



Whitepaper da AWS

# Visão geral das opções de implantação na AWS



# Visão geral das opções de implantação na AWS: Whitepaper da AWS

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestígie a Amazon. Todas as outras marcas comerciais que não são propriedade da Amazon pertencem aos respectivos proprietários, os quais podem ou não ser afiliados, estar conectados ou ser patrocinados pela Amazon.

---

# Table of Contents

Resumo .....	1
Resumo .....	1
Introdução .....	2
Serviços de implantação da AWS .....	3
AWS CloudFormation .....	3
AWS Elastic Beanstalk .....	6
AWS CodeDeploy .....	10
AWS CodeDeploy para AWS Lambda .....	13
Amazon Elastic Container Service .....	13
Amazon ECS Anywhere .....	17
Amazon Elastic Container Service em AWS Outposts .....	18
Amazon Elastic Kubernetes Service .....	18
Amazon EKS Anywhere .....	22
AWS App Runner .....	23
Amazon Lightsail .....	24
Contêineres Amazon Lightsail .....	25
Serviço Red Hat OpenShift na AWS .....	26
Zonas locais da AWS .....	26
AWS Wavelength .....	27
Serviços adicionais de implantação .....	27
Amazon Simple Storage Service .....	27
AWS Proton .....	28
AWS App2Container .....	28
AWS Copilot .....	29
AWS Serverless Application Model .....	29
AWS Cloud Development Kit (AWS CDK) .....	30
Amazon EC2 Image Builder .....	31
Estratégias de implantação .....	33
Pré-cozimento versus inicialização AMIs .....	33
Implantações azuis/verdes .....	33
Implantações contínuas .....	34
Implantações do Canary .....	34
Implantações no local .....	35
Combinando serviços de implantação .....	35

---

Conclusão .....	37
Colaboradores .....	38
Outras fontes de leitura .....	39
Revisões do documento .....	40
Avisos .....	41
.....	xlii

# Visão geral das opções de implantação na AWS

Data de publicação: 31 de maio de 2024 () [Revisões do documento](#)

## Resumo

A Amazon Web Services (AWS) oferece várias opções para provisionar a infraestrutura e implantar seus aplicativos. Não importa se a arquitetura do seu aplicativo é um simples aplicativo web de três camadas ou um conjunto complexo de cargas de trabalho, a AWS oferece serviços de implantação para atender aos requisitos do seu aplicativo e da sua organização.

Este whitepaper é destinado a pessoas que buscam uma visão geral dos diferentes serviços de implantação oferecidos pela AWS. Ele apresenta os recursos comuns disponíveis nesses serviços de implantação e articula estratégias básicas para implantar e atualizar pilhas de aplicativos.

# Introdução

Projetar uma solução de implantação para seu aplicativo é uma parte essencial da criação de um aplicativo bem arquitetado na AWS. Com base na natureza do seu aplicativo e nos serviços subjacentes que ele exige, você pode usar os serviços da AWS para criar uma solução de implantação flexível que pode ser personalizada para atender às necessidades do seu aplicativo e da sua organização.

O catálogo cada vez maior de serviços da AWS não só complica o processo de decidir quais serviços irão compor sua arquitetura de aplicativos, mas também o processo de decidir como você criará, gerenciará e atualizará seu aplicativo. Ao projetar uma solução de implantação na AWS, você deve considerar como sua solução abordará os seguintes recursos:

- Provisão - Crie a infraestrutura bruta ou a infraestrutura de serviços gerenciados necessária para seu aplicativo.
- Configurar - Personalize sua infraestrutura com base no ambiente, no tempo de execução, na segurança, na disponibilidade, no desempenho, na rede ou em outros requisitos de aplicativos.
- Implantar - Instale ou atualize os componentes do seu aplicativo nos recursos de infraestrutura e gerencie a transição de uma versão anterior do aplicativo para uma nova versão do aplicativo.
- Dimensionar — ajuste de forma proativa ou reativa a quantidade de recursos disponíveis para seu aplicativo com base em um conjunto de critérios definidos pelo usuário.
- Monitore - Forneça visibilidade dos recursos que são lançados como parte da arquitetura do seu aplicativo. Monitore o uso de recursos, o sucesso ou a falha da implantação, a integridade do aplicativo, os registros do aplicativo, o desvio de configuração e muito mais.

Este whitepaper destaca os serviços de implantação oferecidos pela AWS e descreve estratégias para projetar uma arquitetura de implantação bem-sucedida para qualquer tipo de aplicativo.

# Serviços de implantação da AWS

A tarefa de projetar uma solução de implantação escalável, eficiente e econômica não deve se limitar à forma como você atualizará a versão do aplicativo, mas também deve considerar como você gerenciará a infraestrutura de suporte durante todo o ciclo de vida do aplicativo. Provisionamento de recursos, gerenciamento de configuração, implantação de aplicativos, atualizações de software, monitoramento, controle de acesso e outras preocupações são fatores importantes a serem considerados ao projetar uma solução de implantação.

Os serviços da AWS podem fornecer recursos de gerenciamento para um ou mais aspectos do ciclo de vida do seu aplicativo. Dependendo do equilíbrio de controle desejado (gerenciamento manual de recursos) versus conveniência (gerenciamento de recursos da AWS) e do tipo de aplicativo, esses serviços podem ser usados sozinhos ou combinados para criar uma solução de implantação rica em recursos. Esta seção fornecerá uma visão geral dos serviços da AWS que podem ser usados para permitir que as organizações criem e entreguem aplicativos com mais rapidez e confiabilidade.

## AWS CloudFormation

[AWS CloudFormation](#) é um serviço que permite aos clientes provisionar e gerenciar praticamente qualquer recurso da AWS usando uma linguagem de modelo personalizada expressa em YAML ou JSON. Um AWS CloudFormation modelo cria recursos de infraestrutura em um grupo chamado pilha e permite que você defina e personalize todos os componentes necessários para operar seu aplicativo, mantendo o controle total desses recursos. O uso de modelos introduz a capacidade de implementar o controle de versão em sua infraestrutura e a capacidade de replicar sua infraestrutura de forma rápida e confiável.

AWS CloudFormation oferece controle granular sobre o provisionamento e o gerenciamento de todos os componentes da infraestrutura de aplicativos, desde componentes de baixo nível, como tabelas de rotas ou configurações de sub-rede, até componentes de alto nível, como distribuições. CloudFront AWS CloudFormation é comumente usado com outros serviços de implantação da AWS ou ferramentas de terceiros, combinado AWS CloudFormation com serviços de implantação mais especializados para gerenciar implantações de código de aplicativo em componentes de infraestrutura.

A AWS oferece extensões para o CloudFormation serviço, além de seus recursos básicos:

- [AWS Cloud Development Kit \(AWS CDK\)](#) é um kit de desenvolvimento de software (SDK) de código aberto para modelar programaticamente a infraestrutura da AWS com JavaScript Python TypeScript, Java ou C#/.NET.
- [AWS Serverless Application Model \(AWS SAM\)](#) é uma estrutura de código aberto para simplificar a criação de aplicativos sem servidor na AWS. Ele fornece sintaxe abreviada para expressar funções APIs, bancos de dados e mapeamentos de fontes de eventos.

Tabela 1: recursos AWS CloudFormation de implantação

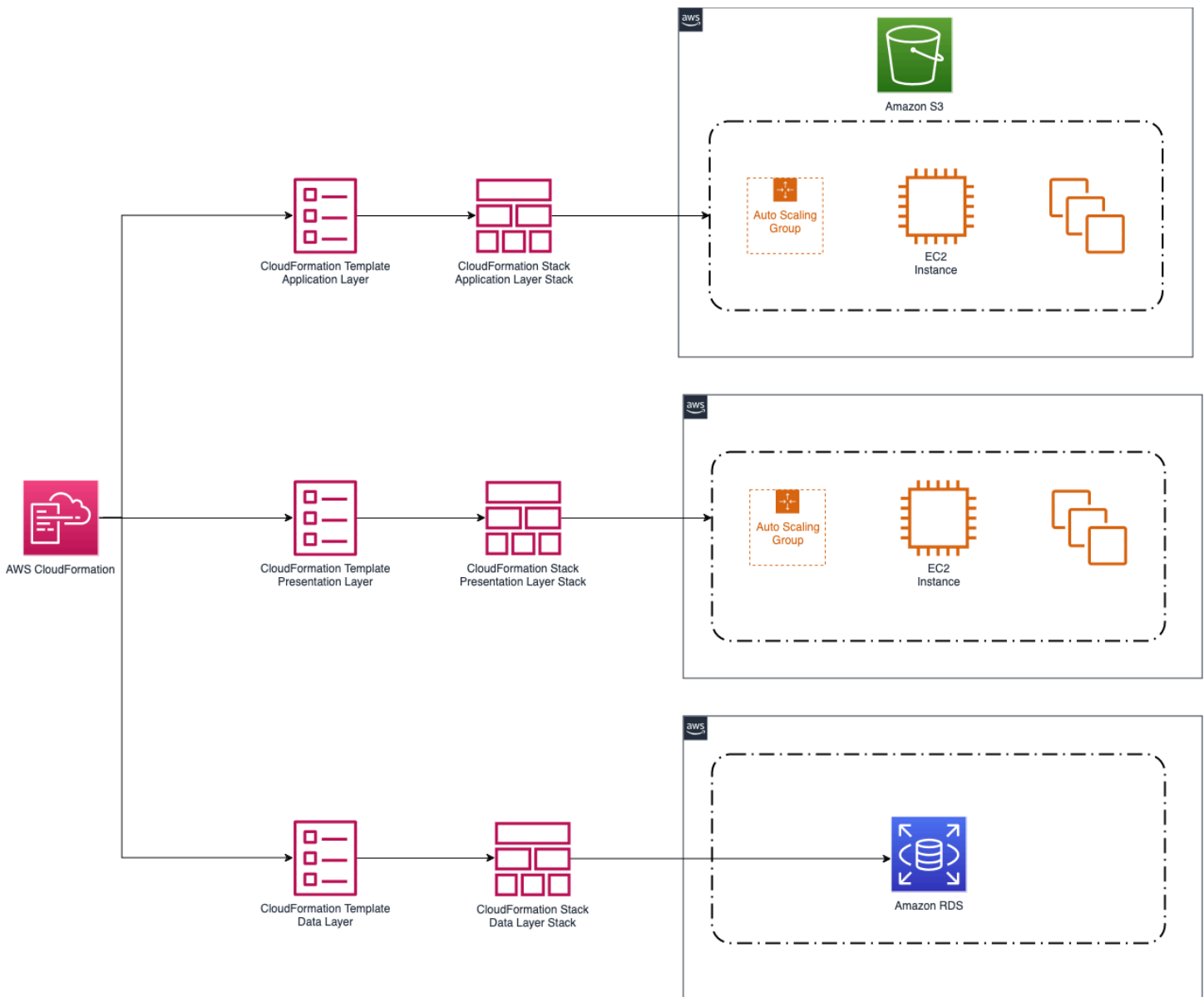
Recurso	Descrição
Provisão	<p>CloudFormation criará e atualizará automaticamente os componentes da infraestrutura definidos em um modelo.</p> <p>Consulte as <a href="#">AWS CloudFormation Melhores Práticas</a> para obter mais detalhes sobre a criação de infraestrutura usando AWS CloudFormation modelos.</p>
Configurar	<p>AWS CloudFormation os modelos oferecem ampla flexibilidade para personalizar e atualizar todos os componentes da infraestrutura.</p> <p>Consulte <a href="#">Anatomia AWS CloudFormation do modelo</a> para obter mais detalhes sobre a personalização de modelos.</p>
Implantar	<p>Atualize seus AWS CloudFormation modelos para alterar os recursos em uma pilha. Dependendo da arquitetura do aplicativo, talvez você precise de um serviço de implantação adicional para atualizar a versão do aplicativo em execução na sua infraestrutura.</p> <p>Consulte <a href="#">Implantação de aplicativos na Amazon EC2 com AWS CloudFormation</a> para obter mais detalhes sobre como AWS</p>



