darüber, dass die Außerbetriebnahme ausgeführt wurde.

  Die folgenden Schritte für die Außerbetriebnahme geben vor, was im Rahmen des Prozesses geprüft werden sollte:

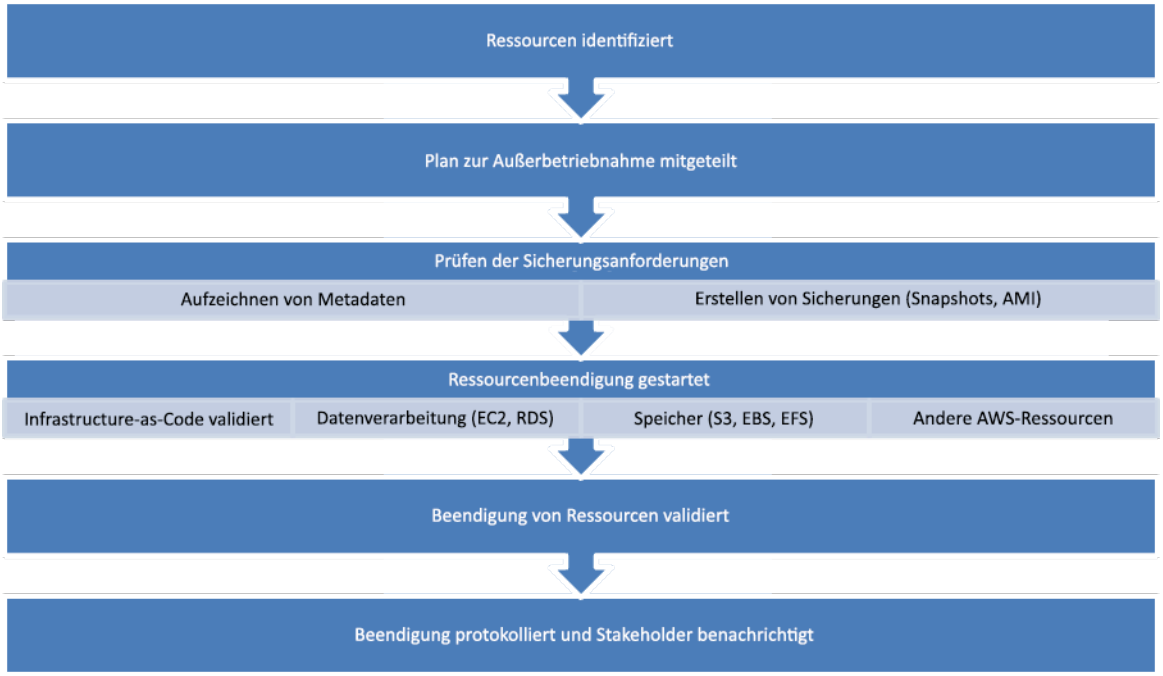
  - Identifizieren der Ressourcen, die außer Betrieb genommen werden sollen: Identifizieren Sie die Ressourcen, die in Ihrer AWS Cloud für die Außerbetriebnahme in Frage kommen. Erfassen Sie alle erforderlichen Informationen und planen Sie die Außerbetriebnahme. Achten Sie bei der Zeitplanung darauf, unerwartete Probleme im Prozess zu berücksichtigen.
  - Koordination und Kommunikation: Arbeiten Sie mit den Eigentümern der Workloads zusammen, um zu bestätigen, dass die Ressource außer Betrieb genommen werden soll.
  - Erfassen von Metadaten und Erstellen von Sicherungen: Erfassen Sie Metadaten (wie öffentliche IPs, Region, AZ, VPC, Subnetz und Sicherheitsgruppen) und erstellen Sie Sicherungen (z. B. Amazon Elastic Block Store-Snapshots oder AMI, Schlüssel- und Zertifikatexporte), wenn dies für die Ressourcen in der Produktionsumgebung erforderlich ist oder es sich um kritische Ressourcen handelt.
  - Validieren von Infrastructure-as-code: Bestimmen Sie, ob Ressourcen mit AWS CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK) oder einem anderen Infrastructure-as-code-Bereitstellungstool bereitgestellt wurden, damit sie bei Bedarf erneut bereitgestellt werden können.
  - Verhindern des Zugriffs: Wenden Sie restriktive Kontrollen für einen bestimmten Zeitraum an, um zu verhindern, dass Ressourcen genutzt werden, während Sie bestimmen, ob diese benötigt werden. Stellen Sie sicher, dass die Ressourcenumgebung bei Bedarf in den ursprünglichen Zustand zurückversetzt werden kann.
  - Einhalten des internen Prozesses für die Außerbetriebnahme: Halten Sie sich an die Verwaltungsaufgaben und den Außerbetriebnahmeprozess Ihrer Organisation, z. B. Entfernen der Ressourcen aus der Organisationsdomäne, Entfernen des DNS-Datensatzes und Entfernen der Ressourcen aus Ihrem Konfigurationsverwaltungstool, Überwachungstool, Automatisierungstools und Sicherheitstools.

Wenn es sich bei der Ressource um eine Amazon EC2-Instance handelt, beachten Sie folgende Liste. Weitere Informationen finden Sie unter „Wie kann ich meine Amazon EC2-Ressourcen löschen oder beenden?

- Beenden Sie alle Ihre Amazon EC2-Instances und Load Balancers. Amazon EC2-Instances sind in der Konsole noch kurze Zeit sichtbar, nachdem sie beendet wurden. Instances, die sich nicht im Ausführungsstatus befinden, werden Ihnen nicht in Rechnung gestellt.
- Löschen Sie Ihre Auto Scaling-Infrastruktur.
- Geben Sie alle Dedicated Hosts frei.
- Löschen Sie alle Amazon EBS-Volumes und Amazon EBS-Snapshots.
- Geben Sie alle elastischen IP-Adressen frei.
- Melden Sie alle Amazon Machine Images (AMIs) ab.
- Beenden Sie alle AWS Elastic Beanstalk-Umgebungen.

Wenn die Ressource ein Objekt im Amazon S3 Glacier-Speicher ist und Sie ein Archiv löschen, bevor die Mindestspeicherdauer erreicht wurde, wird eine anteilige Gebühr für das frühzeitige Löschen in Rechnung gestellt. Die Mindestspeicherdauer für Amazon S3 Glacier ist abhängig von der verwendeten Speicherklasse. Eine Übersicht über die Mindestspeicherdauer der einzelnen Speicherklassen finden Sie in der Übersicht über die Leistung für die verschiedenen Amazon S3-Speicherklassen. Informationen zu Gebühren für vor Ablauf der Mindestspeicherdauer gelöschte Objekte finden Sie in der Amazon S3-Preisübersicht.

Das folgende Flussdiagramm eines einfachen Außerbetriebnahmeprozesses zeigt die einzelnen Schritte. Bestätigen Sie vor der Außerbetriebnahme von Ressourcen, dass die Ressourcen, die Sie für die Außerbetriebnahme identifiziert haben, von der Organisation nicht genutzt werden.



Ablauf für die Außerbetriebnahme von Ressourcen.

Ressourcen

Zugehörige Dokumente:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

Zugehörige Videos:

- Delete CloudFormation stack but retain some resources (Löschen eines CloudFormation-Stacks unter Beibehaltung einiger Ressourcen)
- Find out which user launched Amazon EC2 instance (Ermitteln des Benutzers, der eine EC2-Instance gestartet hat)

Zugehörige Beispiele:

- Amazon EC2-Ressourcen löschen oder beenden
- Find out which user launched Amazon EC2 instance (Ermitteln des Benutzers, der eine EC2-Instance gestartet hat)

COST04-BP03 Außerbetriebnahme von Ressourcen

Außerbetriebnahme von Ressourcen, die durch Ereignisse wie regelmäßige Prüfungen oder Änderungen der Nutzung ausgelöst werden. Die Außerbetriebnahme erfolgt normalerweise regelmäßig und kann manuell oder automatisiert durchgeführt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Häufigkeit und der Aufwand für die Suche nach ungenutzten Ressourcen sollten die potenziellen Einsparungen widerspiegeln, sodass ein Konto mit geringen Kosten seltener analysiert werden sollte als ein Konto mit größeren Kosten. Suchanfragen und Außerbetriebnahmeereignisse können durch Statusänderungen im Workload ausgelöst werden, z. B. ein Produkt, das sich dem Ende seiner Lebensdauer nähert oder ersetzt wird. Suchen und Außerbetriebnahme können auch durch externe Ereignisse ausgelöst werden, wie z. B. Änderungen der Marktbedingungen oder Produktterminierung.

Implementierungsschritte

- Außerbetriebnahme von Ressourcen: Dies ist die Phase, in der AWS-Ressourcen, die nicht mehr benötigt werden oder deren Lizenzvereinbarung abläuft, als veraltet deaktiviert werden. Führen Sie alle abschließenden Prüfungen durch und erstellen Sie Snapshots und Sicherungen, bevor Sie zur Entsorgungsphase übergehen, um unerwünschte Unterbrechungen zu vermeiden. Befolgen Sie den Außerbetriebnahmeprozess, um jede der Ressourcen, die als nicht genutzt identifiziert wurde, außer Betrieb zu nehmen.

Ressourcen

Zugehörige Dokumente:

- AWS Auto Scaling
- AWS Trusted Advisor

Zugehörige Beispiele:

- Well-Architected Labs: Außerbetriebnahme von Ressourcen (Stufe 100)

COST04-BP04 Automatische Stilllegung von Ressourcen

Gestalten Sie Ihren Workload so, dass er die Beendigung von Ressourcen reibungslos handhabt, wenn Sie unkritische Ressourcen, nicht benötigte Ressourcen oder Ressourcen mit geringer Auslastung identifizieren und außer Betrieb nehmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Verwenden Sie die Automatisierung, um die damit verbundenen Kosten für die Außerbetriebnahme zu reduzieren oder zu entfernen. Wenn Sie Ihren Workload so konzipieren, dass er eine automatische Außerbetriebnahme durchführt, werden die gesamten Workload-Kosten während der Nutzungsdauer gesenkt. Sie können AWS Auto Scaling verwenden, um die Außerbetriebnahme durchzuführen. Sie können auch benutzerdefinierten Code mithilfe der API oder des SDK implementieren, um Workload-Ressourcen automatisch außer Betrieb zu nehmen.

Moderne Anwendungen werden Serverless-First erstellt, d. h. mit einer Strategie, die die Nutzung von Serverless-Services priorisiert. AWS hat Serverless-Services für alle drei Stack-Ebenen entwickelt: Datenverarbeitung, Integration und Datenspeicher. Mit einer Serverless-Architektur können Sie in Phasen mit wenig Datenverkehr dank automatischer Skalierung Kosten sparen.

Implementierungsschritte

- Implementieren von AWS Auto Scaling: Konfigurieren Sie unterstützte Ressourcen mit AWS Auto Scaling. Mit AWS Auto Scaling können Sie die Nutzung und Kosteneffizienz bei der Verwendung von AWS-Services optimieren. Wenn die Nachfrage sinkt, entfernt AWS Auto Scaling automatisch überschüssige Ressourcenkapazitäten, damit keine unnötigen Kosten entstehen.
- Konfigurieren von CloudWatch zum Beenden von Instances: Das Beenden von Instances kann mithilfe von CloudWatch-Alarmen konfiguriert werden. Implementieren Sie mithilfe der Metriken aus dem Außerbetriebnahmeprozess einen Alarm mit einer Amazon Elastic Compute Cloud-Aktion. Überprüfen Sie den Vorgang vor der Einführung in einer Nicht-Produktionsumgebung.
- Implementieren von Code innerhalb des Workloads: Sie können Workload-Ressourcen mit dem AWS SDK oder der AWS CLI außer Betrieb nehmen. Implementieren Sie Code innerhalb der in AWS integrierten Anwendung, die nicht mehr verwendete Ressourcen beendet oder entfernt.
- Verwenden von Serverless-Services: Priorisieren Sie das Erstellen von Serverless-Architekturen und ereignisgesteuerten Architekturen in AWS, um Ihre Anwendungen zu erstellen und auszuführen. AWS bietet Services mit verschiedenen Serverless-Technologien an, die von sich aus eine automatisch optimierte Ressourcennutzung und automatisierte

Außerbetriebnahme bereitstellen (Abskalieren und Aufskalieren). Bei Serverless-Anwendungen wird die Ressourcennutzung automatisch optimiert und Ihnen entstehen nie Kosten für die Überbereitstellung.

Ressourcen

Zugehörige Dokumente:

- AWS Auto Scaling
- AWS Trusted Advisor
- Serverless on AWS (Serverless in AWS)
- Create Alarms to Stop, Terminate, Reboot, or Recover an Instance (Erstellen von Alarmen, um eine Instance zu stoppen, zu beenden, neu zu starten oder wiederherzustellen)
- Erste Schritte mit Amazon EC2 Auto Scaling
- Adding terminate actions to Amazon CloudWatch alarms (Hinzufügen von Aktionen zum Beenden in Amazon CloudWatch-Alarmen)

Zugehörige Beispiele:

- Scheduling automatic deletion of AWS CloudFormation stacks (Planen des automatischen Löschens von AWS CloudFormation-Stacks)
- Well-Architected Labs – Automatische Außerbetriebnahme von Ressourcen (Stufe 100)
- Servian AWS Auto Cleanup

COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung

Definieren Sie Richtlinien zur Datenaufbewahrung auf unterstützten Ressourcen, um das Löschen von Objekten gemäß den Anforderungen Ihres Unternehmens durchzuführen. Identifizieren und löschen Sie entbehrliche und verwaiste Ressourcen und Objekte, die nicht mehr benötigt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Mit Richtlinien zur Datenaufbewahrung und Lebenszyklusrichtlinien können Sie die mit der Außerbetriebnahme von Prozessen verbundenen Kosten sowie die Speicherkosten für die identifizierten Ressourcen reduzieren. Die Definition von Richtlinien zur Datenaufbewahrung und Lebenszyklusrichtlinien zur Durchführung einer automatischen Speicherklassenmigration und

Löschung verringert die Gesamtspeicherkosten während des Lebenszyklus. Sie können Amazon Data Lifecycle Manager verwenden, um die Erstellung und Löschung von Amazon Elastic Block Store-Snapshots und Amazon EBS-gestützten Amazon Machine Images (AMIs) zu automatisieren, und Sie können Amazon S3 Intelligent-Tiering oder eine Amazon S3-Lebenszyklus-Konfiguration verwenden, um den Lebenszyklus Ihrer Amazon S3-Objekte zu verwalten. Mithilfe der API oder dem SDK können Sie auch benutzerdefinierten Code implementieren, um Lebenszyklusrichtlinien und Richtlinienregeln für die automatische Löschung von Objekten zu erstellen.

Implementierungsschritte

- Verwenden von Amazon Data Lifecycle Manager: Verwenden Sie Lebenszyklusrichtlinien auf Amazon Data Lifecycle Manager, um die Löschung von Amazon EBS-Snapshots und Amazon EBS-gestützten AMIs zu automatisieren.
- Einrichten der Lebenszyklus-Konfiguration auf einem Bucket: Verwenden Sie die Amazon S3-Lebenszyklus-Konfiguration auf einem Bucket, um Aktionen für Amazon S3 zu definieren, die während des Lebenszyklus des Objekts ergriffen werden sollen, sowie die Löschung am Ende des Lebenszyklus des Objekts basierend auf Ihren geschäftlichen Anforderungen.

Ressourcen

Zugehörige Dokumente:

- AWS Trusted Advisor
- Amazon Data Lifecycle Manager
- So richten Sie die Lebenszyklus-Konfiguration auf dem Amazon S3-Bucket ein

Zugehörige Videos:

- Automate Amazon EBS Snapshots with Amazon Data Lifecycle Manager (EC2-Snapshots mit AWS Lifecycle Manager automatisieren)
- Empty an Amazon S3 bucket using a lifecycle configuration rule (Einen Amazon S3-Bucket unter Verwendung einer Regel für die Lebenszyklus-Konfiguration leeren)

Zugehörige Beispiele:

- Einen Amazon S3-Bucket unter Verwendung einer Regel für die Lebenszyklus-Konfiguration leeren
- Well-Architected Lab: Automatische Außerbetriebnahme von Ressourcen (Stufe 100)

Kostengünstige Ressourcen

Fragen

- KOSTEN 5. Wie können Sie die Kosten bei der Auswahl von Services einschätzen?
- KOSTEN 6. Wie können Sie bei der Auswahl des Ressourcentyps, -umfangs und der Anzahl der Ressourcen Kostenziele erfüllen?
- KOSTEN 7. Wie können Sie Kosten mithilfe von Preismodellen senken?
- KOSTEN 8. Wie können Sie die Kosten für Datenübertragungen planen?

KOSTEN 5. Wie können Sie die Kosten bei der Auswahl von Services einschätzen?

Bei Amazon EC2, Amazon EBS und Amazon S3 handelt es sich um AWS-Services, die als einzelne Bausteine angeboten werden. Verwaltete Services, etwa Amazon RDS und Amazon DynamoDB, sind AWS-Services auf einer höheren Ebene oder Anwendungsebene. Wenn Sie sich für die richtigen Bausteine und verwalteten Services entscheiden, können Sie die Kosten dieses Workloads optimieren. Durch die Nutzung von verwalteten Services können Sie einen Großteil Ihres administrativen und betrieblichen Overheads reduzieren oder beseitigen und damit Kapazitäten für anwendungs- und geschäftsbezogene Aktivitäten gewinnen.

Bewährte Methoden

- COST05-BP01 Ermitteln der Organisationsanforderungen zur Kosteneinschätzung
- COST05-BP02 Analysieren sämtlicher Komponenten dieses Workloads
- COST05-BP03 Durchführen einer gründlichen Analyse der einzelnen Komponenten
- COST05-BP04 Auswahl von Software mit kostengünstiger Lizenzierung
- COST05-BP05 Auswahl von Komponenten dieses Workloads zur Optimierung der Kosten im Einklang mit den Prioritäten der Organisation
- COST05-BP06 Durchführen einer Kostenanalyse für unterschiedliche Nutzungen im Lauf der Zeit

COST05-BP01 Ermitteln der Organisationsanforderungen zur Kosteneinschätzung

Definieren Sie gemeinsam mit den Teammitgliedern für diesen Workload das Gleichgewicht zwischen Kostenoptimierung und anderen Säulen wie Leistung und Zuverlässigkeit.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

In den meisten Unternehmen besteht die Abteilung für Informationstechnologie (IT) aus mehreren kleinen Teams, von denen jedes seine eigene Agenda und seinen eigenen Schwerpunktbereich hat, der die Spezialgebiete und Fähigkeiten seiner Teammitglieder widerspiegelt. Sie müssen die allgemeinen Ziele, Prioritäten und Vorgaben Ihrer Organisation verstehen und wissen, wie jede Abteilung oder jedes Projekt zu diesen Zielen beiträgt. Die Kategorisierung aller wesentlichen Ressourcen, einschließlich Personal, Ausrüstung, Technologie, Material und externer Dienstleistungen, ist für die Erreichung der organisatorischen Ziele und eine umfassende Budgetplanung von entscheidender Bedeutung. Die Anwendung dieses systematischen Ansatzes zur Kostenermittlung und zum Kostenverständnis ist für die Erstellung eines realistischen und soliden Kostenplans für die Organisation von grundlegender Bedeutung.

Bei der Auswahl von Services für Ihren Workload ist es wichtig, dass Sie die Prioritäten Ihres Unternehmens verstehen. Stellen Sie ein Gleichgewicht zwischen Kostenoptimierung und anderen Säulen des Well-Architected Frameworks von AWS her, wie z. B. Leistung und Zuverlässigkeit. Dieser Prozess sollte systematisch und regelmäßig durchgeführt werden, um Veränderungen in den Zielen der Organisation, den Marktbedingungen und der betrieblichen Dynamik zu berücksichtigen. Ein vollständig kostenoptimierter Workload ist die Lösung, die am meisten an den Anforderungen Ihres Unternehmens ausgerichtet ist, nicht notwendigerweise an den niedrigsten Kosten. Treffen Sie sich mit allen Teams innerhalb Ihres Unternehmens, um Informationen zu sammeln, z. B. mit den Produkt-, Geschäfts-, Technik- und Finanz-Teams. Bewerten Sie die Auswirkungen von Kompromissen zwischen konkurrierenden Interessen oder alternativen Ansätzen, um fundiert zu entscheiden, auf welche Bereiche die operativen Anstrengungen konzentriert werden sollten, oder eine geeignete Handlungsweise zu wählen.

Beispielsweise kann die Beschleunigung der Markteinführung neuer Funktionen einer Kostenoptimierung vorgezogen werden oder Sie können eine relationale Datenbank für nicht relationale Daten wählen, um die Migration eines Systems zu vereinfachen, anstatt zu einer für Ihren Datentyp optimierten Datenbank zu migrieren und Ihre Anwendung zu aktualisieren.

Implementierungsschritte

- Ermitteln der Kostenanforderungen der Organisation: Treffen Sie sich mit den Teammitgliedern Ihrer Organisation, einschließlich Mitarbeitern aus dem Produktmanagement, den Anwendungseigentümern, den Entwicklungs- und Betriebsteams, dem Management und den Finanzverantwortlichen. Setzen Sie die Prioritäten hinsichtlich der Well-Architected-Säulen für diesen Workload und seine Komponenten. Die Ausgabe sollte eine Liste der Säulen in der entsprechenden Reihenfolge sein. Sie können jeder Säule auch eine Gewichtung zuweisen, um

anzugeben, wie viel zusätzlicher Fokus sie hat, oder wie ähnlich der Fokus zwischen zwei Säulen ist.

- Adressieren und Dokumentieren der technischen Schulden: Gehen Sie bei der Überprüfung des Workloads auf die technischen Schulden ein. Dokumentieren Sie ein Backlog-Element, um den Workload in Zukunft wieder aufzugreifen und erneut zu überarbeiten oder neu zu strukturieren, mit dem Ziel, ihn weiter zu optimieren. Es ist wichtig, dass Sie die Kompromisse, die Sie eingegangen sind, den anderen Beteiligten klar mitteilen.

Ressourcen

Zugehörige bewährte Methoden:

- REL11-BP07 Architektur Ihres Produkts zur Erfüllung von Verfügbarkeitszielen und Uptime-SLAs (Service Level Agreements)
- OPS01-BP06 Bewerten von Kompromissen

Zugehörige Dokumente:

- AWS-Gesamtbetriebskostenrechner (Total Cost of Ownership, TCO)
- Amazon S3-Speicherklassen
- Cloud-Produkte

COST05-BP02 Analysieren sämtlicher Komponenten dieses Workloads

Stellen Sie sicher, dass jede Workload-Komponente unabhängig von der derzeitigen Größe oder den aktuellen Kosten analysiert wird. Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zu dem potenziellen Nutzen stehen, z. B. bei einer Prüfung der derzeitigen und prognostizierten Kosten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Workload-Komponenten, die der Organisation einen geschäftlichen Nutzen bringen sollen, können verschiedene Services umfassen. Für jede Komponente können Sie bestimmte AWS Cloud-Services auswählen, um den Geschäftsanforderungen gerecht zu werden. Diese Auswahl könnte von Faktoren wie der Vertrautheit mit diesen Services oder früheren Erfahrungen mit ihnen beeinflusst sein.

Nachdem Sie die Anforderungen Ihrer Organisation ermittelt haben (wie in COST05-BP01 Ermitteln der Organisationsanforderungen zur Kosteneinschätzung erwähnt), führen Sie eine gründliche Analyse aller Komponenten Ihres Workloads durch. Analysieren Sie jede Komponente unter Berücksichtigung der aktuellen und prognostizierten Kosten und Größen. Wägen Sie die Kosten der Analyse gegen die potenziellen Einsparungen beim Workload während des Lebenszyklus ab. Der Aufwand, der für die Analyse aller Komponenten dieses Workloads betrieben wird, sollte den potenziellen Einsparungen oder Verbesserungen entsprechen, die durch die Optimierung dieser spezifischen Komponente zu erwarten sind. Wenn zum Beispiel die Kosten der vorgeschlagenen Ressource 10 USD/Monat betragen und bei prognostizierter Belastung 15 USD/Monat nicht überschreiten würden, könnte ein Tag Aufwand, um die Kosten um 50 % zu reduzieren (5 USD pro Monat), den potenziellen Nutzen über die Lebensdauer des Systems übersteigen. Verwenden Sie eine schnellere und effizientere datenbasierte Schätzung, um das beste Gesamtergebnis für diese Komponente zu erzielen.

Workloads können sich im Laufe der Zeit ändern. Die richtigen Services sind möglicherweise nicht optimal, wenn sich die Workload-Architektur oder -Nutzung ändert. Die Analyse für die Auswahl von Services muss aktuelle und zukünftige Workload-Zustände und Nutzungsebenen umfassen. Die Implementierung eines Service für den zukünftigen Workload-Status oder die Nutzung kann die Gesamtkosten senken, indem der Aufwand reduziert oder beseitigt wird, der für zukünftige Änderungen erforderlich ist. Zum Beispiel könnte die Verwendung von EMR Serverless zunächst die richtige Wahl sein. Wenn jedoch die Nutzung dieses Services zunimmt, könnte die Umstellung auf EMR in EC2 die Kosten für diese Komponente des Workloads senken.

AWS Cost Explorer und die AWS Cost and Usage Reports (CUR) können die Kosten eines Machbarkeitsnachweises (Proof of Concept, PoC) oder einer laufenden Umgebung analysieren. Sie können AWS -Preisrechner auch verwenden, um die Workload-Kosten zu schätzen.

Schreiben Sie einen Workflow, an den sich die technischen Teams halten, um ihre Workloads zu überprüfen. Halten Sie diesen Workflow einfach, decken Sie aber auch alle notwendigen Schritte ab, um sicherzustellen, dass die Teams jede Komponente des Workloads und seine Preisgestaltung verstehen. Ihre Organisation kann diesen Workflow dann verfolgen und an die spezifischen Bedürfnisse jedes Teams anpassen.

1. Listen Sie jeden Dienst auf, der für den Workload verwendet wird: Dies ist ein guter Ausgangspunkt. Identifizieren Sie alle Services, die derzeit genutzt werden und woher die Kosten stammen.
2. Verstehen Sie, wie die Preisgestaltung für diese Services funktioniert: Machen Sie sich mit dem Preismodell der einzelnen Services vertraut. Verschiedene AWS-Services haben unterschiedliche

Preismodelle, die auf Faktoren wie Nutzungsvolumen, Datenübertragung und Feature-spezifischen Preisen basieren.

3. Konzentrieren Sie sich auf die Services, für die unerwartete Workloadkosten anfallen und die nicht mit der erwarteten Nutzung und dem erwarteten Geschäftsergebnis übereinstimmen: Identifizieren Sie Ausreißer oder Services, bei denen die Kosten nicht proportional zum Wert oder zur Nutzung durch AWS Cost Explorer oder AWS Cost and Usage Report sind. Es ist wichtig, die Kosten mit den Geschäftsergebnissen zu korrelieren, um Optimierungsmaßnahmen zu priorisieren.
4. Nutzen Sie AWS Cost Explorer, CloudWatch Logs, VPC Flow Logs und Amazon S3 Storage Lens, um die Ursache dieser hohen Kosten zu verstehen: Diese Tools sind für die Diagnose hoher Kosten von entscheidender Bedeutung. Jeder Dienst bietet einen anderen Blickwinkel, um die Nutzung und Kosten zu betrachten und zu analysieren. Cost Explorer hilft beispielsweise bei der Bestimmung der Gesamtkostentrends, CloudWatch Logs liefert betriebliche Erkenntnisse, VPC Flow Logs zeigt den IP-Verkehr an und Amazon S3 Storage Lens ist nützlich für Speicheranalysen.
5. Verwenden Sie AWS Budgets, um Budgets für bestimmte Beträge für Services oder Konten festzulegen: Die Festlegung von Budgets ist eine proaktive Methode zur Kostenverwaltung. Nutzen Sie AWS Budgets, um benutzerdefinierte Budgetschwellenwerte festzulegen und Warnmeldungen zu erhalten, wenn die Kosten diese Schwellenwerte überschreiten.
6. Konfigurieren Sie Amazon CloudWatch-Alarme zum Senden von Abrechnungs- und Nutzungsmetriken: Richten Sie Überwachungs- und Warnmeldungen für Kosten- und Nutzungsmetriken ein. CloudWatch-Alarme können Sie benachrichtigen, wenn bestimmte Schwellenwerte überschritten werden, was die Reaktionszeit verbessert.

Erzielen Sie im Laufe der Zeit bemerkenswerte Verbesserungen und finanzielle Einsparungen durch eine strategische Überprüfung aller Workload-Komponenten, unabhängig von ihren gegenwärtigen Merkmalen. Der Aufwand für diesen Überprüfungsprozess sollte bewusst und unter sorgfältiger Abwägung der möglichen Vorteile betrieben werden.

Implementierungsschritte

- Erstellen einer Liste der Workload-Komponenten: Erstellen Sie eine Liste mit den Komponenten Ihres Workloads. Verwenden Sie diese Liste, um zu überprüfen, ob jede Komponente analysiert wurde. Der Aufwand sollte die Kritikalität für den Workload widerspiegeln, die durch die Prioritäten Ihrer Organisation definiert wird. Die Gruppierung von Ressourcen verbessert die Effizienz (z. B. die Speicherung von Produktionsdatenbanken, wenn es mehrere Datenbanken gibt).

- Priorisieren Sie die Komponentenliste: Priorisieren Sie die Komponentenliste entsprechend dem Aufwand. In der Regel erfolgt die Priorisierung nach den Kosten der Komponente – von der teuersten zur günstigsten. Alternativ kann sie auch nach der von den Prioritäten Ihrer Organisation definierten Kritikalität erfolgen.
- Führen Sie die Analyse durch: Überprüfen Sie für jede Komponente auf der Liste die verfügbaren Optionen und Services und wählen Sie die Option aus, die am besten mit Ihren Organisationsprioritäten übereinstimmt.

Ressourcen

Zugehörige Dokumente:

- AWS -Preisrechner
- AWS Cost Explorer
- Amazon S3-Speicherklassen
- AWS Cloud-Produkte

Zugehörige Videos:

- AWS-Serie zur Kostenoptimierung: CloudWatch

COST05-BP03 Durchführen einer gründlichen Analyse der einzelnen Komponenten

Nehmen Sie die Gesamtkosten, die der Organisation durch die einzelnen Komponenten entstehen, unter die Lupe. Berechnen Sie die Gesamtbetriebskosten unter Berücksichtigung der Betriebs- und Verwaltungskosten, insbesondere bei der Nutzung von verwalteten Services durch den Cloud-Anbieter. Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen, z. B. muss die Zeit, die für die Analyse benötigt wird, den Komponentenkosten entsprechen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Bedenken Sie die Zeitersparnis, die es Ihrem Team ermöglicht, sich auf das Aufholen technischen Rückstands, Innovation, wertschöpfende Funktionen und die Herausarbeitung eines Alleinstellungsmerkmals zu konzentrieren. So könnten Sie beispielsweise Ihre Datenbank von Ihrer lokalen Umgebung so schnell wie möglich in die Cloud verlagern (auch als Hostwechsel

bekannt) und die Optimierung im Nachgang ausführen. Es lohnt sich, die möglichen Einsparungen zu untersuchen, die Sie durch den Einsatz von verwalteten Services auf AWS erzielen könnten, die Lizenzkosten entfernen oder reduzieren. Verwaltete Services auf AWS eliminieren den betrieblichen und administrativen Aufwand für die Wartung eines Service, wie das Patching oder die Aktualisierung des Betriebssystems, sodass Sie sich auf Innovationen und das Geschäft konzentrieren können.

Da verwaltete Services in der großen Cloud-Umgebung ausgeführt werden, profitieren Sie hier von geringeren Kosten pro Transaktion oder Service. Sie können potenzielle Optimierungen vornehmen, um konkrete Vorteile zu erzielen, ohne die Kernarchitektur der Anwendung zu ändern. Beispielsweise ist es möglich, den Zeitaufwand, den Sie für die Verwaltung von Datenbank-Instances aufbringen, zu verringern, indem Sie zu einer Database-as-a-Service-Plattform wie Amazon Relational Database Service (Amazon RDS) migrieren oder Ihre Anwendung in eine vollständig verwaltete Plattform wie AWS Elastic Beanstalk migrieren.

Verwaltete Services weisen in der Regel Attribute auf, die Sie festlegen können, um zu gewährleisten, dass ausreichend Kapazität bereitsteht. Sie müssen diese Attribute festlegen und überwachen, damit Ihre überschüssige Kapazität auf ein Minimum begrenzt und die Leistung maximiert werden. Sie können die Attribute der AWS Managed Services mithilfe der AWS Management Console oder AWS-APIs und SDKs ändern, um den Ressourcenbedarf an den sich ändernden Bedarf anzupassen. So können Sie beispielsweise die Anzahl der Knoten in einem Amazon EMR-Cluster (oder einem Amazon Redshift-Cluster) auf- oder abskalieren.

Außerdem können Sie mehrere Instances in eine AWS-Ressource legen, um eine Nutzung mit höherer Dichte zu aktivieren. Sie können beispielsweise mehrere kleine Datenbanken auf einer einzelnen Amazon Relational Database Service (Amazon RDS) Datenbank-Instance bereitstellen. Mit zunehmendem Wachstum können Sie eine der Datenbanken über einen Snapshot- und Wiederherstellungsprozess auf eine spezielle Amazon RDS-Datenbank-Instance migrieren.

Wenn Sie Workloads auf verwalteten Services bereitstellen, müssen Sie sich mit den Anforderungen für das Anpassen der Service-Kapazität vertraut machen. Diese Anforderungen sind in der Regel Zeit, Aufwand und die Auswirkungen auf den normalen Workload-Betrieb. Die bereitgestellte Ressource muss Zeit für Änderungen einräumen und den erforderlichen Overhead bereitstellen, damit dies möglich ist. Der laufende Aufwand für das Ändern von Services kann praktisch auf null reduziert werden, wenn Sie APIs und SDKs verwenden, die mit System- und Überwachungs-Tools wie Amazon CloudWatch integriert sind.

Amazon RDS, Amazon Redshift und Amazon ElastiCache bieten einen verwalteten Analyseservice. Amazon Athena, Amazon EMR, and Amazon OpenSearch Service stellen einen verwalteten Datenbankservice bereit.

AMS ist ein Service, der die AWS-Infrastruktur für Unternehmenskunden und -partner betreibt. Es bietet eine sichere und konforme Umgebung, in der Sie Ihre Workloads bereitstellen können. AMS verwendet Enterprise-Cloud-Betriebsmodelle mit Automatisierung, damit Sie Ihre Unternehmensanforderungen erfüllen, schneller in die Cloud wechseln und Ihre laufenden Verwaltungskosten senken können.

Implementierungsschritte

- Durchführen einer gründliche Analyse: Arbeiten Sie anhand der Komponentenliste jede Komponente von der höchsten Priorität bis zur niedrigsten Priorität ab. Führen Sie für die Komponenten mit höherer Priorität sowie für die teureren Komponenten zusätzliche Analysen durch und bewerten Sie alle verfügbaren Optionen und deren langfristige Auswirkungen. Bewerten Sie bei Komponenten mit niedrigerer Priorität, ob Änderungen in der Nutzung die Priorität der Komponente ändern. Führen Sie anschließend eine Analyse des angemessenen Aufwands durch.
- Vergleichen von verwalteten und nicht verwalteten Ressourcen: Berücksichtigen Sie die Betriebskosten für die von Ihnen verwalteten Ressourcen und vergleichen Sie sie mit von AWS verwalteten Ressourcen. Prüfen Sie beispielsweise Ihre Datenbanken, die auf Amazon EC2-Instances ausgeführt werden, und vergleichen Sie sie mit Amazon RDS-Optionen (ein AWS von verwalteter Service) oder Amazon EMR verglichen mit der Ausführung von Apache Spark auf Amazon EC2. Recherchieren Sie sorgfältig, welche Optionen Sie beim Wechsel von einem selbstverwalteten Workload zu einem vollständig verwalteten AWS-Workload haben. Berücksichtigen Sie dabei die drei wichtigsten Faktoren: die Art des verwalteten Service, den Sie verwenden möchten, den Prozess, den Sie zur Migration Ihrer Daten verwenden, und ein Verständnis des AWS-Modells der geteilten Verantwortung.

Ressourcen

Zugehörige Dokumente:

- AWS-Gesamtbetriebskostenrechner (Total Cost of Ownership, TCO)
- Amazon S3-Speicherklassen
- AWS Cloud-Produkte
- AWS-Modell der geteilten Verantwortung

Zugehörige Videos:

- Why move to a managed database? (Warum zu einer verwalteten Datenbank wechseln?)

- What is Amazon EMR and how can I use it for processing data? (Was ist Amazon EMR und wie kann ich es für die Verarbeitung von Daten verwenden?)

Zugehörige Beispiele:

- Warum zu einer verwalteten Datenbank wechseln
- Daten von identischen SQL Server-Datenbanken mithilfe von AWS DMS in eine einzelne Amazon RDS for SQL Server-Datenbank konsolidieren
- Daten in großem Umfang an Amazon Managed Streaming for Apache Kafka (Amazon MSK) übermitteln
- Eine ASP.NET-Webanwendung zu AWS Elastic Beanstalk migrieren

COST05-BP04 Auswahl von Software mit kostengünstiger Lizenzierung

Open-Source-Software eliminiert Softwarelizenzkosten, die erhebliche Kosten in Workloads verursachen können. Wenn lizenzierte Software erforderlich ist, vermeiden Sie Lizenzen, die an beliebige Attribute wie CPUs gebunden sind, und suchen Sie nach Lizenzen, die an die Ausgabe oder Ergebnisse gebunden sind. Die Kosten dieser Lizenzen lassen sich besser auf die von ihnen bereitgestellten Vorteile skalieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Der Begriff „Open Source“ hat seinen Ursprung in der Softwareentwicklung und bedeutet, dass die Software bestimmte Kriterien für die freie Verteilung erfüllt. Open-Source-Software zeichnet sich durch einen Quellcode aus, der von jedem eingesehen, verändert und verbessert werden kann. Auf Grundlage der geschäftlichen Anforderungen, der Fähigkeiten der Techniker, der prognostizierten Nutzung oder anderer technologischer Abhängigkeiten können Organisationen die Verwendung von Open-Source-Software in AWS in Betracht ziehen, um ihre Lizenzkosten zu minimieren. Mit anderen Worten, die Kosten für Softwarelizenzen können durch den Einsatz von Open-Source-Software gesenkt werden. Dies kann erhebliche Auswirkungen auf die Workload-Kosten haben, da die Größe des Workloads skaliert wird.

Wägen Sie die Vorteile lizenzierter Software gegen die Gesamtkosten ab, um Ihren Workload zu optimieren. Modellieren Sie Änderungen bei der Lizenzierung und wie sich diese auf Ihre Workload-Kosten auswirken würden. Wenn ein Anbieter die Kosten Ihrer Datenbanklizenz ändert, untersuchen

Sie, wie sich dies auf die Gesamteffizienz Ihres Workloads auswirkt. Berücksichtigen Sie historische Preisankündigungen von Ihren Anbietern für Trends bei Lizenzänderungen in ihren Produkten. Die Lizenzkosten können auch unabhängig vom Durchsatz oder der Nutzung skaliert werden, z. B. Lizenzen, die nach Hardware skaliert werden (CPU-gebundene Lizenzen). Diese Lizenzen sollten vermieden werden, da sich die Kosten ohne entsprechende Ergebnisse schnell erhöhen können.

Wenn Sie beispielsweise eine Amazon EC2-Instance in us-east-1 mit einem Linux-Betriebssystem betreiben, können Sie die Kosten um etwa 45 % senken, verglichen mit einer anderen Amazon EC2-Instance, die unter Windows läuft.

AWS -Preisrechner bietet eine umfassende Möglichkeit, die Kosten verschiedener Ressourcen mit unterschiedlichen Lizenzoptionen zu vergleichen, z. B. Amazon RDS-Instances und verschiedene Datenbank-Engines. Darüber hinaus bietet das AWS Cost Explorer eine unschätzbare Perspektive für die Kosten bestehender Workloads, insbesondere derjenigen, die mit verschiedenen Lizenzen einhergehen. Für die Lizenzverwaltung bietet AWS License Manager eine optimierte Methode zur Überwachung und Verwaltung von Softwarelizenzen. Kunden können ihre bevorzugte Open-Source-Software in der AWS Cloud bereitstellen und einsetzen.

Implementierungsschritte

- Analysieren der Lizenzoptionen: Überprüfen Sie die Lizenzbedingungen der verfügbaren Software. Suchen Sie nach Open-Source-Versionen, die über die erforderliche Funktionalität verfügen, und stellen Sie fest, ob die Vorteile der lizenzierten Software die Kosten überwiegen. Bei günstigen Bedingungen stimmen die Kosten der Software mit ihren Vorteilen überein.
- Analysieren des Softwareanbieters: Überprüfen Sie alle bisherigen Preis- oder Lizenzänderungen des Anbieters. Suchen Sie nach Änderungen, die nicht im Einklang mit den Ergebnissen stehen, wie z. B. Strafen für die Ausführung auf Hardware oder Plattformen bestimmter Anbieter. Achten Sie zudem darauf, wie mögliche Prüfungen und Strafen durchgeführt werden.

