AWS Well-Architected Framework

Säule „Betriebliche Exzellenz“

Säule „Betriebliche Exzellenz“: AWS Well-Architected Framework

Table of Contents

Säule „Operative Exzellenz“ – AWS-Well-Architected-Framework

Veröffentlichungsdatum: 3. Oktober 2023 (Dokumentversionen)

In diesem Dokument geht es speziell um die Säule der operativen Exzellenz des AWS-Well-Architected-Framework. Es enthält bewährte Methoden für die Konzeption, Übermittlung und Wartung von AWS-Workloads.

Einführung

Das AWS Well-Architected Framework unterstützt Sie dabei, die Vor- und Nachteile der Entscheidungen nachzuvollziehen, die Sie beim Aufbau von Workloads in AWS treffen. Das Framework hilft Ihnen, bewährte Betriebs- und Architekturmethoden für den Entwurf und Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Workloads in der Cloud zu ermitteln. Es bietet eine Möglichkeit, Ihre Betriebsabläufe und Architekturen konsistent auf die Einhaltung bewährter Methoden zu prüfen und Verbesserungspotenzial zu identifizieren. Wir sind der Meinung, dass eine gute auf den Betrieb ausgerichtete Workload-Architektur die Wahrscheinlichkeit für den geschäftlichen Erfolg deutlich erhöht.

Das Framework basiert auf den folgenden sechs Säulen:

- Operative Exzellenz
- Sicherheit
- Zuverlässigkeit
- Leistungseffizienz
- Kostenoptimierung
- Nachhaltigkeit

In diesem Dokument geht es speziell um die Säule der operativen Exzellenz und darum, wie Sie diese als Grundlage für architektonisch gute Lösungen anwenden. Operative Exzellenz ist eine Herausforderung in Umgebungen, in denen das operative Geschäft als eine isolierte und von den unterstützten Geschäftsbereichen und Entwicklungsteams getrennte Funktion wahrgenommen wird. Mithilfe der in diesem Dokument aufgeführten Methoden können Sie Architekturen aufbauen, die

Statuseinblicke geben, für einen effektiven und effizienten Betrieb ausgelegt sind, auf Ereignisse reagieren können und Ihre geschäftlichen Ziele unterstützen.

Dieses Dokument richtet sich an Nutzer in technologischen Rollen, z. B. CTOs (Chief Technology Officers), Architekten, Entwickler und Mitglieder von Operations-Teams. Sie erfahren darin mehr über die bewährten Methoden und Strategien von AWS für die Entwicklung sicherer Cloud-Architekturen für operative Exzellenz. Implementierungsdetails oder Architekturmodelle enthält dieses Dokument nicht. Allerdings beinhaltet es Verweise auf die entsprechenden Ressourcen, in denen Sie solche Informationen finden.

Operational Excellence

Bei Amazon definieren wir betriebliche Exzellenz als Verpflichtung, Software korrekt zu entwickeln und dabei einheitlich ein hervorragendes Kundenerlebnis zu bieten. Sie umfasst bewährte Methoden für die Organisation Ihres Teams, die Gestaltung Ihres Workloads, den Betrieb in großem Maßstab und die Weiterentwicklung im Laufe der Zeit. Betriebliche Exzellenz ermöglicht es Ihrem Team, mehr Zeit für die Entwicklung neuer Funktionen, von denen Kunden profitieren, aufzuwenden und weniger Zeit für Wartung und Fehlerbehebung. Für eine korrekte Entwicklung halten wir uns an bewährte Methoden, die zu gut funktionierenden Systemen, einer ausgewogenen Arbeitsbelastung für Sie und Ihr Team und vor allem zu einem hervorragenden Kundenerlebnis führen.

Ziel betrieblicher Exzellenz ist es, neue Funktionen und Fehlerkorrekturen schnell und zuverlässig in die Hände der Kunden zu geben. Unternehmen, die in betriebliche Exzellenz investieren, stellen ihre Kunden durchweg bei der Entwicklung neuer Funktionen, Vornahme von Änderungen und beim Umgang mit Ausfällen zufrieden. Auf dem Weg dorthin unterstützt die betriebliche Exzellenz die Continuous Integration und Continuous Delivery (CI/CD) (kontinuierliche Integration und kontinuierliche Bereitstellung), indem sie Entwickler dabei unterstützt, durchgängig qualitativ hochwertige Ergebnisse zu erzielen.

Designprinzipien

Nachfolgend finden Sie die Designprinzipien für betriebliche Exzellenz in der Cloud:

- Betriebliche Vorgänge als Code ausführen („Operations-as-Code“): In der Cloud können Sie die gleichen technischen Vorgehensweisen wie bei Anwendungscode in Ihrer gesamten Umgebung anwenden. Sie können sämtliche Workloads (Anwendungen, Infrastruktur usw.) als Code definieren und mit Code aktualisieren. Sie können Ihre operativen Verfahren als Code skripten und deren Ausführung automatisieren, indem Sie sie als Reaktion auf Ereignisse starten. Indem der Betrieb als Code ausgeführt wird, werden menschliche Fehler ausgeräumt und einheitliche Reaktionen auf Ereignisse geschaffen.
- Häufige, kleine, umkehrbare Änderungen vornehmen: Entwerfen Sie skalierbare und lose gekoppelte Workloads, sodass Komponenten regelmäßig aktualisiert werden können. Automatisierte Bereitstellungstechniken in Verbindung mit kleineren, inkrementellen Änderungen verringern den Angriffsradius 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.

- Bessern Sie betriebliche Verfahren regelmäßig nach: Entwickeln Sie bei der Weiterentwicklung Ihrer Workloads auch Ihre Abläufe entsprechend weiter. 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. Teilen Sie bewährte Praktiken und bilden Sie Ihre Teams auf spielerische Weise weiter.
- Beugen Sie Fehlern vor: Führen Sie vorbeugende Übungen durch, um potenzielle Fehlerquellen zu identifizieren, damit diese behoben oder umgangen werden können. Testen Sie Ihre Ausfallszenarien und stellen Sie sicher, dass Sie deren Auswirkungen kennen. Testen Sie Ihre Reaktionsverfahren, um sicherzustellen, dass diese wirksam sind und dass Ihre Teams mit deren Ausführung vertraut sind. Legen Sie regelmäßige Termine fest, an denen getestet wird, wie Workloads und Teams auf simulierte Ereignisse reagieren.
- Lernen Sie aus allen betrieblichen Ausfällen: Ziehen Sie aus allen betrieblichen Zwischenfällen und Ausfällen entsprechende Lehren und treiben Sie geeignete Verbesserungen voran. Geben Sie Ihre Erkenntnisse an alle Teams in Ihrer gesamten Organisation weiter.
- Verwenden Sie verwaltete Services: Verringern Sie die operative Belastung, indem Sie verwaltete AWS-Services nutzen, wo immer dies möglich ist. Erstellen Sie operative Verfahren für die Interaktion mit diesen Services.
- Implementieren Sie Beobachtbarkeit für umsetzbare Erkenntnisse: Verschaffen Sie sich einen umfassenden Überblick über das Verhalten, die Leistung, die Zuverlässigkeit, die Kosten und den Zustand von Workloads. 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.

Definition

Die bewährte Methoden für betriebliche 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 die 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. Sie müssen diese Risiken verstehen und eine fundierte Entscheidung dazu treffen, 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, helfen 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.

Organisation

Sie müssen die Prioritäten Ihres Unternehmens, die Unternehmensstruktur und die Unterstützung Ihrer Teammitglieder durch Ihr Unternehmen verstehen, damit sie Ihre Geschäftsergebnisse unterstützen können.

Um operative Exzellenz zu ermöglichen, müssen Sie Folgendes verstehen:

Themen

- Unternehmensprioritäten
- Betriebsmodell
- Unternehmenskultur

Unternehmensprioritäten

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. Überprüfen Sie Ihre Prioritäten regelmäßig, damit sie aktualisiert werden können, wenn sich die Anforderungen Ihrer Organisation ändern.

Bewährte Methoden

- OPS01-BP01 Bedürfnisse externer Kunden 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
- OPS01-BP07 Abwägen von Vorteilen und Risiken

OPS01-BP01 Bedürfnisse externer Kunden bewerten

Binden Sie alle wichtigen Beteiligten 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.

Gängige Antimuster:

- 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 eine neue Funktion, 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

- Kenntnis der geschäftlichen Anforderungen: Der geschäftliche Erfolg basiert auf gemeinsamen Zielen und der Kommunikation zwischen allen Beteiligten, zu denen auch die Teams aus den Bereichen Betriebswirtschaft, Entwicklung und Operationen gehören.
  - Überprüfen der geschäftlichen Ziele, Anforderungen und Prioritäten externer Kunden: Führen Sie wichtige Beteiligte 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 eines gemeinsamen Verständnisses: 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 Dokumente:

- AWS Well-Architected Framework-Konzepte – Feedbackschleife

OPS01-BP02 Bedürfnisse interner Kunden bewerten

Binden Sie alle wichtigen Beteiligten 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.

Anhand Ihrer etablierten Prioritäten können Sie dann 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.

Gängige Antimuster:

- 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

- Kenntnis der geschäftlichen Anforderungen: Der geschäftliche Erfolg basiert auf gemeinsamen Zielen und der Kommunikation zwischen allen Beteiligten, zu denen auch die Teams aus den Bereichen Geschäft, Entwicklung und Betrieb gehören.
  - Überprüfen der geschäftlichen Ziele, Anforderungen und Prioritäten interner Kunden: Führen Sie wichtige Beteiligte 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.
  - Übereinstimmendes Verständnis: Sorgen Sie dafür, dass alle Beteiligten die Geschäftsfunktionen des Workloads und die Rollen der einzelnen Teams bei den Workload-spezifischen Betriebsabläufen kennen. Außerdem sollte bekannt sein, wie diese Faktoren Ihre gemeinsamen Geschäftsziele mit internen und externen Kunden beeinflussen.

Ressourcen

Zugehörige Dokumente:

- AWS Well-Architected Framework-Konzepte – Feedbackschleife

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 bietet das AWS Well-Architected Tool , 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 Enterprise Support-Kunden kommt auch eine Betriebsüberprüfungin Frage, die ihnen helfen soll, Lücken in ihrem Ansatz für den Betrieb in der Cloud zu identifizieren.

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.

Gängige Antimuster:

- 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: Wenn Sie die Bedrohungen für Ihr Unternehmen und Ihren Workload identifizieren und verstehen, können Sie bestimmen, 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

- Bedrohungslandschaft bewerten: 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 eines Bedrohungsmodells: 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 Dokumente:

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

OPS01-BP06 Bewerten von Kompromissen

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.

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. Bei der Schulung Ihrer Teams sollten Sie die vom AWS Support (AWS Knowledge Center, AWS-Diskussionsforenund AWS Support Center) bereitgestellten Ressourcen und AWS-Dokumentation nutzen, um Ihre Teams zu schulen. 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.

Gängige Antimuster:

- Sie verwenden eine relationale Datenbank, um Zeitreihendaten und nicht relationale Daten zu verwalten. Es gibt Datenbankoptionen, die für Ihre verwendeten Datentypen optimiert sind. Sie sind sich der Vorteile aber nicht bewusst, da Sie die Unterschiede zwischen den Lösungsangeboten nicht evaluiert haben.
- Ihre Investoren fordern, dass Sie die Compliance mit Payment Card Industry Data Security Standards (PCI DSS) nachweisen. Sie denken nicht über die möglichen Kompromisse zwischen der Erfüllung dieser Anfrage und der Fortsetzung Ihrer derzeitigen Entwicklungsaktivitäten nach. Stattdessen fahren Sie mit der Entwicklung fort, ohne einen Compliance-Nachweis zu liefern. Ihre Investoren beenden die Unterstützung Ihres Unternehmens, da sie Bedenken bezüglich der Sicherheit Ihrer Plattform und ihrer Investitionen haben.

Vorteile der Einführung dieser bewährten Methode: Wenn Sie die Auswirkungen und Konsequenzen Ihrer Entscheidungen verstehen, können Sie die vorhandenen Optionen priorisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Kompromisse bewerten: Bewerten Sie die Auswirkungen von Kompromissen bei konkurrierenden Interessen, um fundiert zu entscheiden, auf welche Bereiche die operativen Anstrengungen konzentriert werden sollten. So kann beispielsweise die Beschleunigung der Markteinführung neuer Funktionen einen höheren Stellenwert haben als die Kostenoptimierung.
- 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. Bei der Schulung Ihrer Teams sollten Sie die vom AWS Support (AWS Knowledge Center,

AWS Discussion Forums und AWS Support Center) bereitgestellten Ressourcen und AWS-Dokumente nutzen. 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.

Ressourcen

Zugehörige Dokumente:

- AWS-Blog
- AWS Cloud-Compliance
- AWS-Diskussionsforen
- AWS-Dokumentation nutzen,
- AWS Knowledge Center
- AWS Support
- AWS Support Center
- Die Amazon Builders’ Library
- im offiziellen AWS-Podcast

OPS01-BP07 Abwägen von Vorteilen und Risiken

Wägen Sie die Vorteile und Risiken ab, um fundiert zu entscheiden, auf welche Bereiche die operativen Anstrengungen konzentriert werden sollten. So kann es beispielsweise sinnvoll sein, einen Workload mit noch offenen Problemen bereitzustellen, um den Kunden wichtige neue Funktionen zur Verfügung zu stellen. 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 Risikoproblem zu beheben.

Manchmal kann es vorkommen, dass man zu viel Augenmerk auf eine kleine Auswahl von operativen Prioritäten richtet. Gehen Sie langfristig gut ausgewogen vor, um sicherzustellen, dass erforderliche Fähigkeiten entwickelt und Risiken verwaltet werden. Wenn sich Anforderungen ändern, aktualisieren Sie Ihre Prioritäten entsprechend.

Gängige Antimuster:

- Sie haben sich entschieden, eine Bibliothek einzubinden, die „alle nötigen Funktionen“ bietet und von einem Ihrer Entwickler „im Internet gefunden“ wurde. 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.
- Sie haben sich entschieden, eine neue Funktion zu entwickeln und bereitzustellen, statt ein vorhandenes Problem zu beheben. Sie haben keine Bewertung der Risiken durchgeführt, die das vorhandene Problem in der bereitgestellten Funktion bergen könnte, und wissen nicht, welche Folgen daraus für Ihre Kunden entstehen.
- Sie haben sich entschieden, eine häufig von Kunden angeforderte Funktion nicht bereitzustellen, weil Ihr Compliance-Team unbestimmte Bedenken geäußert hat.

Vorteile der Einführung dieser bewährten Methode: Wenn Sie die verfügbaren Vorteile Ihrer Optionen ermitteln und sich der Risiken für Ihr Unternehmen bewusst sind, können Sie fundierte Entscheidungen treffen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

- Abwägen von Vorteilen und Risiken: Wägen Sie den Nutzen von Entscheidungen gegen die damit einhergehenden Risiken ab.
  - Ermitteln von Vorteilen: Ermitteln Sie die Vorteile auf Basis der geschäftlichen Ziele, Anforderungen und Prioritäten. Zu diesen Prioritäten können beispielsweise eine kurze Markteinführungszeit, Sicherheit, Zuverlässigkeit, Leistung und Kosten zählen.
  - Ermitteln von Risiken: Ermitteln Sie die Risiken auf Basis der geschäftlichen Ziele, Anforderungen und Prioritäten. Zu diesen Prioritäten können beispielsweise eine kurze Markteinführungszeit, Sicherheit, Zuverlässigkeit, Leistung und Kosten zählen.
  - 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 Beteiligten, 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, und anhand 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.

Betriebsmodell

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams müssen ihre Rollen beim Erfolg anderer Teams verstehen, die Rolle anderer Teams bei ihrem eigenen Erfolg und sie müssen 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.

Die Anforderungen eines Teams werden durch die Branche, das Unternehmen, die Zusammensetzung des Teams und die Merkmale seiner Workloads beeinflusst. Es ist nicht sinnvoll, davon auszugehen, dass ein einziges Betriebsmodell alle Teams und ihre Workloads unterstützen kann.

Die Anzahl der Betriebsmodelle in einem Unternehmen wird mit der Anzahl der Entwicklungsteams wahrscheinlich steigen. Möglicherweise müssen Sie eine Kombination aus Betriebsmodellen verwenden.

Die Übernahme von Standards und die Nutzung von Services kann den Betrieb vereinfachen und den Support-Aufwand in Ihrem Betriebsmodell begrenzen. Der Vorteil der Entwicklungsbemühungen zu gemeinsamen Standards wird durch die Anzahl der Teams verstärkt, die den Standard eingeführt haben und neue Funktionen übernehmen werden.

Es ist wichtig, dass Mechanismen vorhanden sind, um Ergänzungen, Änderungen und Ausnahmen zu Standards zur Unterstützung der Aktivitäten der Teams anzufordern. Ohne diese Option werden Standards zu einer Einschränkung der Innovation. Anträge sollten nach einer Bewertung der Vorteile und Risiken genehmigt werden, wenn sie sinnvoll sind.

Eine klar definierte Gruppe von Verantwortlichkeiten verringert die Häufigkeit widersprüchlicher und redundanter Bemühungen. Geschäftsergebnisse sind leichter zu erzielen, wenn es eine starke Ausrichtung und Beziehungen zwischen Geschäfts-, Entwicklungs- und Betriebsteams gibt.

Betriebsmodell-2-mal-2-Darstellungen

Diese Betriebsmodell-2-mal-2-Darstellungen sind Abbildungen, die Ihnen helfen, die Beziehungen zwischen Teams in Ihrer Umgebung zu verstehen. Diese Diagramme konzentrieren sich darauf, wer was tut und welche Beziehungen zwischen Teams bestehen. Wir werden jedoch auch Governance und Entscheidungsfindung im Kontext dieser Beispiele besprechen.

Unsere Teams haben möglicherweise Zuständigkeiten in mehreren Teilen mehrerer Modelle, abhängig von den Workloads, die sie unterstützen. Möglicherweise möchten Sie spezialisiertere

Disziplinbereiche als die beschriebenen High-Level-Bereiche aufschlüsseln. Es besteht das Potenzial für endlose Variationen bei diesen Modellen, wenn Sie Aktivitäten trennen oder aggregieren oder Teams überlagern und spezifischere Details bereitstellen.

Sie können feststellen, dass Sie sich überschneidende oder nicht erkannte Funktionen in Teams haben, die einen zusätzlichen Vorteil bieten oder zu Effizienzsteigerungen führen können. Sie können auch unbefriedigte Bedürfnisse in Ihrem Unternehmen identifizieren, die Sie berücksichtigen können.

Prüfen Sie bei der Bewertung der organisatorischen Veränderungen die Kompromisse zwischen Modellen, wo sich Ihre einzelnen Teams innerhalb der Modelle befinden (jetzt und nach der Änderung), wie sich die Beziehung und Verantwortlichkeiten Ihrer Teams ändern werden und ob die Vorteile die Auswirkungen auf Ihr Unternehmen rechtfertigen.

Sie können mit jedem der folgenden vier Betriebsmodelle erfolgreich sein. Einige Modelle eignen sich besser für bestimmte Anwendungsfälle oder an bestimmten Punkten in Ihrer Entwicklung. Einige dieser Modelle bieten möglicherweise Vorteile gegenüber denjenigen, die in Ihrer Umgebung verwendet werden.

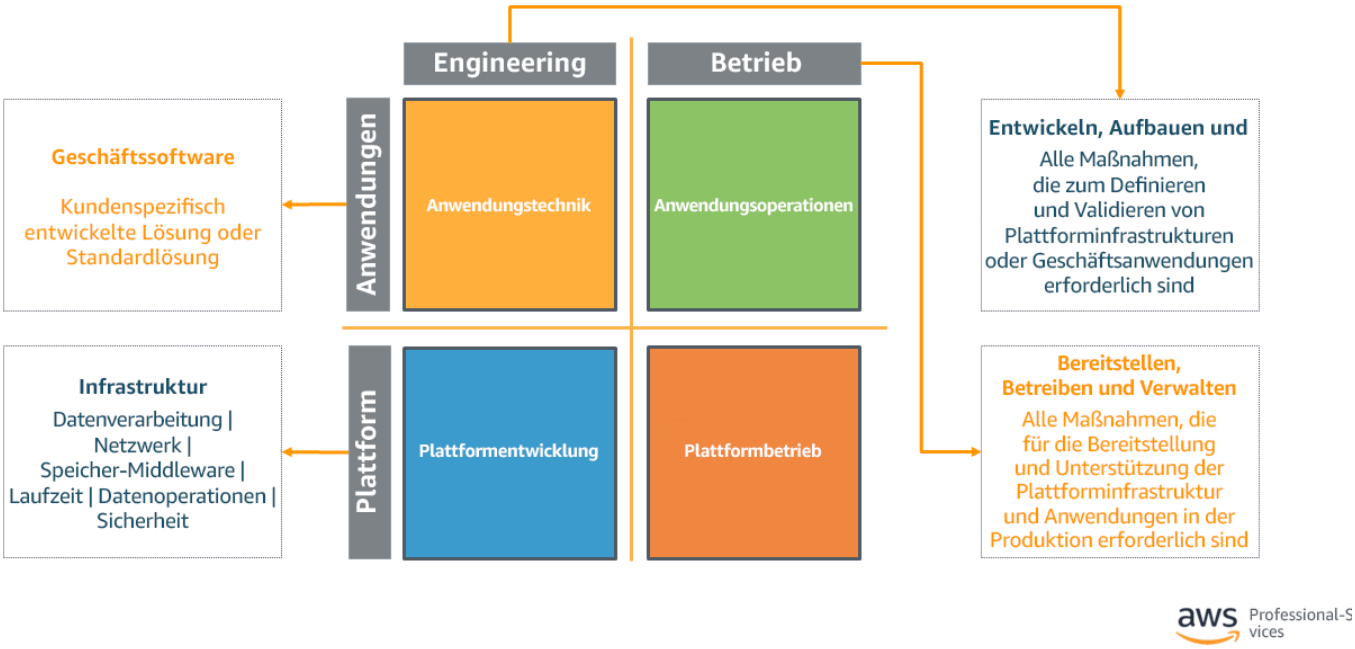
Themen

Vollständig getrenntes Betriebsmodell

Im folgenden Diagramm werden auf der vertikalen Achse Anwendungen und Infrastruktur angezeigt. Anwendungen beziehen sich auf den Workload, der einem Geschäftsergebnis dient, und können kundenspezifisch entwickelte oder gekaufte Software sein. Infrastruktur bezieht sich auf die physische und virtuelle Infrastruktur und andere Software, die diesen Workload unterstützt.