Recurso	Descrição
	CloudFormation pode ser usado como uma solução de implantação.
Escala	AWS CloudFormation não lidará automaticamente com o escalonamento da infraestrutura em seu nome; no entanto, você pode configurar políticas de escalonamento automático para seus recursos em um AWS CloudFormation modelo.
Monitorar	<p>AWS CloudFormation fornece monitoramento nativo do sucesso ou falha das atualizações da infraestrutura definida em um modelo, bem como detecção de desvios para monitorar quando os recursos definidos em um modelo não atendem às especificações. Soluções adicionais de monitoramento precisarão estar em vigor para monitoramento e métricas em nível de aplicativo.</p> <p>Consulte <a href="#">Monitorando o progresso de uma atualização de pilha</a> para obter mais detalhes sobre como AWS CloudFormation monitora as atualizações de infraestrutura.</p>

O diagrama a seguir mostra um caso de uso comum para AWS CloudFormation. Aqui, os AWS CloudFormation modelos são criados para definir todos os componentes de infraestrutura necessários para criar um aplicativo web simples de três camadas. Neste exemplo, estamos usando scripts de bootstrap definidos em AWS CloudFormation para implantar a versão mais recente do nosso aplicativo nas EC2 instâncias da Amazon; no entanto, também é uma prática comum combinar serviços de implantação adicionais com AWS CloudFormation (usando AWS CloudFormation somente para seus recursos de gerenciamento e provisionamento de infraestrutura). Observe que mais de um AWS CloudFormation modelo é usado para criar a infraestrutura. No diagrama, AWS CloudFormation é usado para criar todos os componentes da infraestrutura, incluindo funções do IAM VPCs, sub-redes, tabelas de rotas, grupos de segurança e políticas de bucket do Amazon S3.

AWS CloudFormation Modelos separados são usados para criar cada domínio da arquitetura do aplicativo.



AWS CloudFormation caso de uso

## AWS Elastic Beanstalk

[AWS Elastic Beanstalk](#) é um easy-to-use serviço para implantar e escalar aplicativos e serviços web desenvolvidos com Java, .NET, .NET Core, PHP, Node.js, Python, Ruby, Go ou Docker em servidores conhecidos, como Apache, Nginx, Passenger e IIS. O Elastic Beanstalk é uma solução completa de gerenciamento de aplicativos e gerencia todas as tarefas de infraestrutura e plataforma em seu nome.

Com o Elastic Beanstalk, você pode rapidamente implantar, gerenciar e escalar aplicativos sem a carga operacional de gerenciar a infraestrutura. O Elastic Beanstalk reduz a complexidade do gerenciamento de aplicativos web, tornando-o uma boa opção para organizações que são novatas na AWS ou desejam implantar uma aplicação web o mais rápido possível.

Ao usar o Elastic Beanstalk como sua solução de implantação, basta fazer o upload do código-fonte e o Elastic Beanstalk provisionará e operará toda a infraestrutura necessária, incluindo servidores, bancos de dados, balanceadores de carga, redes e grupos de auto scaling. Embora esses recursos sejam criados em seu nome, você mantém o controle total desses recursos, permitindo que os desenvolvedores os personalizem conforme necessário. O Elastic Beanstalk atende aos critérios de conformidade com ISO, PCI, SOC 1, SOC 2 e SOC 3, juntamente com os critérios de elegibilidade à HIPAA. Isso significa que os aplicativos executados no Elastic Beanstalk podem processar dados financeiros regulamentados ou informações de saúde protegidas (PHI).

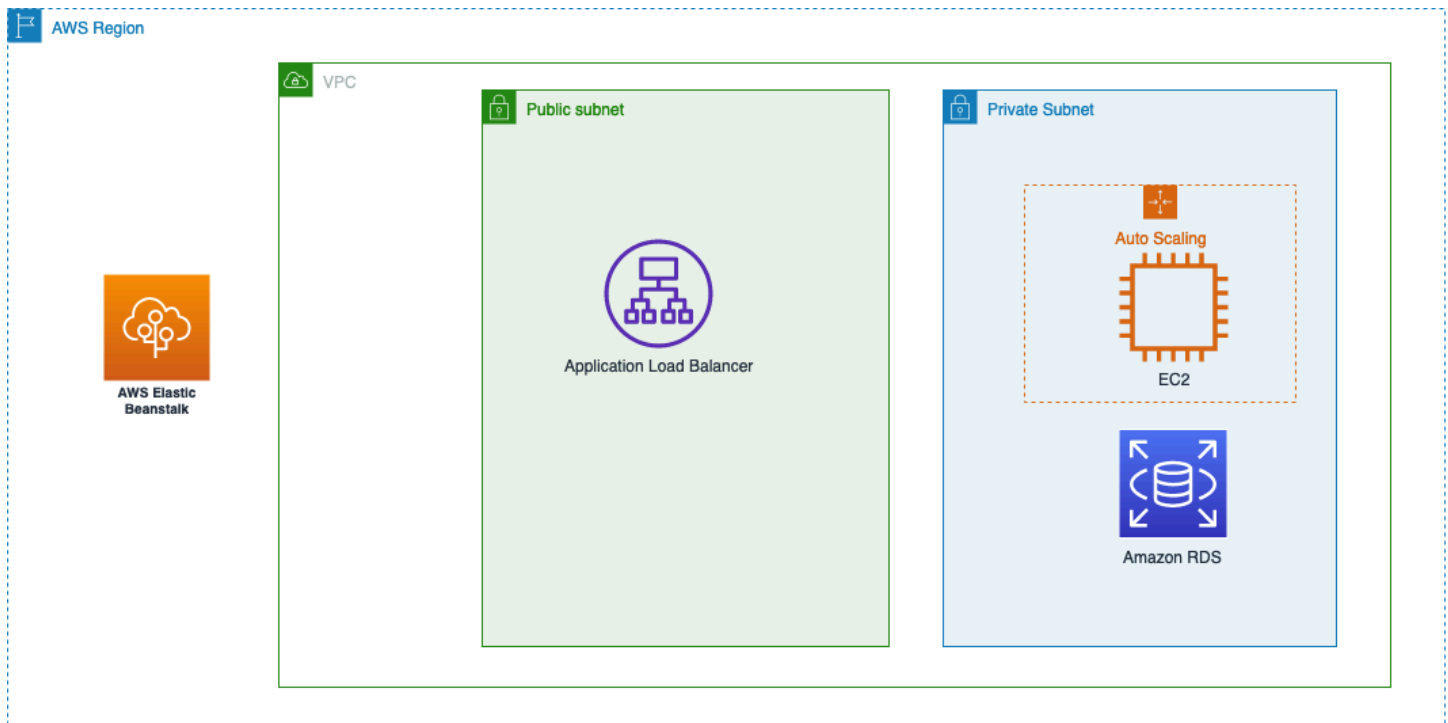
Tabela 2: Recursos AWS Elastic Beanstalk de implantação

Recurso	Descrição
Provisão	<p>O Elastic Beanstalk criará todos os componentes de infraestrutura necessários para operar um aplicativo ou serviço web executado em uma de suas plataformas suportadas. Se você precisar de infraestrutura adicional, ela deverá ser criada fora do Elastic Beanstalk.</p> <p>Consulte Plataformas do <a href="#">Elastic Beanstalk</a> para obter mais detalhes sobre as plataformas de aplicativos web suportadas pelo Elastic Beanstalk.</p>
Configurar	<p>O Elastic Beanstalk fornece uma ampla variedade de opções para personalizar os recursos em seu ambiente.</p> <p>Consulte <a href="#">Configuração de ambientes do Elastic Beanstalk</a> para obter mais informações sobre a personalização dos recursos criados pelo Elastic Beanstalk.</p>