Ressourcen

Zugehörige Dokumente:

- Open Source in AWS
- AWS-Gesamtbetriebskostenrechner (Total Cost of Ownership, TCO)
- Amazon S3-Speicherklassen
- Cloud-Produkte

Zugehörige Beispiele:

- Open-Source-Blogs
- AWS Open-Source-Blogs
- Optimierung und Lizenzbewertung

COST05-BP05 Auswahl von Komponenten dieses Workloads zur Optimierung der Kosten im Einklang mit den Prioritäten der Organisation

Berücksichtigen Sie bei der Auswahl sämtlicher Komponenten für Ihren Workload die Kosten. Dies umfasst die Nutzung von verwalteten Services und Services auf Anwendungsebene oder einer Serverless-, Container- oder ereignisgesteuerten Architektur, um die Gesamtkosten zu verringern. Minimieren Sie Lizenzkosten mithilfe von Open-Source-Software, Software, für die keine Lizenzgebühren anfallen, oder Alternativen zur Verringerung der Ausgaben.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Berücksichtigen Sie die Kosten von Services und Optionen, wenn Sie alle Komponenten auswählen. Dies beinhaltet auch die Verwendung von Services auf Anwendungsebene sowie verwalteter Services wie etwa Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Simple Notification Service (Amazon SNS) und Amazon Simple Email Service (Amazon SES) zur Reduzierung der Gesamtkosten der Organisation.

Verwenden Sie Serverless-Lösungen und Container für die Datenverarbeitung, zum Beispiel AWS Lambda und Amazon Simple Storage Service (Amazon S3) für statische Websites. Containerisieren Sie Ihre Anwendung wenn möglich und verwenden Sie verwaltete AWS-Container-Services wie Amazon Elastic Container Service (Amazon ECS) oder Amazon Elastic Kubernetes Service (Amazon EKS).

Minimieren Sie Lizenzkosten, indem Sie Open-Source-Software oder Software ohne Lizenzgebühren verwenden, wie z. B. Amazon Linux für Datenverarbeitungs-Workloads. Alternativ können Sie Datenbanken auch zu Amazon Aurora migrieren.

Sie können serverlose Services oder Services auf Anwendungsebene wie Lambda, Amazon Simple Queue Service (Amazon SQS), Amazon SNSund Amazon SES. Mit diesen Services müssen Sie keine Ressourcen mehr verwalten und sie stellen die Funktion der Codeausführung,

Warteschlangenservices und Nachrichtenzustellung bereit. Der andere Vorteil besteht darin, dass die Leistung und Kosten entsprechend der Nutzung skaliert werden, was eine effiziente Kostenzuordnung ermöglicht.

Die Verwendung einer ereignisorientierten Architektur ist auch mit Serverless-Services möglich. Ereignisgesteuerte Architekturen sind Push-basiert, es geschieht also alles On-Demand, während das Ereignis im Router auftritt. So bezahlen Sie nicht für eine kontinuierliche Abfragung, um auf ein Ereignis zu prüfen. Das Ergebnis; weniger Verbrauch der Netzwerkbandbreite, weniger CPU-Nutzung, weniger nicht genutzte Flottenkapazität und weniger SSL-/TLS-Handshakes.

Weitere Informationen zur Serverless-Technologie finden Sie im Whitepaper "Well-Architected Serverless Application Lens".

Implementierungsschritte

- Auswahl der einzelnen Services zur Kostenoptimierung: Wählen Sie unter Verwendung Ihrer Prioritätenliste und Analyse jede Option aus, die am besten mit Ihren Organisationsprioritäten übereinstimmt. Statt die Kapazität zu erhöhen, um die Nachfrage zu erfüllen, denken Sie über andere Optionen nach, die eine bessere Leistung mit geringeren Kosten bedeuten können. Wenn Sie beispielsweise den erwarteten Datenverkehr für Ihre Datenbanken auf AWS prüfen, entweder die Instance vergrößern oder Amazon ElastiCache-Services (Redis oder Memcached) verwenden müssen, um Ihren Datenbanken zwischengespeicherte Mechanismen bereitzustellen.
- Ereignisgesteuerte Architektur bewerten: Durch die Verwendung einer Serverless-Architektur können Sie auch eine ereignisgesteuerte Architektur für verteilte, auf Microservices basierende Anwendungen erstellen. So erhalten Sie skalierbare, resiliente, agile und kostengünstige Lösungen.

Ressourcen

Zugehörige Dokumente:

- AWS-Rechner für Gesamtbetriebskosten (TCO)
- AWS Serverless
- Was ist ereignisgesteuerte Architektur?
- Amazon S3-Speicherklassen
- Cloud-Produkte
- Amazon ElastiCache (Redis OSS)

Zugehörige Beispiele:

- Erste Schritte mit ereignisgesteuerter Architektur
- Ereignisorientierte Architektur
- Wie Statsig mit Amazon ElastiCache (Redis OSS) 100 Mal kosteneffizienter ausgeführt wird
- Bewährte Methoden für die Arbeit mit AWS Lambda-Funktionen

COST05-BP06 Durchführen einer Kostenanalyse für unterschiedliche Nutzungen im Lauf der Zeit

Workloads können sich im Laufe der Zeit ändern. Einige Services oder Funktionen sind auf unterschiedlichen Nutzungsebenen kostengünstiger. Wenn Sie jede Komponente im zeitlichen Verlauf und mit einer prognostizierten Nutzung analysieren, bleibt dieser Workload über seine gesamte Lebensdauer hinweg kostengünstig.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn AWS neue Services und Funktionen veröffentlicht, können sich die optimalen Services für Ihren Workload ändern. Der erforderliche Aufwand sollte potenzielle Vorteile widerspiegeln. Die Häufigkeit der Workload-Überprüfung hängt von den Anforderungen Ihres Unternehmens ab. Wenn es sich um einen Workload mit erheblichen Kosten handelt, wird die Implementierung neuer Services früher die Kosteneinsparungen maximieren, sodass eine häufigere Überprüfung von Vorteil sein kann. Ein weiterer Auslöser für die Überprüfung ist die Änderung der Nutzungsmuster. Signifikante Änderungen bei der Nutzung können darauf hinweisen, dass alternative Services optimaler wären.

Wenn Sie Daten in AWS Cloud verschieben müssen, können Sie aus einer Vielzahl von AWS-Services und Partnertools auswählen, die Sie bei der Migration Ihrer Datensätze unterstützen, ganz gleich, ob es sich um Dateien, Datenbanken, Computerabbilder, Block-Volumes oder sogar Bandsicherungen handelt. Wenn Sie zum Beispiel große Datenmengen zu und von AWS verschieben oder Daten am Edge verarbeiten möchten, können Sie eines der speziell entwickelten AWS-Geräte verwenden, um kostengünstig Petabytes an Daten offline zu verschieben. Bei höheren Datenübertragungsraten kann ein Direct Connect-Service beispielsweise günstiger als ein VPN sein und die erforderliche konsistente Konnektivität für Ihr Unternehmen bereitstellen.

Prüfen Sie Ihre Skalierungsaktivität basierend auf der Kostenanalyse für unterschiedliche Nutzungen im Laufe der Zeit. Analysieren Sie das Ergebnis, um herauszufinden, ob die Skalierungsrichtlinie so angepasst werden kann, dass Instances mit mehreren Instance-Typen und Kaufoptionen hinzugefügt

werden können. Überprüfen Sie Ihre Einstellungen, um zu sehen, ob das Minimum zur Verarbeitung von Benutzeranfragen reduziert werden kann (jedoch mit einer kleineren Flottengröße), und fügen Sie mehr Ressourcen hinzu, um die erwartete hohe Nachfrage zu erfüllen.

Führen Sie eine Kostenanalyse für unterschiedliche Nutzungen im Lauf der Zeit durch, indem Sie mit Stakeholdern in Ihrem Unternehmen sprechen und die Prognosefunktion von AWS Cost Explorer verwenden, um die potenziellen Auswirkungen von Serviceänderungen zu prognostizieren. Überwachen Sie Auslöser auf Nutzungsebene mithilfe von AWS Budgets, CloudWatch-Fakturierungsalarmen und AWS Cost Anomaly Detection, um die kosteneffektivsten Services früher zu identifizieren und zu implementieren.

Implementierungsschritte

- Definieren vorhergesagter Nutzungsmuster: Dokumentieren Sie in Zusammenarbeit mit Unternehmensbereichen, wie z. B. Marketing- und Produktbesitzern, wie die erwarteten und vorausgesagten Nutzungsmuster für die Verarbeitungslast aussehen werden. Sprechen Sie mit Business-Stakeholdern über historische und prognostizierte Kosten und gestiegene Nutzungen und stellen Sie sicher, dass solche Steigerungen mit den Geschäftsanforderungen übereinstimmen. Ermitteln Sie Kalendertage, -wochen oder -monate, in denen Sie mit einer erhöhten Nutzung Ihrer AWS-Ressourcen rechnen. Dies bedeutet, dass Sie die Kapazität der vorhandenen Ressourcen erhöhen oder zusätzliche Services einführen sollten, um die Kosten zu senken und die Leistung zu steigern.
- Durchführen einer Kostenanalyse bei vorhergesagter Nutzung: Führen Sie mithilfe der definierten Nutzungsmuster die Analyse an jedem dieser Punkte durch. Der Analyseaufwand sollte das potenzielle Ergebnis widerspiegeln. Wenn beispielsweise die Änderung der Nutzung groß ist, sollte eine gründliche Analyse durchgeführt werden, um etwaige Kosten und Änderungen zu überprüfen. Mit anderen Worten: Wenn die Kosten steigen, sollte auch die Nutzung für Unternehmen zunehmen.

Ressourcen

Zugehörige Dokumente:

- AWS-Gesamtbetriebskostenrechner (Total Cost of Ownership, TCO)
- Amazon S3-Speicherklassen
- Cloud-Produkte
- Amazon EC2 Auto Scaling

- Cloud-Datenmigration
- AWS Snow Family

Zugehörige Videos:

- AWS OpsHub for Snow Family

KOSTEN 6. Wie können Sie bei der Auswahl des Ressourcentyps, -umfangs und der Anzahl der Ressourcen Kostenziele erfüllen?

Stellen Sie sicher, dass Sie den geeigneten Ressourcenumfang und die Anzahl der Ressourcen für die jeweilige Aufgabe auswählen. Durch die Auswahl des kostengünstigsten Typs, Umfangs und der kostengünstigsten Anzahl minimieren Sie die Verschwendung von Ressourcen.

Bewährte Methoden

- COST06-BP01 Durchführen einer Kostenmodellierung
- COST06-BP02 Auswahl von Ressourcentyp, -größe und -anzahl basierend auf Daten
- COST06-BP03 Auswahl von Ressourcentyp, -umfang und -anzahl basierend auf Metriken
- COST06-BP04 Erwägen Sie die Verwendung gemeinsam genutzter Ressourcen

COST06-BP01 Durchführen einer Kostenmodellierung

Identifizieren Sie die Anforderungen des Unternehmens (z. B. Geschäftsanforderungen und bestehende Verpflichtungen) und führen Sie eine Kostenmodellierung (Gesamtkosten) des Workloads und aller seiner Komponenten durch. Führen Sie Benchmark-Aktivitäten für den Workload unter verschiedenen prognostizierten Belastungen durch und vergleichen Sie die Kosten. Der Modellierungsaufwand sollte in einem angemessenen Verhältnis zu dem potenziellen Nutzen stehen, z. B. muss der Zeitaufwand den Komponentenkosten entsprechen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Führen Sie eine Kostenmodellierung für Ihren Workload und jede ihrer Komponenten durch, um das Gleichgewicht zwischen Ressourcen zu verstehen und die richtige Größe für jede Ressource im Workload zu finden, unter Berücksichtigung eines bestimmten Leistungsgrads. Ein Verständnis der Kostenerwägungen kann den Geschäftsfall und die Entscheidungsfindung Ihres Unternehmens

bei der Bewertung der Ergebnisse der Wertrealisierung für die geplante Workload-Bereitstellung unterstützen.

Führen Sie Benchmark-Aktivitäten für den Workload unter verschiedenen prognostizierten Belastungen durch und vergleichen Sie die Kosten. Der Modellierungsaufwand sollte in einem angemessenen Verhältnis zu dem potenziellen Nutzen stehen, z. B. muss der Zeitaufwand proportional zu den Komponentenkosten oder prognostizierten Einsparungen sein. Die bewährten Methoden hierzu finden Sie im Abschnitt „Prüfverfahren“ des Whitepapers „Säule für Leistungseffizienz“ im AWS Well-Architected Framework.

Ein Beispiel: Zur Erstellung einer Kostenmodellierung für einen Workload, der aus Datenverarbeitungsressourcen besteht, kann AWS Compute Optimizer Sie bei der Kostenmodellierung für die Ausführung von Workloads unterstützen. Es bietet Empfehlungen zur richtigen Dimensionierung für Datenverarbeitungsressourcen basierend auf der bisherigen Nutzung. Stellen Sie sicher, dass CloudWatch-Agents in den Amazon EC2-Instances bereitgestellt wird, um Speichermetriken zu sammeln, die Ihnen helfen, genauere Empfehlungen innerhalb von AWS Compute Optimizer abzugeben. Dies ist die ideale Datenquelle für Datenverarbeitungsressourcen, da es sich um einen kostenlosen Service handelt,der Machine Learning nutzt, um je nach Risikograd mehrere Empfehlungen zu geben.

Es gibt mehrere Services, die Sie mit benutzerdefinierten Protokollen als Datenquellen für Dimensionierungen für andere Services und Workload-Komponenten verwenden können, wie AWS Trusted Advisor, Amazon CloudWatch und Amazon CloudWatch Logs. AWS Trusted Advisor prüft Ressourcen und kennzeichnet solche mit geringer Auslastung, was Ihnen helfen kann, Ihre Ressourcen richtig zu dimensionieren und ein Kostenmodell zu erstellen.

Im Folgenden finden Sie Empfehlungen für die Kostenmodellierung von Daten und Metriken:

- Die Überwachung muss die Benutzererfahrung genau widerspiegeln. Wählen Sie die richtige Detaillierung für die Dauer aus, und wählen Sie das Maximum oder den 99. Perzentil statt des Durchschnitts aus.
- Wählen Sie die richtige Aufschlüsselung für die Dauer der Analyse aus, die für die Deckung der Workload-Zyklen erforderlich ist. Bei einer zweiwöchigen Analyse könnten Sie beispielsweise einen monatlichen Zyklus mit hoher Nutzung übersehen, der zu einer Unterbereitstellung führen könnte.
- Wählen Sie die richtigen AWS-Services für Ihren geplanten Workload danach, wie Ihre bestehenden Verpflichtungen, ausgewählten Preismodelle für andere Workloads und die Fähigkeit, Innovationen schneller umzusetzen und sich auf Ihren Kerngeschäftswert zu konzentrieren, aussehen.

Implementierungsschritte

- Durchführen einer Kostenmodellierung: Stellen Sie den Workload oder einen Machbarkeitsnachweis in einem separaten Konto mit den spezifischen zu testenden Ressourcentypen und -umfängen bereit. Führen Sie den Workload mit den Testdaten aus und zeichnen die Ergebnisse zusammen mit den Kostendaten zum Zeitpunkt der Testausführung auf. Anschließend stellen Sie den Workload erneut bereit oder ändern die Ressourcentypen und -umfänge und führen den Test noch einmal aus. Fügen Sie die Lizenzgebühren für alle Produkte, die Sie möglicherweise mit diesen Ressourcen verwenden, sowie die geschätzten Betriebskosten (Arbeits- oder Ingenieurkosten) für die Bereitstellung und Verwaltung dieser Ressourcen bei der Erstellung der Kostenmodelle hinzu. Erwägen Sie eine Kostenmodellierung für einen bestimmten Zeitraum (stündlich, täglich, monatlich, jährlich oder drei Jahre).

Ressourcen

Zugehörige Dokumente:

- AWS Auto Scaling
- Ermittlung von Möglichkeiten zur richtigen Dimensionierung
- Amazon CloudWatch – Funktionen
- Kostenoptimierung: Richtige Amazon EC2-Dimensionierung
- AWS Compute Optimizer
- AWS-Preisrechner

Zugehörige Beispiele:

- Durchführen einer datengesteuerten Kostenmodellierung
- Schätzen der Kosten geplanter AWS-Ressourcenkonfigurationen
- Wählen der richtigen AWS-Tools

COST06-BP02 Auswahl von Ressourcentyp, -größe und -anzahl basierend auf Daten

Wählen Sie die Ressourcengröße oder den -typ basierend auf Daten zum Workload und der Ressourcenmerkmale aus. Zu berücksichtigen sind hier beispielsweise Datenverarbeitung, Speicher, Durchsatz oder Schreibintensität. Diese Auswahl erfolgt in der Regel unter Verwendung

einer früheren (On-Premises)-Version des Workloads, der Dokumentation oder anderer Informationsquellen über den Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Amazon EC2 bietet eine große Auswahl an Instance-Typen mit unterschiedlichen CPU-, Arbeitsspeicher-, Speicher- und Netzwerkkapazitäten für verschiedene Anwendungsfälle. Diese Instance-Typen bieten unterschiedliche Kombinationen von CPU-, Arbeitsspeicher-, Speicher- und Netzwerkkapazitäten, sodass Sie bei der Wahl der richtigen Ressourcenkombination für Ihre Projekte flexibel sind. Jeder Instance-Typ ist in mehreren Größen verfügbar, sodass Sie Ihre Ressourcen an die Anforderungen Ihres Workloads anpassen können. Um herauszufinden, welchen Instance-Typ Sie benötigen, informieren Sie sich über die Systemanforderungen der Anwendung oder Software, die Sie auf Ihrer Instance ausführen möchten. Diese Angaben sollten Folgendes umfassen:

- Betriebssystem
- Anzahl der CPU-Kerne
- GPU-Kerne
- Größe des Systemspeichers (RAM)
- Speichertyp und Umgebung
- Anforderung an die Netzwerkbandbreite

Ermitteln Sie den Zweck der Rechenanforderungen und welche Instance benötigt wird, um anschließend die verschiedenen Amazon EC2-Instance-Familien zu untersuchen. Amazon bietet die folgenden Instance-Typfamilien an:

- Allzweck
- Für die Datenverarbeitung optimiert
- Arbeitsspeicheroptimiert
- Speicheroptimiert
- Accelerated Computing
- HPC-optimiert

Für ein tiefergehendes Verständnis der jeweiligen Zwecke und Anwendungsfälle, die eine bestimmte Amazon EC2-Instance-Familie erfüllen kann, siehe AWS-Instance-Typen.

Die Erfassung der Systemanforderungen ist entscheidend, damit Sie die passende Instance-Familie und den geeigneten Instance-Typ für Ihre Anforderungen auswählen können. Die Namen der Instance-Typen setzen sich aus dem Familiennamen und der Größe der Instance zusammen. Die Instance t2.micro zum Beispiel gehört zur T2-Familie und entspricht der Micro-Größe.

Wählen Sie die Ressourcengröße oder den -typ basierend auf dem Workload und den Ressourcenmerkmalen aus (beispielsweise Datenverarbeitung, Speicher, Durchsatz oder Schreibintensität). Diese Auswahl erfolgt in der Regel unter Verwendung der Kostenmodellierung, einer früheren Version des Workloads (z. B. einer On-Premises-Version), mithilfe der Dokumentation oder unter Verwendung anderer Informationsquellen über den Workload (Whitepaper, veröffentlichte Lösungen). Die Verwendung von AWS Pricing Calculators oder Kostenmanagement-Tools kann dabei helfen, fundierte Entscheidungen über Instance-Typen, -Größen und -Konfigurationen zu treffen.

Implementierungsschritte

- Wählen Sie Ressourcen anhand von Daten aus: Verwenden Sie Ihre Kostenmodellierungsdaten, um den erwarteten Workload-Nutzungsgrad auszuwählen, und wählen Sie den angegebenen Ressourcentyp und die -größe aus. Bestimmen Sie auf Grundlage der Kostenmodellierungsdaten die Anzahl der virtuellen CPUs, den Gesamtspeicher (GiB), das lokale Speichervolumen der Instance (GB), die Amazon EBS-Volumes und das Leistungsniveau des Netzwerks unter Berücksichtigung der für die Instance erforderlichen Datenübertragungsrate. Treffen Sie Ihre Auswahl stets auf Grundlage detaillierter Analysen und genauer Daten, um die Leistung zu optimieren und gleichzeitig die Kosten effektiv zu verwalten.

Ressourcen

Zugehörige Dokumente:

- AWS-Instance-Typen
- AWS Auto Scaling
- Amazon CloudWatch – Funktionen
- Kostenoptimierung: Richtige EC2-Dimensionierung

Zugehörige Videos:

- Auswahl der richtigen Amazon EC2-Instance für Ihre Workloads

- Die richtige Dimensionierung Ihres Services

Zugehörige Beispiele:

- Es ist jetzt noch einfacher, Amazon EC2-Instance-Typen zu finden und zu vergleichen

COST06-BP03 Auswahl von Ressourcentyp, -umfang und -anzahl basierend auf Metriken

Nutzen Sie Metriken aus dem derzeit aktiven Workload für die Auswahl des richtigen Umfangs und Typs, um Kosten zu optimieren. Sorgen Sie für die richtige Bereitstellung von Durchsatz, Umfang und Speicher für Computing-, Speicher-, Daten- und Netzwerkservices. Dies kann mit einer Feedback-Schleife wie Auto Scaling oder durch benutzerdefinierten Code im Workload erfolgen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Erstellen Sie eine Feedback-Schleife innerhalb des Workloads, die aktive Metriken aus dem laufenden Workload verwendet, um Änderungen an diesem Workload vorzunehmen. Sie können einen verwalteten Service wie AWS Auto Scaling verwenden, den Sie so konfigurieren, dass er die richtigen Dimensionierungsvorgänge für Sie durchführt. AWS stellt außerdem APIs, SDKs und Funktionen bereit, mit denen Ressourcen mit minimalem Aufwand angepasst werden können. Sie können einen Workload so programmieren, dass eine Amazon EC2-Instance angehalten und gestartet wird, um eine Änderung der Instance-Größe oder des Instance-Typs zuzulassen. Dies bietet die Vorteile der richtigen Dimensionierung und eliminiert nahezu alle Betriebskosten, die für die Änderung erforderlich sind.

Einige AWS-Services verfügen über eine automatische Auswahl von Typ oder Größe, z. B. Amazon Simple Storage Service Intelligent-Tiering. Amazon S3 Intelligent-Tiering verschiebt Ihre Daten automatisch zwischen zwei Zugriffsebenen: Häufiger Zugriff und seltener Zugriff, basierend auf Ihren Nutzungsmustern.

Implementierungsschritte

- Steigern der Beobachtbarkeit durch Konfigurieren von Workload-Metriken: Erfassen Sie wichtige Metriken für den Workload. Diese Metriken geben die Kundenerfahrung an, z. B. die Workload-Ausgabe. Sie passen sich außerdem an die Unterschiede zwischen Ressourcentypen und -umfängen, z. B. CPU- und Speichernutzung, an. Analysieren Sie bei Computing-Ressourcen Leistungsdaten, um die Größe der Amazon EC2-Instances richtig zu bemessen. Ermitteln Sie

inaktive und nicht ausgelastete Instances. Schlüsselmetriken sind CPU- und Speicherauslastung (z. B. 40 % CPU-Auslastung in 90 % der Zeit, wie im Artikel zum Ermitteln der richtigen Dimensionierung, wenn AWS Compute Optimizer und die Arbeitsspeicherauslastung aktiviert sind, beschrieben). Ermitteln Sie Instances mit einer maximalen CPU- und Speicherauslastung von unter 40 % in einem Zeitraum von vier Wochen. Bei diesen Instances sollte die Größe angepasst werden, um die Kosten zu reduzieren. Bei Speicherressourcen wie Amazon S3 können Sie Amazon S3 Storage Lens verwenden. Hiermit sehen Sie standardmäßig 28 Metriken aus unterschiedlichen Kategorien auf Bucket-Ebene sowie historische Daten für 14 Tage im Dashboard. Sie können das Amazon S3 Storage Lens-Dashboard nach Übersichtswerten und Kostenoptimierung oder nach Ereignissen sortieren, um bestimmte Metriken zu analysieren.

- Anzeigen von Empfehlungen zur Umfangsanpassung: Anhand der Empfehlungen in AWS Compute Optimizer und dem Amazon EC2-Tool zur Umfangsanpassung in der Kostenverwaltungskonsole oder durch Prüfen der Umfangsanpassung für Ressourcen in AWS Trusted Advisor können Sie Anpassungen an Ihren Workloads vornehmen. Achten Sie darauf, die richtigen Tools zur Umfangsanpassung verschiedener Ressourcen zu verwenden, und halten Sie sich an die Richtlinien für die Dimensionierung, abhängig davon, ob es sich um eine Amazon EC2-Instance, AWS-Speicherklassen oder Amazon RDS-Instance-Typen handelt. Bei Speicherressourcen können Sie Amazon S3 Storage Lens verwenden. Hiermit erhalten Sie Einblicke in die Objektspeichernutzung und Aktivitätstrends und finden Empfehlungen zur Kostenoptimierung und zum Anwenden von bewährten Methoden zum Schutz der Daten. Anhand der kontextbezogenen Empfehlungen, die Amazon S3 Storage Lens aus der Analyse von Metriken in Ihrer Organisation ableitet, können Sie direkt Schritte zur Speicheroptimierung ergreifen.
- Automatische Auswahl des Ressourcentyps und des Umfangs basierend auf Metriken: Mithilfe der Workload-Metriken können Sie Ihre Workload-Ressourcen manuell oder automatisch auswählen. Bei Computing-Ressourcen kann die Konfiguration von AWS Auto Scaling oder die Implementierung von Code in Ihrer Anwendung den Aufwand reduzieren, der bei häufigen Änderungen erforderlich ist. So lassen sich Änderungen möglicherweise früher implementieren, als dies mit einem manuellen Prozess der Fall wäre. Mit nur einer Auto Scaling-Gruppe können Sie eine Flotte von On-Demand-Instances und Spot Instances starten und automatisch skalieren. Sie erhalten nicht nur Rabatte für Spot Instances, sondern können auch Reserved Instances oder einen Savings Plan nutzen, um ermäßigte Tarife gegenüber den normalen Preisen für On-Demand-Instances zu erhalten. Durch die Kombination dieser Faktoren sparen Sie Kosten für Amazon EC2-Instances und können die gewünschte Skalierung und Leistung für Ihre Anwendung festlegen. Sie können auch eine Strategie der attributbasierten Auswahl des Instance-Typs (ABS) in Auto Scaling Groups (ASG) einsetzen und so die Instance-Anforderungen in Form einer Gruppe von Attributen ausdrücken, z. B. vCPU, Arbeitsspeicher und Speicher. Mit Amazon

EC2 Spot Instances können Sie automatisch Instance-Typen neuerer Generationen verwenden, sobald sie veröffentlicht werden, und auf eine größeres Speicherangebot zugreifen. Amazon EC2 Fleet und Amazon EC2 Auto Scaling wählen Instances aus, die den angegebenen Attributen entsprechen, und starten diese. So müssen Sie Instance-Typen nicht mehr manuell auswählen. Bei Speicherressourcen können Sie die Funktionen Amazon S3 Intelligent-Tiering und Amazon EFS Infrequent Access nutzen. Hiermit werden automatisch die Speicherklassen ausgewählt, die automatisch zur Einsparung von Speicherkosten führen, wenn sich Datenzugriffsmuster ändern, ohne Leistungsbeeinträchtigungen oder Betriebsaufwand.

Ressourcen

Zugehörige Dokumente:

- AWS Auto Scaling
- AWS Right-Sizing (Größenanpassung in AWS)
- AWS Compute Optimizer
- Amazon CloudWatch – Funktionen
- Einrichten von CloudWatch
- CloudWatch: Veröffentlichen benutzerdefinierter Metriken
- Erste Schritte mit Amazon EC2 Auto Scaling
- Amazon S3 Storage Lens
- Amazon S3 Intelligent-Tiering
- Amazon EFS Infrequent Access
- Launch an Amazon EC2 Instance Using the SDK (Starten einer Amazon EC2-Instance mit SDK)

Zugehörige Videos:

- Right Size Your Services (Die richtige Dimensionierung Ihrer Services)

Zugehörige Beispiele:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet (Attributbasierte Auswahl des Instance-Typs für EC2 Auto Scaling und EC2 Fleet)
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling (Kostenoptimierung von Amazon Elastic Container Service mit geplanter Skalierung)

- Predictive scaling with Amazon EC2 Auto Scaling (Vorausschauende Skalierung mit Amazon EC2 Auto Scaling)
- Optimize Costs and Gain Visibility into Usage with Amazon S3 Storage Lens (Kostenoptimierung und Einblicke in die Auslastung mit Amazon S3 Storage Lens)
- Well-Architected Labs: Empfehlungen zur Dimensionierung (Stufe 100)
- Well-Architected Labs: Rightsizing with AWS Compute Optimizer and Memory Utilization Enabled (Level 200) (Größenanpassung, wenn Compute Optimizer und Speicherauslastung aktiviert sind)

COST06-BP04 Erwägen Sie die Verwendung gemeinsam genutzter Ressourcen

Für Services, die bereits auf Organisationsebene für mehrere Geschäftseinheiten bereitgestellt werden, sollten Sie die Verwendung gemeinsam genutzter Ressourcen erwägen, um die Auslastung zu erhöhen und die Gesamtbetriebskosten (TCO) zu senken. Die Verwendung gemeinsam genutzter Ressourcen kann eine kostengünstige Option sein, um Verwaltung und Kosten zu zentralisieren, indem bestehende Lösungen oder gemeinsam genutzte Komponenten oder beides verwendet werden. Verwalten Sie allgemeine Funktionen wie Überwachung, Backups und Konnektivität entweder innerhalb einer Kontogrenze oder in einem dedizierten Konto. Sie können auch die Kosten senken, indem Sie Standardisierung implementieren und Doppelarbeit sowie Komplexität reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wenn mehrere Workloads dieselbe Funktion ausführen, verwenden Sie vorhandene Lösungen und gemeinsam genutzte Komponenten, um Verwaltung und Kosten zu optimieren. Erwägen Sie die Nutzung vorhandener Ressourcen (insbesondere gemeinsam genutzter Ressourcen), z. B. Datenbankserver oder Verzeichnisservices, die nicht zur Produktion verwendet werden, um die Cloud-Kosten zu senken, indem Sie bewährte Sicherheitsmethoden und Organisationsvorschriften befolgen. Für eine optimale Wertschöpfung und Effizienz ist entscheidend, die Kosten (mithilfe von Kostenauflistung und Rückbuchung) den relevanten Geschäftsbereichen zuzuordnen, die den Konsum antreiben.

Kostenauflistung bezieht sich auf Berichte, in denen die Cloud-Kosten in zuteilbare Kategorien wie Verbraucher, Geschäftseinheiten, Hauptbuchkonten oder andere verantwortliche Entitäten unterteilt werden. Mit Kostenauflistungen sollen Teams, Geschäftseinheiten oder Einzelpersonen die Kosten ihrer verbrauchten Cloud-Ressourcen mitgeteilt werden.

Rückbuchung bedeutet, zentrale Serviceausgaben den Kostenträgern zuzuordnen, und zwar auf der Grundlage einer Strategie, die für einen bestimmten Finanzmanagementprozess geeignet ist. Für Kunden werden bei einer Rückbuchung die Kosten, die von einem Shared-Services-Konto anfallen, verschiedenen Finanzkostenkategorien zugeordnet, die für einen Kundenberichtsprozess geeignet sind. Durch die Einrichtung von Rückbuchungsmechanismen können Sie die Kosten melden, die verschiedenen Geschäftseinheiten, Produkten und Teams entstanden sind.

Workloads können als kritisch und unkritisch eingestuft werden. Verwenden Sie auf der Grundlage dieser Klassifizierung gemeinsam genutzte Ressourcen mit allgemeinen Konfigurationen für weniger kritische Workloads. Reservieren Sie dedizierte Server ausschließlich für kritische Workloads, um die Kosten weiter zu optimieren. Teilen Sie Ressourcen oder stellen Sie sie für mehrere Konten bereit, um sie effizient zu verwalten. Selbst in unterschiedlichen Entwicklungs-, Test- und Produktionsumgebungen ist eine sichere gemeinsame Nutzung möglich, ohne die Organisationsstruktur zu beeinträchtigen.

Verwenden Sie Daten zur Zuordnung geteilter Kosten, mit deren Hilfe Sie die Kosten einzelner Geschäftsentitäten basierend auf der Verwendung gemeinsam genutzter Computing- und Speicherressourcen durch die Anwendung zuordnen können, um Ihr Verständnis zu verbessern und die Kosten und Nutzung für containerisierte Anwendungen zu optimieren. Daten zur Zuordnung geteilter Kosten helfen Ihnen dabei, bei Container-Workloads, die auf Amazon Elastic Container Service (Amazon ECS) oder Amazon Elastic Kubernetes Service (Amazon EKS) ausgeführt werden, Kostenauflistung und Rückbuchung auf Aufgabenebene zu erreichen.

Erstellen Sie für verteilte Architekturen eine Shared-Services-VPC, die den zentralisierten Zugriff auf gemeinsam genutzte Services ermöglicht, die für Workloads in allen VPCs erforderlich sind. Diese gemeinsam genutzten Services können Ressourcen wie Verzeichnisservices oder VPC-Endpunkte umfassen. Zur Reduzierung des Verwaltungsaufwands und der Kosten empfiehlt die gemeinsame Nutzung von Ressourcen von einem zentralen Standort, anstatt sie in jeder einzelnen VPC zu erstellen.

Durch die Verwendung gemeinsam genutzter Ressourcen können Sie Betriebskosten sparen, die Ressourcenauslastung maximieren und die Konsistenz verbessern. In einem Design mit mehreren Konten können Sie einige AWS-Services zentral hosten und über mehrere Anwendungen und Konten an einem zentralen Punkt darauf zugreifen, um Kosten zu sparen. Sie können mit AWS Resource Access Manager (AWS RAM) weitere verbreitete Ressourcen freigeben, z. B. VPC-Subnetze und AWS Transit Gateway-Anhänge, AWS Network Firewall oder Amazon SageMaker AI-Pipelines. In einer Mehrkonten-Umgebung ermöglicht AWS RAM die einmalige Erstellung einer Ressource und ihre Freigabe für andere Konten.

Organisationen sollten die geteilten Kosten effektiv markieren und sicherstellen, dass kein erheblicher Teil ihrer Kosten unmarkiert oder nicht zugewiesen ist. Wenn Sie die gemeinsamen Kosten nicht effektiv verteilen und niemand die Verantwortung für die Verwaltung gemeinsamer übernimmt, können die Kosten für eine gemeinsame Cloud in die Höhe schießen. Sie müssen sich bewusst sein, wo Kosten auf Ressourcen-, Workload-, Team- oder Organisationsebene entstanden sind, da dieses Wissen Ihr Verständnis für den auf der jeweiligen Ebene geschaffenen Mehrwert im Vergleich zu den erzielten Geschäftsergebnissen verbessert. Letztlich profitieren Organisationen von Kosteneinsparungen, die sich aus der gemeinsamen Nutzung der Cloud-Infrastruktur ergeben. Fördern Sie die Kostenzuordnung für gemeinsam genutzte Cloud-Ressourcen, um die Cloud-Ausgaben zu optimieren.

Implementierungsschritte

- Vorhandene Ressourcen bewerten: Prüfen Sie bestehende Workloads, die ähnliche Services für Ihr Workload verwenden. Ziehen Sie abhängig von den Komponenten des Workloads vorhandene Plattformen in Betracht, sofern die Geschäftslogik oder die technischen Anforderungen dies zulassen.
- Gemeinsame Nutzung von Ressourcen in AWS RAM verwenden und entsprechend einschränken: Verwenden Sie AWS RAM, um Ressourcen mit anderen AWS-Konten innerhalb Ihrer Organisation zu teilen. Wenn Sie Ressourcen gemeinsam nutzen, müssen Sie Ressourcen nicht in mehreren Konten duplizieren, wodurch der betriebliche Aufwand der Ressourcenverwaltung minimiert wird. Dieser Prozess unterstützt die sichere Freigabe der Ressourcen, die Sie erstellt haben, an Rollen und Benutzer:innen in Ihrem Konto sowie an andere AWS-Konten.
- Ressourcen markieren: Markieren Sie Ressourcen, die für die Kostenberichterstattung infrage kommen, und kategorisieren Sie sie in Kostenkategorien. Aktivieren Sie diese kostenbezogenen Ressourcen-Tags für die Kostenzuordnung, um sich einen Überblick über den AWS-Ressourcenverbrauch zu verschaffen. Achten Sie darauf, ein angemessenes Maß an Granularität in Bezug auf Kosten- und Nutzungstransparenz zu schaffen, und beeinflussen Sie das Cloud-Nutzungsverhalten durch Kostenzuordnungsberichte und KPI-Tracking.

Ressourcen

Zugehörige bewährte Methoden:

- SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation

Zugehörige Dokumente:

- Was ist AWS Resource Access Manager?
- AWS-Services, die Sie mit AWS Organizations verwenden können
- Gemeinsam nutzbare AWS-Ressourcen
- AWS Kosten- und Nutzungsabfragen (CUR)

Zugehörige Videos:

- AWS Resource Access Manager – detaillierte Zugriffskontrolle mit verwalteten Berechtigungen
- So entwerfen Sie Ihre AWS Kostenzuordnungsstrategie
- AWS-Kostenkategorien

Zugehörige Beispiele:

- Rückbuchungen für gemeinsam verwendete Services: Ein AWS Transit Gateway-Beispiel
- Einrichtung eines Rückbuchungs-/Kostenauflistungsmodells für Savings Plans mithilfe von CUR
- Verwendung von VPC-Sharing für eine kostengünstige Microservice-Architektur mit mehreren Konten
- Verbesserte Kostentransparenz von Amazon EKS mit AWS-Daten zur Zuordnung geteilter Kosten
- Verbesserte Kostentransparenz von Amazon ECS und AWS Batch mit AWS-Daten zur Zuordnung geteilter Kosten

KOSTEN 7. Wie können Sie Kosten mithilfe von Preismodellen senken?

Verwenden Sie das Preismodell, das sich für Ihre Ressourcen am besten eignet. So halten Sie die Ausgaben möglichst niedrig.

Bewährte Methoden

- COST07-BP01 Durchführen einer Preismodellanalyse
- COST07-BP02 Auswählen von Regionen auf Basis der Kosten
- COST07-BP03 Auswahl von Drittanbietervereinbarungen mit kosteneffizienten Bedingungen
- COST07-BP04 Implementieren von Preismodellen für alle Komponenten dieses Workloads
- COST07-BP05 Durchführen einer Preismodellanalyse auf Verwaltungskontoebene

COST07-BP01 Durchführen einer Preismodellanalyse

Analysieren Sie die einzelnen Komponenten des Workloads. Stellen Sie fest, ob die Komponente und die Ressourcen über einen längeren Zeitraum (für Bindungsrabatte) oder dynamisch und kurz ausgeführt werden (für Spot- oder On-Demand-Zwecke). Analysieren Sie den Workload mithilfe der Empfehlungen in Tools für die Kostenverwaltung und wenden Sie Geschäftsregeln auf diese Empfehlungen an, um hohe Erträge zu erzielen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

AWS verfügt über mehrere Preismodelle, mit denen Sie für Ihre Ressourcen auf die kostengünstigste Art und Weise bezahlen können, die den Anforderungen Ihres Unternehmens entspricht und vom jeweiligen Produkt abhängt. Arbeiten Sie mit Ihren Teams zusammen, um das am besten geeignete Preismodell zu bestimmen. Häufig besteht das Preismodell aus einer Kombination aus verschiedenen Optionen, die sich nach Ihrer Verfügbarkeit richtet.

Im Fall von On-Demand-Instances zahlen Sie für die Datenverarbeitungs- oder Datenbankkapazitäten auf Stunden- oder Sekundenbasis (mindestens 60 Sekunden), abhängig von den Instances, die Sie ausführen. Es sind keine langfristigen Verpflichtungen oder Vorauszahlungen erforderlich.

Bei Savings Plans handelt es sich um ein flexibles Preismodell, das günstige Preise für die Nutzung von Amazon EC2, Lambda und AWS Fargate bietet. Im Gegenzug verpflichten Sie sich zu einer konstanten Nutzungsmenge (gemessen in Dollar/Stunde) für die Dauer von einem Jahr oder drei Jahren.

Spot Instances sind ein Preismechanismus für Amazon EC2, der es ermöglicht, ohne Vorabverpflichtungen freie Datenverarbeitungskapazität zu einem ermäßigten Stundensatz (bis zu 90 % Rabatt im Vergleich zum On-Demand-Preis) anzufordern.

Im Fall von Reserved Instances zahlen Sie im Voraus für die Kapazität und erhalten bis zu 75 Prozent Rabatt. Weitere Informationen finden Sie unter Optimierung der Kosten mit Reservierungen.

Sie könnten einen Savings Plans für die mit der Produktion, der Qualität und den Entwicklungsumgebungen verbundenen Ressourcen hinzufügen. Da Sandbox-Ressourcen nur bei Bedarf aktiviert werden, könnten Sie alternativ ein On-Demand-Modell für die Ressourcen in dieser Umgebung wählen. Verwenden Sie Spot Instances von Amazon, um die Kosten für Amazon EC2 zu senken, oder verwenden Sie Compute Savings Plans, um die Kosten für Amazon EC2, Fargate

und Lambda zu reduzieren. Das Empfehlungstool AWS Cost Explorer stellt Möglichkeiten für an feste Kapazität gebundene Rabatte mit Savings Plans vor.

Wenn Sie in der Vergangenheit bereits Reserved Instances für Amazon EC2 erworben oder in Ihrem Unternehmen Verfahren zur Kostenzuordnung eingeführt haben, können Sie Amazon EC2 Reserved Instances vorerst weiterhin verwenden. Wir empfehlen jedoch, eine Strategie für die zukünftige Verwendung von Savings Plans als flexibleren Mechanismus zur Kostenreduzierung zu entwickeln. Sie können die Empfehlungen zu Savings Plans (SP) in AWS Cost Management jederzeit aktualisieren, um neue Empfehlungen zu Savings Plans zu generieren. Verwenden Sie Reserved Instances (RI), um die Kosten für Amazon RDS, Amazon Redshift, Amazon ElastiCache und Amazon OpenSearch Service zu reduzieren. Es stehen drei Optionen für Savings Plans und Reserved Instances zur Verfügung: vollständige Vorauszahlung, teilweise Vorauszahlung und keine Vorauszahlung. Nutzen Sie die in AWS Cost Explorer bereitgestellten Kaufempfehlungen für RI und SP.

