Livre blanc AWS

Présentation d' DevOps on AWS

Présentation d' DevOps on AWS: Livre blanc AWS

Table of Contents

Présentation d' DevOps on AWS

Date de publication : 7 avril 2023 (Révisions du document)

Aujourd'hui plus que jamais, les entreprises se lancent dans leur transformation numérique pour établir des liens plus étroits avec leurs clients, afin de créer une valeur commerciale durable et durable. Organisations de toutes formes et de toutes tailles révolutionnent leurs concurrents et pénètrent de nouveaux marchés en innovant plus rapidement que jamais. Pour ces organisations, il est important de se concentrer sur l'innovation et la disruption logicielle, d'où la nécessité de rationaliser la fourniture de leurs logiciels. Organisations qui raccourcissent le délai entre l'idée et la production en faisant de la rapidité et de l'agilité une priorité pourraient être les disrupteurs de demain.

Bien que plusieurs facteurs soient à prendre en compte pour devenir le prochain disrupteur numérique, ce livre blanc se concentre sur DevOps les services et fonctionnalités de la plateforme Amazon Web Services (AWS) qui aideront une entreprise à accroître sa capacité à fournir des applications et des services à grande vitesse.

Introduction

DevOps est la combinaison de philosophies culturelles, de pratiques d'ingénierie et d'outils qui augmentent la capacité d'une organisation à fournir des applications et des services à grande vitesse et de meilleure qualité. Au fil du temps, plusieurs pratiques essentielles sont apparues lors de l'adoption DevOps : l'intégration continue (CI), la livraison continue (CD), l'infrastructure en tant que code (iAc), ainsi que la surveillance et la journalisation.

Ce paper met en lumière AWS les fonctionnalités qui vous aident à accélérer votre DevOps parcours et la manière dont les AWS services peuvent vous aider à éliminer les charges lourdes indifférenciées associées à DevOps l'adaptation. Il décrit également comment créer une capacité d'intégration et de diffusion continues sans gérer de serveurs ni créer de nœuds, et comment utiliser IaC pour provisionner et gérer vos ressources cloud de manière cohérente et reproductible.

- Intégration continue : pratique de développement logiciel dans laquelle les développeurs fusionnent régulièrement leurs modifications de code dans un référentiel central, après quoi des builds et des tests automatisés sont exécutés.
- Livraison continue : pratique de développement logiciel dans laquelle les modifications de code sont automatiquement créées, testées et préparées pour une mise en production.

- Infrastructure en tant que code : pratique dans laquelle l'infrastructure est provisionnée et gérée à l'aide de techniques de développement de code et de logiciels, telles que le contrôle de version et l'intégration continue.
- Surveillance et journalisation : permet aux entreprises de voir l'impact des performances des applications et de l'infrastructure sur l'expérience de l'utilisateur final de leurs produits.
- Communication et collaboration : des pratiques sont établies pour rapprocher les équipes, en créant des flux de travail et en répartissant les responsabilités en la matière DevOps.
- Sécurité : Cela devrait être une préoccupation transversale. Vos pipelines d'intégration continue et de livraison continue (CI/CD) et les services associés doivent être protégés et des autorisations de contrôle d'accès appropriées doivent être mises en place.

L'examen de chacun de ces principes révèle un lien étroit avec les offres disponibles auprès de AWS.

Êtes-vous Well-Architected ?

L'AWS Well-Architected Framework vous aide à comprendre les avantages et les inconvénients des décisions que vous prenez lors de la création de systèmes dans le cloud. Les six piliers du Framework vous permettent d'apprendre les meilleures pratiques architecturales pour concevoir et exploiter des systèmes fiables, sécurisés, efficaces, rentables et durables. À l'aide de l'outil AWS Well-Architected, disponible gratuitement dans l'AWS Management Console, vous pouvez évaluer vos charges de travail par rapport à ces meilleures pratiques en répondant à une série de questions pour chaque pilier.

Intégration continue

L'intégration continue (CI) est une pratique de développement logiciel dans laquelle les développeurs fusionnent régulièrement leurs modifications de code dans un référentiel de code central, après quoi des builds et des tests automatisés sont exécutés. CI permet de détecter et de corriger les bogues plus rapidement, d'améliorer la qualité des logiciels et de réduire le temps nécessaire à la validation et à la publication de nouvelles mises à jour logicielles.

AWS propose les services suivants pour une intégration continue :

Rubriques

- AWS CodeCommit
- AWS CodeBuild
- AWS CodeArtifact

AWS CodeCommit

AWS CodeCommitest un service de contrôle de source géré, sécurisé et hautement évolutif qui héberge des référentiels git privés. CodeCommit vous n'avez plus besoin d'exploiter votre propre système de contrôle de source et il n'y a aucun matériel à provisionner et à dimensionner, ni aucun logiciel à installer, configurer et exploiter. Vous pouvez l'utiliser CodeCommit pour stocker n'importe quoi, du code aux fichiers binaires, et il prend en charge les fonctionnalités standard de GitHub, ce qui lui permet de fonctionner parfaitement avec vos outils Git existants. Votre équipe peut également utiliser les outils CodeCommit de code en ligne pour parcourir, modifier et collaborer sur des projets. AWS CodeCommit présente plusieurs avantages :

- Collaboration — AWS CodeCommit est conçu pour le développement collaboratif de logiciels. Vous pouvez facilement valider, bifurquer et fusionner votre code, ce qui vous permet de garder facilement le contrôle des projets de votre équipe. CodeCommit prend également en charge les pull requests, qui fournissent un mécanisme permettant de demander des révisions de code et de discuter du code avec des collaborateurs.
- Chiffrement — Vous pouvez transférer vos fichiers depuis et vers le AWS CodeCommit protocole HTTPS ou SSH, selon vos préférences. Vos référentiels sont également automatiquement chiffrés au repos via AWS Key Management Service(AWS KMS) à l'aide de clés spécifiques au client.
- Contrôle d'accès : AWS CodeCommit utilise Gestion des identités et des accès AWS(IAM) pour contrôler et surveiller qui peut accéder à vos données, ainsi que comment, quand et où ils

peuvent y accéder. CodeCommit vous permet également de surveiller vos référentiels via AWS CloudTrailAmazon CloudWatch.

Haute disponibilité et durabilité : AWS CodeCommit stocke vos référentiels dans Amazon Simple Storage Service (Amazon S3) et Amazon DynamoDB. Vos données cryptées sont stockées de manière redondante sur plusieurs sites. Cette architecture augmente la disponibilité et la durabilité des données de votre référentiel.

- Notifications et scripts personnalisés : vous pouvez désormais recevoir des notifications pour les événements ayant un impact sur vos référentiels. Les notifications seront transmises sous forme de notifications Amazon Simple Notification Service (Amazon SNS). Chaque notification comprendra un message d'état ainsi qu'un lien vers les ressources dont l'événement a généré cette notification. En outre, à l'aide d'indices AWS CodeCommit du référentiel, vous pouvez envoyer des notifications et créer des webhooks HTTP avec Amazon SNS ou AWS Lambdainvoquer des fonctions en réponse aux événements du référentiel que vous choisissez.

AWS CodeBuild

AWS CodeBuildest un service d'intégration continue entièrement géré qui compile le code source, exécute des tests et produit des progiciels prêts à être déployés. Vous n'avez pas besoin de provisionner, de gérer et de dimensionner vos propres serveurs de construction. CodeBuild peut utiliser GitHub Enterprise ou Amazon S3 comme fournisseur de source. GitHub BitBucket AWS CodeCommit

CodeBuild évolue en continu et peut traiter plusieurs builds simultanément. CodeBuild propose différents environnements préconfigurés pour différentes versions de Microsoft Windows et Linux. Les clients peuvent également apporter leurs environnements de construction personnalisés sous forme de conteneurs Docker. CodeBuild s'intègre également à des outils open source tels que Jenkins et Spinnaker.