Recurso	Descrição
Implantar	<p>O Elastic Beanstalk gerencia automaticamente as implantações de aplicativos e cria um ambiente que executa uma nova versão do seu aplicativo sem afetar os usuários existentes.</p> <p>Consulte <a href="#">Implantando aplicativos em AWS Elastic Beanstalk para</a> obter mais detalhes sobre implantações de aplicativos com o Elastic Beanstalk.</p>
Escala	<p>O Elastic Beanstalk usa o Elastic Load Balancing e o Auto Scaling para escalar automaticamente seu aplicativo para dentro e para fora com base em suas necessidades específicas. Várias zonas de disponibilidade oferecem a opção de melhorar a confiabilidade e a disponibilidade do aplicativo.</p> <p>Consulte o <a href="#">Grupo de Auto Scaling para seu ambiente do Elastic Beanstalk para obter mais detalhes sobre o escalonamento automático com o Elastic Beanstalk.</a></p>
Monitorar	<p>O Elastic Beanstalk oferece monitoramento de ambiente integrado para aplicativos, incluindo sucesso/falhas na implantação, integridade do ambiente, desempenho de recursos e registros de aplicativos.</p> <p>Consulte <a href="#">Monitoramento de um ambiente</a> para obter mais detalhes sobre o monitoramento completo com o Elastic Beanstalk.</p>

Recurso	Descrição
Suporte Graviton	Os processadores AWS Graviton baseados em arm64 oferecem a melhor relação preço/desempenho para suas cargas de trabalho na nuvem executadas na Amazon. EC2 Com o AWS Graviton no Elastic Beanstalk, você pode EC2 selecionar tipos de instância da Amazon para atender às necessidades de otimização de suas cargas de trabalho e se beneficiar de uma melhor relação preço/desempenho em relação a um processador comparável baseado em x86.

O Elastic Beanstalk facilita a rápida implantação e o gerenciamento de aplicativos web na AWS. O exemplo a seguir mostra um caso de uso geral do Elastic Beanstalk, pois ele é usado para implantar uma aplicação web simples. Toda a infraestrutura de aplicativos (incluindo grupos de segurança, funções do IAM e CloudWatch alarmes) é criada e gerenciada pelo Elastic Beanstalk. As EC2 instâncias da Amazon são provisionadas automaticamente com um ambiente de execução e pacotes de implantação. Os ambientes do Elastic Beanstalk podem se integrar a recursos como o Amazon Relational Database Service (Amazon RDS), que são criados fora do Elastic Beanstalk.



## AWS Elastic Beanstalk caso de uso

## AWS CodeDeploy

[AWS CodeDeploy](#) é um serviço de implantação totalmente gerenciado que automatiza implantações de aplicativos em serviços computacionais como Amazon, EC2 [Amazon Elastic Container Service](#) (Amazon ECS) ou servidores locais [AWS Lambda](#). As organizações podem usar CodeDeploy para automatizar as implantações de um aplicativo e remover operações manuais propensas a erros do processo de implantação. CodeDeploy pode ser usado com uma grande variedade de conteúdo de aplicativos, incluindo código, funções sem servidor, arquivos de configuração e muito mais.

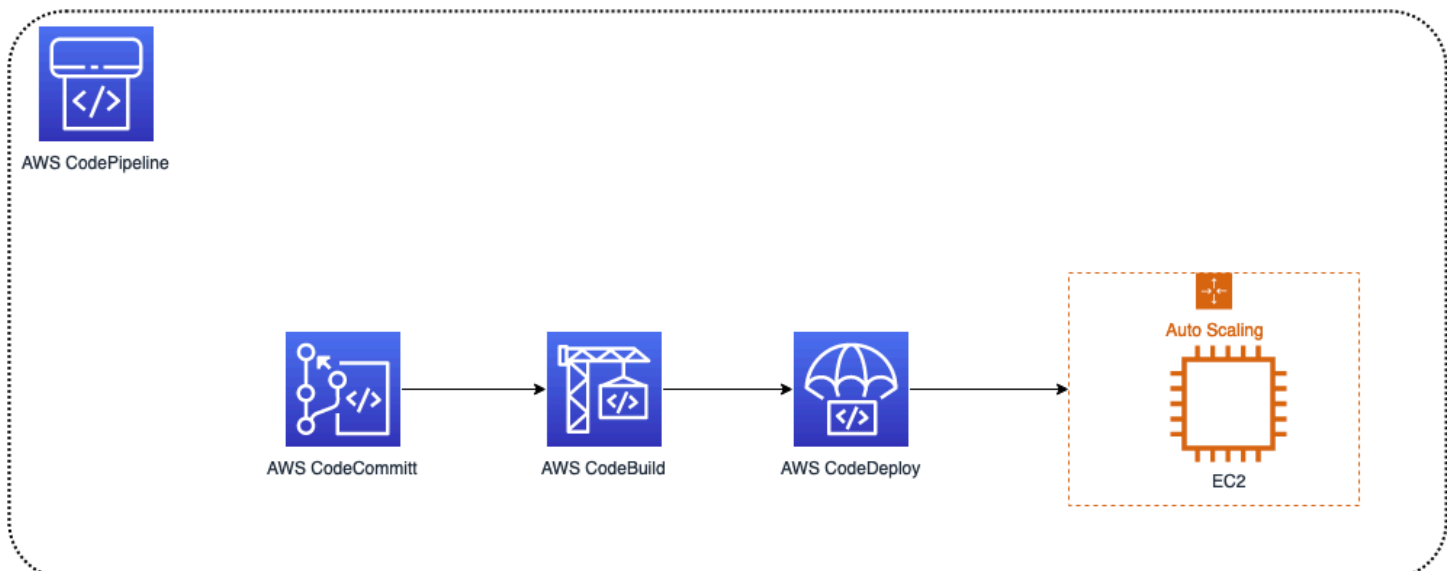
CodeDeploy destina-se a ser usado como um serviço básico focado em ajudar os desenvolvedores de aplicativos a implantar e atualizar o software que está sendo executado na infraestrutura existente. Ela não é uma solução de gerenciamento de end-to-end aplicativos e deve ser usada em conjunto com outros serviços de implantação da AWS [AWS CodeStar](#) [AWS CodePipeline](#), como outras [ferramentas para desenvolvedores da AWS](#) e serviços de terceiros (consulte [Integrações de AWS CodeDeploy produtos para obter uma lista completa de integrações](#) de produtos) como parte de um pipeline completo de CI/CD. Além disso, CodeDeploy não gerencia a criação de recursos em nome do usuário.

Tabela 3: recursos AWS CodeDeploy de implantação

Recurso	Descrição
Provisão	<p>CodeDeploy é destinado ao uso com recursos computacionais existentes e não cria recursos em seu nome. CodeDeploy exige que os recursos computacionais sejam organizados em uma construção chamada grupo de implantação para implantar o conteúdo do aplicativo.</p> <p>Consulte <a href="#">Trabalhando com grupos de implantação em CodeDeploy</a> para obter mais detalhes sobre CodeDeploy a vinculação a recursos computacionais.</p>
Configurar	<p>CodeDeploy usa um arquivo de especificação do aplicativo para definir personalizações para recursos computacionais.</p> <p>Consulte <a href="#">Referência de CodeDeploy AppSpec arquivo</a> para obter mais detalhes sobre as personalizações de recursos com CodeDeploy.</p>
Implantar	<p>Dependendo do tipo de recurso computacional CodeDeploy usado com, CodeDeploy oferece estratégias diferentes para implantar seu aplicativo.</p> <p>Consulte <a href="#">Trabalhando com implantações em CodeDeploy</a> para obter mais detalhes sobre os tipos de processos de implantação suportados.</p>
Escala	<p>CodeDeploy não oferece suporte ao dimensionamento de sua infraestrutura de aplicativos subjacente; no entanto, dependendo de suas <a href="#">configurações de implantação, ele pode criar recursos adicionais para suportar implantações azul/verdes</a>.</p>

Recurso	Descrição
Monitorar	<p>CodeDeploy pode monitorar o sucesso ou o fracasso das implantações e oferece um histórico de todas as implantações, mas não fornece métricas de desempenho ou de nível de aplicativo.</p> <p>Consulte <a href="#">Monitoramento de implantações em CodeDeploy</a> para obter mais detalhes sobre os tipos de recursos de monitoramento oferecidos pelo CodeDeploy</p>

O diagrama a seguir ilustra um caso de uso geral CodeDeploy como parte de uma solução completa de CI/CD. Neste exemplo, CodeDeploy é usado em conjunto com ferramentas adicionais para desenvolvedores da AWS, a saber AWS CodePipeline (automatizar pipelines de CI/CD), [AWS CodeBuild](#) (criar e testar componentes do aplicativo) e (repositório de código-fonte) para implantar um aplicativo em um grupo de instâncias da Amazon. [AWS CodeCommit](#) EC2 CodeDeploy é usado com outras ferramentas como parte de um pipeline completo de CI/CD. CodeDeploy gerencia a implantação de componentes do aplicativo em recursos computacionais que fazem parte de um grupo de implantação. Todos os componentes da infraestrutura são criados fora do CodeDeploy.



## AWS CodeDeploy caso de uso



## AWS CodeDeploy para AWS Lambda

AWS CodeDeploy for AWS Lambda permite que você automatize suas implantações sem servidor, oferecendo maior controle e visibilidade sobre as versões de seus aplicativos. Você pode usar CodeDeploy para implantar uma nova versão da sua função sem servidor para uma pequena porcentagem de usuários ou tráfego e aumentar gradualmente o tráfego à medida que ganha confiança na nova versão. Com CodeDeploy, você pode definir grupos de implantação, que representam um conjunto de funções Lambda que recebem tráfego da mesma fonte de eventos. Por exemplo, você pode criar um grupo de implantação para um conjunto de funções do Lambda que são iniciadas pelo API Gateway ou por uma regra da Amazon EventBridge. Em seguida, você pode criar uma implantação usando CodeDeploy, que implanta a nova versão da sua função sem servidor em um grupo de implantação especificado.