Auf der horizontalen Achse haben wir Engineering und Operations. Engineering bezieht sich auf die Entwicklung, Erstellung und das Testen von Anwendungen und Infrastruktur. Operations ist die Bereitstellung, Aktualisierung und laufende Unterstützung von Anwendungen und Infrastruktur.

Traditionelles Modell



In vielen Unternehmen ist dieses „vollständig getrennte“ Modell vorhanden. Die Aktivitäten in jedem Quadranten werden von einem separaten Team ausgeführt. Die Arbeit wird zwischen Teams über Mechanismen wie Arbeitsanfragen, Arbeitswarteschlangen, Tickets oder über ein IT-Service-Management (ITSM)-System weitergegeben.

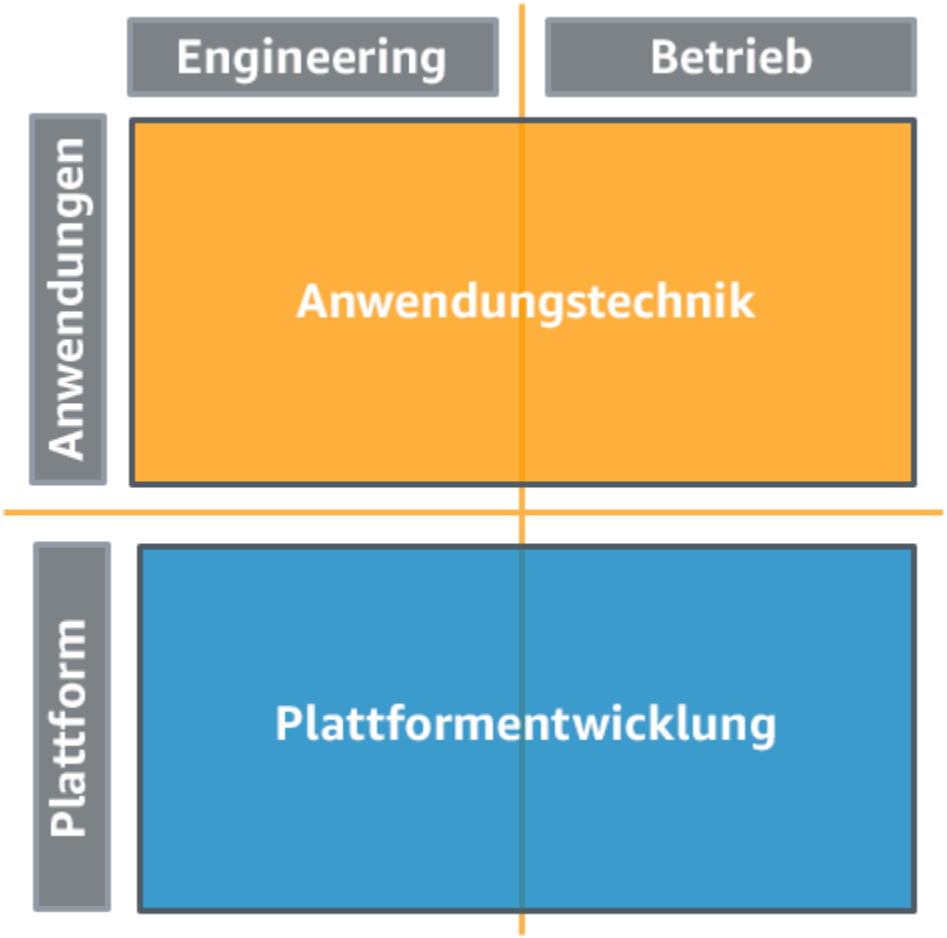
Der Übergang von Aufgaben zu oder zwischen Teams erhöht die Komplexität und schafft Engpässe und Verzögerungen. Anfragen können verzögert werden, bis sie eine Priorität haben. Verspätet erkannte Fehler erfordern möglicherweise eine erhebliche Nachbearbeitung und müssen möglicherweise die gleichen Teams und ihre Funktionen erneut durchlaufen. Wenn es Vorfälle gibt, die Maßnahmen durch Technikerteams erfordern, verzögern sich ihre Antworten durch die Übergabe der Aktivität.

Es besteht ein höheres Risiko einer Fehlausrichtung, wenn Geschäfts-, Entwicklungs- und Betriebsteams um die ausgeführten Aktivitäten oder Funktionen herum organisiert sind. Dies kann dazu führen, dass Teams sich auf ihre spezifischen Verantwortlichkeiten konzentrieren, anstatt sich auf das Erreichen von Geschäftsergebnissen zu konzentrieren. Teams können eng spezifiziert, physisch isoliert oder logisch isoliert sein, was die Kommunikation und Zusammenarbeit behindert.

Getrenntes Application Engineering and Operations (AEO) und Infrastructure Engineering and Operations (IEO) mit zentralisierter Governance

Dieses Modell "Getrennte AEO und IEO" folgt einer "You build it you run it"-Methodik.

Ihre Anwendungstechniker und Entwickler führen sowohl das Engineering als auch den Betrieb ihrer Workloads durch. Ebenso führen Ihre Infrastrukturtechniker sowohl das Engineering als auch den Betrieb der Plattformen durch, die sie zur Unterstützung von Anwendungsteams verwenden.



In diesem Beispiel behandeln wir Governance als zentralisiert. Standards werden an die Anwendungsteams verteilt, bereitgestellt oder weitergegeben.

Sie sollten Tools oder Services verwenden, mit denen Sie Ihre Umgebungen kontenübergreifend verwalten können, z. B. AWS Organizations. Services wie AWS Control Tower erweitern diese Verwaltungsfunktion, sodass Sie Pläne (die Ihre Betriebsmodelle unterstützen) für die Einrichtung von Konten definieren, laufende Governance mit AWS Organizations anwenden und die Bereitstellung neuer Konten automatisieren können.

"You build it you run it" bedeutet nicht, dass das Anwendungsteam für den gesamten Stack, die Tool-Chain und die Plattform verantwortlich ist.

Das Plattform-Engineering-Team bietet eine standardisierte Reihe von Services (z. B. Entwicklungstools, Überwachungstools, Sicherungs- und Wiederherstellungstools sowie Netzwerk) für das Anwendungsteam. Das Plattformteam kann dem Anwendungsteam auch Zugriff auf genehmigte Cloud-Anbieter-Services, bestimmte Konfigurationen derselben oder beides gewähren.

Mechanismen, die eine Self-Service-Funktion für die Bereitstellung genehmigter Services und Konfigurationen bereitstellen, wie z. B. Service Catalog, können helfen, Verzögerungen im Zusammenhang mit Erfüllungsanfragen einzuschränken und gleichzeitig Governance zu erzwingen.

Das Plattformteam ermöglicht eine vollständige Stack-Transparenz, sodass Anwendungsteams, die ihre Anwendungen nutzen, zwischen Problemen mit ihren Anwendungskomponenten und den Services und Infrastrukturkomponenten unterscheiden können. Das Plattformteam kann auch Unterstützung bei der Konfiguration dieser Services leisten und Anleitungen zur Verbesserung des Betriebs der Anwendungsteams bieten.

Wie bereits erwähnt, ist es wichtig, dass für das Anwendungsteam Mechanismen vorhanden sind, um Ergänzungen, Änderungen und Ausnahmen zu Standards zur Unterstützung der Aktivitäten der Teams und der Innovation ihrer Anwendung anzufordern.

Das "Separated AEO IEO"-Modell bietet starke Feedback-Schleifen für Anwendungsteams. Der tägliche Betrieb eines Workloads erhöht den Kontakt mit Kunden entweder durch direkte Interaktion oder indirekt durch Support- und Funktionsanfragen. Durch diese erhöhte Sichtbarkeit können Anwendungsteams Probleme schneller beheben. Das tiefere Engagement und die engere Beziehung bieten Einblicke in die Kundenbedürfnisse und ermöglichen schnellere Innovationen.

All dies gilt auch für das Plattformteam, das die Anwendungsteams unterstützt.

Übernommene Standards können vorab für die Verwendung genehmigt werden, wodurch der für die Produktion erforderliche Prüfungsumfang reduziert wird. Durch den Einsatz von durch das Plattformteam bereitgestellte unterstützte und getestete Standards kann die Häufigkeit von Problemen mit diesen Services reduziert werden. Durch die Übernahme von Standards können sich Anwendungsteams auf die Differenzierung ihrer Workloads konzentrieren.

Getrennte AEO und IEO mit zentralisierter Governance und einem Serviceanbieter

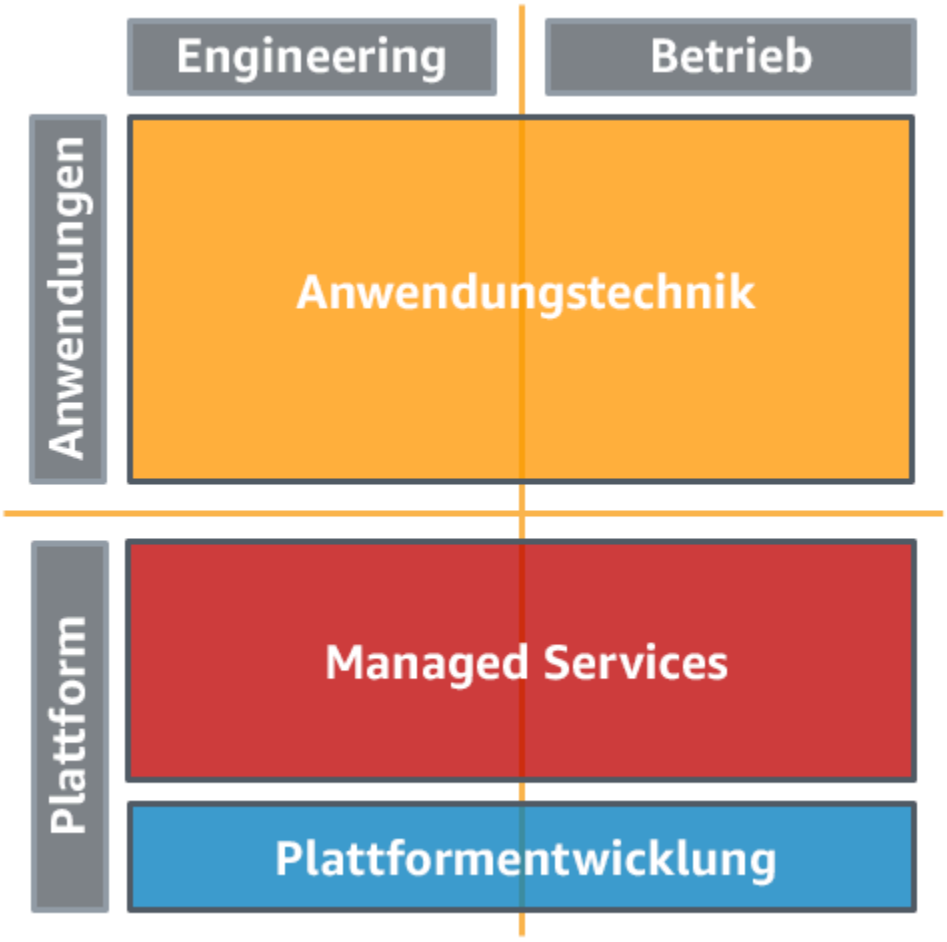
Dieses Modell "Getrennte AEO und IEO" folgt einer "You build it you run it"-Methodik.

Ihre Anwendungstechniker und Entwickler führen sowohl das Engineering als auch den Betrieb ihrer Workloads durch.

Ihr Unternehmen verfügt möglicherweise nicht über die vorhandenen Fähigkeiten oder Teammitglieder, um ein dediziertes Plattform-Engineering- und Betriebsteam zu unterstützen, oder Sie möchten nicht mehr Zeit und Aufwand dafür investieren.

Alternativ können Sie ein Plattformteam haben, das sich auf die Entwicklung von Funktionen konzentriert, die Ihr Unternehmen differenzieren, aber Sie möchten den undifferenzierten täglichen Betrieb an einen Outsourcer auslagern.

Anbieter von verwalteten Services wie AWS Managed Services, AWS Managed Services-Partneroder Anbieter von verwalteten Services im AWS-Partnernetzwerkstellen Fachwissen zur Implementierung von Cloud-Umgebungen bereit und unterstützen Ihre Sicherheits- und Compliance-Anforderungen und Geschäftsziele.



Für diese Variante behandeln wir Governance als zentralisiert und verwaltet durch das Plattformteam, wobei die Kontoerstellung und Richtlinien mit AWS Organizations und AWS Control Tower verwaltet werden.

Dieses Modell erfordert, dass Sie Ihre Mechanismen so ändern, dass sie mit denen Ihres Serviceproviders zusammenarbeiten. Es löst nicht Engpässe und Verzögerungen, die durch den Übergang von Aufgaben zwischen Teams, einschließlich Ihres Serviceproviders, oder durch den potenziellen Nachbearbeitungsaufwand im Zusammenhang mit der verspäteten Fehlererkennung entstehen.

Sie profitieren von den Standards, bewährten Methoden, Prozessen und dem Fachwissen Ihrer Anbieter. Außerdem profitieren Sie von den Vorteilen der fortlaufenden Entwicklung ihrer Service-Angebote.

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.

Getrennte AEO und IEO mit zentralisierter Governance und einem internen Serviceanbieter-Beratungspartner

Dieses „Getrennte AEO und IEO“-Modell folgt einer „You build it you run it“-Methodik.

Sie möchten, dass Ihre Anwendungsteams die technischen und betrieblichen Aktivitäten für ihre Workloads durchführen und eine eher DevOps-ähnliche Kultur annehmen.

Ihre Anwendungsteams sind möglicherweise gerade dabei zu migrieren, die Cloud einzuführen oder Ihre Workloads zu modernisieren, und verfügen nicht über die erforderlichen Fähigkeiten, um die Cloud und den Cloud-Betrieb adäquat zu unterstützen. Dieser Mangel an Fähigkeiten oder Vertrautheit des Anwendungsteams kann Ihre Bemühungen behindern.

Richten Sie ein Cloud Center of Enablement-Team (CCoE) ein, das ein Forum bietet, um Fragen zu stellen, Bedürfnisse zu diskutieren und Lösungen zu finden und diesem Problem so zu begegnen. Je nach den Bedürfnissen Ihres Unternehmens kann das CCoE ein spezielles Expertenteam oder ein virtuelles Team sein, dessen Teilnehmer aus Ihrem gesamten Unternehmen ausgewählt werden. Das CCoE ermöglicht die Cloud-Transformation für Teams, etabliert eine zentralisierte Cloud-Governance und definiert Standards für das Konto- und Organisationsmanagement. Außerdem identifizieren sie erfolgreiche Referenzarchitekturen und Muster für den Einsatz in Unternehmen.

Wir bezeichnen CCoE als Cloud Center of Enablement und nicht als Cloud Center of Excellence, um den Erfolg der unterstützten Teams und das Erreichen von Geschäftsergebnissen in den Vordergrund zu stellen.

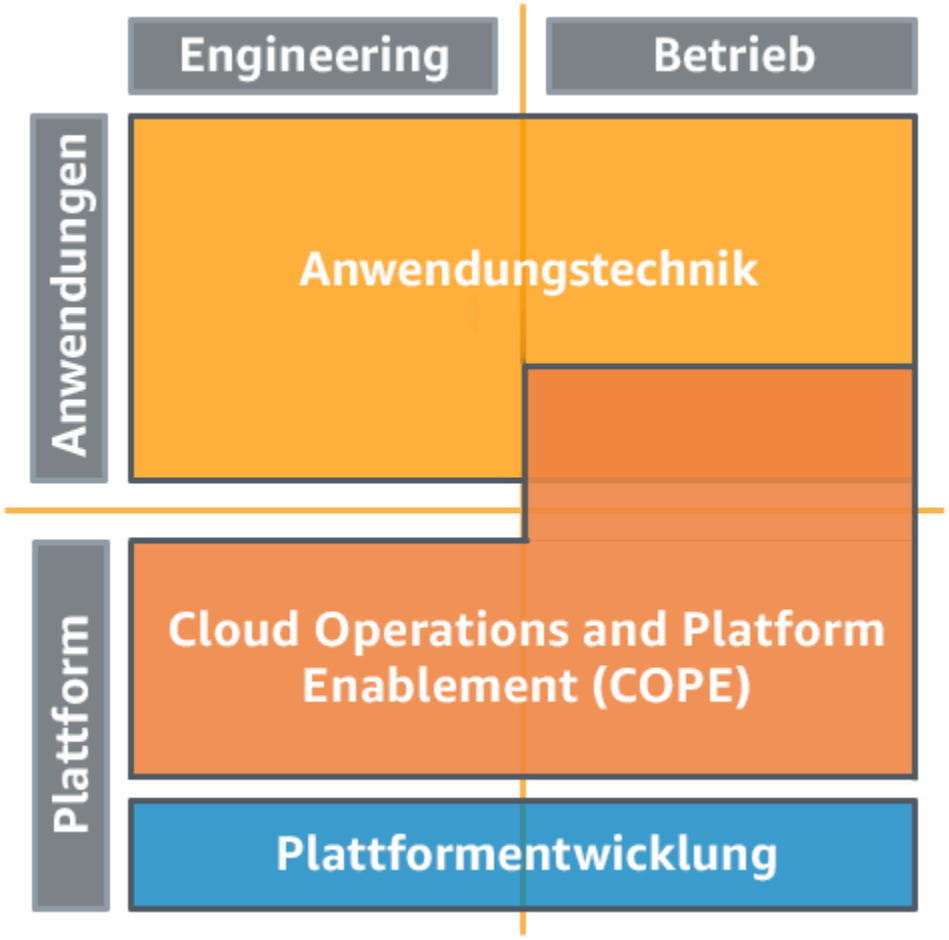
Ihr Plattform-Engineering-Team entwickelt die wichtigsten gemeinsamen Plattformfunktionen auf der Grundlage dieser Standards, die von den Anwendungsteams übernommen werden können. Sie kodifizieren die Unternehmensreferenzarchitekturen und -muster, die den Anwendungsteams über einen Selbstbedienungsmechanismus zur Verfügung gestellt werden. Mithilfe eines Services wie AWS Service Catalog können die Anwendungsteams genehmigte Referenzarchitekturen, -muster, -dienste und -konfigurationen einsetzen, die standardmäßig mit den zentralisierten Governance- und Sicherheitsstandards konform sind.

Das Plattform-Engineering-Team bietet eine standardisierte Reihe von Services (z. B. Entwicklungstools, Überwachungstools, Sicherungs- und Wiederherstellungstools sowie Netzwerk) für das Anwendungsteam.

Ihr Unternehmen verfügt über einen „internen MSP- und Beratungspartner“, der die standardisierten Services verwaltet und unterstützt und den Anwendungsteams beim Aufbau ihrer Cloud-Präsenz auf der Grundlage der Referenzarchitekturen und -muster behilflich ist. Dieses „Cloud Operations and Platform Enablement (COPE)“-Team arbeitet mit den Anwendungsteams zusammen, um sie bei der Einrichtung eines Basisbetriebs zu unterstützen, wobei die Anwendungsteams im Laufe der Zeit immer mehr Verantwortung für ihre Systeme und Ressourcen übernehmen. Das COPE-Team treibt gemeinsam mit den CCoE- und Platform-Engineering-Teams kontinuierliche Verbesserungen voran und fungiert als Ansprechpartner für die Anwendungsteams.

Die Anwendungsteams erhalten Unterstützung bei der Einrichtung von Umgebungen, CICD-Pipelines, Änderungsverwaltung, Beobachtbarkeit und Überwachung sowie bei der Einrichtung von Incident- und Event-Management-Prozessen, die bei Bedarf mit denen des Unternehmens integriert werden. Das COPE-Team beteiligt sich gemeinsam mit den Anwendungsteams an der Durchführung dieser operativen Tätigkeiten, wobei die Beteiligung des COPE-Teams im Laufe der Zeit abnimmt, wenn die Anwendungsteams die Verantwortung übernehmen.

Das Anwendungsteam profitiert von den Fähigkeiten des COPE-Teams und den Erfahrungen, die das Unternehmen gemacht hat. Sie werden durch den Integritätsschutz geschützt, der durch die zentralisierte Governance geschaffen wurden. Das Anwendungsteam baut auf anerkannten Erfolgen auf und profitiert von der kontinuierlichen Weiterentwicklung der von ihm angenommenen Organisationsstandards. Durch den Prozess der Beobachtung und Überwachung erhalten sie einen besseren Einblick in die Funktionsweise ihres Workloads und können die Auswirkungen von Änderungen, die sie an ihrem Workload vornehmen, besser verstehen.



Das COPE-Team behält den Zugang, der für die Unterstützung von Betriebsaktivitäten, die Bereitstellung einer Unternehmenssicht, die das Anwendungsteam übergreift, und die Unterstützung beim Management kritischer Vorfälle erforderlich ist. Das COPE-Team behält die Verantwortung für Tätigkeiten, die als undifferenzierte Schwerstarbeit angesehen werden und die es durch Standardlösungen, die in großem Umfang unterstützt werden können, erfüllt. Sie verwalten auch weiterhin gut verstandene programmatische und automatisierte Betriebsaktivitäten für die Anwendungsteams, damit diese sich auf die Differenzierung ihrer Anwendungen konzentrieren können.