Um Möglichkeiten für Spot-Workloads zu finden, verwenden Sie eine stündliche Ansicht Ihrer Gesamtnutzung und suchen Sie nach regelmäßigen Zeiträumen mit sich ändernder Nutzung oder Elastizität. Sie können Spot Instances für verschiedene fehlertolerante und flexible Anwendungen verwenden. Beispiele sind statuslose Webserver, API-Endpunkte, Big-Data- und Analyseanwendungen, containerisierte Workloads, CI/CD und weitere flexible Workloads.

Ermitteln Sie, ob Ihre Amazon EC2- und Amazon RDS-Instances deaktiviert werden können, wenn sie nicht genutzt werden (nach Geschäftsschluss und am Wochenende). Dadurch können Sie die Kosten verglichen mit einem Einsatz rund um die Uhr um 70 % oder mehr reduzieren. Wenn Sie über Amazon Redshift-Cluster verfügen, die nur zu bestimmten Zeiten verfügbar sein müssen, können Sie den Cluster anhalten und zu einem späteren Zeitpunkt neu starten. Wenn der Amazon Redshift-Cluster oder die Amazon EC2- und Amazon RDS-Instances beendet werden, fallen keine Datenverarbeitungskosten mehr, sondern nur noch die Speichergebühren an.

Beachten Sie, dass es sich bei On-Demand-Kapazitätsreservierungen (ODCR) nicht um einen Preisnachlass handelt. Kapazitätsreservierungen werden zum entsprechenden On-Demand-Tarif in Rechnung gestellt, unabhängig davon, ob Sie Instances in reservierter Kapazität ausführen oder nicht. Sie sollten in Betracht gezogen werden, wenn Sie ausreichend Kapazität für die Ressourcen bereitstellen müssen, die Sie ausführen möchten. ODCRs müssen nicht an langfristige Verpflichtungen gebunden sein. Sie können gekündigt werden, wenn Sie sie nicht mehr benötigen. Sie können jedoch auch von den Rabatten profitieren, die Savings Plans oder Reserved Instances bieten.

Implementierungsschritte

- Analysieren der Workload-Elastizität: Verwenden Sie die stündliche Granularität im Cost Explorer oder ein benutzerdefiniertes Dashboard, um die Elastizität Ihres Workloads zu analysieren. Suchen Sie nach regelmäßigen Änderungen hinsichtlich der Anzahl der Instances, die ausgeführt werden. Instances mit kurzer Dauer sind Kandidaten für Spot Instances oder Spot Fleet.
  - Well-Architected Lab: Cost Explorer
  - Well-Architected Lab: Cost Visualization (Well-Architected Lab: Kostenvisualisierung)
- Überprüfen bestehender Preisverträge: Überprüfen Sie laufende Verträge oder Verpflichtungen für langfristige Anforderungen. Analysieren Sie, was Sie aktuell haben und inwiefern diese Verpflichtungen genutzt werden. Nutzen Sie bereits vorhandene vertragliche Rabatte oder Unternehmensverträge. Unternehmensverträge bieten den Kunden die Möglichkeit, die Vereinbarungen optimal an ihre Anforderungen anzupassen. Ziehen Sie bei langfristigen Verpflichtungen reservierte Preisrabatte, Reserved Instances oder Savings Plans für den spezifischen Instance-Typ, die Instance-Familie, AWS-Region und Availability Zones in Betracht.
- Durchführen einer Analyse des Bindungsrabatts: Sehen Sie sich unter Verwendung des Cost Explorer in Ihrem Konto die Empfehlungen für Savings Plans und Reserved Instances an. Um sicherzustellen, dass Sie die richtigen Empfehlungen mit den erforderlichen Rabatten und Risiken implementieren, befolgen Sie die Well-Architected Labs.

Ressourcen

Zugehörige Dokumente:

- Zugreifen auf Empfehlungen für Reserved Instances
- Instance-Kaufoptionen
- AWS Enterprise

Zugehörige Videos:

- Einsparen von bis zu 90 % und Ausführen der Produktions-Workloads mit Spot

Zugehörige Beispiele:

- Well-Architected Lab: Cost Explorer
- Well-Architected Lab: Cost Visualization (Well-Architected Lab: Kostenvisualisierung)
- Well-Architected Lab: Pricing Models (Well-Architected Lab: Preismodelle)

COST07-BP02 Auswählen von Regionen auf Basis der Kosten

Die Ressourcenpreise können je nach Region abweichen. Ermitteln Sie regionale Kostenunterschiede und stellen Sie nur in Regionen mit höheren Kosten bereit, um die Anforderungen an Latenzzeiten, Datenresilienz und Datensouveränität zu erfüllen. Die Berücksichtigung der Regionskosten sorgt dafür, dass Sie den niedrigsten Gesamtpreis für diesen Workload zahlen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Die AWS Cloud-Infrastruktur ist global, wird an mehreren Standorten weltweitgehostet und basiert auf AWS-Regionen, Availability Zones, Local Zones, AWS Outposts und Wavelength Zones. Eine Region ist ein physischer Ort auf der Welt. Jede Region ist ein separates geografisches Gebiet, in dem AWS mehrere Availability Zones hat. Availability Zones sind mehrere isolierte Standorte innerhalb jeder Region. Sie bestehen aus mindestens einem eigenständigen Rechenzentrum mit einer redundanten Stromversorgung, einem Netzwerk sowie Konnektivität.

Jede AWS-Region wird im Rahmen der jeweilig gültigen lokalen Marktbedingungen betrieben, und die Ressourcenpreise können von Region zu Region variieren, da es beispielsweise Unterschiede bei den Kosten für Land, Glasfaser, Strom und bei den Steuern gibt. Wählen Sie eine spezifische Region aus, in der Sie eine Komponente oder Ihre gesamte Lösung ausführen möchten, sodass Sie weltweit einen Betrieb zu den geringstmöglichen Kosten gewährleisten. Mithilfe des AWS-Rechners können Sie die Kosten Ihres Workloads in verschiedenen Regionen einschätzen. Suchen Sie dazu Services nach Standorttyp (Region, Wavelength Zone und Local Zone) und Region.

Wenn Sie die Architektur Ihrer Lösungen aufbauen, hat es sich bewährt zu versuchen, Computing-Ressourcen zugunsten einer geringeren Latenz und einer stärkeren Datensouveränität näher an die Benutzer zu bringen. Wählen Sie den geografischen Standort auf der Grundlage Ihrer Geschäfts-, Datenschutz-, Leistungs- und Sicherheitsanforderungen. Verwenden Sie für Anwendungen mit globalen Endbenutzern mehrere Standorte.

Nutzen Sie Regionen, die niedrigere Preise für AWS-Services anbieten, um Ihre Workloads bereitzustellen, wenn Sie keine Verpflichtungen in Bezug auf Datenschutz, Sicherheit und geschäftliche Anforderungen haben. Wenn Ihre Standardregion zum Beispiel ap-southeasth-2 (Sydney) ist und es keine Einschränkungen (z. B. Datenschutz, Sicherheit) für die Verwendung anderer Regionen gibt, ist die Bereitstellung nicht kritischer Amazon EC2-Instances (Entwicklung und Test) in der Region north-east-1 (Nord-Virginia) kostengünstiger.

	Compliance	*Latenz*	*Kosten*	*Services/Funktionen*
Region 1	✓	15 ms	$$	✓
Region 2	✓	20 ms	$$$	X
Region 3	✓	80 ms	$	✓
Region 4	✓	15 ms	$$	✓
Region 5	✓	20 ms	$$$	X
Region 6	✓	15 ms	$	✓
Region 7	✓	80 ms	$	✓
Region 8	✓	15 ms	$	X

Matrixtabelle für Regionsfunktionen

Die obige Matrixtabelle zeigt uns, dass Region 4 die beste Option für dieses gegebene Szenario ist, da die Latenz im Vergleich zu anderen Regionen gering ist, der Service verfügbar ist und es sich um die kostengünstigste Region handelt.

Implementierungsschritte

- Überprüfen der AWS-Region-Preise: Analysieren Sie die Workload-Kosten in der aktuellen Region. Berechnen Sie die Kosten in anderen verfügbaren Regionen, beginnend mit den höchsten Kosten nach Service und Verwendungstyp. Migrieren Sie in die neue Region, wenn die prognostizierte Einsparung die Kosten für das Verschieben der Komponente oder des Workloads überwiegt.
- Überprüfen der Anforderungen für Multi-Region-Bereitstellungen: Analysieren Sie Ihre geschäftlichen Anforderungen und Verpflichtungen (Datenschutz, Sicherheit oder Leistung), um herauszufinden, ob für Sie Beschränkungen gelten, sodass Sie nicht mehrere Regionen verwenden können. Wenn Sie sich nicht auf eine einzelne Region beschränken müssen, verwenden Sie mehrere Regionen.
- Analysieren der erforderlichen Datenübertragungen: Berücksichtigen Sie bei der Auswahl von Regionen die Datenübertragungskosten. Halten Sie Ihre Daten in der Nähe des Kunden und in der Nähe der Ressourcen. Wählen Sie weniger kostenintensive AWS-Regionen, in denen ein Datenfluss und nur minimale Datenübertragung besteht. Abhängig von Ihren

Geschäftsanforderungen für die Datenübertragung können Sie Amazon CloudFront, AWS PrivateLink, AWS Direct Connectund AWS Virtual Private Network verwenden, um Ihre Netzwerkkosten zu senken, die Leistung zu verbessern und die Sicherheit zu erhöhen.

Ressourcen

Zugehörige Dokumente:

- Zugreifen auf Empfehlungen für Reserved Instances
- Amazon EC2-Preise
- Kaufoptionen für Instances
- Tabelle „Region“

Zugehörige Videos:

- Einsparen von bis zu 90 % und Ausführen der Produktions-Workloads mit Spot

Zugehörige Beispiele:

- Überblick über die Datenübertragungskosten für gängige Architekturen
- Kostenerwägungen für globale Bereitstellungen
- „Relevante Aspekte bei der Wahl einer Region für Ihre Workloads“ erläutert
- Well-Architected Labs: Beschränken der Servicenutzung nach Region (Stufe 200)

COST07-BP03 Auswahl von Drittanbietervereinbarungen mit kosteneffizienten Bedingungen

Kosteneffiziente Vereinbarungen und Bedingungen stellen sicher, dass die Kosten dieser Services mit den von ihnen bereitgestellten Vorteilen skaliert werden. Wählen Sie Vereinbarungen und Preise aus, die skaliert werden, wenn sie Ihrem Unternehmen zusätzliche Vorteile bieten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Es gibt mehrere Produkte auf dem Markt, die Ihnen helfen, die Kosten für Ihre Cloud-Umgebungen zu verwalten. Sie unterscheiden sich teilweise in Bezug auf die Features, die von den Bedürfnissen der Kunden abhängen. So konzentrieren sich einige auf die Kostenkontrolle oder Kostentransparenz

und andere auf die Kostenoptimierung. Ein Schlüsselfaktor für eine effektive Kostenoptimierung und Governance ist die Verwendung des richtigen Tools mit den erforderlichen Features und dem richtigen Preismodell. Diese Produkte unterscheiden sich in ihren Preismodellen. Bei manchen wird ein bestimmter Prozentsatz Ihrer monatlichen Rechnung berechnet, bei anderen ein Prozentsatz Ihrer erzielten Einsparungen. Im Idealfall sollten Sie nur für das bezahlen, was Sie benötigen.

Wenn Sie Lösungen oder Services von Drittanbietern in der Cloud nutzen, ist es wichtig, dass die Preisstrukturen an Ihren gewünschten Ergebnissen ausgerichtet sind. Die Preise sollten mit den Ergebnissen und dem Wert skaliert werden, den sie bieten. Beispielsweise kostet Software, deren Preis auf einem Prozentsatz der erzielten Einsparungen basiert, umso mehr, je mehr Sie sparen (Ergebnis). Lizenzvereinbarungen, bei denen Sie mit steigenden Ausgaben mehr bezahlen, sind möglicherweise nicht immer in Ihrem Interesse, um die Kosten zu optimieren. Wenn der Anbieter jedoch klare Vorteile für alle Bestandteile Ihrer Rechnung bietet, könnte diese Preisstaffelung gerechtfertigt sein.

So kann beispielsweise eine Lösung, die Empfehlungen für Amazon EC2 bereitstellt und einen Prozentsatz Ihrer gesamten Rechnung berechnet, teurer werden, wenn Sie andere Services nutzen, die für Sie keinen Vorteil bieten. Ein weiteres Beispiel ist ein verwalteter Service, der zu einem Prozentsatz der Kosten für verwaltete Ressourcen in Rechnung gestellt wird. Eine höhere Instance-Größe erfordert möglicherweise nicht notwendigerweise mehr Verwaltungsaufwand, kann aber teurer werden. Stellen Sie sicher, dass diese Service-Preisvereinbarungen ein Kostenoptimierungsprogramm oder entsprechende Features in ihrem Service enthalten, um die Effizienz zu steigern.

Die Kunden finden diese auf dem Markt befindlichen Produkte vielleicht fortschrittlicher oder benutzerfreundlicher. Sie müssen die Kosten für diese Produkte berücksichtigen und über mögliche langfristige Kostenoptimierungen nachdenken.

Implementierungsschritte

- Analyse von Vereinbarungen und Bedingungen Dritter: Überprüfen Sie die Preise in Drittanbietervereinbarungen. Führen Sie die Modellierung für verschiedene Nutzungsebenen durch und berücksichtigen Sie neue Kosten, wie z. B. die Nutzung neuer Services oder Erweiterungen der aktuellen Services aufgrund des Workload-Wachstums. Entscheiden Sie, ob die zusätzlichen Kosten Ihrem Unternehmen die erforderlichen Vorteile bieten.

Ressourcen

Zugehörige Dokumente:

- Zugreifen auf Empfehlungen für Reserved Instances
- Kaufoptionen für Instances

Zugehörige Videos:

- Einsparen von bis zu 90 % und Ausführen der Produktions-Workloads mit Spot

COST07-BP04 Implementieren von Preismodellen für alle Komponenten dieses Workloads

Dauerhaft ausgeführte Ressourcen sollten reservierte Kapazität wie Savings Plans oder Reserved Instances nutzen. Die kurzfristige Kapazität wird für die Verwendung von Spot Instances oder einer Spot-Flotte konfiguriert. On-Demand-Instances werden nur für kurzfristige Workloads verwendet, die nicht unterbrochen werden können und nicht lange genug für reservierte Kapazitäten ausgeführt werden – typischerweise 25 bis 75 % des Zeitraums, je nach Ressourcentyp.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

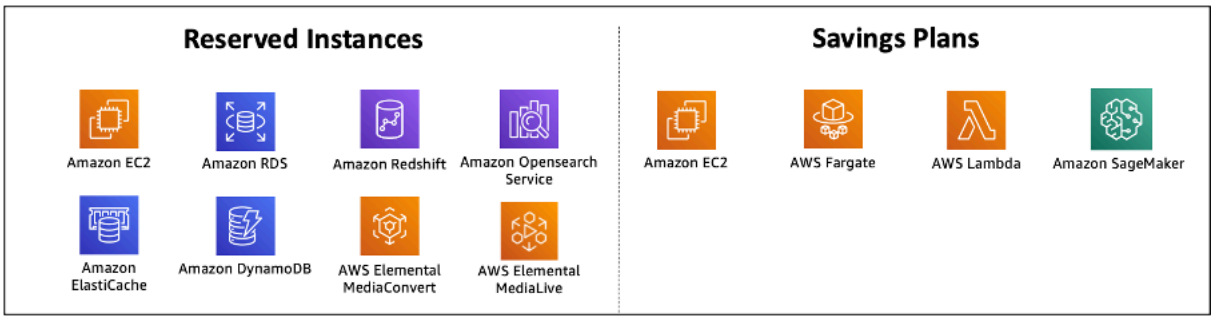
Implementierungsleitfaden

Um die Kosteneffizienz zu verbessern, bietet AWS mehrere Empfehlungen für Verpflichtungen auf Grundlage Ihrer bisherigen Nutzung an. Anhand dieser Empfehlungen können Sie nachvollziehen, was Sie einsparen können und wie die Verpflichtung verwendet wird. Sie können diese Services als On-Demand- oder Spot-Konfiguration nutzen oder sich für einen bestimmten Zeitraum verpflichten und Ihre On-Demand-Kosten mithilfe von Reserved Instances (RIs) und Savings Plans (SPs) reduzieren. Zur Optimierung Ihres Workloads müssen Sie nicht nur die einzelnen Workload-Komponenten und die verschiedenen AWS-Services berücksichtigen, sondern auch die Bindungsrabatte, Kaufoptionen und Spot Instances für diese Services.

Beachten Sie die Anforderungen der jeweiligen Workload-Komponenten sowie die verschiedenen Preismodelle für diese Services. Definieren Sie die Verfügbarkeitsanforderungen dieser Komponenten. Stellen Sie fest, ob mehrere unabhängige Ressourcen vorhanden sind, die die Funktion im Workload ausführen, und welche Workload-Anforderungen im Laufe der Zeit gelten. Vergleichen Sie die Kosten der Ressourcen unter Verwendung des standardmäßigen On-Demand-Preismodells und anderer anwendbarer Modelle. Beziehen Sie potenzielle Änderungen in Ressourcen oder Workload-Komponenten in Ihre Überlegungen ein.

Sehen wir uns zum Beispiel diese Webanwendungsarchitektur in AWS an. Dieser Beispiel-Workload besteht aus mehreren AWS-Services, wie z. B. Amazon Route 53, AWS WAF, Amazon CloudFront,

Amazon EC2-Instances, Amazon RDS-Instances, Load Balancers, Amazon S3-Speicher und Amazon Elastic File System (Amazon EFS). Sie müssen jeden dieser Services überprüfen und mögliche Kosteneinsparungen durch die verschiedenen Preismodelle ermitteln. Einige von ihnen können für RIs oder SPs in Frage kommen, während andere nur On-Demand verfügbar sind. Wie die folgende Abbildung zeigt, können einige der AWS-Services mithilfe von RIs oder SPs bereitgestellt werden.



AWS-Services, die über Reserved Instances und Savings Plans bereitgestellt werden

Implementierungsschritte

- Implementieren von Preismodellen: Kaufen Sie anhand Ihrer Analyseergebnisse Savings Plans, Reservierte Instances oder implementieren Sie Spot Instances. Wenn Sie sich zum ersten Mal verpflichten, wählen Sie die 5 oder 10 besten Empfehlungen aus der Liste aus und beobachten und analysieren Sie die Ergebnisse in den nächsten ein bis zwei Monaten. AWS Cost Management Console begleitet Sie durch den Prozess. Überprüfen Sie die RI- oder SP-Empfehlungen von der Konsole aus, passen Sie die Empfehlungen an (Typ, Zahlung und Laufzeit) und überprüfen Sie die stündliche Verpflichtung (z. B. 20 USD pro Stunde) und legen Sie sie dann in den Warenkorb. Die Rabatte gelten automatisch für die berechtigte Nutzung. Erwerben Sie in regelmäßigen Zyklen eine geringe Anzahl von Bindungsrabatten, (z. B. alle 2 Wochen oder monatlich). Implementieren Sie Spot Instances für Workloads, die unterbrochen werden können oder zustandslos sind. Wählen Sie anschließend On-Demand-Amazon EC2-Instances aus und weisen Sie Ressourcen für die verbleibenden Anforderungen zu.
- Workload-Überprüfungszyklus: Implementieren Sie einen Überprüfungszyklus für den Workload, der speziell die Abdeckung des Preismodells analysiert. Sobald der Workload den erforderlichen Umfang erreicht hat, können Sie teilweise (alle paar Monate) oder wenn sich die Nutzung Ihrer Organisation ändert, zusätzliche Bindungsrabatte erwerben.

Ressourcen

Zugehörige Dokumente:

- Die Empfehlungen zu Ihren Savings Plans verstehen
- Zugreifen auf Empfehlungen für Reserved Instances
- Erwerb von Reserved Instances
- Kaufoptionen für Instances
- Spot Instances
- Reservierungsmodelle für andere AWS-Services
- Unterstützte Savings Plans-Services

Zugehörige Videos:

- Einsparen von bis zu 90 % und Ausführen der Produktions-Workloads mit Spot

Zugehörige Beispiele:

- Was sollte ich vor dem Kauf eines Savings Plans beachten?
- Wie kann ich Cost Explorer verwenden, um meine Ausgaben und Nutzung zu verfolgen?

COST07-BP05 Durchführen einer Preismodellanalyse auf Verwaltungskontoebene

Prüfen Sie die Tools für die Fakturierung und Kostenverwaltung und informieren Sie sich über empfohlene Rabatte bei Bindung und Reservierungen, um regelmäßige Analysen auf Ebene des Verwaltungskontos auszuführen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Niedrig

Implementierungsleitfaden

Durch die regelmäßige Kostenmodellierung können Sie Möglichkeiten zur Optimierung über mehrere Workloads hinweg implementieren. Wenn beispielsweise mehrere Workloads On-Demand-Instances verwenden, ist das Änderungsrisiko insgesamt niedriger und die Nutzung eines auf fester Kapazität basierenden Rabatts kann zu niedrigeren Gesamtkosten führen. Es wird empfohlen, Analysen in regelmäßigen Zyklen von zwei Wochen bis zu einem Monat durchzuführen. Auf diese Weise können Sie kleine Anpassungskäufe tätigen, sodass sich die Abdeckung Ihrer Preismodelle mit Ihren sich ändernden Workloads und ihren Komponenten weiter entwickelt.

Verwenden Sie das AWS Cost Explorer -Empfehlungstool, um Möglichkeiten für an feste Kapazität gebundene Rabatte in Ihrem Verwaltungskonto zu finden. Empfehlungen auf der Ebene des

Verwaltungskontos werden unter Berücksichtigung der Nutzung aller Konten in Ihrer AWS-Organisation berechnet, die über Reserve Instances (RI) oder Savings Plans (SP) verfügen. Sie werden auch berechnet, wenn die Rabattteilung aktiviert ist, um eine Festlegung zu empfehlen, mit der die Ersparnisse auf allen Konten maximiert werden.

Beim Kauf auf Verwaltungskontoebene werden zwar in vielen Fällen maximale Einsparungen erzielt, es kann jedoch Situationen geben, in denen Sie den Kauf von SPs auf der verknüpften Kontoebene in Betracht ziehen könnten, z. B. wenn Sie möchten, dass die Rabatte zuerst für die Nutzung in diesem bestimmten verknüpften Konto gelten. Empfehlungen für Mitgliedskonten werden auf Ebene der einzelnen Konten berechnet, um die Einsparungen für das jeweilige Konto zu maximieren. Wenn Ihr Konto sowohl RI- als auch SP-Bindungen umfasst, werden diese in der folgenden Reihenfolge angewendet:

1. Zonen-RI
2. Standard-RI
3. Convertible RI
4. Instance Savings Plan
5. Compute Savings Plan

Wenn Sie einen SP auf Verwaltungskontoebene erwerben, werden die Einsparungen auf der Grundlage des höchsten bis niedrigsten Rabattprozentsatzes berechnet. SPs auf Verwaltungskontoebene überprüfen alle verknüpften Konten und wenden die Ersparnisse dort an, wo der Rabatt am höchsten ist. Wenn Sie einschränken möchten, wo die Ersparnisse verwendet werden, können Sie auf der verknüpften Kontoebene einen Savings Plan erwerben. Jedes Mal, wenn auf diesem Konto berechtigte Computing-Services ausgeführt werden, wird der Rabatt zuerst dort angewendet. Wenn auf dem Konto keine berechtigten Computing-Services ausgeführt werden, wird der Rabatt auf die anderen verknüpften Konten unter demselben Verwaltungskonto aufgeteilt. Die gemeinsame Nutzung von Rabatten ist standardmäßig aktiviert, kann aber bei Bedarf deaktiviert werden.

In einer konsolidierten Abrechnungsfamilie werden Savings Plans zuerst auf die Nutzung des Inhaberkontos und dann auf die Nutzung anderer Konten angewendet. Dies ist nur dann der Fall, wenn Sie das Teilen aktiviert haben. Ihre Savings Plans werden zuerst auf Ihren höchsten Sparprozentsatz angewendet. Wenn es mehrere Nutzungen mit denselben Sparprozentsätzen gibt, werden Savings Plans auf die erste Nutzung mit der niedrigsten Savings Plans-Rate angewendet. Savings Plans gelten so lange, bis keine Restnutzungen mehr zur Verfügung stehen oder Ihre Bindung ausgeschöpft ist. Jede verbleibende Nutzung wird zu den On-Demand-Tarifen abgerechnet.

Sie können die Empfehlungen zu Savings Plans im AWS-Kostenmanagement jederzeit aktualisieren, um neue Empfehlungen für Savings Plans zu generieren.

Nach der Analyse der Flexibilität der Instances können Sie sich entsprechend den Empfehlungen festlegen. Erstellen Sie eine Kostenmodellierung, indem Sie die kurzfristigen Kosten des Workloads mit möglichen verschiedenen Ressourcenoptionen analysieren und die AWS-Preismodelle analysieren und an Ihren geschäftlichen Anforderungen ausrichten, um die Gesamtbetriebskosten sowie die Kostenoptimierung zu ermitteln.

Implementierungsschritte

Durchführen einer Analyse des Bindungsrabatts: Sehen Sie sich unter Verwendung des Cost Explorer in Ihrem Konto die Empfehlungen für Savings Plans und Reserved Instances an. Stellen Sie sicher, dass Sie die Empfehlungen zu Savings Plans verstehen, und schätzen Sie Ihre monatlichen Ausgaben und Einsparungen. Sehen Sie sich die Empfehlungen auf Ebene des Verwaltungskontos an, die unter Berücksichtigung der Nutzung aller Mitgliedskonten in Ihrer AWS-Organisation berechnet werden, für die das Teilen der Rabatte für RI oder Savings Plans aktiviert ist, um maximale Einsparungen über alle Konten hinweg zu ermöglichen. Sie können sicherstellen, dass Sie die richtigen Empfehlungen mit den erforderlichen Rabatten und Risiken implementieren, indem Sie die Well-Architected Labs befolgen.

Ressourcen

Zugehörige Dokumente:

- Wie werden die Preise für AWS berechnet?
- Kaufoptionen für Instances
- Saving Plan Overview (Übersicht über Savings Plans)
- Saving Plan recommendations (Empfehlungen zu Savings Plans)
- Zugreifen auf Empfehlungen für Reserved Instances
- Die Empfehlungen zu Ihren Savings Plans verstehen
- How Savings Plans apply to your AWS usage (So wirken sich Savings Plans auf Ihre Nutzung von AWS aus)
- Saving Plans with Consolidated Billing (Savings Plans mit konsolidierter Fakturierung)
- Aktivieren des Teilens der Rabatte für Reserved Instances und Savings Plans

Zugehörige Videos:

- Einsparen von bis zu 90 % und Ausführen der Produktions-Workloads mit Spot

Zugehörige Beispiele:

- AWS Well-Architected Lab: Preismodelle (Stufe 200)
- AWS Well-Architected Labs: Preismodellanalyse (Stufe 200)
- Was sollte ich vor dem Kauf eines Savings Plan beachten?
- So lässt sich das Bindungsrisiko mit rollierenden Savings Plans verringern
- When to Use Spot-Instances (Wann Sie Spot-Instances verwenden sollten)

KOSTEN 8. Wie können Sie die Kosten für Datenübertragungen planen?

Damit Sie architekturbezogene Entscheidungen zur Kostenminimierung treffen können, müssen Sie unbedingt die Datenübertragungskosten einplanen und überwachen. Eine geringfügige, aber effektive Änderung an der Architektur kann Ihre Betriebskosten über einen längeren Zeitraum hinweg erheblich senken.

Bewährte Methoden

- COST08-BP01 Durchführen einer Datenübertragungsmodellierung
- COST08-BP02 Auswahl von Komponenten zur Optimierung der Datenübertragungskosten
- COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten

COST08-BP01 Durchführen einer Datenübertragungsmodellierung

Stellen Sie die Organisationsanforderungen zusammen und führen Sie eine Datenübertragungsmodellierung des Workloads und ihrer einzelnen Komponenten durch. Dadurch wird der niedrigste Kostenpunkt für die jeweiligen aktuellen Datenübertragungsanforderungen ermittelt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Wenn eine Lösung in der Cloud entworfen wird, werden die Gebühren für die Datenübertragung in der Regel vernachlässigt, da die Architektur üblicherweise in On-Premises-Rechenzentren entworfen wird oder es an Kenntnissen mangelt. Die Gebühren für die Datenübertragung in AWS

hängen von der Quelle, dem Ziel und dem Volumen des Datenverkehrs ab. Die Berücksichtigung dieser Gebühren in der Entwicklungsphase kann zu Kosteneinsparungen führen. Für eine genaue Einschätzung der Gesamtbetriebskosten (TCO) ist es sehr wichtig zu verstehen, wo die Datenübertragung in Ihrem Workload stattfindet, wie hoch die Kosten der Übertragung sind und welcher Nutzen damit verbunden ist. Auf diese Weise können Sie eine fundierte Entscheidung treffen, die Architekturentscheidung zu ändern oder zu akzeptieren. Sie können beispielsweise über eine Multi-Availability Zone-Konfiguration verfügen, in der Sie Daten zwischen den Availability Zones replizieren.

Sie modellieren die Komponenten der Services, die die Daten in Ihrem Workload übertragen, und entscheiden, dass dies akzeptable Kosten sind (ähnlich wie bei der Zahlung für Datenverarbeitung und Speicher in beiden Availability Zones), um die erforderliche Zuverlässigkeit und Ausfallsicherheit zu erreichen. Modellieren Sie die Kosten über verschiedene Nutzungsstufen. Die Workload-Nutzung kann sich im Laufe der Zeit ändern und verschiedene Services können auf verschiedenen Ebenen kostengünstiger sein.

Denken Sie bei der Modellierung Ihrer Datenübertragung daran, wie viele Daten aufgenommen werden und woher diese Daten stammen. Berücksichtigen Sie außerdem, wie viele Daten verarbeitet werden und wie viel Speicher- oder Rechnerkapazität benötigt wird. Befolgen Sie bei der Modellierung bewährte Methoden für Ihre Workload-Architektur, um Ihre potenziellen Datenübertragungskosten zu optimieren.

AWS -Preisrechner kann Ihnen helfen, die geschätzten Kosten für bestimmte AWS-Services und den erwarteten Datentransfer einzuschätzen. Wird bereits ein Workload ausgeführt (zu Testzwecken oder in einer Vorproduktionsumgebung), verwenden Sie AWS Cost Explorer oder AWS Cost and Usage Report (CUR), um Ihre Datenübertragungskosten zu verstehen und zu modellieren. Konfigurieren Sie einen Machbarkeitsnachweis (PoC) oder testen Sie Ihren Workload und führen Sie einen Test mit einer realistischen simulierten Last aus. Sie können Ihre Kosten bei verschiedenen Workload-Nachfragen modellieren.

Implementierungsschritte

- Ermitteln der Anforderungen: Was ist das primäre Ziel und was sind die geschäftlichen Anforderungen für die geplante Datenübertragung zwischen Quelle und Ziel? Was ist das erwartete Geschäftsergebnis am Ende? Ermitteln Sie die Geschäftsanforderungen und definieren Sie das erwartete Ergebnis.
- Ermitteln von Quelle und Ziel: Was ist die Datenquelle und das Ziel für die Datenübertragung, z. B. innerhalb der AWS-Regionen, in AWS Services oder ins Internet?

  - Datenübertragung innerhalb einer AWS-Region
  - Datenübertragung zwischen AWS-Regionen
  - Datenübertragung in das Internet
- Ermittlung der Datenklassifizierungen: Welche Datenklassifizierung gilt für diese Datenübertragung? Um welche Art von Daten handelt es sich? Um wie viele Daten handelt es sich? Wie häufig müssen die Daten übertragen werden? Handelt es sich um sensible Daten?
- Ermitteln der zu verwendenden AWS-Services oder -Tools: Welche AWS-Services werden für diese Datenübertragung verwendet? Ist es möglich, einen bereits bereitgestellten Service für einen anderen Workload zu verwenden?
- Berechnen der Datenübertragungskosten: Verwenden Sie AWS Pricing, die Datenübertragungsmodellierung, die Sie zuvor erstellt haben, um die Datenübertragungskosten für den Workload zu berechnen. Berechnen Sie die Datenübertragungskosten bei verschiedenen Nutzungsstufen, sowohl bei erhöhter als auch bei verringerter Workload-Nutzung. Wenn es mehrere Optionen für die Workload-Architektur gibt, berechnen Sie die Kosten für jede Option zum Vergleich.
- Verknüpfen der Kosten mit den Ergebnissen: Geben Sie für alle anfallenden Datenübertragungskosten das Ergebnis an, das damit für den Workload erreicht wird. Erfolgt der Transfer zwischen Komponenten, kann dies für die Entkopplung verwendet werden. Erfolgt der Transfer zwischen Availability Zones, kann dies zur Redundanz verwendet werden.
- Erstellen der Datenübertragungsmodellierung: Nachdem Sie alle Informationen zusammengetragen haben, erstellen Sie eine konzeptionelle Basisdatenübertragungsmodellierung für mehrere Anwendungsfälle und unterschiedliche Workloads.

Ressourcen

Zugehörige Dokumente:

- AWS-Caching-Lösungen
- AWS-Preise
- Amazon EC2-Preise
- Amazon VPC-Preise
- Grundlegendes zu den Gebühren für die Datenübertragung

Zugehörige Videos:

- Monitoring and Optimizing Your Data Transfer Costs (Datenübertragungskosten überwachen und optimieren)
- S3 Transfer Acceleration

Zugehörige Beispiele:

- Überblick über die Datenübertragungskosten für gängige Architekturen
- AWS Prescriptive Guidance für Netzwerke

COST08-BP02 Auswahl von Komponenten zur Optimierung der Datenübertragungskosten

Alle Komponenten sind ausgewählt und die Architektur ist so konzipiert, dass die Datenübertragungskosten gesenkt werden. Dies umfasst auch die Verwendung von Komponenten wie WAN-Optimierung und Multi-Availability-Zone-Konfigurationen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Eine Architektur für die Datenübertragung minimiert die Kosten für die Datenübertragung. Dies kann auch die Nutzung von CDNs bedeuten, um die Daten näher an den Benutzern zu platzieren, oder die Verwendung spezieller Netzwerk-Links von Ihrem Standort zu AWS. Sie können auch WAN-Optimierung und Anwendungsoptimierung verwenden, um die Datenmenge zu reduzieren, die zwischen Komponenten übertragen wird.

Bei der Übertragung von Daten zu oder innerhalb von AWS Cloud ist es wichtig, das Ziel auf der Grundlage der verschiedenen Anwendungsfälle, der Art der Daten und der verfügbaren Netzwerkressourcen zu kennen, um die richtigen AWS-Services zur Optimierung der Datenübertragung auszuwählen. AWS bietet eine Reihe von Datenübertragungsservices, die auf die verschiedenen Anforderungen der Datenmigration zugeschnitten sind. Wählen Sie die richtigen Optionen für die Datenspeicherung und Datenübertragung auf Grundlage der geschäftlichen Anforderungen in Ihrer Organisation.

Beachten Sie bei der Planung oder Überprüfung Ihrer Workload-Architektur die folgenden Punkte:

- Verwenden von VPC-Endpunkten in AWS: VPC-Endpunkte ermöglichen private Verbindungen zwischen Ihrer VPC und unterstützten AWS-Services. So vermeiden Sie die Nutzung des öffentlichen Internets, was zu Kosten für die Datenübertragung führen kann.

- Verwenden eines NAT-Gateways: Verwenden Sie ein NAT-Gateway, damit Instances in einem privaten Subnetz eine Verbindung zum Internet oder zu den Diensten außerhalb Ihrer VPC herstellen können. Überprüfen Sie, ob sich die Ressourcen hinter dem NAT-Gateway, die den meisten Datenverkehr senden, in derselben Availability Zone befinden wie das NAT-Gateway. Wenn dies nicht der Fall ist, erstellen Sie neue NAT-Gateways in derselben Availability Zone wie die Ressource, um die Gebühren für die Datenübertragung zwischen den Zonen zu reduzieren.
- Verwenden Sie AWS Direct Connect AWS Direct Connect, um das öffentliche Internet zu umgehen und eine direkte, private Verbindung zwischen Ihrem On-Premises-Netzwerk und AWS herzustellen. Dies kann kostengünstiger und konsistenter sein als die Übertragung großer Datenmengen über das Internet.
- Vermeidung von Datenübertragungen über regionale Grenzen hinweg: Bei Datenübertragungen zwischen AWS-Regionen (von einer Region in eine andere) fallen in der Regel Gebühren an. Die Entscheidung, einen multiregionalen Weg einzuschlagen, sollte gut überlegt sein. Weitere Informationen finden Sie unter Szenarien mit mehreren Regionen.
- Überwachen der Datenübertragung: Verwenden Sie Amazon CloudWatch und VPC Flow-Protokolle, um Einzelheiten über Ihre Datenübertragung und Netzwerknutzung zu erfassen. Analysieren Sie den erfassten Netzwerkverkehr in Ihren VPCs, z. B. die IP-Adresse oder den Bereich, der von und zu Netzwerkschnittstellen geht.
- Analysieren Ihrer Netzwerknutzung: Verwenden Sie Mess- und Berichtstools wie AWS Cost Explorer, CUDOS Dashboards oder CloudWatch, um die Datenübertragungskosten Ihres Workloads zu verstehen.

Implementierungsschritte

- Auswählen der Komponenten für die Datenübertragung: Konzentrieren Sie sich anhand der in COST08-BP01 Durchführen einer Datenübertragungsmodellierung erläuterten Datenübertragungsmodellierung darauf, wo die größten Datenübertragungskosten anfallen oder anfallen würden, wenn sich die Workload-Nutzung ändert. Suchen Sie nach alternativen Architekturen oder zusätzlichen Komponenten, die den Datenübertragungsbedarf beseitigen oder reduzieren (oder die Kosten senken).

Ressourcen

Zugehörige bewährte Methoden:

- COST08-BP01 Durchführen einer Datenübertragungsmodellierung

- COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten

Zugehörige Dokumente:

- Cloud-Datenmigration
- AWS-Caching-Lösungen
- Schnellere Bereitstellung von Inhalten mit Amazon CloudFront

Zugehörige Beispiele:

- Überblick über die Datenübertragungskosten für gängige Architekturen
- AWS-Tipps zur Netzwerkoptimierung
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format (Optimieren der Leistung und Reduzieren der Kosten für die Netzwerk-Analytik mit VPC Flow Logs im Apache Parquet-Format)

COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten

Implementieren Sie Services zur Verringerung der Datenübertragung. Verwenden Sie beispielsweise Edge-Standorte oder Content Delivery Networks (CDN), um Inhalte für Endbenutzer bereitzustellen, erstellen Sie Caching-Ebenen vor Ihren Anwendungsservern oder Datenbanken und verwenden Sie dedizierte Netzwerkverbindungen anstelle von VPNs für die Konnektivität zur Cloud.

Risikostufe bei fehlender Befolgung dieser Best Practice: Mittel

Implementierungsleitfaden

Es gibt verschiedene AWS-Services, mit denen Sie die Nutzung Ihrer Netzwerkdatenübertragung optimieren können. Abhängig von den Komponenten, dem Typ und der Cloud-Architektur Ihres Workloads können diese Services Sie bei der Komprimierung, beim Caching sowie der gemeinsamen Nutzung und Verteilung Ihres Datenverkehrs in der Cloud unterstützen.

- Amazon CloudFront ist ein weltweites Inhaltsbereitstellungsnetzwerk, das Daten bei niedriger Latenz und hohen Datenübertragungsgeschwindigkeiten bereitstellt. Es stellt Daten an Edge-Standorten rund um die Welt in den Cache und reduziert damit die Belastung Ihrer Ressourcen. Durch die Verwendung von CloudFront können Sie den administrativen Aufwand für die Bereitstellung von Inhalten für eine große Anzahl an Benutzern weltweit bei minimaler Latenz

reduzieren. Das Security Savings Bundle kann Ihnen dabei helfen, bis zu 30 % Ihrer CloudFront-Nutzung einzusparen, wenn Sie planen, Ihre Nutzung im Laufe der Zeit zu erhöhen.

- AWS Direct Connect ermöglicht es Ihnen, eine dedizierte Netzwerkverbindung zu AWS aufzubauen. Damit können Sie Netzwerkkosten reduzieren, die Bandbreite erhöhen und eine im Vergleich zu Internet-basierten Verbindungen gleichbleibendere Netzwerkerfahrung bieten.
- AWS VPN ermöglicht es Ihnen, eine sichere und private Verbindung zwischen Ihrem privaten Netzwerk und dem globalen AWS-Netzwerk herzustellen. Dies ist ideal für kleine Niederlassungen oder Geschäftspartner, da es vereinfachte Konnektivität bietet und ein vollständig verwalteter und elastischer Service ist.
- VPC-Endpunkte ermöglichen die Konnektivität zwischen AWS-Services über private Netzwerke und können verwendet werden, um Kosten für öffentliche Datenübertragungen und NAT Gateway zu reduzieren. Für Gateway-VPC-Endpunkte fallen keine stündlichen Gebühren an und sie unterstützen Amazon S3 und Amazon DynamoDB. Schnittstellen-VPC-Endpunkte werden von AWS PrivateLink bereitgestellt und für sie fällt eine Gebühr pro Stunde und Nutzungskosten pro GB an.
- NAT-Gateways bieten integrierte Skalierung und Verwaltung, wodurch die Kosten im Vergleich zu einer eigenständigen NAT-Instance reduziert werden. Platzieren Sie NAT-Gateways in denselben Availability Zones wie Instances mit hohem Datenverkehr und erwägen Sie die Verwendung von VPC-Endpunkten für die Instances, die auf Amazon DynamoDB oder Amazon S3 zugreifen müssen, um die Datenübertragungs- und Verarbeitungskosten zu senken.
- Verwenden Sie AWS Snow Family Geräte, die über Rechenressourcen zum Erfassen und Verarbeiten von Daten am Netzwerk-Edge verfügen. AWS Snow Family-Geräte (Snowball Edge, Snowball Edge und Snowmobile) ermöglichen es Ihnen, Petabytes an Daten kostengünstig und offline in die AWS Cloud zu verschieben.