CodeBuild peut également créer des rapports pour des tests unitaires, fonctionnels ou d'intégration. Ces rapports fournissent une vue visuelle du nombre de cas de tests exécutés et du nombre de tests réussis ou échoués. Le processus de création peut également être exécuté dans un Amazon Virtual Private Cloud (Amazon VPC), ce qui peut être utile si vos services d'intégration ou vos bases de données sont déployés dans un VPC.

AWS CodeArtifact

AWS CodeArtifactest un service de référentiel d'artefacts entièrement géré qui peut être utilisé par les entreprises pour stocker, publier et partager en toute sécurité les progiciels utilisés dans leur processus de développement logiciel. CodeArtifact peut être configuré pour récupérer automatiquement les packages logiciels et les dépendances des référentiels d'artefacts publics afin que les développeurs aient accès aux dernières versions.

Les équipes de développement de logiciels s'appuient de plus en plus sur des packages open source pour effectuer les tâches courantes de leur package d'applications. Il est devenu essentiel pour les équipes de développement de logiciels de garder le contrôle sur une version particulière du logiciel open source afin de s'assurer que le logiciel est exempt de vulnérabilités. Avec CodeArtifact, vous pouvez configurer des contrôles pour faire appliquer cela.

CodeArtifact fonctionne avec les gestionnaires de packages et les outils de création couramment utilisés tels que Maven, Gradle, npm, yarn, twine et pip, ce qui facilite son intégration dans les flux de travail de développement existants.

Livraison continue

La livraison continue (CD) est une pratique de développement logiciel dans laquelle les modifications de code sont automatiquement préparées pour une mise en production. Pilier du développement d'applications modernes, la livraison continue s'appuie sur l'intégration continue en déployant toutes les modifications de code dans un environnement de test et/ou un environnement de production après la phase de création. Lorsqu'il est correctement implémenté, les développeurs disposeront toujours d'un artefact de build prêt à être déployé qui a passé un processus de test standardisé.

La livraison continue permet aux développeurs d'automatiser les tests au-delà des tests unitaires, afin de vérifier les mises à jour des applications dans plusieurs dimensions avant de les déployer chez les clients.

Ces tests peuvent inclure des tests d'interface utilisateur, des tests de charge, des tests d'intégration, des tests de fiabilité des API, etc. Cela permet aux développeurs de valider les mises à jour de manière plus approfondie et de détecter les problèmes de manière préventive. Grâce au cloud, il est facile et rentable d'automatiser la création et la réplication de plusieurs environnements à des fins de test, ce qui était auparavant difficile à réaliser sur site.

AWS propose les services suivants pour une livraison continue :

- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline

Rubriques

- AWS CodeDeploy
- AWS CodePipeline

AWS CodeDeploy

AWS CodeDeployest un service de déploiement entièrement géré qui automatise les déploiements de logiciels vers divers services informatiques tels qu'Amazon Elastic Compute Cloud EC2 (Amazon) et vos AWS Lambda serveurs sur site. AWS Fargate AWS CodeDeploy vous permet de publier rapidement de nouvelles fonctionnalités, d'éviter les temps d'arrêt lors du déploiement des

applications et de gérer la complexité de la mise à jour de vos applications. Vous pouvez l'utiliser CodeDeploy pour automatiser les déploiements de logiciels, réduisant ainsi le besoin d'opérations manuelles sujettes aux erreurs. Le service s'adapte à vos besoins de déploiement.

CodeDeploy présente plusieurs avantages conformes au DevOps principe du déploiement continu :

- Déploiements automatisés : automatise CodeDeploy entièrement les déploiements de logiciels, ce qui vous permet de les déployer de manière fiable et rapide.
- Contrôle centralisé : vous CodeDeploy permet de lancer et de suivre facilement l'état de vos déploiements d'applications via le AWS Management Console ou le AWS CLI. CodeDeployvous fournit un rapport détaillé vous permettant de voir quand et où chaque révision de l'application a été déployée. Vous pouvez également créer des notifications push pour recevoir des mises à jour en direct sur vos déploiements.
- Minimisez les temps d'arrêt : CodeDeploy permet de maximiser la disponibilité de vos applications pendant le processus de déploiement du logiciel. Il introduit les modifications de manière incrémentielle et suit l'état de santé des applications conformément à des règles configurables. Les déploiements de logiciels peuvent facilement être arrêtés et annulés en cas d'erreur.
- Facile à adopter : CodeDeploy fonctionne avec n'importe quelle application et offre la même expérience sur différentes plateformes et langues. Vous pouvez facilement réutiliser votre code de configuration existant. CodeDeploy peut également s'intégrer à votre processus de publication logicielle existant ou à votre chaîne d'outils de livraison continue (par exemple AWS CodePipeline GitHub, Jenkins).

AWS CodeDeploy prend en charge plusieurs options de déploiement. Pour plus d'informations, reportez-vous à la section Stratégies de déploiement de ce document.

AWS CodePipeline

AWS CodePipelineest un service de livraison continue que vous pouvez utiliser pour modéliser, visualiser et automatiser les étapes nécessaires à la publication de votre logiciel. Avec AWS CodePipeline, vous modélisez le processus de publication complet pour créer votre code, le déployer dans des environnements de pré-production, tester votre application et la mettre en production. AWS CodePipeline construit, teste et déploie ensuite votre application conformément au flux de travail défini chaque fois qu'un changement de code est effectué. Vous pouvez intégrer les outils des partenaires et vos propres outils personnalisés à n'importe quelle étape du processus de publication afin de créer une solution de livraison end-to-end continue.

AWS CodePipeline présente plusieurs avantages conformes au DevOps principe du déploiement continu :

- Livraison rapide : AWS CodePipeline automatise le processus de publication de vos logiciels, ce qui vous permet de proposer rapidement de nouvelles fonctionnalités à vos utilisateurs. Vous pouvez ainsi réagir rapidement aux commentaires et proposer de nouvelles fonctionnalités à vos utilisateurs plus rapidement. CodePipeline
- Qualité améliorée : en automatisant vos processus de création, de test et de publication, vous pouvez AWS CodePipeline augmenter la vitesse et la qualité de vos mises à jour logicielles en exécutant toutes les nouvelles modifications par le biais d'un ensemble cohérent de contrôles de qualité.
- Facile à intégrer : AWS CodePipeline peut facilement être étendu pour s'adapter à vos besoins spécifiques. Vous pouvez utiliser les plugins prédéfinis ou vos propres plugins personnalisés à n'importe quelle étape de votre processus de publication. Par exemple, vous pouvez extraire votre code source GitHub, utiliser votre serveur de build Jenkins sur site, exécuter des tests de charge à l'aide d'un service tiers ou transmettre des informations de déploiement à votre tableau de bord des opérations personnalisé.
- Flux de travail configurable : vous AWS CodePipeline permet de modéliser les différentes étapes du processus de publication de votre logiciel à l'aide de l'interface de console AWS CLICloudFormation, du, ou d'AWS SDKs. Vous pouvez facilement spécifier les tests à exécuter et personnaliser les étapes de déploiement de votre application et de ses dépendances.

Stratégies de déploiement

Les stratégies de déploiement définissent la manière dont vous souhaitez fournir votre logiciel. Organisations suivent différentes stratégies de déploiement en fonction de leur modèle commercial. Certains choisissent de fournir des logiciels entièrement testés, tandis que d'autres souhaitent que leurs utilisateurs fournissent des commentaires et les laissent évaluer les fonctionnalités en cours de développement (telles que les versions bêta). La section suivante décrit les différentes stratégies de déploiement.