Sie profitieren von den Standards, bewährten Verfahren, Prozessen und dem Fachwissen Ihres Unternehmens, das sich aus den Erfolgen Ihrer Teams ergibt. Sie schaffen einen Mechanismus, um diese erfolgreichen Muster für neue Teams, die die Cloud einführen oder modernisieren, zu reproduzieren. Bei diesem Modell liegt der Schwerpunkt auf der Fähigkeit des COPE-Teams, das Anwendungsteam bei der Etablierung und dem Übergang von Wissen und Artefakten zu unterstützen. Es verringert die operative Belastung der Anwendungsteams und birgt das Risiko, dass

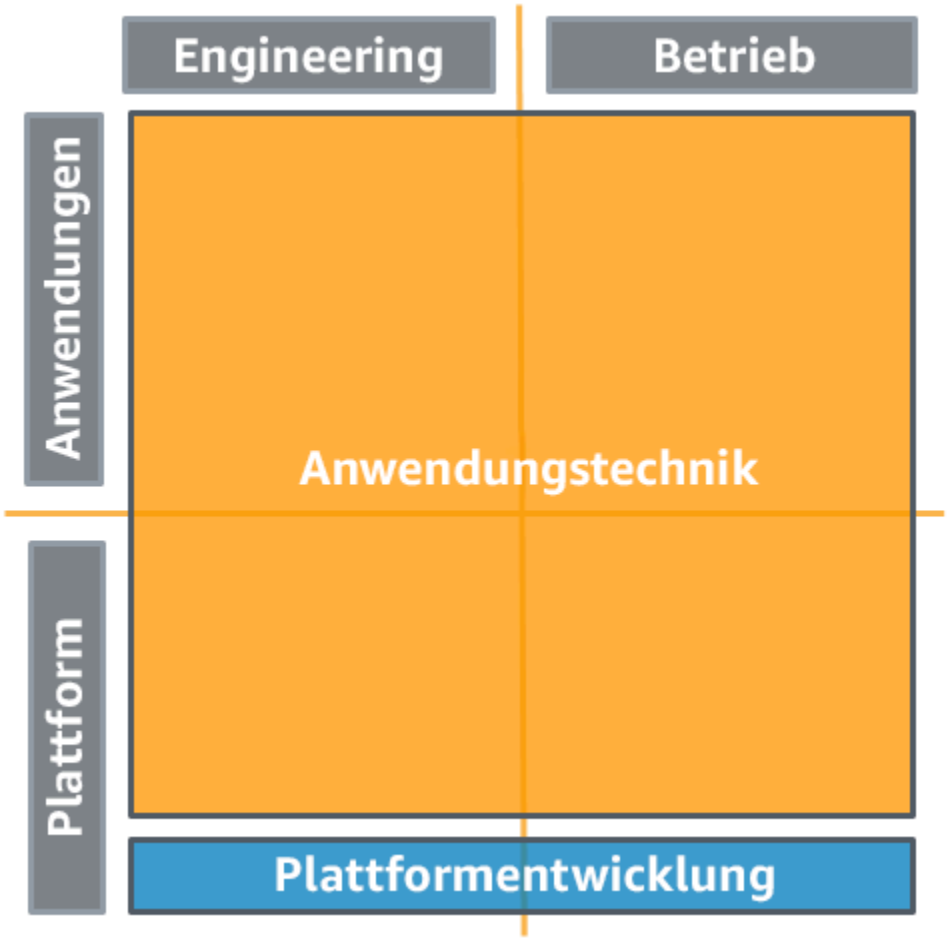
die Anwendungsteams nicht weitgehend unabhängig werden. Es stellt Beziehungen zwischen CCoE, COPE und Anwendungsteams her und schafft so eine Feedbackschleife, die weitere Entwicklungen und Innovationen unterstützt.

Die Einrichtung von CCoE- und COPE-Teams und die Festlegung unternehmensweiter Standards können die Cloud-Einführung erleichtern und Modernisierungsbemühungen unterstützen. Durch die zusätzliche Unterstützung eines COPE-Teams, das Ihren Anwendungsteams als Berater und Partner zur Seite steht, können Sie Hindernisse aus dem Weg räumen, die die Annahme der nützlichen Cloud-Funktionen durch die Anwendungsteams verzögern.

Getrennte AEO und IEO mit dezentralisierter Governance

Dieses Modell "Getrennte AEO und IEO" folgt einer "You build it you run it"-Methodik.

Ihre Anwendungstechniker und Entwickler führen sowohl das Engineering als auch den Betrieb ihrer Workloads durch. Ebenso führen Ihre Infrastrukturtechniker sowohl das Engineering als auch den Betrieb der Plattformen durch, die sie zur Unterstützung von Anwendungsteams verwenden.



In diesem Beispiel behandeln wir Governance als dezentralisiert.

Standards werden nach wie vor vom Plattformteam verteilt, bereitgestellt oder an Anwendungsteams weitergegeben, aber Anwendungsteams können neue Plattformfunktionen zur Unterstützung ihres Workloads entwickeln und betreiben.

Bei diesem Modell gibt es weniger Einschränkungen für das Anwendungsteam, aber das ist mit einer erheblichen Zunahme der Verantwortlichkeiten verbunden. Zusätzliche Fähigkeiten und potenziell auch zusätzliche Teammitglieder müssen vorhanden sein, um die zusätzlichen Plattformfunktionen zu unterstützen. Das Risiko signifikanter Nachbearbeitung wird erhöht, wenn die Qualifikationen nicht ausreichend sind und Fehler nicht frühzeitig erkannt werden.

Sie sollten Richtlinien erzwingen, die nicht spezifisch an Anwendungsteams delegiert sind. Verwenden Sie Tools oder Services, mit denen Sie Ihre Umgebungen kontenübergreifend zentral steuern können, z. B. AWS Organizations. Services wie AWS Control Tower erweitern diese Verwaltungsfunktion, sodass Sie Pläne (die Ihre Betriebsmodelle unterstützen) für die Einrichtung von Konten definieren, laufende Governance mit AWS Organizations anwenden und die Bereitstellung neuer Konten automatisieren können.

Es ist vorteilhaft, dass das Anwendungsteam Mechanismen hat, um Ergänzungen und Änderungen an Standards anzufordern. Sie können möglicherweise neue Standards bereitstellen, die anderen Anwendungsteams Vorteile bieten können. Die Plattformteams können entscheiden, dass die direkte Unterstützung für diese zusätzlichen Funktionen eine effektive Unterstützung für Geschäftsergebnisse darstellt.

Dieses Modell begrenzt Einschränkungen bei einer Innovation mit erheblichen Anforderungen an Fähigkeiten und Teammitglieder. Es behebt viele der Engpässe und Verzögerungen, die durch den Übergang von Aufgaben zwischen Teams entstehen, und fördert gleichzeitig die Entwicklung effektiver Beziehungen zwischen Teams und Kunden.

Beziehungen und Eigentümerschaft

Ihr Betriebsmodell definiert die Beziehungen zwischen Teams und unterstützt identifizierbare Eigentümerschaft und Verantwortlichkeit.

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 Teammitglieder wissen, wofür sie verantwortlich sind
- OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeit und Eigentümerschaft sind vorhanden
- OPS02-BP06 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden
- OPS02-BP07 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 Verantwortlichkeit und die Kontaktinformationen angegeben sind.

Implementierungsschritte

1. Beginnen Sie damit, die Verantwortlichkeit 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 allen 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 wenn erforderlich 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 Tagging Best Practices (Bewährte Methoden für das Tagging).
4. Erstellen Sie für andere Ressourcen, Plattformen und Infrastruktur eine Dokumentation zur Verantwortlichkeit. Diese sollte für alle Teammitglieder zugänglich sein.

Grad des Aufwands für den Implementierungsplan: 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 – Die Prozesse und Verfahren für den Support von Ressourcen hängen von der Verantwortlichkeit für die Ressource ab.
- OPS02-BP04 Teammitglieder wissen, wofür sie verantwortlich sind – Die Teammitglieder müssen verstehen, für welche Ressourcen sie verantwortlich sind.
- OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeit und Eigentümerschaft sind vorhanden – Die Verantwortlichkeit muss sich über Mechanismen wie Tags oder Kontaktinformationen zum Konto ermitteln lassen.

Zugehörige Dokumente:

- AWS Account Management - Updating contact information (AWS Account Management – Aktualisieren der Kontaktinformationen)
- AWS Config-Regeln – required-tags
- AWS Organizations – Aktualisieren alternativer Kontakte in Ihrer Organisation

- AWS-Whitepaper Tagging Best Practices (Bewährte Methoden für das Tagging)

Zugehörige Beispiele:

- AWS Config Rules - Amazon EC2 with required tags and valid values (AWS Config-Regeln – Amazon EC2 mit erforderlichen Tags und gültigen Werten)

Zugehörige Services:

- AWS Config
- 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.

Vorteile der Einführung dieser bewährten Methode: Anhand der Zuständigkeit kann identifiziert werden, wer Verbesserungen genehmigen, diese Verbesserungen implementieren oder beides durchführen kann.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Prozesse und Verfahren haben feste Besitzer, die für ihre Definition verantwortlich sind: Dokumentieren Sie die Prozesse und Verfahren, die in Ihrer Umgebung angewendet werden, sowie die Person oder Personen, die für die Definition verantwortlich sind.
  - Identifizieren von Prozessen und Verfahren: Identifizieren Sie die Betriebsaktivitäten, die zur Unterstützung Ihrer Workloads durchgeführt werden. Dokumentieren Sie diese Aktivitäten an einem auffindbaren Ort.
  - Definieren der Zuständigkeit für die Definition eines Prozesses oder Verfahrens: 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 verantwortlich, detailliertes Feedback bereitzustellen, das für die Verbesserung der Aktivität erforderlich ist.

- Erfassen der Zuständigkeit in den Metadaten des Aktivitätsartefakts: Verfahren, die in Services wie AWS Systems Manager (durch Dokumente) und AWS Lambda (als Funktionen) automatisiert werden, unterstützen die Erfassung von Metadateninformationen als Tags. Erfassen Sie die Ressourcenzuständigkeit mithilfe von Tags oder Ressourcengruppen und geben Sie Zuständigkeits- und Kontaktinformationen an. Verwenden Sie AWS Organizations, um Markierungsrichtlinien zu erstellen und zu gewährleisten, dass Zuständigkeits- und Kontaktinformationen erfasst werden.

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.

Vorteile der Einführung dieser bewährten Methode:i Wenn die verantwortliche Person für die Durchführung einer Aktivität bekannt ist, wissen Sie, 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.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind: Erfassen Sie die Verantwortung für die Durchführung von Prozessen und Verfahren in Ihrer Umgebung.
  - Identifizieren von Prozessen und Verfahren: Identifizieren Sie die Betriebsaktivitäten, die zur Unterstützung Ihrer Workloads durchgeführt werden. Dokumentieren Sie diese Aktivitäten an einem auffindbaren Ort.
  - Definieren der Verantwortlichkeit für die Durchführung von Aktivitäten: Legen Sie das Team fest, das für eine Aktivität verantwortlich ist. Stellen Sie sicher, dass die Teammitglieder die Details der Aktivität und die erforderlichen Qualifikationen haben und über die entsprechenden Berechtigungen, Zugriffsrechte und Tools für die Durchführung der Aktivität verfügen. Sie müssen die Bedingung kennen, unter denen die Aktivität ausgeführt werden soll (z. B. nach

einem Ereignis oder gemäß einem Zeitplan). Diese Informationen sollten leicht auffindbar sein, damit Mitglieder Ihrer Organisation herausfinden können, an wen sie sich für bestimmte Anforderungen wenden müssen (Team oder Person).

OPS02-BP04 Teammitglieder wissen, wofür sie verantwortlich sind

Wenn Ihnen die Verantwortlichkeiten Ihrer Rolle bekannt sind und Sie wissen, wie Sie zu Geschäftsergebnissen beitragen, können Sie Ihre Aufgaben entsprechend priorisieren und die Bedeutung Ihrer Rolle nachvollziehen. Auf diese Weise können Teammitglieder Anforderungen erkennen und entsprechend reagieren.

Vorteile der Nutzung dieser bewährten Methode: Das Verständnis Ihrer Verantwortlichkeiten wirkt sich auf Ihre Entscheidungen, Ihre Aktionen und die Übergabe von Aktivitäten an die ordnungsgemäßen Besitzer aus.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

- Sicherstellen, dass Teammitglieder ihre Rollen und Verantwortlichkeiten verstehen: Legen Sie die Rollen und Verantwortlichkeiten von Teammitgliedern fest und stellen Sie sicher, dass sie die Erwartungen ihrer Rolle verstehen. Diese Informationen sollten leicht auffindbar sein, damit Mitglieder Ihrer Organisation herausfinden können, an wen sie sich für bestimmte Anforderungen wenden müssen (Team oder Person).

OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeit und Eigentümerschaft sind vorhanden

Wenn keine Person oder Personen festgelegt sind, gibt es definierte Eskalationsabläufe, um eine Person zu kontaktieren, die berechtigt ist, die fehlende Zuständigkeit zuzuweisen oder die Erfüllung einer Anforderung zu planen.

Vorteile der Einführung 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. Das Vorhandensein einer festgelegten Person, die berechtigt ist, Verantwortlichkeiten oder Zuständigkeiten zuzuweisen oder die Erfüllung von Anforderungen zu planen, reduziert das Risiko, dass Aufgaben liegen bleiben oder Anforderungen nicht erfüllt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Mechanismen zur Identifizierung von Verantwortlichkeit und Eigentümerschaft sind vorhanden: Stellen Sie Mitgliedern Ihrer Organisation zugängliche Mechanismen bereit, um Zuständigkeiten und Verantwortlichkeiten zu ermitteln und zuzuordnen. Auf diese Weise können sie bestimmen, an wen sie sich für bestimmte Anforderungen wenden müssen (Team oder Person).

OPS02-BP06 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.

Grad des Aufwands für den Implementierungsplan: 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 - Foundation playbook for AWS large migrations: Creating RACI matrices (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-BP07 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.

Grad des Aufwands für den Implementierungsplan: 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 - Empowering Innovation with the Two-Pizza Team (AWS Executive Insights – Mit dem Zwei-Pizza-Team Innovationen vorantreiben)
- Introduction to DevOps on AWS - Two-Pizza Teams (Einführung in DevOps in AWS – Zwei-Pizza-Teams)

Unternehmenskultur

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

Bewährte Methoden

- OPS03-BP01 Förderung durch die Geschäftsführung

- 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 in die Lage versetzt und ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern:
- OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten
- OPS03-BP08 Unterschiedliche Meinungen werden innerhalb des Teams und teamübergreifend gefördert und sind erwünscht

OPS03-BP01 Förderung durch die Geschäftsführung

Die Geschäftsführung legt klare Erwartungen für das Unternehmen fest und bewertet den Erfolg. Die Geschäftsführung ist Sponsor, Fürsprecher und treibende Kraft für die Übernahme bewährter Methoden und die Weiterentwicklung des Unternehmens

Vorteile der Einführung dieser bewährten Methode: Eine engagierte Geschäftsführung, klar kommunizierte Erwartungen und gemeinsame Ziele stellen sicher, dass die Teammitglieder wissen, was von ihnen erwartet wird. Mit der Erfolgsevaluierung können die Hindernisse auf dem Weg zum Erfolg identifiziert und durch die Intervention der Geschäftsführung oder ihrer Delegierten behoben werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Förderung durch Geschäftsführung: Die Geschäftsführung legt klare Erwartungen für das Unternehmen fest und bewertet den Erfolg. Die Geschäftsführung ist Sponsor, Fürsprecher und treibende Kraft für die Übernahme bewährter Methoden und die Weiterentwicklung des Unternehmens
  - Festlegen von Erwartungen: Definieren und veröffentlichen Sie Ziele für Ihre Teams einschließlich der Art, wie diese Ziele gemessen werden.
  - Verfolgen der Zielerreichung: Überprüfen Sie regelmäßig die stufenweise Erreichung von Zielen und teilen Sie den entsprechenden Teams die Ergebnisse mit, damit geeignete Maßnahmen ergriffen werden können, wenn angepeilte Ergebnisse gefährdet sind.

- Bereitstellen der erforderlichen Ressourcen zum Erreichen Ihrer Ziele: Überprüfen Sie regelmäßig, ob die vorhandenen Ressourcen noch ausreichen oder ob aufgrund neuer Informationen, Änderungen an Zielen, Verantwortlichkeiten oder Ihrer Geschäftsumgebung zusätzliche Ressourcen benötigt werden.
- Unterstützen Ihrer Teams: Bleiben Sie mit Ihren Teams in Verbindung, damit Sie wissen, wie es ihnen ergeht und ob es äußere beeinträchtigende Faktoren gibt. 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. Treten Sie für Ihre Teams ein und beseitigen Sie Hindernisse und unnötige Bürden.
- Treibende Kraft für Übernahme bewährter Methoden: Würdigen Sie bewährte Methoden, die messbare Vorteile bieten, und geben Sie ihren Entwicklern und Anwendern Anerkennung. Ermutigen Sie Ihre Teams zur Annahme dieser Methoden, um die Vorteile noch zu verstärken.
- Treibende Kraft für die Entwicklung Ihrer Teams: Schaffen Sie eine Kultur der kontinuierlichen Verbesserung. Fördern Sie das Wachstum und die Entwicklung sowohl im Persönlichen als auch im Betrieblichen. Setzen Sie langfristige Ziele, die stufenweise Erfolge über einen längeren Zeitraum hinweg erfordern. Passen Sie diese Vision an Ihre Anforderungen, Geschäftsziele und Ihre Geschäftsumgebung an, wenn sie sich ändern.

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

Der/die Verantwortliche des Workload hat klare Anweisungen und Zuständigkeitsbereiche festgelegt, damit alle Teammitglieder direkt reagieren können, wenn die Ziele gefährdet sind. Es werden Eskalationsmechanismen verwendet, damit klare Anweisungen gelten, wenn Ereignisse außerhalb des festgelegten Zuständigkeitsbereichs liegen.

Vorteile der Einführung dieser bewährten Methode: Wenn Sie Änderungen frühzeitig testen und validieren, können Sie Probleme mit minimalen Kosten beheben und die Auswirkungen auf Ihre Kunden einschränken. Durch Tests vor der Bereitstellung minimieren Sie die Fehler.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Befugnis der Teammitglieder zu Maßnahmen bei Gefährdung der angepeilten Ergebnisse: Geben Sie Ihren Teammitgliedern die erforderlichen Berechtigungen, Hilfsmittel und Möglichkeiten, damit sie die benötigten Fertigkeiten für eine effektive Reaktion einüben können.

- Befähigen der Teammitglieder zum Einüben der erforderlichen Fertigkeiten für die Reaktion: Stellen Sie alternative sichere Umgebungen bereit, in denen Prozesse und Verfahren sicher getestet und eingeübt werden können. Führen Sie Ernstfallübungen durch, damit Ihre Teammitglieder Erfahrung beim Reagieren auf reale Vorfälle in simulierten und sicheren Umgebungen sammeln können.
- Definieren und Bestätigen der Befugnis von Teammitgliedern zum Ergreifen von Maßnahmen: Verschaffen Sie den Teammitgliedern die erforderliche Autorität, um Maßnahmen zu ergreifen, indem Sie ihnen Berechtigungen und Zugriff auf ihre Workloads und Komponenten geben. Sagen Sie ihnen deutlich, dass sie befugt sind, Maßnahmen zu ergreifen, wenn die Ziele gefährdet sind.

OPS03-BP03 Eskalation wird empfohlen

Teammitglieder verfügen über entsprechende Mechanismen und werden ermutigt, Bedenken an Entscheidungsträger und Beteiligte zu eskalieren, wenn ihnen Ziele als gefährdet erscheinen. Die Eskalation sollte früh und oft durchgeführt werden, damit Risiken identifiziert und Vorfälle verhindert werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Ermutigen zu einem frühen und häufigen Eskalieren: Bestätigen Sie im Unternehmen, dass die frühe und oftmalige Eskalation die bewährte Methode ist. Bestätigen und 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.
  - Bereitstellung eines Mechanismus für die Eskalation: Sorgen Sie für dokumentierte Verfahren, die definieren, wann und wie eine Eskalation erfolgen soll. Dokumentieren Sie eine Abfolge von Personen mit zunehmender Autorität zum Ergreifen oder Bestätigen von Maßnahmen und ihre Kontaktinformationen. Die Eskalation sollte so weit gehen, bis das Teammitglied der Meinung ist, dass das Problem an eine Person übergeben wurde, die damit umgehen kann, oder bis die Person kontaktiert wurde, die für das Risiko und den Betrieb des Workload verantwortlich ist. Letztendlich ist diese Person für alle Entscheidungen zu ihrem Workload verantwortlich. Eskalationen müssen die Art des Risikos, die Bedeutung des Workload, die betroffenen Personen, die Auswirkungen und die Dringlichkeit bzw. den voraussichtlichen Zeitpunkt der Auswirkungen enthalten.

- Schutz von eskalierenden Mitarbeitern: Stellen Sie eine Richtlinie bereit, die Teammitglieder vor Konsequenzen schützt, wenn sie zu einem nicht reagierenden Entscheidungsträger oder Verantwortlichen eskalieren. Schaffen Sie Mechanismen, durch die überprüft wird, ob dies geschieht, und leiten Sie entsprechende Maßnahmen ein.

OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar

Es gibt Mechanismen und diese werden angewandt, um Teammitglieder rechtzeitig über bekannte Risiken und geplante Ereignisse zu informieren. Erforderlicher Kontext, Details und Zeit (wenn möglich) werden bereitgestellt, um festzustellen, ob und welche Maßnahmen erforderlich sind, und um rechtzeitig Maßnahmen ergreifen zu können. Zum Beispiel die Benachrichtigung über Software-Schwachstellen, damit Patches beschleunigt werden können, oder die Benachrichtigung über geplante Verkaufsaktionen, damit ein Einfrieren von Änderungen implementiert werden kann, um das Risiko einer Service-Unterbrechung zu vermeiden. Geplante Ereignisse können in einem Änderungskalender oder Wartungsplan aufgezeichnet werden, so dass Teammitglieder feststellen können, welche Aktivitäten ausstehen.

Gewünschtes Ergebnis:

- Die Kommunikation sorgt für Kontext, Details und zeitliche Erwartungen.
- Die Teammitglieder haben eine klare Vorstellung davon, wann und wie sie in Reaktion auf Kommunikationen handeln müssen.
- Nutzen Sie Änderungskalender, um auf erwartete Änderungen aufmerksam zu machen.

Typische Anti-Muster:

- Mehrere Male pro Woche ereignen sich falsche Alarme. Sie stellen die Benachrichtigung jedes Mal auf stumm.
- Sie bitten Ihre Sicherheitsgruppen um eine Änderung, erhalten jedoch keine Information darüber, bis wann sie diese erwarten können.
- Sie erhalten immer wieder Chat-Benachrichtigungen, wenn Systeme hochskaliert werden, ohne dass eine Maßnahme erforderlich ist. Sie nutzen daraufhin den Chat-Kanal nicht mehr und verpassen eine wichtige Benachrichtigung.
- Es erfolgt eine Änderung im Produktionsbereich, ohne dass das Operations-Team darüber informiert wurde. Die Änderung löst einen Alarm aus und das On-Call-Team wird aktiviert.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Organisation vermeidet „Alarm-Ermüdung“.
- Teammitglieder können mit dem erforderlichen Kontext und angemessenen Erwartungen handeln.
- Änderungen können in Änderungszeitfenstern vorgenommen werden, was Risiken vermindert.