CodeDeploy também permite que você defina uma configuração de implantação, que especifica as configurações de uma implantação, como o tipo de implantação, a estratégia de implantação e as regras de mudança de tráfego. Você pode usar a estratégia de implantação do Canary para implantar a nova versão da sua função sem servidor em uma pequena porcentagem do tráfego e monitorar a integridade e o desempenho da nova versão antes de aumentar o tráfego para ela.

Ao usar o CodeDeploy modo sem servidor, você pode automatizar seu processo de implantação, reduzir o tempo e o esforço necessários para lançar novas versões do seu aplicativo e aumentar a estabilidade e a confiabilidade de suas funções sem servidor.

## Amazon Elastic Container Service

O Amazon Elastic Container Service (Amazon ECS) é um serviço de orquestração de contêineres totalmente gerenciado que oferece suporte a contêineres Docker e permite que você execute facilmente aplicativos em um cluster gerenciado. O Amazon ECS elimina a necessidade de instalar, operar e escalar a infraestrutura de gerenciamento de contêineres e simplifica a criação de ambientes com os principais recursos familiares da AWS, como [Security Groups](#), [Elastic Load Balancing](#) e [AWS Identity and Access Management](#) (IAM).

Ao executar aplicativos no Amazon ECS, você pode optar por fornecer a potência computacional subjacente para seus contêineres com EC2 instâncias da Amazon ou com [AWS Fargate](#) mecanismo de computação sem servidor para contêineres. Em ambos os casos, o Amazon ECS coloca e escala automaticamente seus contêineres em seu cluster de acordo com as configurações definidas pelo usuário. Embora o Amazon ECS não crie componentes de infraestrutura como Load

Balancers ou funções do IAM em seu nome, o serviço Amazon ECS fornece vários APIs para simplificar a criação e o uso desses recursos em um cluster do Amazon ECS.

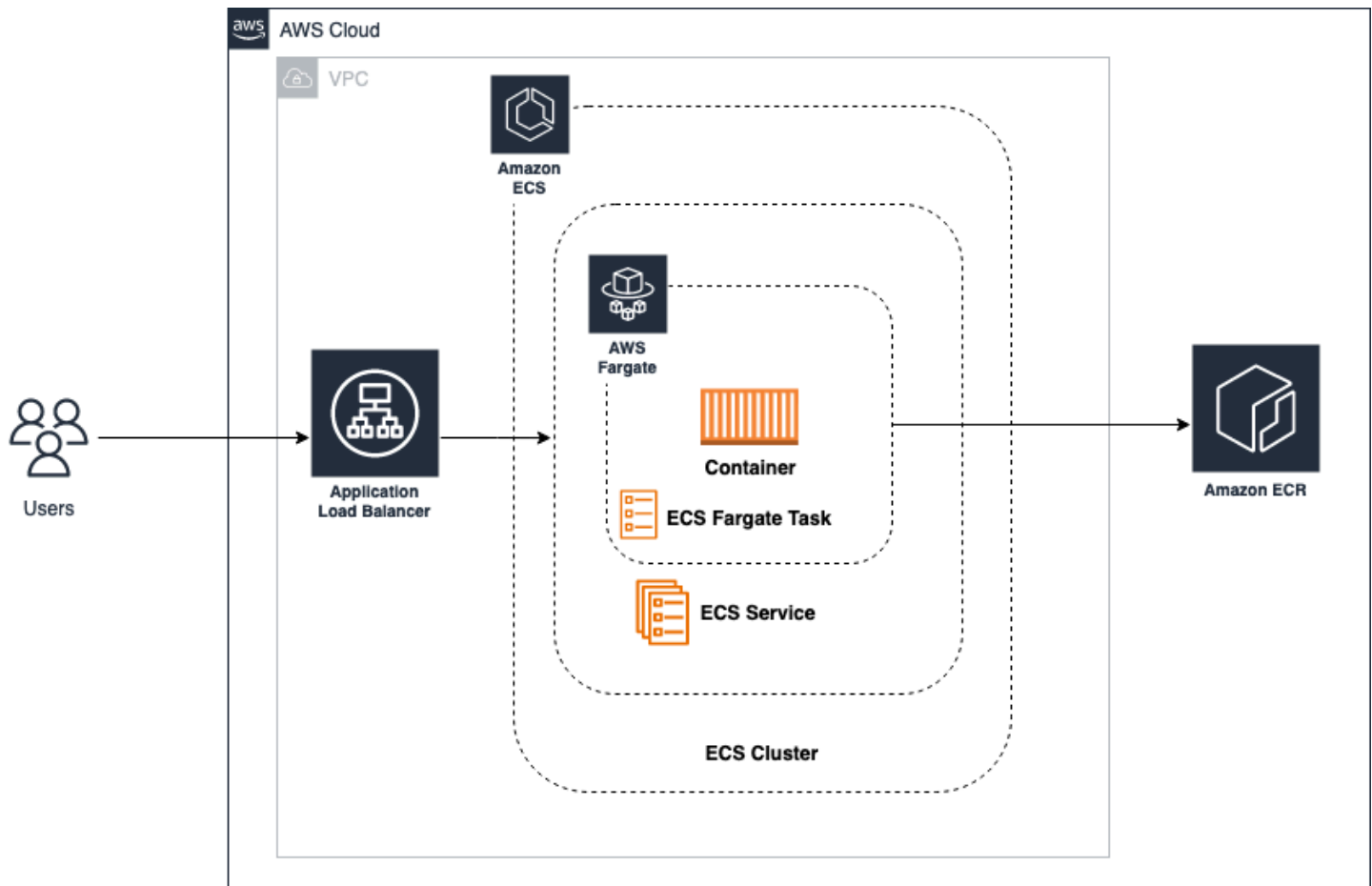
O Amazon ECS permite que os desenvolvedores tenham controle direto e refinado sobre todos os componentes da infraestrutura, permitindo a criação de arquiteturas de aplicativos personalizadas. Além disso, o Amazon ECS oferece suporte a diferentes estratégias de implantação para atualizar suas imagens de contêiner de aplicativos.

Tabela 4: Recursos de implantação do Amazon ECS

Recurso	Descrição
Provisão	<p>O Amazon ECS provisionará novas instâncias de contêineres de aplicativos e recursos computacionais com base em políticas de escalabilidade e configurações do Amazon ECS. Recursos de infraestrutura, como Load Balancers, precisarão ser criados fora do Amazon ECS.</p> <p>Consulte <a href="#">Getting Started with Amazon ECS</a> para obter mais detalhes sobre os tipos de recursos que podem ser criados com o Amazon ECS.</p>
Configurar	<p>O Amazon ECS oferece suporte à personalização dos recursos computacionais criados para executar um aplicativo em contêineres, bem como das condições de tempo de execução dos contêineres do aplicativo (por exemplo, variáveis de ambiente, portas expostas, memória/CPU reservada). A personalização dos recursos computacionais subjacentes só está disponível se você usar instâncias da Amazon EC2.</p> <p>Consulte Como <a href="#">criar um cluster</a> para obter mais detalhes sobre como personalizar um</p>

Recurso	Descrição
	cluster do Amazon ECS para executar aplicativos em contêineres.
Implantar	<p>O Amazon ECS oferece suporte a várias estratégias de implantação para seus aplicativos em contêineres.</p> <p>Consulte <a href="#">Tipos de implantação do Amazon ECS</a> para obter mais detalhes sobre os tipos de processos de implantação que são compatíveis.</p>
Escala	<p>O Amazon ECS pode ser usado com políticas de escalabilidade automática para ajustar automaticamente o número de contêineres em execução no seu cluster do Amazon ECS.</p> <p>Consulte <a href="#">Service Auto Scaling</a> para obter mais detalhes sobre como configurar o auto scaling para seus aplicativos em contêineres no Amazon ECS.</p>
Monitorar	<p>O Amazon ECS oferece suporte ao monitoramento de recursos computacionais e contêineres de aplicativos com CloudWatch</p> <p>Consulte <a href="#">Monitoramento do Amazon ECS</a> para obter mais detalhes sobre os tipos de recursos de monitoramento oferecidos pelo Amazon ECS.</p>

O diagrama a seguir ilustra o Amazon ECS sendo usado para gerenciar um aplicativo simples em contêineres. Neste exemplo, componentes de infraestrutura são criados fora do Amazon ECS, e o Amazon ECS é usado para gerenciar a implantação e a operação de contêineres de aplicativos no cluster



## Caso de uso do Amazon ECS

### Note

- A infraestrutura de aplicativos (incluindo repositórios do Amazon Elastic Container Registry (Amazon ECR), configurações do Amazon ECS e Load Balancers) é provisionada e gerenciada fora da sua implantação do Amazon ECS.
- O Amazon ECS gerencia a implantação de contêineres de aplicativos executados dentro do serviço Amazon ECS como tarefas provenientes de um registro de contêineres como o Amazon ECR.

