Whitepaper AWS

Panoramica delle opzioni di distribuzione su AWS

Panoramica delle opzioni di distribuzione su AWS: Whitepaper AWS

Table of Contents

Panoramica delle opzioni di distribuzione su AWS

Data di pubblicazione: 31 maggio 2024 () Revisioni del documento

Sintesi

Amazon Web Services (AWS) offre diverse opzioni per il provisioning dell'infrastruttura e la distribuzione delle applicazioni. Che l'architettura dell'applicazione sia una semplice applicazione Web a tre livelli o un insieme complesso di carichi di lavoro, AWS offre servizi di distribuzione per soddisfare i requisiti della tua applicazione e della tua organizzazione.

Questo white paper è destinato alle persone che cercano una panoramica dei diversi servizi di distribuzione offerti da AWS. Descrive le funzionalità comuni disponibili in questi servizi di distribuzione e articola le strategie di base per la distribuzione e l'aggiornamento degli stack di applicazioni.

Introduzione

La progettazione di una soluzione di distribuzione per la tua applicazione è una parte fondamentale della creazione di un'applicazione ben architettata su AWS. In base alla natura dell'applicazione e ai servizi sottostanti richiesti, è possibile utilizzare i servizi AWS per creare una soluzione di distribuzione flessibile che può essere personalizzata per soddisfare le esigenze sia dell'applicazione che dell'organizzazione.

Il catalogo in costante crescita di servizi AWS non solo complica il processo di decisione di quali servizi comporranno l'architettura della tua applicazione, ma anche il processo di decisione su come creare, gestire e aggiornare la tua applicazione. Quando si progetta una soluzione di distribuzione su AWS, è necessario considerare in che modo la soluzione soddisferà le seguenti funzionalità:

- Fornitura: crea l'infrastruttura grezza o l'infrastruttura di servizi gestiti necessaria per la tua applicazione.
- Configurazione: personalizza l'infrastruttura in base all'ambiente, al runtime, alla sicurezza, alla disponibilità, alle prestazioni, alla rete o ad altri requisiti applicativi.
- Implementazione: installa o aggiorna i componenti dell'applicazione sulle risorse dell'infrastruttura e gestisci la transizione da una versione precedente dell'applicazione a una nuova versione dell'applicazione.
- Scalabilità: regola in modo proattivo o reattivo la quantità di risorse disponibili per l'applicazione in base a una serie di criteri definiti dall'utente.
- Monitoraggio: offri visibilità sulle risorse che vengono lanciate come parte dell'architettura dell'applicazione. Tieni traccia dell'utilizzo delle risorse, del successo o del fallimento dell'implementazione, dello stato delle applicazioni, dei log delle applicazioni, dei cambiamenti nella configurazione e altro ancora.

Questo white paper evidenzia i servizi di distribuzione offerti da AWS e delinea le strategie per progettare un'architettura di distribuzione di successo per qualsiasi tipo di applicazione.

Servizi di distribuzione AWS

Il compito di progettare una soluzione di implementazione scalabile, efficiente ed economica non dovrebbe limitarsi al modo in cui verrà aggiornata la versione dell'applicazione, ma dovrebbe anche considerare come gestire l'infrastruttura di supporto durante l'intero ciclo di vita dell'applicazione. Il provisioning delle risorse, la gestione della configurazione, la distribuzione delle applicazioni, gli aggiornamenti software, il monitoraggio, il controllo degli accessi e altre preoccupazioni sono tutti fattori importanti da considerare nella progettazione di una soluzione di distribuzione.

I servizi AWS possono fornire funzionalità di gestione per uno o più aspetti del ciclo di vita delle applicazioni. A seconda dell'equilibrio desiderato tra controllo (gestione manuale delle risorse) e convenienza (gestione delle risorse AWS) e del tipo di applicazione, questi servizi possono essere utilizzati singolarmente o combinati per creare una soluzione di distribuzione ricca di funzionalità. Questa sezione fornirà una panoramica dei servizi AWS che possono essere utilizzati per consentire alle organizzazioni di creare e distribuire applicazioni in modo più rapido e affidabile.

AWS CloudFormation

AWS CloudFormationè un servizio che consente ai clienti di fornire e gestire quasi tutte le risorse AWS utilizzando un linguaggio di template personalizzato espresso in YAML o JSON. Un AWS CloudFormation modello crea risorse di infrastruttura in un gruppo chiamato stack e consente di definire e personalizzare tutti i componenti necessari per il funzionamento dell'applicazione mantenendo il pieno controllo di tali risorse. L'utilizzo dei modelli offre la possibilità di implementare il controllo delle versioni sull'infrastruttura e la possibilità di replicare l'infrastruttura in modo rapido e affidabile.

AWS CloudFormation offre un controllo granulare sul provisioning e sulla gestione di tutti i componenti dell'infrastruttura applicativa, da componenti di basso livello come tabelle di routing o configurazioni di sottorete, a componenti di alto livello come le distribuzioni. CloudFront AWS CloudFormation è comunemente usato con altri servizi di distribuzione AWS o strumenti di terze parti, in combinazione AWS CloudFormation con servizi di distribuzione più specializzati per gestire le distribuzioni di codice applicativo sui componenti dell'infrastruttura.

AWS offre estensioni al CloudFormation servizio oltre alle sue funzionalità di base:

- AWS Cloud Development Kit (AWS CDK)è un kit di sviluppo software (SDK) open source per modellare in modo programmatico l'infrastruttura AWS con JavaScript Python TypeScript, Java o C#/.NET.

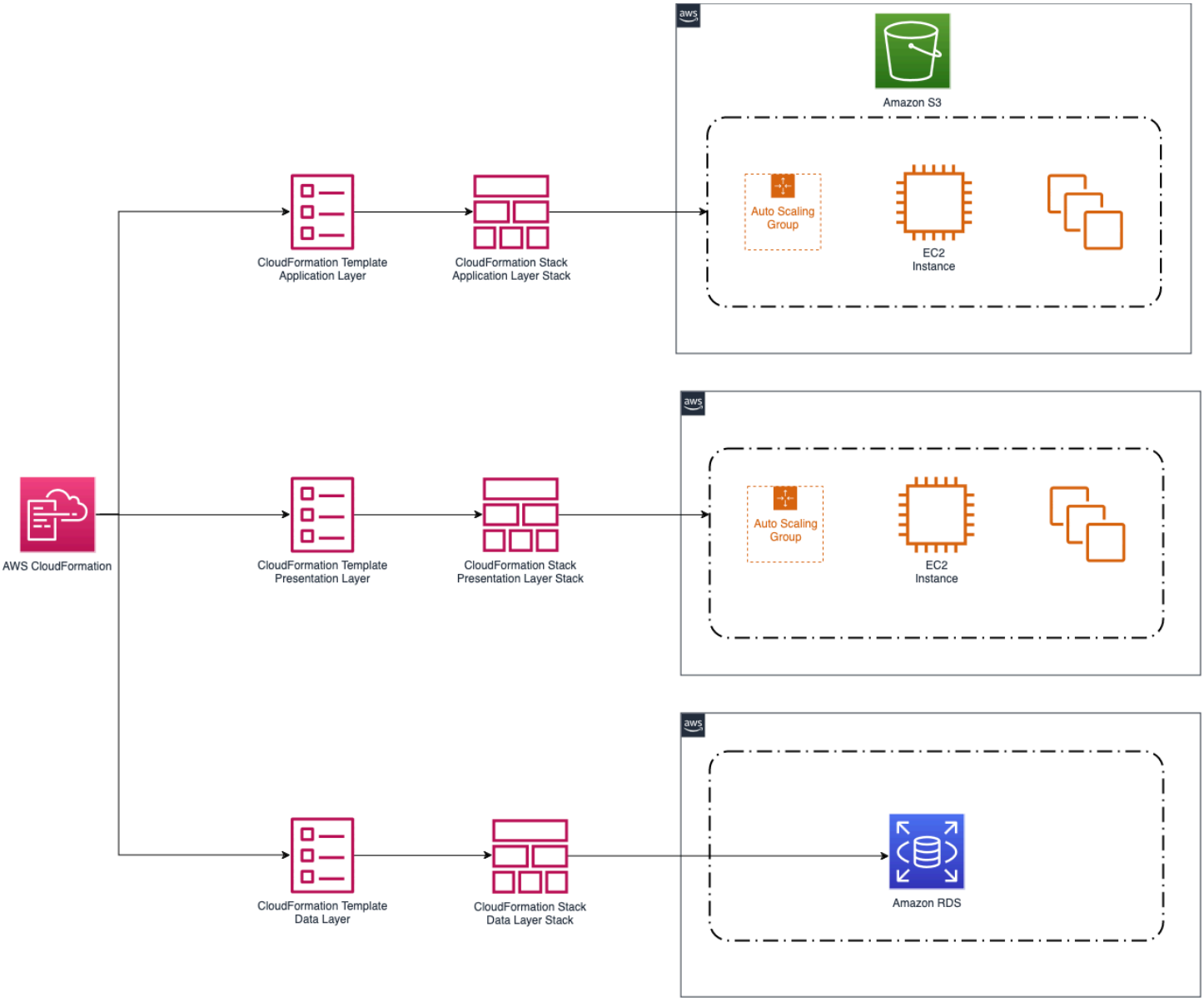
- AWS Serverless Application Model(AWS SAM) è un framework open source per semplificare la creazione di applicazioni serverless su AWS. Fornisce una sintassi abbreviata per esprimere funzioni APIs, database e mappature delle sorgenti degli eventi.

Tabella 1: funzionalità di distribuzione AWS CloudFormation

Funzionalità	Descrizione
Fornitura	CloudFormation creerà e aggiornerà automatic amente i componenti dell'infrastruttura definiti in un modello. Fai riferimento alle AWS CloudFormation Best Practice per maggiori dettagli sulla creazione di infrastrutture utilizzando AWS CloudFormation i modelli.
Configura	AWS CloudFormation i modelli offrono un'ampia flessibilità per personalizzare e aggiornare tutti i componenti dell'infrastruttura. Fai riferimento a AWS CloudFormation Tem plate Anatomy per maggiori dettagli sulla personalizzazione dei modelli.
Implementazione	Aggiorna i tuoi AWS CloudFormation modelli per modificare le risorse in una pila. A seconda dell'architettura dell'applicazione, potrebbe essere necessario un servizio di distribuzione aggiuntivo per aggiornare la versione dell'appl icazione in esecuzione sull'infrastruttura. Consulta la sezione Distribuzione di applicazi oni su Amazon EC2 AWS CloudFormation per maggiori dettagli su come AWS CloudForm ation può essere utilizzata come soluzione di distribuzione.