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. Identifizieren und entfernen Sie Alarme, die falsch positiv oder immer aktiv sind. Nutzen Sie Änderungskalender, damit die Teammitglieder wissen, wann sie Maßnahmen ergreifen können und welche Aktivitäten ausstehen. Prüfen Sie, ob die Kommunikation zu klaren Maßnahmen mit erforderlichem Kontext führt.

Kundenbeispiel

AnyCompany Retail verwendet Chat als wichtigstes Kommunikationsmedium. Alarme 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. Man verwendet einen Änderungskalender für die Planung größerer Änderungen an Produktionssystemen.

Implementierungsschritte

1. Analysieren Sie Ihre Alarme auf falsch positive Alarme oder solche, die ständig ausgelöst werden. Entfernen oder ändern Sie diese, so dass sie nur ausgelöst werden, wenn menschliche Interventionen erforderlich sind. Stellen Sie ein Runbook oder Playbook für ausgelöste Alarme bereit.
   a. Mit AWS Systems Manager Documents können Sie Runbooks oder Playbooks für Alarme erstellen.
2. 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 Alarme senden und auf Ereignisse reagieren.

3. 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.
4. 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 auf AWS benachrichtigt zu werden.

Ressourcen

Zugehörige bewährte Methoden:

- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren – Sorgen Sie bei bekannten Ergebnissen mit einem Runbook dafür, dass Kommunikationsinhalte in Handlungen umgesetzt werden können.
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen – Wenn das Ergebnis nicht bekannt ist, können Kommunikationsinhalte mithilfe von Playbooks in Handlungen umgesetzt werden.

Zugehörige Dokumente:

- AWS Security Bulletins (AWS-Sicherheitsberichte)
- Open CVE

Zugehörige Beispiele:

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

Zugehörige Services:

- AWS Chatbot

- AWS Systems Manager Change Calendar
- AWS Systems Manager Documents (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 in die Lage versetzt und ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern:

Teams müssen ihre Fertigkeiten ausbauen, um neue Technologien nutzen und mit veränderten Anforderungen und Aufgaben Ihrer Workloads umgehen zu können. Neue Fertigkeiten im Umgang mit neuen Technologien erhöhen oftmals die Zufriedenheit der Teammitglieder und ermöglichen neue Innovationen. Unterstützen Sie Ihre Teammitglieder beim Erlangen und Bewahren von Branchenzertifizierungen, mit denen ihre zunehmenden Fertigkeiten bestätigt und anerkannt werden. Führen Sie funktionsübergreifende Schulungen 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 stellt Ressourcen bereit, darunter das Erste Schritte – AWS Resource Center, AWS-Blogs, AWS Online Tech Talks, AWS-Veranstaltungen und -Webinaresowie die AWS Well-Architected Labs, die Anleitungen, Beispiele und detaillierte Walkthroughs zur Schulung Ihrer Teams bieten.

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

Sie sollten die von AWS bereitgestellten Schulungsressourcen nutzen, z. B. die Well-Architected Labs, den AWS Support (AWS Knowledge Center, AWS Diskussionsforenund AWS Support Center) bereitgestellten Ressourcen und AWS-Dokumentation nutzen, um Ihre Teams zu schulen. Wenn Sie

eine Frage zu AWS haben, können Sie sich über das AWS Support Center an den AWS Support wenden.

AWS Training und Zertifizierung bietet einige kostenlose 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 und -Fertigkeiten Ihres Teams auszubauen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Teammitglieder werden in die Lage versetzt und ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern: Zur Einführung neuer Technologien, um Innovationen und Änderungen bei Bedarf und Zuständigkeiten bei der Unterstützung Ihrer Workloads zu unterstützen, ist fortlaufende Bildung notwendig.
  - Bereitstellen von Ressourcen für die Weiterbildung: Stellen Sie eine spezielle strukturierte Lernzeit, Schulungsmaterialien und Laborressourcen bereit. Unterstützen Sie die Teilnahme an Konferenzen und bei professionellen Organisationen, die Möglichkeiten zum Lernen von Lehrenden und anderen Fachleuten bieten. Sorgen Sie dafür, dass erfahrene Teammitglieder neueren Teammitgliedern als Mentoren dienen können, oder dass sie sich Arbeitsweisen, Methoden und Fertigkeiten von ihnen abschauen können. Ermutigen Sie dazu, auch etwas über Inhalte zu lernen, die nicht direkt mit der Arbeit zusammenhängen, um den Horizont zu erweitern.
  - 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 ihre Fertigkeiten und bewährten Methoden austauschen können, wovon letztendlich das gesamte Unternehmen profitiert.
  - Unterstützen beim Erlangen und Bewahren von Branchenzertifizierungen: Unterstützen Sie Ihre Teammitglieder beim Erlangen und Bewahren von Branchenzertifizierungen, durch die das Gelernte bestätigt wird und die Erfolge anerkannt werden.

Ressourcen

Zugehörige Dokumente:

- Erste Schritte – AWS Resource Center
- AWS-Blogs

- AWS Cloud-Compliance
- AWS Diskussionsforen
- AWS-Dokumentation nutzen,
- AWS Online Tech Talks
- AWS-Veranstaltungen und -Webinare
- AWS Knowledge Center
- AWS Support
- AWS Training und Zertifizierung
- AWS Well-Architected Labs,
- Die Amazon Builders’ Library
- im offiziellen AWS-Podcast.

OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten

Legen Sie eine angemessene Teamgröße fest und stellen Sie die erforderlichen Hilfsmittel und Ressourcen für die Workloads bereit. Die Überlastung von Teammitgliedern erhöht das Risiko von Vorfällen durch menschliches Versagen. Investitionen in Tools und Ressourcen (z. B. Automatisierung für häufige Aufgaben) können die Effektivität Ihres Teams deutlich steigern, wodurch es sich ggf. um zusätzliche Aufgaben kümmern kann.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Angemessene Teamplanung: Stellen Sie sicher, dass Sie die Bedeutung und die maßgeblichen Faktoren des Erfolgs oder Misserfolgs Ihrer Teams kennen. Unterstützen Sie Teams mit erforderlichen Ressourcen.
  - Verstehen der Teamleistung: Messen Sie die Erreichung von Betriebsergebnissen und die Entwicklung von Assets durch Ihre Teams. Verfolgen Sie Änderungen bei dem Output und der Fehlerrate im Zeitverlauf. Sprechen Sie mit Teams, um sich über ihre arbeitsbezogenen Herausforderungen zu informieren (z. B. zunehmende Aufgaben, technologische Veränderungen, Verlust von Mitarbeitern oder steigende Kundenzahl).
  - Verstehen der Auswirkungen auf die Teamleistung: Bleiben Sie mit Ihren Teams in Verbindung, damit Sie wissen, wie es ihnen ergeht und ob es äußere beeinträchtigende Faktoren gibt. 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. Treten Sie für Ihre Teams ein und beseitigen Sie Hindernisse und unnötige Bürden.

- Bereitstellen der erforderlichen Ressourcen für den Erfolg von Teams: Überprüfen Sie regelmäßig, ob die vorhandenen Ressourcen noch ausreichen oder zusätzliche Ressourcen benötigt werden, und unterstützen Sie die Teams durch entsprechende Korrekturen.

OPS03-BP08 Unterschiedliche Meinungen werden innerhalb des Teams und teamübergreifend gefördert und sind erwünscht

Nutzen Sie die funktionsübergreifende Diversität, um verschiedene einzigartige Perspektiven zu erhalten. Nutzen Sie diese Perspektive, um Innovation zu fördern, Ihre Annahmen in Frage zu stellen und das Risiko einer Verzerrung durch automatische Bestätigung zu reduzieren. Erweitern Sie Inklusion, Diversität und Offenheit innerhalb Ihrer Teams, um nützliche Perspektiven zu gewinnen.

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.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

- Berücksichtigen unterschiedlicher Meinungen und Perspektiven: Ermutigen Sie alle anderen, einen Beitrag zu leisten. Geben Sie unterrepräsentierten Gruppen eine Stimme. Rotieren Sie die Rollen und Zuständigkeiten in Meetings.
  - Erweitern von Rollen und Zuständigkeiten: Bieten Sie Teammitgliedern die Möglichkeit, Rollen zu übernehmen, die ihnen fremd sind. Sie sammeln Erfahrung und erhalten neue Perspektiven durch die Rolle und den resultierenden Austausch mit neuen Teammitgliedern, zu denen sie möglicherweise andernfalls keinen Kontakt hätten. Sie werden die neue Rolle und die Teammitglieder mit ihren Erfahrungen und Perspektiven bereichern. Aus der erweiterten Perspektive können sich neue Geschäftschancen oder neue Verbesserungsmöglichkeiten ergeben. Lassen Sie Mitglieder innerhalb eines Teams abwechselnd allgemeine Aufgaben übernehmen, die normalerweise andere ausführen, um ihre Anforderungen und Auswirkungen zu verstehen.
- Bereitstellen einer sicheren und freundlichen Umgebung: Stellen Sie Richtlinien und Kontrollen zum Schutz der geistigen und physischen Sicherheit der Teammitglieder in Ihrem Unternehmen bereit. Die Teammitglieder müssen ohne Angst vor Vergeltung zusammenarbeiten können.

Wenn sich Teammitglieder sicher und willkommen fühlen, ist die Wahrscheinlichkeit höher, dass sie engagiert und produktiv bleiben. Je vielfältiger Ihr Unternehmen ist, desto besser können Sie andere verstehen, 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 Zugänglichkeit, unerschlossene Marktsegmente, unbehandelte Risiken in Ihrer Umgebung).

- Ermöglichen der vollständigen Teilnahme von Teammitgliedern: Stellen Sie die Ressourcen bereit, die Ihre Mitarbeiter zur vollständigen Teilnahme an allen arbeitsbezogenen Tätigkeiten benötigen. Teammitglieder haben Fertigkeiten entwickelt, mit denen sie ihre täglichen Herausforderungen meistern. Diese einzigartigen Fertigkeiten können Ihrem Unternehmen einen erheblichen Vorteil bieten. Wenn Sie die Teammitglieder mit den notwendigen Ressourcen ausstatten, werden die Vorteile ihres Beitrags verstärkt.

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.

Zur Vorbereitung auf operative Exzellenz müssen Sie die folgenden Punkte berücksichtigen:

Themen

Implementieren von Beobachtbarkeit

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

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.

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 Leistungskennzahlen (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 bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um Anwendungstelemetrie für Ihren Workload zu implementieren, verwenden Sie AWS-Services wie Amazon CloudWatch und AWS X-Ray. Amazon CloudWatch bietet Ihnen eine umfassende Suite aus Überwachungstools, mit denen Sie Ihre Ressourcen und Anwendungen in AWS und On-Premises überwachen 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, sodass Sie ein tiefes Verständnis des Verhaltens Ihrer Workloads erhalten. Mit Funktionen wie Service-Maps, Latenzverteilungen und Trace-Zeitplänen liefert 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. Stellen Sie den CloudWatch Agent bereit: 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. Definieren und überwachen Sie Geschäfts-KPIs: Richten Sie benutzerdefinierte Metriken ein, die mit Ihren Geschäftsziele übereinstimmen.
4. Instrumentieren Sie Ihre Anwendung mit AWS X-Ray: Neben der Bereitstellung des CloudWatch Agent 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.
5. Standardisieren Sie die Datenerfassung in Ihrer Anwendung: Standardisieren Sie die Datenerfassungspraktiken in Ihrer gesamten Anwendung. Einheitlichkeit hilft bei der Korrelation und Analyse von Daten und liefert einen umfassende Überblick über das Verhalten Ihrer Anwendung.
6. Analysieren Sie die Daten und setzen Sie Erkenntnisse um: Sobald die Datenerfassung und Normalisierung abgeschlossen sind, verwenden Sie Amazon CloudWatch für Metriken- und Protokollanalysen und AWS X-Ray für die Trace-Analyse. Eine solche Analyse kann wichtige Erkenntnisse über den Zustand, die Leistung und das Verhalten Ihrer Arbeitslast liefern und so Ihren Entscheidungsprozess beeinflussen.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung

- OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie
- OPS04-BP05 Implementieren der verteilten Nachverfolgung

Zugehörige Dokumente:

- AWS Observability Best Practices (Bewährte Methoden zur Beobachtbarkeit für AWS)
- CloudWatch-Benutzerhandbuch
- AWS X-Ray-Entwicklerhandbuch
- Instrumentieren verteilter Systeme für Einblicke in die Betriebsabläufe
- AWS Observability Skill Builder Course (Skill-Builder-Kurs zur Beobachtbarkeit in AWS)
- Neuerungen bei Amazon CloudWatch
- Neuerungen bei AWS X-Ray

Zugehörige Videos:

- AWS re:Invent 2022 – Observability best practices at Amazon (AWS re:Invent 2022 – Bewährte Überwachungsmethoden bei Amazon)
- AWS re:Invent 2022 – Developing an observability strategy (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. Indem Sie Ihre Anwendung so instrumentieren, dass sie Metriken, Protokolle und Traces zu diesen Abhängigkeiten ausgibt, gewinnen Sie ein besseres Verständnis potenzieller Engpässe, Leistungsprobleme oder Ausfälle, 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: sicherstellen, 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 bei fehlender Befolgung dieser bewährten Methode: 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 Benachrichtigungen 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. Externe Abhängigkeiten identifizieren: 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. Eine Überwachungsstrategie entwickeln: 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 Benachrichtigungen ein, die Sie über Statusänderungen oder Leistungsabweichungen informieren.
3. Den Amazon CloudWatch Internet Monitor nutzen: Er liefert Erkenntnisse zum globalen Internet und hilft Ihnen, Ausfälle oder Störungen zu verstehen, die sich auf Ihre externen Abhängigkeiten auswirken könnten.
4. Informiert bleiben mit dem AWS Health Dashboard: Dieses Dashboard stellt Alarme bereit und empfiehlt Abhilfemaßnahmen, wenn in AWS Ereignisse eintreten, die möglicherweise Ihre Services betreffen.
5. Ihre Anwendung instrumentieren mit AWS X-Ray: AWS X-Ray bietet Ihnen Erkenntnisse zur 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. Den Amazon DevOps Guru einsetzen: Dieser Machine Learning-gestützte Service identifiziert operative Probleme, prognostiziert das Auftreten kritischer Probleme und empfiehlt spezifische Maßnahmen. So 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. Regelmäßig überwachen: 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. Nach Änderungen validieren: 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 Ermitteln wichtiger Leistungskennzahlen
- OPS04-BP02 Implementieren einer Anwendungstelemetrie
- OPS04-BP03 Implementieren von Telemetrie für Benutzererfahrung
- OPS04-BP05 Implementieren der verteilten Nachverfolgung

Zugehörige Dokumente:

- Was ist AWS Health?
- Verwendung von Amazon CloudWatch Internet Monitor
- AWS X-Ray-Entwicklerhandbuch
- Amazon DevOps Guru-Benutzerhandbuch

Zugehörige Videos:

- Visibility into how internet issues impact app performance (Wie sich Internetprobleme auf die Leistung von Apps auswirken)
- Introduction to Amazon DevOps Guru (Einführung in Amazon DevOps Guru)

Zugehörige Beispiele:

- Gaining operational insights with AIOps using Amazon DevOps Guru (Operative Erkenntnisse gewinnen mit AIOps und Amazon DevOps Guru)
- AWS Health Aware

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

Design für den Betrieb

Verwenden Sie Strategien, die die Übertragung von Änderungen auf die Produktionsumgebung verbessern und Faktorwechsel, schnelles Feedback zur Qualität sowie eine schnelle Fehlerbehebung ermöglichen. 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.

In AWS können Sie sämtliche Workloads (Anwendungen, Infrastruktur, Richtlinien, Governance und Betrieb) als Code einsehen. Alles kann in Code definiert und mittels Code aktualisiert werden. Das bedeutet, dass Sie bei jedem Element Ihres Stacks die gleiche technischen Vorgehensweise wie bei Anwendungscode anwenden können.

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 Feedbackschleifen 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. Ihr Unternehmen 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 bei der Implementierung von Software 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 z. B. automatische Aktualisierungen der Skalierungsrichtlinien, werden im Voraus getestet, um den Anforderungen des Datenverkehrs gerecht zu werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: 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. Ihr Unternehmen hat einen Teststandard, den alle Änderungen erfüllen müssen.

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?
   a. Die Referenzarchitektur für AWS-Bereitstellungs-Pipelines 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. Amazon CodeGuru Reviewer kann Ihren Anwendungscode auf Fehler prüfen.
   b. Nutzen Sie AWS CodeBuild, um Tests auf Software-Artefakten durchzuführen.
   c. AWS CodePipeline kann Ihre Softwaretests in eine Pipeline orchestrieren.

Ressourcen

Zugehörige bewährte Methoden:

- OPS05-BP01 Verwendung einer Versionskontrolle
- OPS05-BP06 Gemeinsame Design-Standards
- OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung

Zugehörige Dokumente:

- Adopt a test-driven development approach (Einführung eines testgesteuerten Entwicklungsansatzes)
- Automated AWS CloudFormation Testing Pipeline with TaskCat and CodePipeline (Automatisierte CloudFormation-Testpipeline mit TaskCat und CodePipeline)
- Building end-to-end AWS DevSecOps CI/CD pipeline with open source SCA, SAST, and DAST tools (Erstellen einer End-to-End-AWS DevSecOps-CI/CD-Pipeline mit Open-Source-SCA-, -SAST- und -DAST-Tools)

- Getting started with testing serverless applications (Erste Schritte beim Testen von Serverless-Anwendungen)
- My CI/CD pipeline is my release captain (Meine CI/CD-Pipeline ist mein Release Captain)
- Practicing Continuous Integration and Continuous Delivery on AWS Whitepaper (Durchführung von Continuous Integration und Continuous Delivery in AWS – Whitepaper)

Zugehörige Videos:

- AWS re:Invent 2020: Testable infrastructure: Integration testing on AWS (AWS re:Invent 2020: Testbare Infrastruktur: Integrationstests auf AWS)
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development (AWS Summit ANZ 2021 – Vorantreiben einer „Test-First“-Strategie mit CDK und testgesteuerter Entwicklung)
- Testing Your Infrastructure as Code with AWS CDK (Testen Ihrer Infrastruktur als Code mit AWS CDK)

Zugehörige Ressourcen:

- AWS Deployment Pipeline Reference Architecture - Application (Referenzarchitektur für AWS-Bereitstellungs-Pipelines – Anwendung)
- AWS Kubernetes DevSecOps Pipeline
- Policy as Code Workshop – Test Driven Development (Richtlinie als Code – Workshop – testgesteuerte Entwicklung)
- Run unit tests for a Node.js application from GitHub by using AWS CodeBuild (Tests von Einheiten für eine Node.js-Anwendung aus GitHub mithilfe von AWS CodeBuild ausführen)
- Use Serverspec for test-driven development of infrastructure code (Serverspec für die testgesteuerte Entwicklung von Infrastrukturcode verwenden)

Zugehörige Services:

- 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.

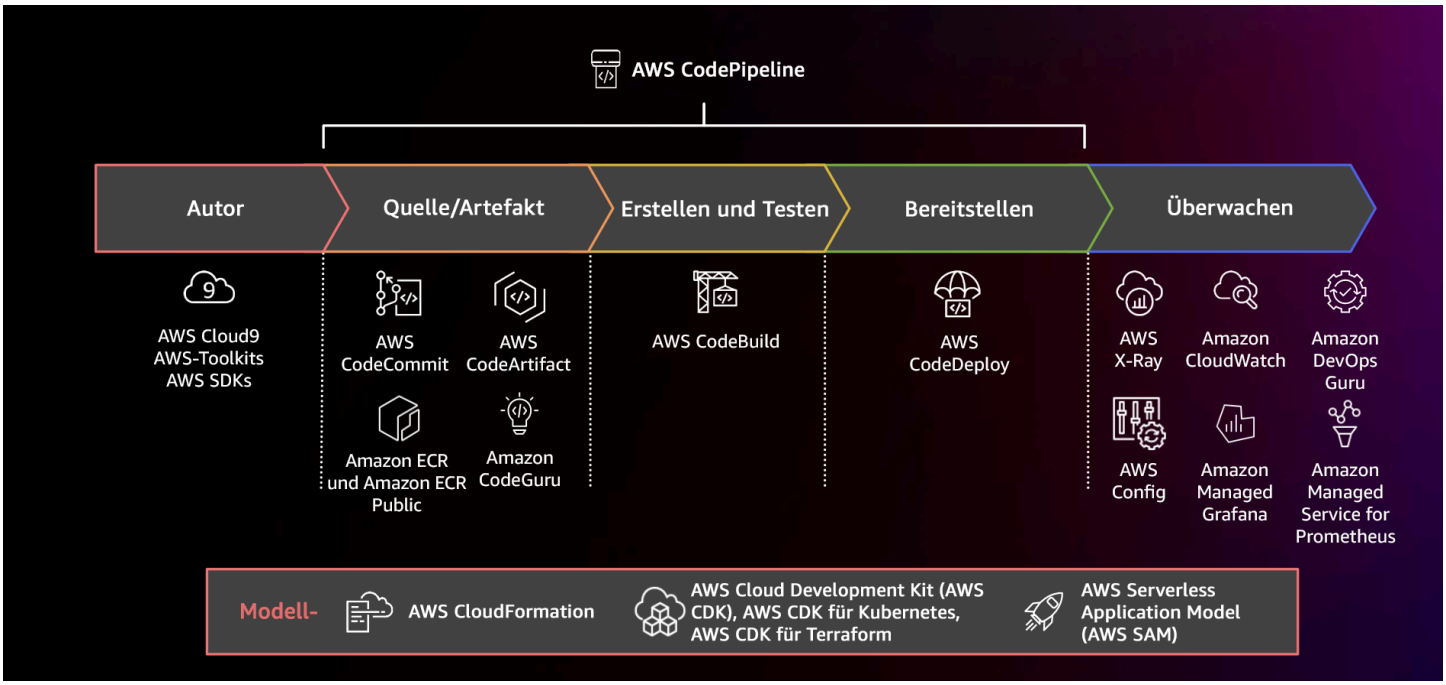
Vorteile der Nutzung dieser bewährten Methode: Mithilfe von Mechanismen zur Verwaltung von Erstellungs- und Bereitstellungsaktivitäten reduzieren Sie den Aufwand für wiederholte Aufgaben, verschaffen Ihren Teammitgliedern die Zeit, sich auf ihre wichtigen Aufgaben zu konzentrieren, und begrenzen die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

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 Umsetzung von Methoden zur Verbesserung der Codequalität können Sie die Anzahl der Probleme minimieren, die bei der Produktion noch vorhanden sind. Die Codequalität wird durch bewährte Methoden wie Pair-Programming und Code-Reviews verbessert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: 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.