Déploiements sur place

Dans cette stratégie, la version précédente de l'application sur chaque ressource de calcul est arrêtée, la dernière application est installée et la nouvelle version de l'application est démarrée et validée. Cela permet aux déploiements d'applications de se dérouler en perturbant le moins possible l'infrastructure sous-jacente. Avec un déploiement sur place, vous pouvez déployer votre application sans créer de nouvelle infrastructure ; toutefois, la disponibilité de votre application peut être affectée lors de ces déploiements. Cette approche minimise également les coûts d'infrastructure et les frais de gestion associés à la création de nouvelles ressources. Vous pouvez utiliser un équilibreur de charge afin que chaque instance soit désenregistrée pendant son déploiement, puis remise en service une fois le déploiement terminé. Les déploiements sur place peuvent être effectués all-at-once, en supposant une panne de service, ou sous forme de mise à jour continue. AWS CodeDeploy et AWS Elastic Beanstalk proposent des configurations de déploiement one-at-a-time pour, et. half-at-a-time all-at-once

Déploiement bleu/vert

Le déploiement bleu/vert, parfois appelé déploiement, vous aide à minimiser les red/black deployment, is a technique for releasing applications by shifting traffic between two identical environments running differing versions of the application. Blue/green temps d'arrêt lors des mises à jour des applications, en atténuant les risques liés aux temps d'arrêt et aux fonctionnalités de restauration.

Les déploiements bleu/vert vous permettent de lancer une nouvelle version (verte) de votre application parallèlement à l'ancienne version (bleue), de surveiller et de tester la nouvelle version avant de rediriger le trafic vers celle-ci, en revenant à la détection des problèmes.

Déploiement Canary

L'objectif d'un déploiement Canary est de réduire le risque lié au déploiement d'une nouvelle version ayant un impact sur la charge de travail. La méthode déploiera progressivement la nouvelle version, la rendant visible aux nouveaux utilisateurs de manière lente. Au fur et à mesure que vous aurez confiance dans le déploiement, vous le déploierez pour remplacer la version actuelle dans son intégralité.

Déploiement linéaire

Le déploiement linéaire signifie que le trafic est déplacé par incréments égaux, avec un nombre égal de minutes entre chaque incrément. Vous pouvez choisir parmi les options linéaires prédéfinies qui définissent le pourcentage de trafic déplacé pour chaque incrément et le nombre de minutes entre chaque incrément.

All-at-once déploiement

All-at-oncele déploiement signifie que tout le trafic est transféré de l'environnement d'origine vers l'environnement de remplacement en une seule fois.

Matrice des stratégies de déploiement

La matrice suivante répertorie les stratégies de déploiement prises en charge pour Amazon Elastic Container Service (Amazon ECS) et Amazon EC2 /on-premise. AWS Lambda

- Amazon ECS est un service d'orchestration entièrement géré.
- AWS Lambda vous permet d'exécuter du code sans provisionner ni gérer de serveurs.
- Amazon vous EC2 permet d'exécuter une capacité de calcul sécurisée et redimensionnable dans le cloud.

Stratégie de déploiement	Amazon ECS	AWS Lambda	EC2Amazon/sur site	
Sur place	✓	✓	✓	
Bleu/vert	✓	✓	✓*	
Canary	✓	✓	X	
Linéaire	✓	✓	X	
UN ll-at-once	✓	✓	X	

> **Note**
>
> Blue/green deployment with EC2/on-premises ne fonctionne qu'avec les EC2 instances.

AWS Elastic Beanstalk stratégies de déploiement

AWS Elastic Beanstalkprend en charge les types de stratégies de déploiement suivants :

- A ll-at-once Effectue le déploiement sur place sur toutes les instances.
- Rolling Divise les instances en lots et les déploie sur un lot à la fois.
- Utiliser un lot supplémentaire Divise les déploiements en lots, mais pour le premier lot, il crée de nouvelles EC2 instances au lieu de les déployer sur les instances existantes. EC2

- Immuable Si vous devez effectuer un déploiement avec une nouvelle instance au lieu d'utiliser une instance existante.
- Répartition du trafic Effectue un déploiement immuable, puis transmet le pourcentage du trafic aux nouvelles instances pendant une durée prédéterminée. Si les instances restent saines, transférez tout le trafic vers les nouvelles instances et arrêtez les anciennes instances.

Infrastructure en tant que code

L'un des principes fondamentaux DevOps est de traiter l'infrastructure de la même manière que les développeurs traitent le code. Le code de l'application possède un format et une syntaxe définis. Si le code n'est pas écrit conformément aux règles du langage de programmation, les applications ne peuvent pas être créées. Le code est stocké dans un système de gestion des versions ou de contrôle de source qui enregistre l'historique du développement du code, des modifications et des corrections de bogues. Lorsque le code est compilé ou intégré à des applications, nous nous attendons à ce qu'une application cohérente soit créée, et que la compilation soit reproductible et fiable.

Pratiquer l'infrastructure en tant que code signifie appliquer la même rigueur au développement du code d'application au provisionnement de l'infrastructure. Toutes les configurations doivent être définies de manière déclarative et stockées dans un système de contrôle de source tel que AWS CodeCommitle code d'application. Le provisionnement, l'orchestration et le déploiement de l'infrastructure doivent également prendre en charge l'utilisation de l'infrastructure en tant que code.

L'infrastructure était traditionnellement provisionnée à l'aide d'une combinaison de scripts et de processus manuels. Parfois, ces scripts étaient stockés dans des systèmes de contrôle de version ou documentés étape par étape dans des fichiers texte ou des livres d'exécution. Souvent, la personne qui écrit les livres d'exécution n'est pas la même personne qui exécute ces scripts ou qui suit les livres d'exécution. Si ces scripts ou runbooks ne sont pas fréquemment mis à jour, ils peuvent devenir un obstacle lors des déploiements. Cela se traduit par la création de nouveaux environnements qui ne sont pas toujours reproductibles, fiables ou cohérents.

En revanche, AWS fournit un moyen DevOps ciblé de créer et de maintenir une infrastructure. De la même manière que les développeurs de logiciels écrivent le code des applications, AWS fournit des services qui permettent la création, le déploiement et la maintenance de l'infrastructure de manière programmatique, descriptive et déclarative. Ces services apportent rigueur, clarté et fiabilité. Les AWS services décrits dans ce paper sont au cœur d'une DevOps méthodologie et constituent le fondement de nombreux principes et pratiques de haut niveau AWS DevOps .

AWS propose les services suivants pour définir l'infrastructure en tant que code.

Services

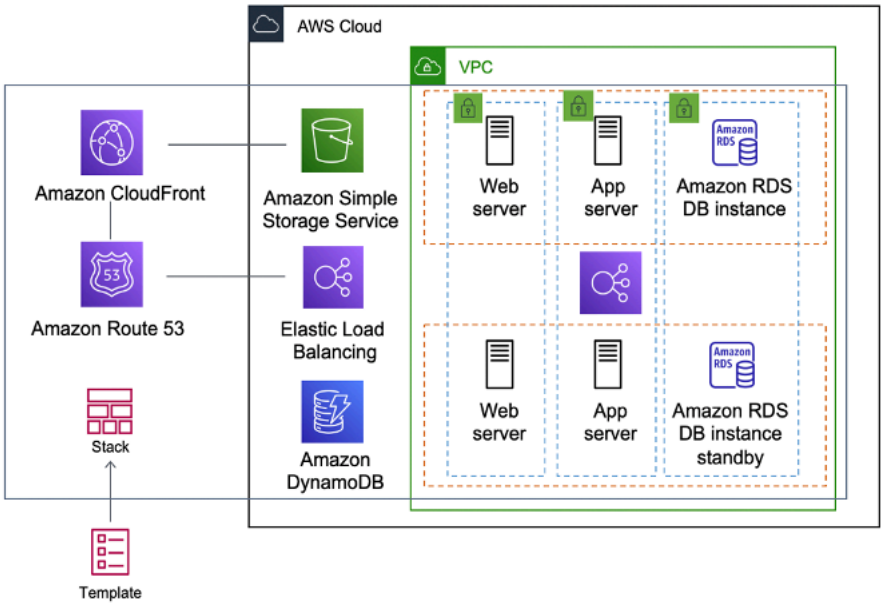
- CloudFormation
- AWS Serverless Application Model
- AWS Cloud Development Kit (AWS CDK)