Funzionalità	Descrizione
Dimensionare	AWS CloudFormation non gestirà automatic amente la scalabilità dell'infrastruttura per tuo conto; tuttavia, puoi configurare le politiche di scalabilità automatica per le tue risorse in un AWS CloudFormation modello.
Monitoraggio	AWS CloudFormation fornisce il monitorag gio nativo dell'esito positivo o negativo degli aggiornamenti dell'infrastruttura definiti in un modello, nonché il rilevamento delle deviazion i per monitorare quando le risorse definite in un modello non soddisfano le specifich e. Dovranno essere disponibili soluzioni di monitoraggio aggiuntive per il monitoraggio e le metriche a livello di applicazione. Per maggiori dettagli su come monitora gli aggiornamenti dell'infrastruttura, consulta Monitoring the Progress of a Stack Update. AWS CloudFormation

Il diagramma seguente mostra un caso d'uso comune per. AWS CloudFormation Qui vengono creati AWS CloudFormation modelli per definire tutti i componenti dell'infrastruttura necessari per creare una semplice applicazione web a tre livelli. In questo esempio, utilizziamo gli script di bootstrap definiti in AWS CloudFormation per distribuire la versione più recente della nostra applicazione su EC2 istanze Amazon; tuttavia, è anche pratica comune combinare servizi di distribuzione aggiuntivi con AWS CloudFormation (utilizzando AWS CloudFormation solo per la gestione dell'infrastruttura e le funzionalità di provisioning). Tieni presente che per creare l'infrastruttura viene utilizzato più AWS CloudFormation di un modello. Nel diagramma, AWS CloudFormation viene utilizzato per creare tutti i componenti dell'infrastruttura, inclusi ruoli IAM, sottoreti VPCs, tabelle di routing, gruppi di sicurezza e policy di bucket Amazon S3. Vengono utilizzati AWS CloudFormation modelli separati per creare ogni dominio dell'architettura dell'applicazione.



AWS CloudFormation caso d'uso

AWS Elastic Beanstalk

AWS Elastic Beanstalkè un easy-to-use servizio per la distribuzione e la scalabilità di applicazioni e servizi Web sviluppati con Java, .NET, .NET Core, PHP, Node.js, Python, Ruby, Go o Docker su server familiari come Apache, Nginx, Passenger e IIS. Elastic Beanstalk è una soluzione completa per la gestione delle applicazioni che gestisce tutte le attività dell'infrastruttura e della piattaforma per tuo conto.

Con Elastic Beanstalk, puoi distribuire, gestire e scalare rapidamente le applicazioni senza l'onere operativo della gestione dell'infrastruttura. Elastic Beanstalk riduce la complessità di gestione delle applicazioni Web, rendendolo una buona scelta per le organizzazioni che non conoscono AWS o desiderano implementare un'applicazione Web il più rapidamente possibile.

Quando si utilizza Elastic Beanstalk come soluzione di distribuzione, è sufficiente caricare il codice sorgente ed Elastic Beanstalk fornirà e gestirà tutta l'infrastruttura necessaria, inclusi server, database, sistemi di bilanciamento del carico, reti e gruppi di auto scaling. Sebbene queste risorse siano create per tuo conto, tu ne mantieni il pieno controllo, permettendo agli sviluppatori di personalizzarle secondo necessità. Elastic Beanstalk soddisfa i criteri di conformità ISO, PCI, SOC 1, SOC 2 e SOC 3 oltre ai criteri di idoneità HIPAA. Ciò significa che le applicazioni in esecuzione su Elastic Beanstalk possono elaborare dati finanziari regolamentati o informazioni sanitarie protette (PHI).

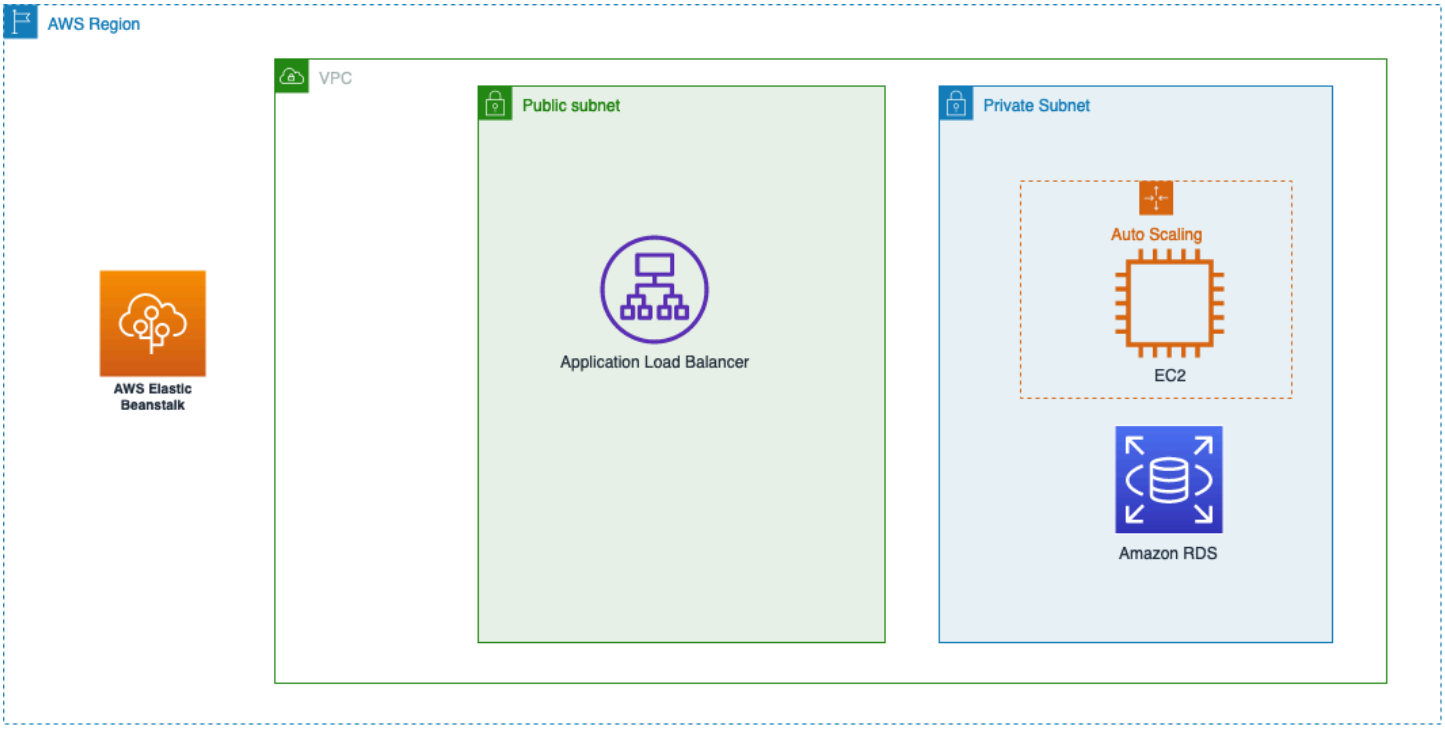
Tabella 2: Funzionalità di distribuzione AWS Elastic Beanstalk

Funzionalità	Descrizione
Fornitura	Elastic Beanstalk creerà tutti i componenti dell'infrastruttura necessari per far funzionare un'applicazione o un servizio Web che funziona su una delle piattaforme supportate. Se hai bisogno di un'infrastruttura aggiuntiva, questa dovrà essere creata all'esterno di Elastic Beanstalk. Fai riferimento a Elastic Beanstalk Platforms per maggiori dettagli sulle piattaforme di applicazioni web supportate da Elastic Beanstalk.
Configura	Elastic Beanstalk offre un'ampia gamma di opzioni per personalizzare le risorse del tuo ambiente. Consulta Configurazione degli ambienti Elastic Beanstalk per ulteriori informazioni sulla

Funzionalità	Descrizione
	personalizzazione delle risorse create da Elastic Beanstalk.
Implementazione	Elastic Beanstalk gestisce automaticamente le distribuzioni delle applicazioni e crea un ambiente che esegue una nuova versione dell'applicazione senza influire sugli utenti esistenti. Per ulteriori dettagli sulle distribuzioni delle applicazioni con AWS Elastic Beanstalk Elastic Beanstalk, consulta Deploying Applications.
Dimensionare	Elastic Beanstalk utilizza Elastic Load Balancing e Auto Scaling per scalare automatic amente l'applicazione in entrata e in uscita in base alle sue esigenze specifiche. Le zone di disponibilità multiple offrono la possibilità di migliorare l'affidabilità e la disponibilità delle applicazioni. Consulta Auto Scaling Group per il tuo ambiente Elastic Beanstalk per maggiori dettagli sulla scalabilità automatica con Elastic Beanstalk.
Monitoraggio	Elastic Beanstalk offre il monitoraggio integrato dell'ambiente per le applicazioni, tra cui errori o errori di implementazione, integrità dell'ambi ente, prestazioni delle risorse e registri delle applicazioni. Consulta Monitoraggio di un ambiente per maggiori dettagli sul monitoraggio completo dello stack con Elastic Beanstalk.

Funzionalità	Descrizione
Supporto Graviton	I processori basati su AWS Graviton arm64 offrono il miglior rapporto prezzo/prestazioni per i carichi di lavoro cloud in esecuzione su Amazon. EC2 Con AWS Graviton su Elastic Beanstalk, puoi EC2 selezionare tipi di istanze Amazon per soddisfare le esigenze di ottimizza zione dei tuoi carichi di lavoro e beneficiare di migliori prestazioni in termini di prezzo rispetto a un processore comparabile basato su x86.

Elastic Beanstalk semplifica la distribuzione e la gestione rapide delle applicazioni Web in AWS. L'esempio seguente mostra un caso d'uso generale per Elastic Beanstalk in quanto viene utilizzato per distribuire una semplice applicazione Web. Tutta l'infrastruttura applicativa (inclusi i gruppi di sicurezza, i ruoli IAM e gli CloudWatch allarmi) viene creata e gestita da Elastic Beanstalk. Alle EC2 istanze Amazon viene fornito automaticamente l'ambiente di runtime e i pacchetti di distribuzione. Gli ambienti Elastic Beanstalk possono integrarsi con risorse come Amazon Relational Database Service (Amazon RDS) create all'esterno di Elastic Beanstalk.



AWS Elastic Beanstalk caso d'uso

AWS CodeDeploy

AWS CodeDeployè un servizio di distribuzione completamente gestito che automatizza le distribuzioni di applicazioni su servizi di calcolo come Amazon, EC2 Amazon Elastic Container Service (Amazon ECS) o server locali. AWS Lambda Le organizzazioni possono utilizzare CodeDeploy per automatizzare le distribuzioni di un'applicazione e rimuovere le operazioni manuali soggette a errori dal processo di distribuzione. CodeDeploy può essere utilizzato con un'ampia varietà di contenuti applicativi, tra cui codice, funzioni serverless, file di configurazione e altro ancora.

CodeDeploy è pensato per essere utilizzato come servizio fondamentale incentrato sull'aiutare gli sviluppatori di applicazioni a distribuire e aggiornare il software in esecuzione sull'infrastruttura esistente. Non è una soluzione di gestione delle end-to-end applicazioni ed è pensata per essere utilizzata insieme ad altri servizi di distribuzione AWS come AWS CodeStaraltri strumenti per sviluppatori AWS e servizi di terze parti (consulta AWS CodeDeploy Product Integrations per un elenco completo delle integrazioni di prodotti) come parte di una pipeline CI/CD completa. AWS CodePipeline Inoltre, CodeDeploy non gestisce la creazione di risorse per conto dell'utente.