Implementierungsschritte

- Services implementieren: Wählen Sie für Ihren Service und Workload-Typ anhand der Datenübertragungsmodellierung und der Informationen in den VPC-Flow-Protokollen die entsprechenden AWS-Netzwerk-Services aus. Sehen Sie sich an, wo sich die höchsten Kosten und Volumenströme befinden. Überprüfen Sie die AWS-Services und ermitteln Sie, ob es einen Service gibt, der die Übertragung reduziert oder entfernt, insbesondere die Netzwerk- und Inhaltsbereitstellung. Suchen Sie auch nach Caching-Services, bei denen wiederholt auf Daten oder große Datenmengen zugegriffen wird.

Ressourcen

Zugehörige Dokumente:

- AWS Direct Connect
- Unsere AWS-Produkte entdecken
- AWS-Caching-Lösungen
- Amazon CloudFront
- AWS Snow Family
- Amazon CloudFront Security Savings Bundle

Zugehörige Videos:

- Monitoring and Optimizing Your Data Transfer Costs (Datenübertragungskosten überwachen und optimieren)
- AWS-Serie zur Kostenoptimierung: CloudFront
- How can I reduce data transfer charges for my NAT gateway? (Wie lassen sich die Datenübertragungsgebühren für mein NAT-Gateway senken?)

Zugehörige Beispiele:

- Rückbuchungen für gemeinsam verwendete Services: Ein AWS Transit Gateway-Transit-Gateway-Beispiel
- Details zur AWS-Datenübertragung in Kosten- und Nutzungsberichten mit Athena-Abfragen und QuickSight verstehen
- Überblick über die Datenübertragungskosten für gängige Architekturen
- Analysieren der Datenübertragungskosten mit AWS Cost Explorer
- Kostenoptimierung Ihrer AWS-Architekturen durch die Nutzung von Amazon CloudFront-Funktionen
- How can I reduce data transfer charges for my NAT gateway? (Wie lassen sich die Datenübertragungsgebühren für mein NAT-Gateway senken?)

Verwaltung von Nachfrage und Bereitstellung von Ressourcen

Frage

- KOSTEN 9. Wie verwalten Sie die Nachfrage und stellen Ressourcen bereit?

KOSTEN 9. Wie verwalten Sie die Nachfrage und stellen Ressourcen bereit?

Stellen Sie bei einem Workload mit ausgewogenen Ausgaben und Leistungen sicher, dass alles, wofür Sie bezahlen, genutzt wird, und vermeiden Sie eine erhebliche Unterauslastung der Instances. Eine verschobene Auslastungsmetrik in einer der Richtungen wirkt sich nachteilig auf Ihr Unternehmen aus, entweder im Hinblick auf die Betriebskosten (verschlechterte Leistung aufgrund von Überbelegung) oder auf die verschwendeten AWS-Ausgaben (aufgrund von Überversorgung).

Bewährte Methoden

- COST09-BP01 Analyse des Workload-Bedarfs
- COST09-BP02 Implementieren eines Puffers oder einer Drosselung zur Bedarfsverwaltung
- COST09-BP03 Dynamische Bereitstellung von Ressourcen

COST09-BP01 Analyse des Workload-Bedarfs

Analysieren Sie den Bedarf des Workloads im gesamten Zeitverlauf. Stellen Sie sicher, dass die Analyse saisonale Trends berücksichtigt und die Betriebsbedingungen über die gesamte Lebensdauer des Workloads genau wiedergibt. Der Analyseaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen, z. B. muss der Zeitaufwand den Workload-Kosten entsprechen.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Die Analyse des Workload-Bedarfs für Cloud-Computing erfordert ein Verständnis der Muster und Eigenschaften von Computing-Aufgaben, die in der Cloud-Umgebung initiiert werden. Diese Analyse hilft Benutzern, die Ressourcenzuweisung zu optimieren, Kosten zu verwalten und sicherzustellen, dass die Leistung den Anforderungen entspricht.

Informieren Sie sich über die Anforderungen des Workloads. Die Anforderungen Ihrer Organisation sollten die Workload-Reaktionszeiten für Anforderungen angeben. Anhand der Reaktionszeit kann bestimmt werden, ob der Bedarf gut gesteuert wird oder ob das Ressourcenangebot geändert werden sollte, um der Nachfrage gerecht zu werden.

Die Analyse sollte die Vorhersehbarkeit und Wiederholbarkeit der Nachfrage, die Änderungsrate der Nachfrage und die Menge der Nachfrageänderungen umfassen. Die Analyse muss über einen

ausreichend langen Zeitraum ausgeführt werden, um saisonale Abweichungen wie Arbeiten am Ende des Monats oder Feiertagsspitzen einzubeziehen.

Der Analyseaufwand sollte die potenziellen Vorteile der Implementierung einer Skalierung widerspiegeln. Betrachten Sie die erwarteten Gesamtkosten der Komponente sowie etwaige Erhöhungen oder Verringerungen der Nutzung und der Kosten während der Workload-Lebensdauer.

Im Folgenden sind einige wichtige Aspekte aufgeführt, die bei der Durchführung der Workload-Bedarfsanalyse für Cloud-Computing zu berücksichtigen sind:

1. Ressourcennutzung und Leistungskennzahlen: Analysieren Sie, wie AWS-Ressourcen im Laufe der Zeit genutzt werden. Ermitteln Sie Nutzungsmuster während und außerhalb der Spitzenzeiten, um die Ressourcenzuweisung und Skalierungsstrategien zu optimieren. Überwachen Sie Leistungskennzahlen wie Reaktionszeiten, Latenz, Durchsatz und Fehlerraten. Diese Kennzahlen helfen bei der Bewertung des Gesamtzustands und der Effizienz der Cloud-Infrastruktur.
2. Skalierungsverhalten von Benutzern und Anwendungen: Verstehen Sie das Benutzerverhalten und wie es sich auf den Workload-Bedarf auswirkt. Das Untersuchen von Mustern beim Benutzerdatenverkehr trägt dazu bei, die Bereitstellung von Inhalten und die Reaktionsfähigkeit von Anwendungen zu verbessern. Analysieren Sie, wie Workloads mit steigender Nachfrage skaliert werden. Bestimmen Sie, ob die Parameter für die automatische Skalierung korrekt und effektiv für den Umgang mit Auslastungsschwankungen konfiguriert sind.
3. Workload-Typen: Identifizieren Sie die verschiedenen Arten von Workloads, die in der Cloud ausgeführt werden, wie Batch-Verarbeitung, Echtzeitdatenverarbeitung, Webanwendungen, Datenbanken oder Machine Learning. Jeder Workload-Typ kann unterschiedliche Ressourcenanforderungen und Leistungsprofile aufweisen.
4. Service Level Agreements (SLAs): Vergleichen Sie die tatsächliche Leistung mit den SLAs, um deren Einhaltung sicherzustellen und Bereiche zu identifizieren, die verbessert werden müssen.

Nutzen Sie Instrumentierungsservices wie Amazon CloudWatch zur Erfassung und Nachverfolgung von Metriken, Überwachung von Protokolldateien, Festlegung von Alarmen und automatischen Reaktionen auf Änderungen in den AWS-Ressourcen. Mit Amazon CloudWatch erhalten Sie außerdem einen systemweiten Einblick in die Auslastung Ihrer Ressourcen, die Anwendungsleistung sowie die Integrität Ihrer Betriebsabläufe.

Mit AWS Trusted Advisorkönnen Sie Ihre Ressourcen gemäß bewährten Methoden bereitstellen und so die Systemleistung und -zuverlässigkeit verbessern, die Sicherheit erhöhen und nach Einsparungsmöglichkeiten suchen. Darüber hinaus können Sie Nicht-Produktions-Instances

deaktivieren und Amazon CloudWatch und Auto Scaling verwenden, um auf Steigerungen und Reduzierungen des Bedarfs zu reagieren.

Schließlich können Sie AWS Cost Explorer oder QuickSight mit der AWS Cost and Usage Report(CUR)-Datei oder Ihren Anwendungsprotokollen verwenden, um eine erweiterte Analyse des Workload-Bedarfs durchzuführen.

Insgesamt ermöglicht eine umfassende Analyse des Workload-Bedarfs es Unternehmen, fundierte Entscheidungen zur Bereitstellung, Skalierung und Optimierung von Ressourcen zu treffen, was zu einer besseren Leistung, Kosteneffizienz und Benutzerzufriedenheit führt.

Implementierungsschritte

- Vorhandene Workload-Daten analysieren: Analysieren Sie Daten aus dem vorhandenen Workload, früheren Versionen des Workloads oder vorhergesagten Nutzungsmustern. Verwenden Sie Amazon CloudWatch, Protokolldateien und Überwachungsdaten, um einen Einblick in die Nutzung des Workloads zu erhalten. Analysieren Sie einen vollständigen Workload-Zyklus und erfassen Sie Daten für alle saisonalen Änderungen, z. B. Ereignisse am Monatsende oder am Ende des Jahres. Der in der Analyse reflektierte Aufwand sollte die Workload-Merkmale widerspiegeln. Der größte Aufwand sollte für hochwertige Workloads mit den größten Nachfrageänderungen betrieben werden. Der geringste Aufwand sollte für Workloads mit niedrigem Wert und geringfügigen Nachfrageänderungen betrieben werden.
- Vorhersage externer Einflüsse Treffen Sie Teammitglieder aus der gesamten Organisation, die die Nachfrage im Workload beeinflussen oder ändern können. Häufig betroffene Teams sind Vertrieb, Marketing oder Business Development. Arbeiten Sie mit ihnen zusammen, um die Zyklen kennenzulernen, in denen sie arbeiten, und um zu erfahren, ob es Ereignisse gibt, die die Nachfrage des Workloads ändern könnten. Erstellen Sie eine Prognose des Workload-Bedarfs anhand dieser Daten.

Ressourcen

Zugehörige Dokumente:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling

- Mit dem AWS Instance Scheduler
- Erste Schritte mit Amazon SQS
- AWS Cost Explorer
- QuickSight

Zugehörige Videos:

Zugehörige Beispiele:

- Monitor, Track and Analyze for cost optimization (Überwachung, Nachverfolgung und Analysen für die Kostenoptimierung)
- Suchen und Analysieren von Protokollen in CloudWatch

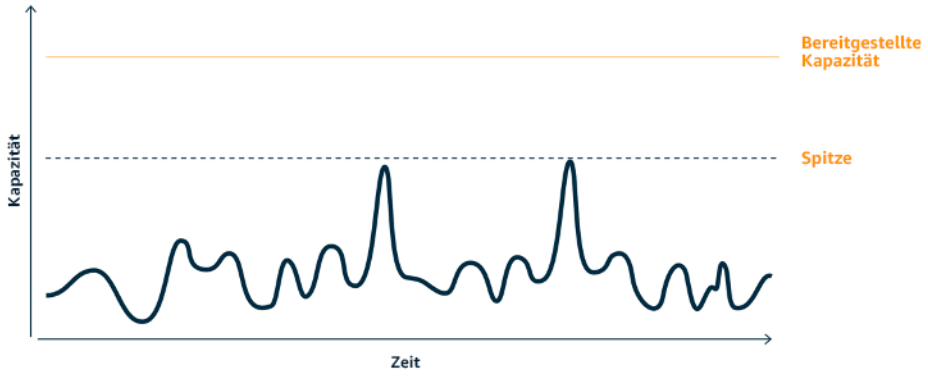
COST09-BP02 Implementieren eines Puffers oder einer Drosselung zur Bedarfsverwaltung

Pufferung und Drosselung ändern den Bedarf Ihres Workloads und glätten alle Spitzen. Implementieren Sie die Drosselung, wenn Ihre Clients Wiederholungen durchführen. Implementieren Sie die Pufferung, um die Anforderung zu speichern und die Verarbeitung auf einen späteren Zeitpunkt zu verschieben. Stellen Sie sicher, dass Ihre Drosselungen und Puffer so konzipiert sind, dass Clients in der erforderlichen Zeit eine Antwort erhalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Implementierung einer Pufferung oder Drosselung ist beim Cloud-Computing von entscheidender Bedeutung, um die Nachfrage zu steuern und die für den Workload benötigte bereitgestellte Kapazität zu reduzieren. Für eine optimale Leistung ist es unerlässlich, die Gesamtnachfrage, einschließlich der Spitzen, sowie die Geschwindigkeit, mit der sich die Anfragen ändern, und die erforderliche Reaktionszeit zu messen. Wenn Clients die Möglichkeit haben, ihre Anfragen erneut zu senden, ist es praktisch, eine Drosselung vorzunehmen. Umgekehrt ist für Clients ohne Wiederholungsfunktionen die Implementierung einer Pufferlösung der ideale Ansatz. Solche Puffer rationalisieren den Eingang von Anfragen und optimieren die Interaktion von Anwendungen mit unterschiedlichen Betriebsgeschwindigkeiten.



Bedarfskurve mit zwei deutlichen Spitzen, die hohe bereitgestellte Kapazität erfordern

Nehmen wir einen Workload mit der nachfolgend gezeigten Bedarfskurve. Dieser Workload hat zwei Spitzen und um damit umzugehen, wird die Ressourcenkapazität bereitgestellt, die hier durch die orangefarbene Linie angezeigt wird. Die für diesen Workload aufgewendeten Ressourcen und die eingesetzte Energie werden nicht durch die Fläche unter der Bedarfskurve, sondern von der Linie für die bereitgestellte Kapazität angezeigt, da die bereitgestellte Kapazität zur Bewältigung dieser beiden Spitzen benötigt wird. Die Verflachung der Bedarfskurve kann Ihnen dabei helfen, die bereitgestellte Kapazität für einen Workload zu verringern und dessen Umweltauswirkungen zu reduzieren. Um die Spitzen abzuflachen, sollten Sie eine Lösung zur Drosselung oder Pufferung in Betracht ziehen.

Um dies besser zu verstehen, werden wir uns kurz die Drosselung und Pufferung ansehen.

Drosselung: Wenn die Quelle der Nachfrage über eine Wiederholungsfunktion verfügt, können Sie die Drosselung implementieren. Die Drosselung teilt der Quelle mit, dass wenn sie die Anfrage zum aktuellen Zeitpunkt nicht bedienen kann, sie es später erneut versuchen sollte. Die Quelle wartet einen bestimmten Zeitraum und wiederholt die Anfrage. Die Implementierung der Drosselung hat den Vorteil, dass die maximale Menge an Ressourcen und Kosten des Workloads begrenzt wird. In AWS können Sie Amazon API Gateway verwenden, um die Drosselung zu implementieren.

Pufferbasiert: Ein pufferbasierter Ansatz verwendet Produzenten (Komponenten, die Nachrichten an die Warteschlange senden), Verbraucher (Komponenten, die Nachrichten aus der Warteschlange empfangen) und eine Warteschlange (die Nachrichten enthält), um die Nachrichten zu speichern. Nachrichten können dadurch von Verbrauchern in der für ihre Geschäftsanforderungen passenden Geschwindigkeit gelesen und verarbeitet werden. Durch die Verwendung einer pufferbasierten Methodik werden die Nachrichten von den Produzenten in Warteschlangen oder Streams gespeichert und können von den Verbrauchern in einem Tempo abgerufen werden, das sich an deren betrieblichen Anforderungen orientiert.

In AWS können Sie aus mehreren Services wählen, um einen Pufferungsansatz zu implementieren. Amazon Simple Queue Service (Amazon SQS) ist ein verwalteter Service, der Warteschlangen bereitstellt, die es einem einzelnen Verbraucher ermöglichen, einzelne Nachrichten zu lesen. Amazon Kinesis bietet einen Stream, der es vielen Verbrauchern ermöglicht, dieselben Nachrichten zu lesen.

Durch Pufferung und Drosselung können Spitzenwerte abgeflacht werden, indem die Anforderungen an Ihren Workload angepasst werden. Verwenden Sie die Drosselung, wenn Clients Aktionen wiederholen, und nutzen Sie die Pufferung, um Anfragen zurückzuhalten und später zu verarbeiten. Stellen Sie bei der Architektur mit einem pufferbasierten Ansatz sicher, dass Sie Ihren Workload so gestalten, dass er die Anfrage in der erforderlichen Zeit erfüllt, und dass Sie doppelte Arbeitsanfragen verarbeiten können. Analysieren Sie den Gesamtbedarf, die Änderungsrate und die erforderliche Reaktionszeit, um die korrekte Größe der erforderlichen Drosselung oder des Puffers zu bestimmen.

Implementierungsschritte

- Analysieren Sie die Client-Anfragen: Analysieren Sie die Client-Anfragen, um festzustellen, ob sie in der Lage sind, Wiederholungen durchzuführen. Für Clients, die keine Wiederholungen durchführen können, müssen Puffer implementiert werden. Analysieren Sie den Gesamtbedarf, die Änderungsrate und die erforderliche Reaktionszeit, um die Größe der erforderlichen Drosselung oder des Puffers zu bestimmen.
- Implementieren eines Puffers oder einer Drosselung: Implementieren Sie einen Puffer oder eine Drosselung im Workload. Eine Warteschlange wie Amazon Simple Queue Service (Amazon SQS) kann für Ihre Workload-Komponenten einen Puffer bereitstellen. Amazon API Gateway kann eine Drosselung für Ihre Workload-Komponenten bereitstellen.

Ressourcen

Zugehörige bewährte Methoden:

- SUS02-BP06 Implementierung von Pufferung oder Drosselung, um die Bedarfskurve zu verflachen
- REL05-BP02 Drosselung von Anfragen

Zugehörige Dokumente:

- AWS Auto Scaling
- AWS Instance Scheduler
- Amazon API Gateway

- Amazon Simple Queue Service
- Erste Schritte mit Amazon SQS
- Amazon Kinesis

Zugehörige Videos:

- Choosing the Right Messaging Service for Your Distributed App (Den richtigen Messaging-Service für Ihre verteilte App auswählen)

Zugehörige Beispiele:

- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- Drosselung einer mehrstufigen, Multi-Mandanten REST-API in großem Umfang mit API Gateway
- Aktivieren von Tiering und Drosselung in einer Amazon EKS-SaaS-Lösung mit mehreren Mandanten mithilfe von Amazon API Gateway
- Application integration Using Queues and Messages (Anwendungsintegration mit Warteschlangen und Nachrichten)

COST09-BP03 Dynamische Bereitstellung von Ressourcen

Ressourcen werden geplant bereitgestellt. Dies kann bedarfsbasiert sein, z. B. durch Auto Scaling, oder zeitbasiert, wobei der Bedarf vorhersehbar ist und Ressourcen basierend auf der Zeit bereitgestellt werden. Diese Methoden führen zur geringsten Anzahl an Über- oder Unterversorgungen.

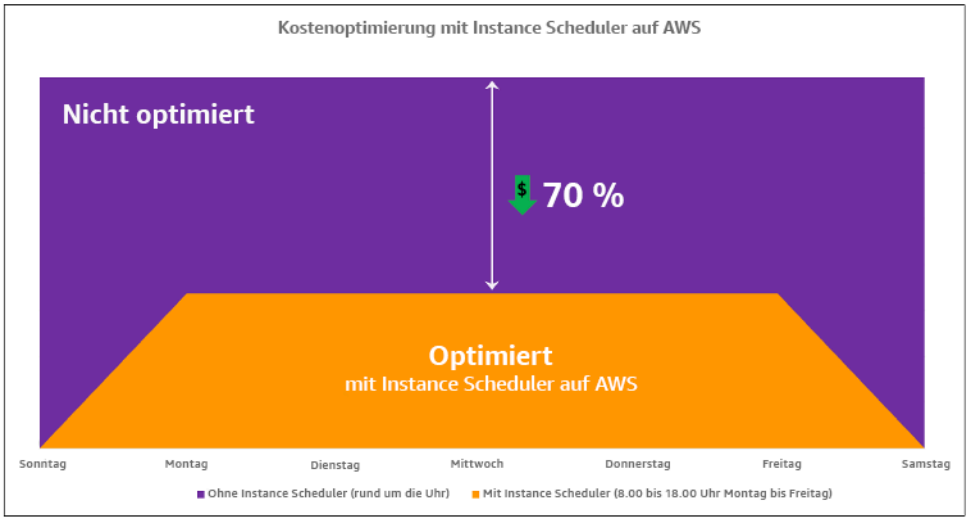
Risikostufe bei fehlender Befolgung dieser Best Practice: Niedrig

Implementierungsleitfaden

Es gibt verschiedene Möglichkeiten für AWS-Kunden, die für ihre Anwendungen verfügbaren Ressourcen zu erhöhen und Ressourcen bereitzustellen, um der Nachfrage gerecht zu werden. Eine dieser Optionen ist die Verwendung von AWS Instance Scheduler, der das Starten und Stoppen von Amazon Elastic Compute Cloud (Amazon EC2)- und (Amazon Relational Database Service) Amazon RDS-Instances automatisiert. Die andere Option ist die Verwendung von AWS Auto Scaling, mit der Sie Ihre Computing-Ressourcen automatisch an die Anforderungen Ihrer Anwendung oder Ihres Services anpassen können. Wenn Sie Ressourcen bedarfsgerecht bereitstellen, zahlen Sie nur

für die Ressourcen, die Sie tatsächlich nutzen. So senken Sie die Kosten, indem Sie Ressourcen bereitstellen, wenn sie benötigt werden, und sie beenden, wenn sie nicht mehr benötigt werden.

Mit dem AWS Instance Scheduler können Sie den Start und das Ende Ihrer Amazon EC2- und Amazon RDS-Instances zu definierten Zeiten konfigurieren. So können Sie die Nachfrage nach denselben Ressourcen innerhalb eines konsistenten Zeitmusters befriedigen. Beispiel: Ein Benutzer greift jeden Tag um acht Uhr morgens auf Amazon EC2-Instances zu, die er nach sechs Uhr abends nicht mehr benötigt. Durch diese Lösung lassen sich die Betriebskosten senken, indem sie nicht genutzte Ressourcen stoppt und sie bei Bedarf wieder startet.



Kostenoptimierung mit AWS Instance Scheduler

Mit AWS Systems Manager Quick Setup können Sie mithilfe einer einfachen Benutzeroberfläche auch ganz einfach Zeitpläne für Ihre Amazon EC2-Instances in Ihren Konten und Regionen konfigurieren. Sie können Amazon EC2- oder Amazon RDS-Instances mit dem AWS Instance Scheduler planen und bestehende Instances stoppen und starten. Sie können jedoch keine Instances stoppen und starten, die Teil Ihrer Auto Scaling-Gruppe (ASG) sind oder die Services wie Amazon Redshift oder Amazon OpenSearch Service verwalten. Auto Scaling-Gruppen haben ihre eigene Planung für die Instances in der Gruppe und diese Instances werden erstellt.

Mit AWS Auto Scaling können Sie Ihre Kapazität anpassen, um eine stabile, vorhersehbare Leistung zu möglichst niedrigen Kosten aufrechtzuerhalten. Es handelt sich um einen vollständig verwalteten und kostenlosen Service zur Skalierung Ihrer Anwendung, der sich in Amazon EC2-Instances und Spot-Flotten, Amazon ECS, Amazon DynamoDB und Amazon Aurora integrieren lässt. Auto Scaling bietet eine automatische Ressourcenerkennung, um zu helfen, Ressourcen in Ihrem Workload zu finden, die konfiguriert werden können. Es verfügt über integrierte Skalierungsstrategien zur

Optimierung der Leistung, der Kosten oder eines Gleichgewichts zwischen beiden Ressourcen und bietet eine prädiktive Skalierung, um regelmäßig auftretende Spitzen zu unterstützen.

Für die Skalierung Ihrer Auto Scaling-Gruppe haben Sie mehrere Skalierungsoptionen:

- Beibehaltung der aktuellen Instance-Levels zu jeder Zeit
- Manuelles Skalieren
- Skalieren auf der Grundlage eines Zeitplans
- Skalieren nach Bedarf
- Verwenden vorausschauender Skalierung

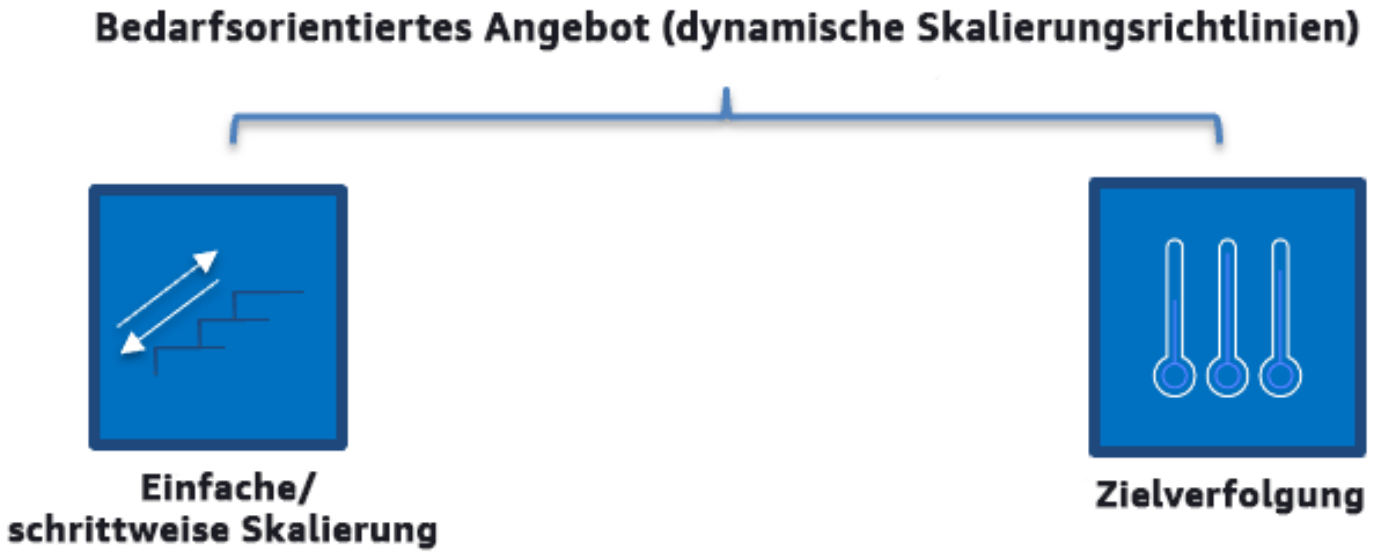
Auto Scaling-Richtlinien unterscheiden sich und können in dynamische und geplante Skalierungsrichtlinien unterteilt werden. Dynamische Richtlinien sind für manuelle oder dynamische Skalierung, die geplant oder prädiktiv sein kann. Sie können Skalierungsrichtlinien für dynamische, geplante und prädiktive Skalierung verwenden. Sie können auch Metriken und Alarme von Amazon CloudWatch verwenden, um Skalierungsereignisse für Ihren Workload auszulösen. Wir empfehlen Ihnen die Verwendung von Startvorlagen, mit denen Sie auf die neuesten Funktionen und Verbesserungen zugreifen können. Nicht alle Auto Scaling-Funktionen sind verfügbar, wenn Sie Startkonfigurationen verwenden. Sie können beispielsweise keine Auto Scaling-Gruppe erstellen, die sowohl Spot- als auch On-Demand-Instances startet oder mehrere Instance-Typen definiert. Sie müssen eine Startvorlage verwenden, um diese Funktionen zu konfigurieren. Wenn Sie Startvorlagen verwenden, empfehlen wir Ihnen, jede einzelne davon zu versionieren. Mit der Versionsverwaltung von Startvorlagen können Sie eine Teilmenge des gesamten Parametersatzes erstellen. Anschließend können Sie sie wiederverwenden, um andere Versionen derselben Startvorlage zu erstellen.

Verwenden Sie AWS Auto Scaling oder implementieren Sie die Skalierung in Ihren Code mit den AWS APIs oder SDKs. Dies reduziert Ihre Gesamtkosten für den Workload, da die Betriebskosten durch manuelle Änderungen an Ihrer Umgebung wegfallen, und kann viel schneller durchgeführt werden. So können Sie sicherstellen, dass Ihre Workload-Ressourcen jederzeit mit Ihrem Bedarf übereinstimmen. Damit Sie diese bewährte Methode befolgen und Ressourcen dynamisch für Ihr Unternehmen bereitstellen können, sollten Sie die horizontale und vertikale Skalierung in der AWS Cloud sowie die Art der auf den Amazon EC2-Instances ausgeführten Anwendungen verstehen. Ihr Cloud Financial Management-Team sollte am besten mit den technischen Teams zusammenarbeiten, um diese bewährte Methode zu befolgen.

Elastic Load Balancing (Elastic Load Balancing) unterstützt Sie bei der Skalierung durch die Verteilung der Nachfrage auf mehrere Ressourcen. Mit ASG und Elastic Load Balancing können Sie eingehende Anfragen verwalten, indem Sie den Datenverkehr optimal weiterleiten, sodass keine Instance in einer Auto Scaling-Gruppe überlastet wird. Die Anfragen werden nacheinander auf alle Ziele einer Zielgruppe verteilt, ohne Rücksicht auf Kapazität oder Auslastung.

Typische Metriken können Amazon EC2-Standardmetriken sein, z. B. CPU-Auslastung, Netzwerkdurchsatz und Elastic Load Balancing-beobachtete Anforderungs- und Antwortlatenz. Wenn möglich, sollten Sie eine Metrik verwenden, die auf das Kundenerlebnis hinweist. In der Regel handelt es sich um eine benutzerdefinierte Metrik, die aus Anwendungscode innerhalb Ihres Workloads stammen kann. Um in diesem Dokument zu erläutern, wie die Nachfrage dynamisch gedeckt werden kann, werden wir Auto Scaling in zwei Kategorien einteilen, nämlich nachfragebasierte und zeitbasierte Angebotsmodelle, und uns eingehend mit den einzelnen Modellen befassen.

Nachfragebasiertes Angebot: Nutzen Sie die Elastizität der Cloud, um Ressourcen bereitzustellen, die sich ändernden Anforderungen gerecht werden, indem Sie sich auf den Nachfragestatus nahezu in Echtzeit verlassen. Verwenden Sie für nachfragebasierte Bereitstellung APIs oder Servicefunktionen, um die Menge der Cloud-Ressourcen in Ihrer Architektur programmgesteuert zu variieren. Auf diese Weise können Sie Komponenten in Ihrer Architektur skalieren und die Anzahl der Ressourcen in Bedarfsspitzenzeiten zur Aufrechterhalten der Leistung erhöhen und die Kapazität zur Reduzierung der Kosten herabsetzen, wenn der Bedarf abklingt.

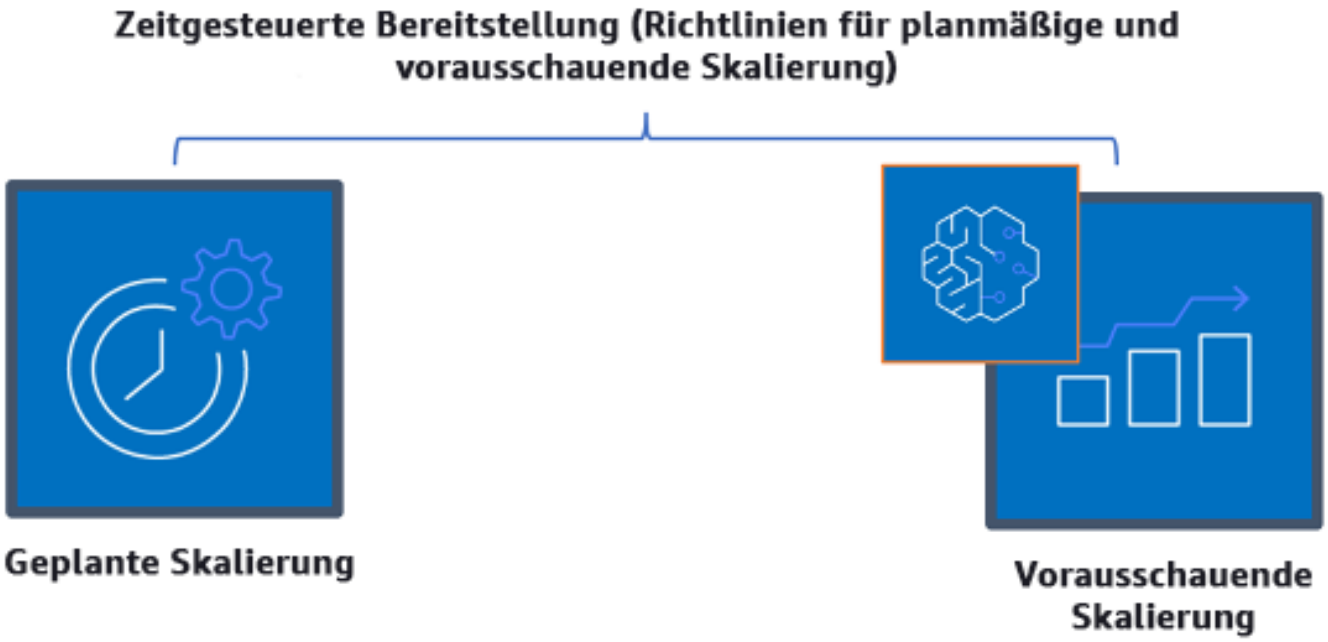


Bedarfsbasierte dynamische Skalierungsrichtlinien

- Einfache/schrittweise Skalierung: Überwacht Metriken und fügt Instances gemäß den vom Kunden manuell definierten Schritten hinzu oder entfernt sie.
- Zielverfolgung: Ein thermostatähnlicher Steuermechanismus, der automatisch Instances hinzufügt oder entfernt, um die Metriken an einem vom Kunden definierten Ziel zu halten.

Beim Aufbau der Architektur mit einem bedarfsbasierten Ansatz sollten Sie die folgenden beiden wichtigen Aspekte berücksichtigen: 1. Machen Sie sich damit vertraut, wie schnell Sie neue Ressourcen bereitstellen müssen. 2. Machen Sie sich damit vertraut, dass sich die Größe der Marge zwischen Angebot und Nachfrage ändern wird. Sie müssen darauf vorbereitet sein, das Intervall der Änderung in Bezug auf die Nachfrage zu verarbeiten, und auch Ressourcenfehler einkalkulieren.

Zeitbasiertes Angebot: Ein zeitbasierter Ansatz richtet die Ressourcenkapazität an Bedarfen aus, die prognostizierbar sind oder zeitlich gut definiert werden können. Dieser Ansatz ist in der Regel nicht abhängig vom Nutzungsgrad der Ressourcen. Mit einem zeitbasierten Ansatz können Sie sicherstellen, dass Ressourcen zu dem Zeitpunkt zur Verfügung stehen, zu dem sie benötigt werden, und ohne Verzögerung aufgrund von Startverfahren und System- oder Konsistenzprüfungen bereitgestellt werden können. Durch die Verwendung eines zeitbasierten Ansatzes können Sie zusätzliche Ressourcen hinzufügen oder die Kapazität in Spitzenzeiten erhöhen.



Zeitbasierte Skalierungsrichtlinien

Sie können geplantes oder vorausschauendes Auto Scaling verwenden, um einen zeitbasierten Ansatz zu implementieren. Workloads können zu bestimmten Zeiten auf Basis eines Zeitplans auf- oder abskaliert werden, z. B. zu Beginn der Geschäftszeiten. Dadurch sind ausreichende Ressourcen verfügbar, wenn die Benutzer ankommen oder die Nachfrage steigt. Die vorausschauende Skalierung verwendet Muster zum Aufskalieren, während bei der geplanten Skalierung vordefinierte Zeiten für die Aufskalierung verwendet werden. Sie können auch eine Strategie der attributbasierten Auswahl des Instance-Typs (ABS) in Auto Scaling-Gruppen einsetzen und so die Instance-Anforderungen in Form einer Gruppe von Attributen ausdrücken, z. B. vCPU, Arbeitsspeicher und Speicher. Darüber hinaus können Sie automatisch Instance-Typen neuerer Generationen verwenden, sobald sie veröffentlicht werden, und mit Amazon EC2 Spot-Instances auf eine größeres Speicherangebot zugreifen. Amazon EC2-Flotte und Amazon EC2 Auto Scaling wählen Instances aus, die den angegebenen Attributen entsprechen, und starten diese. So müssen Sie Instance-Typen nicht mehr manuell auswählen.

Sie können die AWS-APIs und SDKs und AWS CloudFormation nutzen, um vollständige Umgebungen bei Bedarf bereitzustellen oder zu deaktivieren. Dieser Ansatz eignet sich hervorragend für Entwicklungs- und Testumgebungen, die nur zu Geschäftszeiten oder in bestimmten Zeiträumen ausgeführt werden. Mit APIs können Sie die Größe der Ressourcen innerhalb einer Umgebung skalieren (Stichwort: vertikales Skalieren). So könnten Sie beispielsweise einen Produktions-Workload hochskalieren, indem Sie die Instance-Größe oder -Klasse ändern. Stoppen und starten Sie dazu die Instance, und wählen Sie eine andere Instance-Größe oder -Klasse aus. Diese Technik kann auch auf andere Ressourcen angewendet werden, z. B. Amazon EBS Elastic Volumes, bei denen Sie im laufenden Betrieb die Größe ändern, die Leistung anpassen (IOPS) oder den Volume-Typ ändern können.

Beim Aufbau der Architektur mit einem zeitbasierten Ansatz sollten Sie die beiden folgenden wichtigen Aspekte berücksichtigen: 1: Wie konsistent ist das Nutzungsmuster? 2. Wie wirken sich Musteränderungen aus? Sie können die Treffergenauigkeit für Prognosen durch die Überwachung Ihrer Workloads und die Verwendung von Business Intelligence erhöhen. Wenn Sie signifikante Änderungen im Nutzungsmuster erkennen, können Sie die Zeiten ändern, um eine Deckung zu gewährleisten.

Implementierungsschritte

- Konfigurieren der geplanten Skalierung: Für vorhersehbare Änderungen des Bedarfs kann die zeitbasierte Skalierung die richtige Anzahl an Ressourcen in einem angemessenen Zeitraum bereitstellen. Es ist auch nützlich, wenn die Ressourcenerstellung und -konfiguration nicht schnell genug ist, um bei Bedarf auf Änderungen zu reagieren. Mithilfe der Workload-Analyse konfigurieren

Sie die geplante Skalierung mithilfe von AWS Auto Scaling. Zur Konfiguration der zeitbasierten Planung können Sie die vorausschauende Skalierung der geplanten Skalierung verwenden, um im Vorfeld die Anzahl der Amazon EC2-Instances in Ihren Auto Scaling-Gruppen entsprechend den erwarteten oder prognostizierbaren Lastveränderungen zu erhöhen.

- Konfigurieren der vorausschauenden Skalierung: Mit der vorausschauenden Skalierung können Sie im Voraus die Amazon EC2-Instances in Ihrer Auto Scaling-Gruppe anhand von täglichen und wöchentlichen Mustern in Datenverkehrsflüssen erhöhen. Wenn Sie regelmäßige Spitzen beim Datenverkehr sowie Anwendungen haben, die lange brauchen, um zu starten, sollten Sie die vorausschauende Skalierung in Betracht ziehen. Die vorausschauende Skalierung kann Ihnen helfen, schneller zu skalieren, indem die Kapazität vor der prognostizierten Last initialisiert wird, im Gegensatz zur dynamischen Skalierung, die nur reaktiv ist. Wenn die Benutzer Ihre Workloads beispielsweise mit Beginn der Geschäftszeiten nutzen und sie nach Geschäftsschluss nicht mehr brauchen, kann die vorausschauende Skalierung die Kapazität vor den Geschäftszeiten erhöhen. Die Verzögerung, die bei der dynamischen Skalierung entsteht, bis sie auf den veränderten Datenverkehr reagiert, entfällt somit.
- Konfigurieren von dynamischem Auto Scaling: Verwenden Sie Auto Scaling, um die Skalierung auf der Grundlage von aktiven Workload-Metriken zu konfigurieren. Verwenden Sie die Analyse und konfigurieren Sie Auto Scaling so, dass es auf den richtigen Ressourcenebenen gestartet wird. Achten Sie darauf, dass der Workload in der erforderlichen Zeit skaliert wird. Mit nur einer Auto Scaling-Gruppe können Sie eine Flotte von On-Demand-Instances und Spot-Instances starten und automatisch skalieren. Sie erhalten nicht nur Rabatte für Spot-Instances, sondern können auch Reserved Instances oder einen Savings Plan nutzen, um ermäßigte Tarife gegenüber den normalen Preisen für On-Demand-Instances zu erhalten. Durch die Kombination dieser Faktoren sparen Sie Kosten für Amazon EC2-Instances und können die gewünschte Skalierung und Leistung für Ihre Anwendung festlegen.