- Kit de développement cloud AWS pour Kubernetes
- Kit de développement cloud AWS pour Terraform
- API de commande du Cloud AWS

CloudFormation

AWS CloudFormation est un service qui permet aux développeurs de créer AWS des ressources de manière ordonnée et prévisible. Les ressources sont écrites dans des fichiers texte au format JSON ou YAML. Les modèles nécessitent une syntaxe et une structure spécifiques, qui dépendent des types de ressources créées et gérées. Vous créez vos ressources au format JSON ou YAML avec n'importe quel éditeur de code, par exemple AWS Cloud9, vous les intégrez dans un système de contrôle de version, puis CloudFormation vous créez les services spécifiés de manière sûre et reproductible.

Un CloudFormation modèle est déployé dans l' AWS environnement sous forme de pile. Vous pouvez gérer les piles via le AWS Management Console AWS Command Line Interface, ou CloudFormation APIs. Si vous devez apporter des modifications aux ressources en cours d'exécution d'une pile, vous devez mettre à jour la pile. Avant d'apporter des modifications à vos ressources, vous pouvez générer un jeu de modifications, qui représente un résumé de ces modifications. Les ensembles de modifications vous permettent de voir comment vos modifications peuvent avoir un impact sur vos ressources courantes, en particulier pour les ressources critiques, avant de les mettre en œuvre.



AWS CloudFormation création d'un environnement complet (pile) à partir d'un modèle

Vous pouvez utiliser un modèle unique pour créer et mettre à jour un environnement complet, ou des modèles distincts pour gérer plusieurs couches au sein d'un environnement. Cela permet de moduler les modèles et fournit également une couche de gouvernance importante pour de nombreuses organisations.

Lorsque vous créez ou mettez à jour une pile dans la CloudFormation console, des événements s'affichent pour indiquer l'état de la configuration. En cas d'erreur, la pile est rétablie par défaut à son état précédent. Amazon SNS fournit des notifications sur les événements. Par exemple, vous pouvez utiliser Amazon SNS pour suivre la progression de la création et de la suppression de piles par e-mail et pour intégrer d'autres processus par programmation.

AWS CloudFormation facilite l'organisation et le déploiement d'un ensemble de AWS ressources et vous permet de décrire les dépendances éventuelles ou de transmettre des paramètres spéciaux lors de la configuration de la pile.

CloudFormation Les modèles vous permettent de travailler avec un large éventail de AWS services, tels qu'Amazon S3, Auto Scaling, Amazon CloudFront, Amazon DynamoDB, Amazon, EC2 ElastiCache AWS Elastic Beanstalk Amazon, Elastic Load Balancing, IAM, OpsWorks AWS et Amazon VPC. Pour obtenir la liste la plus récente des ressources prises en charge, reportez-vous à la référence des types de AWS ressources et de propriétés.

AWS Serverless Application Model

Le AWS Serverless Application Model (AWS SAM) est un cadre open source que vous pouvez utiliser pour construire des applications sans serveur sur AWS.

AWS SAM s'intègre à d'autres AWS services, de sorte que la création d'applications sans serveur AWS SAM offre les avantages suivants :

- Configuration à déploiement unique : AWS SAM permet d'organiser facilement les composants et les ressources connexes et de fonctionner sur une seule pile. Vous pouvez l'utiliser AWS SAM pour partager la configuration (telle que la mémoire et les délais d'expiration) entre les ressources et déployer toutes les ressources associées ensemble sous la forme d'une seule entité versionnée.
- Extension de CloudFormation — Comme il AWS SAM s'agit d'une extension de CloudFormation, vous bénéficiez des capacités de déploiement fiables de CloudFormation. Vous pouvez définir des ressources en les utilisant CloudFormation dans votre AWS SAM modèle.
- Bonnes pratiques intégrées : vous pouvez les utiliser AWS SAM pour définir et déployer votre iAc. Cela permet d'utiliser et d'appliquer les bonnes pratiques telles que les révisions de code.

AWS Cloud Development Kit (AWS CDK)

Il s'AWS Cloud Development Kit (AWS CDK)agit d'un framework de développement de logiciels open source permettant de modéliser et de provisionner les ressources de vos applications cloud à l'aide de langages de programmation familiers. AWS CDK vous permet de modéliser l'infrastructure d'applications à l'aide de Python TypeScript, Java et .NET. Les développeurs peuvent tirer parti de leur environnement de développement intégré (IDE) existant, en utilisant des outils tels que la saisie semi-automatique et la documentation en ligne pour accélérer le développement de l'infrastructure.

AWS CDK utilise CloudFormation en arrière-plan pour fournir des ressources de manière sûre et reproductible. Les constructions sont les éléments de base du code CDK. Une construction représente un composant cloud et encapsule tout ce qui est CloudFormation nécessaire pour créer le composant. AWS CDK Cela inclut la bibliothèque AWS Construct, qui contient des constructions représentant de nombreux AWS services. En combinant des structures, vous pouvez créer rapidement et facilement des architectures complexes dans AWS lesquelles vous pourrez les déployer.

Kit de développement cloud AWS pour Kubernetes

AWS Cloud Development Kit for Kubernetes est un framework de développement logiciel open source permettant de définir des applications Kubernetes à l'aide de langages de programmation polyvalents.

Une fois que vous avez défini votre application dans un langage de programmation (à la date de cette publication, seul Python est pris en charge), cdk8s convertira la description de votre application en YAML antérieur à TypeScript Kubernetes. Ce fichier YAML peut ensuite être consommé par n'importe quel cluster Kubernetes s'exécutant n'importe où. La structure étant définie dans un langage de programmation, vous pouvez utiliser les fonctionnalités riches fournies par ce langage de programmation. Vous pouvez utiliser la fonctionnalité d'abstraction du langage de programmation pour créer votre propre code standard et le réutiliser dans tous les déploiements.

Kit de développement cloud AWS pour Terraform

Construit sur la base de la bibliothèque open source JSII, CDK for Terraform (CDKTF) vous permet d'écrire des configurations Terraform dans le langage C#, Python TypeScript, Java ou Go de votre choix tout en bénéficiant de l'écosystème complet des fournisseurs et modules Terraform. Vous pouvez importer n'importe quel fournisseur ou module existant depuis le registre Terraform dans

votre application, et CDKTF générera des classes de ressources avec lesquelles vous pourrez interagir dans votre langage de programmation cible.

Avec CDKTF, les développeurs peuvent configurer leur IaC sans changer de contexte depuis leur langage de programmation habituel, en utilisant les mêmes outils et la même syntaxe pour fournir des ressources d'infrastructure similaires à la logique métier des applications. Les équipes peuvent collaborer selon une syntaxe familière, tout en utilisant la puissance de l'écosystème Terraform et en déployant leurs configurations d'infrastructure via des pipelines de déploiement Terraform établis.

API de commande du Cloud AWS

API de commande du Cloud AWSest une nouvelle AWS fonctionnalité qui introduit un ensemble commun de fonctionnalités de création, de lecture, de mise à jour, de suppression et de liste (CRUDL) APIs pour aider les développeurs à gérer leur infrastructure cloud de manière simple et cohérente. L'API Cloud Control commune APIs permet aux développeurs de gérer de manière uniforme le cycle de vie d'AWS et des services tiers.