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. Amazon CodeGuru Reviewer kann Machine-Learning-Programmierempfehlungen für Java- und Python-Code bereitstellen.
   b. Sie können mit AWS Cloud9 gemeinsame Entwicklungsumgebungen erstellen und Code in Teamarbeit entwickeln.

Aufwand für den Implementierungsplan: 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-BP06 Gemeinsame Design-Standards - Sie können Designstandards als Teil Ihrer Codequalitätsverfahren gemeinsam nutzen.

Zugehörige Dokumente:

- Agile Software Guide (Leitfaden für agile Software)
- My CI/CD pipeline is my release captain (Meine CI/CD-Pipeline ist mein Release Captain)
- Automatisieren von Codeüberprüfungen mit Amazon CodeGuru Reviewer
- Adopt a test-driven development approach (Einführung eines testgesteuerten Entwicklungsansatzes)
- How DevFactory builds better applications with Amazon CodeGuru (So entwickelt DevFactory mit Amazon CodeGuru bessere Anwendungen)
- On Pair Programming (Über Pair-Programming)
- RENGA Inc. automates code reviews with Amazon CodeGuru (RENGA Inc. automatisiert Code-Reviews mit Amazon CodeGuru)

- The Art of Agile Development: Test-Driven Development (Die Kunst der agilen Entwicklung: Testbasierte Entwicklung)
- Why code reviews matter (and actually save time!) (Warum Code-Reviews wichtig sind (und tatsächlich Zeit sparen!))

Zugehörige Videos:

- AWS re:Invent 2020: Continuous improvement of code quality with Amazon CodeGuru (AWS re:Invent 2020: Kontinuierliche Verbesserung der Codequalität mit Amazon CodeGuru)
- AWS Summit ANZ 2021 - Driving a test-first strategy with CDK and test driven development (AWS Summit ANZ 2021 – Vorantreiben einer „Test-First“-Strategie mit CDK und testgesteuerter Entwicklung)

Zugehörige Services:

- 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)

Bereitstellungsrisiken abschwächen

Verwenden Sie Ansätze, die ein schnelles Feedback zur Qualität liefern und eine umgehende Wiederherstellung des vorherigen Zustands nach 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.

Das Design Ihres Workloads sollte beinhalten, wie es bereitgestellt, aktualisiert und betrieben werden soll. Sie werden technische Methoden implementieren möchten, die auf die Reduzierung von Mängeln sowie auf schnelle und sichere Fehlerbehebungen ausgerichtet sind.

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, Funktionen 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 Best Practice: Mittel

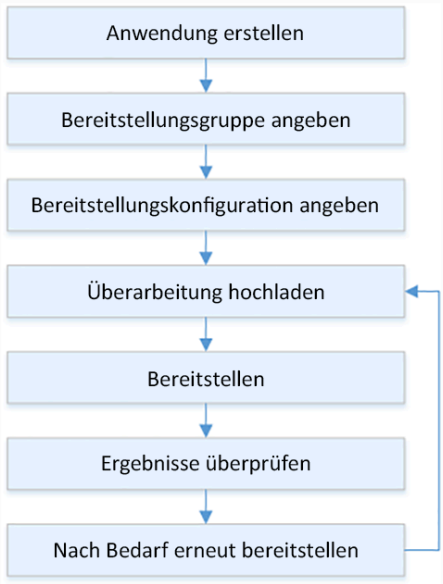
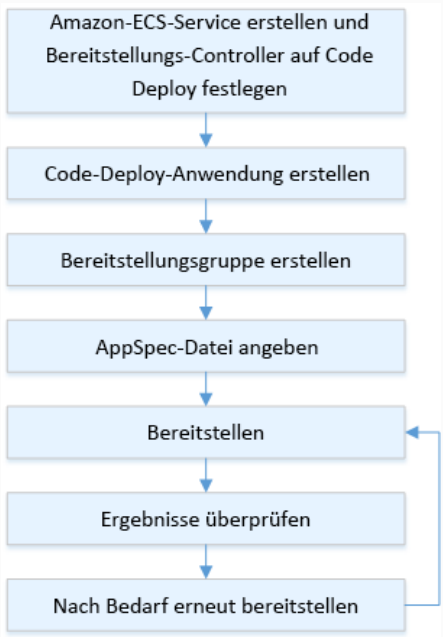
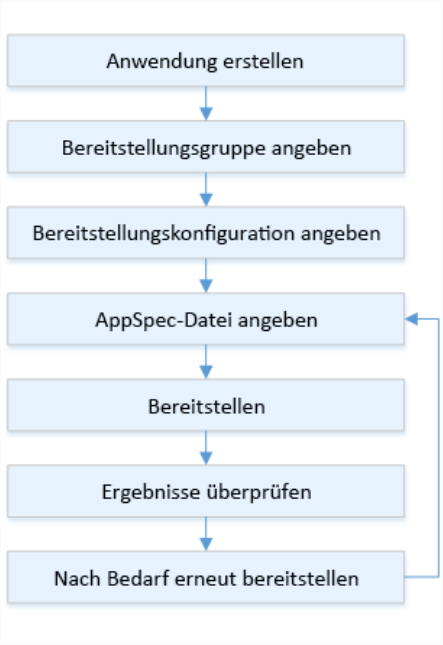
Implementierungsleitfaden

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 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 | 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)

Zugehörige Beispiele:

- Testen einer Blau/Grün-Bereitstellung in AWS CodeDeploy
- Workshop | Erstellen von CI/CD-Pipelines für Lambda-Canary-Bereitstellungen mit AWS CDK
- Workshop | Blue/Green and Canary Deployment for EKS and ECS (Workshop | Blau/Grün- und Canary-Bereitstellungen für EKS und ECS)
- Workshop | Building a Cross-account CI/CD Pipeline (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 (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)

Operative Bereitschaft und Änderungsverwaltung

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. Verwalten Sie den Änderungsfluss in Ihre Umgebungen.

Sie sollten einen konsistenten Prozess (inklusive manueller und automatisierter Checklisten) anwenden, 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 werden Sie in Runbooks notieren und Playbooks werden Ihnen bei der Lösung von Problemen helfen. Verwenden Sie einen Mechanismus zur Verwaltung von Änderungen, der die Erzielung eines geschäftlichen Nutzens unterstützt und dazu beiträgt, mit den Änderungen verbundene Risiken zu mindern.

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.

AWS 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

A Runbooks 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 Schritt-für-Schritt-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 sind an einer zentralen Stelle gespeichert und werden häufig aktualisiert.

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 bei fehlender Befolgung dieser Best Practice: 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 Entwicklung Ihrer Organisation automatisiert werden. Mit Services wie AWS Systems Manager Automationkönnen Sie Textdateien zu

Automatisierungen transformieren, die Sie für Ihren Workload ausführen können. Diese Automatisierungen können als Reaktion auf Ereignisse ausgeführt werden, was den operativen Aufwand für die Wartung des Workloads reduziert.

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-Titel ## Runbook-Informationen | Runbook-ID | Beschreibung | Verwendete Tools
 | Spezielle Berechtigungen | Runbook-Autor | Letzte Aktualisierung | Eskalations-POC |
 |-------|-------|-------|-------|-------|-------|-------| | RUN001 | Wofür ist dieses
 Runbook bestimmt? Was ist das gewünschte Ergebnis? | Tools | Berechtigungen| Ihr Name
 | 2022-09-21 | Eskalationsname | ## Schritte 1. Schritt eins 2. Schritt zwei
```

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. Geben Sie den `Runbook-Titel` ein und füllen Sie die erforderlichen Felder unter `Runbook-Informationen`aus.
4. Füllen Sie beginnend mit 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 werden Sie eine Bibliothek von Runbooks aufbauen. Beginnen Sie mit der Automatisierung von Runbooks, wenn diese Bibliothek wächst.

Aufwand für den Implementierungsplan: Niedrig. Eine Schritt-für-Schritt-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: Es sollte eine verantwortliche Person für jedes Runbook geben, die das jeweilige Runbook verwaltet und aktualisiert.
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen: Runbooks und Playbooks sind sich zwar ähnlich, es gibt jedoch einen wichtigen Unterschied: Ein Runbook hat ein gewünschtes Ergebnis. Häufig werden Runbooks ausgelöst, wenn ein Playbook die Ursache für ein Problem identifiziert hat.
- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen: Runbooks sind Bestandteil guter Verfahren für die Verwaltung von Ereignissen, Vorfällen und Problemen.
- OPS10-BP02 Implementieren eines Prozesses für jeden Alarm: Runbooks und Playbooks sollten verwendet werden, um auf Warnungen zu reagieren. Mit der Zeit sollten diese Reaktionen automatisiert werden.
- OPS11-BP04 Wissensmanagement: Die Verwaltung und Aktualisierung ist ein wesentlicher Bestandteil des Wissensmanagement.

Zugehörige Dokumente:

- Operative Kompetenz durch automatisierte Playbooks und Runbooks
- AWS Systems Manager: Mit Runbooks arbeiten

- 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 guide to runbooks, incident reports, and incident response (SEC318-R1)
- Automatisierung von IT-Abläufen in AWS | Amazon Web Services
- Integration von Skripts in AWS Systems Manager

Zugehörige Beispiele:

- AWS Systems Manager: Automation-Walkthroughs
- 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
- Well-Architected Labs: Automatisieren von Vorgängen mit Playbooks und Runbooks

Zugehörige Services:

- AWS Systems Manager Automation

OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen

Playbooks sind Schritt-für-Schritt-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 Beteiligten 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 Warnungen verwendet werden, sollte Ihr Team in den Warnungen 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 genutzt 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 Entwicklung Ihrer Organisation automatisiert werden. Mit Services wie AWS Systems Manager Automationskann einfacher Text in Automatisierungen umgewandelt werden. Diese Automatisierungen können dann für Ihren Workload ausgeführt werden, um die Untersuchungen zu beschleunigen. Sie können in 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 zur Reaktion auf Vorfälle verwenden. Dieser Service bietet eine einzige Oberfläche für die Untersuchung von Vorfällen, die Information der Beteiligten über Untersuchung und Abhilfemaßnahmen und die Zusammenarbeit während des gesamten Vorgangs. Er verwendet AWS Systems Manager Automations 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-Titel ## Playbook-Info | Playbook-ID | Beschreibung |
 Verwendete Tools | Besondere Berechtigungen | Playbook-Autor | Letzte
 Aktualisierung | Eskalation-POC | Beteiligte | Kommunikationsplan |
 |-------|-------|-------|-------|-------|-------|-------|-------|-------| | RUN001
 | Wofür ist dieses Playbook? Für welchen Vorfall wird es verwendet? | Tools |
 Berechtigungen | Ihr Name | 21.09.2022 | Eskalationsname | Name des Beteiligten | Wie
 werden während der Untersuchung Aktualisierungen mitgeteilt? | ## Schritte 1. Schritt
 eins 2. Schritt zwei
```

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 anhand der Markdown-Vorlage den Abschnitt `Name des Playbooks` und die Felder unter `Playbook-Info`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 Beteiligten 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: Es sollte eine verantwortliche Person für jedes Runbook geben, die das jeweilige Runbook verwaltet und aktualisiert.
- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren: Runbooks und Playbooks sind sich zwar ähnlich, es gibt jedoch einen wichtigen Unterschied: Ein Runbook hat ein gewünschtes Ergebnis. Häufig werden Runbooks verwendet, wenn ein Playbook die Ursache für ein Problem identifiziert hat.
- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen: Runbooks sind Bestandteil guter Verfahren für die Verwaltung von Ereignissen, Vorfällen und Problemen.
- OPS10-BP02 Implementieren eines Prozesses für jeden Alarm: Runbooks und Playbooks sollten verwendet werden, um auf Warnungen zu reagieren. Mit der Zeit sollten diese Reaktionen automatisiert werden.
- OPS11-BP04 Wissensmanagement: Die Verwaltung und Aktualisierung ist ein wesentlicher Bestandteil des Wissensmanagements.

Zugehörige Dokumente:

- Operative Kompetenz durch automatisierte Playbooks und Runbooks
- AWS Systems Manager: Mit Runbooks arbeiten
- Verwendung von AWS Systems Manager-Automation-Runbooks zur Lösung operativer Aufgaben

Zugehörige Videos:

- AWS re:Invent 2019: DIY guide to runbooks, incident reports, and incident response (SEC318-R1) (AWS re:Invent 2019: DIY-Leitfaden für Runbooks, Vorfallberichte und Vorfallreaktion (SEC318-R1))
- AWS Systems Manager Incident Manager - AWS Virtual Workshops (AWS Systems Manager Incident Manager – virtuelle AWS-Workshops)
- Integrate Scripts into AWS Systems Manager (Integration von Skripts in AWS Systems Manager)

Zugehörige Beispiele:

- AWS Customer Playbook Framework
- AWS Systems Manager: Walkthroughs 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

Erfolg bedeutet, dass die gewünschten Ergebnisse erreicht werden. Gemessen wird der Erfolg über Metriken, die Sie definieren. Durch das Verständnis des Zustands Ihres Workloads und Ihrer Betriebsabläufe können Sie feststellen, wann organisatorische und betriebliche Ergebnisse gefährdet werden oder gefährdet sind und entsprechend reagieren.

Um erfolgreich zu sein, müssen Sie folgende Voraussetzungen erfüllen:

Themen

Nutzung der Workload-Beobachtbarkeit

Sorgen Sie für einen optimalen Zustand des Workloads, indem Sie 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.

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.

AWS bietet spezielle Tools wie Amazon CloudWatch zur Überwachung und Protokollierung und AWS X-Ray zur verteilten Nachverfolgung. Diese Services lassen sich mühelos in verschiedene AWS-Ressourcen integrieren und ermöglichen eine effiziente Datenerfassung, die Einrichtung von

Warnmeldungen auf der Grundlage vordefinierter Schwellenwerte und die Darstellung von Daten auf Dashboards zur einfachen Interpretation. Mithilfe dieser Erkenntnisse können Sie fundierte, datengestützte Entscheidungen treffen, die Ihren betrieblichen Zielen entsprechen.

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 Dashboards erstellen

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.
- Sich ausschließlich auf die manuelle Überprüfung von Protokollen verlassen, 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 bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Amazon CloudWatch Logs ist ein leistungsstarkes Tool für die Protokollanalyse. Integrierte Funktionen wie CloudWatch Logs Insights und Contributor Insights machen das Ableiten aussagekräftiger Informationen aus Protokollen intuitiv und effizient.

Implementierungsschritte

1. CloudWatch Logs einrichten: Konfigurieren Sie Anwendungen und Services so, dass Protokolle an CloudWatch Logs gesendet werden.
2. CloudWatch Logs Insights einrichten: 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.
3. Erkenntnisse von Mitwirkenden nutzen: Verwenden Sie CloudWatch Contributor Insights, um Top-Talker in Dimensionen mit hoher Kardinalität wie IP-Adressen oder Benutzeragenten zu identifizieren.
4. CloudWatch Logs-Metrikfilter implementieren: Konfigurieren Sie metrische CloudWatch-Protokollfilter um Protokolldaten in umsetzbare Metriken zu konvertieren. Auf diese Weise können Sie Alarme einstellen oder Muster näher analysieren.
5. 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:

- Analyzing Log Data with CloudWatch Logs Insights (Analysieren von Protokolldaten mit CloudWatch Logs Insights)
- Using CloudWatch Contributor Insights (Nutzung von CloudWatch Contributor Insights)
- Creating and Managing CloudWatch Logs Log Metric Filters (Erstellen und Verwalten von CloudWatch Logs-Metrikfiltern)

Zugehörige Videos:

- Analyze Log Data with CloudWatch Logs Insights (Analysieren von Protokolldaten mit CloudWatch Logs Insights)
- Use CloudWatch Contributor Insights to Analyze High-Cardinality Data (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 die Benutzererfahrung verbessert werden.

Gewünschtes Ergebnis: Sie verschaffen sich einen klaren Überblick über die verteilten Abläufe Ihrer Anwendung und erzielen dadurch eine schnellere Problemlösung und eine verbesserte Benutzererfahrung.

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 bei fehlender Befolgung dieser bewährten Methode: 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. Funktionen 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. Integrieren Sie AWS X-Ray: Stellen Sie sicher, dass X-Ray in Ihre Anwendungen integriert ist, um Trace-Daten zu erfassen.
2. Analysieren Sie X-Ray-Metriken: Untersuchen Sie anhand von X-Ray-Traces abgeleitete Metriken wie Latenz, Anfrageraten, Fehlerraten und Antwortzeitverteilungen mithilfe der Service-Karte, um den Status der Anwendung zu überwachen.
3. Verwenden Sie ServiceLens: Nutzen Sie die ServiceLens-Karte 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 die X-Ray Insights zur automatisierten 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. Verwenden Sie X-Ray Analytics: X-Ray Analytics ermöglicht es Ihnen, Daten gründlich zu untersuchen, Muster zu lokalisieren und Erkenntnisse zu gewinnen.
6. Verwenden Sie 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. Nutzen Sie Amazon DevOps Guru: Setzen Sie Amazon DevOps Guru ein, um von Machine Learning-Modellen zu profitieren, die betriebliche Anomalien in Traces lokalisieren.

8. Verwenden Sie CloudWatch Synthetics: Nutzen Sie CloudWatch Synthetics, um Canaries für die kontinuierliche Überwachung Ihrer Endgeräte und Workflows zu erstellen. Sie können diese Canaries in X-Ray integrieren, um Trace-Daten für eine eingehende Analyse der getesteten Anwendungen bereitzustellen.
9. Verwenden Sie Real User Monitoring (RUM): Mit AWS X-Ray und CloudWatch RUMkönnen Sie den Anforderungspfad ausgehend von den Endbenutzern Ihrer Anwendung über nachgelagerte AWS Managed Services analysieren und debuggen. Auf diese Weise können Sie Latenztrends und Fehler identifizieren, die sich auf Ihre Benutzer auswirken.
10. Korrelieren Sie Daten mit Protokollen: Bringen Sie Trace-Daten mit zugehörigen Protokollen innerhalb der X-Ray Trace-Ansicht in Zusammenhang, um eine detaillierte Perspektive auf das Anwendungsverhalten zu erhalten. Auf diese Weise können Sie Protokollereignisse anzeigen, die direkt mit verfolgten Transaktionen verknüpft sind.

Aufwand für den Implementierungsplan: Mittel.

Ressourcen

Zugehörige bewährte Methoden:

- OPS08-BP01 Analysieren von Workload-Metriken
- OPS08-BP02 Analysieren von Workload-Protokollen

Zugehörige Dokumente:

- Using ServiceLens to Monitor Application Health (Verwenden von ServiceLens zur Überwachung des Zustands Ihrer Anwendungen)
- Exploring Trace Data with X-Ray Analytics (Erkunden von Trace-Daten mit X-Ray Analytics)
- Detecting Anomalies in Traces with X-Ray Insights (Mit X-Ray Insights Anomalien in Traces erkennen)
- Continuous Monitoring with CloudWatch Synthetics (Fortlaufende Überwachung mit CloudWatch Synthetics)

Zugehörige Videos:

- Analyze and Debug Applications Using Amazon CloudWatch Synthetics and AWS X-Ray (Analysieren und Debuggen von Anwendungen mithilfe von Amazon CloudWatch Synthetics und AWS X-Ray)
- Use AWS X-Ray Insights (Nutzung von AWS X-Ray-Insights)

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- Implementing X-Ray with AWS Lambda (Implementieren von X-Ray mit AWS Lambda)
- Vorlagen für CloudWatch Synthetics-Canaries

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 Benachrichtigungen, 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 der 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 Alarmermüdung 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 Alarmierungs- und Kommunikationstools.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: 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. Legen Sie die wichtigsten Leistungskennzahlen (KPIs) fest: Identifizieren Sie die KPIs Ihrer Anwendung. Warnmeldungen sollten mit diesen KPIs verknüpft werden, damit sie die Auswirkungen auf das Unternehmen genau widerspiegeln.
2. Implementieren Sie die Erkennung von Anomalien:
   - Verwenden Sie AWS Cost Anomaly Detection: Richten Sie AWS Cost Anomaly Detection ein, damit ungewöhnliche Muster automatisch erkannt werden und sichergestellt wird, dass Warnmeldungen nur bei echten Anomalien generiert werden.
   - Nutzen Sie 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, damit Sie bei erkannten Problemen gewarnt werden.
   - Verwenden Sie DevOps Guru:
     a. Nutzen Sie die Machine Learning-Fähigkeiten von Amazon DevOps Guru für die Erkennung betrieblicher Anomalien anhand vorhandener Daten.
     b. Navigieren Sie zu den Benachrichtigungseinstellungen in DevOps Guru, um Anomaliewarnungen einzurichten.
3. Implementieren Sie umsetzbare Warnmeldungen: Entwerfen Sie Warnmeldungen, die angemessene Informationen für sofortige Maßnahmen liefern.
4. Reduzieren Sie Alarmermüdung: Minimieren Sie die Zahl der Warnmeldungen, die nicht kritisch sind. Wenn Teams mit einer zu großen Zahl an unbedeutenden Warnmeldungen

überhäuft werden, kann dies dazu führen, dass sie kritische Probleme übersehen und der Warnmechanismus allgemein an Effektivität verliert.

5. Richten Sie zusammengesetzte Alarme ein: Verwenden Sie zusammengesetzte Amazon CloudWatch-Alarme, um mehrere Alarme zu konsolidieren.
6. Ermöglichen Sie Alarm-Tools: Integrieren Sie Tools wie Ops Genie und PagerDuty.
7. Nutzen Sie AWS Chatbot: Setzen Sie AWS Chatbotein, um Warnmeldungen an Chime, Microsoft Teams und Slack weiterzuleiten.
8. Stützen Sie Warnungen auf Protokollen: Verwenden Sie metrische Protokollfilter in CloudWatch, um Alarme auf der Grundlage bestimmter Protokollereignisse zu erstellen.
9. Überprüfen und wiederholen: Überprüfen und verfeinern Sie die Warnkonfigurationen regelmäßig.

