AWS-Whitepaper

Einführung in DevOps on AWS

Einführung in DevOps on AWS: AWS-Whitepaper

Table of Contents

Einführung in DevOps on AWS

Datum der Veröffentlichung: 7. April 2023 (Dokumentversionen)

Unternehmen begeben sich heute mehr denn je auf den Weg der digitalen Transformation, um engere Verbindungen zu ihren Kunden aufzubauen und so einen nachhaltigen und dauerhaften Geschäftswert zu erzielen. Organizations aller Formen und Größen stellen ihre Konkurrenz auf den Kopf und erschließen neue Märkte, indem sie schneller als je zuvor innovativ sind. Für diese Unternehmen ist es wichtig, sich auf Innovation und Software-Disruption zu konzentrieren, weshalb es entscheidend ist, ihre Softwarebereitstellung zu rationalisieren. Organizations, die ihre Zeit von der Idee bis zur Produktion verkürzen und dabei Geschwindigkeit und Agilität zur Priorität machen, könnten die Disruptoren von morgen sein.

Es gibt zwar mehrere Faktoren, die berücksichtigt werden müssen, um der nächste digitale disruptor zu werden, aber dieses Whitepaper konzentriert sich auf DevOps die Dienste und Funktionen der Amazon Web Services (AWS) -Plattform, die dazu beitragen werden, die Fähigkeit eines Unternehmens zu verbessern, Anwendungen und Dienste mit hoher Geschwindigkeit bereitzustellen.

Einführung

DevOps ist die Kombination aus kulturellen Philosophien, technischen Praktiken und Tools, die die Fähigkeit eines Unternehmens verbessern, Anwendungen und Dienste mit hoher Geschwindigkeit und besserer Qualität bereitzustellen. Im Laufe der Zeit haben sich bei der Einführung mehrere grundlegende Praktiken herauskristallisiert DevOps: Continuous Integration (CI), Continuous Delivery (CD), Infrastructure as Code (IaC) sowie Überwachung und Protokollierung.

In diesem paper werden AWS Funktionen vorgestellt, die Ihnen helfen, Ihre DevOps Reise zu beschleunigen, und wie AWS Services Ihnen helfen können, die undifferenzierte Arbeit zu beseitigen, die mit der DevOps Anpassung verbunden ist. Außerdem wird beschrieben, wie Sie kontinuierliche Integrations- und Bereitstellungsfunktionen aufbauen können, ohne Server verwalten oder Knoten aufbauen zu müssen, und wie Sie mithilfe von IaC Ihre Cloud-Ressourcen konsistent und wiederholbar bereitstellen und verwalten können.

- Kontinuierliche Integration: Ein Verfahren zur Softwareentwicklung, bei dem Entwickler ihre Codeänderungen regelmäßig in einem zentralen Repository zusammenführen. Danach werden automatisierte Builds und Tests ausgeführt.
- Kontinuierliche Bereitstellung: Eine Softwareentwicklungspraxis, bei der Codeänderungen automatisch erstellt, getestet und für eine Produktionsfreigabe vorbereitet werden.

- Infrastruktur als Code: Eine Praxis, bei der Infrastruktur mithilfe von Code- und Softwareentwicklungstechniken wie Versionskontrolle und kontinuierlicher Integration bereitgestellt und verwaltet wird.
- Überwachung und Protokollierung: Ermöglicht es Unternehmen, zu sehen, wie sich die Anwendungs- und Infrastrukturleistung auf das Nutzererlebnis ihrer Produkte auswirkt.
- Kommunikation und Zusammenarbeit: Es werden Verfahren eingeführt, um die Teams näher zusammenzubringen und Arbeitsabläufe zu entwickeln und die Zuständigkeiten dafür zu verteilen DevOps.
- Sicherheit: Sollte ein bereichsübergreifendes Anliegen sein. Ihre CI/CD-Pipelines (Continuous Integration and Continuous Delivery) und die damit verbundenen Dienste sollten geschützt und angemessene Zugriffsberechtigungen eingerichtet werden.

Eine Untersuchung der einzelnen Prinzipien zeigt, dass ein enger Zusammenhang mit den Angeboten von besteht. AWS

Sind Sie Well-Architected?

Das AWS Well-Architected Framework hilft Ihnen dabei, die Vor- und Nachteile der Entscheidungen zu verstehen, die Sie beim Aufbau von Systemen in der Cloud treffen. Die sechs Säulen des Frameworks ermöglichen es Ihnen, bewährte Architekturpraktiken für den Entwurf und Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Systeme kennenzulernen. Mit dem AWS Well-Architected Tool, das kostenlos in der AWS-Managementkonsole verfügbar ist, können Sie Ihre Workloads anhand dieser bewährten Methoden überprüfen, indem Sie für jede Säule eine Reihe von Fragen beantworten.

Kontinuierliche Integration

Continuous Integration (CI) ist eine Softwareentwicklungspraxis, bei der Entwickler ihre Codeänderungen regelmäßig in einem zentralen Code-Repository zusammenführen. Danach werden automatisierte Builds und Tests ausgeführt. CI hilft dabei, Fehler schneller zu finden und zu beheben, die Softwarequalität zu verbessern und den Zeitaufwand für die Validierung und Veröffentlichung neuer Softwareupdates zu reduzieren.

AWS bietet die folgenden Dienste für die kontinuierliche Integration:

Themen

- AWS CodeCommit
- AWS CodeBuild
- AWS CodeArtifact

AWS CodeCommit

AWS CodeCommitist ein sicherer, hoch skalierbarer, verwalteter Quellcodeverwaltungsdienst, der private Git-Repositorys hostet. CodeCommit reduziert die Notwendigkeit, dass Sie Ihr eigenes Quellcodeverwaltungssystem betreiben müssen, und es gibt keine Hardware, die bereitgestellt und skaliert werden muss, oder Software, die installiert, konfiguriert und betrieben werden muss. Sie können CodeCommit damit alles speichern, von Code bis hin zu Binärdateien, und es unterstützt die Standardfunktionalität von GitHub, sodass es nahtlos mit Ihren vorhandenen Git-basierten Tools zusammenarbeitet. Ihr Team kann auch die Online-Code-Tools verwenden CodeCommit, um Projekte zu durchsuchen, zu bearbeiten und gemeinsam daran zu arbeiten. AWS CodeCommit hat mehrere Vorteile:

- Zusammenarbeit — AWS CodeCommit ist für die kollaborative Softwareentwicklung konzipiert. Sie können Ihren Code einfach übertragen, verzweigen und zusammenführen, sodass Sie die Kontrolle über die Projekte Ihres Teams behalten können. CodeCommit unterstützt auch Pull-Requests, die einen Mechanismus bieten, um Code-Reviews anzufordern und Code mit Mitarbeitern zu besprechen.
- Verschlüsselung — Sie können Ihre Dateien nach Belieben AWS CodeCommit über HTTPS oder SSH zu und von dort übertragen. Ihre Repositorys werden auch im Ruhezustand automatisch über AWS Key Management Service(AWS KMS) mit kundenspezifischen Schlüsseln verschlüsselt.

- Zugriffskontrolle — AWS CodeCommit verwendet AWS Identity and Access Management(IAM), um zu kontrollieren und zu überwachen, wer auf Ihre Daten zugreifen kann und wie, wann und wo sie darauf zugreifen können. CodeCommit hilft Ihnen auch dabei, Ihre Repositorys über AWS CloudTrailund Amazon CloudWatch zu überwachen.

  Hohe Verfügbarkeit und Haltbarkeit — AWS CodeCommit speichert Ihre Repositorys in Amazon Simple Storage Service (Amazon S3) und Amazon DynamoDB. Ihre verschlüsselten Daten werden redundant an mehreren Standorten gespeichert. Diese Architektur erhöht die Verfügbarkeit und Haltbarkeit Ihrer Repository-Daten.
- Benachrichtigungen und benutzerdefinierte Skripte — Sie können jetzt Benachrichtigungen über Ereignisse erhalten, die sich auf Ihre Repositorys auswirken. Benachrichtigungen werden als Amazon Simple Notification Service (Amazon SNS) -Benachrichtigungen gesendet. Jede Benachrichtigung enthält eine Statusmeldung sowie einen Link zu den Ressourcen, deren Ereignis die Benachrichtigung ausgelöst hat. Darüber hinaus können Sie mithilfe von AWS CodeCommit Repository-Cues Benachrichtigungen senden und HTTP-Webhooks mit Amazon SNS erstellen oder AWS LambdaFunktionen als Reaktion auf die von Ihnen ausgewählten Repository-Ereignisse aufrufen.

AWS CodeBuild

AWS CodeBuild ist ein vollständig verwalteter Service für die kontinuierliche Integration, der Quellcode kompiliert, Tests ausführt und Softwarepakete generiert, die direkt bereitgestellt werden können. Sie müssen Ihre eigenen Build-Server nicht bereitstellen, verwalten und skalieren. CodeBuild kann entweder GitHub Enterprise GitHub, BitBucket AWS CodeCommit, oder Amazon S3 als Quellanbieter verwenden.