En tant que développeur, vous préférerez peut-être simplifier la façon dont vous gérez le cycle de vie de toutes vos ressources. Vous pouvez utiliser le modèle de configuration uniforme des ressources de l'API Cloud Control avec un format prédéfini pour standardiser la configuration de vos ressources cloud. De plus, vous bénéficierez d'un comportement uniforme des API (éléments de réponse et erreurs) lors de la gestion de vos ressources.

Par exemple, il vous sera facile de déboguer les erreurs lors des opérations CRUDL grâce à des codes d'erreur uniformes affichés par l'API Cloud Control, indépendants des ressources sur lesquelles vous opérez. Grâce à l'API Cloud Control, vous pourrez également configurer facilement les dépendances entre ressources. Vous n'aurez également plus besoin de créer et de gérer du code personnalisé pour les outils de plusieurs fournisseurs et APIs d'utiliser AWS conjointement des ressources tierces.

Automatisation et outillage

L'automatisation DevOps est une autre philosophie et pratique de base. L'automatisation se concentre sur l'installation, la configuration, le déploiement et le support de l'infrastructure et des applications qui s'y exécutent. En utilisant l'automatisation, vous pouvez configurer des environnements plus rapidement de manière standardisée et reproductible. La suppression des processus manuels est essentielle à la réussite d'une DevOps stratégie. Historiquement, la configuration des serveurs et le déploiement des applications étaient principalement des processus manuels. Les environnements deviennent atypiques et il est difficile de reproduire un environnement lorsque des problèmes surviennent.

Le recours à l'automatisation est essentiel pour tirer pleinement parti des avantages du cloud. En interne, AWS s'appuie largement sur l'automatisation pour fournir les fonctionnalités de base que sont l'élasticité et l'évolutivité.

Les processus manuels sont sujets aux erreurs, peu fiables et inadéquats pour soutenir une entreprise agile. Souvent, une organisation peut mobiliser des ressources hautement qualifiées pour fournir une configuration manuelle, alors qu'il serait préférable de consacrer du temps à d'autres activités plus critiques et à plus forte valeur ajoutée au sein de l'entreprise.

Les environnements d'exploitation modernes s'appuient généralement sur l'automatisation complète pour éliminer les interventions manuelles ou l'accès aux environnements de production. Cela inclut la publication de tous les logiciels, la configuration de la machine, les correctifs du système d'exploitation, le dépannage ou la correction de bogues. De nombreux niveaux de pratiques d'automatisation peuvent être utilisés ensemble pour fournir un processus end-to-end automatisé de niveau supérieur.

L'automatisation présente les principaux avantages suivants :

- Changements rapides
- Productivité améliorée
- Configurations répétables
- Environnements reproductibles
- Élasticité
- Dimensionnement automatique
- Tests automatisés

L'automatisation est la pierre angulaire des AWS services et est prise en charge en interne dans tous les services, fonctionnalités et offres.

Rubriques

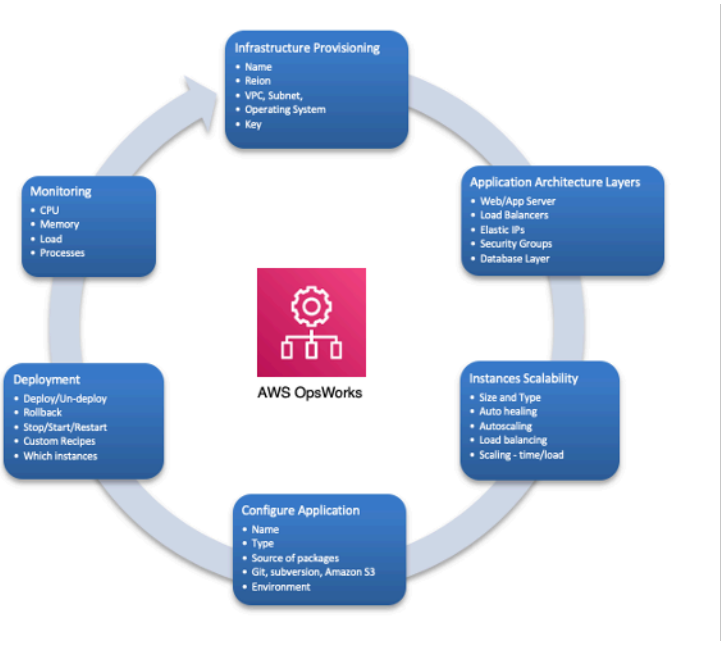
- AWS OpsWorks
- AWS Elastic Beanstalk
- EC2 Image Builder
- AWS Proton
- AWS Service Catalog
- AWS Cloud9
- AWS CloudShell
- Amazon CodeGuru

AWS OpsWorks

AWS OpsWorksva DevOps encore plus loin que les principes de AWS Elastic Beanstalk. Il peut être considéré comme un service de gestion d'applications plutôt que comme un simple conteneur d'applications. OpsWorks fournit des niveaux d'automatisation encore plus élevés, avec des fonctionnalités supplémentaires telles que l'intégration au logiciel de gestion des configurations (Chef) et la gestion du cycle de vie des applications. Vous pouvez utiliser la gestion du cycle de vie des applications pour définir le moment où les ressources sont configurées, déployées, dédéployées ou supprimées.

Pour plus de flexibilité AWS OpsWorks , vous avez défini votre application dans des piles configurables. Vous pouvez également sélectionner des piles d'applications prédéfinies. Les piles d'applications contiennent toutes les ressources AWS dont votre application a besoin, notamment les serveurs d'applications, les serveurs Web, les bases de données et les équilibreurs de charge.

Les piles d'applications sont organisées en couches architecturales afin que les piles puissent être maintenues indépendamment les unes des autres. Les couches d'exemple peuvent inclure le niveau Web, le niveau application et le niveau base de données. Dès le départ, AWS simplifie OpsWorks également la configuration des groupes AWS Auto Scaling et des équilibreurs de charge Elastic Load Balancing (ELB), illustrant ainsi le DevOps principe de l'automatisation. Tout comme AWS Elastic Beanstalk OpsWorks , AWS prend en charge le contrôle de version des applications, le déploiement continu et la gestion de la configuration de l'infrastructure



OpsWorks affichage des DevOps fonctionnalités et de l'architecture

AWS OpsWorks soutient également les DevOps pratiques de surveillance et de journalisation (abordées dans la section suivante). L'assistance en matière de surveillance est fournie par Amazon CloudWatch. Tous les événements du cycle de vie sont enregistrés, et un journal Chef distinct documente toutes les recettes Chef exécutées, ainsi que toutes les exceptions.

AWS Elastic Beanstalk

AWS Elastic Beanstalk est un service permettant de déployer et de mettre à l'échelle rapidement des applications et des services web développés avec Java, .NET, PHP, Node.js, Python, Ruby, Go et Docker sur des serveurs courants, tels qu'Apache, NGINX, Passenger et IIS.

Elastic Beanstalk est une abstraction basée sur Amazon EC2, Auto Scaling, et simplifie le déploiement en fournissant des fonctionnalités supplémentaires telles que le clonage, les déploiements blue/green , l'interface de ligne de commande Elastic Beanstalk (EB CLI) et l'intégration avec AWS Toolkit pour Visual Studio, Visual Studio Code, Eclipse et IntelliJ pour augmenter la productivité des développeurs.