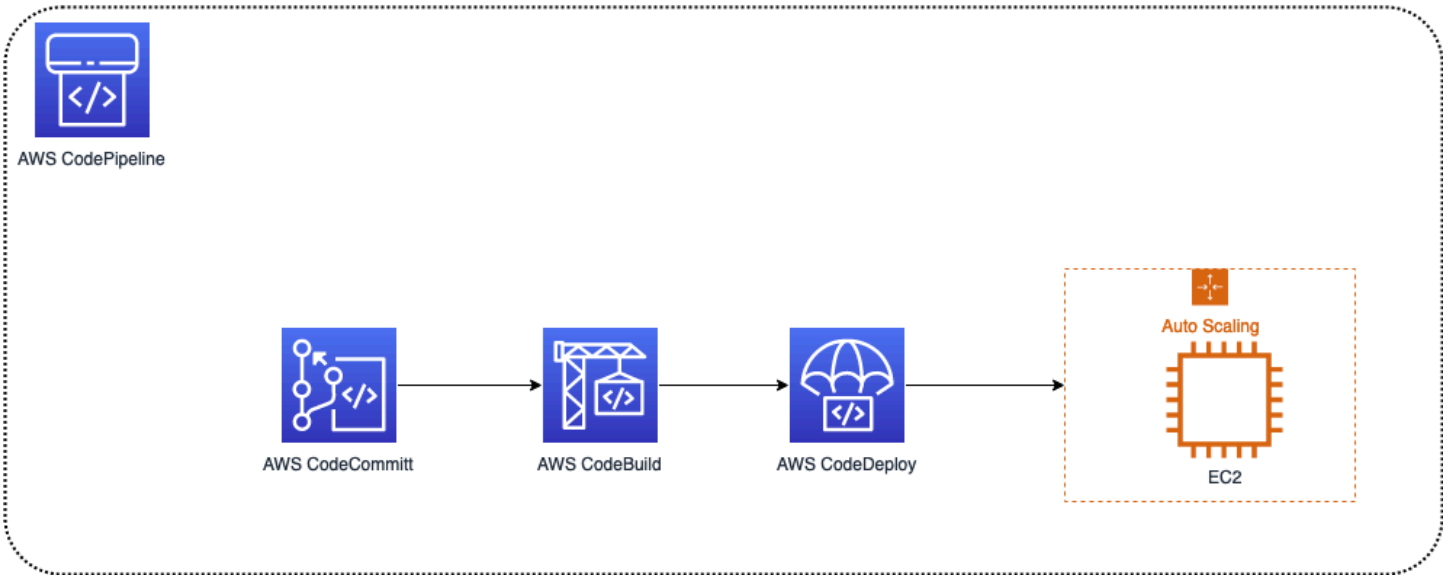
Tabella 3: funzionalità AWS CodeDeploy di distribuzione

Funzionalità	Descrizione
Fornitura	CodeDeploy è destinato all'uso con risorse di elaborazione esistenti e non crea risorse per conto dell'utente. CodeDeploy richiede che le risorse di elaborazione siano organizzate in un costrutto chiamato gruppo di distribuzione per distribuire il contenuto delle applicazioni. Per maggiori dettagli sul collegamento alle risorse di elaborazione, consulta Working CodeDeploy with Deployment Groups in. CodeDeploy
Configura	CodeDeploy utilizza un file di specifiche dell'applicazione per definire le personali zzazioni delle risorse di calcolo.

Funzionalità	Descrizione
	Fai riferimento a CodeDeploy AppSpec File Reference per maggiori dettagli sulle personalizzazioni delle risorse con. CodeDeploy
Implementazione	A seconda del tipo di risorsa di elaborazione CodeDeploy utilizzata, CodeDeploy offre diverse strategie per la distribuzione dell'applicazione. Per maggiori dettagli sui tipi di processi di distribuzione supportati CodeDeploy, consulta Working with Deployments in.
Dimensionare	CodeDeploy non supporta la scalabilità dell'infrastruttura applicativa sottostante; tuttavia, a seconda delle configurazioni di distribuzione, potrebbe creare risorse aggiuntive per supportare le distribuzioni blu/verdi.
Monitoraggio	CodeDeploy può monitorare il successo o il fallimento delle distribuzioni e offre una cronologia di tutte le implementazioni, ma non fornisce parametri prestazionali o a livello di applicazione. Fai riferimento a Monitoring Deployments in CodeDeploy per maggiori dettagli sui tipi di funzionalità di monitoraggio offerte da CodeDeploy

Il diagramma seguente illustra un caso d'uso generale CodeDeploy come parte di una soluzione CI/CD completa. In questo esempio, CodeDeploy viene utilizzato insieme ad altri strumenti di sviluppo AWS, in particolare AWS CodePipeline (automazione delle pipeline CI/CD), AWS CodeBuild(creazione e test dei componenti dell'applicazione) e AWS CodeCommit(repository di codice sorgente) per distribuire un'applicazione su un gruppo di istanze Amazon. EC2 CodeDeploy viene utilizzato con altri strumenti come parte di una pipeline CI/CD completa. CodeDeploy gestisce

la distribuzione dei componenti dell'applicazione su risorse di calcolo che fanno parte di un gruppo di distribuzione. Tutti i componenti dell'infrastruttura vengono creati all'esterno di. CodeDeploy



AWS CodeDeploy caso d'uso

AWS CodeDeploy per AWS Lambda

AWS CodeDeploy per AWS Lambda consente di automatizzare le implementazioni serverless, offrendo maggiore controllo e visibilità sulle versioni delle applicazioni. Puoi utilizzarla CodeDeploy per distribuire una nuova versione della tua funzione serverless a una piccola percentuale di utenti o traffico e aumentare gradualmente il traffico man mano che acquisisci fiducia nella nuova versione. Con CodeDeploy, puoi definire gruppi di distribuzione, che rappresentano un insieme di funzioni Lambda che ricevono traffico dalla stessa origine di eventi. Ad esempio, puoi creare un gruppo di distribuzione per un set di funzioni Lambda avviate da API Gateway o da una regola Amazon. EventBridge È quindi possibile creare una distribuzione utilizzando CodeDeploy, che distribuisce la nuova versione della funzione serverless in un gruppo di distribuzione specificato.

CodeDeploy consente inoltre di definire una configurazione di distribuzione, che specifica le impostazioni per una distribuzione, come il tipo di distribuzione, la strategia di distribuzione e le regole di spostamento del traffico. Puoi utilizzare la strategia di distribuzione di Canary per distribuire la nuova versione della tua funzione serverless su una piccola percentuale di traffico e monitorare lo stato e le prestazioni della nuova versione prima di aumentare il traffico verso di essa.

Utilizzando CodeDeploy for serverless, puoi automatizzare il processo di distribuzione, ridurre il tempo e l'impegno necessari per rilasciare nuove versioni dell'applicazione e aumentare la stabilità e l'affidabilità delle tue funzioni serverless.

Amazon Elastic Container Service

Amazon Elastic Container Service (Amazon ECS) è un servizio di orchestrazione di container completamente gestito che supporta i contenitori Docker e consente di eseguire facilmente le applicazioni su un cluster gestito. Amazon ECS elimina la necessità di installare, gestire e scalare l'infrastruttura di gestione dei container e semplifica la creazione di ambienti con funzionalità di base AWS familiari come Security Groups, Elastic Load Balancing AWS Identity and Access Management (IAM).

Quando esegui applicazioni su Amazon ECS, puoi scegliere di fornire la potenza di calcolo sottostante per i tuoi contenitori con EC2 istanze Amazon o con AWS Fargateun motore di elaborazione serverless per contenitori. In entrambi i casi, Amazon ECS posiziona e ridimensiona automaticamente i contenitori sul cluster in base alle configurazioni definite dall'utente. Sebbene Amazon ECS non crei componenti dell'infrastruttura come Load Balancers o ruoli IAM per tuo conto, il servizio Amazon ECS fornisce una serie di soluzioni APIs per semplificare la creazione e l'uso di queste risorse in un cluster Amazon ECS.

Amazon ECS consente agli sviluppatori di avere un controllo diretto e preciso su tutti i componenti dell'infrastruttura, permettendo la creazione di architetture applicative personalizzate. Inoltre, Amazon ECS supporta diverse strategie di distribuzione per aggiornare le immagini dei contenitori delle applicazioni.

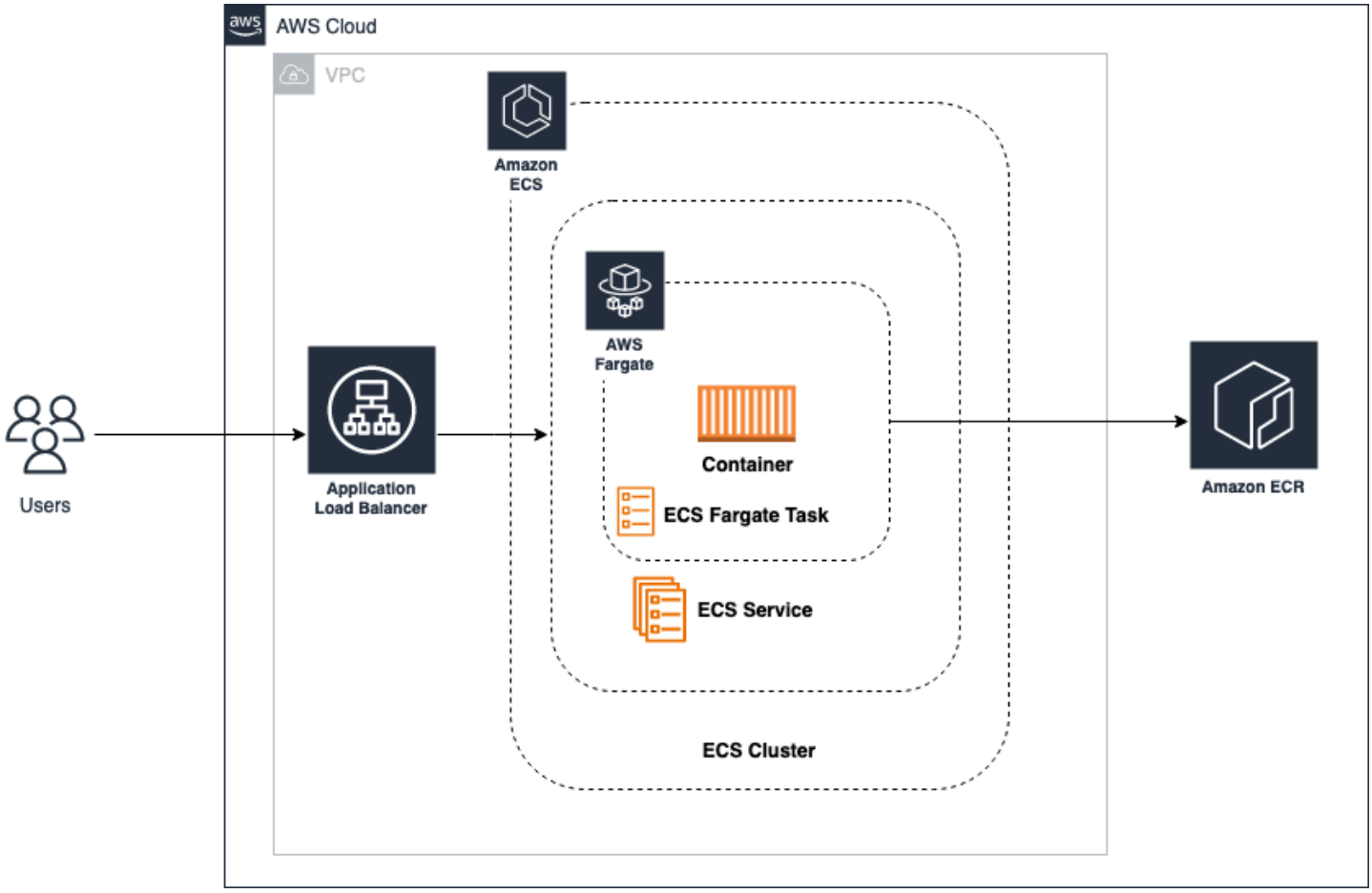
Tabella 4: caratteristiche di distribuzione di Amazon ECS