CodeBuild skaliert kontinuierlich und kann mehrere Builds gleichzeitig verarbeiten. CodeBuild bietet verschiedene vorkonfigurierte Umgebungen für verschiedene Versionen von Microsoft Windows und Linux. Kunden können ihre maßgeschneiderten Build-Umgebungen auch als Docker-Container mitbringen. CodeBuild lässt sich auch in Open-Source-Tools wie Jenkins und Spinnaker integrieren.

CodeBuild kann auch Berichte für Einheiten-, Funktions- oder Integrationstests erstellen. Diese Berichte bieten einen visuellen Überblick darüber, wie viele Testfälle ausgeführt wurden und wie viele bestanden oder fehlgeschlagen sind. Der Erstellungsprozess kann auch in einer Amazon Virtual Private Cloud (Amazon VPC) ausgeführt werden. Dies kann hilfreich sein, wenn Ihre Integrationsdienste oder Datenbanken in einer VPC bereitgestellt werden.

AWS CodeArtifact

AWS CodeArtifactist ein vollständig verwalteter Artefakt-Repository-Service, der von Unternehmen verwendet werden kann, um Softwarepakete, die in ihrem Softwareentwicklungsprozess verwendet werden, sicher zu speichern, zu veröffentlichen und gemeinsam zu nutzen. CodeArtifact kann so konfiguriert werden, dass Softwarepakete und Abhängigkeiten automatisch aus öffentlichen Artefakt-Repositorys abgerufen werden, sodass Entwickler Zugriff auf die neuesten Versionen haben.

Softwareentwicklungsteams verlassen sich zunehmend auf Open-Source-Pakete, um allgemeine Aufgaben in ihren Anwendungspaketen auszuführen. Für Softwareentwicklungsteams ist es von entscheidender Bedeutung geworden, die Kontrolle über eine bestimmte Version der Open-Source-Software zu behalten, um sicherzustellen, dass die Software frei von Sicherheitslücken ist. Mit CodeArtifact können Sie Kontrollen einrichten, um dies durchzusetzen.

CodeArtifact funktioniert mit häufig verwendeten Paketmanagern und Build-Tools wie Maven, Gradle, npm, yarn, twine und pip und erleichtert so die Integration in bestehende Entwicklungsworkflows.

Fortlaufende Lieferung

Continuous Delivery (CD) ist eine Softwareentwicklungspraxis, bei der Codeänderungen automatisch für eine Produktionsfreigabe vorbereitet werden. Continuous Delivery ist ein Eckpfeiler der modernen Anwendungsentwicklung und erweitert die kontinuierliche Integration, indem alle Codeänderungen nach der Build-Phase in einer Test- und/oder Produktionsumgebung bereitgestellt werden. Bei ordnungsgemäßer Implementierung verfügen Entwickler immer über ein einsatzbereites Build-Artefakt, das einen standardisierten Testprozess durchlaufen hat.

Continuous Delivery ermöglicht es Entwicklern, Tests zu automatisieren, die über Komponententests hinausgehen, sodass sie Anwendungsaktualisierungen vor der Bereitstellung für Kunden in mehreren Dimensionen überprüfen können.

Diese Tests können UI-Tests, Auslastungstests, Integrationstests, API-Zuverlässigkeitstests und mehr umfassen. Dies hilft Entwicklern, Updates gründlicher zu validieren und Probleme präventiv zu entdecken. Mithilfe der Cloud ist es einfach und kostengünstig, die Erstellung und Replikation mehrerer Testumgebungen zu automatisieren, was zuvor vor Ort schwierig war.

AWS bietet die folgenden Services für Continuous Delivery:

- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline

Themen

- AWS CodeDeploy
- AWS CodePipeline

AWS CodeDeploy

AWS CodeDeployist ein vollständig verwalteter Bereitstellungsservice, der Softwarebereitstellungen für eine Vielzahl von Rechendiensten wie Amazon Elastic Compute Cloud (Amazon EC2), AWS Fargate AWS Lambda, und auf Ihren lokalen Servern automatisiert. AWS CodeDeploy erleichtert Ihnen die schnelle Veröffentlichung neuer Funktionen, hilft Ihnen, Ausfallzeiten bei der Anwendungsbereitstellung zu vermeiden, und bewältigt die Komplexität der Aktualisierung Ihrer

Anwendungen. Sie können CodeDeploy damit Softwarebereitstellungen automatisieren und so den Bedarf an fehleranfälligen manuellen Vorgängen reduzieren. Der Service lässt sich an Ihre Bereitstellungsanforderungen anpassen.

CodeDeploy hat mehrere Vorteile, die dem DevOps Prinzip der kontinuierlichen Bereitstellung entsprechen:

- Automatisierte Bereitstellungen — automatisiert Softwarebereitstellungen CodeDeploy vollständig, sodass Sie sie zuverlässig und schnell bereitstellen können.
- Zentralisierte Steuerung — CodeDeploy ermöglicht es Ihnen, Ihre Anwendungsbereitstellungen einfach zu starten und den Status Ihrer Anwendungsbereitstellungen über die oder die zu verfolgen. AWS-Managementkonsole AWS CLI CodeDeploybietet Ihnen einen detaillierten Bericht, anhand dessen Sie sehen können, wann und wo die einzelnen Anwendungsversionen bereitgestellt wurden. Sie können auch Push-Benachrichtigungen erstellen, um Live-Updates über Ihre Bereitstellungen zu erhalten.
- Minimierung von Ausfallzeiten — CodeDeploy trägt dazu bei, die Verfügbarkeit Ihrer Anwendungen während des Softwarebereitstellungsprozesses zu maximieren. Es führt schrittweise Änderungen ein und verfolgt den Zustand der Anwendung anhand konfigurierbarer Regeln. Softwarebereitstellungen können einfach gestoppt und bei Fehlern rückgängig gemacht werden.
- Einfach zu implementieren — CodeDeploy funktioniert mit jeder Anwendung und bietet dieselbe Benutzererfahrung auf verschiedenen Plattformen und Sprachen. Sie können Ihren vorhandenen Setup-Code problemlos wiederverwenden. CodeDeploy kann auch in Ihren bestehenden Software-Release-Prozess oder Ihre Continuous-Delivery-Toolchain (z. B., AWS CodePipeline GitHub, Jenkins) integriert werden.

AWS CodeDeploy unterstützt mehrere Bereitstellungsoptionen. Weitere Informationen finden Sie im Abschnitt Bereitstellungsstrategien dieses Dokuments.

AWS CodePipeline

AWS CodePipelineist ein Continuous-Delivery-Service, mit dem Sie die zur Veröffentlichung Ihrer Software erforderlichen Schritte modellieren, visualisieren und automatisieren können. Mit AWS CodePipeline modellieren Sie den vollständigen Release-Prozess für die Erstellung Ihres Codes, die Bereitstellung in Vorproduktionsumgebungen, das Testen Ihrer Anwendung und deren Freigabe für die Produktion. AWS CodePipeline erstellt, testet und stellt Ihre Anwendung dann bei jeder Codeänderung gemäß dem definierten Workflow bereit. Sie können Partner-Tools und Ihre eigenen

benutzerdefinierten Tools in jede Phase des Release-Prozesses integrieren, um eine end-to-end Continuous-Delivery-Lösung zu bilden.

AWS CodePipeline hat mehrere Vorteile, die dem DevOps Prinzip der kontinuierlichen Bereitstellung entsprechen:

- Schnelle Bereitstellung — AWS CodePipeline automatisiert Ihren Softwareveröffentlichungsprozess, sodass Sie Ihren Benutzern schnell neue Funktionen zur Verfügung stellen können. Mit CodePipeline können Sie schnell auf Feedback reagieren und Ihren Benutzern neue Funktionen schneller zur Verfügung stellen.
- Verbesserte Qualität — Durch die Automatisierung Ihrer Build-, Test- und Release-Prozesse AWS CodePipeline können Sie die Geschwindigkeit und Qualität Ihrer Softwareupdates erhöhen, indem Sie alle neuen Änderungen einer konsistenten Reihe von Qualitätsprüfungen unterziehen.
- Einfach zu integrieren — AWS CodePipeline kann einfach erweitert werden, um es an Ihre spezifischen Bedürfnisse anzupassen. Sie können die vorgefertigten Plugins oder Ihre eigenen benutzerdefinierten Plugins in jedem Schritt Ihres Veröffentlichungsprozesses verwenden. Sie können beispielsweise Ihren Quellcode von Ihrem lokalen Jenkins-Build-Server abrufen GitHub, Lasttests mit einem Drittanbieter-Service ausführen oder Bereitstellungsinformationen an Ihr benutzerdefiniertes Operations-Dashboard weitergeben.
- Konfigurierbarer Workflow — AWS CodePipeline ermöglicht es Ihnen, die verschiedenen Phasen Ihres Softwareveröffentlichungsprozesses mithilfe der Konsolenschnittstelle, der AWS CLICloudFormation, oder der AWS zu modellieren SDKs. Sie können ganz einfach angeben, welche Tests ausgeführt werden sollen, und die Schritte zur Bereitstellung Ihrer Anwendung und ihrer Abhängigkeiten anpassen.