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:

- Using Amazon CloudWatch Alarms (Verwenden von Amazon CloudWatch-Alarmen)
- Create a composite alarm (Erstellung eines zusammengesetzten Alarms)
- Create a CloudWatch alarm based on anomaly detection (Erstellung eines CloudWatch-Alarms auf der Grundlage der Anomalieerkennung)
- DevOps Guru Notifications (DevOps Guru-Benachrichtigungen)
- X-Ray Insights notifications (X-Ray Insights--Benachrichtigungen)

- Monitor, operate, and troubleshoot your AWS resources with interactive ChatOps (Überwachung, Betrieb und Fehlerbehebung Ihrer AWS-Ressourcen mit interaktiven ChatOps)
- Amazon CloudWatch-Integrationsleitfaden | PagerDuty
- Integrate OpsGenie with Amazon CloudWatch (Integration von OpsGenie in Amazon CloudWatch)

Zugehörige Videos:

- Create Composite Alarms in Amazon CloudWatch (Erstellung zusammengesetzter Alarme in Amazon CloudWatch)
- AWS Chatbot Overview (AWS Chatbot-Übersicht)
- AWS on Air ft. Mutative Commands in AWS Chatbot (AWS on Air mit veränderlichen Befehlen in AWS Chatbot)

Zugehörige Beispiele:

- Alarme, Vorfallmanagement und Problembehebung in der Cloud mit Amazon CloudWatch
- Tutorial: Creating an Amazon EventBridge rule that sends notifications to AWS Chatbot (Erstellen einer Amazon EventBridge-Regel, die Benachrichtigungen an AWS Chatbot sendet)
- Workshop zur Beobachtbarkeit

OPS08-BP05 Dashboards erstellen

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 der Nutzung 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 bei fehlender Befolgung dieser bewährten Methode: 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. Einfaches Dashboard erstellen: Erstellen Sie ein neues Dashboard in CloudWatchund geben Sie ihm einen aussagekräftigen Namen.
2. Markdown-Widgets verwenden: Bevor Sie sich mit Metriken befassen, sollten Sie Markdown-Widgets nutzen, um Ihr Dashboard oben mit Kontext zu versehen. 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. Dashboard-Variablen erstellen: Integrieren Sie gegebenenfalls Dashboard-Variablen, um dynamische und flexible Ansichten zu ermöglichen.
4. Metrik-Widgets erstellen: 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. Log Insights-Abfragen verwenden: Nutzen Sie CloudWatch Logs Insights, um umsetzbare Metriken aus Ihren Protokollen abzurufen und diese Erkenntnisse auf Ihrem Dashboard anzuzeigen.
6. Alarme einrichten: Integrieren Sie CloudWatch-Alarme in Ihr Dashboard, um einen raschen Überblick über alle Metriken zu erhalten, die ihre Schwellenwerte überschreiten.
7. Contributor Insights verwenden: Integrieren Sie CloudWatch Contributor Insights, um Felder mit hoher Kardinalität zu analysieren und ein besseres Verständnis der wichtigsten Mitwirkenden Ihrer Ressource zu erhalten.
8. Benutzerdefinierte Widgets entwerfen: Für spezielle Anforderungen, die von Standard-Widgets nicht erfüllt werden, sollten Sie es in Betracht ziehen, benutzerdefinierte Widgets zu erstellen. Diese können Daten aus verschiedenen Quellen abrufen oder sie auf einzigartige Weise darstellen.
9. Wiederholen und verfeinern: 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:

- Building Dashboards for Operational Visibility (Erstellung von Dashboards für operative Sichtbarkeit)
- Using Amazon CloudWatch Dashboards (Verwenden von Amazon CloudWatch-Dashboards)

Zugehörige Videos:

- Create Cross Account & Cross Region CloudWatch Dashboards (Konto- und regionenübergreifende CloudWatch-Dashboards erstellen)
- AWS re:Invent 2021 - Gain enterprise visibility with AWS Cloud operation dashboards (AWS re:Invent 2021: Mehr Unternehmenstransparenz mit geschäftsorientierten AWS Cloud-Dashboards)

Zugehörige Beispiele:

- Workshop zur Beobachtbarkeit
- Anwendungsüberwachung mit Amazon CloudWatch

Grundlegendes zum betrieblichen Status

Definieren, erfassen und analysieren Sie Metriken für Operationen, um einen Einblick in die Aktivitäten Ihrer Operations-Teams zu erhalten. Dies ist wichtig, damit Sie bei Bedarf entsprechende Maßnahmen ergreifen können.

Ihre Organisation sollte in der Lage sein, den operativen Zustand problemlos in Erfahrung zu bringen. Sie sollten die Geschäftsziele Ihrer Operations-Teams definieren, wichtige Leistungsindikatoren identifizieren, die diese widerspiegeln, und dann Metriken auf der Grundlage der Betriebsergebnisse entwickeln, um nützliche Erkenntnisse zu gewinnen. Aus diesen Metriken sollten Sie Dashboards und Berichte erstellen, die Aufschluss über geschäftliche und technische Aspekte geben, damit Führungskräfte und Stakeholder gut fundierte Entscheidungen treffen können.

AWS macht es einfacher, Ihre Betriebsprotokolle zusammenzuführen und zu analysieren, sodass Sie Metriken generieren, den Status Ihrer betrieblichen Abläufe kennen und Einblicke in die Abläufe im Laufe der Zeit gewinnen 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:

- Amazon 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)

Reagieren auf Ereignisse

Sie sollten für betriebliche Ereignisse Vorsorge tragen. Das gilt sowohl für geplante Ereignisse (z. B. Verkaufsaktionen, Bereitstellungen oder Fehlertests) als auch für ungeplante Ereignisse (z. B. Auslastungsspitzen oder Ausfälle von Komponenten). Beim Reagieren auf Alarme sollten Sie Ihre Runbooks und Playbooks zu Rate ziehen, um konsistente Resultate zu erbringen. Für definierte Alarme sollte eine Rolle oder ein Team als Besitzer festgelegt sein, das für die Reaktion und Eskalation verantwortlich ist. Sie werden auch wissen möchten, welche geschäftlichen Auswirkungen Systemkomponenten haben, um bei Bedarf zielgerichtete Maßnahmen einleiten zu können. Nach Ereignissen sollten Sie eine Ursachenanalyse durchführen und anschließend dafür sorgen, dass sich der Fehler nicht wiederholt, oder notieren, wie sich das Problem zukünftig umgehen lässt.

AWS stellt geeignete Tools für alle Aspekte Ihrer Workloads und Betriebsabläufe als Code bereit und macht es Ihnen damit leichter, auf Ereignisse zu reagieren. Mithilfe dieser Tools können Sie Reaktionen auf betriebliche Ereignisse in Skripts definieren und diese Skripts dann als Reaktion auf Überwachungsdaten starten.

In AWS können Sie die Zeitdauer von Wiederherstellungsvorgängen verkürzen, indem Sie ausgefallene Komponenten einfach durch funktionierende Versionen ersetzen lassen, anstatt sie zu reparieren. Die ausgefallene Ressource können Sie dann genauer untersuchen, nachdem sie außer Betrieb genommen wurde.

Bewährte Methoden

- OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen
- OPS10-BP02 Implementieren eines Prozesses für jeden Alarm

- OPS10-BP03 Priorisieren von betrieblichen Ereignissen auf Basis der Auswirkung auf das Unternehmen
- OPS10-BP04 Definieren von Eskalationspfaden
- OPS10-BP05 Definieren eines Kundenkommunikationsplans für Ausfälle
- 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

Ihre Organisation hat Prozesse für die Bewältigung von Ereignissen, Vorfällen und Problemen. Ereignisse sind Dinge, die in Ihrem Workload auftreten, aber möglicherweise kein Eingreifen erfordern. Vorfälle sind Ereignisse, die ein Eingreifen erfordern. Probleme sind wiederkehrende Ereignisse, die ein Eingreifen erfordern oder nicht behoben werden können. Sie benötigen Prozesse, um die Auswirkungen solcher Ereignisse auf Ihr Unternehmen zu mindern und um sicherzustellen, dass Sie in angemessener Weise darauf reagieren.

Wenn Ihr Workload von Vorfällen und Problemen betroffen ist, benötigen Sie Prozesse, um diese zu bewältigen. Wie informieren Sie Stakeholder über den Status des Ereignisses? Wer leitet die Reaktion? Welche Tools verwenden Sie, um das Ereignis abzumildern? Dies sind Beispiele für Fragen, die Sie beantworten müssen, um einen fundierten Reaktionsprozess einführen zu können.

Prozesse müssen an zentraler Stelle dokumentiert werden und allen am Workload Beteiligten zur Verfügung stehen. Wenn Sie nicht über ein zentrales Wiki oder einen zentralen Dokumentenspeicher verfügen, können Sie dafür ein Repository für die Versionskontrolle verwenden. Sie halten diese Pläne aktuell, wenn sich die Prozesse weiterentwickeln.

Probleme sind Kandidaten für eine Automatisierung. Diese Ereignisse nehmen Zeit in Anspruch, die Sie eigentlich für Innovationen benötigen. Beginnen Sie mit der Entwicklung eines wiederholbaren Prozesses, um das Problem abzumildern. Konzentrieren Sie sich im Laufe der Zeit darauf, die Abmilderung zu automatisieren oder das zugrunde liegende Problem zu beheben. Dadurch sparen Sie Zeit ein, die Sie für Verbesserungen an Ihrem Workload aufwenden können.

Gewünschtes Ergebnis: Ihre Organisation hat einen Prozess für die Bewältigung von Ereignissen, Vorfällen und Problemen. Diese Prozesse werden dokumentiert und an zentraler Stelle gespeichert. Sie werden aktualisiert, wenn sich die Prozesse ändern.

Typische Anti-Muster:

- Ein Vorfall tritt am Wochenende ein und der Entwickler, der Rufbereitschaft hat, weiß nicht, was zu tun ist.
- Ein Kunde sendet Ihnen eine E-Mail, dass die Anwendung nicht verfügbar ist. Sie starten den Server neu, um das Problem zu beheben. Dies kommt häufig vor.
- Es gibt einen Vorfall und mehrere Teams arbeiten unabhängig voneinander daran, das Problem zu beheben.
- Es kommt zu Bereitstellungen in Ihrem Workload, die nicht dokumentiert werden.

Vorteile der Nutzung dieser bewährten Methode:

- Es gibt einen Prüfpfad der Ereignisse in Ihrem Workload.
- Die erforderliche Zeit für die Wiederherstellung nach einem Vorfall verringert sich.
- Die Teammitglieder können Vorfälle und Probleme einheitlich beheben.
- Bei der Untersuchung eines Vorfalls sind die Anstrengungen stärker miteinander verbunden.

Risikostufe bei fehlender Befolgung dieser Best Practice: Hoch

Implementierungsleitfaden

Wenn Sie diese Best Practice 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. Probleme werden identifiziert, priorisiert und behoben.

Kundenbeispiel

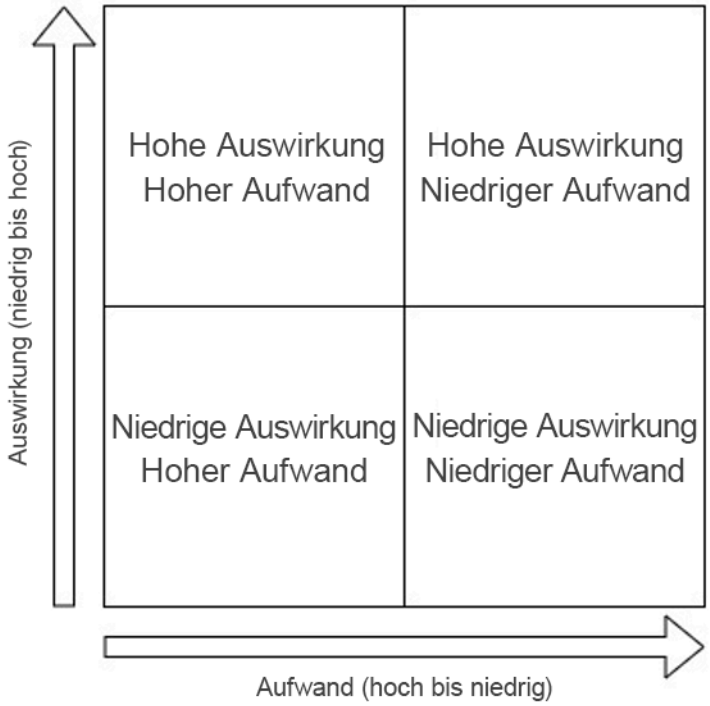
AnyCompany Retail verwendet einen Teil seines internen Wikis für Prozesse zur Verwaltung von Ereignissen, Vorfällen und Problemen. Alle Ereignisse werden an Amazon EventBridgegesendet. Probleme werden in AWS Systems Manager OpsCenter als OpsItems identifiziert und zur Behebung priorisiert, sodass undifferenzierter Arbeitsaufwand reduziert wird. Wenn die Prozesse sich ändern, werden sie im internen Wiki aktualisiert. Das Unternehmen nutzt AWS Systems Manager Incident Manager für die Verwaltung von Vorfällen und das Koordinieren von Maßnahmen zur Abmilderung.

Implementierungsschritte

1. Ereignisse

- Verfolgen Sie Ereignisse in Ihrem Workload nach, auch wenn kein menschliches Eingreifen erforderlich ist.
- Entwickeln Sie gemeinsam mit den Workload-Stakeholdern eine Liste der Ereignisse, die nachverfolgt werden sollten. Beispiele sind abgeschlossene Bereitstellungen oder erfolgreiche Patches.
- Sie können Services wie Amazon EventBridge oder Amazon Simple Notification Service nutzen, um benutzerdefinierte Ereignisse für die Nachverfolgung zu generieren.

2. Vorfälle
   - Definieren Sie zunächst den Kommunikationsplan für Vorfälle. Welche Stakeholder müssen informiert werden? Wie werden Sie sie auf dem Laufenden halten? Wer leitet die Koordination der Arbeiten? Wir empfehlen, einen internen Chat-Kanal für die Kommunikation und Koordination einzurichten.
   - Definieren Sie Eskalationspfade für die Teams, die Ihren Workload unterstützen, insbesondere wenn es im Team keine Rufbereitschaft gibt. Basierend auf Ihrem Support-Level können Sie auch einen Fall beim AWS Support öffnen.
   - Erstellen Sie ein Playbook, um den Vorfall zu untersuchen. Dieses sollte den Kommunikationsplan sowie detaillierte Maßnahmen zur Untersuchung beinhalten. Nehmen Sie in Ihre Untersuchung auch die Überprüfung von AWS Health Dashboard auf.
   - Dokumentieren Sie Ihren Reaktionsplan für Vorfälle. Kommunizieren Sie den Plan für das Vorfallmanagement, damit interne und externe Kunden die Regeln der Interaktion verstehen und wissen, was von ihnen erwartet wird. Schulen Sie die Teammitglieder hinsichtlich der Verwendung.
   - Kunden können Incident Manager nutzen, um ihren Reaktionsplan für Vorfälle einzurichten und zu verwalten.
   - 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.
3. Probleme
   - Probleme müssen identifiziert und in Ihrem ITSM-System nachverfolgt werden.
   - Identifizieren Sie alle bekannten Probleme und priorisieren Sie sie nach Aufwand der Behebung und Auswirkungen auf den Workload.



- Beheben Sie zunächst Probleme, die mit erheblichen Auswirkungen und geringem Aufwand verbunden sind. Sobald diese behoben sind, wechseln Sie zu Problemen, die in den Quadranten der Probleme mit geringen Auswirkungen und geringem Aufwand fallen.
- Sie können Systems Manager OpsCenter verwenden, um diese Probleme zu identifizieren, Runbooks daran anzufügen und sie nachzuverfolgen.

Aufwand für den Implementierungsplan: Mittel. Sie benötigen einen Prozess und Tools, um diese Best Practice zu implementieren. Dokumentieren Sie Ihre Prozesse und stellen Sie sie allen am Workload Beteiligten zur Verfügung. Aktualisieren Sie sie häufig. Sie haben einen Prozess für die Verwaltung und Abmilderung oder Behebung von Problemen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren: Bekannte Probleme benötigen ein angefügtes Runbook, damit die Maßnahmen zur Abmilderung einheitlich sind.
- OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen: Vorfälle müssen mithilfe von Playbooks untersucht werden.
- OPS11-BP02 Durchführen von Analysen nach Vorfällen: Führen Sie nach der Wiederherstellung nach einem Vorfall stets eine Post-Mortem-Analyse durch.

Zugehörige Dokumente:

- Atlassian - Incident management in the age of DevOps
- Leitfaden für AWS Security Incident Response
- Incident Management in the Age of DevOps and SRE
- PagerDuty - What is Incident Management?

Zugehörige Videos:

- AWS re:Invent 2020: Incident management in a distributed organization
- AWS re:Invent 2021 - Building next-gen applications with event-driven architectures
- AWS Supports You | Exploring the Incident Management Tabletop Exercise
- AWS Systems Manager Incident Manager - AWS Virtual Workshops
- AWS What's Next ft. Incident Manager | AWS Events

Zugehörige Beispiele:

- AWS Management and Governance Tools Workshop - OpsCenter
- AWS Proactive Services – Incident Management Workshop
- Building an event-driven application with Amazon EventBridge
- Building event-driven architectures on AWS

Zugehörige Services:

- Amazon EventBridge
- Amazon SNS
- AWS Health Dashboard
- AWS Systems Manager Incident Manager
- AWS Systems Manager OpsCenter

OPS10-BP02 Implementieren eines Prozesses für jeden Alarm

Legen Sie für jedes Ereignis, für das Sie einen Alarm auslösen, eine klar definierte Reaktion (Runbook oder Playbook) mit einem eigens dafür angegebenen Besitzer fest. Dies gewährleistet

eine effektive und schnelle Reaktion auf Betriebsereignisse und verhindert, dass aktionsrelevante Ereignisse aufgrund weniger wichtiger Benachrichtigungen übersehen werden.

Gängige Antimuster:

- Ihr Überwachungssystem präsentiert Ihnen einen Stream genehmigter Verbindungen zusammen mit anderen Nachrichten. Die Menge der Nachrichten ist so groß, dass Sie regelmäßig Fehlermeldungen verpassen, die eigentlich Ihren Eingriff erfordern würden.
- Sie erhalten eine Warnung, dass die Website nicht verfügbar ist. Es gibt keinen definierten Prozess dafür, wann dies geschieht. Sie müssen das Problem mit einem Ad-hoc-Ansatz diagnostizieren und lösen. Durch die individuelle Fehlerbehebung ohne vorgefertigte Prozesse verlängert sich die Zeit bis zur Wiederherstellung.

Vorteile der Einführung dieser bewährten Praxis: Indem Sie nur benachrichtigt werden, wenn tatsächlich eine Aktion erforderlich ist, verhindern Sie, dass wichtige Warnungen in einer Flut unwichtiger Informationen untergehen. Durch einen Prozess, der nur aktionsrelevante Warnungen ausgibt, ermöglichen Sie eine konsistente und schnelle Reaktion auf die Ereignisse in Ihrer Umgebung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Prozess pro Alarm: Jedem Ereignis, für das Sie eine Warnung auslösen, sollte eine klar definierte Reaktion (Runbook oder Playbook) mit einem speziellen Besitzer (z. B. eine Person, ein Team oder eine Rolle) zugewiesen sein, der für die erfolgreiche Ausführung verantwortlich ist. Die Reaktion kann zwar automatisiert oder von einem anderen Team übernommen werden, aber der Besitzer trägt die Verantwortung dafür, dass der Prozess die erwarteten Ergebnisse liefert. Diese Prozesse gewährleisten eine effektive und schnelle Reaktion auf Betriebsereignisse und verhindern, dass aktionsrelevante Ereignisse aufgrund weniger wichtiger Benachrichtigungen übersehen werden. Beispielsweise kann eine automatische Skalierung zur Skalierung eines Web-Front-End-Systems verwendet werden, aber das Team des operativen Bereichs könnte dafür verantwortlich sein, dass die Regeln und Limits der automatischen Skalierung den Anforderungen des Workloads entsprechen.

Ressourcen

Verbundene Dokumente:

- Amazon CloudWatch-Funktionen
- Was ist Amazon CloudWatch Events?

Verbundene Videos:

- Erstellen eines Überwachungsplans

OPS10-BP03 Priorisieren von betrieblichen Ereignissen auf Basis der Auswirkung auf das Unternehmen

Stellen Sie sicher, dass bei mehreren Ereignissen, die eine Intervention erfordern, zuerst diejenigen angegangen werden, die für das Unternehmen die größte Tragweite haben. Zu den Auswirkungen können Todesfälle oder Verletzungen, finanzielle Verluste oder Rufschädigung bzw. Vertrauensverlust gehören.

Gängige Antimuster:

- Sie erhalten eine Supportanfrage, in der Sie für einen Benutzer eine Druckerkonfiguration hinzufügen sollen. Während der Arbeit an dem Problem erhalten Sie eine Supportanfrage, dass Ihre Website für den Einzelhandel nicht mehr aufrufbar ist. Nachdem Sie die Druckerkonfiguration für den Benutzer abgeschlossen haben, beginnen Sie mit der Arbeit am Problem mit der Website.
- Sie werden benachrichtigt, dass sowohl Ihre Einzelhandelswebsite als auch Ihr System für die Lohn- und Gehaltsabrechnung ausgefallen sind. Sie wissen nicht, welches Problem Priorität haben sollte.

Vorteile der Einführung dieser bewährten Methode: Durch die Priorisierung von Reaktionen auf Vorfälle mit der größten Auswirkung auf das Unternehmen kommen Sie mit den Auswirkungen leichter zurecht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Priorisieren von operativen Ereignissen basierend auf den Auswirkungen auf das Geschäft: Wenn mehrere Ereignisse Eingriffe erfordern, stellen Sie sicher, dass diejenigen, die für das Geschäft am wichtigsten sind, zuerst behandelt werden. Zu den Auswirkungen können Todesfälle

oder Verletzungen, finanzielle Verluste, Verstöße gegen Vorschriften oder Rufschädigung bzw. Vertrauensverlust gehören.