Funzionalità	Descrizione
Fornitura	Amazon ECS fornirà nuove istanze di container applicativi e risorse di calcolo in base a politiche di scalabilità e configurazioni Amazon ECS. Le risorse dell'infrastruttura come Load Balancers dovranno essere create al di fuori di Amazon ECS. Per ulteriori dettagli sui tipi di risorse che possono essere create con Amazon ECS, consulta Getting Started with Amazon ECS.
Configura	Amazon ECS supporta la personalizzazione delle risorse di calcolo create per eseguire

Funzionalità	Descrizione
	un'applicazione containerizzata, nonché le condizioni di runtime dei contenitori dell'appl icazione (ad esempio, variabili di ambiente, porte esposte, memoria/CPU riservate). La personalizzazione delle risorse di elaborazione sottostanti è disponibile solo se si utilizzano istanze Amazon EC2. Consulta Creazione di un cluster per maggiori dettagli su come personalizzare un cluster Amazon ECS per eseguire applicazioni containerizzate.
Implementazione	Amazon ECS supporta diverse strategie di distribuzione per le tue applicazioni container izzate. Consulta i tipi di distribuzione di Amazon ECS per maggiori dettagli sui tipi di processi di distribuzione supportati.
Dimensionare	Amazon ECS può essere utilizzato con policy di auto scaling per regolare automaticamente il numero di container in esecuzione nel cluster Amazon ECS. Consulta Service Auto Scaling per maggiori dettagli sulla configurazione della scalabilità automatica per le tue applicazioni container izzate su Amazon ECS.

Funzionalità	Descrizione
Monitoraggio	Amazon ECS supporta il monitoraggio di risorse di calcolo e contenitori di applicazioni con. CloudWatch Consulta Monitoring Amazon ECS per maggiori dettagli sui tipi di funzionalità di monitoraggio offerte da Amazon ECS.

Il diagramma seguente illustra l'uso di Amazon ECS per gestire una semplice applicazione containerizzata. In questo esempio, i componenti dell'infrastruttura vengono creati all'esterno di Amazon ECS e Amazon ECS viene utilizzato per gestire la distribuzione e il funzionamento dei contenitori di applicazioni nel cluster.



Caso d'uso di Amazon ECS

Note

- L'infrastruttura applicativa (inclusi i repository Amazon Elastic Container Registry (Amazon ECR), le configurazioni Amazon ECS e Load Balancers) viene fornita e gestita all'esterno della distribuzione Amazon ECS.
- Amazon ECS gestisce la distribuzione di contenitori di applicazioni in esecuzione all'interno del servizio Amazon ECS come attività che provengono da un registro di container come Amazon ECR.

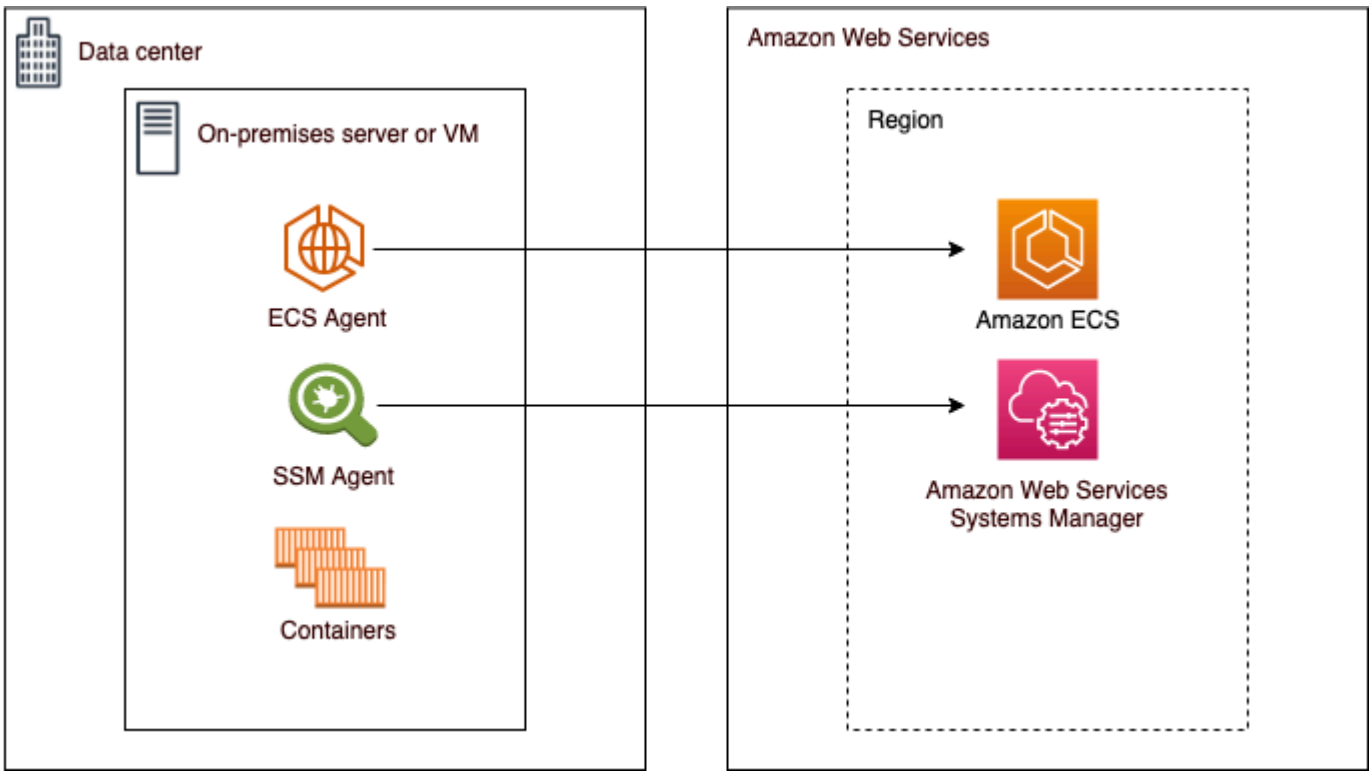
Amazon ECS supporta più tipi di istanze di container come Linux e Windows, nonché tipi di istanze esterne come una macchina virtuale (VM) locale con Amazon ECS Anywhere.

Amazon ECS Anywhere

Amazon ECS Anywhere ti consente di eseguire attività Amazon ECS ovunque, sia in locale che in altri ambienti cloud. Con Amazon ECS Anywhere, puoi distribuire e gestire facilmente applicazioni containerizzate sulla tua infrastruttura ibrida, mantenendo al contempo un'esperienza operativa coerente. Il servizio funziona estendendo la piattaforma Amazon ECS a qualsiasi ambiente, inclusi data center locali, uffici remoti e altri ambienti cloud. Ti consente di utilizzare gli stessi strumenti APIs e gli stessi familiari di Amazon ECS per distribuire e gestire contenitori in tutti i tuoi ambienti, senza doverti preoccupare dell'infrastruttura sottostante.

Amazon ECS Anywhere utilizza l'agente Amazon ECS per gestire la distribuzione e il ciclo di vita dei contenitori, consentendoti di utilizzare le stesse definizioni di attività e gli stessi file di configurazione di Amazon ECS utilizzati in. Cloud AWS Ciò può contribuire a semplificare il processo di distribuzione e gestione dei container nell'infrastruttura ibrida e a ridurre il tempo e l'impegno necessari per la configurazione e la gestione manuali.

Con Amazon ECS Anywhere, puoi anche sfruttare altri servizi AWS, come IAM e Amazon ECR AWS CloudFormation, per gestire le tue applicazioni containerizzate. Questo può contribuire a garantire che le tue applicazioni siano sicure, conformi e integrate con altri servizi AWS.



Amazon ECS Anywhere architecture

Amazon Elastic Container Service su AWS Outposts

Amazon ECS on AWS Outposts è un servizio AWS completamente gestito che ti consente di eseguire attività di Amazon ECS in locale, utilizzando APIs gli stessi strumenti utilizzati in. Cloud AWS Con Amazon ECS attivo AWS Outposts, puoi distribuire e gestire applicazioni containerizzate in modo coerente e familiare, indipendentemente dal fatto che tu le esegua in locale o nel cloud. AWS Outposts è un servizio completamente gestito che estende l'infrastruttura APIs, i servizi e gli strumenti AWS agli ambienti locali. Con Amazon ECS attivo AWS Outposts, puoi eseguire attività Amazon ECS su hardware dedicato alla tua organizzazione, senza doverti preoccupare dell'infrastruttura sottostante. Questo può aiutarti a garantire che le tue applicazioni siano distribuite in modo sicuro e conforme, consentendoti al contempo di sfruttare la flessibilità e la scalabilità del cloud.

Amazon ECS on AWS Outposts funziona distribuendo un set di servizi AWS e APIs nel tuo ambiente locale, il che ti consente di eseguire attività di Amazon ECS su hardware dedicato. Ciò include l'agente Amazon ECS, che gestisce la distribuzione e il ciclo di vita dei container, e l' AWS Outposts infrastruttura, che fornisce un ambiente sicuro e conforme per l'esecuzione di applicazioni containerizzate. Con Amazon ECS attivo AWS Outposts, puoi utilizzare lo stesso Amazon ECS APIs e gli stessi strumenti che usi in Cloud AWS, semplificando la distribuzione e la gestione di applicazioni containerizzate in modo coerente e familiare. Questo può aiutare a ridurre il tempo

e l'impegno necessari per la configurazione e la gestione manuali e a migliorare la coerenza e l'affidabilità dell'infrastruttura ibrida. Amazon ECS on si integra AWS Outposts anche con altri servizi AWS, come IAM e Amazon ECR AWS CloudFormation, per gestire le tue applicazioni containerizzate. Questo può contribuire a garantire che le tue applicazioni siano sicure, conformi e integrate con altri servizi AWS.

Amazon Elastic Kubernetes Service

Amazon Elastic Kubernetes Service (Amazon EKS) è un servizio conforme a Kubernetes completamente gestito e certificato che semplifica il processo di creazione, protezione, funzionamento e manutenzione dei cluster Kubernetes su AWS. Amazon EKS si integra con i principali servizi AWS come CloudWatch Auto Scaling Groups e IAM per fornire un'esperienza senza interruzioni per il monitoraggio, la scalabilità e il bilanciamento del carico delle applicazioni containerizzate.