Bereitstellungsstrategien

Bereitstellungsstrategien definieren, wie Sie Ihre Software bereitstellen möchten. Organizations verfolgen je nach Geschäftsmodell unterschiedliche Bereitstellungsstrategien. Einige entscheiden sich dafür, vollständig getestete Software anzubieten, und andere möchten vielleicht, dass ihre Benutzer Feedback geben und ihre Benutzer Funktionen testen lassen, die sich noch in der Entwicklung befinden (z. B. Betaversionen). Im folgenden Abschnitt werden verschiedene Bereitstellungsstrategien beschrieben.

In-Situ-Bereitstellungen

Bei dieser Strategie wird die vorherige Version der Anwendung auf jeder Rechenressource gestoppt, die neueste Anwendung installiert und die neue Version der Anwendung gestartet und validiert. Auf diese Weise können Anwendungsbereitstellungen mit minimaler Beeinträchtigung der zugrunde liegenden Infrastruktur fortgesetzt werden. Bei einer direkten Bereitstellung können Sie Ihre Anwendung bereitstellen, ohne eine neue Infrastruktur erstellen zu müssen. Allerdings kann die Verfügbarkeit Ihrer Anwendung während dieser Bereitstellungen beeinträchtigt werden. Dieser Ansatz minimiert auch die Infrastrukturkosten und den Verwaltungsaufwand, der mit der Erstellung neuer Ressourcen verbunden ist. Sie können einen Load Balancer verwenden, sodass jede Instanz während der Bereitstellung deregistriert und nach Abschluss der Bereitstellung wieder betriebsbereit ist. Direkte Bereitstellungen können all-at-once unter der Annahme eines Serviceausfalls oder als fortlaufendes Update erfolgen. AWS CodeDeploy und AWS Elastic Beanstalk bieten Bereitstellungskonfigurationen für one-at-a-time half-at-a-time, und. all-at-once

Blau/Grün-Bereitstellung

Die blaue/grüne Bereitstellung, manchmal auch als red/black deployment, is a technique for releasing applications by shifting traffic between two identical environments running differing versions of the application. Blue/green Bereitstellung bezeichnet, hilft Ihnen dabei, Ausfallzeiten bei Anwendungsupdates zu minimieren und so die Risiken im Zusammenhang mit Ausfallzeiten und Rollback-Funktionen zu minimieren.

Blau/Grün-Bereitstellungen ermöglichen es Ihnen, neben der alten Version (blau) auch eine neue Version (grün) Ihrer Anwendung zu starten und die neue Version zu überwachen und zu testen, bevor Sie den Datenverkehr zu ihr umleiten, und bei Problemerkennung ein Rollback durchführen.

Bereitstellung auf Canary

Der Zweck einer Bereitstellung auf Canary besteht darin, das Risiko der Bereitstellung einer neuen Version zu verringern, die sich auf die Arbeitslast auswirkt. Bei dieser Methode wird die neue Version schrittweise bereitgestellt, sodass sie für neue Benutzer langsam sichtbar wird. Sobald Sie Vertrauen in die Bereitstellung gewinnen, werden Sie sie bereitstellen, um die aktuelle Version vollständig zu ersetzen.

Lineare Bereitstellung

Lineare Bereitstellung bedeutet, dass der Verkehr in gleichen Schritten verschoben wird, wobei zwischen den einzelnen Schritten die gleiche Anzahl von Minuten liegt. Sie können aus vordefinierten linearen Optionen wählen, die den prozentualen Anteil des Datenverkehrs angeben, der in jedem Inkrementschritt verschoben wird, sowie die Anzahl der Minuten zwischen den einzelnen Inkrementschritten.

All-at-once Einsatz

All-at-onceBereitstellung bedeutet, dass der gesamte Datenverkehr auf einmal von der ursprünglichen Umgebung in die Ersatzumgebung verlagert wird.

Matrix der Bereitstellungsstrategien

In der folgenden Matrix sind die unterstützten Bereitstellungsstrategien für Amazon Elastic Container Service (Amazon ECS) und Amazon EC2 /on-premises aufgeführt. AWS Lambda

- Amazon ECS ist ein vollständig verwalteter Orchestrierungsservice.
- AWS Lambda ermöglicht es Ihnen, Code auszuführen, ohne Server bereitzustellen oder zu verwalten.
- Amazon EC2 ermöglicht es Ihnen, sichere, anpassbare Rechenkapazität in der Cloud zu betreiben.

Bereitstellungsstrategie	Amazon ECS	AWS Lambda	Amazon EC2 / vor Ort
In-Situ	✓	✓	✓
Blau/Grün	✓	✓	✓*
Canary	✓	✓	X
Linear	✓	✓	X
Ein ll-at-once	✓	✓	X

> **Note**
>
> Blue/green deployment with EC2/on-premises funktioniert nur mit EC2 Instanzen.

AWS Elastic Beanstalk Bereitstellungsstrategien

AWS Elastic Beanstalkunterstützt die folgenden Arten von Bereitstellungsstrategien:

- A ll-at-once Führt die Bereitstellung vor Ort auf allen Instanzen durch.
- Beim Rolling werden die Instanzen in Batches aufgeteilt und jeweils für einen Stapel bereitgestellt.

- Beim Rolling mit zusätzlichem Batch werden die Bereitstellungen in Batches aufgeteilt, für den ersten Batch werden jedoch neue EC2 Instanzen erstellt, anstatt sie auf den vorhandenen Instanzen bereitzustellen. EC2
- Unveränderlich, wenn Sie die Bereitstellung mit einer neuen Instanz durchführen müssen, anstatt eine bestehende Instanz zu verwenden.
- Traffic Splitting Führt eine unveränderliche Bereitstellung durch und leitet dann für einen vorher festgelegten Zeitraum einen Prozentsatz des Datenverkehrs an die neuen Instances weiter. Wenn die Instances fehlerfrei bleiben, leiten Sie den gesamten Traffic an neue Instances weiter und fahren Sie alte Instances herunter.

Infrastructure as Code

Ein grundlegendes Prinzip von DevOps besteht darin, die Infrastruktur genauso zu behandeln, wie Entwickler Code behandeln. Der Anwendungscode hat ein definiertes Format und eine definierte Syntax. Wenn der Code nicht nach den Regeln der Programmiersprache geschrieben ist, können keine Anwendungen erstellt werden. Code wird in einem Versionsverwaltungs- oder Quellcodeverwaltungssystem gespeichert, das eine Historie der Codeentwicklung, Änderungen und Bugfixes protokolliert. Wenn Code kompiliert oder in Anwendungen integriert wird, erwarten wir, dass eine konsistente Anwendung erstellt wird und dass der Build wiederholbar und zuverlässig ist.

Infrastruktur als Code zu praktizieren bedeutet, dieselbe Strenge wie bei der Entwicklung von Anwendungscode auch auf die Bereitstellung der Infrastruktur anzuwenden. Alle Konfigurationen sollten deklarativ definiert und in einem Quellcodeverwaltungssystem gespeichert werden AWS CodeCommit, z. B. genauso wie der Anwendungscode. Die Bereitstellung, Orchestrierung und Bereitstellung der Infrastruktur sollte auch die Verwendung der Infrastruktur als Code unterstützen.

Die Infrastruktur wurde traditionell mithilfe einer Kombination aus Skripten und manuellen Prozessen bereitgestellt. Manchmal wurden diese Skripts in Versionskontrollsystemen gespeichert oder Schritt für Schritt in Textdateien oder Runbooks dokumentiert. Oft ist die Person, die die Runbooks schreibt, nicht dieselbe Person, die diese Skripts ausführt oder die Runbooks durchsucht. Wenn diese Skripts oder Runbooks nicht häufig aktualisiert werden, können sie bei Bereitstellungen zu einem echten Hingucker werden. Dies führt dazu, dass neue Umgebungen nicht immer wiederholbar, zuverlässig oder konsistent sind.

Im Gegensatz dazu AWS bietet es eine DevOps gezielte Methode zur Schaffung und Wartung der Infrastruktur. Ähnlich wie Softwareentwickler Anwendungscode schreiben, AWS stellt es Dienste bereit, die die Erstellung, Bereitstellung und Wartung der Infrastruktur auf programmatische, beschreibende und deklarative Weise ermöglichen. Diese Dienste bieten Genauigkeit, Klarheit und Zuverlässigkeit. Die in diesem paper erörterten AWS Dienstleistungen sind der Kern einer DevOps Methodik und bilden die Grundlage zahlreicher AWS DevOps übergeordneter Prinzipien und Praktiken.

AWS bietet die folgenden Dienste zur Definition von Infrastruktur als Code an.