EC2 Image Builder

EC2 Image Builder est un AWS service entièrement géré qui vous aide à automatiser la création, la maintenance, la validation, le partage et le déploiement d'une AMI personnalisée, sécurisée et

personnalisée pour up-to-date Linux ou Windows. EC2 Image Builder peut également être utilisé pour créer des images de conteneurs. Vous pouvez utiliser le AWS Management Console AWS CLI, le ou APIs pour créer des images personnalisées dans votre AWS compte.

EC2 Image Builder réduit considérablement les efforts liés à la conservation up-to-date et à la sécurité des images en fournissant une interface graphique simple, une automatisation intégrée et des paramètres de sécurité AWS fournis. Avec EC2 Image Builder, il n'y a aucune étape manuelle pour mettre à jour une image et vous n'avez pas besoin de créer votre propre pipeline d'automatisation.

AWS Proton

AWS Protonpermet aux équipes de plateforme de connecter et de coordonner les différents outils dont vos équipes de développement ont besoin pour le provisionnement de l'infrastructure, les déploiements de code, la surveillance et les mises à jour. AWS Proton permet une infrastructure automatisée sous forme de provisionnement de code et de déploiement d'applications sans serveur et basées sur des conteneurs.

AWS Proton permet aux équipes chargées de la plateforme de définir leur infrastructure et leurs outils de déploiement, tout en offrant aux développeurs une expérience en libre-service pour obtenir une infrastructure et déployer du code. Grâce à AWS Proton cela, les équipes de la plateforme fournissent des ressources partagées et définissent les piles d'applications, y compris les CI/CD pipelines et les outils d'observabilité. Vous pouvez ensuite gérer les fonctionnalités d'infrastructure et de déploiement disponibles pour les développeurs.

AWS Service Catalog

AWS Service Catalogpermet aux entreprises de créer et de gérer des catalogues de services informatiques approuvés. AWS Ces services informatiques peuvent inclure des images de machines virtuelles, des serveurs, des logiciels, des bases de données, etc., ainsi que des architectures d'applications complètes à plusieurs niveaux. AWS Service Catalog vous permet de gérer de manière centralisée les services informatiques, les applications, les ressources et les métadonnées déployés afin de garantir une gouvernance cohérente de vos modèles iAc.

Vous pouvez ainsi répondre à vos exigences de conformité tout en vous assurant que vos clients peuvent déployer rapidement les services informatiques approuvés dont ils ont besoin. AWS Service Catalog Les utilisateurs finaux peuvent déployer rapidement uniquement les services informatiques approuvés dont ils ont besoin, en respectant les contraintes définies par votre organisation.

AWS Cloud9

AWS Cloud9est un IDE basé sur le cloud qui vous permet d'écrire, d'exécuter et de déboguer votre code avec un simple navigateur. Il inclut un éditeur de code, un débogueur et un terminal. AWS Cloud9 est livré avec des outils essentiels pour les langages de programmation courants JavaScript, notamment Python, PHP, etc. Vous n'avez donc pas besoin d'installer de fichiers ou de configurer votre machine de développement pour démarrer de nouveaux projets. Comme votre AWS Cloud9 IDE est basé sur le cloud, vous pouvez travailler sur vos projets depuis votre bureau, votre domicile ou n'importe où à l'aide d'une machine connectée à Internet.

AWS CloudShell

AWS CloudShellest un shell basé sur un navigateur qui facilite la gestion, l'exploration et l'interaction en toute sécurité de vos AWS ressources. AWS CloudShell est pré-authentifié avec les informations d'identification de votre console. Les outils de développement et d'exploitation courants sont préinstallés, il n'est donc pas nécessaire d'installer ou de configurer le logiciel sur votre machine locale.

Amazon CodeGuru

Amazon CodeGuru est un outil de développement qui fournit des recommandations intelligentes pour améliorer la qualité du code et identifier les lignes de code les plus coûteuses d'une application. CodeGuru Intégrez-le à votre flux de travail de développement logiciel existant pour automatiser les révisions de code pendant le développement des applications, surveiller en permanence les performances des applications en production et fournir des recommandations et des indices visuels sur la manière d'améliorer la qualité du code, les performances des applications et de réduire les coûts globaux. CodeGuru comporte deux éléments :

- Amazon CodeGuru Reviewer — Amazon CodeGuru Reviewer est un service de révision de code automatique qui identifie les défauts critiques et les écarts par rapport aux meilleures pratiques de codage pour le code Java et Python. Il analyse les lignes de code d'une pull request et fournit des recommandations intelligentes basées sur les normes apprises dans le cadre de grands projets open source ainsi que sur la base de code Amazon.
- Amazon CodeGuru Profiler — Amazon CodeGuru Profiler analyse le profil d'exécution de l'application et fournit des recommandations et des visualisations intelligentes qui aident les développeurs à améliorer les performances des parties les plus pertinentes de leur code.

Surveillance et observabilité

La communication et la collaboration sont fondamentales dans une DevOps philosophie. Pour faciliter cela, le feedback est essentiel. Ces commentaires sont fournis par notre suite de services de surveillance et d'observabilité.

AWS fournit les services de surveillance et de journalisation suivants :

Rubriques

- CloudWatch Métriques Amazon
- CloudWatch Alarmes Amazon
- Amazon CloudWatch Logs
- Informations sur Amazon CloudWatch Logs
- CloudWatch Événements Amazon
- Amazon EventBridge
- AWS CloudTrail
- Amazon DevOps Guru
- AWS X-Ray
- Amazon Managed Service for Prometheus
- Amazon Managed Grafana

CloudWatch Métriques Amazon

CloudWatch Les métriques Amazon collectent automatiquement des données à partir de AWS services tels que EC2 les instances Amazon, les volumes Amazon EBS et les instances de base de données (DB) Amazon RDS. Ces métriques peuvent ensuite être organisées sous forme de tableaux de bord et des alarmes ou des événements peuvent être créés pour déclencher des événements ou effectuer des actions Auto Scaling.

CloudWatch Alarmes Amazon

Vous pouvez configurer des alarmes à l'aide des CloudWatch alarmes Amazon en fonction des métriques collectées par Amazon CloudWatch Metrics. L'alarme peut ensuite envoyer une notification à la rubrique Amazon SNS ou lancer des actions Auto Scaling. Une alarme nécessite une période

(durée d'évaluation d'une métrique), une période d'évaluation (nombre de points de données les plus récents) et des points de données pour déclencher une alarme (nombre de points de données pendant la période d'évaluation).

Amazon CloudWatch Logs

Amazon CloudWatch Logs est un service d'agrégation et de surveillance des journaux. AWS CodeBuild CodeCommit, CodeDeploy et CodePipeline fournissez des intégrations avec les CloudWatch journaux afin que tous les journaux puissent être surveillés de manière centralisée. En outre, les services mentionnés précédemment avec lesquels divers autres AWS services s'intègrent directement CloudWatch.

Avec CloudWatch Logs, vous pouvez :

- Interrogez les données de votre journal
- Surveillez les journaux des EC2 instances Amazon
- Surveiller les événements AWS CloudTrail enregistrés
- Définition de la politique de conservation des journaux

Informations sur Amazon CloudWatch Logs

Amazon CloudWatch Logs Insights analyse vos journaux et vous permet d'effectuer des requêtes et des visualisations interactives. Il comprend différents formats de journaux et découvre automatiquement les champs des journaux JSON.

CloudWatch Événements Amazon

Amazon CloudWatch Events fournit un flux en temps quasi réel d'événements système décrivant les modifications apportées aux AWS ressources. A l'aide de règles simples et rapidement configurées, vous pouvez faire correspondre des événements et les acheminer vers un ou plusieurs flux ou une ou plusieurs fonctions cibles.

CloudWatch Events prend conscience des changements opérationnels au fur et à mesure qu'ils se produisent. CloudWatch Events répond à ces changements opérationnels et prend les mesures correctives nécessaires, en envoyant des messages pour répondre à l'environnement, en activant des fonctions, en apportant des modifications et en capturant des informations d'état.