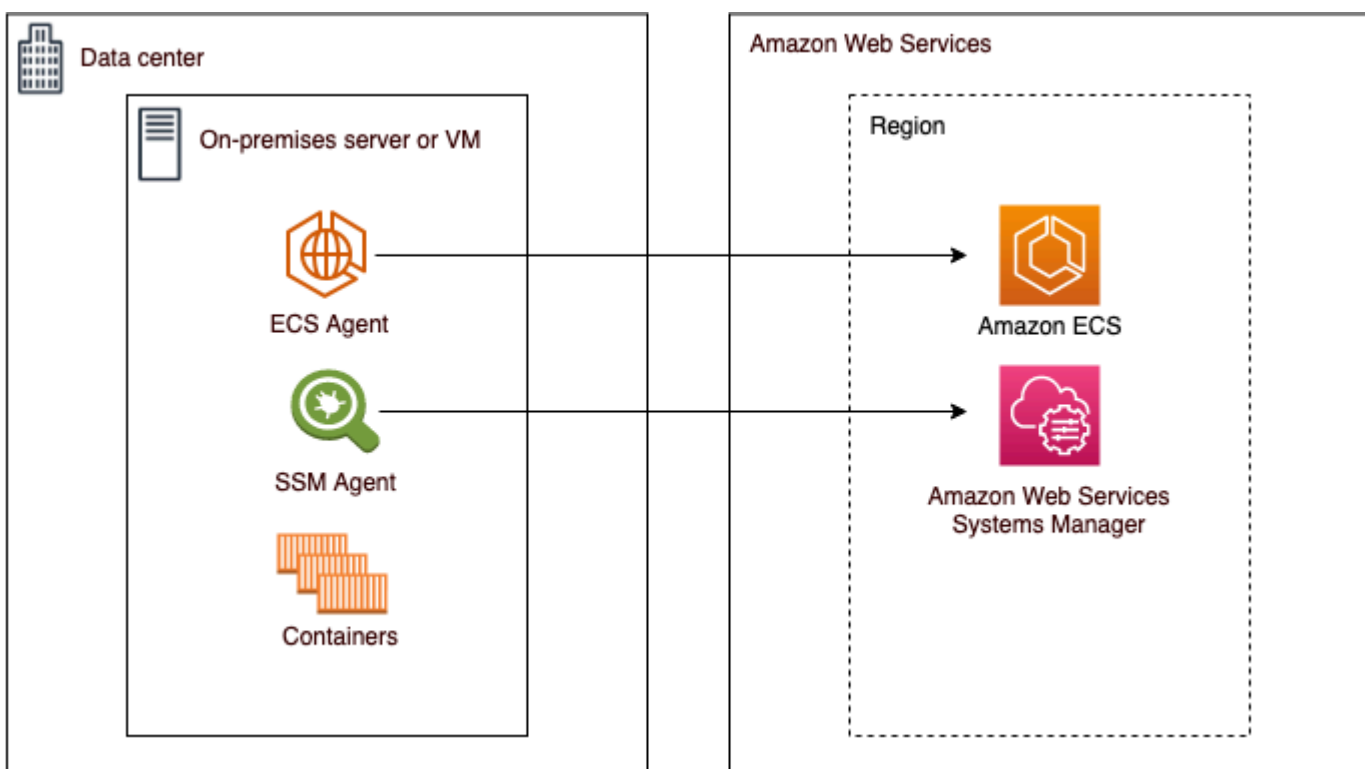
O Amazon ECS suporta vários tipos de instância de contêiner, como Linux e Windows, bem como tipos de instância externa, como uma máquina virtual (VM) local com o Amazon ECS Anywhere.

## Amazon ECS Anywhere

O [Amazon ECS](#) Anywhere permite que você execute tarefas do Amazon ECS em qualquer lugar, seja no local ou em outros ambientes de nuvem. Com o Amazon ECS Anywhere, você pode facilmente implantar e gerenciar aplicativos em contêineres em toda a sua infraestrutura híbrida, mantendo uma experiência operacional consistente. O serviço funciona estendendo a plataforma Amazon ECS para qualquer ambiente, incluindo datacenters locais, escritórios remotos e outros ambientes em nuvem. Ele permite que você use as mesmas ferramentas APIs e o mesmo Amazon ECS familiar para implantar e gerenciar contêineres em todos os seus ambientes, sem precisar se preocupar com a infraestrutura subjacente.

O Amazon ECS Anywhere usa o agente do Amazon ECS para gerenciar a implantação e o ciclo de vida dos contêineres, permitindo que você use as mesmas definições de tarefas e arquivos de configuração do Amazon ECS que você usa no. Nuvem AWS Isso pode ajudar a simplificar o processo de implantação e gerenciamento de contêineres em sua infraestrutura híbrida e reduzir o tempo e o esforço necessários para configuração e gerenciamento manuais.

Com o Amazon ECS Anywhere, você também pode aproveitar outros serviços da AWS, como IAM e Amazon ECR AWS CloudFormation, para gerenciar seus aplicativos em contêineres. Isso pode ajudar a garantir que seus aplicativos sejam seguros, compatíveis e integrados a outros serviços da AWS.



## Amazon ECS Anywhere architecture

# Amazon Elastic Container Service em AWS Outposts

[O Amazon ECS on AWS Outposts](#) é um serviço da AWS totalmente gerenciado que permite que você execute tarefas do Amazon ECS localmente, usando as mesmas APIs e ferramentas que você usa no Nuvem AWS. Com o Amazon ECS ativado AWS Outposts, você pode implantar e gerenciar aplicativos em contêineres de forma consistente e familiar, independentemente de executá-los localmente ou na nuvem. AWS Outposts é um serviço totalmente gerenciado que estende a infraestrutura APIs, os serviços e as ferramentas da AWS para seus ambientes locais. Com o Amazon ECS ativado AWS Outposts, você pode executar tarefas do Amazon ECS em hardware dedicado à sua organização, sem precisar se preocupar com a infraestrutura subjacente. Isso pode ajudar a garantir que seus aplicativos sejam implantados de forma segura e compatível, além de permitir que você aproveite a flexibilidade e a escalabilidade da nuvem.

O Amazon ECS on AWS Outposts funciona implantando um conjunto de serviços da AWS e APIs em seu ambiente local, o que permite que você execute tarefas do Amazon ECS em hardware dedicado. Isso inclui o agente Amazon ECS, que gerencia a implantação e o ciclo de vida dos contêineres, e a AWS Outposts infraestrutura, que fornece um ambiente seguro e compatível para a execução de aplicativos em contêineres. Com o Amazon ECS ativado AWS Outposts, você pode usar o mesmo Amazon ECS APIs e as mesmas ferramentas que usa no Nuvem AWS, facilitando a implantação e o gerenciamento de aplicativos em contêineres de forma consistente e familiar. Isso pode ajudar a reduzir o tempo e o esforço necessários para configuração e gerenciamento manuais e melhorar a consistência e a confiabilidade em toda a sua infraestrutura híbrida. O Amazon ECS on AWS Outposts também se integra a outros serviços da AWS, como IAM e Amazon ECR AWS CloudFormation, para gerenciar seus aplicativos em contêineres. Isso pode ajudar a garantir que seus aplicativos sejam seguros, compatíveis e integrados a outros serviços da AWS.

## Amazon Elastic Kubernetes Service

[O Amazon Elastic Kubernetes Service \(Amazon EKS\)](#) é um serviço totalmente gerenciado e certificado em conformidade com o Kubernetes que simplifica o processo de criação, proteção, operação e manutenção de clusters Kubernetes na AWS. O Amazon EKS se integra aos principais serviços da AWS CloudWatch, como Auto Scaling Groups e IAM, para fornecer uma experiência perfeita de monitoramento, escalabilidade e balanceamento de carga de seus aplicativos em contêineres.

O Amazon EKS fornece um plano de controle escalável e altamente disponível para cargas de trabalho do Kubernetes. Ao executar aplicativos no Amazon EKS, como no Amazon ECS, você pode optar por fornecer a potência computacional subjacente para seus contêineres com EC2 instâncias da Amazon ou com AWS Fargate

O Amazon VPC Lattice é um serviço de rede de aplicativos totalmente gerenciado, construído diretamente na infraestrutura de rede da AWS, que você pode usar para conectar, proteger e monitorar seus serviços em várias contas e nuvens privadas virtuais (). VPCs Com o Amazon EKS, você pode aproveitar o VPC Lattice por meio do uso do AWS Gateway API Controller, uma implementação da API Kubernetes Gateway. Usando o VPC Lattice, você pode configurar a conectividade entre clusters com a semântica padrão do Kubernetes de maneira simples e consistente.

Você pode usar o Amazon EKS com qualquer uma das seguintes opções de implantação:

- [Amazon EKS Distro](#) – O Amazon EKS Distro é uma distribuição do mesmo software do Kubernetes de código aberto e das dependências implantadas pelo Amazon EKS na nuvem. Amazon EKS Distro segue o mesmo ciclo de lançamento de versões do Kubernetes que o Amazon EKS, e é fornecido como um projeto - de código aberto. Para saber mais, consulte [Amazon EKS Distro](#).
- [Amazon EKS ativado AWS Outposts](#) — AWS Outposts habilita serviços, infraestrutura e modelos operacionais nativos da AWS em suas instalações locais. Com o Amazon EKS ativado AWS Outposts, você pode optar por executar clusters estendidos ou locais. Com clusters estendidos, o plano de controle do Kubernetes é executado em um Região da AWS e os nós são executados em AWS Outposts Com clusters locais, todo o cluster do Kubernetes é executado localmente AWS Outposts, incluindo o plano de controle e os nós do Kubernetes.
- [Amazon EKS Anywhere](#) – O Amazon EKS Anywhere é uma opção de implantação para o Amazon EKS que permite criar e operar facilmente clusters do Kubernetes on-premises. Tanto o Amazon EKS quanto o Amazon EKS Anywhere são baseados no Amazon EKS Distro. Para saber mais sobre o Amazon EKS Anywhere, consulte [Executando cargas de trabalho de contêineres híbridos com o Amazon EKS Anywhere](#), [Visão geral do Amazon EKS Anywhere](#) e [Comparando o Amazon EKS Anywhere com o Amazon EKS](#).