Dienstleistungen

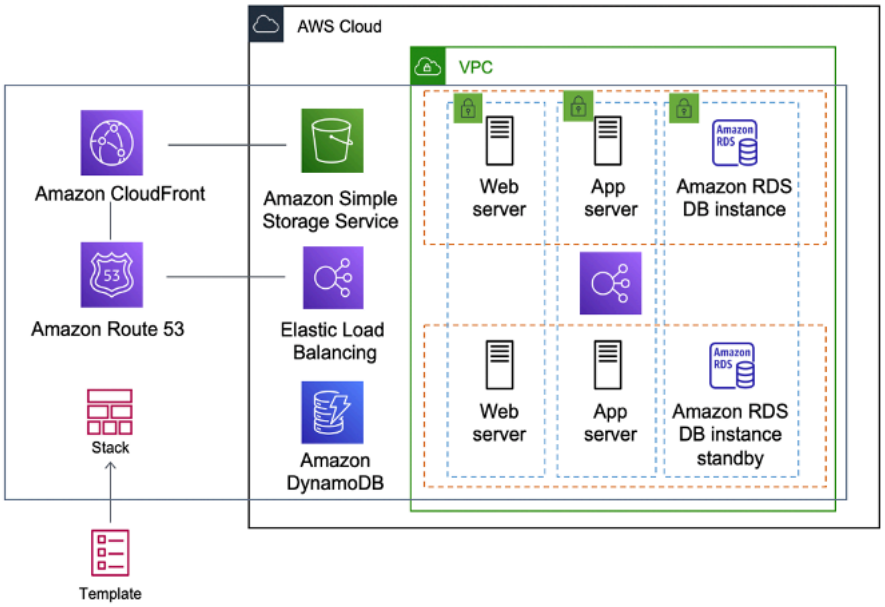
- CloudFormation
- AWS Serverless Application Model

- AWS Cloud Development Kit (AWS CDK)
- AWS-Cloud-Entwicklungskit für Kubernetes
- AWS-Cloud-Entwicklungskit für Terraform
- AWS -Cloud-Control- API

CloudFormation

AWS CloudFormation ist ein Dienst, der es Entwicklern ermöglicht, AWS Ressourcen auf geordnete und vorhersehbare Weise zu erstellen. Ressourcen werden in Textdateien im JSON- oder YAML-Format geschrieben. Die Vorlagen erfordern eine bestimmte Syntax und Struktur, die von den Arten der zu erstellenden und zu verwaltenden Ressourcen abhängt. Sie verfassen Ihre Ressourcen in JSON oder YAML mit einem beliebigen Code-Editor AWS Cloud9, checken sie in ein Versionskontrollsystem ein und erstellen dann die CloudFormation angegebenen Dienste auf sichere, wiederholbare Weise.

Eine CloudFormation Vorlage wird als Stack in der AWS Umgebung bereitgestellt. Sie können Stapel über AWS-Managementkonsole AWS Command Line Interface, oder CloudFormation APIs verwalten. Wenn Sie Änderungen an den laufenden Ressourcen in einem Stack vornehmen müssen, aktualisieren Sie den Stack. Bevor Sie Änderungen an Ihren Ressourcen vornehmen, können Sie einen Änderungssatz generieren, eine Zusammenfassung der von Ihnen vorgeschlagenen Änderungen. Mithilfe von Änderungssätzen können Sie vor der Implementierung feststellen, wie sich Ihre Änderungen auf Ihre laufenden Ressourcen auswirken könnten, insbesondere auf kritische Ressourcen.



AWS CloudFormation Erstellen einer gesamten Umgebung (Stapel) aus einer Vorlage

Sie können eine einzelne Vorlage verwenden, um eine gesamte Umgebung zu erstellen und zu aktualisieren, oder separate Vorlagen, um mehrere Ebenen innerhalb einer Umgebung zu verwalten. Dies ermöglicht die Modularisierung von Vorlagen und bietet zudem eine Verwaltungsebene, die für viele Organisationen wichtig ist.

Wenn Sie einen Stack in der CloudFormation Konsole erstellen oder aktualisieren, werden Ereignisse angezeigt, die den Status der Konfiguration anzeigen. Wenn ein Fehler auftritt, wird der Stack standardmäßig auf seinen vorherigen Status zurückgesetzt. Amazon SNS bietet Benachrichtigungen zu Ereignissen. Sie können Amazon SNS beispielsweise verwenden, um den Fortschritt bei der Erstellung und Löschung von Stacks per E-Mail zu verfolgen und programmgesteuert in andere Prozesse zu integrieren.

AWS CloudFormation macht es einfach, eine Sammlung von AWS Ressourcen zu organisieren und bereitzustellen, und ermöglicht es Ihnen, bei der Konfiguration des Stacks alle Abhängigkeiten zu beschreiben oder spezielle Parameter zu übergeben.

Mit CloudFormation Vorlagen können Sie mit einer Vielzahl von AWS Services wie Amazon S3, Auto Scaling, Amazon, Amazon DynamoDB CloudFront, Amazon, Amazon EC2, Elastic Load Balancing ElastiCache AWS Elastic Beanstalk, IAM OpsWorks, AWS und Amazon VPC arbeiten. Die aktuelle Liste der unterstützten Ressourcen finden Sie in der Referenz zu AWS Ressourcen- und Eigenschaftstypen.

AWS Serverless Application Model

Das AWS Serverless Application Model(AWS SAM) ist ein Open-Source-Framework, mit dem Sie serverlose Anwendungen erstellen können. AWS

AWS SAM lässt sich in andere AWS Dienste integrieren, sodass die Erstellung serverloser Anwendungen die folgenden AWS SAM Vorteile bietet:

- Konfiguration mit einer einzigen Bereitstellung — AWS SAM erleichtert die Organisation verwandter Komponenten und Ressourcen sowie den Betrieb auf einem einzigen Stack. Sie können AWS SAM damit Konfigurationen (wie Arbeitsspeicher und Timeouts) von mehreren Ressourcen gemeinsam nutzen und alle zugehörigen Ressourcen zusammen als eine einzige, versionierte Einheit bereitstellen.
- Erweiterung von CloudFormation — Da AWS SAM es sich um eine Erweiterung von handelt CloudFormation, erhalten Sie die zuverlässigen Bereitstellungsfunktionen von. CloudFormation

Sie können Ressourcen definieren, indem Sie sie CloudFormation in Ihrer AWS SAM Vorlage verwenden.

- Integrierte Best Practices — Sie können sie verwenden AWS SAM , um Ihr IaC zu definieren und bereitzustellen. Auf diese Weise können Sie bewährte Verfahren wie Code-Reviews anwenden und durchsetzen.

AWS Cloud Development Kit (AWS CDK)

Das AWS Cloud Development Kit (AWS CDK)ist ein Open-Source-Framework für die Softwareentwicklung, mit dem Sie Ihre Cloud-Anwendungsressourcen mithilfe vertrauter Programmiersprachen modellieren und bereitstellen können. AWS CDK ermöglicht es Ihnen, die Anwendungsinfrastruktur mithilfe von Python TypeScript, Java und .NET zu modellieren. Entwickler können ihre bestehende integrierte Entwicklungsumgebung (IDE) nutzen und Tools wie Autocomplete und Inline-Dokumentation verwenden, um die Entwicklung der Infrastruktur zu beschleunigen.

AWS CDK nutzt CloudFormation im Hintergrund, um Ressourcen auf sichere und wiederholbare Weise bereitzustellen. Konstrukte sind die Grundbausteine des CDK-Codes. Ein Konstrukt stellt eine Cloud-Komponente dar und kapselt alles, was zur Erstellung der Komponente CloudFormation benötigt wird. AWS CDK Dazu gehört die AWS Construct Library, die Konstrukte enthält, die viele AWS Services repräsentieren. Durch die Kombination von Konstrukten können Sie schnell und einfach komplexe Architekturen für die Bereitstellung in erstellen. AWS

AWS-Cloud-Entwicklungskit für Kubernetes

Das AWS Cloud Development Kit for Kubernetes ist ein Open-Source-Framework für die Softwareentwicklung zur Definition von Kubernetes-Anwendungen mithilfe allgemeiner Programmiersprachen.

Sobald Sie Ihre Anwendung in einer Programmiersprache definiert haben (zum Zeitpunkt dieser Veröffentlichung TypeScript werden nur Python und Python unterstützt), konvertiert cdk8s Ihre Anwendungsbeschreibung in eine frühere Version von Kubernetes YAML. Diese YAML-Datei kann dann von jedem Kubernetes-Cluster verwendet werden, der überall läuft. Da die Struktur in einer Programmiersprache definiert ist, können Sie die umfangreichen Funktionen der Programmiersprache nutzen. Sie können die Abstraktionsfunktion der Programmiersprache verwenden, um Ihren eigenen Standardcode zu erstellen und ihn in allen Bereitstellungen wiederzuverwenden.

AWS-Cloud-Entwicklungskit für Terraform

CDK for Terraform (CDKTF) basiert auf der Open-Source-JSII-Bibliothek und ermöglicht es Ihnen, Terraform-Konfigurationen in C#, Python, Java oder Go Ihrer Wahl zu schreiben und trotzdem vom gesamten Ökosystem der Terraform-Anbieter und -Module zu profitieren. TypeScript Sie können jeden vorhandenen Anbieter oder jedes Modul aus der Terraform Registry in Ihre Anwendung importieren, und CDKTF generiert Ressourcenklassen, mit denen Sie in Ihrer Zielprogrammiersprache interagieren können.

Mit CDKTF können Entwickler ihr IaC einrichten, ohne den Kontext von ihrer vertrauten Programmiersprache wechseln zu müssen. Dabei können sie dieselben Tools und dieselbe Syntax verwenden, um Infrastrukturressourcen bereitzustellen, die der Geschäftslogik der Anwendung ähneln. Teams können in vertrauter Syntax zusammenarbeiten und gleichzeitig die Leistungsfähigkeit des Terraform-Ökosystems nutzen und ihre Infrastrukturkonfigurationen über etablierte Terraform-Bereitstellungspipelines bereitstellen.