Amazon EKS offre un piano di controllo scalabile e ad alta disponibilità per i carichi di lavoro Kubernetes. Quando esegui applicazioni su Amazon EKS, come con Amazon ECS, puoi scegliere di fornire la potenza di elaborazione sottostante per i tuoi contenitori con EC2 istanze Amazon o con. AWS Fargate

Amazon VPC Lattice è un servizio di rete di applicazioni completamente gestito integrato direttamente nell'infrastruttura di rete AWS che puoi utilizzare per connettere, proteggere e monitorare i tuoi servizi su più account e cloud privati virtuali (). VPCs Con Amazon EKS, puoi sfruttare VPC Lattice tramite l'uso del controller API AWS Gateway, un'implementazione dell'API Kubernetes Gateway. Utilizzando VPC Lattice, puoi configurare la connettività tra cluster con la semantica standard di Kubernetes in modo semplice e coerente.

Puoi utilizzare Amazon EKS con una delle seguenti opzioni di distribuzione:

- Amazon EKS Distro – Amazon EKS Distro è una distribuzione dello stesso software Kubernetes open source e delle dipendenze distribuite da Amazon EKS nel cloud. Amazon EKS Distro segue lo stesso ciclo di rilascio della versione Kubernetes di Amazon EKS e viene fornito come progetto open source. Per ulteriori informazioni, consulta Amazon EKS Distro.
- Amazon EKS attivo AWS Outposts: AWS Outposts abilita i servizi, l'infrastruttura e i modelli operativi AWS nativi nelle tue strutture locali. Amazon EKS on AWS Outposts, puoi scegliere di eseguire cluster estesi o locali. Con i cluster estesi, il piano di controllo di Kubernetes viene eseguito in un Regione AWS e i nodi vengono eseguiti. AWS Outposts Con i cluster locali, l'intero

cluster Kubernetes viene eseguito localmente, inclusi sia il piano di controllo Kubernetes che AWS Outposts i nodi.

- Amazon EKS Anywhere – Amazon EKS Anywhere è un'opzione di implementazione per Amazon EKS che consente di creare e gestire facilmente cluster Kubernetes On-Premise. Sia Amazon EKS che Amazon EKS Anywhere sono costruiti su Amazon EKS. Per ulteriori informazioni su Amazon EKS Anywhere, consulta Esecuzione di carichi di lavoro di container ibridi con Amazon EKS Anywhere, Panoramica di Amazon EKS Anywhere e Confronto tra Amazon EKS Anywhere e Amazon EKS.

Quando scegli le opzioni di implementazione da utilizzare per il cluster Kubernetes, considera quanto segue:

Tabella 5: Funzionalità di implementazione di Kubernetes

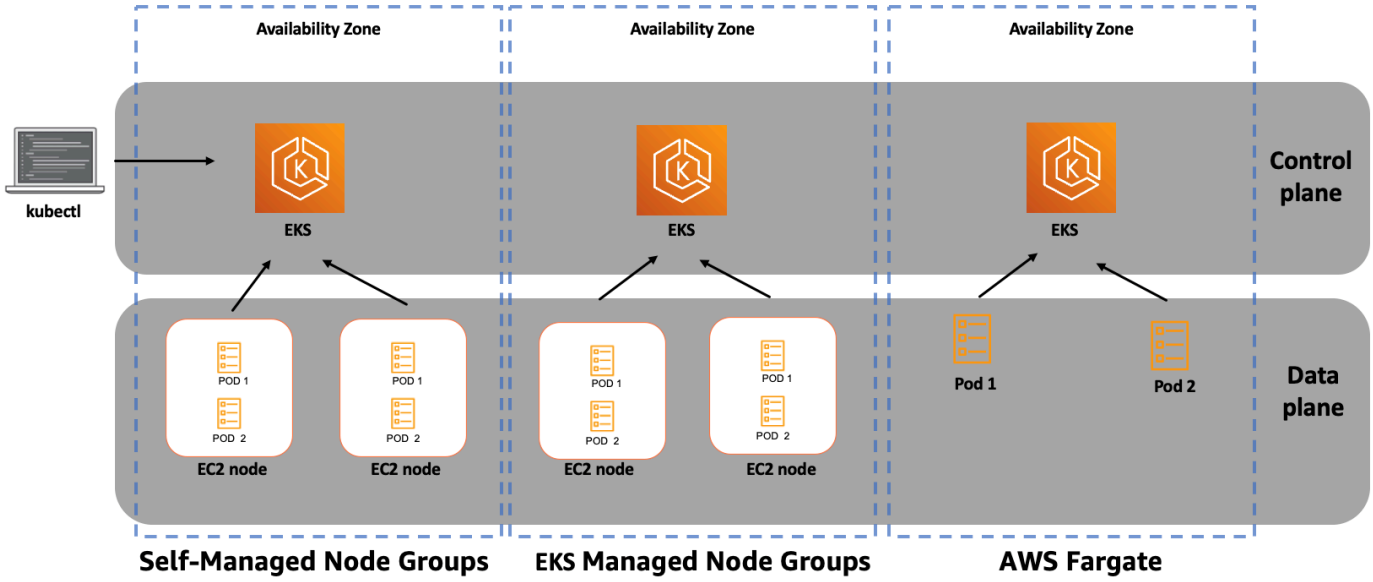
Funzionalità	Amazon EKS	Amazon EKS su AWS Outposts	Amazon EKS Anywhere	Amazon EKS Distro
Hardware	Fornito da AWS	Fornito da AWS	Fornito dall'utente	Fornito dall'utente
Posizione di implementazione	Cloud AWS	Il tuo data center	Il tuo data center	Il tuo data center
Posizione del piano di controllo Kubernetes	Cloud AWS	Cloud AWS o il tuo data center	Il tuo data center	Il tuo data center
Posizione del piano dati Kubernetes	Cloud AWS	Il tuo data center	Il tuo data center	Il tuo data center
Supporto	AWS supporto	AWS supporto	AWS supporto	Supporto della community OSS

Tabella 6: caratteristiche di distribuzione di Amazon EKS

Funzionalità	Descrizione
Fornitura	Amazon EKS fornisce determinate risorse per supportare le applicazioni containerizzate: • Load Balancer, se necessario • Risorse di calcolo o lavoratori (Amazon EKS supporta Windows e Linux) • Istanze o pod di Application Container Per ulteriori dettagli sul provisioning dei cluster Amazon EKS, consulta la sezione Guida introduttiva ad Amazon EKS.
Configura	Amazon EKS supporta la personalizzazione delle risorse di elaborazione (worker) se utilizzi EC2 istanze Amazon per fornire potenza di calcolo. Amazon EKS supporta anche la personalizzazione delle condizioni di runtime dei contenitori delle applicazioni (pod). Per maggiori dettagli, consulta la documenta zione sulla configurazione dei Worker Nodes e Fargate Pod.
Implementazione	Amazon EKS supporta le stesse strategie di implementazione di Kubernetes. Per ulteriori dettagli, consulta Scrittura di una specifica di implementazione di Kubernetes -> Strategia.
Dimensionare	Amazon EKS ridimensiona i lavoratori con Kubernetes Cluster Autoscaler e i pod con Kubernetes Horizontal Pod Autoscaler e Kubernetes Vertical Pod Autoscaler. Amazon EKS supporta anche Karpenter, un autoscale r di cluster Kubernetes open source, flessibile e ad alte prestazioni per contribuire a migliorar

Funzionalità	Descrizione
	e la disponibilità delle applicazioni e l'efficienza del cluster lanciando rapidamente risorse di calcolo della giusta dimensione in risposta al cambiamento del carico delle applicazioni.
Monitoraggio	I log del piano di controllo di Amazon EKS forniscono informazioni di controllo e diagnosti ca direttamente ai CloudWatch registri. Il piano di controllo di Amazon EKS si integra anche con AWS CloudTrail la registrazione delle azioni intraprese in Amazon EKS. Per ulteriori dettagli, consulta Logging and Monitoring Amazon EKS.

Amazon EKS consente alle organizzazioni di sfruttare strumenti e plug-in Kubernetes open source e può essere una buona scelta per le organizzazioni che migrano ad AWS con ambienti Kubernetes esistenti. Il diagramma seguente illustra l'uso di Amazon EKS per gestire un'applicazione containerizzata generale.



Amazon EKS use case

Amazon EKS Anywhere

Amazon EKS Anywhere ti consente di creare e gestire cluster Kubernetes sulla tua infrastruttura. Amazon EKS Anywhere si basa sui punti di forza di Amazon EKS Distro e fornisce software open source aggiornato e con patch in modo da poter disporre di un ambiente Kubernetes locale più affidabile di un'offerta Kubernetes autogestita.

Amazon EKS Anywhere crea un cluster Kubernetes in locale presso un provider selezionato. I provider supportati includono Bare Metal (tramite Tinkerbell) CloudStack e vSphere. Per gestire quel cluster, puoi eseguire i comandi di creazione ed eliminazione del cluster da un computer amministrativo Ubuntu o Mac.