Ressourcen

Zugehörige Dokumente:

- Mit AWS Auto Scaling
- Mit dem AWS Instance Scheduler
- Scale the size of your Auto Scaling group (Skalieren der Größe Ihrer Auto Scaling-Gruppe)
- Erste Schritte mit Amazon EC2 Auto Scaling
- Erste Schritte mit Amazon SQS
- Geplante Skalierung für Amazon EC2 Auto Scaling

- Vorausschauende Skalierung für Amazon EC2 Auto Scaling

Zugehörige Videos:

- Zielverfolgungs-Skalierungsrichtlinien für Auto Scaling
- Mit dem AWS Instance Scheduler

Zugehörige Beispiele:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet (Attributbasierte Auswahl des Instance-Typs für EC2 Auto Scaling und EC2 Fleet)
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling (Kostenoptimierung von Amazon Elastic Container Service mit geplanter Skalierung)
- Vorausschauende Skalierung mit Amazon EC2 Auto Scaling
- How do I use Instance Scheduler with AWS CloudFormation to schedule Amazon EC2 instances? (Wie verwende ich Instance Scheduler mit CloudFormation zur Planung von EC2-Instances?)

Optimierung im Laufe der Zeit

Fragen

- KOSTEN 10. Wie können Sie neue Services bewerten?
- KOSTEN 11. Wie bewerten Sie die Kosten des Aufwands?

KOSTEN 10. Wie können Sie neue Services bewerten?

Im Zuge der Veröffentlichung neuer Services und Funktionen durch AWS empfiehlt es sich, dass Sie Ihre bestehenden Entscheidungen zur Architektur überdenken, um sicherzustellen, dass diese weiterhin so kostengünstig wie möglich sind.

Bewährte Methoden

- COST10-BP01 Entwickeln eines Prüfprozesses für Workloads
- COST10-BP02 Regelmäßige Prüfung und Analyse des betreffenden Workloads

COST10-BP01 Entwickeln eines Prüfprozesses für Workloads

Entwickeln Sie einen Prozess, der die Kriterien und den Prozess für die Workload-Prüfung definiert. Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen. Beispielsweise ist es sinnvoll, zentrale Workloads oder Workloads, deren Wert mehr als 10 % der Rechnung ausmacht, vierteljährlich oder alle sechs Monate zu prüfen, während Workloads mit einem Wert von weniger als 10 % der Rechnung jährlich überprüft werden sollten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Um sicherzustellen, dass Sie immer den kosteneffizientesten Workload haben, müssen Sie den Workload regelmäßig überprüfen, um zu wissen, ob es Möglichkeiten gibt, neue Services, Funktionen und Komponenten zu implementieren. Damit Sie insgesamt niedrigere Kosten erzielen, muss der Prozess proportional zu den potenziellen Einsparungen sein. Beispielsweise sollten Workloads, die 50 % Ihrer Gesamtausgaben ausmachen, regelmäßiger und gründlicher überprüft werden als Workloads, die 5 % Ihrer Gesamtausgaben ausmachen. Lassen Sie auch externe Faktoren oder Volatilität in Ihre Überlegungen einfließen. Wenn der Workload eine bestimmte Geografie oder ein bestimmtes Marktsegment abzielt und Änderungen in diesem Bereich vorhergesagt werden, können häufigere Überprüfungen zu Kosteneinsparungen führen. Ein weiterer Faktor bei der Überprüfung ist die Implementierung von Änderungen. Wenn es erhebliche Kosten für das Testen und Validieren von Änderungen gibt, sollten Überprüfungen seltener erfolgen.

Denken Sie an die langfristigen Kosten für die Wartung veralteter Komponenten und Ressourcen sowie die Unfähigkeit, neue Funktionen in diese zu implementieren. Die aktuellen Kosten für Tests und Validierung können den vorgeschlagenen Vorteil übersteigen. Doch mit der Zeit können sich die Kosten für die Änderung erheblich erhöhen, da die Lücke zwischen dem Workload und den aktuellen Technologien zunimmt, was zu noch größeren Kosten führt. Beispielsweise sind die Kosten für den Wechsel zu einer neuen Programmiersprache derzeit möglicherweise nicht günstig. In fünf Jahren können sich jedoch die Kosten für Personen, die in dieser Sprache qualifiziert sind, erhöhen. Aufgrund des Wachstums des Workloads würden Sie ein noch größeres System in die neue Sprache verlagern, was noch mehr Aufwand erfordert als zuvor.

Unterteilen Sie Ihren Workload in Komponenten, weisen Sie die Kosten der Komponente zu (eine Schätzung reicht aus) und listen Sie dann die Faktoren (z. B. Aufwand und externe Märkte) neben den einzelnen Komponenten auf. Verwenden Sie diese Indikatoren, um eine Überprüfungshäufigkeit für jeden Workload zu bestimmen. Zum Beispiel können bei Webservern hohe Kosten, geringer Änderungsaufwand und hohe externe Faktoren anfallen, was zu einer hohen Überprüfungshäufigkeit

führt. Bei einer zentralen Datenbank können mittlere Kosten, hoher Änderungsaufwand und niedrige externe Faktoren anfallen, was zu einer mittleren Überprüfungshäufigkeit führt.

Definieren Sie einen Prozess, mit dem sich neu verfügbare Services, Designmuster, Ressourcentypen und Konfigurationen zur Optimierung Ihrer Workload-Kosten bewerten lassen. Ähnlich wie bei der Prüfung der Säule „Leistungseffizienz“ und der Prüfung der Säule „Zuverlässigkeit“ identifizieren, validieren und priorisieren Sie Optimierungs- und Verbesserungsmaßnahmen. Führen Sie eine Problembehandlung durch und nehmen Sie diese in Ihr Backlog auf.

Implementierungsschritte

- Definieren der Überprüfungsfrequenz: Legen Sie fest, wie häufig der Workload und seine Komponenten überprüft werden sollen. Reservieren Sie Zeit und Ressourcen, um eine kontinuierliche Verbesserungen zu ermöglichen, und prüfen Sie die Häufigkeit, um Ihren Workload zu optimieren und effizienter zu gestalten. Dies ist eine Kombination von Faktoren und kann sich von Workload zu Workload innerhalb Ihres Unternehmens und zwischen Komponenten im Workload unterscheiden. Häufige Faktoren sind u. a. die Bedeutung für die Organisation, gemessen in Bezug auf Umsatz oder Marke, die Gesamtkosten für die Ausführung des Workloads (einschließlich Betriebs- und Ressourcenkosten), die Komplexität des Workloads, wie einfach es ist, eine Änderung zu implementieren, Softwarelizenzvereinbarungen sowie Änderungen, die aufgrund mangelhafter Lizenzen erhebliche Erhöhungen der Lizenzkosten verursachen würden. Komponenten können funktional oder technisch definiert werden, z. B. Webserver und Datenbanken oder Rechen- und Speicherressourcen. Gleichen Sie die Faktoren entsprechend aus und entwickeln Sie einen Zeitraum für den Workload und seine Komponenten. Sie können sich entscheiden, den vollständigen Workload alle 18 Monate, die Webserver alle 6 Monate, die Datenbank alle 12 Monate, die Datenverarbeitungs- und Kurzzeitspeicherung alle 6 Monate und die Langzeitspeicherung alle 12 Monate zu überprüfen.
- Definieren einer gründlichen Überprüfung: Legen Sie fest, wie viel Aufwand für die Prüfung des Workloads oder der Workload-Komponenten aufgewendet wird. Ähnlich wie bei der Überprüfungsfrequenz geht es hier um mehrere Faktoren, die ausgeglichen sein müssen. Bewerten und priorisieren Sie regelmäßig Verbesserungsmöglichkeiten, um die Maßnahmen dort zu intensivieren, wo sie den größten Nutzen bringen. So erfahren Sie auch, wie viel Aufwand für diese Aktivitäten erforderlich ist. Wenn die erwarteten Ergebnisse die Ziele nicht erfüllen und der Aufwand mehr kostet, wiederholen Sie den Versuch mit alternativen Vorgehensweisen. Bei Ihren Prüfungen sollten auch Zeit und Ressourcen genutzt werden, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen. Sie können beispielsweise entscheiden, für die Analyse der

Datenbankkomponente eine Woche, für die Analyse von Datenverarbeitungsressourcen eine Woche und für die Analyse von Speicherprüfungen vier Stunden aufzuwenden.

Ressourcen

Zugehörige Dokumente:

- AWS News-Blog
- Arten von Cloud Computing
- Neuerungen bei AWS

Zugehörige Beispiele:

- AWS Support Proactive Services (AWS Support für Proactive Services)
- Regular workload reviews for SAP workloads (Regelmäßige Workload-Prüfungen für SAP-Workloads)

COST10-BP02 Regelmäßige Prüfung und Analyse des betreffenden Workloads

Bestehende Workloads werden basierend auf den einzelnen definierten Prozessen regelmäßig überprüft, um zu ermitteln, ob neue Services übernommen, vorhandene Services ersetzt oder die Architektur von Workloads geändert werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

AWS fügt laufend neue Funktionen hinzu, sodass Sie mit der neuesten Technologie experimentieren und schneller Innovationen einführen können. Unter Neuerungen bei AWS erfahren Sie, wie AWS dies ermöglicht. Darüber hinaus finden Sie hier eine kurze Übersicht über die Services und Funktionen von AWS sowie öffentliche Ankündigungen zur regionalen Expansion. Sie können sich eingehender über die angekündigten Veröffentlichungen informieren und diese zur Prüfung und Analyse Ihrer bestehenden Workloads verwenden. Um die Vorteile neuer AWS-Services und -Funktionen zu nutzen, müssen Sie Ihre Workloads prüfen und die neuen Services und Funktionen wie erforderlich implementieren. Dies bedeutet, dass Sie möglicherweise vorhandene Services, die Sie für Ihren Workload verwenden, ersetzen oder Ihren Workload modernisieren müssen, um diese neuen AWS-Services einzuführen. Sie können beispielsweise Ihre Workloads überprüfen und die Messaging-Komponente durch Amazon Simple Email Service ersetzen. Dadurch entfallen die Kosten

für den Betrieb und die Verwaltung einer Flotte von Instances, während die gesamte Funktionalität zu geringeren Kosten bereitgestellt wird.

Bei der Analyse Ihres Workloads und der Ermittlung potenzieller Chancen sollten Sie nicht nur neue Services, sondern auch neue Möglichkeiten zur Entwicklung von Lösungen berücksichtigen. Sehen Sie sich die Videos unter This is My Architecture (Dies ist meine Architektur) auf AWS an, um mehr über die Architekturdesigns anderer Kunden sowie ihre Herausforderungen und Lösungen zu erfahren. Die All-In-Reihe bietet weitere Informationen zu praktischen Anwendungen der AWS-Services und stellt Kundengeschichten vor. Sie können sich auch die Videoreihe Back to Basics (Zurück zu den Grundlagen) ansehen, in der bewährte Methoden zur grundlegenden Cloud-Architektur erklärt, untersucht und aufgeschlüsselt werden. Eine weitere Quelle ist die Videoreihe How to Build This (Anleitungen zur Entwicklung), die Menschen mit guten Ideen dabei unterstützen soll, ihr Minimum Viable Product (MVP, Minimalprodukt) mithilfe der Services von AWS zum Leben zu erwecken. Hier finden Entwickler mit guten Ideen aus aller Welt Architekturanleitungen von erfahrenen AWS Solution Architects. In unseren Ressourcenmaterialien unter Erste Schritte finden Sie darüber hinaus ausführliche Tutorials.

Befolgen Sie vor der Durchführung Ihres Überprüfungsprozesses die Anforderungen Ihres Unternehmens in Bezug auf den Workload, die Sicherheit und den Datenschutz, um einen spezifischen Service oder eine spezifische Region zu nutzen. Befolgen Sie während des vereinbarten Überprüfungsprozesses die Leistungsanforderungen.

Implementierungsschritte

- Regelmäßige Überprüfung des Workloads: Führen Sie mit Ihrem definierten Prozess Überprüfungen mit der angegebenen Häufigkeit durch. Stellen Sie sicher, dass Sie den richtigen Aufwand für jede Komponente aufwenden. Dieser Prozess ähnelt dem anfänglichen Designprozess, bei dem Sie Services für die Kostenoptimierung ausgewählt haben. Analysieren Sie die Services und die Vorteile, die sie mit sich bringen würden, sowie den Zeitfaktor bei den Änderungskosten. Analysieren Sie nicht nur die langfristigen Vorteile.
- Implementieren neuer Services: Wenn es das Ziel der Analyse ist, Änderungen zu implementieren, führen Sie zunächst eine Analyse der Basisanforderungen des Workloads durch, um die aktuellen Kosten für jede Ausgabe festzustellen. Implementieren Sie die Änderungen und führen Sie dann eine Analyse durch, um die neuen Kosten für jede Ausgabe zu bestätigen.

Ressourcen

Zugehörige Dokumente:

- AWS News-Blog
- Neuerungen bei AWS
- AWS-Dokumentation
- Erste Schritte mit AWS
- AWS General Resources (Allgemeine AWS-Ressourcen)

Zugehörige Videos:

- AWS - This is My Architecture (AWS – Dies ist meine Architektur)
- AWS - Back to Basics (AWS – Zurück zu den Grundlagen)
- AWS – All-In-Reihe
- How to Build This (Anleitungen zur Entwicklung)

KOSTEN 11. Wie bewerten Sie die Kosten des Aufwands?

Bewährte Methoden

- COST11-BP01 Durchführen von Automatisierungen für Betriebsabläufe

COST11-BP01 Durchführen von Automatisierungen für Betriebsabläufe

Bewerten Sie die Betriebskosten in der Cloud und konzentrieren Sie sich dabei auf die Quantifizierung der Zeit- und Aufwandsersparnisse bei administrativen Aufgaben und Bereitstellungen, der Minimierung des Risikos menschlicher Fehler, Compliance und anderen Abläufen durch Automatisierung. Ermitteln Sie den Zeitaufwand und die damit verbundenen Kosten, die für den operativen Aufwand erforderlich sind, und implementieren Sie die Automatisierung von Verwaltungsaufgaben, um den manuellen Aufwand zu minimieren, wo immer dies möglich ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Die Automatisierung von Abläufen reduziert die Häufigkeit von manuellen Aufgaben, optimiert die Effizienz und bietet Kunden Vorteile, indem sie eine konsistente und zuverlässige Umgebung bei der Bereitstellung, Verwaltung oder dem Betrieb von Workloads ermöglicht. Sie können Infrastrukturressourcen von manuellen Betriebsaufgaben entlasten und für hochwertigere Aufgaben

sowie Innovationen einsetzen, was den Unternehmenswert verbessert. Unternehmen benötigen eine bewährte, getestete Möglichkeit, ihre Workloads in der Cloud zu verwalten. Diese Lösung muss sicher, schnell und kosteneffizient sein. Darüber hinaus darf sie nur ein minimales Risiko bei maximaler Zuverlässigkeit mit sich bringen.

Beginnen Sie damit, Ihre Betriebsabläufe basierend auf dem erforderlichen Aufwand zu priorisieren, indem Sie sich die Gesamtbetriebskosten ansehen. Beispiel: Wie lange dauert es, neue Ressourcen in der Cloud bereitzustellen, vorhandene Ressourcen zu optimieren oder die notwendigen Konfigurationen zu implementieren? Sehen Sie sich die Gesamtkosten für die menschliche Arbeitskraft an und berücksichtigen Sie dabei die Kosten für Betriebsabläufe und Verwaltung. Priorisieren Sie die Automatisierung von Verwaltungsaufgaben, um die menschliche Arbeitskraft zu reduzieren.

Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen. Beispiel: Untersuchen Sie den Zeitaufwand für das manuelle im Vergleich zum automatischen Ausführen von Aufgaben. Priorisieren Sie die Automatisierung sich wiederholender, hochwertiger, zeitaufwändiger und komplexer Aktivitäten. Aufgaben mit hohem Wert oder einem hohen Risiko von menschlichen Fehlern sind in der Regel der bessere Ausgangspunkt für Automatisierungen, da das Risiko oft unerwünschte zusätzliche Betriebskosten (z. B. für Überstunden des Betriebsteams) mit sich bringt.

Verwenden Sie Automatisierungstools wie AWS Systems Manager oder AWS Config, um Betriebs-, Compliance-, Überwachungs-, Lebenszyklus- und Kündigungsprozesse zu optimieren. Mit AWS-Services und -Tools und Produkten von Drittanbietern können Sie die von Ihnen implementierten Automatisierungen an Ihre spezifischen Anforderungen anpassen. Die folgende Tabelle zeigt einige der zentralen Betriebsfunktionen der AWS-Services, mit denen Sie die Verwaltung und die Betriebsabläufe automatisieren können:

- AWS Audit Manager: Kontinuierliche Überprüfung Ihrer AWS-Nutzung, um die Risiko- und Compliance-Bewertung zu vereinfachen.
- AWS Backup: Zentrale Verwaltung und Automatisierung des Datenschutzes.
- AWS Config: Konfigurieren von Computing-Ressourcen sowie Bewerten, Prüfen und Evaluieren von Konfigurationen und Ressourceninventar.
- AWS CloudFormation: Launchen von hochverfügbaren Ressourcen mit Infrastruktur as Code.
- AWS CloudTrail: IT-Änderungsverwaltung, Compliance und Kontrolle.
- Amazon EventBridge: Planen von Ereignissen und Auslösen von Maßnahmen durch AWS Lambda.

- AWS Lambda: Automatisieren sich wiederholender Prozesse, indem sie durch Ereignisse ausgelöst oder mit AWS EventBridge nach einem festen Zeitplan ausgeführt werden.
- AWS Systems Manager: Starten und Beenden von Workloads, Patchen von Betriebssystemen, automatische Konfiguration und dauerhafte Verwaltung.
- AWS Step Functions: Planen von Aufträgen und Automatisieren von Workflows.
- AWS Service Catalog: Vorlagennutzung und Infrastructure as Code mit Compliance und Kontrolle.

Wenn Sie unverzüglich Automatisierungen mit den Produkten und Services von AWS einführen möchten, in Ihrer Organisation jedoch nicht über die erforderliche Kompetenz verfügen, wenden Sie sich an AWS Managed Services (AMS), AWS Professional Services oder AWS-Partner, um die Automatisierung in höherem Umfang zu nutzen und Ihre Operational Excellence in der Cloud zu verbessern.

AWS Managed Services (AMS) ist ein Service, der die AWS-Infrastruktur für Unternehmenskunden und -partner betreibt. Er bietet eine sichere und konforme Umgebung, in der Sie Ihre Workloads bereitstellen können. AMS verwendet Enterprise-Cloud-Betriebsmodelle mit Automatisierung, damit Sie Ihre Organisationsanforderungen erfüllen, schneller in die Cloud wechseln und Ihre laufenden Verwaltungskosten senken können.

AWS Professional Services kann Sie auch dabei unterstützen, die gewünschten Geschäftsziele zu erreichen und Betriebsabläufe mit AWS zu automatisieren. Sie unterstützen die Kunden bei der Bereitstellung von automatisierten, robusten und agilen IT-Abläufen sowie für die Cloud optimierten Governance-Funktionen. Detaillierte Überwachungsbeispiele und empfohlene bewährte Methoden finden Sie im Whitepaper zur Säule für die betriebliche Effizienz.

Implementierungsschritte

- Einmal entwickeln und mehrmals bereitstellen: Verwenden Sie Infrastructure as Code wie beispielsweise CloudFormation, AWS SDK oder AWS CLI zur einmaligen Bereitstellung und mehrfachen Nutzung für ähnliche Umgebungen oder für die Notfallwiederherstellung. Nutzen Sie während der Bereitstellung Tags, um die Nutzung wie in anderen bewährten Methoden beschrieben zu verfolgen. Verwenden Sie AWS Launch Wizard, um die erforderliche Zeit für die Bereitstellung vieler beliebter Unternehmens-Workloads zu reduzieren. AWS Launch Wizard leitet Sie durch die Dimensionierung, Konfiguration und Bereitstellung von Unternehmens-Workloads gemäß den bewährten Methoden von AWS. Sie können auch den Service Catalog verwenden. Dieser unterstützt Sie bei der Erstellung und Verwaltung von genehmigten Vorlagen für

Infrastructure as Code zur Verwendung in AWS, sodass alle Mitarbeiter genehmigte Selfservice-Cloud-Ressourcen erkunden können.

- Automatisieren Sie die kontinuierliche Compliance: Erwägen Sie, die Auswertung und Korrektur aufgezeichneter Konfigurationen anhand vordefinierter Standards zu automatisieren. Durch die Kombination von AWS Organizations mit den Funktionen von AWS Config und AWS CloudFormation können Sie die Konfigurations-Compliance für Hunderte von Mitgliedskonten in großem Umfang effizient verwalten und automatisieren. Sie können Änderungen an Konfigurationen und Beziehungen zwischen AWS-Ressourcen überprüfen und den Verlauf einer Ressourcenkonfiguration nachverfolgen.
- Automatisieren Sie Überwachungsaufgaben: AWS bietet verschiedene Tools, mit denen Sie Services überwachen können. Sie können diese Tools so konfigurieren, dass sie Überwachungsaufgaben automatisieren. Erstellen und implementieren Sie einen Überwachungsplan, der Überwachungsdaten aus allen Teilen Ihrer Workload erfasst, sodass Sie einen Mehrpunktausfall, falls ein solcher auftritt, einfacher debuggen können. Beispielsweise können Sie die automatisierten Überwachungstools verwenden, um Amazon EC2 zu beobachten und benachrichtigt zu werden, wenn bei Systemstatusprüfungen, Instance-Statusprüfungen und Amazon CloudWatch-Alarmen etwas nicht stimmt.
- Automatisieren Sie die Wartung und Betriebsabläufe: Führen Sie Routineaufgaben automatisch ohne menschliche Eingriffe aus. Wenn Sie die Services und Tools von AWS verwenden, können Sie auswählen, welche AWS-Automatisierungen Sie implementieren und an Ihre spezifischen Anforderungen anpassen möchten. Verwenden Sie beispielsweise EC2 Image Builder zum Entwickeln, Testen und Bereitstellen von virtuellen Maschinen und Container-Images zur Verwendung in AWS oder On-Premises oder zum Patchen Ihrer EC2 Instances mit AWS-SSM. Wenn die gewünschte Aktion nicht mit den Services von AWS ausgeführt werden kann oder Sie komplexere Aktionen mit Filterung der Ressourcen benötigen, automatisieren Sie Ihre Betriebsabläufe mit AWS Command Line Interface- (AWS CLI) oder AWS-SDK-Tools. AWS CLI bietet die Möglichkeit, die gesamte Kontrolle und Verwaltung von AWS-Services mit Skripts zu automatisieren, ohne dass die AWS Management Console verwendet werden muss. Wählen Sie Ihre bevorzugten AWS-SDKs aus, um mit den AWS-Services zu interagieren. Weitere Codebeispiele finden Sie unter Repository mit Codebeispielen für das AWS-SDK.
- Schaffen Sie einen kontinuierlichen Lebenszyklus mit Automatisierungen: Es ist wichtig, dass Sie ausgereifte Lebenszyklusrichtlinien einrichten und beibehalten, nicht nur für Vorschriften oder Redundanz, sondern auch für die Kostenoptimierung. Sie können AWS Backup verwenden, um den Datenschutz von Datenspeichern wie Buckets, Volumes, Datenbanken und Dateisystemen zentral zu verwalten und zu automatisieren. Mit Amazon Data Lifecycle Manager lassen sich

außerdem das Erstellen, Aufbewahren und Löschen von EBS-Snapshots und EBS-gestützten AMIs automatisieren.

- Löschen Sie unnötige Ressourcen: Es ist durchaus üblich, ungenutzte Ressourcen in der Sandbox oder in Entwicklungs-AWS-Konten anzusammeln. Entwickler erstellen und experimentieren im Rahmen des normalen Entwicklungszyklus mit verschiedenen Services und Ressourcen, und dann löschen sie diese Ressourcen nicht, wenn sie nicht mehr benötigt werden. Ungenutzte Ressourcen können unnötige und manchmal hohe Kosten für die Organisation verursachen. Durch das Löschen dieser Ressourcen können die Kosten für den Betrieb dieser Umgebungen gesenkt werden. Vergewissern Sie sich im Zweifelsfall, dass diese Daten nicht benötigt werden oder gesichert sind. Sie können AWS CloudFormation verwenden, um bereitgestellte Stapel zu bereinigen, wodurch die meisten in der Vorlage definierten Ressourcen automatisch gelöscht werden. Alternativ können Sie mit Tools wie aws-nuke eine Automatisierung zum Löschen von AWS-Ressourcen einrichten.

Ressourcen

Zugehörige Dokumente:

- Modernisierung der Betriebsabläufe in der AWS Cloud
- AWS-Services für die Automatisierung
- Infrastructure and automation (Infrastruktur und Automatisierung)
- AWS Systems Manager-Automatisierung
- Automated and manual monitoring (Automatisierte und manuelle Überwachung)
- AWS-Automatisierungen für SAP-Administration und -Betrieb
- AWS Managed Services
- AWS Professional Services

Zugehörige Videos:

- Umfangreiche Automatisierung der kontinuierlichen Compliance in AWS
- AWS Backup Demo: Cross-Account & Cross-Region Backup (AWS Backup Demo: Konto- und regionsübergreifendes Backup)
- Patchen für Ihre Amazon EC2-Instances

Zugehörige Beispiele:

- Reinventing automated operations (Part I) (Automatisierte Betriebsabläufe neu erfinden (Teil I))
- Reinventing automated operations (Part II) (Automatisierte Betriebsabläufe neu erfinden (Teil II))
- Automatisierung der Löschung von AWS-Ressourcen mit asw-nuke
- Löschen ungenutzter Amazon EBS-Volumes durch AWS Config und AWS SSM
- Umfangreiche Automatisierung der kontinuierlichen Compliance in AWS
- IT-Automatisierungen mit AWS Lambda

Nachhaltigkeit

Bei der Säule „Nachhaltigkeit“ geht es darum, die Auswirkungen der genutzten Services zu verstehen, diese über den gesamten Workload-Lebenszyklus hinweg zu quantifizieren sowie konzeptionelle Grundsätze und bewährte Methoden einzusetzen, die dabei helfen, diese Auswirkungen zu reduzieren, wenn Cloud-Workloads erstellt werden. Verbindliche Leitlinien zur Implementierung finden Sie im Whitepaper zur Säule „Nachhaltigkeit“.

Bereiche für bewährte Methoden

- Auswahl von Regionen erläutert
- Ausrichtung am Bedarf
- Software und Architektur
- Daten
- Hardware und Services
- Prozess und Kultur

Auswahl von Regionen erläutert

Frage

- SUS 1 Wie wählen Sie Regionen für Ihren Workload aus?

SUS 1 Wie wählen Sie Regionen für Ihren Workload aus?

Welche Region Sie für Ihren Workload auswählen, hat signifikante Auswirkungen auf seine KPIs, u. a. Leistung, Kosten und CO2-Bilanz. Um diese KPIs effizient zu verbessern, sollten Sie die Regionen für Ihren Workload abhängig von den Unternehmensanforderungen und Nachhaltigkeitszielen auswählen.

Bewährte Methoden

- SUS01-BP01 Auswählen der Region auf Grundlage von Unternehmensanforderungen und Nachhaltigkeitszielen

SUS01-BP01 Auswählen der Region auf Grundlage von Unternehmensanforderungen und Nachhaltigkeitszielen

Wählen Sie eine Region für Ihren Workload auf Grundlage Ihrer Geschäftsanforderungen und Nachhaltigkeitsvorgaben aus, um so KPIs wie Leistung, Kosten und CO2-Bilanz zu optimieren.

Typische Anti-Muster:

- Sie wählen die Region des Workloads auf Grundlage Ihres eigenen Standorts aus.
- Sie konsolidieren alle Workload-Ressourcen an einem geografischen Standort.

Vorteile der Einführung dieser bewährten Methode: Wenn Sie einen Workload in der Nähe von Amazon-Projekten für erneuerbare Energien oder in Regionen mit nachweislich niedrigen Kohlendioxidemissionen platzieren, kann die CO2-Bilanz eines Clouds-Workloads gesenkt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die AWS Cloud ist ein ständig wachsendes Netzwerk aus Regionen und Points of Presence (PoP), die durch eine globale Netzwerkinfrastruktur verbunden werden. Welche Region Sie für Ihren Workload auswählen, hat signifikante Auswirkungen auf seine KPIs, u. a. Leistung, Kosten und CO2-Bilanz. Um diese KPIs effizient zu verbessern, sollten Sie die Regionen für Ihren Workload abhängig von den Unternehmensanforderungen sowie Nachhaltigkeitszielen auswählen.

Implementierungsschritte

- Befolgen Sie diese Schritte, um potenzielle Regionen für Ihren Workload zu bewerten und in die engere Auswahl zu nehmen. Berücksichtigen Sie dabei die Anforderungen Ihres Unternehmens, u. a. im Bezug auf Compliance, verfügbare Funktionen, Kosten und Latenz:
  - Vergewissern Sie sich, dass die Regionen konform sind und die entsprechenden lokalen Vorschriften erfüllen.
  - Prüfen Sie anhand der Liste regionaler AWS-Services, ob die Regionen über die für Ihren Workload erforderlichen Services und Features verfügen.

  - Berechnen Sie die Kosten des Workloads in jeder Region mithilfe des AWS -Preisrechner.
  - Testen Sie die Netzwerklatenz zwischen den Standorten Ihrer Endbenutzer und jeder AWS-Region.
- Wählen Sie Regionen in der Nähe von Amazon-Projekten für erneuerbare Energien aus. Es sollte sich um Regionen handeln, in denen das Stromnetz nachweislich geringere Kohlendioxidemissionen generiert als andere Standorte (oder Regionen).
  - Ermitteln Sie die relevanten Nachhaltigkeitsrichtlinien, um die jährlichen CO2-Emissionen gemäß dem Greenhouse Gas Protocol zu nachzuverfolgen und zu vergleichen (marktbasierte und standortbasierte Verfahren).
  - Wählen Sie die Region entsprechend dem Verfahren aus, mit dem Sie CO2-Emissionen nachverfolgen. Weitere Informationen zum Auswählen einer Region anhand von Nachhaltigkeitsrichtlinien finden Sie im Artikel zum Auswählen einer Region für Ihren Workload auf Grundlage von Nachhaltigkeitszielen.

Ressourcen

Zugehörige Dokumente:

- Grundlagen zu CO2-Emissionsschätzungen
- Amazon Weltweit
- Methodik für erneuerbare Energien
- „Relevante Aspekte bei der Wahl einer Region für Ihre Workloads“ erläutert

Zugehörige Videos:

- AWSre:Invent 2023 – Nachhaltigkeitsinnovationen in der globalen AWS-Infrastruktur
- AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft
- AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen
- AWS re:Invent 2022 – Nachhaltige Architektur und Reduzieren der AWS-CO2-Bilanz
- AWS re:Invent 2022 – Nachhaltigkeit in der globalen AWS-Infrastruktur

Ausrichtung am Bedarf

Frage

- SUS 2 Wie richten Sie Cloud-Ressourcen am Bedarf aus?

SUS 2 Wie richten Sie Cloud-Ressourcen am Bedarf aus?

Die Art und Weise, wie Benutzer und Anwendungen Ihre Workloads und andere Ressourcen nutzen, kann Sie bei der Identifizierung von Verbesserungen unterstützen, um Nachhaltigkeitsziele zu erreichen. Skalieren Sie Ihre Infrastruktur so, dass Sie den Bedarf kontinuierlich anpassen können. Sorgen Sie zudem dafür, dass zur Unterstützung Ihrer Benutzer nicht mehr Ressourcen verwendet werden als unbedingt nötig. Richten Sie Service-Levels an den Kundenanforderungen aus. Positionieren Sie Ressourcen so, dass die Netzwerkkapazitäten, die für Benutzer und Anwendungen erforderlich sind, begrenzt werden. Entfernen Sie ungenutzte Komponenten. Stellen Sie Teammitgliedern Geräte zur Verfügung, die ihre Anforderungen bei geringstmöglichen Auswirkungen auf die Nachhaltigkeit erfüllen.

Bewährte Methoden

- SUS02-BP01 Dynamisches Skalieren der Workload-Infrastruktur
- SUS02-BP02 Ausrichten von SLAs an Nachhaltigkeitszielen
- SUS02-BP03 Beenden der Erstellung und Wartung nicht verwendeter Komponenten
- SUS02-BP04 Optimieren der geografischen Platzierung von Workloads auf der Grundlage ihrer Netzwerkanforderungen
- SUS02-BP05 Optimieren von Ressourcen für Teammitglieder im Hinblick auf die ausgeführten Aktivitäten
- SUS02-BP06 Implementierung von Pufferung oder Drosselung, um die Bedarfskurve zu verflachen

SUS02-BP01 Dynamisches Skalieren der Workload-Infrastruktur

Nutzen Sie die Elastizität der Cloud und skalieren Sie Ihre Infrastruktur dynamisch, um das Angebot an Cloud-Ressourcen an die Nachfrage anzupassen und eine Überbereitstellung bei Ihren Workloads zu vermeiden.

Typische Anti-Muster:

- Sie skalieren Ihre Infrastruktur nicht mit der Benutzerlast.
- Sie skalieren Ihre Infrastruktur stets manuell.
- Sie belassen die erhöhte Kapazität nach dem Hochskalieren, anstatt wieder herunterzuskalieren.

Vorteile der Einführung dieser bewährten Methode: Das Konfigurieren und Testen der Workload-Elastizität trägt dazu bei, das Angebot an Cloud-Ressourcen effizient an die Nachfrage anzupassen und eine Überbereitstellung von Kapazitäten zu vermeiden. Sie können die Vorteile der Elastizität in der Cloud nutzen, um die Kapazität während und nach Nachfragespitzen automatisch zu skalieren und so sicherzustellen, dass Sie nur die richtige Anzahl von Ressourcen nutzen, die für die Erfüllung Ihrer Geschäftsanforderungen erforderlich ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Cloud bietet Ihnen die Flexibilität, Ressourcen dynamisch durch verschiedene Mechanismen zu erweitern oder zu reduzieren, um einem veränderten Bedarf gerecht zu werden. Eine optimale Abstimmung von Angebot und Nachfrage führt zu den geringsten Auswirkungen auf die Umgebung für einen bestimmten Workload.

Die Nachfrage kann fest oder variabel sein und erfordert Metriken und Automatisierung, um sicherzustellen, dass die Verwaltung nicht zur Last wird. Anwendungen können vertikal (nach oben oder unten) skaliert werden, indem die Instance-Größe geändert wird, horizontal (nach innen oder außen), indem die Anzahl der Instances geändert wird, oder eine Kombination aus beidem.

Sie können verschiedene Ansätze nutzen, um das Angebot an Ressourcen auf die Nachfrage abzustimmen.

- Zielverfolgungsansatz: Überwachen Sie Ihre Skalierungsmetriken und erhöhen oder verringern Sie die Kapazität automatisch Ihrem Bedarf entsprechend.
- Prädiktives Skalieren: Skalieren Sie entsprechend der erwarteten täglichen und wöchentlichen Entwicklungen.
- Zeitplanbasierter Ansatz: Legen Sie Ihren eigenen Skalierungsplan entsprechend den vorhersehbaren Auslastungsänderungen fest.
- Service-Skalierung: Wählen Sie Services (wie Serverless), die nativ planmäßig skalierbar sind oder das Auto-Scaling als Funktion bieten.

Identifizieren Sie Zeiträume mit geringer oder gar keiner Nutzung und skalieren Sie Ressourcen, um überschüssige Kapazitäten zu entfernen und die Effizienz zu verbessern.

Implementierungsschritte

- Elastizität ermöglicht das Anpassen der verfügbaren Ressourcen an den Bedarf. Instances, Container und Funktionen bieten Mechanismen für Elastizität, entweder in Kombination mit Auto-Scaling oder als Funktion des Services. AWS bietet eine Reihe von Mechanismen für das Auto-Scaling, um sicherzustellen, dass Workloads in Zeiten geringer Benutzerlast schnell und einfach herunterskaliert werden können. Hier sind einige Beispiele für Auto-Scaling-Mechanismen:

Auto scaling mechanism	Where to use
Amazon EC2 Auto Scaling	Verwenden Sie diesen Mechanismus, um zu überprüfen, ob Sie die richtige Anzahl an Amazon EC2-Instances zur Verfügung haben, um die Benutzerlast für Ihre Anwendung zu bewältigen.
Application Auto Scaling	Verwenden Sie diesen Mechanismus,um die Ressourcen für einzelne AWS-Services über Amazon EC2 hinaus automatisch zu skalieren, z. B. Lambda-Funktionen oder Amazon Elastic Container Service (Amazon ECS)-Services.
Kubernetes Cluster Autoscaler	Verwenden Sie diesen Mechanismus, um Kubernetes-Cluster in AWS automatisch zu skalieren.

- Das Skalieren wird häufig im Zusammenhang mit Datenverarbeitungsservices wie Amazon EC2-Instances oder AWS Lambda-Funktionen genannt. Ziehen Sie die Konfiguration von nicht Daten verarbeitenden Services wie Amazon DynamoDB-Lese- und Schreibkapazitätseinheiten oder Amazon Kinesis Data Streams-Shards in Betracht, um die Nachfrage zu decken.
- Prüfen Sie, ob die Metriken zum Hoch- oder Herunterskalieren für die jeweilige Art des bereitgestellten Workloads überprüft werden. Wenn Sie eine Anwendung zur Video-Transkodierung bereitstellen, wird eine CPU-Auslastung von 100 % erwartet, weshalb dies nicht die Hauptmetrik sein sollte. Sie können bei Bedarf eine benutzerdefinierte Metrik (z. B. die Speichernutzung) für Ihre Skalierungsrichtlinie verwenden. Beachten Sie zur Auswahl der geeigneten Metriken die folgenden Hinweise zu Amazon EC2:

  - Es sollte sich um eine gültige Nutzungsmetrik handeln, die beschreibt, wie stark eine Instance genutzt wird.
  - Der Metrikwert muss proportional zur Anzahl der Instances in der Auto Scaling-Gruppe steigen oder sinken.
- Verwenden Sie für Ihre Auto Scaling-Gruppe eine dynamische Skalierung anstelle einer manuellen Skalierung. Wir empfehlen außerdem, dass Sie bei der dynamischen Skalierung Richtlinien zur Zielverfolgung verwenden.
- Stellen Sie sicher, dass Workload-Bereitstellungen sowohl Hoch- als auch Herunterskalierungsereignisse verarbeiten können. Erstellen Sie Testszenarien für Herunterskalierungsereignisse, um zu überprüfen, ob sich der Workload wie erwartet verhält und die Benutzererfahrung nicht beeinträchtigt (z. B. Verlust von Sticky Sessions). Sie können die Aktivitätshistorie verwenden, um eine Skalierungsaktivität für eine Auto Scaling-Gruppe zu überprüfen.
- Evaluieren Sie Ihren Workload auf vorhersagbare Muster und skalieren Sie proaktiv, wenn Sie vorhergesagte und geplante Änderungen der Nachfrage erwarten. Mit der prädiktiven Skalierung können Sie die Notwendigkeit einer Überbereitstellung von Kapazität vermeiden. Weitere Einzelheiten finden Sie unter Prädiktive Skalierung mit Amazon EC2 Auto Scaling.

Ressourcen

Zugehörige Dokumente:

- Erste Schritte mit Amazon EC2 Auto Scaling
- Prädiktive Skalierung für EC2, unterstützt von Machine Learning
- Analyse des Benutzerverhaltens mit Amazon OpenSearch Service, Amazon Data Firehose und Kibana
- Was ist Amazon CloudWatch?
- Überwachen der DB-Last mit Performance Insights auf Amazon RDS
- Vorstellung von nativer Unterstützung für die prädiktive Skalierung mit Amazon EC2 Auto Scaling
- Vorstellung von Karpenter – Open-Source-Kubernetes-Cluster-Autoscaler mit hoher Leistung
- Detaillierte Einblicke in Amazon ECS Cluster Auto Scaling

Zugehörige Videos:

- AWS re:Invent 2023 – Skalierung in AWS für die ersten 10 Millionen Benutzer:innen

- AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft
- AWS re:Invent 2022 – Entwickeln einer kosten-, energie- und ressourceneffizienten Computing-Umgebung
- AWS re:Invent 2022 – Container-Skalierung von einem/einer Benutzer:in auf mehrere Millionen
- AWS re:Invent 2023 – Skalierung der FM-Inferenz auf Hunderte von Modellen mit Amazon SageMaker AI
- AWS re:Invent 2023 – Nutzung der Leistungsfähigkeit von Karpenter für die Skalierung, Optimierung und Aktualisierung von Kubernetes

Zugehörige Beispiele:

- AutoScaling

SUS02-BP02 Ausrichten von SLAs an Nachhaltigkeitszielen

Überprüfen und optimieren Sie die Service Level Agreements (SLA) für Workloads auf der Grundlage Ihrer Nachhaltigkeitsziele, um die für die Unterstützung Ihres Workloads erforderlichen Ressourcen zu minimieren und gleichzeitig die Geschäftsanforderungen zu erfüllen.

Typische Anti-Muster:

- Workload-SLAs sind unbekannt oder nicht eindeutig.
- Sie definieren Ihre SLA nur für Verfügbarkeit und Leistung.
- Sie verwenden die gleichen Designmuster (wie Multi-AZ-Architektur) für alle Ihre Workloads.

Vorteile der Einführung dieser bewährten Methode: Die Abstimmung von SLAs mit Nachhaltigkeitszielen führt zu einer optimalen Ressourcennutzung bei gleichzeitiger Erfüllung der Geschäftsanforderungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

SLAs definieren das von einem Cloud-Workload erwartete Serviceniveau, z. B. Antwortzeit, Verfügbarkeit und Datenaufbewahrung. Sie beeinflussen die Architektur, die Ressourcennutzung und die Umweltauswirkungen eines Cloud-Workloads. Prüfen Sie regelmäßig die SLAs und gehen

Sie Kompromisse ein, indem Sie die Ressourcennutzung in akzeptabler Weise verringern, um Auswirkungen auf die Nachhaltigkeit zu reduzieren.