AWS -Cloud-Control- API

AWS -Cloud-Control- APIist eine neue AWS Funktion, die einen gemeinsamen Satz von Create, Read, Update, Delete and List (CRUDL) einführt, mit dem Entwickler ihre Cloud-Infrastruktur APIs auf einfache und konsistente Weise verwalten können. Die gemeinsame Cloud Control-API APIs ermöglicht es Entwicklern, den Lebenszyklus von AWS- und Drittanbieter-Services einheitlich zu verwalten.

Als Entwickler ziehen Sie es vielleicht vor, die Verwaltung des Lebenszyklus all Ihrer Ressourcen zu vereinfachen. Sie können das einheitliche Ressourcenkonfigurationsmodell der Cloud Control API mit einem vordefinierten Format verwenden, um Ihre Cloud-Ressourcenkonfiguration zu standardisieren. Darüber hinaus profitieren Sie bei der Verwaltung Ihrer Ressourcen von einem einheitlichen API-Verhalten (Antwortelemente und Fehler).

Sie werden es beispielsweise leicht finden, Fehler bei CRUDL-Vorgängen mithilfe einheitlicher Fehlercodes zu debuggen, die von der Cloud Control-API angezeigt werden und unabhängig von den Ressourcen sind, mit denen Sie arbeiten. Mit der Cloud Control API werden Sie es auch einfach finden, ressourcenübergreifende Abhängigkeiten zu konfigurieren. Außerdem müssen Sie keinen benutzerdefinierten Code mehr für Tools mehrerer Anbieter erstellen und verwalten und APIs Ressourcen von Drittanbietern gemeinsam verwenden AWS .

Automatisierung und Werkzeugbau

Eine weitere Kernphilosophie und -praxis von DevOps ist die Automatisierung. Die Automatisierung konzentriert sich auf die Einrichtung, Konfiguration, Bereitstellung und Unterstützung der Infrastruktur und der darauf ausgeführten Anwendungen. Mithilfe von Automatisierung können Sie Umgebungen schneller, standardisiert und wiederholbar einrichten. Der Wegfall manueller Prozesse ist der Schlüssel zu einer erfolgreichen DevOps Strategie. In der Vergangenheit waren Serverkonfiguration und Anwendungsbereitstellung überwiegend ein manueller Prozess. Umgebungen werden nicht mehr standardisiert, und es ist schwierig, eine Umgebung zu reproduzieren, wenn Probleme auftreten.

Der Einsatz von Automatisierung ist entscheidend, um die Vorteile der Cloud voll auszuschöpfen. Intern ist AWS stark auf Automatisierung angewiesen, um die Kernfunktionen Elastizität und Skalierbarkeit bereitzustellen.

Manuelle Prozesse sind fehleranfällig, unzuverlässig und unzureichend, um ein agiles Unternehmen zu unterstützen. Häufig kann ein Unternehmen hochqualifizierte Ressourcen für die manuelle Konfiguration einsetzen, obwohl die Zeit besser für die Unterstützung anderer, kritischerer und wertvollerer Aktivitäten innerhalb des Unternehmens genutzt werden könnte.

Moderne Betriebsumgebungen setzen in der Regel auf vollständige Automatisierung, um manuelle Eingriffe oder den Zugriff auf Produktionsumgebungen zu vermeiden. Dazu gehören alle Softwareveröffentlichungen, Maschinenkonfigurationen, Betriebssystem-Patches, Problembehebungen und Bugfixes. Viele Automatisierungsebenen können zusammen verwendet werden, um einen end-to-end automatisierten Prozess auf höherer Ebene bereitzustellen.

Die Automatisierung bietet die folgenden Hauptvorteile:

- Schnelle Änderungen
- Verbesserte Produktivität
- Wiederholbare Konfigurationen
- Reproduzierbare Umgebungen
- Elastizität
- Auto Scaling
- Automatisierte Tests

Automatisierung ist ein Eckpfeiler von AWS Services und wird intern in allen Diensten, Funktionen und Angeboten unterstützt.

Themen

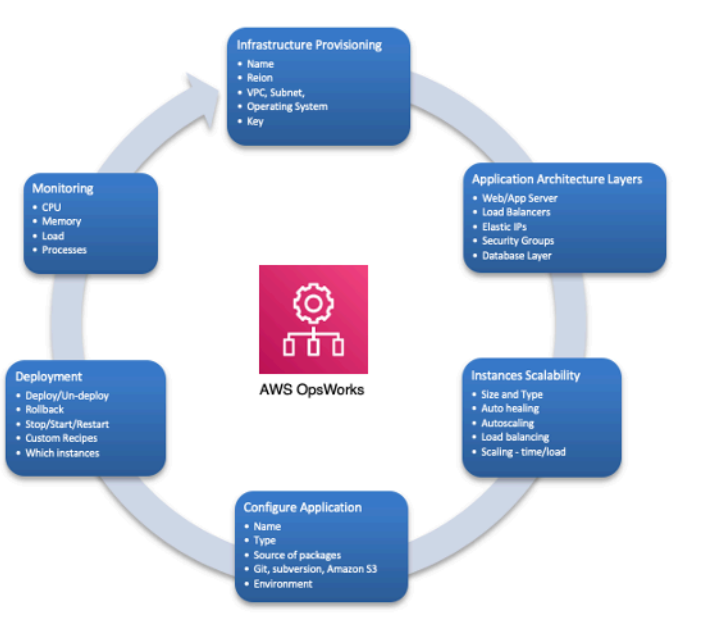
- AWS OpsWorks
- AWS Elastic Beanstalk
- EC2 Image Builder
- AWS Proton
- AWS Service Catalog
- AWS Cloud9
- AWS CloudShell
- Amazon CodeGuru

AWS OpsWorks

AWS OpsWorksgeht mit den Prinzipien von DevOps noch weiter als AWS Elastic Beanstalk. Es kann eher als Anwendungsverwaltungsdienst denn als einfacher Anwendungscontainer betrachtet werden. OpsWorks bietet noch mehr Automatisierungsgrade mit zusätzlichen Funktionen wie der Integration mit der Konfigurationsmanagement-Software (Chef) und dem Anwendungslebenszyklusmanagement. Sie können das Anwendungslebenszyklusmanagement verwenden, um zu definieren, wann Ressourcen eingerichtet, konfiguriert, bereitgestellt, nicht bereitgestellt oder beendet werden.

Für zusätzliche Flexibilität AWS OpsWorks können Sie Ihre Anwendung in konfigurierbaren Stacks definieren. Sie können auch vordefinierte Anwendungsstapel auswählen. Anwendungsstapel enthalten alle Bereitstellungen für AWS-Ressourcen, die Ihre Anwendung benötigt, einschließlich Anwendungsserver, Webserver, Datenbanken und Load Balancer.

Anwendungsstapel sind in Architekturebenen organisiert, sodass Stacks unabhängig voneinander verwaltet werden können. Zu den Beispielebenen könnten die Webebene, die Anwendungsebene und die Datenbankebene gehören. Standardmäßig vereinfacht AWS OpsWorks auch die Einrichtung von AWS Auto Scaling Scaling-Gruppen und Elastic Load Balancing (ELB) -Load Balancing (ELB) -Load Balancers, was das DevOps Prinzip der Automatisierung weiter veranschaulicht. Genau wie AWS Elastic Beanstalk OpsWorks unterstützt AWS die Versionsverwaltung von Anwendungen, die kontinuierliche Bereitstellung und das Infrastrukturkonfigurationsmanagement.



OpsWorks zeigt DevOps Merkmale und Architektur

AWS OpsWorks unterstützt auch die DevOps Methoden der Überwachung und Protokollierung (die im nächsten Abschnitt behandelt werden). Die Unterstützung bei der Überwachung wird von Amazon bereitgestellt CloudWatch. Alle Lebenszyklusereignisse werden protokolliert, und ein separates Chef-Protokoll dokumentiert alle ausgeführten Chef-Rezepte sowie alle Ausnahmen.

AWS Elastic Beanstalk

AWS Elastic Beanstalk ist ein Service zur schnellen Bereitstellung und Skalierung von Webanwendungen, die mit Java, .NET, PHP, Node.js, Python, Ruby, Go und Docker auf bekannten Servern wie Apache, NGINX, Passenger und IIS entwickelt wurden.

Elastic Beanstalk ist eine Abstraktion auf Amazon EC2, Auto Scaling, und vereinfacht die Bereitstellung durch zusätzliche Funktionen wie Klonen, blue/green Bereitstellungen, Elastic Beanstalk Command Line Interface (EB CLI) und Integration mit AWS Toolkit for Visual Studio, Visual Studio Code, Eclipse und IntelliJ zur Steigerung der Entwicklerproduktivität.