OPS10-BP04 Definieren von Eskalationspfaden

Definieren Sie Eskalationspfade in Ihren Runbooks und Playbooks und legen Sie auch fest, was eine Eskalation auslöst. Erarbeiten Sie zudem Verfahren für die Eskalation. Weisen Sie jeder Aktion explizit Besitzer zu, um effektive und schnelle Reaktionen auf betriebliche Ereignisse zu gewährleisten.

Legen Sie fest, wann jemand eine Entscheidung treffen muss, bevor eine Aktion durchgeführt wird. Arbeiten Sie mit Entscheidungsträgern zusammen, um diese Entscheidung im Voraus treffen und die Aktion vorab genehmigen zu lassen, damit MTTR nicht auf eine Antwort wartet.

Gängige Antimuster:

- Ihre Einzelhandelswebsite ist nicht mehr aufrufbar. Sie verstehen das Runbook für die Wiederherstellung der Website nicht. Sie rufen Kollegen in der Hoffnung an, dass Ihnen jemand helfen kann.
- Sie erhalten eine Supportanfrage zu einer nicht erreichbaren Anwendung. Sie haben keine Berechtigungen für die Systemverwaltung. Sie wissen nicht, wer die Berechtigungen dafür hat. Sie versuchen, sich an den Besitzer des Systems zu wenden, der die Anfrage gestellt hat, und erhalten keine Antwort. Sie haben keine Kontakte für das System und Ihre Kollegen kennen sich damit nicht aus.

Vorteile der Einführung dieser bewährten Methode: Durch das Definieren von Eskalationen sowie von Auslösern und Verfahren für die Eskalation können Ressourcen einem Vorfall systematisch mit einer für die Auswirkungen geeigneten Menge hinzugefügt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Eskalationspfade definieren: Definieren Sie Eskalationspfade in Ihren Runbooks und Playbooks und legen Sie auch fest, was eine Eskalation auslöst. Erarbeiten Sie zudem Verfahren für die Eskalation. Beispielsweise kann ein Problem von den Support-Technikern eine Stufe höher an leitende Support-Techniker eskaliert werden, wenn das Problem nicht durch Runbooks gelöst werden kann oder wenn eine vordefinierte Zeitspanne verstrichen ist. Ein weiteres

Beispiel für einen geeigneten Eskalationspfad bei einem Workload ist die Weiterleitung von den leitenden Support-Technikern an das Entwicklungsteam, wenn die Playbooks keinen Korrekturpfad ermitteln können oder wenn eine vordefinierte Zeitspanne verstrichen ist. Weisen Sie jeder Aktion explizit Besitzer zu, um effektive und schnelle Reaktionen auf betriebliche Ereignisse zu gewährleisten. Eskalationen können auch Dritte beinhalten. Beispiele hierfür sind Anbieter von Netzwerkkonnektivität oder Software. Eskalationen können festgelegte autorisierte Entscheidungsträger für betroffene Systeme einbeziehen.

OPS10-BP05 Definieren eines Kundenkommunikationsplans für Ausfälle

Definieren und testen Sie einen Kommunikationsplan für Systemausfälle, auf den Sie sich verlassen können, um Ihre Kunden und Stakeholder bei Ausfällen auf dem Laufenden zu halten. Kommunizieren Sie direkt mit Ihren Benutzern – sowohl wenn die von ihnen genutzten Services beeinträchtigt werden als auch wenn die Services wieder normal funktionieren.

Gewünschtes Ergebnis:

- Sie verfügen über einen Kommunikationsplan für Situationen, die von geplanten Wartungsarbeiten bis hin zu großen, unerwarteten Fehlern reichen – einschließlich der Anwendung von Notfallwiederherstellungsplänen.
- In Ihrer Kommunikation stellen Sie klare und transparente Informationen zu Systemproblemen bereit, damit Ihre Kunden keine falschen Annahmen bezüglich der Leistung ihrer Systeme anstellen müssen.
- Sie verwenden individuelle Fehlermeldungen und Statusseiten, um Spitzen im Bereich der Helpdesk-Anfragen zu reduzieren und die Benutzer zu informieren.
- Der Kommunikationsplan wird regelmäßig getestet, um sicherzustellen, dass er bei einem tatsächlichen Ausfall wie vorgesehen funktioniert.

Typische Anti-Muster:

- Ein Workload-Ausfall tritt auf, aber Sie haben keinen Kommunikationsplan. Benutzer überhäufen Ihr Troubleticketsystem mit Anfragen, weil sie keine Informationen über den Ausfall haben.
- Sie senden während eines Ausfalls eine E-Mail-Benachrichtigung an Ihre Benutzer. Sie enthält keinen Zeitplan für die Wiederherstellung des Service, sodass die Benutzer nicht entsprechend planen können.

- Es gibt einen Kommunikationsplan für Ausfälle, aber er wurde nie getestet. Es kommt zu einem Ausfall und der Kommunikationsplan schlägt fehl, weil ein kritischer Schritt ausgelassen wurde, der beim Testen hätte erkannt werden können.
- Während eines Ausfalls senden Sie eine Benachrichtigung an die Benutzer. Diese enthält zu viele technische Details und Informationen, die unter Ihrer AWS NDA stehen.

Vorteile der Nutzung dieser bewährten Methode:

- Die kontinuierliche Kommunikation während des Ausfalls stellt sicher, dass die Kunden über den Fortschritt bei den Problemen und die geschätzte Zeit bis zur Lösung informiert sind.
- Die Entwicklung eines klar definierten Kommunikationsplans stellt sicher, dass Ihre Kunden und Endbenutzer gut informiert sind. So können sie die erforderlichen zusätzlichen Schritte unternehmen, um die Auswirkungen eines Ausfalls abzumildern.
- Mit einer angemessenen Kommunikation und einer stärkeren Sensibilisierung für geplante und ungeplante Ausfälle können Sie die Kundenzufriedenheit verbessern, ungewollte Reaktionen begrenzen und die Kundenbindung fördern.
- Eine rechtzeitige und transparente Kommunikation bei Systemausfällen schafft Vertrauen, das für eine gute Beziehung zwischen Ihnen und Ihren Kunden erforderlich ist.
- Eine bewährte Kommunikationsstrategie während eines Ausfalls oder einer Krise verhindert Spekulationen und Gerüchte. Diese könnten Ihre Möglichkeiten zur Wiederherstellung beeinträchtigen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: mittel

Implementierungsleitfaden

Kommunikationspläne, die Ihre Kunden während eines Ausfalls auf dem Laufenden halten, sind umfassend und decken mehrere Schnittstellen ab – einschließlich kundenseitiger Fehleranzeigen, individueller API-Fehlermeldungen, Systemstatus-Banner und Health-Statusseiten. Wenn Ihr System registrierte Benutzer umfasst, können Sie über Messaging-Kanäle wie E-Mail, SMS oder Push-Benachrichtigungen kommunizieren, um personalisierte Nachrichten an Ihre Kunden zu senden.

Tools zur Kundenkommunikation

Als erste Maßnahme sollten Web- und mobile Anwendungen während eines Ausfalls freundliche und informative Fehlermeldungen bereitstellen. Sie sollten außerdem die Möglichkeit bieten, den Datenverkehr auf eine Statusseite umzuleiten. Amazon CloudFront ist ein vollständig

verwaltetes Content Delivery Network (CDN), das Funktionen zur Definition und Bereitstellung angepasster Fehlerinhalte umfasst. Angepasste Fehlerseiten in CloudFront eignen sich als erste Kommunikationsebene für das Messaging bei Ausfällen auf Komponentenebene. CloudFront kann außerdem die Verwaltung und Aktivierung einer Statusseite vereinfachen, die alle Anfragen während geplanter oder ungeplanter Ausfälle auffängt.

Angepasste API-Fehlermeldungen können dazu beitragen, die Auswirkungen von Ausfällen auf einzelne Services zu erkennen und zu verringern. Mit Amazon API Gateway können Sie angepasste Antworten für Ihre REST-APIs konfigurieren. So können Sie API-Kunden klare und aussagekräftige Messaging-Meldungen zur Verfügung stellen, wenn API Gateway Backend-Services nicht erreichen kann. Außerdem können angepasste Messaging-Inhalte für Banner und Benachrichtigungen verwendet werden, falls eine bestimmte Funktion des Systems aufgrund von Ausfällen auf der Service-Schicht beeinträchtigt ist.

Das direkte Messaging ist die am stärksten personalisierte Form des Messagings für Kunden. Amazon Pinpoint ist ein verwalteter Service für die skalierbare Multi-Channel-Kommunikation. Amazon Pinpoint bietet Ihnen die Möglichkeit, Kampagnen zu erstellen, mit denen Sie das Messaging über SMS, E-Mail, Sprachnachrichten, Push-Benachrichtigungen oder von Ihnen definierte, maßgeschneiderte Kanäle umfassend an Ihren Kundenstamm verteilen können. Wenn Sie das Messaging mit Amazon Pinpoint verwalten, sind Nachrichtenkampagnen klar definiert, testbar und können intelligent auf spezifische Kundensegmente angewendet werden. Einmal eingerichtet, können Kampagnen geplant oder durch Ereignisse ausgelöst werden und lassen sich leicht testen.

Kundenbeispiel

Wenn der Workload gestört ist, sendet AnyCompany Retail eine E-Mail-Benachrichtigung an seine Benutzer. In der E-Mail wird beschrieben, welche Funktionen beeinträchtigt sind. Es wird eine realistische Einschätzung dazu bereitgestellt, wann der Service wiederhergestellt sein wird. Darüber hinaus gibt es eine Statusseite, die Echtzeitinformationen über den Zustand des Workloads anzeigt. Der Kommunikationsplan wird zweimal pro Jahr in einer Entwicklungsumgebung getestet, um seine Effektivität zu validieren.

Implementierungsschritte

1. Bestimmen Sie die Kommunikationskanäle für Ihre Messaging-Strategie. Berücksichtigen Sie die architektonischen Aspekte Ihrer Anwendung und bestimmen Sie die beste Strategie für die Übermittlung von Feedback an Ihre Kunden. Dazu könnten eine oder mehrere der skizzierten Strategien zum Einsatz kommen – einschließlich Fehler- und Statusseiten, angepasste API-Fehlerantworten oder ein Direkt-Messaging.

2. Entwerfen Sie Statusseiten für Ihre Anwendung. Wenn Sie festgestellt haben, dass Statusseiten oder angepasste Fehlerseiten für Ihre Kunden geeignet sind, müssen Sie den Inhalt und das Messaging für diese Seiten entwerfen. Fehlerseiten erklären den Benutzern, warum eine Anwendung nicht verfügbar ist, wann sie wieder verfügbar sein wird und was sie in der Zwischenzeit tun können. Falls Ihre Anwendung Amazon CloudFront verwendet, können Sie angepasste Fehlerantworten bereitstellen oder Lambda@Edge verwenden, um Fehler zu übersetzen und Seiteninhalte umzuschreiben. Mit CloudFront können Sie außerdem den Inhalt Ihrer Anwendung in einen statischen Amazon S3-Inhaltsursprung umwandeln, der Ihre Wartungs- oder Ausfallstatusseite enthält.
3. Entwerfen Sie den passenden Satz von API-Fehlerstatuswerten für Ihren Service. Fehlermeldungen, die im Fall von nicht erreichbaren Backend-Services von API Gateway erzeugt werden, sowie Ausnahmen auf der Service-Schicht enthalten möglicherweise keine für Endbenutzer geeigneten Meldungen. Mit angepassten Fehlerantworten von API Gateway können Sie HTTP-Antwortcodes zu kuratierten API-Fehlermeldungen zuordnen – und zwar ohne Codeänderungen an Ihren Backend-Services vornehmen zu müssen.
4. Entwerfen Sie das Messaging aus einer geschäftlichen Perspektive, sodass es für die Endbenutzer Ihres Systems relevant ist und keine technischen Details enthält. Denken Sie an Ihre Zielgruppe und stimmen Sie Ihr Messaging darauf ab. So können Sie beispielsweise interne Benutzer auf einen Workaround oder ein manuelles Verfahren hinweisen, das alternative Systeme nutzt. Externe Benutzer können gebeten werden, zu warten, bis das System wiederhergestellt ist, oder Updates zu abonnieren, damit sie eine Benachrichtigung erhalten, sobald das System wiederhergestellt ist. Definieren Sie das genehmigte Messaging für verschiedene Szenarien, einschließlich unerwarteter Ausfälle, geplanter Wartungsarbeiten und teilweiser Systemfehler, bei denen eine bestimmte Funktion beeinträchtigt oder nicht verfügbar ist.
5. Erstellen Sie Vorlagen und automatisieren Sie Ihr Messaging für Kunden. Sobald Sie den Inhalt Ihrer Nachrichten festgelegt haben, können Sie Amazon Pinpoint oder andere Tools verwenden, um Ihre Messaging-Kampagne zu automatisieren. Mit Amazon Pinpoint können Sie Kundenzielsegmente für bestimmte betroffene Benutzer erstellen und Nachrichten in Vorlagen umwandeln. Lesen Sie das Amazon Pinpoint-Tutorial, um zu erfahren, wie Sie eine Messaging-Kampagne einrichten.
6. Vermeiden Sie eine enge Kopplung von Messaging-Funktionen an Ihr kundenseitiges System. Ihre Messaging-Strategie sollte nicht von Daten oder Services des Systems abhängig sein. So stellen Sie sicher, dass Sie auch bei Ausfällen erfolgreich Nachrichten versenden können. Ziehen Sie in Betracht, Möglichkeiten zum Versenden von Nachrichten aus mehr als einer Availability Zone oder Region zu schaffen, um die Verfügbarkeit des Messagings zu gewährleisten. Wenn Sie AWS-

Services zum Versenden von Nachrichten verwenden, nutzen Sie Operationen auf Datenebene über Operationen auf Steuerebene, um Ihr Messaging auszulösen.

Grad des Aufwands für den Implementierungsplan: hoch Die Entwicklung eines Kommunikationsplans und der Mechanismen zum Senden von Nachrichten kann einen erheblichen Aufwand darstellen.

Ressourcen

Zugehörige bewährte Methoden:

- OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren - Ihr Kommunikationsplan sollte mit einem Runbook verknüpft sein, damit Ihre Mitarbeiter wissen, wie sie zu reagieren haben.
- OPS11-BP02 Durchführen von Analysen nach Vorfällen - Führen Sie nach einem Ausfall eine Post-Incident-Analyse durch, um Mechanismen zur Vermeidung eines weiteren Ausfalls zu ermitteln.

Zugehörige Dokumente:

- Error Handling Patterns in Amazon API Gateway and AWS Lambda (Muster für die Fehlerbehandlung in Amazon API Gateway und AWS Lambda)
- Amazon API Gateway-Antworten in API Gateway

Zugehörige Beispiele:

- AWS Health-Dashboard
- Summary of the AWS Service Event in the Northern Virginia (US-EAST-1) Region (Zusammenfassung des AWS-Service-Ereignisses in der Region Nord-Virginia (US-EAST-1))

Zugehörige Services:

- AWS Support
- AWS Kundenvereinbarung
- Amazon CloudFront
- Amazon API Gateway
- Amazon Pinpoint

- Amazon S3

OPS10-BP06 Bekanntgeben des Status über Dashboards

Stellen Sie Dashboards zur Verfügung, die auf die jeweilige Zielgruppe zugeschnitten sind (z. B. interne technische Teams, Führungskräfte und Kunden), um diese über den aktuellen Betriebsstatus des Unternehmens zu informieren und interessante Metriken bereitzustellen.

Sie können Dashboards mithilfe von Amazon CloudWatch Dashboards auf anpassbaren Homepages in der CloudWatch-Konsole erstellen. Mit Business-Intelligence-Services wie Amazon QuickSight können Sie interaktive Dashboards für Ihren Workload und den Betriebszustand (z. B. Bestellraten, verbundene Benutzer und Transaktionszeiten) erstellen und veröffentlichen. Erstellen Sie Dashboards, die Ihre Metriken auf System- und Geschäftsebene anzeigen.

Gängige Antimuster:

- Auf Anfrage führen Sie für die Verwaltung einen Bericht über die aktuelle Nutzung Ihrer Anwendung aus.
- Während eines Vorfalls werden Sie alle 20 Minuten von einem besorgten Besitzer eines Systems mit der Frage kontaktiert, ob der Fehler bereits behoben wurde.

Vorteile der Einführung dieser bewährten Methode: Durch das Erstellen von Dashboards aktivieren Sie den Self-Service-Zugriff auf Informationen. Dadurch können Ihre Kunden sich selbst informieren und feststellen, ob sie Maßnahmen ergreifen müssen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Status über Dashboards kommunizieren: Stellen Sie Dashboards zur Verfügung, die auf die jeweilige Zielgruppe zugeschnitten sind (z. B. interne technische Teams, Führungskräfte und Kunden), um diese über den aktuellen Betriebsstatus des Unternehmens zu informieren und interessante Metriken bereitzustellen. Die Bereitstellung einer Self-Service-Option für Statusinformationen reduziert Störungen aufgrund von gezielten Statusanfragen durch das Team des operativen Bereichs. Zu den Beispielen gehören Amazon CloudWatch-Dashboards und AWS Health Dashboard.
  - CloudWatch-Dashboards erstellen und nutzen benutzerdefinierte Metrikansichten

Ressourcen

Zugehörige Dokumente:

- Amazon QuickSight
- CloudWatch-Dashboards erstellen und nutzen benutzerdefinierte Metrikansichten

OPS10-BP07 Automatisieren von Reaktionen auf Ereignisse

Automatisieren Sie Reaktionen auf Ereignisse, um Fehler zu reduzieren, die durch manuelle Prozesse entstehen, und um schnelle und konsistente Reaktionen zu gewährleisten.

Es gibt mehrere Möglichkeiten, um Runbook- und Playbook-Aktionen auf AWS zu automatisieren. Um auf ein Ereignis aufgrund einer Statusänderung in Ihren AWS-Ressourcen oder von Ihren eigenen benutzerdefinierten Ereignissen zu reagieren, sollten Sie CloudWatch Events-Regeln erstellen, um Antworten über CloudWatch-Ziele (zum Beispiel Lambda-Funktionen, Amazon Simple Notification Service-Themen (Amazon SNS), Amazon ECS-Aufgaben und AWS Systems Manager Automation) auszulösen.

Für Reaktionen auf eine Metrik, die einen Schwellenwert für eine Ressource überschreitet (z. B. eine Wartezeit), sollten Sie CloudWatch-Alarme erstellen, um mittels Amazon EC2 oder Auto Scaling-Aktionen eine oder mehrere Aktionen durchzuführen oder um eine Benachrichtigung an ein Amazon SNS-Thema zu senden. Wenn als Reaktion auf einen Alarm benutzerdefinierte Aktionen durchgeführt werden sollen, rufen Sie Lambda per Amazon SNS-Benachrichtigung auf. Veröffentlichen Sie Ereignisbenachrichtigungen und Eskalationsmitteilungen per Amazon SNS, um alle Betroffenen zu informieren.

AWS unterstützt über die AWS-Service-APIs und -SDKs auch Systeme von Drittanbietern. Es gibt eine Reihe von Überwachungs-Tools, die von AWS-Partnern und Dritten zur Verfügung gestellt werden und die Überwachung, Benachrichtigungen und Reaktionen ermöglichen. Dazu gehören zum Beispiel New Relic, Splunk, Loggly, SumoLogic und Datadog.

Für den Fall, dass bei wichtigen Vorgängen automatisierte Verfahren fehlschlagen, sollten Sie manuelle Verfahren bereithalten.

Gängige Antimuster:

- Ein Entwickler überprüft seinen Code. Aufgrund des Ereignisses hätte ein Build gestartet und Tests hätten durchgeführt werden können, aber stattdessen passiert nichts.

- Ihre Anwendung protokolliert einen bestimmten Fehler, bevor sie nicht mehr funktioniert. Das Verfahren zum Neustarten der Anwendung ist bekannt und könnte skriptbasiert ausgeführt werden. Sie können das Protokollereignis verwenden, um ein Skript aufzurufen und die Anwendung neu zu starten. Stattdessen werden Sie am Sonntagmorgen um 3 Uhr geweckt, da Sie als verantwortliche Person für die Behebung von Problemen des Systems Bereitschaftsdienst haben, als der Fehler auftritt.

Vorteile der Einführung dieser bewährten Methode: Dank automatisierter Reaktionen auf Ereignisse reduzieren Sie die Reaktionszeit und begrenzen das Fehlerpotenzial manueller Aktivitäten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

- Reaktionen auf Ereignisse automatisieren: Automatisieren Sie Reaktionen auf Ereignisse, um Fehler zu reduzieren, die durch manuelle Prozesse entstehen, und um schnelle und konsistente Reaktionen zu gewährleisten.
  - Was ist Amazon CloudWatch Events?
  - Erstellen einer CloudWatch Events-Regel, die nach einem Ereignis ausgelöst wird
  - Erstellen einer CloudWatch Events-Regel, die nach einem AWS-API-Aufruf mithilfe von AWS CloudTrail ausgelöst wird
  - CloudWatch Events-Ereignisbeispiele aus unterstützten Services

Ressourcen

Zugehörige Dokumente:

- Amazon CloudWatch-Funktionen
- CloudWatch Events-Ereignisbeispiele aus unterstützten Services
- Erstellen einer CloudWatch Events-Regel, die nach einem AWS-API-Aufruf mithilfe von AWS CloudTrail ausgelöst wird
- Erstellen einer CloudWatch Events-Regel, die nach einem Ereignis ausgelöst wird
- Was ist Amazon CloudWatch Events?

Relevante Videos:

- Erstellen eines Überwachungsplans

Zugehörige Beispiele:

Weiterentwicklung

Weiterentwicklung bedeutet eine ständige Verbesserung im Laufe der Zeit. Implementieren Sie häufige kleine inkrementelle Änderungen basierend auf den aus Ihren Betriebsaktivitäten gewonnenen Erfahrungen.