Vous pouvez configurer des règles dans Amazon CloudWatch Events pour vous avertir des modifications apportées aux AWS services et intégrer ces événements à d'autres systèmes tiers utilisant Amazon EventBridge. Les services AWS DevOps connexes intégrés à CloudWatch Events sont les suivants.

- Événements d'Application Auto Scaling
- Événements CodeBuild
- Événements CodeCommit
- CodeDeploy Événements
- CodePipeline Événements

Amazon EventBridge

> **Note**
>
> Amazon CloudWatch Events EventBridge est le même service sous-jacent et l'API, mais EventBridge offrent davantage de fonctionnalités.

Amazon EventBridge est un bus d'événements sans serveur qui permet des intégrations entre les AWS services, les logiciels en tant que services (SaaS) et vos applications. En plus de créer des applications pilotées par les événements, elles EventBridge peuvent être utilisées pour signaler les événements provenant de services tels que CodeBuild CodeDeploy, CodePipeline, et CodeCommit.

AWS CloudTrail

Pour adopter les DevOps principes de collaboration, de communication et de transparence, il est important de comprendre qui apporte des modifications à votre infrastructure. En AWS, cette transparence est assurée par AWS CloudTrail. Toutes les AWS interactions sont gérées par le biais AWS d'appels d'API qui sont surveillés et enregistrés par AWS CloudTrail. Tous les fichiers journaux générés sont stockés dans un compartiment Amazon S3 que vous définissez. Les fichiers journaux sont chiffrés à l'aide du chiffrement côté serveur (SSE) Amazon S3. Tous les appels d'API sont enregistrés, qu'ils proviennent directement d'un utilisateur ou qu'ils proviennent d'un AWS service pour le compte d'un utilisateur. De nombreux groupes peuvent tirer parti CloudTrail des journaux, notamment les équipes opérationnelles pour le support, les équipes de sécurité pour la gouvernance et les équipes financières pour la facturation.

Amazon DevOps Guru

Amazon DevOps Guru est un service basé sur l'apprentissage automatique (ML) conçu pour améliorer facilement les performances opérationnelles et la disponibilité d'une application. DevOps Guru aide à détecter les comportements qui s'écartent des modèles opérationnels normaux, afin que vous puissiez identifier les problèmes opérationnels bien avant qu'ils n'aient un impact sur vos clients.

DevOps Guru utilise des modèles de machine learning basés sur des années d'expérience sur Amazon.com et sur l'excellence AWS opérationnelle pour aider à identifier les comportements anormaux des applications (par exemple, latence accrue, taux d'erreur, contraintes de ressources, etc.) et à identifier les problèmes critiques susceptibles d'entraîner des pannes ou des interruptions de service.

Lorsque DevOps Guru identifie un problème critique, il économise du temps de débogage en récupérant des informations pertinentes et spécifiques à partir d'un grand nombre de sources de données. Il envoie automatiquement une alerte et fournit un résumé des anomalies associées, ainsi que le contexte indiquant quand et où le problème s'est produit.

AWS X-Ray

AWS X-Rayaide les développeurs à analyser et à déboguer les applications distribuées de production, telles que celles créées à l'aide d'une architecture de microservices. Avec X-Ray, vous pouvez comprendre les performances de votre application et de ses services sous-jacents afin d'identifier et de résoudre les causes profondes des problèmes et des erreurs de performance. X-Ray fournit une end-to-end vue des demandes au fur et à mesure qu'elles transitent dans votre application et affiche une carte des composants sous-jacents de votre application. X-Ray vous permet de :

- Création d'une carte des services — En suivant les demandes adressées à vos applications, X-Ray peut créer une carte des services utilisés par votre application. Cela vous donne une vue des connexions entre les services de votre application et vous permet de créer un arbre de dépendances, de détecter la latence ou les erreurs lorsque vous travaillez dans des zones de AWS disponibilité ou des régions, de vous concentrer sur les services qui ne fonctionnent pas comme prévu, etc.
- Identifiez les erreurs et les bogues — X-Ray peut automatiquement mettre en évidence les bogues ou les erreurs dans le code de votre application en analysant le code de réponse pour chaque demande envoyée à votre application. Cela permet de déboguer facilement le code de l'application sans que vous ayez à reproduire le bogue ou l'erreur.

- Créez vos propres applications d'analyse et de visualisation : X-Ray fournit un ensemble de requêtes APIs que vous pouvez utiliser pour créer vos propres applications d'analyse et de visualisation qui utilisent les données enregistrées par X-Ray.

Amazon Managed Service for Prometheus

Amazon Managed Service for Prometheus est un service de surveillance sans serveur pour les métriques compatible avec le logiciel libre Prometheus, qui vous permet de surveiller et d'émettre des alertes en toute sécurité sur les environnements de conteneurs. Amazon Managed Service for Prometheus simplifie le démarrage des applications de surveillance sur Amazon Elastic Kubernetes Service, Amazon Elastic Container AWS Fargate Service et sur les clusters Kubernetes autogérés.

Amazon Managed Grafana

Amazon Managed Grafana est un service entièrement géré proposant des visualisations de données riches et interactives pour aider les clients à analyser, surveiller et générer des alertes sur les métriques, les journaux et les traces provenant de plusieurs sources de données. Vous pouvez créer des tableaux de bord interactifs et les partager avec tous les membres de votre organisation grâce à un service évolutif automatique, hautement disponible et sécurisé pour l'entreprise.

Communication et collaboration

Que vous adoptiez DevOps la culture dans votre organisation ou que vous subissiez une transformation DevOps culturelle, la communication et la collaboration sont des éléments importants de votre approche. Chez Amazon, nous avons réalisé qu'il était nécessaire de changer l'état d'esprit de nos équipes et avons donc adopté le concept des équipes à deux pizzas.

« Nous essayons de créer des équipes qui ne sont pas plus nombreuses que ce que l'on peut nourrir avec deux pizzas », explique Bezos. « C'est ce que nous appelons la règle de l'équipe des deux pizzas. »

Plus l'équipe est petite, meilleure est la collaboration. La collaboration est très importante, car les versions logicielles évoluent plus rapidement que jamais. Et la capacité d'une équipe à fournir le logiciel peut être un facteur de différenciation pour votre organisation par rapport à vos concurrents. Imaginez une situation dans laquelle une nouvelle fonctionnalité du produit doit être publiée ou un bogue doit être corrigé. Vous voulez que cela se fasse le plus rapidement possible, afin de réduire le go-to-market temps imparti. Vous ne voulez pas que la transformation soit un processus lent ; vous voulez une approche agile dans laquelle les vagues de changements commencent à avoir un impact.

La communication entre les équipes est également importante à mesure que vous évoluez vers le modèle de responsabilité partagée et que vous commencez à sortir de l'approche de développement cloisonnée. Cela introduit le concept de propriété dans l'équipe et change son point de vue pour considérer le processus comme une end-to-end entreprise. Votre équipe ne doit pas considérer vos environnements de production comme des boîtes noires où ils n'ont aucune visibilité.

La transformation culturelle est également importante, car vous êtes peut-être en train de constituer une DevOps équipe commune ou d'avoir un membre DevOps ciblé dans votre équipe. Ces deux approches introduisent une responsabilité partagée au sein de l'équipe.

Sécurité

Que vous procédiez à une DevOps transformation ou que vous mettiez en œuvre des DevOps principes pour la première fois, vous devez considérer la sécurité comme intégrée à vos DevOps processus. Cela devrait être une préoccupation transversale à toutes les étapes de conception et de déploiement des tests.