EC2 Image Builder

EC2 Image Builder ist ein vollständig verwalteter AWS Service, mit dem Sie die Erstellung, Wartung, Validierung, gemeinsame Nutzung und Bereitstellung von benutzerdefinierten, sicheren und benutzerdefinierten up-to-date Linux- oder Windows-AMIs automatisieren können. EC2 Image

Builder kann auch zum Erstellen von Container-Images verwendet werden. Sie können das AWS-Managementkonsole, das oder verwenden AWS CLI, APIs um benutzerdefinierte Images in Ihrem AWS Konto zu erstellen.

EC2 Image Builder reduziert den Aufwand für die Aufbewahrung up-to-date und Sicherheit von Bildern erheblich, da es eine einfache grafische Oberfläche, integrierte Automatisierung und AWS bereitgestellte Sicherheitseinstellungen bietet. Mit EC2 Image Builder gibt es keine manuellen Schritte zum Aktualisieren eines Images und Sie müssen auch keine eigene Automatisierungspipeline erstellen.

AWS Proton

AWS Protonermöglicht Plattformteams, all die verschiedenen Tools, die Ihre Entwicklungsteams für die Bereitstellung von Infrastruktur, Codebereitstellung, Überwachung und Updates benötigen, miteinander zu verbinden und zu koordinieren. AWS Proton ermöglicht automatisierte Infrastruktur als Codebereitstellung und Bereitstellung von serverlosen und containerbasierten Anwendungen.

AWS Proton ermöglicht Plattformteams, ihre Infrastruktur und Bereitstellungstools zu definieren, und bietet Entwicklern gleichzeitig ein Self-Service-Erlebnis, um Infrastruktur zu erhalten und Code bereitzustellen. Dadurch AWS Proton stellen Plattformteams gemeinsam genutzte Ressourcen bereit und definieren Anwendungs-Stacks, einschließlich CI/CD Pipelines und Observability-Tools. Anschließend können Sie verwalten, welche Infrastruktur- und Bereitstellungsfunktionen Entwicklern zur Verfügung stehen.

AWS Service Catalog

AWS Service Catalogermöglicht es Unternehmen, Kataloge mit IT-Services zu erstellen und zu verwalten, für AWS die sie zugelassen sind. Diese IT-Services können alles umfassen, von Images virtueller Maschinen, Servern, Software, Datenbanken und mehr bis hin zu kompletten mehrstufigen Anwendungsarchitekturen. AWS Service Catalog ermöglicht es Ihnen, bereitgestellte IT-Services, Anwendungen, Ressourcen und Metadaten zentral zu verwalten, um eine konsistente Verwaltung Ihrer IaC-Vorlagen zu erreichen.

Damit können Sie Ihre Compliance-Anforderungen erfüllen und gleichzeitig sicherstellen AWS Service Catalog, dass Ihre Kunden schnell die genehmigten IT-Services bereitstellen können, die sie benötigen. Endbenutzer können schnell nur die jeweils benötigten genehmigten IT-Services bereitstellen, wobei die Einschränkungen Ihrer Organisation berücksichtigt werden.

AWS Cloud9

AWS Cloud9ist eine cloudbasierte IDE, mit der Sie Ihren Code mit nur einem Browser schreiben, ausführen und debuggen können. Sie umfasst einen Code-Editor, einen Debugger und ein Terminal. AWS Cloud9 enthält wichtige Tools für beliebte Programmiersprachen wie Python JavaScript, PHP und mehr, sodass Sie keine Dateien installieren oder Ihren Entwicklungscomputer konfigurieren müssen, um neue Projekte zu starten. Da Ihre AWS Cloud9 IDE cloudbasiert ist, können Sie von Ihrem Büro, zu Hause oder von überall aus mit einem mit dem Internet verbundenen Computer an Ihren Projekten arbeiten.

AWS CloudShell

AWS CloudShellist eine browserbasierte Shell, die es einfacher macht, Ihre Ressourcen sicher zu verwalten, zu erkunden und mit ihnen zu interagieren. AWS AWS CloudShell ist mit Ihren Konsolenanmeldedaten vorab authentifiziert. Gängige Entwicklungs- und Betriebstools sind vorinstalliert, sodass Sie keine Software auf Ihrem lokalen Computer installieren oder konfigurieren müssen.

Amazon CodeGuru

Amazon CodeGuru ist ein Entwicklertool, das intelligente Empfehlungen zur Verbesserung der Codequalität und zur Identifizierung der teuersten Codezeilen einer Anwendung bietet. Integrieren Sie es CodeGuru in Ihren bestehenden Softwareentwicklungs-Workflow, um Codeprüfungen während der Anwendungsentwicklung zu automatisieren und die Leistung der Anwendung in der Produktion kontinuierlich zu überwachen und Empfehlungen und visuelle Hinweise zur Verbesserung der Codequalität und Anwendungsleistung sowie zur Senkung der Gesamtkosten bereitzustellen. CodeGuru besteht aus zwei Komponenten:

- Amazon CodeGuru Reviewer — Amazon CodeGuru Reviewer ist ein automatisierter Code-Review-Service, der kritische Fehler und Abweichungen von den bewährten Programmierpraktiken für Java- und Python-Code identifiziert. Es scannt die Codezeilen innerhalb einer Pull-Anfrage und bietet intelligente Empfehlungen, die auf Standards basieren, die aus großen Open-Source-Projekten sowie der Amazon-Codebasis gewonnen wurden.
- Amazon CodeGuru Profiler — Amazon CodeGuru Profiler analysiert das Laufzeitprofil der Anwendung und bietet intelligente Empfehlungen und Visualisierungen, die Entwicklern helfen, die Leistung der relevantesten Teile ihres Codes zu verbessern.

Überwachung und Beobachtbarkeit

Kommunikation und Zusammenarbeit sind in einer DevOps Philosophie von grundlegender Bedeutung. Um dies zu ermöglichen, ist Feedback von entscheidender Bedeutung. Dieses Feedback wird von unseren Überwachungs- und Beobachtungsdiensten bereitgestellt.

AWS bietet die folgenden Dienste für die Überwachung und Protokollierung:

Themen

- CloudWatch Amazon-Metriken
- CloudWatch Amazon-Alarme
- CloudWatch Amazon-Protokolle
- Amazon CloudWatch Logs Einblicke
- CloudWatch Amazon-Veranstaltungen
- Amazon EventBridge
- AWS CloudTrail
- DevOpsAmazon-Guru
- AWS X-Ray
- Amazon Managed Service für Prometheus
- Amazon Managed Grafana

CloudWatch Amazon-Metriken

CloudWatch Amazon-Metriken erfassen automatisch Daten von AWS Services wie EC2 Amazon-Instances, Amazon EBS-Volumes und Amazon RDS-Datenbank-Instances (DB). Diese Metriken können dann als Dashboards organisiert und Alarme oder Ereignisse erstellt werden, um Ereignisse auszulösen oder Auto Scaling Scaling-Aktionen auszuführen.

CloudWatch Amazon-Alarme

Sie können Alarme mithilfe von CloudWatch Amazon-Alarmen einrichten, die auf den von Amazon Metrics gesammelten CloudWatch Metriken basieren. Der Alarm kann dann eine Benachrichtigung an das Amazon SNS SNS-Thema senden oder Auto Scaling Scaling-Aktionen einleiten. Ein Alarm erfordert einen Zeitraum (Dauer der Auswertung einer Metrik), einen Evaluierungszeitraum (Anzahl

der neuesten Datenpunkte) und Datenpunkte bis zum Alarm (Anzahl der Datenpunkte innerhalb des Bewertungszeitraums).

CloudWatch Amazon-Protokolle

Amazon CloudWatch Logs ist ein Service zur Aggregation und Überwachung von Protokollen. AWS CodeBuild CodeCommit, CodeDeploy und CodePipeline bieten Integrationen mit CloudWatch Protokollen, sodass alle Protokolle zentral überwacht werden können. Darüber hinaus bieten die zuvor genannten Dienste verschiedene andere AWS Dienste eine direkte Integration CloudWatch.

Mit CloudWatch Logs können Sie:

- Fragen Sie Ihre Logdaten ab
- Überwachen Sie Protokolle von EC2 Amazon-Instances
- Überwachen Sie AWS CloudTrail protokollierte Ereignisse
- Definieren Sie eine Richtlinie zur Aufbewahrung von Protokollen

Amazon CloudWatch Logs Einblicke

Amazon CloudWatch Logs Insights scannt Ihre Protokolle und ermöglicht es Ihnen, interaktive Abfragen und Visualisierungen durchzuführen. Es versteht verschiedene Protokollformate und erkennt automatisch Felder aus JSON-Protokollen.

CloudWatch Amazon-Veranstaltungen

Amazon CloudWatch Events bietet einen Stream von Systemereignissen, die Änderungen an AWS Ressourcen beschreiben, nahezu in Echtzeit. Mit einfachen Regeln, die sich schnell einrichten lassen, können Sie Ereignisse ordnen und sie an eine oder mehrere Zielfunktionen oder Streams weiterleiten.

CloudWatch Events wird sich betrieblicher Änderungen bewusst, sobald sie eintreten. CloudWatch Events reagiert auf diese betrieblichen Änderungen und ergreift bei Bedarf Korrekturmaßnahmen, indem Nachrichten gesendet werden, um auf die Umgebung zu reagieren, Funktionen aktiviert, Änderungen vorgenommen und Statusinformationen erfasst werden.