Implementierungsschritte

- Nachhaltigkeitsziele verstehen: Identifizieren Sie Nachhaltigkeitsziele in Ihrer Organisation, z. B. Reduzierung des CO2-Ausstoßes oder Verbesserung der Ressourcennutzung.
- SLAs überprüfen: Bewerten Sie Ihre SLAs, um festzustellen, ob sie Ihre Geschäftsanforderungen erfüllen. Wenn Sie die SLAs überschreiten, führen Sie eine weitere Überprüfung durch.
- Kompromisse verstehen: Machen Sie sich ein Bild von den Kompromissen in Bezug auf die Komplexität Ihrer Workloads (z. B. hohe Anzahl gleichzeitiger Benutzer:innen), Leistung (z. B. Latenz) und Auswirkungen auf die Nachhaltigkeit (z. B. benötigte Ressourcen). In der Regel geht die Priorisierung von zwei der Faktoren auf Kosten des dritten.
- SLAs anpassen: Passen Sie Ihre SLAs an, indem Sie Kompromisse eingehen, die die Service Level angemessen verringern und somit die Auswirkungen auf die Nachhaltigkeit reduzieren.
  - Nachhaltigkeit und Zuverlässigkeit: Workloads mit hoher Verfügbarkeit verbrauchen in der Regel mehr Ressourcen.
  - Nachhaltigkeit und Leistung: Der Einsatz von mehr Ressourcen zur Leistungssteigerung könnte die Umwelt stärker belasten.
  - Nachhaltigkeit und Sicherheit: Übermäßig sichere Workloads könnten die Umwelt stärker belasten..
- Nachhaltigkeits-SLAs definieren, wenn möglich: Fügen Sie Nachhaltigkeits-SLAs für Ihr Workload hinzu. Definieren Sie beispielsweise ein Mindestnutzungsniveau als Nachhaltigkeits-SLA für Ihre Computing-Instances.
- Effiziente Designmuster verwenden: Nutzen Sie Designmuster wie Microservices auf AWS, die geschäftskritische Funktionen priorisieren, und lassen Sie für nicht kritische Funktionen niedrigere Service Level zu (z. B. für Reaktions- und Wiederherstellungszeiten).
- Verantwortlichkeiten festlegen und kommunizieren: Teilen Sie die SLAs mit allen relevanten Stakeholdern, einschließlich Ihres Entwicklungsteams und Ihrer Kunden. Verwenden Sie Berichte, um die SLAs zu verfolgen und zu überwachen. Weisen Sie Verantwortlichkeiten zu, um die Nachhaltigkeitsziele Ihrer SLAs zu erreichen.
- Anreize und Prämien nutzen: Nutzen Sie Anreize und Prämien, um SLAs zu erreichen oder zu übertreffen, die mit den Nachhaltigkeitszielen übereinstimmen.
- Überprüfen und Wiederholen: Überprüfen Sie Ihre SLAs und passen Sie sie regelmäßig an, damit sie mit den sich entwickelnden Nachhaltigkeits- und Leistungszielen übereinstimmen.

Ressourcen

Zugehörige Dokumente:

- Resilienzmuster und Kompromisse verstehen, um eine effiziente Architektur in der Cloud zu entwickeln
- Bedeutung von Dienstleistungsvereinbarungen für SaaS-Anbieter

Zugehörige Videos:

- AWS re:Invent 2023 – Kapazität, Verfügbarkeit, Kosteneffizienz: Wählen Sie drei Optionen aus
- AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft
- AWS re:Invent 2023 – Fortschrittliche Integrationsmuster und Kompromisse für lose gekoppelte Systeme
- AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen
- AWS re:Invent 2022 – Entwickeln einer kosten-, energie- und ressourceneffizienten Computing-Umgebung

SUS02-BP03 Beenden der Erstellung und Wartung nicht verwendeter Komponenten

Nehmen Sie nicht verwendete Ressourcen in Ihrem Workload außer Betrieb, um die Anzahl der Cloud-Ressourcen zu verringern, die zur Unterstützung Ihres Bedarfs und zur Minimierung von Verschwendung erforderlich sind.

Typische Anti-Muster:

- Sie analysieren Ihre Anwendung nicht auf Ressourcen, die redundant sind oder nicht mehr benötigt werden.
- Sie entfernen keine redundanten oder nicht mehr benötigten Ressourcen.

Vorteile der Nutzung dieser bewährten Methode: Das Entfernen nicht genutzter Ressourcen setzt Kapazitäten frei und verbessert die allgemeine Effizienz des Workloads.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Nicht verwendete Ressourcen verbrauchen Cloud-Kapazitäten wie Speicherplatz oder Rechenleistung. Wenn Sie solche Ressourcen identifizieren und eliminieren, können Sie diese Kapazitäten freisetzen, was zu einer effizienteren Cloud-Architektur führt. Analysieren Sie Anwendungsressourcen (wie vorab kompilierte Berichte, Datensätze, statische Bilder) sowie Zugriffsmuster für Komponenten, um Redundanzen, eine zu geringe Auslastung und mögliche Kandidaten für die Außerbetriebnahme zu identifizieren. Entfernen Sie diese redundanten Ressourcen, um die Ressourcenverschwendung in Ihrem Workload zu reduzieren.

Implementierungsschritte

- Bestandsaufnahme durchführen: Führen Sie eine umfassende Bestandsaufnahme durch, um alle Komponenten innerhalb Ihres Workload zu identifizieren.
- Nutzung analysieren: Verwenden Sie die kontinuierliche Überwachung, um statische Komponenten zu identifizieren, die nicht mehr benötigt werden.
- Ungenutzte Komponenten entfernen: Entwickeln Sie einen Plan, um Komponenten zu entfernen, die nicht mehr benötigt werden.
  - Prüfen Sie vor dem Entfernen einer Ressource die Auswirkungen dieser Maßnahme auf die Architektur.
  - Konsolidieren Sie sich überschneidende generierte Komponenten, um eine redundante Verarbeitung zu entfernen.
  - Aktualisieren Sie Ihre Anwendungen, damit diese nicht mehr benötigte Ressourcen nicht weiter produzieren und speichern.
- Mit Dritten kommunizieren: Weisen Sie Dritte an, die Erstellung und Speicherung von Komponenten einzustellen, die in Ihrem Auftrag verwaltet und nicht mehr benötigt werden. Bitten Sie darum, dass redundante Komponenten konsolidiert werden.
- Lebenszyklusrichtlinien verwenden: Verwenden Sie Lebenszyklusrichtlinien, damit ungenutzte Komponenten automatisch gelöscht werden.
  - Mit Amazon S3-Lebenszyklen können Sie Ihre Objekte während ihres gesamten Lebenszyklus verwalten.
  - Mit Amazon Data Lifecycle Manager lassen sich das Erstellen, Aufbewahren und Löschen von Amazon EBS-Snapshots und Amazon EBS-gestützten AMIs automatisieren.
- Überprüfen und optimieren: Überprüfen Sie regelmäßig Ihren Workload, um ungenutzte Komponenten zu identifizieren und zu entfernen.

Ressourcen

Zugehörige Dokumente:

- Optimieren Ihrer AWS-Infrastruktur für Nachhaltigkeit, Teil II: Speicher
- Wie beende ich aktive Ressourcen, die ich in meinem AWS-Konto nicht mehr benötige?

Zugehörige Videos:

- AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft
- AWS re:Invent 2022 – Bewahrung und Wertmaximierung von digitalen Medienkomponenten mithilfe von Amazon S3
- AWS re:Invent 2023 – Optimieren der Kosten in Ihren Umgebungen mit mehreren Konten

SUS02-BP04 Optimieren der geografischen Platzierung von Workloads auf der Grundlage ihrer Netzwerkanforderungen

Wählen Sie Cloud-Standorte und -Services für Ihren Workload, die die Entfernungen reduzieren, über die Netzwerkdatenverkehr übertragen werden muss, um die Zahl der Netzwerkressourcen zu verringern, die zur Unterstützung Ihres Workloads erforderlich sind.

Typische Anti-Muster:

- Sie wählen die Region des Workloads auf der Grundlage Ihres eigenen Standorts aus.
- Sie konsolidieren alle Workload-Ressourcen an einem geografischen Standort.
- Der gesamte Datenverkehr fließt durch Ihre bestehenden Rechenzentren.

Vorteile der Nutzung dieser bewährten Methode: Die Platzierung von Workloads in der Nähe ihrer Nutzer bietet die geringstmögliche Latenz und verringert gleichzeitig die Bewegung der Daten durch das Netzwerk und damit die Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die AWS Cloud-Infrastruktur basiert auf Standortoptionen wie etwa Regionen, Availability Zones, Platzierungsgruppen und Edge-Standorten wie AWS Outposts und AWS Local Zones. Diese

Standortoptionen stellen die Konnektivität zwischen Anwendungskomponenten, Cloud-Services, Edge-Netzwerken und On-Premises-Rechenzentren sicher.

Analysieren Sie die Netzwerkzugriffsmuster in Ihrem Workload, um festzustellen, wie diese verwendet werden können, um die Entfernungen für den Netzwerkdatenverkehr zu reduzieren.

Implementierungsschritte

- Analysieren Sie die Netzwerkzugriffsmuster in Ihrem Workload, um zu ermitteln, wie die Benutzer Ihre Anwendung verwenden.
  - Verwenden Sie Überwachungstools wie Amazon CloudWatch und AWS CloudTrail, um Daten zu Netzwerkaktivitäten zu erfassen.
  - Analysen Sie die Daten, um das Netzwerkzugriffsmuster zu identifizieren.
- Wählen Sie die Regionen für Ihre Workload-Bereitstellung auf der Grundlage der folgenden zentralen Elemente aus:
  - Ihrem Nachhaltigkeitsziel: wie unter Auswahl von Regionen erläutert.
  - Dem Speicherort Ihrer Daten: Für datenintensive Anwendungen (wie Big Data oder Machine Learning) sollte der Anwendungscode so nahe wie möglich zu den Daten ausgeführt werden.
  - Den Standorten Ihrer Benutzer: Wählen Sie bei nutzerorientierten Anwendungen eine Region (oder Regionen) möglichst nahe an den Benutzern Ihres Workloads.
  - Anderen einschränkenden Faktoren: Denken Sie dabei etwa an Kosten und Compliance, wie in Überlegungen bei der Auswahl einer Region für Ihren Workload beschrieben.
- Verwenden Sie lokale Zwischenspeicherung oder AWS-Zwischenspeicherungslösungen für häufig genutzte Ressourcen zur Verbesserung der Leistung, zur Verringerung von Datenverschiebungen und zur Reduzierung der Umweltauswirkungen.

Service	When to use
Amazon CloudFront	Verwenden Sie dies für die Zwischens peicherung statischer Inhalte wie Bilder, Skripts und Videos sowie dynamischer Inhalte wie API-Antworten oder Webanwendungen.
Amazon ElastiCache	Verwenden Sie dies für die Zwischens peicherung von Inhalten für Webanwend ungen.

Service	When to use
DynamoDB Accelerator	Verwenden Sie dies für die Add-in-Speicher-Beschleunigung für Ihre DynamoDB-Tabellen.

- Nutzen Sie Services, die Ihnen dabei helfen können, Code näher an den Nutzern Ihres Workloads auszuführen:

Service	When to use
Lambda@Edge	Verwenden Sie dies für rechenintensive Anwendungen, die initiiert werden, wenn sich Objekte nicht im Zwischenspeicher befinden.
Amazon CloudFront-Funktionen	Verwenden Sie diese für einfache Anwendung sfälle wie HTTP(s)-Anfragen oder Antwortma nipulationen, die von kurzlebigen Funktionen initiiert werden können.
AWS IoT Greengrass	Verwenden Sie dies für die Ausführung lokaler Rechenoperationen, Messaging sowie die Datenzwischenspeicherung für verbundene Geräte.

- Nutzen Sie Verbindungspooling, um die erneute Nutzung von Verbindungen zu ermöglichen und die Zahl der erforderlichen Ressourcen zu reduzieren.
- Verwenden Sie verteilte Datenspeicher, die nicht auf persistente Verbindungen und synchrone Updates angewiesen sind, um regionale Benutzergruppen zu unterstützen.
- Ersetzen Sie vorab bereitgestellte statische Netzwerkkapazität durch geteilte dynamische Kapazitäten und teilen Sie die Auswirkungen von Netzwerkkapazitäten auf die Nachhaltigkeit mit anderen Abonnenten.

Ressourcen

Zugehörige Dokumente:

- Optimieren Ihrer AWS-Infrastruktur für Nachhaltigkeit, Teil III: Netzwerke
- Amazon ElastiCache-Dokumentation

- Was ist Amazon CloudFront?
- Hauptfunktionen von Amazon CloudFront
- Globale AWS-Infrastruktur
- AWS Local Zones und AWS Outposts: Die Auswahl der richtigen Technologie für Ihren Edge-Workload
- Platzierungsgruppen
- AWS Local Zones
- AWS Outposts

Zugehörige Videos:

- Das Geheimnis der Datenübertragung in AWS lüften
- Skalierung der Netzwerkleistung auf Amazon EC2-Instances der nächsten Generation
- Erklärungsvideo zu AWS Local Zones
- AWS Outposts: Übersicht und Funktionsweise
- AWS re:Invent 2023 – Eine Migrationsstrategie für Edge- und On-Premises-Workloads
- AWS re:Invent 2021 – AWS Outposts: Das AWS-Erlebnis on-premises
- AWS re:Invent 2020 – AWS Wavelength: Apps mit ultraniedriger Latenz am 5G-Edge ausführen
- AWS re:Invent 2022 – AWS Local Zones: Entwickeln von Anwendungen für einen verteilten Edge
- AWS re:Invent 2021 – Entwicklung von Websites mit niedriger Latenz mit Amazon CloudFront
- AWS re:Invent 2022 – Verbessern der Leistung und Verfügbarkeit mit AWS Global Accelerator
- AWS re:Invent 2022 – Aufbau Ihres globalen Wide Area Networks mit AWS
- AWS re:Invent 2020: Globales Datenverkehrsmanagement mit Amazon Route 53

Zugehörige Beispiele:

- AWS Networking-Workshops
- Nachhaltige Architektur — Minimierung des Datenverkehrs zwischen Netzwerken

SUS02-BP05 Optimieren von Ressourcen für Teammitglieder im Hinblick auf die ausgeführten Aktivitäten

Optimieren Sie die Ressourcen, die Teammitgliedern zur Verfügung gestellt werden, um negative Auswirkungen auf die Nachhaltigkeit zu minimieren und gleichzeitig ihre Anforderungen zu erfüllen.

Typische Anti-Muster:

- Sie berücksichtigen nicht die Auswirkungen der von Ihren Teammitgliedern verwendeten Geräte auf die Gesamteffizienz Ihrer Cloud-Anwendung.
- Sie verwalten und aktualisieren die von Team-Mitgliedern verwendeten Ressourcen manuell.

Vorteile der Nutzung dieser bewährten Methode: Die Optimierung der Teammitglieder-Ressourcen verbessert die allgemeine Effizienz Cloud-fähiger Anwendungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Verstehen Sie die Ressourcen, mit denen Ihre Teammitglieder Ihre Services nutzen, deren erwartete Lebensdauer sowie die finanziellen und nachhaltigkeitsbezogenen Auswirkungen. Implementieren Sie Strategien zur Optimierung dieser Ressourcen. Beispielsweise können Sie komplexe Vorgänge wie Rendering und Kompilierung auf intensiv genutzter und skalierbarer Infrastruktur anstatt auf weniger ausgelasteten Einzelbenutzersystemen mit hohem Energieverbrauch ausführen.

Implementierungsschritte

- Energieeffiziente Workstations verwenden: Stellen Sie den Teammitgliedern energieeffiziente Workstations und Peripheriegeräte zur Verfügung. Verwenden Sie effiziente Energiemanagementfeatures (wie den Energiesparmodus) auf diesen Geräten, um ihren Energieverbrauch zu reduzieren.
- Virtualisierung verwenden: Verwenden Sie virtuelle Desktops und Anwendungs-Streaming, um Upgrade- und Geräteanforderungen zu begrenzen.
- Remote-Zusammenarbeit fördern: Ermutigen Sie die Teammitglieder, Tools für die Remote-Zusammenarbeit wie Amazon Chime oder AWS Wickr zu verwenden, um den Reisebedarf und die damit verbundenen CO2-Emissionen zu reduzieren.
- Energieeffiziente Software verwenden: Stellen Sie den Teammitgliedern energieeffiziente Software zur Verfügung, indem Sie nicht benötigte Features und Prozesse entfernen oder deaktivieren.

- Lebenszyklen verwalten: Evaluieren Sie die Auswirkungen von Prozessen und Systemen auf die Lebenszyklen von Geräten. Wählen Sie Lösungen aus, die den Bedarf für Geräteaussetzungen minimieren und gleichzeitig die geschäftlichen Anforderungen erfüllen. Pflegen und aktualisieren Sie regelmäßig Workstations oder Software, um die Effizienz aufrechtzuerhalten und zu verbessern.
- Remote-Verwaltung für Geräte: Implementieren Sie die Remote-Verwaltung für Geräte, um die Anzahl der erforderlichen Geschäftsreisen zu reduzieren.
  - AWS Systems Manager Fleet Manager ist eine vereinheitlichte UI-Umgebung, mit der Sie Ihre auf AWS oder On-Premises ausgeführten Knoten aus der Ferne überwachen können.

Ressourcen

Zugehörige Dokumente:

- Was ist Amazon WorkSpaces?
- Kostenoptimierer für Amazon WorkSpaces
- Amazon AppStream 2.0 Documentation (Dokumentation zu Amazon AppStream 2.0)
- NICE DCV

Zugehörige Videos:

- Verwalten der Kosten für Amazon WorkSpaces in AWS

SUS02-BP06 Implementierung von Pufferung oder Drosselung, um die Bedarfskurve zu verflachen

Pufferung und Drosselung verflachen die Bedarfskurve und reduzieren die erforderliche bereitgestellte Kapazität für Ihr Workload.

Typische Anti-Muster:

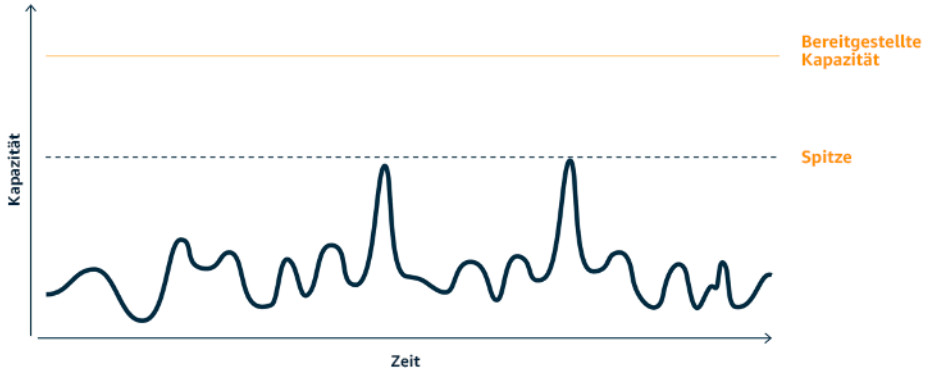
- Sie verarbeiten die Client-Anfragen sofort, obwohl dies nicht erforderlich ist.
- Sie analysieren die Anforderungen für Client-Anfragen nicht.

Vorteile der Nutzung dieser bewährten Methode: Das Verflachen der Bedarfskurve reduziert die erforderliche bereitgestellte Kapazität für den Workload. Die Reduzierung der bereitgestellten Kapazität bedeutet geringeren Energieverbrauch und geringere Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

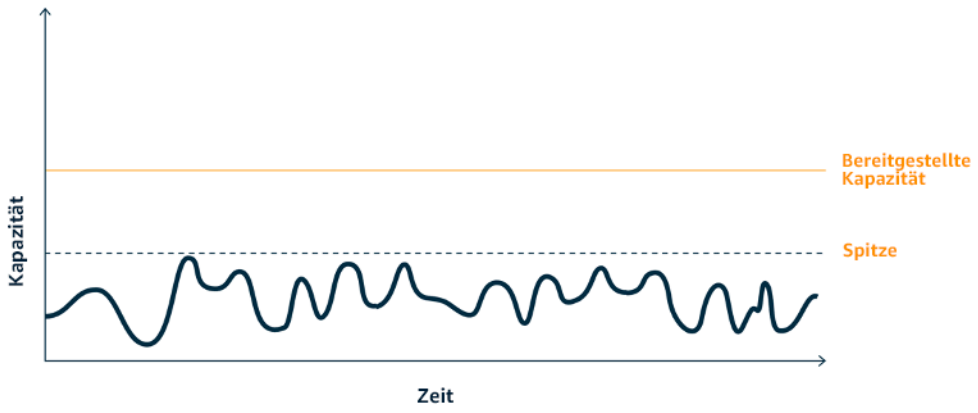
Implementierungsleitfaden

Die Verflachung der Bedarfskurve kann Ihnen dabei helfen, die bereitgestellte Kapazität für einen Workload zu verringern und dessen Umweltauswirkungen zu reduzieren. Nehmen wir einen Workload mit der nachfolgend gezeigten Bedarfskurve. Dieser Workload hat zwei Spitzen und um damit umzugehen, wird die Ressourcenkapazität bereitgestellt, die hier durch die orangefarbene Linie angezeigt wird. Die für diesen Workload aufgewendeten Ressourcen und die eingesetzte Energie werden nicht durch die Fläche unter der Bedarfskurve, sondern von der Linie für die bereitgestellte Kapazität angezeigt, da für den Umgang mit den beiden Spitzen bereitgestellte Kapazität erforderlich ist.



Bedarfskurve mit zwei deutlichen Spitzen, die hohe bereitgestellte Kapazität erfordern.

Sie können Pufferung oder Drosselung verwenden, um die Bedarfskurve zu beeinflussen und die Spitzen abzumildern, was weniger bereitgestellte Kapazität und einen geringeren Energieverbrauch bedeutet. Implementieren Sie Drosselung, wenn Ihre Clients wiederholte Versuche durchführen können. Implementieren Sie die Pufferung, um die Anforderung zu speichern und die Verarbeitung auf einen späteren Zeitpunkt zu verschieben.



Auswirkungen des Drosselns auf die Bedarfskurve und die bereitgestellte Kapazität.

Implementierungsschritte

- Analysieren Sie die Client-Anfragen, um festzulegen, wie darauf zu reagieren ist. Wichtige Faktoren dabei sind:
  - Kann diese Anfrage in asynchroner Weise verarbeitet werden?
  - Kann der Client die Anfrage erneut versuchen?
- Wenn dies der Fall ist, können Sie Drosselung verwenden, die der Quelle mitteilt, dass wenn sie die Anfrage zum aktuellen Zeitpunkt nicht bedienen kann, es später erneut versucht werden sollte.
  - Sie können Amazon API Gateway verwenden, um Drosselung zu implementieren.
- Für Clients, die Anfragen nicht erneut versuchen können, muss zur Verflachung der Bedarfskurve ein Puffer implementiert werden. Ein Puffer verschiebt die Anforderungsverarbeitung, so dass Anwendungen, die mit unterschiedlichen Raten ausgeführt werden, effektiv kommunizieren können. Bei der Pufferung werden Nachrichten von Produzenten in eine Warteschlange oder einen Stream gestellt. Nachrichten können dadurch von Verbrauchern in der für ihre Geschäftsanforderungen passenden Geschwindigkeit gelesen und verarbeitet werden.
  - Amazon Simple Queue Service (Amazon SQS) ist ein verwalteter Service, der Warteschlangen bietet, die es einem einzelnen Verbraucher ermöglichen, individuelle Nachrichten zu lesen.
  - Amazon Kinesis stellt einen Stream bereit, der es vielen Verbrauchern ermöglicht, dieselben Nachrichten zu lesen.
- Analysieren Sie den Gesamtbedarf, die Änderungsrate und die erforderliche Reaktionszeit, um die korrekte Größe der erforderlichen Drosselung oder des Puffers zu bestimmen.

Ressourcen

Zugehörige Dokumente:

- Erste Schritte mit Amazon SQS
- Anwendungsintegration mit Warteschlangen und Nachrichten
- Verwalten und Überwachen der API-Drosselung in Ihren Workloads
- Drosselung einer mehrstufigen, Multi-Mandanten REST-API in großem Umfang mit API Gateway
- Anwendungsintegration mit Warteschlangen und Nachrichten

Zugehörige Videos:

- AWS re:Invent 2022 – Anwendungsintegrationsmuster für Microservices
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon EC2-Strategien zur Kostenoptimierung
- AWS re:Invent 2023 – Fortschrittliche Integrationsmuster und Kompromisse für lose gekoppelte Systeme

Software und Architektur

Frage

- SUS 3 Wie können Sie Software- und Architekturmuster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

SUS 3 Wie können Sie Software- und Architekturmuster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

Bewährte Methoden

- SUS03-BP01 Optimieren von Software und Architektur für asynchrone und geplante Aufträge
- SUS03-BP02 Entfernen oder Refaktorisieren von Workload-Komponenten mit geringer oder keiner Nutzung
- SUS03-BP03 Optimieren von Codebereichen, die die meiste Zeit oder die meisten Ressourcen verbrauchen
- SUS03-BP04 Optimieren der Auswirkungen auf Geräte und Ausrüstung von Kunden
- SUS03-BP05 Verwenden von Softwaremustern und Architekturen, die Datenzugriffs- und Speichermuster optimal unterstützen

SUS03-BP01 Optimieren von Software und Architektur für asynchrone und geplante Aufträge

Verwenden Sie effiziente Software- und Architekturmuster wie warteschlangenbasierte Systeme, um eine durchgängig hohe Auslastung von bereitgestellten Ressourcen zu erzielen.

Typische Anti-Muster:

- Sie stellen zu viele Ressourcen im Cloud-Workload bereit, um auf unerwartete Nachfragesteigerungen reagieren zu können.
- In Ihrer Architektur werden Absender und Empfänger von asynchronen Nachrichten nicht durch eine Messaging-Komponente entkoppelt.

Vorteile der Nutzung dieser bewährten Methode:

- Durch effiziente Software- und Architekturmuster werden ungenutzte Ressourcen in Ihrem Workload minimiert und die allgemeine Effizienz gesteigert.
- Sie können die Verarbeitung unabhängig vom Empfang asynchroner Nachrichten skalieren.
- Durch eine Messaging-Komponente gelten weniger strenge Verfügbarkeitsanforderungen, die mit weniger Ressourcen erfüllt werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Verwenden Sie effiziente Architekturmuster wie eine ereignisgesteuerte Architektur, die zu einer gleichmäßigen Nutzung der Komponenten führen und die Überbereitstellung in Ihrem Workload

minimieren. Durch die Verwendung effizienter Architekturmuster werden ungenutzte Ressourcen, die aufgrund von Änderungen der Nachfrage im Laufe der Zeit nicht genutzt werden, minimiert.

Analysieren Sie die Anforderungen Ihrer Workload-Komponenten und führen Sie Architekturmuster ein, mit denen die allgemeine Auslastung der Ressourcen gesteigert wird. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden.

Implementierungsschritte

- Analysieren Sie die Nachfrage für Ihren Workload, um zu bestimmen, wie diese erfüllt werden kann.
- Verwenden Sie für Anfragen oder Aufträge, für die keine synchronen Antworten erforderlich sind, warteschlangenbasierte Architekturen und Worker mit automatischer Skalierung, durch die die Auslastung maximiert wird. Hier finden Sie einige Beispiele für Situationen, in denen Sie eine warteschlangenbasierte Architektur in Erwägung ziehen sollten:

Queuing mechanism	Description
AWS Batch-Warteschlangen	AWS Batch-Aufträge werden an eine Auftragswarteschlange gesendet, in der sie bleiben, bis ihre Ausführung in einer Datenverarbeitungsumgebung geplant werden kann.
Amazon Simple Queue Service und Amazon EC2 Spot Instances	Durch das Koppeln von Amazon SQS und Spot Instances lassen sich fehlertolerante und effiziente Architekturen erstellen.

- Verwenden Sie für Anfragen oder Aufträge, die jederzeit verarbeitet werden können, Planungsmechanismen zur Auftragsverarbeitung in Batches, um die Effizienz zu steigern. Hier sind einige Beispiele für Planungsmechanismen in AWS:

Scheduling mechanism	Description
Amazon EventBridge Scheduler	Eine Amazon EventBridge-Funktion, mit der Sie in großem Umfang geplante Aufgaben erstellen, ausführen und verwalten können.

Scheduling mechanism	Description
Zeitbasierte AWS Glue-Pläne	Hiermit definieren Sie einen zeitbasierten Plan für Crawler und Aufträge in AWS Glue.
Geplante Amazon Elastic Container Service (Amazon ECS)-Aufgaben	Amazon ECS unterstützt das Erstellen von geplanten Aufgaben. Bei geplanten Aufgaben werden mit Amazon EventBridge-Regeln Aufgaben nach einem Zeitplan oder als Reaktion auf ein EventBridge-Ereignis ausgeführt.
Instance Scheduler	Konfigurieren Sie Zeitpläne zum Starten und Beenden Ihrer Amazon EC2- und Amazon Relational Database Service-Instances.

- Wenn Sie Abfrage- und Webhook-Mechanismen in Ihrer Architektur verwenden, ersetzen Sie diese durch Ereignisse. Erstellen Sie mit ereignisgesteuerten Architekturen hocheffiziente Workloads.
- Nutzen Sie Serverless on AWS, um eine übermäßige Bereitstellung in einer Infrastruktur zu eliminieren.
- Wählen Sie die richtige Größe für Ihre Architektur, um zu vermeiden, dass ungenutzte Ressourcen auf Eingaben warten.
  - Sie können die Empfehlungen zur Dimensionierung in AWS Cost Explorer oder AWS Compute Optimizer zur Identifizierung von Dimensionierungsmöglichkeiten verwenden.
  - Weitere Informationen finden Sie unter Richtige Dimensionierung: Bereitstellung von Instances, die den Workloads entsprechen.

Ressourcen

Zugehörige Dokumente:

- Was ist Amazon Simple Queue Service?
- Was ist Amazon MQ?
- Scaling based on Amazon SQS (Skalierung auf Grundlage von Amazon SQS)
- Was ist AWS Step Functions?
- Was ist AWS Lambda?

- Using AWS Lambda with Amazon SQS (Verwenden von Lambda mit Amazon SQS)
- Was ist Amazon EventBridge?
- Verwaltung asynchroner Workflows mit einer REST-API

Zugehörige Videos:

- AWS re:Invent 2023 – Auf dem Weg zur ereignisgesteuerten Serverless-Architektur
- AWS re:Invent 2023 – Einsatz von Serverless für eine ereignisgesteuerte Architektur und ein domaingesteuertes Design
- AWS re:Invent 2023 – Fortschrittliche ereignisgesteuerte Muster mit Amazon EventBridge
- AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft
- Asynchrone Nachrichtenmuster | AWS-Ereignisse

Zugehörige Beispiele:

- Ereignisgesteuerte Architektur mit AWS-Graviton-Prozessoren und Amazon EC2-Spot-Instances

SUS03-BP02 Entfernen oder Refaktorisieren von Workload-Komponenten mit geringer oder keiner Nutzung

Entfernen Sie ungenutzte Komponenten, die nicht mehr benötigt werden, und refaktorisieren Sie Komponenten mit geringer Nutzung, um die Verschwendung von Ressourcen zu begrenzen.

Typische Anti-Muster:

- Sie prüfen den Nutzungsgrad der einzelnen Komponenten Ihres Workloads nicht regelmäßig.
- Sie prüfen und analysieren nicht die Empfehlungen von AWS-Dimensionierungstools wie etwa AWS Compute Optimizer.

Vorteile der Nutzung dieser bewährten Methode: Das Entfernen nicht genutzter Komponenten minimiert Ausschuss und verbessert die allgemeine Effizienz Ihres Workloads.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Prüfen Sie Ihren Workload, um nicht oder wenig genutzte Komponenten zu identifizieren. Dies ist ein sich wiederholender Verbesserungsprozess, der von Änderungen beim Bedarf oder der Einführung eines neuen Cloud-Services ausgelöst werden kann. Beispielsweise kann ein deutliches Zurückgehen der Laufzeit der AWS Lambda-Funktion darauf hindeuten, dass die Speichergröße reduziert werden muss. Oder wenn AWS neue Services und Funktionen veröffentlicht, können sich die optimalen Services und die Architektur für Ihren Workload ändern.

Überwachen Sie kontinuierlich die Workload-Aktivität und suchen Sie nach Möglichkeiten zur Verbesserung des Nutzungsgrads einzelner Komponenten. Wenn Sie nicht genutzte Komponenten entfernen und Dimensionierungsaktivitäten durchführen, erreichen Sie Ihre geschäftlichen Ziele mit der geringstmöglichen Menge von Cloud-Ressourcen.

Implementierungsschritte

- Machen Sie eine Bestandsaufnahme Ihrer AWS-Ressourcen. In AWS können Sie AWS Resource Explorer einschalten, um Ihre AWS-Ressourcen zu erkunden und zu organisieren. Weitere Informationen finden Sie unter AWS re:Invent 2022 – Verwalten von Ressourcen und Anwendungen im großen Maßstab in AWS.
- Überwachen und erfassen Sie die Nutzungsmetriken für kritische Komponenten Ihres Workloads (etwa CPU-Nutzung, Speichernutzung oder Netzwerkdurchsatz in Amazon CloudWatch-Metriken).
- Identifizieren Sie ungenutzte oder zu wenig genutzte Komponenten in Ihrer Architektur.
  - Prüfen Sie für stabile Workloads regelmäßig AWS-Dimensionierungstools wie AWS Compute Optimizer, um nicht oder wenig genutzte Komponenten zu identifizieren.
  - Prüfen Sie für kurzzeitige Workloads die Nutzungsmetriken, um nicht oder wenig genutzte Komponenten zu identifizieren.
- Nehmen Sie nicht mehr benötigte und dazugehörige Ressourcen (wie etwa Amazon ECR-Images) außer Betrieb.
  - Automatische Bereinigung von nicht verwendeten Images in Amazon ECR
  - Löschen von ungenutzten Amazon Elastic Block Store (Amazon EBS)-Volumes mit AWS Config und AWS Systems Manager
- Konsolidieren oder refaktorisieren Sie nicht ausreichend genutzte Ressourcen mit anderen Ressourcen, um die Nutzungseffizienz zu verbessern. Sie können beispielsweise mehrere kleine Datenbanken auf einer einzelnen Amazon RDS-Datenbank-Instance bereitstellen, anstatt Datenbanken auf einzelnen sehr wenig ausgenutzten Instances auszuführen.

- Verstehen Sie die Ressourcen, die Ihr Workload für die Durchführung einer Arbeitseinheit bereitstellt.

Ressourcen

Zugehörige Dokumente:

- AWS Trusted Advisor
- Was ist Amazon CloudWatch?
- Richtige Dimensionierung: Bereitstellen von Instances entsprechend den Workloads
- Kostenoptimierung mit Empfehlungen zur richtigen Dimensionierung

Zugehörige Videos:

- AWS re:Invent 2023 – Kapazität, Verfügbarkeit, Kosteneffizienz: Wählen Sie drei Optionen aus

Zugehörige Beispiele:

- Optimieren von Hardwaremustern und Beobachtung von Nachhaltigkeits-KPIs

SUS03-BP03 Optimieren von Codebereichen, die die meiste Zeit oder die meisten Ressourcen verbrauchen

Optimieren Sie den Code, der innerhalb der verschiedenen Komponenten Ihrer Architektur ausgeführt wird, um die Ressourcennutzung zu minimieren und die Leistung zu maximieren.

Typische Anti-Muster:

- Sie versäumen die Optimierung Ihres Codes für die Ressourcennutzung.
- Sie reagieren auf Leistungsprobleme normalerweise mit Erhöhung des Ressourceneinsatzes.
- Ihr Code-Prüfungs- und -Entwicklungsprozess verfolgt keine Leistungsänderungen.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung effizienten Codes minimiert die Ressourcennutzung und verbessert die Leistung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Es ist sehr wichtig, jeden funktionalen Bereich, einschließlich des Codes einer für die Cloud erstellten Anwendung, zu untersuchen, um ihre Ressourcennutzung und Leistung zu optimieren. Überwachen Sie kontinuierlich die Leistung Ihres Workloads in Build-Umgebungen und Produktionsbereichen und suchen Sie nach Möglichkeiten, Code-Snippets zu verbessern, die einen besonders hohen Ressourcenverbrauch haben. Führen Sie einen regelmäßigen Prüfungsprozess ein, um Fehler oder Anti-Muster in Ihrem Code zu identifizieren, die Ressourcen in ineffizienter Weise nutzen. Nutzen Sie einfache und effiziente Algorithmen, die dieselben Ergebnisse für Ihre Anwendungsfälle liefern.

Implementierungsschritte

- Effiziente Programmiersprache verwenden: Verwenden Sie das jeweils effizienteste Betriebssystem und die optimale Programmiersprache für den Workload. Für Informationen zu energieeffizienten Programmiersprachen (einschließlich Rust) vgl. Nachhaltigkeit mit Rust.
- KI-Programmierungsbegleiter verwenden: Erwägen Sie die Verwendung eines Begleiters zur KI-Programmierung wie Amazon CodeWhisperer, um Code effizient zu schreiben.
- Code-Überprüfungen automatisieren: Führen Sie bei der Entwicklung Ihrer Workloads einen automatischen Code-Prüfungsprozess ein, um die Qualität zu verbessern sowie Fehler und Anti-Muster zu identifizieren.
  - Automatisieren von Code-Reviews mit Amazon CodeGuru Reviewer
  - Erkennen von Concurrency-Fehlern mit Amazon CodeGuru
  - Verbessern der Codequalität für Python-Anwendungen mit Amazon CodeGuru
- Code-Profiler verwenden: Verwenden Sie einen Code-Profiler für Code-Prüfungen, um die Codebereiche als Optimierungsziele zu identifizieren, die die meiste Zeit oder die meisten Ressourcen verwenden.
  - Reduzieren des CO2-Fußabdrucks Ihrer Organisation mit Amazon CodeGuru Profiler
  - Verständnis der Speichernutzung in Ihrer Java-Anwendung mit Amazon CodeGuru Profiler
  - Verbessern des Kundenkomforts und Senken von Kosten mit Amazon CodeGuru Profiler
- Überwachen und optimieren: Verwenden Sie Ressourcen für die kontinuierliche Überwachung, um Komponenten mit hohem Ressourcenbedarf oder suboptimaler Konfiguration zu identifizieren.
  - Ersetzen Sie rechenintensive Algorithmen durch einfachere und effizientere Versionen, die dieselben Ergebnisse liefern.
  - Entfernen Sie unnötigen Code und überflüssige Formatierungen.

- Code-Refactoring oder -Transformation verwenden: Erkunden Sie die Möglichkeiten der Amazon-Q-Codetransformation für die Wartung und Aktualisierung von Anwendungen.
  - Sprachversionen mit Amazon-Q-Codetransformation upgraden
  - AWS re:Invent 2023 – Automatisierung der App-Upgrades und Wartung mithilfe von Amazon-Q-Codetransformation

Ressourcen

Zugehörige Dokumente:

- Was ist Amazon CodeGuru Profiler?
- FPGA-Instances
- Die AWS-SDKs in Tools zum Entwickeln in AWS

Zugehörige Videos:

- Verbessern der Code-Effizienz mit Amazon CodeGuru Profiler
- AWS re:Invent 2023 – Bewährte Methoden für Amazon CodeWhisperer
- Automatisieren von Codeprüfungen und Empfehlungen zur Anwendungsleistung mit Amazon CodeGuru

Zugehörige Beispiele:

- Code optimieren mit Amazon CodeGuru

SUS03-BP04 Optimieren der Auswirkungen auf Geräte und Ausrüstung von Kunden

Verstehen Sie die in Ihrer Architektur verwendeten Geräte und nutzen Sie Strategien, um ihre Nutzung zu reduzieren. Dies kann die Umweltauswirkungen Ihres Cloud-Workloads insgesamt verringern.

Typische Anti-Muster:

- Sie ignorieren die Umweltauswirkungen der Geräte, die Ihre Kunden verwenden.
- Sie verwalten und aktualisieren die von Kunden verwendeten Ressourcen manuell.

Vorteile der Nutzung dieser bewährten Methode: Die Implementierung von Softwaremustern und Funktionen, die für Kundengeräte optimiert sind, können die Umweltauswirkungen des Cloud-Workloads insgesamt verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Implementierung für Kundengeräte optimierter Softwaremuster und Funktionen können die Umweltauswirkungen auf unterschiedliche Weise reduzieren:

- Die Implementierung neuer abwärtskompatibler Funktionen kann die Anzahl der Hardwareaustauschvorgänge verringern.
- Die Optimierung einer Anwendung, so dass sie effizient auf Geräten ausgeführt werden kann, kann bei der Reduzierung des Energieverbrauchs helfen und die Batterielaufzeit verlängern (falls Batterien zum Einsatz kommen).
- Die Optimierung einer Anwendung für Geräte kann auch Datenübertragungen über das Netzwerk verringern.

Verstehen Sie die in Ihrer Architektur verwendeten Geräte, ihre erwartete Lebensdauer und die Auswirkungen des Austauschs dieser Komponenten. Implementieren Sie Softwaremuster und Funktionen, die dabei helfen, den Energieverbrauch von Geräten zu senken, und den Austausch von Geräten sowie manuelle Upgrades durch Kunden seltener erforderlich machen.

Implementierungsschritte

- Bestandsaufnahme durchführen:Inventarisieren Sie die in ihrer Architektur verwendeten Geräte. Dabei kann es sich um Mobilgeräte, Tablets, IOT-Geräte, Smart Light- oder auch Smartgeräte in einer Fabrik handeln.
- Energieeffiziente Geräte verwenden: Erwägen Sie den Einsatz energieeffizienter Geräte in Ihrer Architektur. Verwenden Sie Energieverwaltungskonfigurationen auf Geräten, um in den Energiesparmodus zu wechseln, wenn sie nicht verwendet werden.
- Effiziente Anwendungen ausführen: Optimieren Sie die Anwendung, die auf den Geräten ausgeführt wird:
  - Verwenden Sie Strategien wie die Ausführung von Aufgaben im Hintergrund, um den Energieverbrauch zu verringern.