Ao escolher quais opções de implantação usar para o cluster do Kubernetes, considere o seguinte:

Tabela 5: Recursos de implantação do Kubernetes

Atributo	Amazon EKS	Amazon EKS em AWS Outposts	Amazon EKS Anywhere	Amazon EKS Distro
Hardware	Fornecido pela AWS	Fornecido pela AWS	Fornecido por você	Fornecido por você
Local de implantação	Nuvem AWS	Seu data center	Seu data center	Seu data center
Localização do ambiente de gerenciamento do Kubernetes	Nuvem AWS	Nuvem AWS ou seu data center	Seu data center	Seu data center
Local do plano de dados do Kubernetes	Nuvem AWS	Seu data center	Seu data center	Seu data center
Suporte	AWS apoio	AWS apoio	AWS apoio	Suporte da comunidade do OSS

Tabela 6: Recursos de implantação do Amazon EKS

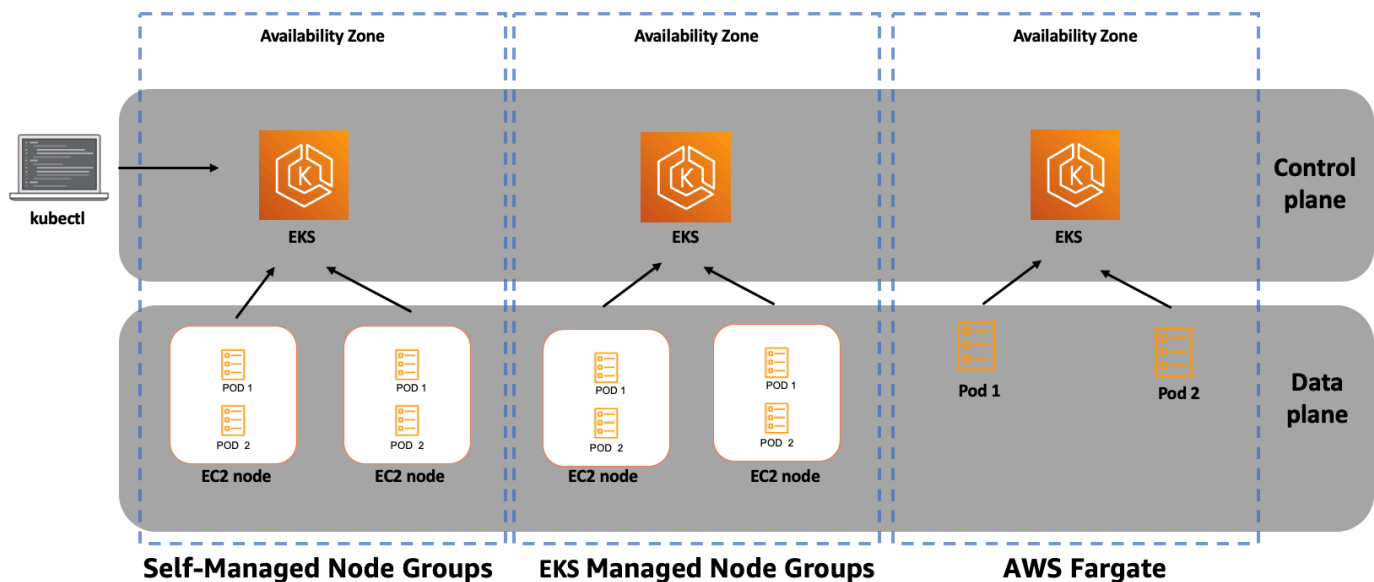
Recurso	Descrição
Provisão	<p>O Amazon EKS provisiona determinados recursos para oferecer suporte a aplicativos em contêineres:</p> <ul style="list-style-type: none"> <li>• Balanceadores de carga, se necessário</li> <li>• Recursos computacionais ou trabalhadores (o Amazon EKS é compatível com Windows e Linux)</li> <li>• Instâncias de contêineres de aplicativos ou pods</li> </ul>



Recurso	Descrição
	<p>Consulte <a href="#">Getting Started with Amazon EKS</a> para obter mais detalhes sobre o provisionamento de clusters do Amazon EKS.</p>
Configurar	<p>O Amazon EKS oferece suporte à personalização dos recursos computacionais (trabalhadores) se você usar EC2 instâncias da Amazon para fornecer potência computacional. O Amazon EKS também oferece suporte à personalização das condições de tempo de execução dos contêineres de aplicativos (pods).</p> <p>Consulte a documentação de <a href="#">configuração do Worker Nodes e do Fargate Pod</a> para obter mais detalhes.</p>
Implantar	<p>O Amazon EKS oferece suporte às mesmas estratégias de implantação do Kubernetes. Consulte Como <a href="#">escrever uma especificação de implantação do Kubernetes -&gt; Estratégia para obter</a> mais detalhes.</p>
Escala	<p>O Amazon EKS escala trabalhadores com o <a href="#">Kubernetes Cluster Autoscaler e pods com Kubernetes Horizontal Pod Autoscaler</a> e Kubernetes Vertical Pod Autoscaler. O Amazon EKS também oferece suporte ao <a href="#">Karpenter</a>, um autoescalador de cluster Kubernetes de código aberto, flexível e de alto desempenho para ajudar a melhorar a disponibilidade e a eficiência do seu aplicativo lançando rapidamente recursos computacionais do tamanho certo em resposta às mudanças na carga do aplicativo.</p>

Recurso	Descrição
Monitorar	<p>Os registros do plano de controle do Amazon EKS fornecem informações de auditoria e diagnóstico diretamente para o CloudWatch Logs. O plano de controle do Amazon EKS também se integra AWS CloudTrail ao registro de ações realizadas no Amazon EKS.</p> <p>Consulte <a href="#">Registro e monitoramento do Amazon EKS</a> para obter mais detalhes.</p>

O Amazon EKS permite que as organizações utilizem ferramentas e plug-ins de código aberto do Kubernetes e pode ser uma boa opção para organizações que migram para a AWS com ambientes Kubernetes existentes. O diagrama a seguir ilustra o Amazon EKS sendo usado para gerenciar um aplicativo geral em contêineres.



Amazon EKS use case

## Amazon EKS Anywhere

O [Amazon EKS Anywhere](#) permite criar e operar clusters Kubernetes em sua própria infraestrutura. O Amazon EKS Anywhere se baseia nos pontos fortes do Amazon EKS Distro e fornece software

de código aberto atualizado e corrigido para que você possa ter um ambiente Kubernetes local mais confiável do que uma oferta de Kubernetes autogerenciada.

O Amazon EKS Anywhere cria um cluster Kubernetes no local para um provedor escolhido. Os fornecedores compatíveis incluem Bare Metal (via Tinkerbell) CloudStack e vSphere. Para gerenciar esse cluster, você pode executar comandos de criação e exclusão de clusters em uma máquina administrativa Ubuntu ou Mac.

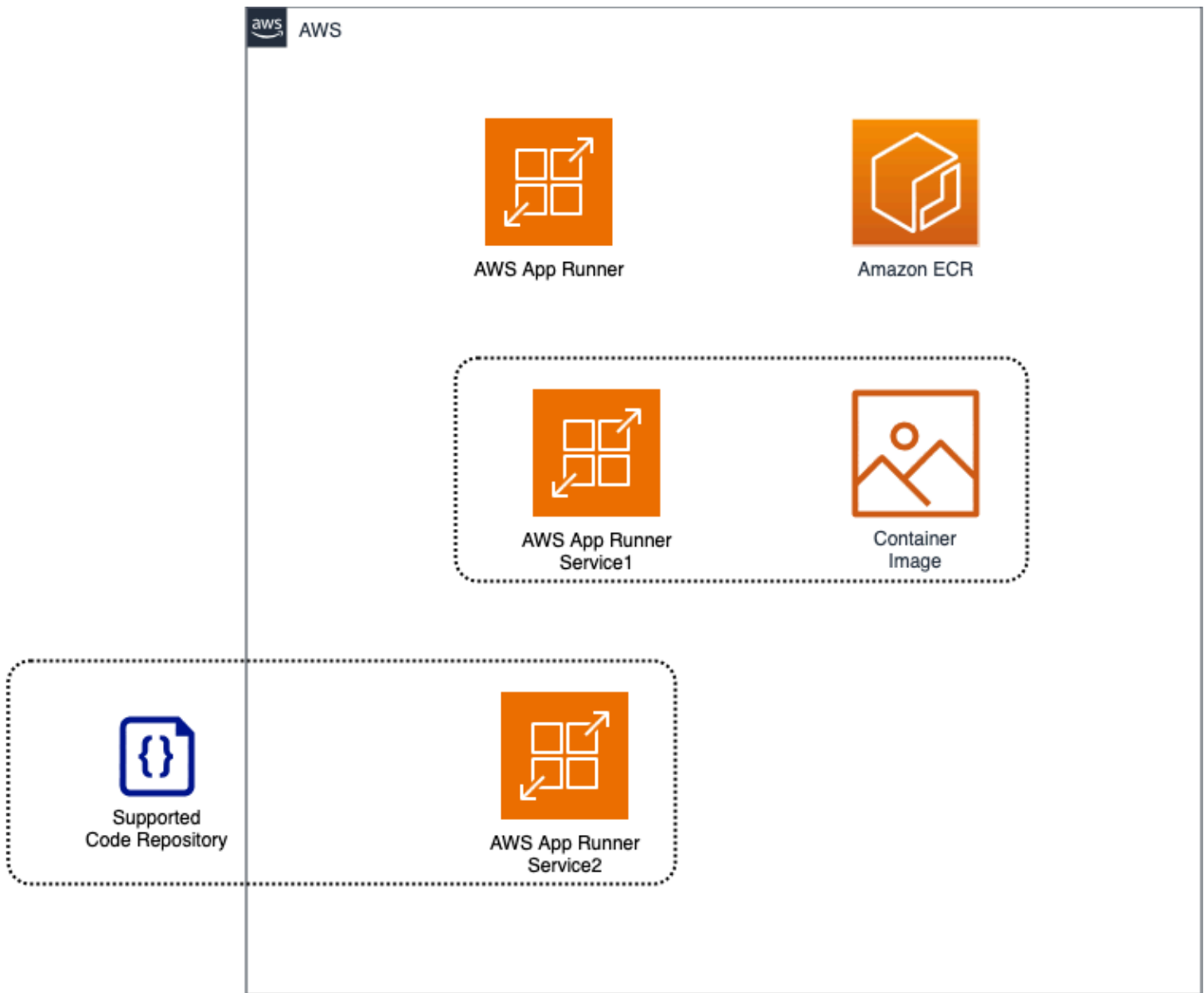
## AWS App Runner

[AWS App Runner](#) é um serviço de aplicativo de contêiner totalmente gerenciado que permite criar, implantar e executar aplicativos web em contêineres e serviços de API sem experiência prévia em infraestrutura ou contêiner. O App Runner se conecta diretamente ao seu repositório de código ou imagem. Ele fornece um pipeline automático de integração e entrega com operações totalmente gerenciadas, alto desempenho, escalabilidade e segurança.