Um Ihre Vorgänge im Laufe der Zeit weiterentwickeln zu können, müssen Sie Folgendes tun:

Themen

Lernen, Teilen und Verbessern

Es ist wichtig, dass Sie regelmäßig Zeiten einplanen, um betriebliche Aktivitäten und Fehler zu analysieren, zu experimentieren und Verbesserungen vorzunehmen. Wenn etwas schief läuft, soll Ihr Team und Ihr ganzes technisches Umfeld daraus lernen. Analysieren Sie Fehler, um daraus etwas zu lernen und entsprechende Verbesserungen zu planen. Gehen Sie Ihre Erkenntnisse mit anderen Teams durch, um sie zu überprüfen.

Bewährte Methoden

OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung

Bewerten Sie Ihren Workload anhand von bewährten Methoden für interne und externe Architekturen. Führen Sie mindestens einmal pro Jahr Überprüfungen des Workloads 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.
- Verbesserungsmöglichkeiten werden in Ihrem Softwareentwicklungsprozess gleichrangig behandelt.

Typische Anti-Muster:

- Sie haben seit der Einführung Ihres Workloads vor einigen Jahren keine Architekturüberprüfung durchgeführt.
- Verbesserungsmöglichkeiten erhalten eine niedrigere Priorität und bleiben im Backlog.
- Es gibt keinen Standard für die Umsetzung von Änderungen an bewährten Methoden für das Unternehmen.

Vorteile der Nutzung dieser bewährten Methode:

- Ihr Workload wird durch bewährte Methoden für die Architektur auf dem neuesten Stand gehalten.
- Ihr Workload wird auf bewusste Weise entwickelt.
- Sie können die bewährten Methoden des Unternehmens nutzen, um alle Workloads zu verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: hoch

Implementierungsleitfaden

Mindestens einmal im Jahr führen Sie 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.

Kundenbeispiel

Alle Workloads bei AnyCompany Retail werden einer jährlichen Architekturprüfung unterzogen. Das Unternehmen hat eine eigene Checkliste mit bewährten Methoden entwickelt, die für alle Workloads gelten. Mithilfe der Fokusbereiche von AWS Well-Architected Tool führt es Überprüfungen durch, indem es das Tool und den Fokusbereich mit bewährten Methoden verwendet. Verbesserungsmöglichkeiten, die sich aus den Prüfungen ergeben, werden in ihren Software-Sprints vorrangig behandelt.

Implementierungsschritte

1. Führen Sie mindestens einmal im Jahr eine Überprüfung der Architektur Ihres Workloads durch. Verwenden Sie einen dokumentierten Architekturstandard mit AWS-spezifischen bewährten Methoden.
   a. Wir empfehlen Ihnen, für diese Prüfungen Ihre eigenen, intern festgelegten Standards zu verwenden. Wenn Sie nicht über einen internen Standard verfügen, empfehlen wir Ihnen die Verwendung des AWS Well-Architected Framework.
   b. Sie können mit AWS Well-Architected Tool einen Fokusbereich Ihrer internen bewährten Methoden erstellen und Ihre Architekturprüfung durchführen.
   c. Kunden können sich an ihren AWS-Lösungsarchitekten wenden, um eine geführte Well-Architected Framework-Prüfung ihres Workloads durchzuführen.
2. Räumen Sie den während der Überprüfung ermittelten Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsprozess Priorität ein.

Grad des Aufwands für den Implementierungsplan: 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 – Eine weitere Quelle für Verbesserungsvorschläge ist die Analyse nach einem Vorfall. Nehmen Sie die gewonnenen Erkenntnisse in Ihre interne Liste der bewährten Methoden für die Architektur auf.
- OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen – Wenn Sie Ihre eigenen bewährten Methoden für die Architektur entwickeln, geben Sie diese in Ihrem Unternehmen weiter.

Zugehörige Dokumente:

- AWS Well-Architected Tool – Fokusbereiche
- AWS Well-Architected Whitepaper – Die Überprüfung
- Customize Well-Architected Reviews using Custom Lenses and the AWS Well-Architected Tool (Well-Architected-Prüfungen mit Fokusbereichen und dem AWS Well-Architected Tool anpassen)
- Implementing the AWS Well-Architected Custom Lens lifecycle in your organization (Implementieren des AWS Well-Architected-Fokusbereich-Lebenszyklus in Ihr Unternehmen)

Zugehörige Videos:

- Well-Architected Labs - Level 100: Custom Lenses on AWS Well-Architected Tool (Well-Architected Labs – Stufe 100: Fokusbereiche auf AWS Well-Architected Tool)

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.

Gängige Antimuster:

- 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 Einführung dieser bewährten Methode: Durch vordefinierte Prozesse zur Bestimmung der Komponenten, Bedingungen, Maßnahmen und Ereignisse, die zu einem Vorfall beigetragen haben, können Sie Verbesserungsmöglichkeiten ermitteln.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Verwenden eines Prozesses zur Ermittlung beitragender Faktoren: 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. Kommunizieren Sie die Ursache, soweit erforderlich, auf die jeweiligen Zielgruppen zugeschnitten.

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 Bedürfnisse externer Kunden 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:

Ermitteln Sie Verbesserungsfaktoren, um das Potenzial besser bewerten und priorisieren zu können.

In AWS können Sie die Protokolle all Ihrer betrieblichen Aktivitäten, Workloads und Infrastruktur zusammenstellen, um einen detaillierten Aktivitätsverlauf zu erstellen. Anschließend können Sie AWS-Tools verwenden, um Ihren Betrieb und den Workload-Zustand im Laufe der Zeit zu analysieren (z. B. Trends zu identifizieren, Ereignisse und Aktivitäten mit Ergebnissen zu korrelieren und zwischen Umgebungen und systemübergreifend zu vergleichen), um Verbesserungsmöglichkeiten basierend auf den auslösenden Faktoren aufzudecken.

Sie sollten API-Aktivitäten mithilfe von CloudTrail verfolgen (per AWS Management Console, Befehlszeilenschnittstelle, SDKs und APIs), um immer zu wissen, was sich bei Ihren Konten tut. Verfolgen Sie Bereitstellungsaktivitäten der AWS Developer Tools mit CloudTrail und CloudWatch nach. Dadurch wird Ihren CloudWatch Logs-Protokolldaten ein detaillierter Aktivitätsverlauf Ihrer Bereitstellungen und deren Ergebnisse hinzugefügt.

Exportieren Sie Ihre Protokolldaten zur langfristigen Speicherung inAmazon S3 . Mit AWS Gluekönnen Sie Ihre Protokolldaten in Amazon S3 für Analysen erkunden und vorbereiten. Verwendung Sie Amazon Athenadurch die native Integration mit AWS Glue, um Ihre Protokolldaten zu analysieren. Verwenden Sie ein Business Intelligence-Tool wie Amazon QuickSight , um Ihre Daten zu visualisieren, zu untersuchen und zu analysieren.

Gängige Antimuster:

- Sie haben ein Skript, das zwar funktioniert, aber optisch nicht viel hermacht. Sie investieren Zeit in das Umschreiben. Es ist jetzt ein wahres Kunstwerk.
- Ihr Start-up versucht, weitere Finanzierung von einem Risikokapitalgeber zu erhalten. Dieser möchte, dass Sie die Compliance mit PCI DSS nachweisen. Sie möchten diesem Wunsch entsprechen und Ihre Compliance dokumentieren. Dabei übersehen Sie jedoch ein Lieferdatum für einen Kunden und verlieren diesen. Vom Grundgedanken her war das nicht verkehrt, Sie fragen sich allerdings, ob Sie richtig gehandelt haben.

Vorteile der Einführung dieser bewährten Methode: Durch die Bestimmung der Kriterien, die Sie für die Verbesserung verwenden möchten, können Sie die Auswirkungen ereignisbasierter Motivationen oder emotionaler Investitionen minimieren.

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 Funktionen und Fähigkeiten.
    - Neuerungen bei AWS
  - Nicht akzeptable Probleme: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten nicht akzeptable Probleme, Fehler und Schwachstellen.
    - Aktuelle AWS-Sicherheitsmitteilungen
    - AWS Trusted Advisor
  - 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 Dokumente:

- Amazon Athena
- Amazon QuickSight
- AWS-Compliance
- Aktuelle Neuigkeiten zur AWS-Compliance
- AWS-Compliance-Programme
- AWS Glue
- Aktuelle AWS-Sicherheitsmitteilungen
- AWS Trusted Advisor
- Exportieren Sie Ihre Protokolldaten zur langfristigen Speicherung inAmazon S3
- Neuerungen bei AWS

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.

Gängige Antimuster:

- Sie sehen, dass die CPU-Auslastung auf einem System 95 % beträgt, und möchten mit Priorität eine Möglichkeit finden, die Auslastung dieses Systems zu reduzieren. Die beste Vorgehensweise ist die Skalierung nach oben. Das System wird als Transcoder verwendet und so skaliert, dass es jederzeit mit 95 % CPU-Auslastung ausgeführt wird. Der Besitzer des Systems hätte Ihnen die Situation erklären können, wenn Sie sich an ihn gewandt hätten. Sie haben Ihre Zeit nicht sinnvoll genutzt.

- Der Besitzer eines Systems behauptet, dass sein System geschäftskritisch sei. Das System wird nicht in einer Umgebung betrieben, die für hohe Sicherheit ausgelegt ist. Zur Verbesserung der Sicherheit implementieren Sie zusätzliche Erkennungs- und Präventivfunktionen, die für geschäftskritische Systeme erforderlich sind. Sie benachrichtigen den Besitzer des Systems, dass die Arbeit abgeschlossen ist und ihm die zusätzlichen Ressourcen in Rechnung gestellt werden. In der Diskussion nach dieser Benachrichtigung erfährt der Besitzer des Systems, dass es eine offizielle Definition für geschäftskritische Systeme gibt, die sein System nicht erfüllt.

Vorteile der Einführung dieser bewährten Methode: Durch die Prüfung von Erkenntnissen zusammen mit Geschäftsinhabern und Fachexperten können Sie ein gemeinsames Verständnis aufbauen und effektiver für Verbesserungen sorgen.

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.

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).

Gängige Antimuster:

- Eine wichtige Verkaufsaktion wurde durch Ihr Wartungsfenster unterbrochen. 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 erlitten einen längeren Ausfall, weil Sie eine fehlerhafte Bibliothek verwendet hatten, die häufig in Ihrem Unternehmen genutzt wird. Seitdem sind Sie zu einer zuverlässigen Bibliothek migriert.

Die anderen Teams in Ihrem Unternehmen wissen nicht, dass diese Gefahr besteht. Wenn Sie sich regelmäßig treffen und diesen Vorfall besprechen würden, wüssten sie über das Risiko Bescheid.

- Die Leistung Ihres Transcoders ist stetig gesunken und beeinträchtigt das Medienteam. Die Leistung ist noch nicht ganz schlimm. Sie haben aber keine Gelegenheit, von dem Problem zu erfahren, bis es so schlimm ist, dass daraus ein Vorfall entsteht. Würden Sie Ihre Betriebsmetriken gemeinsam mit dem Medienteam überprüfen, bestünde die Möglichkeit, die Metriken zu ändern, den vom Team spürbaren Leistungseinbruch zu erkennen und das Problem zu beheben.
- Sie prüfen nicht, wie zufrieden Kunden mit der Erfüllung Ihrer SLAs sind. Sie laufen Gefahr, die mit Kunden vereinbarten SLAs nicht zu erfüllen. Es gibt Geldstrafen im Zusammenhang mit der Nichteinhaltung von mit Kunden vereinbarten SLAs. Würden Sie die Metriken für diese SLAs bei regelmäßigen Treffen überprüfen, hätten Sie die Gelegenheit, das Problem zu erkennen und zu beheben.

Vorteile der Einführung dieser bewährten Methode: Durch regelmäßige Besprechungen zur Überprüfung von Betriebsmetriken, Ereignissen und Vorfällen schaffen Sie ein gemeinsames teamübergreifendes Verständnis, teilen gewonnene Erkenntnisse mit und können Verbesserungen priorisieren und gezielt in Angriff nehmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Prüfungen von Betriebsmetriken: Führen Sie regelmäßig teamübergreifend mit Teilnehmern aus verschiedenen Unternehmensbereichen nachträgliche Analysen der operationsspezifischen Metriken durch. Binden Sie alle Beteiligten, 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 Informationen zunutze, um Verbesserungspotenziale und mögliche Maßnahmen ausfindig zu machen.
  - Amazon CloudWatch
  - Verwenden von Amazon CloudWatch-Metriken
  - Veröffentlichen von benutzerdefinierten Metriken
  - Referenzinformationen zu Metriken und Dimensionen von Amazon CloudWatch

Ressourcen

Zugehörige Dokumente:

- Amazon CloudWatch
- Referenzinformationen zu Metriken und Dimensionen von Amazon CloudWatch
- Veröffentlichen von benutzerdefinierten Metriken
- Verwenden von Amazon CloudWatch-Metriken

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 weitergeben in Ihrem Unternehmen, damit alle davon profitieren. Informationen und Ressourcen sollten Sie weitergeben, um vermeidbare Fehler zu verhindern und Entwicklungsbemühungen zu unterstützen. Dies wird es Ihnen ermöglichen, sich auf die Bereitstellung gewünschter Funktionen zu konzentrieren.

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. Anschließend sollten Sie versionsgesteuerte AWS CodeCommit verwenden, um Anwendungsbibliotheken, skriptbasierte Verfahren, Verfahrens- und andere Systemdokumentationen freizugeben. Geben Sie Ihre Computing-Standards für andere frei, indem Sie den Zugriff auf Ihre AMIs freigeben und die Verwendung Ihrer Lambda-Funktionen kontenübergreifend erlauben. Auch Ihre Infrastrukturstandards sollten Sie als AWS CloudFormation-Vorlagen freigeben.

Über die AWS-APIs und -SDKs können Sie externe und von Drittanbietern stammende Tools und Repositorys integrieren (z. B. GitHub, BitBucket und SourceForge). Achten Sie bei der Freigabe Ihrer Erkenntnisse und Entwicklungen sorgfältig darauf, Berechtigungen so zu strukturieren, dass die Integrität freigegebener Repositorys nicht gefährdet wird.

Gängige Antimuster:

- Sie erlitten einen längeren Ausfall, weil Sie eine fehlerhafte Bibliothek verwendet hatten, die häufig in Ihrem Unternehmen genutzt wird. Seitdem sind Sie zu einer zuverlässigen Bibliothek migriert. Die anderen Teams in Ihrem Unternehmen wissen nicht, dass diese Gefahr besteht. Würden Sie

Ihre Erfahrungen mit dieser Bibliothek dokumentieren und weitergeben, wüssten die anderen Teams über das Risiko Bescheid.

- 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 Ihrem Unternehmen wissen nicht, dass diese Gefahr besteht. Würden Sie Ihre Erfahrungen mit dieser Bibliothek dokumentieren und weitergeben, wüssten die anderen Teams über das Risiko Bescheid.
- 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. Würden Sie Ihre Erfahrungen mit dieser Bibliothek dokumentieren und weitergeben, könnten auch andere davon profitieren.

Vorteile der Einführung dieser bewährten Methode: Gemeinsame Erkenntnisse unterstützen Verbesserungen und ermöglichen, erfahrungsbasierte Vorteile 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 Nebenprodukte. Veröffentlichen Sie beispielsweise aktualisierte Verfahren, Richtlinien, Governance und Best Practices in einem allgemein zugänglichen Wiki oder teilen Sie Skripte, Code und Bibliotheken über ein gemeinsames Repository.
    - Delegieren des Zugriffs auf Ihre AWS-Umgebung
    - Freigeben eines AWS CodeCommit-Repositorys
    - Unkomplizierte Autorisierung von AWS Lambda-Funktionen
    - Freigeben eines AMI mit bestimmten AWS-Konten
    - Schnelles Freigeben von Vorlagen mit einer AWS CloudFormation-Designer-URL
    - Verwenden von AWS Lambda mit Amazon SNS

Ressourcen

Zugehörige Dokumente:

- Unkomplizierte Autorisierung von AWS Lambda-Funktionen
- Freigeben eines AWS CodeCommit-Repositorys
- Freigeben eines AMI mit bestimmten AWS-Konten
- Schnelles Freigeben von Vorlagen mit einer AWS CloudFormation-Designer-URL
- Verwenden von AWS Lambda mit Amazon SNS

Relevante Videos:

- Delegieren des Zugriffs auf Ihre AWS-Umgebung

OPS11-BP09 Einplanen von Zeit für Verbesserungen

Reservieren Sie Zeit und Ressourcen innerhalb Ihrer Prozesse, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen.

In AWS können Sie temporäre Duplikate von Umgebungen erstellen. Das senkt die Risiken, Mühen 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.

Gängige Antimuster:

- Es besteht ein bekanntes Leistungsproblem auf Ihrem Anwendungsserver. Es wird im Backlog hinter jeder geplanten Funktionsimplementierung priorisiert. Bleibt die Rate der hinzugefügten geplanten Funktionen konstant, wird das Leistungsproblem niemals behoben.
- Um kontinuierliche Verbesserungen zu unterstützen, genehmigen Sie den Administratoren und Entwicklern, dass sie ihre Überstunden zur Auswahl und Implementierung von Verbesserungen nutzen können. Es werden niemals Verbesserungen vorgenommen.

Vorteile der Einführung dieser bewährten Methode: Indem Sie Zeit und Ressourcen innerhalb Ihrer Prozesse reservieren, ermöglichen Sie kontinuierliche, schrittweise Verbesserungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

- Zeit für Verbesserungen einplanen: 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. Wenn die Ergebnisse die Ziele nicht erfüllen und die Verbesserung immer noch Priorität hat, versuchen Sie alternative Vorgehensweisen.

Fazit

Operative Exzellenz ist ein fortlaufender und iterativer Prozess.

Richten Sie Ihr Unternehmen für Erfolg ein, indem Sie gemeinsame Ziele haben. Stellen Sie sicher, dass alle ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen und wie sie sich auf die Fähigkeit anderer zum Erfolg auswirken. Stellen Sie Ihren Teammitgliedern Support bereit, damit sie Ihre Geschäftsergebnisse unterstützen können.

Betrachten Sie jeden betrieblichen Vorfall oder Ausfall als eine Gelegenheit, den Betrieb Ihrer Architektur zu verbessern. Durch das Verständnis der Anforderungen Ihrer Workloads, das Vordefinieren von Runbooks für Routineaktivitäten und Playbooks zur Behebung von Problemen, die Verwendung der betrieblichen Vorgänge als Codefunktionen in AWS und das Aufrechterhalten des Situationsbewusstseins sind Ihre Vorgänge besser vorbereitet und Sie können bei Vorfällen effektiver reagieren.

Achten Sie darauf, schrittweise Verbesserungen auf der Grundlage von sich ändernden Prioritäten vorzunehmen, ziehen Sie aus jedem Ereignis entsprechende Erkenntnisse und führen Sie nachträgliche Analysen durch. So steigern Sie die Effizienz und Effektivität Ihrer Aktivitäten und stellen dadurch den Erfolg Ihres Unternehmens sicher.

AWS soll Ihnen helfen, Architekturen zu errichten und zu betreiben, die die Effizienz maximieren, während Sie Bereitstellungen erstellen, die schnell und anpassungsfähig sind. Damit Ihre Workloads operative Exzellenz erreichen, sollten Sie die bewährten Methoden anwenden, die in diesem Dokument aufgeführt sind.

Mitwirkende

- Rich Boyd, Operational Excellence Pillar Lead, Well-Architected, Amazon Web Services
- Jon Steele, Solutions Architect Well-Architected, Amazon Web Services
- Ryan King, Sr. Technical Program Manager, Amazon Web Services
- Chris Kunselman, Advisory Consultant, Amazon Web Services
- Peter Mullen, Advisory Consultant, Amazon Web Services
- Brian Quinn, Sr. Advisory Consultant, Amazon Web Services
- David Stanley, Cloud Operating Model Lead, Amazon Web Services
- Chris Kozlowski, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Alex Livingstone, Principal Specialist Solutions Architect, Cloud Operations, Amazon Web Services
- Paul Moran, Principal Technologist, Enterprise Support, Amazon Web Services
- Peter Mullen, Advisory Consultant, Professional Services, Amazon Web Services
- Chris Pates, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Arvind Raghunathan, Principal Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Ben Mergen, Senior Cost Lead Solutions Architect, Amazon Web Services

Weitere Informationen

Weitere Orientierungshilfe finden Sie in den folgenden Quellen:

- AWS Well-Architected Framework
- AWS-Architekturzentrum

Dokumentversionen

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

Änderung	Beschreibung	Datum
Umfangreiche Aktualisierung und Konsolidierung der Inhalte	Die Inhalte wurden aktualisiert und in mehrere Best-Practice-Bereiche zusammengefasst. Zwei Best-Practice-Bereiche (OPS 04 und OPS 08) wurden neu verfasst und mit neuen Inhalten und Schwerpunkten versehen. Die Best-Practices wurden aktualisiert und in folgende Bereiche zusammeng efasst: Design für den Betrieb, Bereitstellungsris iken abschwächenund Grundlegendes zum betriebli chen Status. Der Best-Prac tice-Bereich OPS 04 wurde geändert in Implementieren von Beobachtbarkeit. Der Best-Practice-Bereich OPS 08 wurde geändert in Nutzung der Workload-Beobachtbarkeit.	October 3, 2023
Updates für das neue Framework	Bewährte Methoden mit verbindlichen Anleitungen aktualisiert und neue bewährte Methoden hinzugefügt.	April 10, 2023

Whitepaper aktualisiert	Bewährte Methoden mit neuen Implementierungsanleitungen aktualisiert.	December 15, 2022
Whitepaper aktualisiert	Weitere bewährte Methoden und Verbesserungspläne hinzugefügt.	October 20, 2022
Kleineres Update	Es wurde eine kleine redaktionelle Aktualisierung vorgenommen.	August 8, 2022
Whitepaper aktualisiert	Aktualisierungen bezüglich neuer AWS-Services und -Funktionen sowie die neuesten bewährten Methoden.	February 2, 2022
Kleineres Update	Säule „Nachhaltigkeit“ wurde zur Einführung hinzugefügt.	December 2, 2021
Updates für das neue Framework	Aktualisierungen bezüglich neuer AWS-Services und -Funktionen sowie die neuesten bewährten Methoden.	July 8, 2020
Whitepaper aktualisiert	Aktualisierungen bezüglich neuer AWS-Services und -Funktionen sowie aktualisierte Verweise.	July 1, 2018
Erstveröffentlichung	Säule „Operative Exzellenz“ – AWS-Well-Architected-Framework veröffentlicht.	November 1, 2017