  - Berücksichtigen Sie beim Erstellen von Nutzlasten Netzwerkbandbreite und Latenz und implementieren Sie Funktionen, mit denen Ihre Anwendungen auch über Verbindungen mit geringer Bandbreite und hoher Latenz gut funktionieren.
  - Wandeln Sie Payloads und Dateien in von den Geräten benötigte optimierte Formate um. Sie können beispielsweise Amazon Elastic Transcoder oder AWS Elemental MediaConvert verwenden, um große, qualitativ hochwertige Digitalmediendateien in Formate umzuwandeln, die Benutzer auf Mobilgeräten abspielen können.
  - Führen Sie rechenintensive Aktivitäten (z. B. das Rendern von Bildern) serverseitig aus oder nutzen Sie Anwendungs-Streaming, um den Benutzerkomfort auf älteren Geräten zu verbessern.
  - Segmentieren und paginieren Sie Ausgaben, besonders für interaktive Sitzungen, um Nutzlasten zu verwalten und lokale Speicheranforderungen zu begrenzen.
- Anbieter einbeziehen: Arbeiten Sie mit Geräteanbietern zusammen, die nachhaltige Materialien verwenden und für Transparenz in ihren Lieferketten und Umweltzertifizierungen sorgen.
- Over-the-Air (OTA)-Updates verwenden: Verwenden Sie den automatisierten Over-the-Air (OTA)-Mechanismus, um Updates auf einem oder mehreren Geräten bereitzustellen.
  - Mit einer CI/CD-Pipeline können Sie mobile Anwendungen aktualisieren.
  - Mit AWS IoT Device Management können Sie verbundene Geräte in großem Umfang aus der Ferne verwalten.
- Verwaltete Gerätefarmen verwenden: Verwenden Sie zum Testen neuer Features und Updates verwaltete Gerätefarmen mit repräsentativen Sätzen von Hardwaregeräten, um den Umfang der unterstützten Geräte zu maximieren. Weitere Informationen finden Sie in SUS06-BP04 Verwenden verwalteter Gerätefarmen für Tests.
- Kontinuierliche Überwachung und Verbesserung: Verfolgen Sie den Energieverbrauch von Geräten, um Verbesserungsmöglichkeiten zu identifizieren. Verwenden Sie neue Technologien oder bewährte Methoden, um die Umweltauswirkungen dieser Geräte zu verbessern.

Ressourcen

Zugehörige Dokumente:

- Was ist AWS Device Farm?
- AppStream 2.0-Dokumentation
- NICE DCV
- OTA-Tutorial zur Aktualisierung der Firmware auf Geräten mit FreeRTOS

- Optimierung Ihrer IoT-Geräte für ökologische Nachhaltigkeit

Zugehörige Videos:

- AWS re:Invent 2023 – Verbessern Sie die Qualität Ihrer Mobil- und Web-Apps mit AWS Device Farm

SUS03-BP05 Verwenden von Softwaremustern und Architekturen, die Datenzugriffs- und Speichermuster optimal unterstützen

Identifizieren Sie, wie Daten in Ihrem Workload verwendet, von Benutzern genutzt, übertragen und gespeichert werden. Verwenden Sie Softwaremuster und Architekturen, die den Datenzugriff und die Speicherung optimal unterstützen, um die zur Unterstützung des Workloads erforderlichen Computing-, Netzwerk- und Speicherressourcen zu reduzieren.

Typische Anti-Muster:

- Sie gehen davon aus, dass für alle Workloads ähnliche Datenspeicher- und Zugriffsmuster gelten.
- Sie verwenden nur eine Speicherebene, vorausgesetzt, dass alle Workloads in diese Ebene passen.
- Sie gehen davon aus, dass Datenzugriffsmuster im Laufe der Zeit konsistent bleiben.
- Ihre Architektur unterstützt potenzielle hohe Bursts beim Datenzugriff, was dazu führt, dass die Ressourcen die meiste Zeit ungenutzt bleiben.

Vorteile der Nutzung dieser bewährten Methode: Die Auswahl und Optimierung Ihrer Architektur auf der Grundlage von Datenzugriffs- und Speichermustern hilft bei der Reduzierung der Entwicklungskomplexität und der Steigerung der allgemeinen Nutzung. Das Verständnis, wann globale Tabellen, Datenpartitionen und Caching verwendet werden sollen, hilft Ihnen dabei, den Betriebsaufwand zu verringern und basierend auf Ihren Workload-Anforderungen zu skalieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Verwenden Sie Software- und Architekturmuster, die optimal zu den Eigenschaften Ihrer Daten und den Zugriffsmustern passen. Verwenden Sie etwa eine moderne Datenarchitektur auf AWS, die die Nutzung speziell erstellter Services ermöglicht, die für Ihre ganz speziellen Analyseanwendungsfälle

optimiert sind. Diese Architekturmuster ermöglichen die effiziente Datenverarbeitung und verringern die Ressourcennutzung.

Implementierungsschritte

- Analysieren Sie die Eigenschaften ihrer Daten und Ihre Zugriffsmuster, um die korrekte Konfiguration für Ihre Cloud-Ressourcen zu identifizieren. Zu den berücksichtigenden Schlüsselmerkmalen gehören:
  - Datentyp: strukturiert, semistrukturiert, unstrukturiert
  - Datenwachstum: begrenzt, unbegrenzt
  - Lebensdauer von Daten: anhaltend, flüchtig, vorübergehend
  - Zugriffsmuster: Lese- oder Schreibzugriff, Häufigkeit von Aktualisierungen, schwankend oder konsistent
- Verwenden Sie Architekturmuster, die Datenzugriffs- und Speichermuster optimal unterstützen.
  - Muster zur Aktivierung der Datenpersistenz
  - Let's Architect! Moderne Datenarchitekturen
  - Datenbanken auf AWS: Das richtige Tool für jede Aufgabe
- Nutzen Sie Technologien, die nativ mit komprimierten Daten funktionieren.
  - Athena Komprimierungs-Support-Dateiformate
  - Formatierungsoptionen für ETL-Eingaben und -Ausgaben in AWS Glue
  - Laden komprimierter Datendateien aus Amazon S3 mit Amazon Redshift
- Verwenden Sie zweckgerichtet erstellte Analyseservices für die Datenverarbeitung in Ihrer Architektur. Ausführlichere Informationen zu speziell entwickelten Analysediensten von AWS finden Sie unter AWS re:Invent 2022 – Erstellen von modernen Datenarchitekturen in AWS.
- Verwenden Sie die Datenbank-Engine, die das dominierende Abfragemuster jeweils am besten unterstützt. Verwalten Sie Ihre Datenbankindizes so, dass sie die effiziente Ausführung von Abfragen unterstützen. Weitere Informationen finden Sie unter AWS-Datenbanken und AWS re:Invent 2022 – Apps mit speziell entwickelten Datenbanken modernisieren.
- Wählen Sie Netzwerkprotokolle aus, die die Menge der genutzten Netzwerkkapazitäten in Ihrer Architektur reduzieren.

Ressourcen

Zugehörige Dokumente:

- COPY aus spaltenbasierten Datenformaten mit Amazon Redshift
- Umwandeln Ihres Eingabedatensatzformats in Firehose
- Verbessern der Abfrageleistung in Amazon Athena durch Umwandlung in Spaltenformate
- Überwachung der DB-Last mit Performance Insights auf Amazon Aurora
- Überwachung der DB-Last mit Performance Insights auf Amazon RDS
- Amazon S3-Intelligent-Tiering-Speicherklasse
- Erstellung eines CQRS-Ereignisspeichers mit Amazon DynamoDB

Zugehörige Videos:

- AWS re:Invent 2022 – Aufbau von Data-Mesh-Architekturen in AWS
- AWS re:Invent 2023 – Vertiefung in Amazon Aurora und seine Innovationen
- AWS re:Invent 2023 – Steigerung der Effizienz von Amazon EBS und der allgemeinen Kosteneffizienz
- AWS re:Invent 2023 – Optimierung der Speicherkosten und der Leistung mit Amazon S3
- AWS re:Invent 2023 – Aufbau und Optimierung eines Data Lake in Amazon S3
- AWS re:Invent 2023 – Fortschrittliche ereignisgesteuerte Muster mit Amazon EventBridge

Zugehörige Beispiele:

- AWS-Workshop „Speziell entwickelte Datenbanken“
- AWS Immersion Day in moderne Datenarchitekturen
- Erstellung eines Data Mesh in AWS

Daten

Frage

- SUS 4 Wie können Sie Datenverwaltungsrichtlinien und -muster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

SUS 4 Wie können Sie Datenverwaltungsrichtlinien und -muster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Implementieren Sie Verfahren für die Datenverwaltung, die den zur Unterstützung Ihres Workloads bereitgestellten Speicher und die für dessen Nutzung erforderlichen Ressourcen reduzieren. Verstehen Sie Ihre Daten und setzen Sie Speichertechnologien und -konfigurationen ein, die den geschäftlichen Mehrwert der Daten und deren Nutzung besser fördern. Verschieben Sie die Daten während des Lebenszyklus zu effizienteren Speichern mit geringerer Leistung, wenn die Anforderungen abnehmen. Löschen Sie Daten, die nicht mehr benötigt werden.

Bewährte Methoden

- SUS04-BP01 Implementieren einer Richtlinie für die Klassifizierung von Daten
- SUS04-BP02 Verwenden von Technologien, die Datenzugriff und Speichermuster unterstützen
- SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien
- SUS04-BP04 Verwendung von Elastizität und Automatisierung zur Erweiterung des Block-Speichers oder des Dateisystems
- SUS04-BP05 Entfernen nicht benötigter oder redundanter Daten
- SUS04-BP06 Verwenden geteilter Dateisysteme oder Objektspeicher für den Zugriff auf allgemeine Daten
- SUS04-BP07 Minimieren von Datenübertragungen zwischen Netzwerken
- SUS04-BP08 Sichern von Daten nur in dem Fall, wenn ihre erneute Erstellung schwierig ist

SUS04-BP01 Implementieren einer Richtlinie für die Klassifizierung von Daten

Klassifizieren Sie die Daten, um zu verstehen, wie wichtig sie für die Geschäftsergebnisse sind, und wählen Sie die richtige energieeffiziente Speicherebene zur Speicherung der Daten.

Typische Anti-Muster:

- Sie identifizieren keine Datenbestände mit ähnlichen Merkmalen (z. B. Sensibilität, Geschäftskritikalität oder gesetzliche Anforderungen), die verarbeitet oder gespeichert werden.
- Sie haben keinen Datenkatalog zur Inventarisierung Ihrer Datenbestände eingeführt.

Vorteile der Nutzung dieser bewährten Methode: Durch die Implementierung einer Datenklassifizierungsrichtlinie können Sie die energieeffizienteste Speicherebene für Daten bestimmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Bei der Datenklassifizierung wird identifiziert, welche Arten von Daten in einem Informationssystem verarbeitet und gespeichert werden, das einer Organisation gehört oder von ihr betrieben wird. Dazu gehört auch die Bestimmung der Kritikalität der Daten und der wahrscheinlichen Auswirkungen von Preisgaben, Verlusten oder Missbrauch von Daten.

Implementieren Sie Richtlinien zur Datenklassifizierung, indem Sie von der kontextuellen Verwendung der Daten ausgehen und ein Kategorisierungsschema erstellen, das den Grad der Kritikalität eines bestimmten Datensatzes für die Abläufe eines Unternehmens berücksichtigt.

Implementierungsschritte

- Bestandsaufnahme vornehmen: Führen Sie eine Bestandsaufnahme der verschiedenen Datentypen durch, die für Ihr Workload vorhanden sind.
- Gruppendaten: Bestimmen Sie die Kritikalität, Vertraulichkeit, Integrität und Verfügbarkeit von Daten auf der Grundlage des Risikos für die Organisation. Verwenden Sie diese Anforderungen, um Daten in eine der von Ihnen gewählten Datenklassifizierungsebenen einzuteilen. Ein Beispiel finden Sie unter Four simple steps to classify your data and secure your startup (Vier einfache Schritte zur Klassifizierung Ihrer Daten und zur Sicherung Ihres Startups).
- Datenklassifizierungsebenen und Richtlinien definieren: Definieren Sie für jede Datengruppe die Datenklassifizierungsebene (z. B. öffentlich oder vertraulich) und die Verarbeitungsrichtlinien. Kennzeichnen Sie Daten entsprechend. Einzelheiten zu den Kategorien für die Datenklassifizierung finden Sie im Data Classification Whitepaper.
- Regelmäßige Überprüfung: Überprüfen und kontrollieren Sie Ihre Umgebung regelmäßig auf nicht markierte und nicht klassifizierte Daten. Verwenden Sie die Automatisierung, um diese Daten zu identifizieren und die Daten entsprechend zu klassifizieren und zu markieren. Ein Beispiel finden Sie unter Datenkatalog und Crawler in AWS Glue.
- Datenkatalog einrichten: Richten Sie einen Datenkatalog mit Prüfungs- und Governance-Funktionen ein.
- Dokumentation: Dokumentieren Sie Datenklassifizierungsrichtlinien und Verarbeitungsverfahren für jede Datenklasse.

Ressourcen

Zugehörige Dokumente:

- Nutzung der AWS Cloud zur Unterstützung der Datenklassifizierung
- Tag-Richtlinien von AWS Organizations

Zugehörige Videos:

- AWS re:Invent 2022 – Mehr Agilität mit Data Governance auf AWS
- AWS re:Invent 2023 – Datenschutz und Ausfallsicherheit mit AWS-Speicher

SUS04-BP02 Verwenden von Technologien, die Datenzugriff und Speichermuster unterstützen

Nutzen Sie Speichertechnologien, die den Zugriff auf Ihre Daten und ihre Speicherung jeweils optimal unterstützen, um die Zahl der bereitgestellten Ressourcen zu minimieren und gleichzeitig den Workload zu unterstützen.

Typische Anti-Muster:

- Sie gehen davon aus, dass für alle Workloads ähnliche Datenspeicher- und Zugriffsmuster gelten.
- Sie verwenden nur eine Speicherebene, vorausgesetzt, dass alle Workloads in diese Ebene passen.
- Sie gehen davon aus, dass Datenzugriffsmuster im Laufe der Zeit konsistent bleiben.

Vorteile der Nutzung dieser bewährten Methode: Die Auswahl und Optimierung Ihrer Speichertechnologien auf der Grundlage von Datenzugriffs- und Speichermustern hilft Ihnen, die erforderlichen Cloud-Ressourcen zu reduzieren, um Ihre Geschäftsanforderungen zu erfüllen und die Gesamteffizienz des Cloud-Workloads zu verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Wählen Sie für maximale Leistungseffizienz die für Ihre Zugriffsmuster geeignete Speicherlösung, oder passen Sie Ihre Zugriffsmuster an die Speicherlösung an.

Implementierungsschritte

- Daten- und Zugriffsmerkmale bewerten: Bewerten Sie Ihre Datenmerkmale und Zugriffsmuster, um die wichtigsten Merkmale Ihres Speicherbedarfs zu erfassen. Zu den berücksichtigenden Schlüsselmerkmalen gehören:

  - Datentyp: strukturiert, semistrukturiert, unstrukturiert
  - Datenwachstum: begrenzt, unbegrenzt
  - Lebensdauer von Daten: anhaltend, flüchtig, vorübergehend
  - Zugriffsmuster: Lese- oder Schreibzugriff, Häufigkeit, schwankend oder konsistent
- Die richtige Speichertechnologie auswählen: Migrieren Sie Daten auf die geeignete Speichertechnologie, die Ihre Datenmerkmale und Zugriffsmuster unterstützt. Hier sind einige Beispiele für AWS-Speichertechnologien und ihre Schlüsselmerkmale:

Type	Technology	Key characteristics
Objektspeicher	Amazon S3	Ein Objektspeicherservice mit unbegrenzter Skalierbarkeit, hoher Verfügbarkeit und mehreren Zugriffsoptionen. Für die Übertragung von Objekten in und aus Amazon S3 und den Zugriff auf diese Objekte können Sie einen Service wie z. B. Transfer Acceleration oder Access Points, um Ihren Standort, Ihre Sicherheitsanforderungen und Zugriffsmuster zu unterstützen.
Archivieren von Speichern	Amazon S3 Glacier	Speicherklasse von Amazon S3 für die Datenarchivierung.
Gemeinsames Dateisystem	Amazon Elastic File System (Amazon EFS)	Mountfähiges Dateisystem, auf das verschiedene Arten von Datenverarbeitungslösungen zugreifen können. Amazon EFS erweitert und verringert den Speicher automatisch und ist leistungs

Type	Technology	Key characteristics
		optimiert, um durchgängig niedrige Latenzen zu bieten.
Gemeinsames Dateisystem	Amazon FSx	basiert auf den neuesten AWS-Datenverarbeitungslösungen und unterstützt vier gängige Dateisysteme: NetApp ONTAP, OpenZFS, Windows File Server und Lustre Die Amazon FSx -Latenz, der Durchsatz und die IOPS variieren je nach Dateisystem und sollten bei der Auswahl des richtigen Dateisystems für Ihre Workload-Anforderungen berücksichtigt werden.
Blockspeicher	Amazon Elastic Block Store (Amazon EBS)	Skalierbarer, hochleistungsfähiger Blockspeicherservice für Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS umfasst SSD-gestützten Speicher für transaktions- und IOPS-intensive Workloads und HDD-gestützten Speicher für durchsatzintensive Workloads.

Type	Technology	Key characteristics
Relationale Datenbank	Amazon Aurora, Amazon RDS, Amazon Redshift	Sie unterstützt AKID-Transaktionen (Atomarität, Konsistenz, Isolation und Dauerhaftigkeit) und gewährleistet die referentielle Integrität sowie eine starke Datenkonsistenz. Bei zahlreichen herkömmlichen Anwendungen, Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) und E-Commerce-Systemen werden relationale Datenbanken zum Speichern der Daten verwendet.
Schlüssel-Werte-Datenbank	Amazon DynamoDB	Für gängige Zugriffsmuster optimiert, üblicherweise zum Speichern und Abrufen großer Datenmengen. Web-Apps mit hohem Datenverkehr, E-Commerce-Systeme und Gaming-Anwendungen sind typische Anwendungsfälle für Schlüssel-Werte-Datenbanken.

- Speicherzuweisung automatisieren: Überwachen Sie bei Speichersystemen mit einer festen Größe, z. B. Amazon EBS oder Amazon FSx, den verfügbaren Speicherplatz und automatisieren die Speicherzuweisung bei Erreichen eines Schwellenwertes. Sie können mithilfe von Amazon CloudWatch verschiedene Metriken für Amazon EBS und Amazon FSx erfassen und analysieren.
- Die richtige Speicherklasse wählen: Wählen Sie die passende Speicherklasse für Ihre Daten.
  - Amazon S3-Speicherklassen können auf Objektebene konfiguriert werden. Ein einzelner Bucket kann Objekte enthalten, die in allen Speicherklassen gespeichert sind.

- Sie können Amazon S3-Lebenszyklusrichtlinien verwenden, um Objekte automatisch zwischen Speicherklassen zu wechseln oder Daten zu entfernen, ohne dass die Anwendung geändert werden muss. Im Allgemeinen müssen Sie bei diesen Speichermechanismen einen Kompromiss zwischen Ressourceneffizienz, Zugriffslatenz und Zuverlässigkeit eingehen.

Ressourcen

Zugehörige Dokumente:

- Amazon EBS-Volume-Typen
- Amazon EC2-Instance-Speicher
- Amazon S3 Intelligent-Tiering
- Amazon EBS-E/A-Merkmale
- Verwenden von Amazon S3-Speicherklassen
- Was ist Amazon S3 Glacier?

Zugehörige Videos:

- AWS re:Invent 2023 – Steigerung der Effizienz von Amazon EBS und der allgemeinen Kosteneffizienz
- AWS re:Invent 2023 – Optimierung der Speicherkosten und der Leistung mit Amazon S3
- AWS re:Invent 2023 – Aufbau und Optimierung eines Data Lake in Amazon S3
- AWS re:Invent 2022 – Erstellen von modernen Datenarchitekturen in AWS
- AWS re:Invent 2022 – Modernisierung von Apps mit speziell entwickelten Datenbanken
- AWS re:Invent 2022 – Aufbau von Data-Mesh-Architekturen in AWS
- AWS re:Invent 2023 – Vertiefung in Amazon Aurora und seine Innovationen
- AWS re:Invent 2023 – Fortschrittliche Datenmodellierung mit Amazon DynamoDB

Zugehörige Beispiele:

- Amazon S3-Beispiele
- AWS-Workshop „Speziell entwickelte Datenbanken“
- Datenbanken für Entwickler

- AWS Immersion Day in moderne Datenarchitekturen
- Erstellung eines Data Mesh in AWS

SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien

Verwalten Sie den Lebenszyklus aller Daten und setzen Sie automatisch Löschen durch, um den für Ihren Workload benötigten Speicher insgesamt zu minimieren.

Typische Anti-Muster:

- Sie löschen Daten manuell.
- Sie löschen keine Workload-Daten.
- Sie verschieben Daten nicht abhängig von den Aufbewahrungs- und Zugriffsanforderungen in energieeffizientere Speicherebenen.

Vorteile der Einführung dieser bewährten Methode: Durch Richtlinien für den Lebenszyklus wird die Effizienz des Datenzugriffs und der Datenaufbewahrung für einen Workload sichergestellt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Datensätze verfügen während ihres Lebenszyklus normalerweise über unterschiedliche Aufbewahrungs- und Zugriffsanforderungen. So kann eine Anwendung z. B. für einen bestimmten Zeitraum häufig Zugriff auf einige Datensätze benötigen. Danach wird nur noch unregelmäßig darauf zugegriffen.

Um Datensätze während ihres Lebenszyklus effizient zu verwalten, konfigurieren Sie Lebenszyklusrichtlinien, d. h. Regeln, die den Umgang mit den Datensätzen definieren.

Mit Lebenszyklus-Konfigurationsregeln können Sie einen bestimmten Speicherservice anweisen, einen Datensatz in energieeffizientere Speicherebenen zu verschieben, ihn zu archivieren oder zu löschen.

Implementierungsschritte

- Klassifizieren Sie die Datensätze in Ihrem Workload.
- Definieren Sie Bearbeitungsverfahren für jede Datenklasse.

- Legen Sie automatisierte Lebenszyklusrichtlinien zur Durchsetzung von Lebenszyklusregeln fest. Hier finden Sie einige Beispiel zum Einrichten von automatisierten Lebenszyklusrichtlinien für unterschiedliche AWS-Speicherservices:

Storage service	How to set automated lifecycle policies
Amazon S3	Mit Amazon S3-Lebenszyklen können Sie Ihre Objekte während ihres gesamten Lebenszyklus verwalten. Wenn die Zugriffsmuster unbekannt oder nicht prognostizierbar sind oder sich ändern, können Sie Amazon S3 Intelligent-Tiering verwenden. Hiermit werden Zugriffsmuster überwacht und Objekte, auf die nicht zugegriffen wurde, automatisch in kostengünstigere Zugriffsebenen verschoben. Anhand von Amazon S3 Storage Lens-Metriken können Sie Optimierungsmöglichkeiten und Lücken im Lebenszyklusmanagement ermitteln.
Amazon Elastic Block Store	Mit Amazon Data Lifecycle Manager lassen sich das Erstellen, Aufbewahren und Löschen von Amazon EBS-Snapshots und Amazon EBS-gestützten AMIs automatisieren.
Amazon Elastic File System	Das Amazon EFS-Lebenszyklusmanagement verwaltet den Dateispeicher für Ihre Dateisysteme automatisch.
Amazon Elastic Container Registry	Amazon ECR-Lebenszyklusrichtlinien automatisieren die Bereinigung von Container-Images, indem Images abhängig von Alter oder Anzahl ablaufen.

Storage service	How to set automated lifecycle policies
AWS Elemental MediaStore	Sie können eine Objektlebenszyklus-Richtlinie verwenden, die steuert, wie lange Objekte im MediaStore-Container gespeichert werden sollen.

- Löschen Sie nicht genutzte Volumes, Snapshots und Daten, deren Aufbewahrungszeitraum abgelaufen ist. Nutzen Sie zum Löschen native Service-Features wie Amazon DynamoDB Gültigkeitsdauer oder Amazon CloudWatch-Protokollaufbewahrung.
- Aggregieren und komprimieren Sie Daten wenn möglich auf der Basis von Lebenszyklusregeln.

Ressourcen

Zugehörige Dokumente:

- Optimieren von Amazon S3-Lebenszyklusregeln mit Amazon S3 Storage Class Analysis
- Evaluieren von Ressourcen mit AWS-Config-Regeln

Zugehörige Videos:

- AWS re:Invent 2021 – Bewährte Methoden für den Amazon S3-Lebenszyklus zur Optimierung Ihrer Speicherausgaben
- AWS re:Invent 2023 – Optimierung der Speicherkosten und der Leistung mit Amazon S3
- Vereinfachen des Datenlebenszyklus und Optimieren von Speicherkosten mit Amazon S3-Lebenszyklen
- Reduzieren von Speicherkosten mit Amazon S3 Storage Lens

SUS04-BP04 Verwendung von Elastizität und Automatisierung zur Erweiterung des Block-Speichers oder des Dateisystems

Verwenden Sie Elastizität und Automatisierung, um den Block-Speicher oder das Dateisystem zu erweitern, wenn das Datenvolumen zunimmt, um den bereitgestellten Gesamtspeicher zu minimieren.

Typische Anti-Muster:

- Sie unterhalten einen großen Block-Speicher oder ein großes Dateisystem für künftige Anforderungen.
- Sie stellen zu viele Input- und Output-Operationen pro Sekunde (IOPS) in Ihrem Dateisystem bereit.
- Sie überwachen die Nutzung Ihrer Daten-Volumes nicht.

Vorteile der Nutzung dieser bewährten Methode: Die Minimierung der übermäßigen Bereitstellung für das Speichersystem reduziert ungenutzte Ressourcen und verbessert die Gesamteffizienz Ihres Workloads.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Erstellen Sie Block-Speicher und Dateisysteme mit Größenzuweisung, Durchsatz und Latenz, die den Anforderungen Ihres Workloads entsprechen. Verwenden Sie Elastizität und Automatisierung, um den Block-Speicher oder das Dateisystem zu erweitern, wenn das Datenvolumen zunimmt, ohne dass diese Speicherservices übermäßig bereitgestellt werden.

Implementierungsschritte

- Stellen Sie bei Speichersystemen mit einer festen Größe wie Amazon EBS sicher, dass Sie die Menge des verwendeten Speichers im Vergleich zur Gesamtspeichergröße überwachen und nach Möglichkeit die Speichergröße beim Erreichen eines Schwellenwerts automatisch erhöhen.
- Verwenden Sie elastische Volumes und verwaltete Blockdaten-Services, um automatisch zusätzlichen Speicher zuzuweisen, wenn die Menge der persistenten Daten wächst. Sie können beispielsweise Amazon EBS Elastic Volumes verwenden, um Volume-Größe, Volume-Typ oder die Leistung Ihrer Amazon EBS-Volumes zu modifizieren.
- Wählen Sie die korrekte Speicherklasse sowie den korrekten Leistungs- und Durchsatz-Modus für Ihr Dateisystem für Ihre geschäftlichen Anforderungen und überschreiten Sie diese nicht.
  - Amazon EFS Leistung
  - Amazon EBS-Volume-Leistung auf Linux-Instances
- Legen Sie Zielstufen für die Nutzung Ihrer Daten-Volumes fest und passen Sie die Größe von Volumes an, die außerhalb der erwarteten Bereiche liegen.
- Passen Sie die Größe schreibgeschützter Volumes an die Datenmenge an.
- Migrieren Sie Daten zu Objektspeichern, um zu vermeiden, dass die überschüssige Kapazität aus Volumes mit fester Größe im Blockspeicher bereitgestellt wird.

- Überprüfen Sie elastische Volumes und Dateisysteme, beenden Sie nicht genutzte und verkleinern Sie zu große Volumes, um sie an den aktuellen Datenumfang anzupassen.

Ressourcen

Zugehörige Dokumente:

- Erweitern des Dateisystems nach der Größenänderung eines EBS-Volumes
- Ändern eines Volume mithilfe von Amazon EBS Elastic Volumes
- Amazon FSx-Documentation
- Was ist Amazon Elastic File System?

Zugehörige Videos:

- Weiterführende Informationen zu Amazon EBS Elastic Volumes
- Amazon EBS und Snapshot-Optimierungsstrategien für bessere Leistung und Kosteneinsparungen
- Amazon EFS mithilfe bewährter Methoden für Kosten und Leistung optimieren

SUS04-BP05 Entfernen nicht benötigter oder redundanter Daten

Entfernen Sie nicht benötigte oder redundante Daten, um die zum Speichern Ihrer Datensätze benötigten Speicherressourcen zu minimieren.

Typische Anti-Muster:

- Sie duplizieren Daten, die leicht abgerufen oder erneut erstellt werden können.
- Sie sichern alle Daten, ohne ihre Kritikalität zu berücksichtigen.
- Sie löschen Daten nur unregelmäßig, nur bei bestimmten Ereignissen oder gar nicht.
- Sie speichern Daten redundant, unabhängig von der Stabilität des Speicherservices.
- Sie aktivieren die Amazon S3-Versionsverwaltung, ohne dass dies geschäftlich gerechtfertigt ist.

Vorteile der Einführung dieser bewährten Methode: Durch das Entfernen nicht benötigter Daten werden die für Ihren Workload benötigte Speichergröße und die Umweltbelastungen durch den Workload reduziert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Speichern Sie keine Daten, die Sie nicht benötigen. Automatisieren Sie das Löschen von nicht benötigten Daten. Verwenden Sie Technologien, die Daten auf Datei- und Blockebene deduplizieren. Nutzen Sie native Servicefunktionen für Replikation und Redundanz.

Implementierungsschritte

- Bewerten Sie, ob Sie das Speichern von Daten vermeiden können, indem Sie vorhandene, öffentlich verfügbare Datensätze in AWS Data Exchange und offene Daten in AWS verwenden.
- Verwenden Sie Mechanismen, die Daten auf Block- und Objektebene deduplizieren können. Hier finden Sie einige Beispiele zum Deduplizieren von Daten in AWS:

Storage service	Deduplication mechanism
Amazon S3	Verwenden Sie AWS Lake Formation FindMatches und das neue FindMatches ML Transform, um übereinstimmende Einträge in einem Datensatz zu finden (auch solche ohne ID).
Amazon FSx	Verwenden Sie die Datendeduplizierung in Amazon FSx für Windows.
Amazon Elastic Block Store-Snapshots	Bei Snapshots handelt es sich um inkrement elle Sicherungen. Das bedeutet, dass nur die Blöcke auf dem Gerät gespeichert werden, die sich seit dem letzten Snapshot geändert haben.

- Analysieren Sie den Datenzugriff, um nicht benötigte Daten zu identifizieren. Automatisieren Sie Lebenszyklusrichtlinien. Nutzen Sie zum Löschen native Servicefunktionen wie Amazon DynamoDB Time To Live Amazon S3-Lebenszyklen oder die Amazon CloudWatch-Protokollaufbewahrung.
- Verwenden Sie Virtualisierungsfunktionen in AWS, um Daten an der Quelle beizubehalten und eine Duplikation zu vermeiden.
  - Cloudnative Datenvirtualisierung in AWS
  - Optimierung von Datenmustern mithilfe von Amazon Redshift Data Sharing

- Verwenden Sie Backup-Technologien, mit denen inkrementelle Sicherungen möglich sind.
- Nutzen Sie zum Erfüllen der Stabilitätsziele die Stabilität von Amazon S3 und Replikation von Amazon EBS anstelle von selbst verwalteten Technologien wie redundanten Arrays unabhängiger Datenträger (Redundant Array Of Independent Disks, RAID).
- Zentralisieren Sie Protokoll- und Nachverfolgungsdaten, deduplizieren Sie identische Protokolleinträge und richten Sie Mechanismen für die Anpassung der Ausführlichkeit ein, wenn notwendig.
- Füllen Sie Zwischenspeicher nur vorab aus, wenn dies begründet werden kann.
- Richten Sie Überwachung und Automatisierung für den Cache ein, um seine Größe entsprechend anzupassen.
- Entfernen Sie veraltete Bereitstellungen und Komponenten aus Objektspeichern und Edge-Zwischenspeichern, wenn Sie neue Versionen Ihres Workloads veröffentlichen.

Ressourcen

Zugehörige Dokumente:

- Change log data retention in CloudWatch Logs (Ändern der Protokolldatenaufbewahrung in CloudWatch Logs)
- Data deduplication on Amazon FSx for Windows File Server (Datendeduplizierung in Amazon FSx für Windows File Server)
- Features of Amazon FSx for ONTAP including data deduplication (Funktionen von Amazon FSx for ONTAP einschließlich Datendeduplizierung)
- Invalidating Files on Amazon CloudFront (Invalidieren von Dateien auf Amazon CloudFront)
- Using AWS Backup to back up and restore Amazon EFS file systems (Verwenden von AWS Backup, um Amazon EFS-Dateisysteme zu sichern und wiederherzustellen)
- Was ist Amazon CloudWatch Logs?
- Arbeiten mit Backups in Amazon RDS
- Integrieren und Deduplizieren von Datensätzen mit AWS Lake Formation

Zugehörige Videos:

- Anwendungsfälle für den Amazon Redshift-Datenaustausch

Zugehörige Beispiele:

- Wie analysiere ich meine Amazon S3-Serverzugriffsprotokolle mit Amazon Athena?

SUS04-BP06 Verwenden geteilter Dateisysteme oder Objektspeicher für den Zugriff auf allgemeine Daten

Verwenden Sie geteilte Dateisysteme oder Speicher, um Datenduplizierungen zu vermeiden und eine effizientere Infrastruktur für Ihren Workload zu ermöglichen.

Typische Anti-Muster:

- Sie stellen für jeden einzelnen Client Speicher bereit.
- Sie trennen Datenvolumina von inaktiven Clients nicht ab.
- Sie ermöglichen keinen Zugriff auf Speicher über Plattformen und Systeme hinweg.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung geteilter Dateisysteme oder Speicher ermöglicht die gemeinsame Nutzung von Daten für mehrere Verbraucher, ohne dass diese dazu kopiert werden müssen. Dies reduziert den Umfang der erforderlichen Speicherressourcen für den Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wenn Sie mehrere Benutzer oder Anwendungen haben, die auf dieselben Datensätze zugreifen müssen, ist die Verwendung geteilter Speichertechnologien wichtig für eine effiziente Infrastruktur für Ihren Workload. Solche Technologien bieten einen zentralen Speicherort für die Speicherung und Verwaltung von Datensätzen und zur Vermeidung von Datenduplizierungen. Dazu wird die Konsistenz der Daten über verschiedene Systeme hinweg durchgesetzt. Hinzu kommt, dass geteilte Speicher die effizientere Nutzung der Rechenleistung ermöglichen, da mehr Computing-Ressourcen gleichzeitig auf Daten zugreifen und diese verarbeiten können.

Rufen Sie Daten von diesen geteilten Speicherservices nur bei Bedarf ab und trennen Sie nicht genutzte Volumes, um Ressourcen freizugeben.

Implementierungsschritte

- Migrieren Sie Daten in einen geteilten Speicher, wenn die Daten mehrfach genutzt werden. Hier sind einige Beispiele für geteilte Speichertechnologien auf AWS:

Storage option	When to use
Amazon EBS Multi-Attach	Amazon EBS Multi-Attach ermöglicht die Anfügung eines einzelnen bereitgestellten IOPS SSD (io1 oder io2)-Volumes an mehrere Instances in derselben Availability Zone.
Amazon EFS	Vgl. Auswahl von Amazon EFS.
Amazon FSx	Vgl. Auswahl eines Amazon FSx-Datei systems.
Amazon S3	Anwendungen, die keine Dateisystemstruktu r benötigen und zur Arbeit mit Objektspe ichern gedacht sind, können Amazon S3 als massive, skalierbare, dauerhafte und kostengünstige Speicherlösung nutzen.

- Kopieren Sie Daten bzw. rufen Sie sie nur dann von geteilten Dateisystemen ab, wenn Sie sie benötigen. Sie können beispielsweise ein Amazon FSx for Lustre-Dateisystem mit Unterstützung durch Amazon S3 erstellen und nur die Teilmenge der Daten laden, die für die Verarbeitung von Aufgaben zu Amazon FSx benötigt werden.
- Löschen Sie Daten entsprechend Ihren Nutzungsmustern, wie in SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien erläutert.
- Trennen Sie Volumes von Clients, die sie nicht aktiv verwenden.

Ressourcen

Zugehörige Dokumente:

- Verknüpfung Ihres Dateisystems mit einem Amazon S3-Bucket
- Amazon EFS für AWS Lambda in Ihren Serverless-Anwendungen verwenden
- Amazon EFS Intelligent-Tiering optimiert die Kosten für Workloads mit wechselnden Zugriffsmustern
- Verwendung von Amazon FSx mit Ihrem On-Premises-Daten-Repository

Zugehörige Videos:

- Optimierung der Speicherkosten mit Amazon EFS
- AWS re:Invent 2023 – Neuerungen bei AWS-Dateispeicher
- AWS re:Invent 2023 – Dateispeicher für Entwickler und Datenwissenschaftler auf Amazon Elastic File System

SUS04-BP07 Minimieren von Datenübertragungen zwischen Netzwerken

Verwenden Sie gemeinsam genutzte Dateisysteme oder Objektspeicher zum Zugriff auf häufig genutzte Daten und minimieren Sie die zur Unterstützung von Datenverschiebungen für Ihren Workload benötigten Netzwerkressourcen.

Typische Anti-Muster:

- Sie speichern alle Daten im selben AWS-Region, unabhängig davon, wo sich deren Benutzer befinden.
- Sie optimieren Datenumfang und -format nicht vor der Verschiebung über das Netzwerk.

Vorteile der Nutzung dieser bewährten Methode: Die Optimierung der Datenverschiebung über das Netzwerk reduziert den Umfang der für den Workload benötigten Netzwerkressourcen und verringert die Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Das Verschieben von Daten in der gesamten Organisation erfordert Computing-, Netzwerk- und Speicherressourcen. Verwenden Sie Techniken zur Minimierung von Datenverschiebungen und verbessern Sie die Gesamteffizienz Ihres Workloads.

Implementierungsschritte

- Berücksichtigen Sie die Nähe zu den Daten oder Benutzer für die Entscheidung bei der Auswahl einer Region für Ihren Workload.
- Partitionieren Sie regional genutzte Services so, dass regionsspezifische Daten in der Region gespeichert werden, in der sie genutzt werden.
- Verwenden Sie effiziente Dateiformate (wie etwa Parquet oder ORC) und komprimieren Sie die Daten, bevor Sie sie über das Netzwerk verschieben.

- Verschieben Sie keine nicht genutzten Daten. Einige Beispiele, die Ihnen helfen können, das Verschieben ungenutzter Daten zu vermeiden:
  - Beschränken Sie API-Antworten nur auf relevante Daten.
  - Aggregieren Sie Daten, wenn keine detaillierten Informationen auf Datensatzebene benötigt werden.
  - Siehe Well-Architected Lab – Optimierung von Datenmustern mit Amazon Redshift Data Sharing.
  - Erwägen Sie die Kontoübergreifende Datenfreigabe in AWS Lake Formation.
- Nutzen Sie Services, die Ihnen dabei helfen können, Code näher an den Benutzern Ihres Workloads auszuführen:

Service	When to use
Lambda@Edge	Verwenden Sie dies für rechenintensive Anwendungen, die ausgeführt werden, wenn sich Objekte nicht im Zwischenspeicher befinden.
CloudFront-Funktionen	Verwenden Sie diese für einfache Anwendung sfälle wie HTTP(s)-Anfragen oder Antwortma nipulationen, die von kurzlebigen Funktionen initiiert werden können.
AWS IoT Greengrass	Führen Sie lokale Rechenoperationen, Messaging sowie die Datenzwischenspeic herung für verbundene Geräte aus.

Ressourcen

Zugehörige Dokumente:

- Optimieren Ihrer AWS-Infrastruktur für Nachhaltigkeit, Teil III: Netzwerke
- Globale AWS-Infrastruktur
- Hauptfunktionen von Amazon CloudFront, einschließlich des globalen Edge-Netzwerks von CloudFront)
- Komprimieren von HTTP-Anforderungen in Amazon OpenSearch Service

- Zwischenkomprimierung der Daten mit Amazon EMR
- Laden komprimierter Datendateien aus Amazon S3 in Amazon Redshift
- Bereitstellen von komprimierten Dateien mit Amazon CloudFront

Zugehörige Videos:

- Entmystifizierung der Datenübertragung in AWS

Zugehörige Beispiele:

- Nachhaltige Architektur – Minimierung des Datenverkehrs zwischen Netzwerken

SUS04-BP08 Sichern von Daten nur in dem Fall, wenn ihre erneute Erstellung schwierig ist

Vermeiden Sie das Sichern von Daten ohne geschäftlichen Wert, um die Anforderungen an Speicherressourcen für Ihren Workload zu minimieren.

Typische Anti-Muster:

- Sie haben keine Sicherungsstrategie für Ihre Daten.
- Sie sichern Daten, die problemlos erneut erstellt werden können.

Vorteile der Nutzung dieser bewährten Methode: Das Vermeiden der Sicherung nichtkritischer Daten reduziert den Umfang der benötigten Speicherressourcen für den Workload und verringert die Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Vermeidung der Sicherung nicht benötigter Daten kann Kosten senken und die von dem Workload verwendeten Speicherressourcen verringern. Sichern Sie nur Daten, die einen geschäftlichen Wert haben oder zur Erfüllung von Compliance-Anforderungen benötigt werden. Prüfen Sie Backup-Richtlinien und vermeiden Sie einen flüchtigen Speicher, der in einem Wiederherstellungsszenario keinen Wert bietet.