Sie können Regeln in Amazon CloudWatch Events konfigurieren, um Sie über Änderungen an AWS Diensten zu informieren und diese Ereignisse mithilfe von Amazon in andere Systeme von

Drittanbietern zu integrieren EventBridge. Im Folgenden sind die AWS DevOps zugehörigen Dienste aufgeführt, die in CloudWatch Events integriert sind.

- Auto Scaling-Ereignisse für Anwendungen
- CodeBuild-Ereignisse
- CodeCommit-Ereignisse
- CodeDeploy --Ereignisse
- CodePipeline --Ereignisse

Amazon EventBridge

> **Note**
>
> Amazon CloudWatch Events und EventBridge sind derselbe zugrunde liegende Service und dieselbe API, EventBridge bieten jedoch mehr Funktionen.

Amazon EventBridge ist ein serverloser Event-Bus, der Integrationen zwischen AWS Services, Software as a Services (SaaS) und Ihren Anwendungen ermöglicht. Kann nicht nur zur Erstellung ereignisgesteuerter Anwendungen verwendet werden, sondern EventBridge kann auch verwendet werden, um über Ereignisse von Diensten wie CodeBuild, CodeDeploy CodePipeline, und zu informieren. CodeCommit

AWS CloudTrail

Um sich die DevOps Prinzipien Zusammenarbeit, Kommunikation und Transparenz zu eigen zu machen, ist es wichtig zu wissen, wer Änderungen an Ihrer Infrastruktur vornimmt. In AWS, diese Transparenz wird bereitgestellt von AWS CloudTrail. Alle AWS Interaktionen werden über AWS API-Aufrufe abgewickelt, die von überwacht und protokolliert werden AWS CloudTrail. Alle generierten Protokolldateien werden in einem Amazon S3 S3-Bucket gespeichert, den Sie definieren. Protokolldateien werden mit der serverseitigen Verschlüsselung (SSE) von Amazon S3 verschlüsselt. Alle API-Aufrufe werden protokolliert, unabhängig davon, ob sie direkt von einem Benutzer oder im Namen eines Benutzers von einem AWS Dienst kommen. Zahlreiche Gruppen können von CloudTrail Protokollen profitieren, darunter Betriebsteams für den Support, Sicherheitsteams für die Verwaltung und Finanzteams für die Abrechnung.

DevOpsAmazon-Guru

Amazon DevOps Guru ist ein auf maschinellem Lernen (ML) basierender Service, mit dem die Betriebsleistung und Verfügbarkeit einer Anwendung auf einfache Weise verbessert werden kann. DevOps Guru hilft dabei, Verhaltensweisen zu erkennen, die von normalen Betriebsmustern abweichen, sodass Sie betriebliche Probleme erkennen können, lange bevor sie sich auf Ihre Kunden auswirken.

DevOps Guru verwendet ML-Modelle, die auf jahrelanger Erfahrung von Amazon.com und AWS operativer Exzellenz basieren, um ungewöhnliches Anwendungsverhalten (z. B. erhöhte Latenz, Fehlerraten, Ressourcenengpässe usw.) zu identifizieren und kritische Probleme aufzudecken, die zu potenziellen Ausfällen oder Serviceunterbrechungen führen könnten.

Wenn DevOps Guru ein kritisches Problem identifiziert, spart es Zeit beim Debuggen, indem es relevante und spezifische Informationen aus einer Vielzahl von Datenquellen abruft und automatisch eine Warnung sendet und eine Zusammenfassung der damit verbundenen Anomalien sowie einen Kontext dafür bereitstellt, wann und wo das Problem aufgetreten ist.

AWS X-Ray

AWS X-Rayhilft Entwicklern dabei, verteilte Produktionsanwendungen zu analysieren und zu debuggen, z. B. solche, die mit einer Microservices-Architektur erstellt wurden. Mit X-Ray können Sie die Leistung Ihrer Anwendung und der zugrunde liegenden Dienste nachvollziehen, um die Hauptursache von Leistungsproblemen und Fehlern zu identifizieren und zu beheben. X-Ray bietet eine end-to-end Übersicht der Anfragen, während sie Ihre Anwendung durchlaufen, und zeigt eine Übersicht der Ihrer Anwendung zugrunde liegenden Komponenten. Mit X-Ray können Sie ganz einfach:

- Eine Service-Map erstellen — Durch die Nachverfolgung von Anfragen an Ihre Anwendungen kann X-Ray eine Übersicht der von Ihrer Anwendung verwendeten Services erstellen. Auf diese Weise erhalten Sie einen Überblick über die Verbindungen zwischen den Diensten in Ihrer Anwendung und können einen Abhängigkeitsbaum erstellen, Latenz oder Fehler bei der Arbeit in verschiedenen AWS Availability Zones oder Regionen erkennen, sich auf Dienste konzentrieren, die nicht wie erwartet funktionieren usw.
- Identifizieren Sie Fehler und Bugs — X-Ray kann automatisch Bugs oder Fehler in Ihrem Anwendungscode hervorheben, indem es den Antwortcode für jede Anfrage an Ihre Anwendung analysiert. Dies ermöglicht ein einfaches Debuggen von Anwendungscode, ohne dass Sie den Bug oder Fehler reproduzieren müssen.

- Erstellen Sie Ihre eigenen Analyse- und Visualisierungs-Apps — X-Ray bietet eine Reihe von Abfragen, mit denen APIs Sie Ihre eigenen Analyse- und Visualisierungs-Apps erstellen können, die die von X-Ray aufgezeichneten Daten verwenden.

Amazon Managed Service für Prometheus

Amazon Managed Service for Prometheus ist ein serverloser Überwachungsservice für Metriken, der mit dem Open-Source-Programm Prometheus kompatibel ist und es Ihnen erleichtert, Containerumgebungen sicher zu überwachen und Warnmeldungen zu senden. Amazon Managed Service for Prometheus reduziert den Aufwand für den Einstieg in die Überwachung von Anwendungen in Amazon Elastic Kubernetes Service, Amazon Elastic Container Service und AWS Fargate selbstverwalteten Kubernetes-Clustern.

Amazon Managed Grafana

Amazon Managed Grafana ist ein vollständig verwalteter Service mit umfangreichen, interaktiven Datenvisualisierungen, der Kunden dabei unterstützt, Metriken, Protokolle und Traces aus mehreren Datenquellen zu analysieren, zu überwachen und Warnmeldungen zu erstellen. Mit einem automatisch skalierten, hochverfügbaren und für Unternehmen sicheren Service können Sie interaktive Dashboards erstellen und diese mit allen Personen in Ihrem Unternehmen teilen.

Kommunikation und Zusammenarbeit

Unabhängig davon, ob Sie DevOps Kultur in Ihrem Unternehmen einführen oder einen DevOps kulturellen Wandel durchlaufen, sind Kommunikation und Zusammenarbeit ein wichtiger Bestandteil Ihres Ansatzes. Wir bei Amazon haben erkannt, dass die Denkweise unserer Teams geändert werden muss, und haben daher das Konzept der Two-Pizza-Teams übernommen.

„Wir versuchen, Teams zusammenzustellen, die nicht größer sind, als dass sie mit zwei Pizzen gefüttert werden können“, sagte Bezos. „Wir nennen das die Zwei-Pizza-Team-Regel.“

Je kleiner das Team, desto besser die Zusammenarbeit. Zusammenarbeit ist sehr wichtig, da Softwareversionen schneller denn je veröffentlicht werden. Und die Fähigkeit eines Teams, die Software bereitzustellen, kann für Ihr Unternehmen ein Unterscheidungsmerkmal gegenüber Ihrer Konkurrenz sein. Stellen Sie sich eine Situation vor, in der eine neue Produktfunktion veröffentlicht oder ein Fehler behoben werden muss. Sie möchten, dass dies so schnell wie möglich geschieht, damit Sie einen kürzeren go-to-market Zeitplan haben können. Sie möchten nicht, dass die Transformation ein langsamer Prozess ist. Sie wollen einen agilen Ansatz, bei dem Wellen von Änderungen beginnen, Wirkung zu zeigen.

Die Kommunikation zwischen den Teams ist auch wichtig, wenn Sie sich dem Modell der gemeinsamen Verantwortung zuwenden und den isolierten Entwicklungsansatz hinter sich lassen. Auf diese Weise wird das Konzept der Eigenverantwortung in das Team eingeführt und die Teammitglieder werden aus einer anderen Perspektive betrachtet, sodass sie den Prozess als ein end-to-end Projekt betrachten. Ihr Team sollte Ihre Produktionsumgebungen nicht als Blackboxen betrachten, in denen sie nicht sichtbar sind.

Kultureller Wandel ist ebenfalls wichtig, weil Sie vielleicht ein gemeinsames DevOps Team aufbauen oder ein engagiertes DevOps Mitglied in Ihrem Team haben. Mit beiden Ansätzen wird gemeinsame Verantwortung im Team eingeführt.

Sicherheit

Ganz gleich, ob Sie gerade eine DevOps Transformation durchmachen oder DevOps Prinzipien zum ersten Mal implementieren, Sie sollten die Sicherheit als integralen Bestandteil Ihrer DevOps Prozesse betrachten. Dies sollte in allen Phasen der Erstellung und der Testbereitstellung ein übergreifendes Anliegen sein.