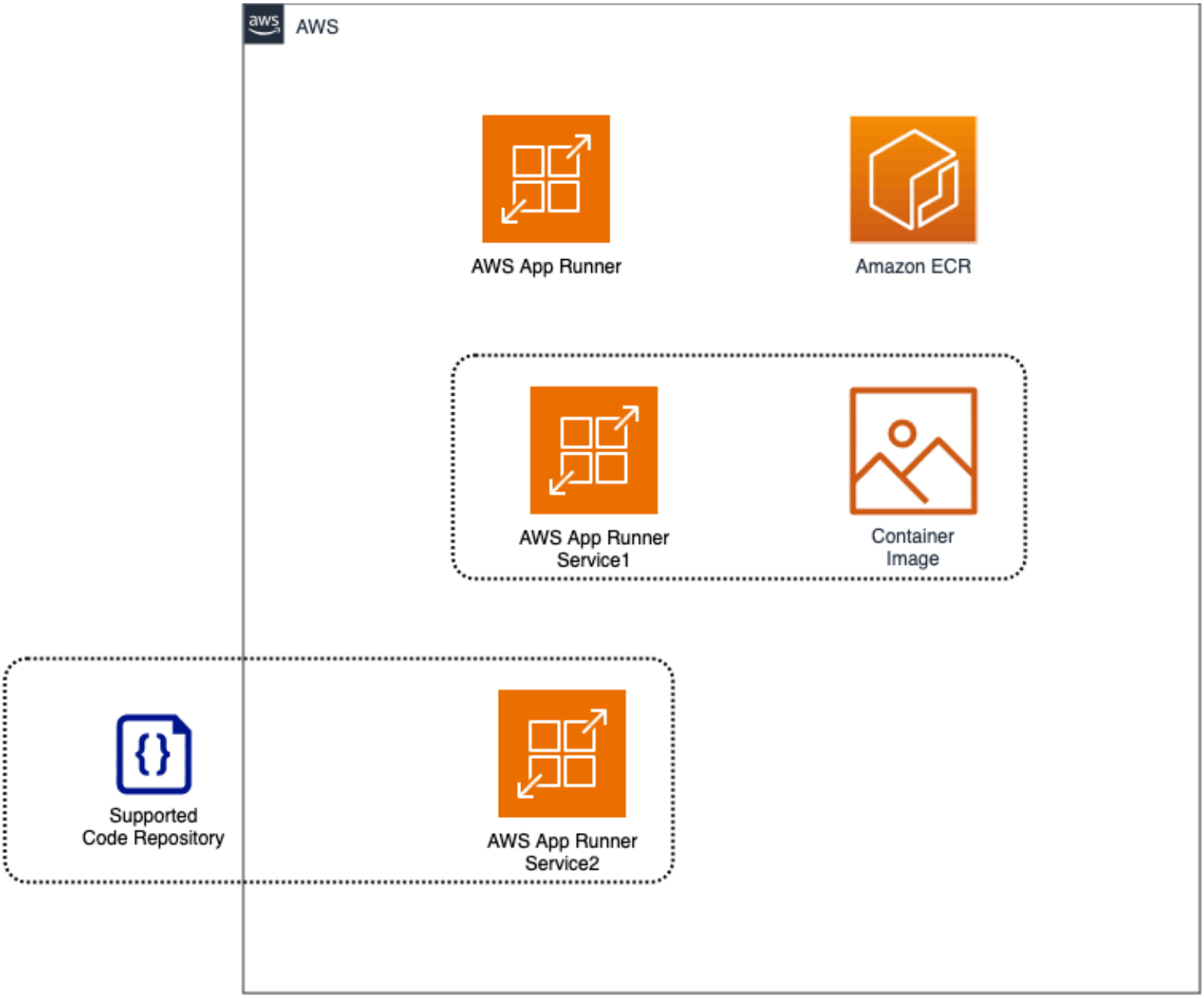
AWS App Runner

AWS App Runnerè un servizio di applicazioni container completamente gestito che consente di creare, distribuire ed eseguire applicazioni Web e servizi API containerizzati senza precedenti esperienze in materia di infrastrutture o container. App Runner si connette direttamente al tuo archivio di codice o immagini. Fornisce una pipeline di integrazione e distribuzione automatica con operazioni completamente gestite, alte prestazioni, scalabilità e sicurezza.

App Runner prende il codice sorgente o l'immagine sorgente da un repository e quindi crea e gestisce un servizio web in esecuzione per te in. Cloud AWS In genere, è necessario chiamare una sola azione di App Runner per creare `CreateService` il servizio. Con un archivio di immagini di origine, fornisci un'immagine del ready-to-use contenitore che App Runner può distribuire per eseguire il tuo servizio web. Con un repository di codice sorgente, fornisci il codice e le istruzioni per creare ed eseguire un servizio web e scegli come destinazione un ambiente di runtime specifico. App Runner supporta diverse piattaforme di programmazione, ognuna con uno o più runtime gestiti per le versioni principali della piattaforma. App Runner supporta immagini di container, runtime e framework web tra cui Node.js e Python. App Runner monitora il numero di richieste simultanee inviate all'applicazione e aggiunge automaticamente istanze aggiuntive in base al volume delle richieste. Se l'applicazione non riceve richieste in entrata, App Runner ridimensionerà i contenitori fino a ridurli a un'istanza fornita, un'istanza con limitazioni della CPU pronta a soddisfare le richieste in entrata in pochi millisecondi.

Al momento, App Runner può recuperare il codice sorgente da un GitHub repository o recuperare l'immagine sorgente da Amazon ECR nel tuo. Account AWS

Il diagramma seguente mostra una panoramica dell'architettura del servizio App Runner. Nel diagramma sono presenti due servizi di esempio: uno distribuisce il codice sorgente da GitHub e l'altro distribuisce un'immagine sorgente da Amazon ECR.



App Runner use case

App Runner supporta lo sviluppo completo, incluse applicazioni web frontend e backend che utilizzano i protocolli HTTP e HTTPS. Queste applicazioni includono servizi API, servizi web di backend e siti Web. App Runner supporta immagini di container, runtime e framework web tra cui Node.js e Python.

Amazon Lightsail

Amazon Lightsail è un servizio cloud semplice ed economico che semplifica l'implementazione e la gestione delle applicazioni nel cloud per piccole imprese, startup e privati. Fornisce un'interfaccia intuitiva che astrae gran parte della gestione dell'infrastruttura sottostante e semplifica l'avvio e l'esecuzione di applicazioni nel cloud. Con Lightsail, puoi implementare e gestire rapidamente server privati virtuali (VPS), database e istanze di archiviazione. Il servizio fornisce istanze preconfigurate ottimizzate per vari carichi di lavoro, come Drupal e Joomla, tra gli altri WordPress. Questo può aiutare a ridurre il tempo e lo sforzo necessari per configurare e configurare l'ambiente. Lightsail fornisce anche un sistema di bilanciamento del carico integrato e una scalabilità automatica, che consentono di gestire le variazioni della domanda di traffico senza interventi manuali. Il servizio fornisce anche monitoraggio e avvisi, così puoi rimanere aggiornato sullo stato e sulle prestazioni delle tue applicazioni.

Uno dei principali vantaggi di Lightsail è la sua semplicità e facilità d'uso. Il servizio è progettato per essere accessibile agli utenti con un'esperienza minima nel cloud computing, il che lo rende una buona opzione per le piccole imprese o i privati che desiderano iniziare rapidamente a utilizzare il cloud. Inoltre, Lightsail è conveniente, con prezzi prevedibili che includono elaborazione, archiviazione e trasferimento dati.

Contenitori Amazon Lightsail

Amazon Lightsail Containers è un servizio di container AWS completamente gestito che semplifica la distribuzione e la gestione di applicazioni containerizzate nel cloud. Fornisce un modo semplice ed economico per avviare ed eseguire container utilizzando i più diffusi strumenti di gestione dei container, come Docker e Kubernetes.

Lightsail Containers fornisce un ambiente integrato per la creazione, il test e la distribuzione di applicazioni containerizzate. Semplifica il processo di implementazione e gestione dei container fornendo un'interfaccia intuitiva che astrae gran parte della gestione dell'infrastruttura sottostante.

Con Lightsail Containers, puoi distribuire le tue applicazioni containerizzate su un VPC in pochi clic. Il servizio fornisce immagini di contenitori preconfigurate per i linguaggi di programmazione più diffusi, come Node.js, Python, Ruby e Java. Questo può aiutare a ridurre il tempo e l'impegno necessari per configurare e configurare l'ambiente container.

Lightsail Containers fornisce anche un sistema di bilanciamento del carico integrato in grado di distribuire automaticamente il traffico tra le istanze dei container, migliorando la disponibilità e la

scalabilità delle applicazioni. Inoltre, il servizio fornisce il ridimensionamento automatico delle istanze dei container, consentendoti di gestire le variazioni della domanda di traffico senza interventi manuali.

Con Lightsail Containers, puoi monitorare le prestazioni delle tue applicazioni containerizzate utilizzando metriche e log integrati. Puoi anche integrarti con altri servizi AWS, come Amazon S3, Amazon RDS e AWS CodePipeline, per creare una pipeline CI/CD completamente automatizzata e integrata per le tue applicazioni containerizzate.

Servizio Red Hat OpenShift su AWS

Servizio Red Hat OpenShift su AWS(ROSA) è un servizio gestito disponibile tramite la Console di gestione AWS. Con ROSA, come OpenShift utente Red Hat, puoi creare, scalare e gestire applicazioni containerizzate su AWS. Puoi usare ROSA per creare cluster Kubernetes utilizzando Red Hat OpenShift APIs e gli strumenti e avere accesso all'intera gamma e profondità dei servizi AWS. ROSA semplifica lo spostamento dei carichi di OpenShift lavoro Red Hat locali su AWS e offre una stretta integrazione con altri servizi AWS. Puoi anche accedere alle OpenShift licenze, alla fatturazione e all'assistenza Red Hat direttamente tramite AWS.

Ogni cluster ROSA è dotato di un piano di controllo e di nodi di calcolo completamente gestiti. L'installazione, la gestione, la manutenzione e gli upgrade vengono eseguiti da Red Hat SRE con il supporto congiunto di Red Hat e Amazon. Sono disponibili anche servizi di cluster (come registrazione, metriche e monitoraggio). Solo i lavoratori Red Hat Enterprise Linux CoreOS (RHCOS) sono supportati da ROSA.

ROSA si integrerà con una gamma di servizi di elaborazione, storage, database, analisi, machine learning, networking, dispositivi mobili e varie applicazioni di AWS, che consentiranno ai clienti di beneficiare del solido portafoglio di servizi AWS scalabili su richiesta in tutto il mondo. Questi servizi nativi AWS saranno direttamente accessibili per distribuire e scalare rapidamente i servizi tramite la stessa interfaccia di gestione.

Zone locali AWS

Una AWS Local Zone è un'estensione Regione AWS di una zona geografica molto vicina ai tuoi utenti. Le zone locali hanno le loro connessioni a Internet e supportano AWS Direct Connect. Le risorse create in una zona locale possono servire gli utenti locali con comunicazioni a latenza molto bassa. Una zona locale è rappresentata da un codice regionale seguito da un identificatore che indica la posizione (ad esempio, us-west-2-lax-1a).

Amazon ECS supporta carichi di lavoro che utilizzano Local Zones quando è richiesta una bassa latenza o l'elaborazione locale dei dati. Il piano di controllo di Amazon ECS funzionerà sempre in. Regione AWS

Amazon EKS supporta alcune risorse nelle zone locali, Ciò include EC2 nodi Amazon autogestiti, volumi Amazon EBS e Application Load Balancer. Il piano di controllo Kubernetes gestito da Amazon EKS viene sempre eseguito nella Regione AWS. Il piano di controllo Kubernetes gestito da Amazon EKS non può essere eseguito nella zona locale. Poiché le zone locali vengono visualizzate come sottoreti all'interno del VPC, Kubernetes considera le risorse della zona locale come parte di tale sottorete.

AWS Wavelength

AWS Wavelengthè un'infrastruttura AWS che consente di distribuire carichi di lavoro più vicini a utenti e dispositivi connessi al 5G. Puoi usare Wavelength per distribuire EC2 istanze Amazon, cluster Amazon EKS e una suite di soluzioni partner supportate disponibili su AWS Marketplace. Le Wavelength Zone sono data center logicamente isolati all'interno delle reti dei provider di telecomunicazioni ricollegati alla regione AWS tramite connettività ridondante, a bassa latenza e ad alto throughput.

Alcune delle caratteristiche principali di Wavelength includono la possibilità di creare EC2 istanze Amazon, volumi Amazon EBS e sottoreti Amazon VPC e gateway carrier nelle zone Wavelength. Puoi anche utilizzare servizi che orchestrano o funzionano con Amazon, EC2 Amazon EBS e Amazon VPC come Amazon Auto EC2 Scaling, cluster Amazon EKS, cluster Amazon ECS, Amazon EC2 Systems Manager CloudWatch, AWS CloudTrail Amazon e Application Load Balancer. AWS CloudFormation I servizi Wavelength fanno parte di un VPC connesso tramite una connessione affidabile e ad alta larghezza di banda a una regione AWS per un facile accesso a servizi tra cui Amazon DynamoDB e Amazon Relational Database Service (Amazon RDS).