O App Runner pega seu código-fonte ou imagem de origem de um repositório e, em seguida, cria e mantém um serviço web em execução para você no. Nuvem AWS Normalmente, você precisa chamar apenas uma ação do App Runner, `CreateService`, para criar seu serviço. Com um repositório de imagens de origem, você fornece uma imagem de ready-to-use contêiner que o App Runner pode implantar para executar seu serviço web. Com um repositório de código-fonte, você fornece seu código e instruções para criar e executar um serviço web e tem como alvo um ambiente de tempo de execução específico. O App Runner oferece suporte a várias plataformas de programação, cada uma com um ou mais tempos de execução gerenciados para as versões principais da plataforma. O App Runner é compatível com imagens de contêiner, bem como tempos de execução e estruturas da web, incluindo Node.js e Python. O App Runner monitora o número de solicitações simultâneas enviadas ao seu aplicativo e adiciona automaticamente instâncias adicionais com base no volume de solicitações. Se seu aplicativo não receber solicitações, o App Runner reduzirá os contêineres para uma instância provisionada, uma instância controlada pela CPU pronta para atender às solicitações recebidas em milissegundos.

No momento, o App Runner pode recuperar seu código-fonte de um GitHub repositório ou recuperar sua imagem de origem do Amazon ECR em seu. Conta da AWS

O diagrama a seguir mostra uma visão geral da arquitetura do serviço App Runner. No diagrama, há dois exemplos de serviços: um implanta o código-fonte do GitHub Amazon ECR e o outro implanta uma imagem fonte do Amazon ECR.



## App Runner use case

O App Runner oferece suporte ao desenvolvimento completo, incluindo aplicativos web de front-end e back-end que usam os protocolos HTTP e HTTPS. Esses aplicativos incluem serviços de API, serviços web de back-end e sites. O App Runner é compatível com imagens de contêiner, bem como tempos de execução e estruturas da web, incluindo Node.js e Python.

## Amazon Lightsail

[O Amazon Lightsail](#) é um serviço de nuvem simples e econômico que facilita a implantação e o gerenciamento de aplicativos na nuvem por pequenas empresas, startups e indivíduos. Ele fornece uma interface fácil de usar que abstrai grande parte do gerenciamento da infraestrutura subjacente

e facilita o lançamento e a execução de aplicativos na nuvem. Com o Lightsail, você pode implantar e gerenciar rapidamente servidores virtuais privados (VPS), bancos de dados e instâncias de armazenamento. O serviço fornece instâncias pré-configuradas que são otimizadas para várias cargas de trabalho WordPress, como Drupal e Joomla, entre outras. Isso pode ajudar a reduzir o tempo e o esforço necessários para instalar e configurar seu ambiente. O Lightsail também fornece um balanceador de carga integrado e escalabilidade automática, permitindo lidar com mudanças na demanda de tráfego sem intervenção manual. O serviço também fornece monitoramento e alertas, para que você possa acompanhar a integridade e o desempenho de seus aplicativos.

Um dos principais benefícios do Lightsail é sua simplicidade e facilidade de uso. O serviço foi projetado para ser acessível a usuários com experiência mínima em computação em nuvem, o que o torna uma boa opção para pequenas empresas ou indivíduos que desejam começar rapidamente na nuvem. Além disso, o Lightsail é econômico, com preços previsíveis que incluem computação, armazenamento e transferência de dados.

## Contêineres Amazon Lightsail

O Amazon Lightsail Containers é um serviço de contêineres totalmente gerenciado da AWS que facilita a implantação e o gerenciamento de aplicativos em contêineres na nuvem. Ele fornece uma maneira simples e econômica de iniciar e executar contêineres usando ferramentas populares de gerenciamento de contêineres, como Docker e Kubernetes.

O Lightsail Containers fornece um ambiente integrado para criar, testar e implantar aplicativos em contêineres. Ele simplifica o processo de implantação e gerenciamento de contêineres, fornecendo uma interface amigável que abstrai grande parte do gerenciamento da infraestrutura subjacente.

Com o Lightsail Containers, você pode implantar seus aplicativos em contêineres em uma VPC com apenas alguns cliques. O serviço fornece imagens de contêiner pré-configuradas para linguagens de programação populares, como Node.js, Python, Ruby e Java. Isso pode ajudar a reduzir o tempo e o esforço necessários para instalar e configurar seu ambiente de contêiner.

O Lightsail Containers também fornece um balanceador de carga integrado que pode distribuir automaticamente o tráfego em suas instâncias de contêiner, melhorando a disponibilidade e a escalabilidade dos aplicativos. Além disso, o serviço fornece escalabilidade automática de instâncias de contêiner, permitindo lidar com mudanças na demanda de tráfego sem intervenção manual.

Com o Lightsail Containers, você pode monitorar o desempenho de seus aplicativos em contêineres usando métricas e registros integrados. Você também pode se integrar a outros serviços da AWS,

como Amazon S3, Amazon RDS e AWS CodePipeline, para criar um pipeline de CI/CD totalmente automatizado e integrado para seus aplicativos em contêineres.

## Serviço Red Hat OpenShift na AWS

[Serviço Red Hat OpenShift na AWS](#)(ROSA) é um serviço gerenciado que está disponível por meio do AWS Management Console. Com o ROSA, como OpenShift usuário da Red Hat, você pode criar, escalar e gerenciar aplicativos em contêineres na AWS. Você pode usar o ROSA para criar clusters Kubernetes usando o Red Hat OpenShift APIs e as ferramentas e ter acesso a toda a amplitude e profundidade dos serviços da AWS. O ROSA simplifica a transferência de cargas de OpenShift trabalho locais da Red Hat para a AWS e oferece uma forte integração com outros serviços da AWS. Você também pode acessar o OpenShift licenciamento, o faturamento e o suporte da Red Hat diretamente por meio da AWS.

Cada cluster ROSA vem com um plano de controle totalmente gerenciado e nós de computação. A instalação, o gerenciamento, a manutenção e os upgrades são realizados pelo Red Hat SRE com suporte conjunto da Red Hat e da Amazon. Serviços de cluster (como registro, métricas e monitoramento) também estão disponíveis. Somente os trabalhadores do Red Hat Enterprise Linux CoreOS (RHCOS) são suportados pelo ROSA.

O ROSA se integrará a uma variedade de serviços de computação, armazenamento, banco de dados, análises, aprendizado de máquina, redes, dispositivos móveis e vários aplicativos da AWS, o que permitirá que os clientes se beneficiem do portfólio robusto de serviços da AWS que escalam sob demanda em todo o mundo. Esses serviços nativos da AWS estarão diretamente acessíveis para implantar e escalar serviços rapidamente por meio da mesma interface de gerenciamento.

## Zonas locais da AWS

Uma [zona local da AWS](#) é uma extensão de uma Região da AWS proximidade geográfica próxima aos seus usuários. As zonas locais têm suas próprias conexões com a Internet e suporte no AWS Direct Connect. Os recursos criados em uma zona local podem atender a usuários locais com comunicações de baixa latência. Uma zona local é representada por um código de região seguido por um identificador que indica a localização (por exemplo, us-west-2-lax-1a).

O Amazon ECS oferece suporte a cargas de trabalho que usam Zonas Locais quando a baixa latência ou o processamento local de dados são necessários. O plano de controle do Amazon ECS sempre funcionará no Região da AWS.

O Amazon EKS é compatível com determinados recursos em zonas locais. Isso inclui [EC2 nós autogerenciados da Amazon](#), volumes do Amazon EBS e balanceadores de carga de aplicativos. O ambiente de gerenciamento do Kubernetes gerenciado pelo Amazon EKS é sempre executado na Região da AWS. O ambiente de gerenciamento do Kubernetes gerenciado pelo Amazon EKS não pode ser executado na zona local. Como o Local Zones aparece como uma sub-rede na VPC, o Kubernetes considera os recursos da zona local como parte dessa sub-rede.

## AWS Wavelength

[AWS Wavelength](#) é uma infraestrutura da AWS que permite implantar cargas de trabalho mais próximas aos usuários e dispositivos conectados ao 5G. Você pode usar o Wavelength para implantar EC2 instâncias da Amazon, clusters do Amazon EKS e um conjunto de soluções de parceiros compatíveis disponíveis no AWS Marketplace. As zonas de Wavelength são datacenters logicamente isolados nas redes dos provedores de telecomunicações que são conectados de volta à região da AWS por meio de conectividade redundante, de baixa latência e alta taxa de transferência.

Alguns dos principais recursos do Wavelength incluem a capacidade de criar EC2 instâncias da Amazon, volumes do Amazon EBS e sub-redes e gateways de operadoras do Amazon VPC em zonas de comprimento de onda. Você também pode usar serviços que orquestram ou trabalham com Amazon, Amazon EBS e EC2 Amazon VPC, como Amazon Auto EC2 Scaling, clusters Amazon EKS, clusters Amazon ECS, Amazon EC2 Systems Manager, Amazon,, e Application CloudWatch Load AWS CloudTrail Balancer AWS CloudFormation. Os serviços Wavelength fazem parte de uma VPC conectada por meio de uma conexão confiável de alta largura de banda a uma região da AWS para facilitar o acesso a serviços, incluindo o Amazon DynamoDB e o Amazon Relational Database Service (Amazon RDS).

## Serviços adicionais de implantação

O [Amazon Simple Storage Service](#) (Amazon S3) pode ser usado como um servidor web para conteúdo estático e aplicativos de página única (SPA). Combinado com CloudFront a Amazon para aumentar o desempenho na entrega de conteúdo estático, usar o Amazon S3 pode ser uma maneira simples e poderosa de implantar e atualizar conteúdo estático. Mais detalhes sobre essa abordagem podem ser encontrados em [Hospedagem de sites estáticos no AWS](#) whitepaper.