Bevor wir uns eingehender mit DevOps der Sicherheit befassen AWS, befasst sich dieser paper mit dem Modell der AWS gemeinsamen Verantwortung.

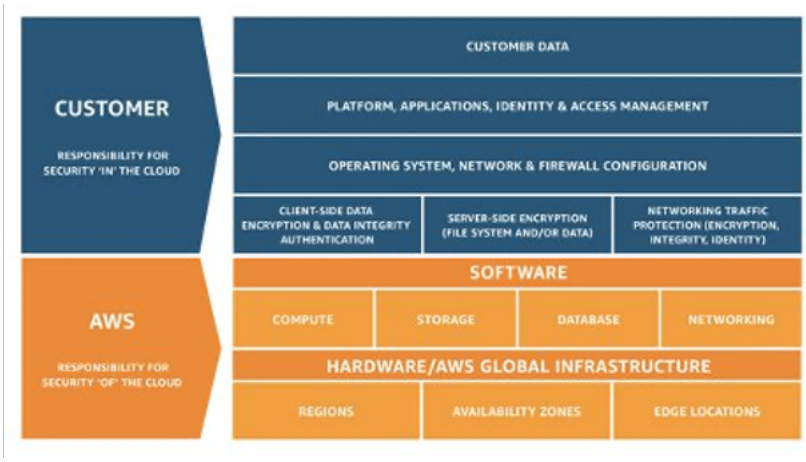
Themen

AWS Modell der geteilten Verantwortung

Sicherheit ist eine gemeinsame Verantwortung zwischen AWS und dem Kunden. Das Modell der geteilten Verantwortung besteht aus folgenden Teilen:

- AWS-Verantwortung „Sicherheit der Cloud“ — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der alle in der angebotenen Dienste ausgeführt werden AWS Cloud. Diese Infrastruktur besteht aus Hardware, Software, Netzwerken und Einrichtungen, die AWS Cloud Dienste ausführen.
- Kundenverantwortung „Sicherheit in der Cloud“ — Die Verantwortung des Kunden richtet sich nach den AWS Cloud Diensten, die ein Kunde auswählt. Dadurch wird der Umfang der Konfiguration bestimmt, die der Kunde im Rahmen seiner Sicherheitsverantwortung durchführen muss.

Dieses gemeinsame Modell kann dazu beitragen, den Kunden beim AWS Betrieb, der Verwaltung und der Kontrolle der Komponenten vom Host-Betriebssystem über die Virtualisierungsebene bis hin zur physischen Sicherheit der Einrichtungen, in denen der Service betrieben wird, zu entlasten. Dies ist in Fällen von entscheidender Bedeutung, in denen Kunden mehr über die Sicherheit ihrer Build-Umgebungen erfahren möchten.



AWS-Modell mit geteilter Verantwortung

Weisen Sie Berechtigungen auf der Grundlage des Berechtigungsmodells mit den geringsten Rechten zu. DevOps Dieses Modell besagt, dass „ein Benutzer (oder Dienst) genau die Zugriffsrechte haben sollte, die erforderlich sind, um die Aufgaben seiner Rolle zu erfüllen — nicht mehr und nicht weniger.“

Die Berechtigungen werden in IAM verwaltet. Sie können IAM verwenden, um zu kontrollieren, wer authentifiziert (angemeldet) und autorisiert ist (über Berechtigungen verfügt), um Ressourcen zu verwenden.

Identitäts- und Zugriffsverwaltung

AWS Identity and Access Management(IAM) definiert die Kontrollen und Richtlinien, die zur Verwaltung des Zugriffs auf Ressourcen verwendet werden. AWS Mit IAM können Sie Benutzer und Gruppen erstellen und Berechtigungen für verschiedene Dienste definieren. DevOps

Zusätzlich zu den Benutzern benötigen verschiedene Dienste möglicherweise auch Zugriff auf AWS Ressourcen. Beispielsweise benötigt Ihr CodeBuild Projekt möglicherweise Zugriff zum Speichern von Docker-Images in Amazon Elastic Container Registry (Amazon ECR) und Berechtigungen zum Schreiben in Amazon ECR. Diese Arten von Berechtigungen werden durch einen speziellen Rollentyp definiert, der als Servicerolle bezeichnet wird.

IAM ist eine Komponente der AWS Sicherheitsinfrastruktur. Mit IAM können Sie Gruppen, Benutzer, Servicerollen und Sicherheitsnachweise wie Passwörter, Zugriffsschlüssel und Berechtigungsrichtlinien zentral verwalten, die steuern, auf welche AWS-Services und -Ressourcen Benutzer zugreifen können. Mit IAM Policy können Sie den Satz von Berechtigungen definieren.

Diese Richtlinie kann dann entweder einer Rolle, einem Benutzer oder einem Dienst zugewiesen werden, um deren Berechtigungen zu definieren.

Sie können IAM auch verwenden, um Rollen zu erstellen, die im Rahmen Ihrer gewünschten DevOps Strategie häufig verwendet werden. In einigen Fällen kann es durchaus sinnvoll sein, die Berechtigungen programmgesteuert zu verwenden, AssumeRoleanstatt die Berechtigungen direkt abzurufen. Wenn ein Dienst oder Benutzer Rollen annimmt, erhält er temporäre Anmeldeinformationen für den Zugriff auf einen Dienst, auf den er normalerweise keinen Zugriff hat.

Schlussfolgerung

Um den Übergang zur Cloud reibungslos, effizient und effektiv zu gestalten, sollten Technologieunternehmen DevOps Prinzipien und Praktiken anwenden. Diese Prinzipien sind in AWS zahlreichen AWS Diensten verankert und bilden deren Eckpfeiler, insbesondere in den Bereitstellungs- und Überwachungsangeboten.

Definieren Sie zunächst Ihre Infrastruktur als Code, der den Dienst AWS CloudFormation oder AWS CDK verwendet. Definieren Sie als Nächstes mithilfe von Diensten wie,, und,,,,,,,,,,,,,,,, die Art und Weise AWS CodeBuild AWS CodeDeploy AWS CodePipeline, wie Ihre Anwendungen die kontinuierliche Bereitstellung nutzen AWS CodeCommit werden. Verwenden Sie auf Anwendungsebene Container wie AWS Elastic Beanstalk Amazon ECS oder Amazon Elastic Kubernetes Service (Amazon EKS). Wird verwendet OpsWorks , um die Konfiguration gängiger Architekturen zu vereinfachen. Die Verwendung dieser Dienste macht es auch einfach, andere wichtige Dienste wie Auto Scaling und Elastic Load Balancing einzubeziehen.

Verwenden Sie schließlich die DevOps Überwachungsstrategie wie Amazon CloudWatch und solide Sicherheitspraktiken wie IAM.

AWS Als Ihr Partner sorgen Ihre DevOps Prinzipien für mehr Agilität in Ihrer Geschäfts- und IT-Organisation und beschleunigen Ihren Weg in die Cloud.

Dokumentversionen

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

Änderung	Beschreibung	Datum
Aktualisiert	Aktualisiert	7. April 2023
Die Abschnitte wurden aktualisiert, um neue Dienste aufzunehmen	Die Abschnitte wurden um neue Dienste erweitert	16. Oktober 2020
Erste Veröffentlichung	Das Whitepaper wurde zuerst veröffentlicht	1. Dezember 2014

Mitwirkende

Zu den Mitwirkenden an diesem Dokument gehören:

- Abhra Sinha, Lösungsarchitektin
- Anil Nadiminti, Lösungsarchitekt
- Muhammad Mansoor, Lösungsarchitekt
- Ajit Zadgaonkar, weltweiter Technologieführer im Bereich Modernisierung
- Juan Lamadrid, Lösungsarchitekt
- Darren Ball, Lösungsarchitekt
- Rajeswari Malladi, Lösungsarchitekt
- Pallavi Nargund, Lösungsarchitekt
- Bert Zahniser, Lösungsarchitekt
- Abdullahi Olaoye, Architekt für Cloud-Lösungen
- Mohamed Kiswani, Manager für Softwareentwicklung
- Tara McCann, Managerin, Lösungsarchitektin

Hinweise

Kunden sind dafür verantwortlich, Ihre eigene unabhängige Bewertung der Informationen in diesem Dokument vorzunehmen. Dieses Dokument: (a) dient nur zu Informationszwecken, (b) stellt aktuelle AWS-Produktangebote und -praktiken dar, die ohne vorherige Ankündigung geändert werden können, und (c) stellt keine Verpflichtungen oder Zusicherungen von AWS und seinen verbundenen Unternehmen, Lieferanten oder Lizenzgebern dar. AWS-Produkte oder -Services werden „wie sie sind“ ohne ausdrückliche oder stillschweigende Garantien, Zusicherungen oder Bedingungen jeglicher Art bereitgestellt. Die Verantwortung und Haftung von AWS gegenüber seinen Kunden werden durch AWS-Vereinbarungen geregelt. Dieses Dokument gehört, weder ganz noch teilweise, nicht zu den Vereinbarung von AWS mit seinen Kunden und ändert diese Vereinbarungen auch nicht.

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