Servizi di distribuzione aggiuntivi

Amazon Simple Storage Service (Amazon S3) può essere utilizzato come server Web per contenuti statici e applicazioni a pagina singola (SPA). In combinazione con Amazon CloudFront per aumentare le prestazioni nella distribuzione di contenuti statici, l'utilizzo di Amazon S3 può essere un modo semplice e potente per distribuire e aggiornare contenuti statici. Maggiori dettagli su questo approccio sono disponibili in Hosting Static Websites on AWS whitepaper.

AWS Proton

AWS Proton è un servizio completamente gestito che semplifica e automatizza il processo di distribuzione e gestione di microservizi e applicazioni basate su contenitori. Fornisce un'esperienza di distribuzione unificata e coerente che si integra con DevOps gli strumenti e i servizi più diffusi, semplificando la gestione e semplificando lo sviluppo delle applicazioni. Proton consente agli sviluppatori di definire e creare componenti applicativi, come infrastruttura, codice e pipeline, come modelli riutilizzabili. Questi modelli possono essere utilizzati per creare più ambienti, come sviluppo, test e produzione, e possono essere condivisi tra team o organizzazioni. Questo approccio aiuta a ridurre la complessità dell'implementazione e della gestione dei microservizi e delle applicazioni basate su contenitori, operazioni che possono richiedere molto tempo e sono soggette a errori.

AWS Proton fornisce modelli predefiniti per tipi comuni di microservizi, come applicazioni Web e database APIs, che possono essere personalizzati per soddisfare esigenze specifiche. Si integra inoltre con DevOps strumenti popolari come AWS CodePipeline, AWS e AWS CodeCommit CodeBuild, per consentire flussi di lavoro di integrazione e distribuzione continui (CI/CD).

Utilizzando AWS Proton, gli sviluppatori possono ridurre il tempo e lo sforzo necessari per distribuire e gestire microservizi e applicazioni basate su contenitori. Questo approccio consente ai team di concentrarsi sullo sviluppo e sul miglioramento delle proprie applicazioni, anziché dedicare tempo al processo di distribuzione e gestione.

AWS App2Container

AWS App2Containerè uno strumento da riga di comando per la migrazione e la modernizzazione delle applicazioni Web Java e.NET in formato contenitore. App2Container analizza e crea un inventario di applicazioni in esecuzione su bare metal, macchine virtuali, EC2 istanze Amazon o nel cloud. Basta selezionare l'applicazione che si desidera containerizzare e App2Container impacchetta gli elementi dell'applicazione e le dipendenze identificate in immagini dei container, configura le porte di rete e genera le definizioni del task ECS e dei pod Kubernetes. App2Container identifica le applicazioni ASP.NET e Java supportate in esecuzione in una macchina virtuale per creare un inventario completo di tutte le applicazioni presenti nell'ambiente. App2Container può containerizzare le applicazioni Web ASP.NET in esecuzione in IIS su Windows o le applicazioni Java in esecuzione su Linux, autonome o su server di applicazioni come Apache Tomcat, Springboot JBoss, IBM Websphere e Oracle Weblogic.

AWS Copilot

AWS Copilot è un'interfaccia a riga di comando (CLI) che puoi utilizzare per avviare e gestire rapidamente applicazioni containerizzate su AWS. Semplifica l'esecuzione di applicazioni su Amazon ECS, Fargate e App Runner. AWS Copilot attualmente supporta sistemi Linux, macOS e Windows. Copilot consente di utilizzare modelli di servizio, come un servizio Web con bilanciamento del carico, per fornire l'infrastruttura, distribuirla in più ambienti come test o produzione e persino utilizzare una AWS CodePipeline pipeline di rilascio per distribuzioni automatizzate.

AWS Serverless Application Model

Il AWS Serverless Application Model(AWS SAM) è un framework open source per la creazione di applicazioni serverless. Fornisce una sintassi abbreviata per esprimere funzioni APIs, database e mappature delle sorgenti degli eventi. Con solo poche righe per risorsa, puoi definire l'applicazione che desideri e modellarla usando YAML. Durante la distribuzione, SAM trasforma ed espande la sintassi SAM in sintassi CloudFormation AWS, consentendoti di creare applicazioni serverless più velocemente.

La AWS SAM CLI è uno strumento a riga di comando open source che semplifica lo sviluppo, il test e la distribuzione di applicazioni serverless su AWS. È un'interfaccia a riga di comando per la creazione di applicazioni serverless utilizzando la specifica AWS SAM, che è un'estensione di AWS. CloudFormation

La AWS SAM CLI consente agli sviluppatori di definire e testare le proprie applicazioni serverless localmente prima di distribuirle su AWS. Fornisce un ambiente di test locale che simula AWS Lambda e API Gateway, consentendo agli sviluppatori di testare il codice e le configurazioni prima di distribuirli nel cloud.

La AWS SAM CLI include anche una serie di funzioni utili, come la distribuzione automatica del codice, la registrazione e le funzionalità di debug. Consente agli sviluppatori di creare, impacchettare e distribuire le proprie applicazioni con un solo comando, riducendo il tempo e lo sforzo necessari per distribuire e gestire applicazioni serverless.

Inoltre, l'AWS SAM CLI fornisce supporto per vari linguaggi di programmazione, tra cui Node.js, Python, Java e.NET Core, tra gli altri. Ciò consente agli sviluppatori di utilizzare il linguaggio e gli strumenti di programmazione preferiti per creare e distribuire applicazioni serverless.

AWS SAM CLI si integra con altri servizi AWS, come AWS e CodePipeline AWS CodeBuild, per fornire una pipeline CI/CD completamente automatizzata e integrata per applicazioni serverless.

Consente inoltre agli sviluppatori di utilizzare altri servizi AWS, come Amazon S3, Amazon DynamoDB e Amazon SNS, come parte delle loro applicazioni serverless.

AWS Cloud Development Kit (AWS CDK)

The AWS Cloud Development Kit (AWS CDK)(AWS CDK) è un framework di sviluppo software open source per definire l'infrastruttura cloud come codice con linguaggi di programmazione moderni e distribuirla tramite AWS CloudFormation. AWS Cloud Development Kit (AWS CDK) accelera lo sviluppo del cloud utilizzando linguaggi di programmazione comuni per modellare le tue applicazioni. L'AWS CDK consente di creare applicazioni affidabili, scalabili ed economiche nel cloud con la notevole potenza espressiva di un linguaggio di programmazione.

Pensa all'AWS CDK come a un toolkit incentrato sugli sviluppatori che sfrutta tutta la potenza dei moderni linguaggi di programmazione per definire la tua infrastruttura AWS come codice. Quando le applicazioni AWS CDK vengono eseguite, vengono compilate in modelli CloudFormation JSON/YAML completi che vengono poi inviati al servizio per il provisioning. CloudFormation Grazie all'utilizzo di AWS CDK CloudFormation, puoi comunque usufruire di tutti i vantaggi offerti CloudFormation , come la distribuzione sicura, il rollback automatico e il rilevamento delle deviazioni.

Questo approccio offre molti vantaggi, tra cui:

- Crea con costrutti di alto livello che forniscono automaticamente impostazioni predefinite sensibili e sicure per le tue risorse AWS, definendo più infrastruttura con meno codice.
- Usa idiomi di programmazione come parametri, condizionali, loop, composizione ed ereditarietà per modellare la progettazione del tuo sistema partendo da elementi costitutivi forniti da AWS e altri.
- Raccogli l'infrastruttura, il codice dell'applicazione e la configurazione in un unico posto, assicurandoti di disporre di un sistema completo e distribuibile sul cloud in ogni fase fondamentale.
- Utilizza pratiche di ingegneria del software come revisioni del codice, test unitari e controllo del codice sorgente per rendere la tua infrastruttura più robusta.
- AWS Solutions Constructs è un'estensione di libreria open source di AWS CDK. AWS Solutions Constructs offre una raccolta di modelli di architettura controllati e multiservizi creati utilizzando le migliori pratiche stabilite da AWS Well-Architected Framework.

AWS Serverless Application Model e AWS CDK astraggono entrambi l'infrastruttura AWS come codice, semplificando la definizione dell'infrastruttura cloud. AWS SAM si concentra specificamente su casi d'uso e architetture serverless e consente di definire l'infrastruttura in modelli JSON/YAML

compatti e dichiarativi. AWS CDK offre un'ampia copertura su tutti i servizi AWS e consente di definire l'infrastruttura cloud in linguaggi di programmazione moderni.

Amazon EC2 Image Builder

EC2 Image Builder semplifica la creazione, il test e la distribuzione di immagini di macchine virtuali e container da utilizzare su AWS o in locale. La conservazione delle immagini delle macchine virtuali e dei container up-to-date può richiedere molto tempo, molte risorse e può essere soggetta a errori. Attualmente, i clienti aggiornano e creano istantanee manualmente VMs o dispongono di team che creano script di automazione per la manutenzione delle immagini. Image Builder riduce in modo significativo lo sforzo di protezione delle immagini up-to-date fornendo un'interfaccia grafica semplice, automazione integrata e impostazioni di sicurezza fornite da AWS. Con Image Builder, non sono necessari passaggi manuali per l'aggiornamento di un'immagine né è necessario creare una pipeline di automazione personalizzata. Image Builder è offerto gratuitamente, a parte il costo delle risorse AWS sottostanti utilizzate per creare, archiviare e condividere le immagini.

EC2 Image Builder può contribuire a semplificare le distribuzioni su AWS semplificando il processo di creazione e gestione di immagini personalizzate da utilizzare con Amazon EC2, container e server locali. Il servizio offre un modo semplificato e flessibile per creare e gestire immagini personalizzate, con pipeline di compilazione automatizzate che consentono di semplificare il processo di creazione e gestione delle immagini.

EC2 Image Builder fornisce un'interfaccia intuitiva che elimina gran parte della gestione dell'infrastruttura sottostante, semplificando la creazione e la gestione di immagini personalizzate per gli sviluppatori. Con EC2 Image Builder, gli sviluppatori possono specificare il sistema operativo, le applicazioni e i pacchetti che desiderano includere nell'immagine e il servizio automatizza il processo di creazione e test dell'immagine, inclusi aggiornamenti, patch e correzioni di sicurezza. Le pipeline di compilazione automatizzate consentono agli sviluppatori di semplificare il processo di creazione e gestione delle immagini, riducendo il tempo e l'impegno necessari per la creazione e il test manuali delle immagini. Ciò può contribuire a migliorare la coerenza, ridurre gli errori e garantire che le immagini siano up-to-date sicure e conformi.

Di seguito sono riportati alcuni dei vantaggi di EC2 Image Builder:

- Creazione semplificata di EC2 immagini: Image Builder offre un modo semplificato e flessibile per creare immagini personalizzate da utilizzare con EC2 Amazon, container e server locali. Questo può aiutare a ridurre il tempo e l'impegno necessari per creare e gestire immagini personalizzate e consentirti di concentrarti su altri aspetti della distribuzione, come lo sviluppo e il test delle applicazioni.