Implementierungsschritte

- Implementieren Sie eine Richtlinie für die Klassifizierung von Daten wie in SUS04-BP01 Implementieren einer Richtlinie für die Klassifizierung von Daten erläutert.
- Nutzen Sie die Wichtigkeit Ihrer Datenklassifizierung und entwerfen Sie eine Sicherungsstrategie auf der Grundlage Ihrer Recovery Time Objective (RTO) und Ihrer Recovery Point Objective (RPO). Vermeiden Sie die Sicherung nichtkritischer Daten.
  - Schließen Sie Daten aus, die problemlos erneut erstellt werden können.
  - Schließen Sie flüchtige Daten von Backups aus.
  - Schließen Sie lokale Kopien von Daten aus, es sei denn, die für die Wiederherstellung dieser Daten von einem gemeinsamen Standort benötigte Zeit überschreitet Ihre Service Level Agreements (SLAs).
- Verwenden Sie eine automatisierte Lösung oder einen verwalteten Service zur Sicherung geschäftskritischer Daten.
  - AWS Backup ist ein vollständig verwalteter Service, der die Zentralisierung und Automatisierung des Schutzes von Daten für AWS-Services in der Cloud und On-Premises vereinfacht. Praktische Anleitungen zur Erstellung automatisierter Sicherungen mit AWS Backup finden Sie unter Well-Architected Labs – Testen der Sicherung und Wiederherstellung Ihrer Daten.
  - Automatisieren Sie Sicherungen und optimieren Sie die Sicherungskosten für Amazon EFS mit AWS Backup.

Ressourcen

Zugehörige bewährte Methoden:

- REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen
- REL09-BP03 Automatische Daten-Backups
- REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen

Zugehörige Dokumente:

- Verwenden von AWS Backup, um Amazon EFS-Dateisysteme zu sichern und wiederherzustellen
- Amazon EBS-Snapshots
- Arbeiten mit Backups in Amazon Relational Database Service
- APN-Partner: Partner, die Sie bei der Sicherung unterstützen können

- AWS Marketplace: Für die Sicherung geeignete Produkte
- Sichern von Amazon EFS
- Sichern von Amazon FSx für Windows File Server
- Backup und Wiederherstellung für Amazon ElastiCache (Redis OSS)

Zugehörige Videos:

- AWS re:Invent 2023 – Backup- und Notfallwiederherstellungsstrategien für höhere Ausfallsicherheit
- AWS re:Invent 2023 – Neuerungen bei AWS Backup
- AWSre:Invent 2021 – Backup, Notfallwiederherstellung und Ransomware-Schutz mit AWS

Zugehörige Beispiele:

- Well-Architected Lab – Backup-Daten

Hardware und Services

Frage

- SUS 5 Wie wählen und nutzen Sie Cloud-Hardware und -Services in Ihrer Architektur so, dass Ihre Nachhaltigkeitsziele unterstützt werden?

SUS 5 Wie wählen und nutzen Sie Cloud-Hardware und -Services in Ihrer Architektur so, dass Ihre Nachhaltigkeitsziele unterstützt werden?

Suchen Sie nach Möglichkeiten, die Auswirkungen auf die Nachhaltigkeit Ihrer Workloads durch Änderungen der Methoden für die Hardwareverwaltung zu reduzieren. Minimieren Sie den Umfang der für die Bereitstellung erforderlichen Hardware und wählen Sie die jeweils effizienteste Hardware und den effizientesten Service für den jeweiligen Workload aus.

Bewährte Methoden

- SUS05-BP01 Verwenden der geringstmöglichen Menge an Hardware zur Erfüllung Ihrer Anforderungen
- SUS05-BP02 Verwenden von Instance-Typen mit den geringsten Auswirkungen
- SUS05-BP03 Verwenden verwalteter Services

- SUS05-BP04 Optimieren der Nutzung von hardwarebasierten Computing-Beschleunigern

SUS05-BP01 Verwenden der geringstmöglichen Menge an Hardware zur Erfüllung Ihrer Anforderungen

Verwenden Sie die geringstmögliche Menge an Hardware für Ihr Workload, um Ihre geschäftlichen Anforderungen in effizienter Weise zu erfüllen.

Typische Anti-Muster:

- Sie überwachen die Ressourcenauslastung nicht.
- Sie haben Ressourcen mit geringer Auslastung in Ihrer Architektur.
- Sie prüfen die Nutzung statischer Hardware nicht, um festzustellen, ob sie neu dimensioniert werden muss.
- Sie formulieren keine Ziele für die Hardwarenutzung in Ihrer Computing-Infrastruktur auf der Grundlage geschäftlicher KPIs.

Vorteile der Nutzung dieser bewährten Methode: Die korrekte Dimensionierung Ihrer Cloud-Ressourcen hilft dabei, die Umweltauswirkungen von Workloads zu reduzieren, Geld zu sparen und Leistungsbenchmarks einzuhalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wählen Sie die optimale Anzahl von Hardwaregeräten für Ihren Workload aus, um die allgemeine Effizienz zu verbessern. AWS Cloud bietet die Flexibilität, Ressourcen dynamisch durch verschiedene Mechanismen wie etwa AWS Auto Scaling zu erweitern oder zu reduzieren, um einem veränderten Bedarf gerecht zu werden. Dazu kommen APIs und SDKs, mit denen Ressourcen mit minimalem Aufwand angepasst werden können. Verwenden Sie diese Möglichkeiten für häufige Änderungen an Ihren Workload-Implementierungen. Verwenden Sie dazu Dimensionierungsanleitungen von AWS-Tools für den effizienten Betrieb Ihrer Cloud-Ressourcen und die Erfüllung Ihrer geschäftlichen Anforderungen.

Implementierungsschritte

- Auswahl des Instance-Typs: Wählen Sie den Instance-Typ aus, der Ihren Anforderungen am besten entspricht. Weitere Informationen zur Auswahl von Amazon Elastic Compute Cloud-

Instances und zur Verwendung von Mechanismen wie der attributbasierten Auswahl des Instance-Typs finden Sie im Folgenden:

- Wie wähle ich einen geeigneten Amazon EC2-Instance-Typ für meinen Workload aus?
- Attributbasierte Auswahl des Instance-Typs für die Amazon EC2-Fleet.
- Erstellen einer Auto Scaling-Gruppe unter Verwendung einer attributbasierten Auswahl des Instance-Typs

- Skalierung: Skalieren Sie variable Workloads in kleinen Schritten.
- Verwendung mehrerer Computing-Einkaufsoptionen: Kombinieren Sie Instance-Flexibilität, Skalierbarkeit und Kosteneinsparungen mit mehreren Computing-Einkaufsoptionen.
  - Amazon EC2 On-Demand-Instances eignen sich am besten für neue, zustandsbehaftete Workloads mit Spitzen, die hinsichtlich Instance-Typ, Standort oder Zeit nicht flexibel sein können.
  - Amazon EC2 Spot Instances eignen sich hervorragend zur Ergänzung der anderen Optionen für Anwendungen, die fehlertolerant und flexibel sind.
  - Nutzen Sie Compute Savings Plans für stabile Workloads, die Flexibilität ermöglichen, wenn sich Ihre Anforderungen (wie AZ, Region, Instance-Familien oder Instance-Typen) ändern.
- Nutzung der Vielfalt von Instances und Availability Zones: Maximieren Sie die Anwendungsverfügbarkeit und nutzen Sie überschüssige Kapazitäten, indem Sie Ihre Instances und Availability Zones diversifizieren.
- Korrekte Dimensionierung von Instances: Verwenden Sie die Empfehlungen zur Dimensionierung in AWS-Tools, um Anpassungen an Ihrem Workload vorzunehmen. Weitere Informationen finden Sie unter Kostenoptimierung mit Empfehlungen zur richtigen Dimensionierung und Richtige Dimensionierung: Bereitstellen von Instances entsprechend den Workloads.
  - Verwenden Sie die Empfehlungen zur Dimensionierung in AWS Cost Explorer oder AWS Compute Optimizer zur Identifizierung von Dimensionierungsmöglichkeiten.
- Verhandlung von Service Level Agreements (SLAs): Verhandeln Sie SLAs, die eine vorübergehende Reduzierung der Kapazität ermöglichen, während die Automatisierung Ersatzressourcen bereitstellt.

Ressourcen

Zugehörige Dokumente:

- Optimieren Ihrer AWS-Infrastruktur für Nachhaltigkeit, Teil I: Datenverarbeitung

- Attributbasierte Auswahl des Instance-Typs für Auto Scaling und die Amazon EC2 Fleet
- AWS Compute Optimizer-Dokumentation
- Ausführen von Lambda: Leistungsoptimierung
- Dokumentation zu Auto Scaling

Zugehörige Videos:

- AWS re:Invent 2023 – Neuerungen bei Amazon EC2
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon Elastic Compute Cloud-Strategien zur Kostenoptimierung
- AWS re:Invent 2022 – Optimierung von Amazon Elastic Kubernetes Service zur Leistungs- und Kostenoptimierung in AWS
- AWS re:Invent 2023 – Nachhaltiges Computing: Reduzierung von Kosten und CO2-Emissionen mit AWS

SUS05-BP02 Verwenden von Instance-Typen mit den geringsten Auswirkungen

Überwachen und nutzen Sie kontinuierlich neue Instance-Typen, um Verbesserungen bei der Energieeffizienz zu nutzen.

Typische Anti-Muster:

- Sie verwenden lediglich eine Familie von Instances.
- Sie verwenden nur x86-Instances.
- Sie geben einen Instance-Typ in Ihrer Amazon EC2 Auto Scaling-Konfiguration an.
- Sie verwenden AWS-Instances in einer Weise, für die sie nicht gedacht sind (beispielsweise Computing-optimierte Instances für speicherintensive Workloads).
- Sie evaluieren nicht regelmäßig neue Instance-Typen.
- Sie prüfen nicht die Empfehlungen von AWS-Dimensionierungstools wie etwa AWS Compute Optimizer.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung energieeffizienter und korrekt dimensionierter Instances können Sie die Umweltauswirkungen und die Kosten Ihrer Workloads deutlich reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Die Verwendung effizienter Instances für Cloud-Workloads ist von entscheidender Bedeutung für eine geringere Ressourcennutzung und die Kosteneffizienz. Überwachen Sie kontinuierlich die Einführung neuer Instance-Typen und nutzen Sie Verbesserungen bei der Energieeffizienz, einschließlich Instance-Typen, die zur Unterstützung spezifischer Workloads bestimmt sind, wie z. B. Machine-Learning-Trainings und -Inferenzen und Videotranskodierung.

Implementierungsschritte

- Kennenlernen der Instance-Typen: Finden Sie Instance-Typen, mit denen Sie die Umweltbelastung Ihrer Workloads verringern können.
  - Abonnieren Sie Neuerungen bei AWS, um sich über die aktuellen AWS-Technologien und -Instances auf dem Laufenden zu halten.
  - Informieren Sie sich über die verschiedenen AWS-Instance-Typen.
  - Informieren Sie sich über AWS-Graviton-basierte Instances, die die beste Leistung pro Watt in Amazon EC2 bieten. Sehen Sie sich re:Invent 2020 – Vertiefung in vom AWS-Graviton2-Prozessor unterstützte Amazon EC2-Instances und Vertiefung in AWS-Graviton3 und Amazon EC2-C7g-Instances an.
- Verwendung von Instance-Typen mit den geringsten Auswirkungen: Planen Sie Ihren Workload und stellen Sie ihn auf Instance-Typen mit den geringsten Auswirkungen um.
  - Definieren Sie einen Prozess zur Evaluierung neuer Features oder Instances für Ihre Workloads. Nutzen Sie die Agilität in der Cloud, um schnell zu testen, wie neue Instance-Typen die ökologische Nachhaltigkeit Ihrer Workloads verbessern können. Nutzen Sie Proxy-Metriken, um zu messen, wie viele Ressourcen Sie für eine Arbeitseinheit benötigen.
  - Modifizieren Sie Ihren Workload nach Möglichkeit so, dass er mit unterschiedlichen Zahlen von vCPUs und Arbeitsspeichergrößen kompatibel ist, um die größtmögliche Auswahl an Instance-Typen zu erhalten.
  - Erwägen Sie die Übertragung Ihres Workloads zu auf Graviton basierenden Instances, um die Leistungseffizienz Ihres Workloads zu verbessern. Weitere Informationen zum Verschieben von Workloads zu AWS Graviton finden Sie unter AWS Graviton Schnellstart und Überlegungen bei der Übertragung von Workloads zu auf AWS Graviton basierenden Amazon Elastic Compute Cloud-Instances.
  - Erwägen Sie die Auswahl der AWS Graviton-Option, wenn Sie verwaltete AWS-Services verwenden.

  - Migrieren Sie Ihren Workload zu Regionen mit Instances, die die geringsten nachhaltigkeitsbezogenen Auswirkungen bieten und dennoch Ihre geschäftlichen Anforderungen erfüllen.
  - Nutzen Sie für Machine Learning-Workloads spezielle Hardware, die auf Ihren Workload abgestimmt ist, z. B. AWS Trainium, AWS Inferentia oder Amazon EC2 DL1. AWS Inferentia-Instances wie Inf2-Instances bieten eine um bis zu 50 % bessere Leistung pro Watt als vergleichbare Amazon EC2-Instances.
  - Verwenden Sie Amazon SageMaker AI Inference Recommender für die korrekte Dimensionierung des ML-Inferenz-Endpunkts.
  - Verwenden Sie für Workloads, bei denen es gelegentlich zu zusätzlichen Kapazitätsanforderungen kommt, Instances mit Spitzenlastleistung.
  - Verwenden Sie für zustandslose und fehlertolerante Workloads Amazon EC2 Spot Instances, um die allgemeine Auslastung der Cloud zu verbessern und die Nachhaltigkeitsauswirkungen ungenutzter Ressourcen zu reduzieren.
- Betrieb und Optimierung: Betreiben und optimieren Sie Ihre Workload-Instance.
  - Prüfen Sie für kurzzeitige Workloads die Instance-Amazon CloudWatch-Metriken wie `CPUUtilization`, um festzustellen, ob die Instance gar nicht oder zu wenig genutzt wird.
  - Prüfen Sie für stabile Workloads in regelmäßigen Intervallen AWS-Dimensionierungstools wie etwa AWS Compute Optimizer, um Möglichkeiten zur Optimierung und zur korrekten Dimensionierung der Instances zu erkennen.
    - Well-Architected Lab – Empfehlungen zur Dimensionierung
    - Well-Architected Lab – Dimensionierung mit Compute Optimizer
    - Well-Architected Lab – Optimieren von Hardwaremustern und Überwachen von KPIs zur Nachhaltigkeit

Ressourcen

Zugehörige Dokumente:

- Optimieren Ihrer AWS-Infrastruktur für Nachhaltigkeit, Teil I: Datenverarbeitung
- AWS Graviton
- Amazon EC2 DL1
- Amazon EC2-Flotten zur Kapazitätsreservierung
- Amazon EC2-Spot-Flotte

- Funktionen: Lambda-Funktionskonfiguration
- Attributbasierte Auswahl des Instance-Typs für die Amazon EC2-Flotte
- Entwicklung nachhaltiger, effizienter und kostenoptimierter Anwendungen auf AWS
- So können Kunden mit dem Contino Sustainability Dashboard ihren CO2-Fußabdruck optimieren

Zugehörige Videos:

- AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads
- AWS re:Invent 2023 – Neue generative KI-Funktionen von Amazon Elastic Compute Cloud in AWS Management Console
- AWS re:Invent 2023 – Neuerungen bei Amazon Elastic Compute Cloud
- AWS re:Invent 2023 – Intelligentes Sparen: Amazon Elastic Compute Cloud-Strategien zur Kostenoptimierung
- AWS re:Invent 2021 – Vertiefung in AWS-Graviton3- und Amazon EC2-C7g-Instances
- AWS re:Invent 2022 – Entwickeln einer kosten-, energie- und ressourceneffizienten Computing-Umgebung

Zugehörige Beispiele:

- Lösung: Anleitung zur Optimierung von Deep-Learning-Workloads für mehr Nachhaltigkeit in AWS
- Migration von Amazon Relational Database Service-Datenbanken zu Graviton

SUS05-BP03 Verwenden verwalteter Services

Verwenden Sie verwaltete Services für effizientere Betriebsabläufe in der Cloud.

Typische Anti-Muster:

- Sie verwenden Amazon EC2-Instances mit geringer Ausnutzung für die Ausführung Ihrer Anwendungen.
- Ihr internes Team verwaltet nur den Workload, ohne Zeit zu haben, sich auf Innovation oder Vereinfachungen zu konzentrieren.
- Sie nutzen und verwalten Technologien für Aufgaben, die effizienter auf verwalteten Services ausgeführt werden können.

Vorteile der Nutzung dieser bewährten Methode:

- Durch die Verwendung verwalteter Services geht die Verantwortung auf AWS über, mit Erkenntnissen zu Millionen von Kunden, was Innovationen und neue Effizienzen ermöglicht.
- Ein verwalteter Service verteilt die Umweltauswirkungen des Services durch Multi-Tenet-Steuerebenen auf mehrere Benutzer.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Verwaltete Services übertragen die Verantwortung für die Wahrung einer hohen durchschnittlichen Nutzung und die Optimierung der Nachhaltigkeit der bereitgestellten Hardware auf AWS. Verwaltete Services eliminieren dazu den betrieblichen und administrativen Aufwand für die Wartung eines Service, so Ihr Team mehr Zeit hat und sich auf Innovationen konzentrieren kann.

Prüfen Sie Ihren Workload, um die Komponenten zu identifizieren, die von verwalteten AWS-Services ersetzt werden können. Beispielsweise bieten Amazon RDS, Amazon Redshift und Amazon ElastiCache einen verwalteten Datenbankservice. Amazon Athena, Amazon EMR und Amazon OpenSearch Service bieten einen verwalteten Analytics-Service.

Implementierungsschritte

1. Inventarisieren Ihres Workload: Inventarisieren Sie Ihren Workload für Services und Komponenten.
2. Identifizieren von Kandidaten: Prüfen und identifizieren Sie Komponenten, die durch verwaltete Services ersetzt werden können. Hier finden Sie einige Beispiele für Situationen, in denen Sie einen verwalteten Service in Erwägung ziehen sollten:

Task	What to use on AWS
Hosten einer Datenbank	Verwenden Sie verwaltete Amazon Relational Database Service (Amazon RDS)-Instances, anstatt Ihre eigenen Amazon RDS-Instances auf Amazon Elastic Compute Cloud (Amazon EC2) zu verwalten.

Task	What to use on AWS
Hosten eines Container-Workloads	Verwenden Sie AWS Fargate, anstatt Ihre eigene Container-Infrastruktur zu implement ieren.
Hosten von Web-Apps	Verwenden Sie AWS Amplify Hosting als vollständig verwalteten CI/CD- und Hosting-S ervice für statische Websites und serverseitig gerenderte Web-Apps.

3. Erstellen eines Migrationsplans: Identifizieren Sie Abhängigkeiten und erstellen Sie einen Migrationsplan. Aktualisieren Sie Runbooks und Playbooks entsprechend.
   - Der AWS Application Discovery Service erfasst und präsentiert automatisch detaillierte Informationen zu Abhängigkeiten und zur Nutzung von Anwendungen, damit Sie bei der Planung Ihrer Migration fundierte Entscheidungen treffen können.
4. Tests: Testen Sie den Service vor der Migration zum verwalteten Service.
5. Ersetzen selbst gehosteter Services: Verwenden Sie Ihren Migrationsplan, um selbst gehostete Services durch verwaltete Services zu ersetzen.
6. Überwachen und anpassen: Überwachen Sie den Service nach der Migration kontinuierlich, um erforderliche Anpassungen vorzunehmen und den Service zu optimieren.

Ressourcen

Zugehörige Dokumente:

- AWS Cloud-Produkte
- AWS-Gesamtbetriebskostenrechner (Total Cost of Ownership, TCO)
- Amazon DocumentDB
- Amazon Elastic Kubernetes Service (EKS)
- Amazon Managed Streaming for Apache Kafka (Amazon MSK)

Zugehörige Videos:

- AWS re:Invent 2021 – Cloud-Betriebsabläufe in großem Umfang mit AWS Managed Services
- AWS re:Invent 2023 – Bewährte Methoden für den Betrieb in AWS

SUS05-BP04 Optimieren der Nutzung von hardwarebasierten Computing-Beschleunigern

Sie können die Nutzung von beschleunigten Computing-Instances optimieren, um die Anforderungen Ihres Workloads an die physische Infrastruktur zu reduzieren.

Typische Anti-Muster:

- Sie überwachen die GPU-Nutzung nicht.
- Sie verwenden eine allgemeine Instance für den Workload, während eine speziell angefertigte Instance eine höhere Leistung, geringere Kosten und eine bessere Leistung pro Watt bieten kann.
- Sie verwenden hardwarebasierte Computing-beschleuniger für Aufgaben, bei denen CPU-basierte Alternativen effizienter sind.

Vorteile der Einführung dieser bewährten Methode: Indem Sie die Nutzung von hardwarebasierten Accelerators optimieren, können Sie die Anforderungen Ihres Workloads an die physische Infrastruktur reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Wenn Sie eine hohe Verarbeitungsleistung benötigen, können Sie beschleunigte Computing-Instances verwenden. Diese bieten Zugriff auf hardwarebasierte Computing-Beschleuniger wie Grafikprozessoren (Graphics Processing Units, GPUs) und Field Programmable Gate Arrays (FPGAs). Diese Hardwarebeschleuniger führen bestimmte Funktionen wie die Grafikverarbeitung oder Datenmusterzuordnung effizienter aus als CPU-basierte Alternativen. Viele beschleunigte Workloads, wie Rendering, Transcodierung und Machine Learning, sind sehr variabel im Bezug auf die Ressourcennutzung. Betreiben Sie diese Hardware nur so lange wie nötig und nehmen Sie sie automatisch außer Betrieb, wenn sie nicht mehr benötigt wird, um den Ressourcenverbrauch zu minimieren.

Implementierungsschritte

- Ermitteln Sie, welche beschleunigten Computing-Instances für Ihre Anforderungen geeignet sind.
- Nutzen Sie für Machine Learning-Workloads spezielle Hardware, die auf Ihren Workload abgestimmt ist, z. B. AWS Trainium, AWS Inferentia oder Amazon EC2 DL1. AWS-Inferentia-Instances wie Inf2-Instances bieten eine um bis zu 50 % bessere Leistung pro Watt als vergleichbare Amazon EC2-Instances.

- Erfassen Sie Nutzungsmetriken für Ihre beschleunigten Computing-Instances. Sie können z. B. CloudWatch-Agents verwenden, um Metriken wie `utilization_gpu` und `utilization_memory` für Ihre GPUs zu erfassen. Dies wird im Artikel zum Erfassen von NVIDIA GPU-Metriken mit Amazon CloudWatch genauer beschrieben.
- Optimieren Sie Code, Netzwerkbetrieb und die Einstellungen von Hardwarebeschleunigern, um sicherzustellen, dass die zugrunde liegende Hardware optimal genutzt wird.
  - Optimieren der GPU-Einstellungen
  - GPU-Überwachung und -Optimierung
  - Optimieren von E/A für die GPU-Leistungsoptimierung von Deep Learning-Training in Amazon SageMaker AI
- Verwenden Sie die aktuellen leistungsstarken Bibliotheken und GPU-Treiber.
- Automatisieren Sie die Freigabe nicht genutzter GPU-Instances.

Ressourcen

Zugehörige Dokumente:

- Accelerated Computing
- Let's Architect! Architecting with custom chips and accelerators (Erstellen von Architekturen mit benutzerdefinierten Chips und Beschleunigern)
- How do I choose the appropriate Amazon EC2 instance type for my workload? (Wie wähle ich einen geeigneten EC2-Instance-Typ für meinen Workload aus?)
- Amazon EC2-VT1-Instances
- Auswählen des besten KI-Accelerators und der Modellkompilierung für Computer Vision Inference mit Amazon SageMaker AI

Zugehörige Videos:

- AWS re:Invent 2021 – Auswählen von Amazon EC2-GPU-Instances für Deep Learning
- AWS Online Tech Talks – Bereitstellung kostengünstiger Deep Learning Inference
- AWS re:Invent 2023 – Moderne KI mit AWS und NVIDIA
- AWS re:Invent 2022 – [NEUER LAUNCH!] Einführung von AWS-Inferentia2-basierten Amazon EC2-Inf2-Instances

- AWS re:Invent 2022 – Beschleunigung von Deep Learning und schnellere Innovationen mit AWS Trainium
- AWS re:Invent 2022 – Deep Learning in AWS mit NVIDIA: Vom Training bis zur Bereitstellung

Prozess und Kultur

Frage

- SUS 6 Wie unterstützen Ihre betrieblichen Prozesse Ihre Nachhaltigkeitsziele?

SUS 6 Wie unterstützen Ihre betrieblichen Prozesse Ihre Nachhaltigkeitsziele?

Reduzieren Sie nachhaltigkeitsbezogene Auswirkungen, indem Sie Ihre Entwicklungs-, Test- und Bereitstellungsmethoden ändern.

Bewährte Methoden

- SUS06-BP01 Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen
- SUS06-BP02 Konstantes Aktualisieren Ihres Workloads
- SUS06-BP03 Höhere Auslastung von Entwicklungsumgebungen
- SUS06-BP04 Verwenden verwalteter Gerätefarmen für Tests

SUS06-BP01 Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen

Nutzen Sie Methoden und Prozesse zur Validierung potenzieller Verbesserung, zur Minimierung von Testkosten und zur Bereitstellung kleinerer Verbesserungen.

Typische Anti-Muster:

- Die Prüfung Ihrer Anwendung auf Nachhaltigkeitsaspekte erfolgt nur einmal zu Beginn des Projekts.
- Ihr Workload stagniert, da der Freigabeprozess zu komplex ist, um kleinere Verbesserungen für die Ressourceneffizienz umzusetzen.
- Sie verfügen über keine Mechanismen zur Verbesserung Ihres Workloads unter Nachhaltigkeitsaspekten.

Vorteile der Nutzung dieser bewährten Methode: Durch die Einrichtung eines Prozesses für die Einführung und Nachverfolgung von Nachhaltigkeitsverbesserungen können Sie kontinuierlich neue Funktionen einführen, Probleme beseitigen und die Workload-Effizienz verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Testen und validieren Sie potenzielle Verbesserungen in Bezug auf die Nachhaltigkeit, bevor Sie sie in der Produktion bereitstellen. Berücksichtigen Sie die Testkosten bei der Berechnung des potenziellen zukünftigen Nutzens einer Verbesserung. Entwickeln Sie kostengünstige Testmethoden, um kleinere Verbesserungen einzuführen.

Implementierungsschritte

- Kenntnis und Kommunikation der Nachhaltigkeitsziele Ihrer Organisation: Machen Sie sich mit den Nachhaltigkeitszielen Ihrer Organisation vertraut, z. B. zur Reduzierung der CO2-Emissionen oder zum verantwortungsvollen Umgang mit Wasser. Übersetzen Sie diese Ziele in Nachhaltigkeitsanforderungen für Ihre Cloud-Workloads. Kommunizieren Sie diese Anforderungen an wichtige Stakeholder.
- Ergänzung des Backlogs mit Nachhaltigkeitsanforderungen: Fügen Sie Ihrem Entwicklungs-Backlog Anforderungen zur Verbesserung der Nachhaltigkeit hinzu.
- Iterieren und verbessern: Verwenden Sie einen iterativen Verbesserungsprozess, um diese Verbesserungen zu identifizieren, zu bewerten, zu priorisieren, zu testen und bereitzustellen.
- Tests unter Verwendung des Minimum Viable Product (MVP): Entwickeln und testen Sie potenzielle Verbesserungen unter Verwendung der Minimum-Viable-Komponenten, um die Kosten und die Umweltauswirkungen der Tests zu reduzieren.
- Prozessoptimierung: Verbessern und optimieren Sie kontinuierlich Ihre Entwicklungsprozesse. Sie können beispielsweise Ihren Softwarebereitstellungsprozess mit Pipelines für die Continuous Integration und Continuous Delivery (CI/CD) automatisieren, um potenzielle Verbesserungen zu testen und bereitzustellen und so den Aufwand zu reduzieren und Fehler durch manuelle Prozesse zu minimieren.
- Schulung und Sensibilisierung: Führen Sie Schulungsprogramme für Ihre Teammitglieder durch, um sie über Nachhaltigkeit und die Auswirkungen ihrer Aktivitäten auf die Nachhaltigkeitsziele Ihrer Organisation aufzuklären.
- Beurteilen und anpassen: Beurteilen Sie kontinuierlich die Auswirkungen von Verbesserungen und nehmen Sie bei Bedarf Anpassungen vor.

Ressourcen

Zugehörige Dokumente:

- AWS unterstützt Lösungen für die Nachhaltigkeit
- Skalierbare, agile Entwicklungspraktiken auf der Grundlage von AWS CodeCommit

Zugehörige Videos:

- AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft
- AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen
- AWS re:Invent 2022 – Nachhaltige Architektur und Reduzieren der AWS-CO2-Bilanz
- AWS re:Invent 2022 – Nachhaltigkeit in der globalen AWS-Infrastruktur
- AWS re:Invent 2023 – Neuerungen bei AWS Beobachtbarkeit und Betrieb

Zugehörige Beispiele:

- Well-Architected Lab – Umwandlung von Kosten- und Nutzenberichten in Effizienzberichte

SUS06-BP02 Konstantes Aktualisieren Ihres Workloads

Halten Sie Ihren Workload auf neustem Stand, um effiziente Funktionen zu übernehmen, Probleme zu beseitigen und die allgemeine Effizienz des Workloads zu wahren.

Typische Anti-Muster:

- Sie gehen davon aus, dass Ihre aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert wird.
- Sie haben keine Systeme oder regelmäßigen Besprechungen zur Prüfung, ob aktualisierte Software und Pakete mit Ihrem Workload kompatibel sind.

Vorteile der Einrichtung dieser bewährten Methode: Wenn Sie einen Prozess einrichten, um Ihren Workload auf neustem Stand zu halten, können Sie neue Funktionen und Kapazitäten nutzen, Probleme lösen und die Workload-Effizienz verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Aktuelle Betriebssysteme, Runtimes, Middleware, Bibliotheken und Anwendungen können die Workload-Effizienz verbessern und die Nutzung effizienterer Technologien unterstützen. Aktuelle Software kann darüber hinaus Funktionen für eine genauere Messung der Auswirkungen Ihres Workloads bereitstellen, da die Anbieter mit ihrer Software ebenfalls Nachhaltigkeitsziele erfüllen müssen. Sorgen Sie für Regelmäßigkeit bei der Aktualisierung Ihres Workloads mit den neuesten Funktionen und Versionen.

Implementierungsschritte

- Definieren eines Prozesses: Definieren Sie einen Prozess und einen Zeitplan, um neue Features oder Instances für Ihre Workloads zu evaluieren. Nutzen Sie die Agilität in der Cloud, um schnell zu testen, wie neue Features Ihre Workloads verbessern können:
  - Reduzierung von Auswirkungen auf die Nachhaltigkeit.
  - Erzielen von Leistungseffizienzen.
  - Beseitigen von Hindernissen für geplante Verbesserungen.
  - Verbesserung Ihrer Fähigkeit für die Messung von und den Umgang mit Nachhaltigkeitsauswirkungen.
- Inventarisierung: Inventarisieren Sie Ihre Workload-Software und -Architektur und identifizieren Sie Komponenten, die aktualisiert werden müssen.
  - Sie können AWS Systems Manager Inventory verwenden, um Betriebssystem (BS)-, Anwendungs- und Instance-Metadaten von Ihren Amazon EC2-Instances zu erfassen und so schnell zu verstehen, welche Instances die Software und die Konfigurationen ausführen, die Ihre Softwarerichtlinie erfordert, und welche Instances aktualisiert werden müssen.
- Kennenlernen des Aktualisierungsverfahrens: Erfahren Sie, wie die Komponenten Ihres Workloads aktualisiert werden.

Workload component	How to update
Machine Images	Verwenden Sie EC2 Image Builder zur Verwaltung von Updates für Amazon Machine Images (AMIs) für Linux- oder Windows Server-Images.
Container-Images	Verwenden Sie Amazon Elastic Container Registry (Amazon ECR) mit Ihrer vorhanden

Workload component	How to update
	en Pipeline zur Verwaltung Amazon Elastic Container Service (Amazon ECS) von Images.
AWS Lambda	AWS Lambda enthält Versionsmanagement-Features.

- Verwendung von Automatisierung: Verwenden Sie Automatisierung für den Aktualisierungsvorgang, um den Aufwand für die Bereitstellung neuer Features zu reduzieren und Fehler zu begrenzen, die durch manuelle Prozesse verursacht werden.
  - Sie können CI/CD verwenden, um AMIs, Container-Images und andere Artefakte im Zusammenhang mit Ihrer Cloud-Anwendung automatisch zu aktualisieren.
  - Sie können Tools wie den AWS Systems Manager Patch Manager verwenden, um den Systemaktualisierungsprozess zu automatisieren und die Aktivitäten mit AWS Systems ManagerMaintenance Windows zu planen.

Ressourcen

Zugehörige Dokumente:

- AWS Architecture Center
- Neuerungen bei AWS
- AWS-Entwickler-Tools

Zugehörige Videos:

- AWS re:Invent 2022 – Optimierung Ihrer AWS-Workloads mit Anleitungen für bewährte Methoden
- All Things Patch: AWS Systems Manager

Zugehörige Beispiele:

- Well-Architected Labs: Bestands- und Patch-Verwaltung
- Lab: AWS Systems Manager

SUS06-BP03 Höhere Auslastung von Entwicklungsumgebungen

Erhöhen Sie die Ausnutzung von Ressourcen zum Entwickeln, Testen und Erstellen Ihrer Workloads.

Typische Anti-Muster:

- Sie stellen Ihre Build-Umgebungen manuell bereit oder beenden sie in dieser Weise.
- Sie lassen Ihre Build-Umgebungen unabhängig von Test-, Build- oder Freigabeaktivitäten laufen (dazu gehört etwa der Betrieb einer Umgebung außerhalb der Arbeitszeit der Mitglieder Ihres Entwicklungsteams).
- Sie stellen übermäßig viele Ressourcen für Ihre Build-Umgebung bereit.

Vorteile der Nutzung dieser bewährten Methode: Durch die Steigerung der Ausnutzung von Build-Umgebungen können Sie die allgemeine Effizienz Ihres Cloud-Workloads verbessern, da die Ressourcen in effizienter Weise Entwicklungs-, Test- und Build-Aktivitäten zugewiesen werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Verwenden Sie Automatisierung und „Infrastructure as Code“, um Build-Umgebungen in Betrieb zu nehmen, wenn sie gebraucht werden, und sie andernfalls zu deaktivieren. Eine typische Vorgehensweise besteht in der Planung von Verfügbarkeitszeiten, die mit den Arbeitszeiten der Entwicklungsteams übereinstimmen. Ihre Testumgebungen sollten der Produktionskonfiguration sehr stark ähneln. Suchen Sie aber nach Möglichkeiten, Instance-Typen mit Burst-Kapazität, Amazon EC2-Spot-Instances, automatisch skalierenden Datenbankservices, Containern und Serverless-Technologien zu verwenden, um die Entwicklungs- und Testkapazität an der Nutzung auszurichten. Begrenzen Sie das Datenvolumen auf die Testanforderungen. Wenn Sie Produktionsdaten für einen Test verwenden, sollten Sie nach Möglichkeiten suchen, Daten aus der Produktion gemeinsam zu nutzen, anstatt Daten hin- und herzuschieben.

Implementierungsschritte

- Infrastructure as Code verwenden: Verwenden Sie Infrastructure as Code, um Ihre Entwicklungsumgebungen bereitzustellen.
- Automatisierung verwenden: Nutzen Sie Automatisierungen, um den Lebenszyklus Ihrer Entwicklungs- und Testumgebungen zu verwalten und die Effizienz Ihrer Entwicklungsressourcen zu maximieren.

- Nutzung maximieren: Verwenden Sie Strategien zur Maximierung der Nutzung von Entwicklungs- und Testumgebungen.
  - Verwenden Sie die geringstmögliche Zahl repräsentativer Umgebungen, um mögliche Verbesserungen zu entwickeln und zu testen.
  - Nutzen Sie nach Möglichkeit Serverless-Technologien.
  - Verwenden Sie On-Demand-Instances, um Entwicklergeräte zu ergänzen.
  - Verwenden Sie Instance-Typen mit Burst-Kapazität, Spot Instances und andere Technologien, um die Entwicklungskapazität an der Nutzung auszurichten.
  - Nutzen Sie native Cloud-Services für den sicheren Instance-Shell-Zugriff, statt Bastion-Host-Flotten bereitzustellen.
  - Skalieren Sie Ihre Build-Ressourcen automatisch je nach Build-Aktivität.

Ressourcen

Zugehörige Dokumente:

- AWS Systems Manager Session Manager
- Amazon EC2-Instances mit Spitzenlastleistung
- Was ist AWS CloudFormation?
- Was ist AWS CodeBuild?
- Instance Scheduler on AWS

Zugehörige Videos:

- AWS re:Invent 2023 – Kontinuierliche Integration und Bereitstellung für AWS

SUS06-BP04 Verwenden verwalteter Gerätefarmen für Tests

Verwenden Sie verwaltete Gerätefarmen zum effektiven Testen neuer Features auf einer repräsentativen Auswahl von Hardwaregeräten.

Typische Anti-Muster:

- Sie testen Ihre Anwendung manuell und stellen sie auf einzelnen physischen Geräten bereit.
- Sie verwenden keinen App-Testservice zum Testen und zum Interagieren mit Ihren Apps (beispielsweise Android, iOS und Web-Apps) auf realen physischen Geräten.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung verwalteter Gerätefarmen zum Testen cloud-fähiger Anwendungen bringt eine Reihe von Vorteilen mit sich:

- Dazu gehören effizientere Funktionen zum Testen von Anwendungen auf einer breiten Palette von Geräten.
- Sie machen hausinterne Infrastruktur zum Testen überflüssig.
- Sie bieten unterschiedliche Gerätetypen, darunter ältere und weniger verbreitete Hardware, was unnötige Geräte-Upgrades eliminiert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: niedrig

Implementierungsleitfaden

Die Verwendung verwalteter Gerätefarmen kann Ihnen dabei helfen, Ihre Testprozesse für neue Funktionen auf einer repräsentativen Auswahl von Hardwaregeräten zu optimieren. Verwaltete Gerätefarmen stellen verschiedene Gerätetypen bereit, unterstützen auch ältere und weniger verbreitete Hardware und vermeiden nachhaltigkeitsbezogene Auswirkungen auf Kunden durch unnötige Geräte-Upgrades.

Implementierungsschritte

- Testanforderungen definieren: Definieren Sie Ihre Testanforderungen und Ihren Testplan (z. B. Testtyp, Betriebssysteme und Testzeitplan).
  - Sie können Amazon CloudWatch RUM verwenden, um clientseitige Daten zu erfassen und zu analysieren und Ihren Testplan zu entwerfen.
- Verwaltete Gerätefarm auswählen: Wählen Sie eine verwaltete Gerätefarm, die Ihre Testanforderungen unterstützen kann. Sie können beispielsweise AWS-Gerätefarm verwenden, um die Auswirkungen Ihrer Änderungen auf eine repräsentative Auswahl von Hardwaregeräten zu testen und zu verstehen.
- Automatisierung verwenden: Verwenden Sie Automatisierung und kontinuierliche Integration/Bereitstellung (CI/CD) für die Planung und Durchführung Ihrer Tests.
  - Integration der AWS-Gerätefarm mit Ihrer CI/CD-Pipeline zur Durchführung Browser-übergreifender Selenium-Tests
  - Erstellen und Testen von iOS- und iPadOS-Apps mit AWS DevOps und mobilen Services
- Prüfen und Anpassen: Prüfen Sie kontinuierlich Ihre Testergebnisse und nehmen Sie die erforderlichen Verbesserungen vor.

Ressourcen

Zugehörige Dokumente:

- AWS Device Farm-Geräteliste
- Anzeige des CloudWatch RUM-Dashboards

Zugehörige Beispiele:

- AWS Device Farm Beispiel-App für Android
- AWS Device Farm Beispiel-App für iOS
- Appium-Web-Tests für AWS Device Farm

Zugehörige Videos:

- AWS re:Invent 2023 – Verbessern Sie die Qualität Ihrer Mobil- und Web-Apps mit der AWS-Gerätefarm
- AWS re:Invent 2021 – Optimierung von Anwendungen durch Endbenutzererkenntnisse mit Amazon CloudWatch RUM

Hinweise

Kunden sind eigenverantwortlich für die unabhängige Bewertung der Informationen in diesem Dokument zuständig. Dieses Dokument: (a) dient rein zu Informationszwecken, (b) spiegelt die aktuellen Produktangebote und Verfahren von AWS wider, die sich ohne vorherige Mitteilung ändern können, und (c) impliziert keinerlei Verpflichtungen oder Zusicherungen seitens AWS und dessen Tochtergesellschaften, Lieferanten oder Lizenzgebern. AWS-Produkte oder -Services werden im vorliegenden Zustand und ohne ausdrückliche oder stillschweigende Gewährleistungen, Zusicherungen oder Bedingungen bereitgestellt. Die Verantwortung und Haftung von AWS gegenüber seinen Kunden wird durch AWS-Vereinbarungen geregelt. Dieses Dokument ist weder ganz noch teilweise Teil der Vereinbarungen zwischen AWS und seinen Kunden und ändert diese Vereinbarungen auch nicht.

Copyright © 2021, Amazon Web Services, Inc. bzw. Tochtergesellschaften des Unternehmens.