Avant d'aborder la question de DevOps la sécurité en profondeur AWS, ce paper examine le modèle de responsabilité AWS partagée.

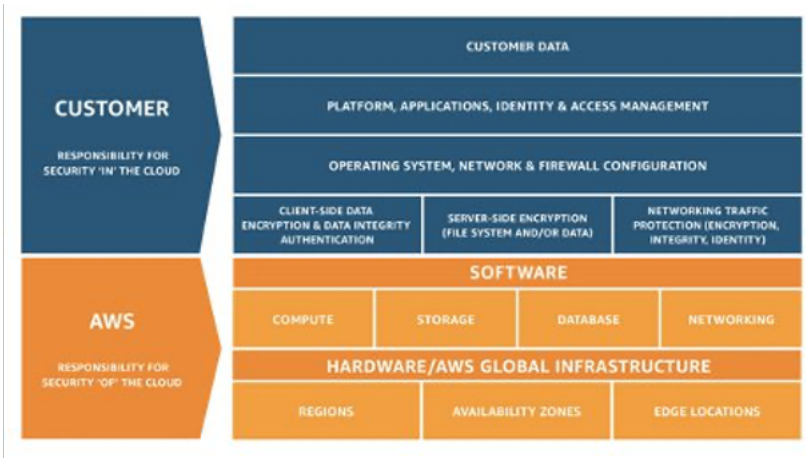
Rubriques

AWS Modèle de responsabilité partagée

La sécurité est une responsabilité partagée entre AWS et le client. Les différentes parties du modèle de responsabilité partagée sont les suivantes :

- Responsabilité d'AWS, « Sécurité du cloud », AWS est chargée de protéger l'infrastructure qui exécute tous les services proposés dans le AWS Cloud. Cette infrastructure comprend le matériel, les logiciels, les réseaux et les installations qui exécutent AWS Cloud les services.
- Responsabilité du client « Sécurité dans le cloud » — La responsabilité du client est déterminée par les AWS Cloud services sélectionnés par le client. Ils déterminent la quantité de travail de configuration que doit réaliser le client dans le cadre de ses responsabilités en matière de sécurité.

Ce modèle partagé peut contribuer à alléger la charge opérationnelle du client en AWS exploitant, en gérant et en contrôlant les composants, depuis le système d'exploitation hôte et la couche de virtualisation jusqu'à la sécurité physique des installations dans lesquelles le service fonctionne. Cela est essentiel dans les cas où le client souhaite comprendre la sécurité de son environnement de construction.



Modèle de responsabilité partagée AWS

Pour DevOps, attribuez des autorisations en fonction du modèle d'autorisation du moindre privilège. Ce modèle indique qu' « un utilisateur (ou un service) doit disposer des droits d'accès exacts nécessaires pour s'acquitter des responsabilités de son rôle, ni plus, ni moins ».

Les autorisations sont conservées dans IAM. Vous pouvez utiliser IAM pour contrôler qui est authentifié (connecté) et autorisé (autorisé) à utiliser les ressources.

Gestion de l’identité et des accès

Gestion des identités et des accès AWS(IAM) définit les contrôles et les politiques utilisés pour gérer l'accès aux AWS ressources. À l'aide d'IAM, vous pouvez créer des utilisateurs et des groupes et définir des autorisations pour différents DevOps services.

Outre les utilisateurs, divers services peuvent également avoir besoin d'accéder à AWS des ressources. Par exemple, votre CodeBuild projet peut avoir besoin d'un accès pour stocker des images Docker dans Amazon Elastic Container Registry (Amazon ECR) et d'autorisations pour écrire sur Amazon ECR. Ces types d'autorisations sont définis par un rôle de type spécial appelé rôle de service.

L'IAM est l'un des composants de l'infrastructure AWS de sécurité. Avec IAM, vous pouvez gérer de manière centralisée les groupes, les utilisateurs, les rôles de service et les informations d'identification de sécurité telles que les mots de passe, les clés d'accès et les politiques d'autorisation qui contrôlent les services et ressources AWS auxquels les utilisateurs peuvent accéder. La politique IAM vous permet de définir l'ensemble des autorisations. Cette politique peut ensuite être attachée à un rôle, à un utilisateur ou à un service pour définir leur autorisation.

Vous pouvez également utiliser IAM pour créer des rôles largement utilisés dans le cadre de la DevOps stratégie que vous souhaitez adopter. Dans certains cas, il peut être parfaitement logique de le faire par programmation AssumeRoleau lieu d'obtenir directement les autorisations. Lorsqu'un service ou un utilisateur assume des rôles, il reçoit des informations d'identification temporaires lui permettant d'accéder à un service auquel il n'a normalement pas accès.

Conclusion

Pour que le passage au cloud soit fluide, efficient et efficace, les entreprises technologiques doivent adopter des DevOps principes et des pratiques. Ces principes sont intégrés et constituent la pierre angulaire de nombreux AWS services, en particulier ceux des offres de déploiement et de surveillance. AWS

Commencez par définir votre infrastructure sous forme de code à l'aide du service AWS CloudFormation ou AWS CDK. Définissez ensuite la manière dont vos applications utiliseront le déploiement continu à l'aide de services tels que AWS CodeBuild AWS CodeDeploy, AWS CodePipeline, et AWS CodeCommit. Au niveau de l'application, utilisez des conteneurs tels qu' AWS Elastic Beanstalk Amazon ECS ou Amazon Elastic Kubernetes Service (Amazon EKS). OpsWorks À utiliser pour simplifier la configuration des architectures courantes. L'utilisation de ces services permet également d'inclure facilement d'autres services importants tels que Auto Scaling et Elastic Load Balancing.

Enfin, utilisez DevOps une stratégie de surveillance telle qu'Amazon CloudWatch et des pratiques de sécurité solides telles que l'IAM.

En AWS tant que partenaire, vos DevOps principes apportent de l'agilité à votre entreprise et à votre service informatique et accélèrent votre transition vers le cloud.

Révisions du document

Pour être informé des mises à jour de ce livre blanc, abonnez-vous au flux RSS.

Modification	Description	Date
Mis à jour	Mis à jour	7 avril 2023
Sections mises à jour pour inclure de nouveaux services	Sections mises à jour pour inclure de nouveaux services	16 octobre 2020
Publication initiale	Livre blanc publié pour la première fois	1er décembre 2014

Collaborateurs

Les personnes qui ont contribué à ce document incluent :

- Abhra Sinha, architecte de solutions
- Anil Nadiminti, architecte de solutions
- Muhammad Mansoor, architecte de solutions
- Ajit Zadgaonkar, leader mondial de la technologie, modernisation
- Juan Lamadrid, architecte de solutions
- Darren Ball, architecte de solutions
- Rajeswari Malladi, architecte de solutions
- Pallavi Nargund, architecte de solutions
- Bert Zahniser, architecte de solutions
- Abdullahi Olaoye, architecte de solutions cloud
- Mohamed Kiswani, responsable du développement logiciel
- Tara McCann, directrice, architecte de solutions

Avis

Il incombe aux clients de procéder à une évaluation indépendante des informations contenues dans le présent document. Ce document : (a) est fourni à titre informatif uniquement, (b) représente les offres de produits et les pratiques actuelles d'AWS, qui sont susceptibles d'être modifiées sans préavis, et (c) ne crée aucun engagement ni aucune garantie de la part d'AWS et de ses filiales, fournisseurs ou concédants de licence. Les produits ou services AWS sont fournis « tels quels » sans garanties, déclarations ou conditions d'aucune sorte, qu'elles soient explicites ou implicites. Les responsabilités et obligations d'AWS vis-à-vis de ses clients sont régies par les contrats AWS. Le présent document ne fait partie d'aucun, et ne modifie aucun, contrat entre AWS et ses clients.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.