- Pipeline automatizzate di creazione di EC2 immagini: Image Builder fornisce pipeline automatizzate per la creazione, il test e la distribuzione di immagini personalizzate, che possono aiutare a semplificare il processo di creazione e gestione delle immagini. Questo può contribuire a garantire che le immagini siano up-to-date sicure e conformi e a ridurre il tempo e l'impegno necessari per la creazione e il test manuali delle immagini.
- Integrazione con i servizi AWS: EC2 Image Builder si integra con altri servizi AWS, come Amazon Elastic Container Registry (ECR) e Amazon Elastic Kubernetes Service (EKS), per consentirti di creare immagini personalizzate da utilizzare con i contenitori. Questo può aiutarti a semplificare il processo di creazione e distribuzione dei container, consentendoti di creare immagini personalizzate che includono applicazioni, librerie e configurazioni.
- Creazione flessibile di EC2 immagini: Image Builder offre un modo flessibile per creare immagini personalizzate, che consente di specificare il sistema operativo, le applicazioni e i pacchetti da includere nell'immagine. Questo può contribuire a garantire che le immagini siano personalizzate in base al caso d'uso e ai requisiti specifici e a ridurre il rischio di errori o incompatibilità durante l'implementazione.
- Maggiore sicurezza e conformità delle EC2 immagini: Image Builder consente di automatizzare i test delle immagini, comprese le scansioni di vulnerabilità e conformità, per garantire che le immagini siano sicure e conformi. Ciò può contribuire a ridurre il rischio di violazioni della sicurezza e a migliorare la conformità, oltre a consentire di distribuire le applicazioni con sicurezza.

Strategie di distribuzione

Oltre a selezionare gli strumenti giusti per aggiornare il codice dell'applicazione e l'infrastruttura di supporto, l'implementazione dei giusti processi di implementazione è una parte fondamentale di una soluzione di distribuzione completa e ben funzionante. I processi di distribuzione scelti per aggiornare l'applicazione possono dipendere dall'equilibrio desiderato tra controllo, velocità, costi, tolleranza al rischio e altri fattori.

Ogni servizio di distribuzione AWS supporta una serie di strategie di distribuzione. Questa sezione fornirà una panoramica delle strategie di distribuzione generiche che possono essere utilizzate con la tua soluzione di distribuzione.

Prebaking vs. bootstrap AMIs

Se la tua applicazione si basa principalmente sulla personalizzazione o sulla distribuzione di applicazioni su EC2 istanze Amazon, puoi ottimizzare le distribuzioni attraverso pratiche di bootstrap e prebaking.

L'installazione dell'applicazione, delle dipendenze o delle personalizzazioni ogni volta che viene lanciata EC2 un'istanza Amazon si chiama bootstrap di un'istanza. Se hai un'applicazione complessa o sono necessari download di grandi dimensioni, ciò può rallentare le implementazioni e gli eventi di scalabilità.

Un'Amazon Machine Image (AMI) fornisce le informazioni necessarie per avviare un'istanza (sistemi operativi, volumi di archiviazione, autorizzazioni, pacchetti software, ecc.). Puoi avviare più istanze identiche da una singola AMI. Ogni volta che viene lanciata un' EC2 istanza, si seleziona l'AMI da utilizzare come modello. La precottura è il processo di incorporamento di una parte significativa degli artefatti dell'applicazione all'interno di un'AMI.

La precottura dei componenti dell'applicazione in un'AMI può velocizzare i tempi di avvio e rendere operativa un'istanza Amazon. EC2 Le pratiche di precottura e avvio possono essere combinate durante il processo di distribuzione per creare rapidamente nuove istanze personalizzate in base all'ambiente corrente.

Distribuzioni blu/verde

Una strategia blue/green deployment is a deployment strategy in which you create two separate, but identical environments. One environment (blue) is running the current application version and one

environment (green) is running the new application version. Using a blue/green di implementazione aumenta la disponibilità delle applicazioni e riduce i rischi di implementazione semplificando il processo di rollback in caso di errore dell'implementazione. Una volta completati i test sull'ambiente verde, il traffico delle applicazioni in tempo reale viene indirizzato all'ambiente verde e l'ambiente blu diventa obsoleto.

Diversi servizi di distribuzione AWS supportano strategie di distribuzione blu/green, tra cui Elastic Beanstalk OpsWorks,, CloudFormation e Amazon ECS. CodeDeploy Fai riferimento a Blue/Green Deployments on AWS per maggiori dettagli e strategie per l'implementazione di processi di distribuzione blu/green per la tua applicazione.

Distribuzioni in sequenza

Una distribuzione progressiva è una strategia di distribuzione che sostituisce lentamente le versioni precedenti di un'applicazione con nuove versioni di un'applicazione sostituendo completamente l'infrastruttura su cui l'applicazione è in esecuzione. Ad esempio, in una distribuzione continua in Amazon ECS, i contenitori che eseguono versioni precedenti dell'applicazione verranno sostituiti one-by-one con contenitori che eseguono nuove versioni dell'applicazione.

Una distribuzione progressiva è generalmente più veloce di una blue/green deployment; however, unlike a blue/green distribuzione, in una distribuzione progressiva non c'è isolamento ambientale tra la vecchia e la nuova versione dell'applicazione. Ciò consente di completare più rapidamente le implementazioni in sequenza, ma aumenta anche i rischi e complica il processo di rollback in caso di errore dell'implementazione.

Le strategie di implementazione progressiva possono essere utilizzate con la maggior parte delle soluzioni di implementazione. Consulta CloudFormationUpdate Policies per ulteriori informazioni sulle distribuzioni in sequenza con CloudFormation; Rolling Updates with Amazon ECS per maggiori dettagli sulle distribuzioni in sequenza con Amazon ECS; Elastic Beanstalk Rolling Environment Configuration Updates per maggiori dettagli sulle distribuzioni in sequenza con Elastic Beanstalk; e Using a Rolling Deployment in per maggiori dettagli sulle distribuzioni in sequenza con. AWS OpsWorks OpsWorks

Implementazioni Canary

Le implementazioni Canary sono un tipo di strategia di implementazione blu/verde più avversa al rischio. Questa strategia prevede un approccio graduale in cui il traffico viene spostato verso una nuova versione dell'applicazione in due incrementi. Il primo incremento è costituito da una piccola

percentuale del traffico, denominata gruppo delle canarie. Questo gruppo viene utilizzato per testare la nuova versione e, se ha esito positivo, il traffico viene spostato sulla nuova versione nel secondo incremento.

Le implementazioni di Canary possono essere implementate in due fasi o in modo lineare. Nell'approccio in due fasi, il nuovo codice applicativo viene distribuito ed esposto per la prova. Una volta accettato, viene distribuito al resto dell'ambiente o in modo lineare. L'approccio lineare prevede l'aumento incrementale del traffico verso la nuova versione dell'applicazione fino a quando tutto il traffico fluirà verso la nuova versione.

Distribuzioni locali

Una distribuzione sul posto è una strategia di distribuzione che aggiorna la versione dell'applicazione senza sostituire alcun componente dell'infrastruttura. In una distribuzione sul posto, la versione precedente dell'applicazione su ogni risorsa di elaborazione viene interrotta, l'applicazione più recente viene installata e la nuova versione dell'applicazione viene avviata e convalidata. Ciò consente alle implementazioni delle applicazioni di procedere con il minimo disturbo all'infrastruttura sottostante.

Una distribuzione sul posto consente di distribuire l'applicazione senza creare nuova infrastruttura; tuttavia, la disponibilità dell'applicazione può risentirne durante queste implementazioni. Questo approccio riduce inoltre al minimo i costi di infrastruttura e il sovraccarico di gestione associato alla creazione di nuove risorse.

Per ulteriori dettagli sull'utilizzo delle strategie di implementazione sul posto con, consulta la sezione Panoramica di un'implementazione sul posto con. CodeDeploy

Combinazione dei servizi di implementazione

Non esiste una soluzione di distribuzione adatta a tutti in AWS. Nel contesto della progettazione di una soluzione di distribuzione, è importante considerare il tipo di applicazione in quanto ciò può determinare quali servizi AWS sono più appropriati. Per fornire funzionalità complete per il provisioning, la configurazione, la distribuzione, la scalabilità e il monitoraggio dell'applicazione, è spesso necessario combinare più servizi di distribuzione

Uno schema comune per le applicazioni su AWS consiste nell'utilizzare CloudFormation (e le sue estensioni) per gestire l'infrastruttura generica e utilizzare una soluzione di distribuzione più specializzata per la gestione degli aggiornamenti delle applicazioni. Nel caso di un'applicazione

containerizzata, CloudFormation potrebbe essere utilizzata per creare l'infrastruttura dell'applicazione e Amazon ECS e Amazon EKS potrebbero essere utilizzati per il provisioning, la distribuzione e il monitoraggio dei container.

I servizi di distribuzione AWS possono anche essere combinati con servizi di distribuzione di terze parti. Ciò consente alle organizzazioni di integrare facilmente i servizi di distribuzione AWS nei CI/CD pipelines or infrastructure management solutions. For example, OpsWorks can be used to synchronize configurations between on-premises and AWS nodes, and CodeDeploy can be used with a number of third-party CI/CD servizi esistenti come parte di una pipeline completa.

Conclusioni

AWS fornisce una serie di strumenti per semplificare e automatizzare il provisioning dell'infrastruttura e la distribuzione delle applicazioni; ogni servizio di distribuzione offre funzionalità diverse per la gestione delle applicazioni. Per creare un'architettura di distribuzione efficace, valuta le funzionalità disponibili di ciascun servizio in base alle esigenze della tua applicazione e della tua organizzazione.

Collaboratori

Hanno collaborato alla stesura del presente documento:

- Manikandan Chandrasekaran, tecnico principale
- Anil Nadiminti, architetto senior delle soluzioni
- Bryant Bost, consulente AWS ProServe

Approfondimenti

Per ulteriori informazioni, consulta:

- Pagina dei white paper di AWS
- Introduzione a DevOps on AWS - Strategie di distribuzione

Revisioni del documento

Per ricevere una notifica sugli aggiornamenti del presente whitepaper, iscriviti al feed RSS.

Modifica	Descrizione	Data
Aggiornamento del whitepaper	Completamente aggiornato per i servizi e le strategie di implementazione più recenti	31 maggio 2024
Aggiornamento secondario	La sezione Blue/Green Deployments è stata rivista per maggiore chiarezza.	8 aprile 2021
Aggiornamento del whitepaper	Aggiornato con i servizi e le funzionalità più recenti.	3 giugno 2020
Pubblicazione iniziale	Whitepaper pubblicato per la prima volta	1 marzo 2015

Note

I clienti sono responsabili della propria valutazione indipendente delle informazioni contenute nel presente documento. Questo documento: (a) è solo a scopo informativo, (b) rappresenta le attuali offerte e pratiche di prodotti AWS, che sono soggette a modifiche senza preavviso, e (c) non crea alcun impegno o garanzia da parte di AWS e delle sue affiliate, fornitori o licenzianti. I prodotti o i servizi AWS sono forniti «così come sono» senza garanzie, dichiarazioni o condizioni di alcun tipo, esplicite o implicite. Le responsabilità di AWS nei confronti dei propri clienti sono definite dai contratti AWS e il presente documento non costituisce parte né modifica qualsivoglia contratto tra AWS e i suoi clienti.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.