## AWS Proton

[O AWS Proton](#) é um serviço totalmente gerenciado que simplifica e automatiza o processo de implantação e gerenciamento de microsserviços e aplicativos baseados em contêineres. Ele fornece uma experiência de implantação unificada e consistente que se integra a DevOps ferramentas e serviços populares, facilitando o gerenciamento e a simplificação do desenvolvimento de aplicativos. O Proton permite que os desenvolvedores definam e criem componentes de aplicativos, como infraestrutura, código e pipelines, como modelos reutilizáveis. Esses modelos podem ser usados para criar vários ambientes, como desenvolvimento, teste e produção, e podem ser compartilhados entre equipes ou organizações. Essa abordagem ajuda a reduzir a complexidade da implantação e do gerenciamento de microsserviços e aplicativos baseados em contêineres, que podem ser demorados e propensos a erros.

O AWS Proton fornece modelos pré-criados para tipos comuns de microsserviços, como aplicativos web e bancos de dados APIs, que podem ser personalizados para atender a necessidades específicas. Ele também se integra a DevOps ferramentas populares, como AWS CodePipeline, AWS e AWS CodeCommit CodeBuild, para permitir fluxos de trabalho de integração e implantação contínuas (CI/CD).

Ao usar o AWS Proton, os desenvolvedores podem reduzir o tempo e o esforço necessários para implantar e gerenciar microsserviços e aplicativos baseados em contêineres. Essa abordagem permite que as equipes se concentrem no desenvolvimento e no aprimoramento de seus aplicativos, em vez de gastar tempo no processo de implantação e gerenciamento.

## AWS App2Container

[AWS App2Container](#) é uma ferramenta de linha de comando para migrar e modernizar aplicativos web Java e .NET em formato de contêiner. O App2Container analisa e cria um inventário de aplicativos executados em bare metal, máquinas virtuais, EC2 instâncias da Amazon ou na nuvem. Basta selecionar o aplicativo que deseja armazenar e o App2Container empacota o artefato do aplicativo e as dependências identificadas em imagens de contêiner, configura as portas de rede e gera a tarefa do ECS e as definições do pod Kubernetes. O App2Container identifica os aplicativos ASP.NET e Java compatíveis em execução em uma máquina virtual para criar um inventário abrangente de todos os aplicativos em seu ambiente. O App2Container pode containerizar aplicativos web ASP.NET executados no IIS no Windows ou aplicativos Java executados no Linux, autônomos ou em servidores de aplicativos, como Apache Tomcat, Springboot JBoss, IBM Websphere e Oracle Weblogic.



## AWS Copilot

O [AWS Copilot](#) é uma interface de linha de comando (CLI) que você pode usar para iniciar e gerenciar rapidamente aplicativos em contêineres na AWS. Ele simplifica a execução de aplicativos no Amazon ECS, Fargate e App Runner. Atualmente, o AWS Copilot é compatível com sistemas Linux, macOS e Windows. O Copilot permite que você use padrões de serviço, como um serviço web com balanceamento de carga, para provisionar infraestrutura, implantar em vários ambientes, como teste ou produção, e até mesmo usar um pipeline de AWS CodePipeline lançamento para implantações automatizadas.

## AWS Serverless Application Model

O [AWS Serverless Application Model](#) (AWS SAM) é uma estrutura de código aberto para criar aplicativos sem servidor. Ele fornece sintaxe abreviada para expressar funções APIs, bancos de dados e mapeamentos de fontes de eventos. Com apenas algumas linhas por recurso, você pode definir o aplicativo desejado e modelá-lo usando YAML. Durante a implantação, o SAM transforma e expande a sintaxe do SAM na sintaxe da AWS CloudFormation, permitindo que você crie aplicativos sem servidor com mais rapidez.

A AWS SAM CLI é uma ferramenta de linha de comando de código aberto que facilita o desenvolvimento, o teste e a implantação de aplicativos sem servidor na AWS. É uma interface de linha de comando para criar aplicativos sem servidor usando a especificação AWS SAM, que é uma extensão da AWS CloudFormation

A AWS SAM CLI permite que os desenvolvedores definam e testem seus aplicativos sem servidor localmente antes de implantá-los na AWS. Ele fornece um ambiente de teste local que simula o AWS Lambda e o API Gateway, permitindo que os desenvolvedores testem seus códigos e configurações antes de implantá-los na nuvem.

A AWS SAM CLI também inclui uma variedade de recursos úteis, como recursos automáticos de implantação de código, registro e depuração. Ele permite que os desenvolvedores criem, empacotem e implantem seus aplicativos com um único comando, reduzindo o tempo e o esforço necessários para implantar e gerenciar aplicativos sem servidor.

Além disso, a CLI do AWS SAM fornece suporte para várias linguagens de programação, incluindo Node.js, Python, Java e .NET Core, entre outras. Isso permite que os desenvolvedores usem sua linguagem de programação e ferramentas preferidas para criar e implantar seus aplicativos sem servidor.

A CLI do AWS SAM se integra a outros serviços da AWS, como AWS e CodePipeline AWS CodeBuild, para fornecer um pipeline de CI/CD totalmente automatizado e integrado para aplicativos sem servidor. Ele também permite que os desenvolvedores usem outros serviços da AWS, como Amazon S3, Amazon DynamoDB e Amazon SNS, como parte de seus aplicativos sem servidor.

## AWS Cloud Development Kit (AWS CDK)

O [AWS Cloud Development Kit \(AWS CDK\)](#) (AWS CDK) é uma estrutura de desenvolvimento de software de código aberto para definir a infraestrutura de nuvem como código com linguagens de programação modernas e implantá-la por meio da AWS CloudFormation. O AWS Cloud Development Kit (AWS CDK) acelera o desenvolvimento da nuvem usando linguagens de programação comuns para modelar seus aplicativos. O AWS CDK permite que você crie aplicativos confiáveis, escaláveis e econômicos na nuvem com o considerável poder expressivo de uma linguagem de programação.

Pense no AWS CDK como um kit de ferramentas centrado no desenvolvedor que aproveita todo o poder das linguagens de programação modernas para definir sua infraestrutura da AWS como código. Quando os aplicativos do AWS CDK são executados, eles são compilados em modelos CloudFormation JSON/YAML totalmente formados que são então enviados ao serviço para provisionamento. CloudFormation Como o AWS CDK aproveita CloudFormation, você ainda desfruta de todos os benefícios oferecidos CloudFormation, como implantação segura, reversão automática e detecção de desvios.

Essa abordagem gera muitos benefícios, incluindo:

- Crie com construções de alto nível que fornecem automaticamente padrões sensatos e seguros para seus recursos da AWS, definindo mais infraestrutura com menos código.
- Use expressões de programação, como parâmetros, condicionais, loops, composição e herança, para modelar o design do seu sistema a partir de blocos de construção fornecidos pela AWS e outros.
- Coloque sua infraestrutura, código de aplicativo e configuração em um só lugar, garantindo que, em cada etapa, você tenha um sistema completo e implantável na nuvem.
- Empregue práticas de engenharia de software, como análises de código, testes unitários e controle de origem, para tornar sua infraestrutura mais robusta.
- O AWS Solutions Constructs é uma extensão de biblioteca de código aberto do AWS CDK. O AWS Solutions Constructs fornece a você uma coleção de padrões de arquitetura multisserviços aprovados, criados usando as melhores práticas estabelecidas pelo AWS Well-Architected Framework.

O AWS Serverless Application Model e o AWS CDK resumem a infraestrutura da AWS como código, facilitando a definição da sua infraestrutura de nuvem. O AWS SAM se concentra especificamente em casos de uso e arquiteturas sem servidor e permite que você defina sua infraestrutura em modelos JSON/YAML compactos e declarativos. O AWS CDK oferece ampla cobertura em todos os serviços da AWS e permite que você defina a infraestrutura de nuvem em linguagens de programação modernas

## Amazon EC2 Image Builder

EC2 O [Image Builder](#) simplifica a criação, o teste e a implantação de imagens de VM e contêiner para uso na AWS ou no local. Manter imagens de VM e contêiner up-to-date pode ser demorado, consumir muitos recursos e estar sujeito a erros. Atualmente, os clientes atualizam e capturam imagens manualmente VMs ou têm equipes que criam scripts de automação para manter as imagens. O Image Builder reduz significativamente o esforço de manter up-to-date as imagens seguras, fornecendo uma interface gráfica simples, automação integrada e configurações de segurança fornecidas pela AWS. Com o Image Builder, não há etapas manuais para atualizar uma imagem nem você precisa criar seu próprio pipeline de automação. O Image Builder é oferecido gratuitamente, exceto pelo custo dos recursos subjacentes da AWS usados para criar, armazenar e compartilhar as imagens.

EC2 O Image Builder pode ajudar a facilitar as implantações na AWS, simplificando o processo de criação e gerenciamento de imagens personalizadas para uso com a Amazon EC2, contêineres e servidores locais. O serviço fornece uma maneira simplificada e flexível de criar e gerenciar imagens personalizadas, com pipelines de criação automatizados que permitem agilizar o processo de criação e gerenciamento de imagens.

EC2 O Image Builder fornece uma interface fácil de usar que abstrai grande parte do gerenciamento da infraestrutura subjacente, facilitando aos desenvolvedores a criação e o gerenciamento de imagens personalizadas. Com o EC2 Image Builder, os desenvolvedores podem especificar o sistema operacional, os aplicativos e os pacotes que desejam incluir na imagem, e o serviço automatiza o processo de criação e teste da imagem, incluindo atualizações, patches e correções de segurança. Os pipelines de criação automatizados permitem que os desenvolvedores simplifiquem o processo de criação e gerenciamento de imagens, reduzindo o tempo e o esforço necessários para a criação e o teste manuais de imagens. Isso pode ajudar a melhorar a consistência, reduzir erros e garantir que as imagens sejam up-to-date seguras e compatíveis.

A seguir estão alguns dos benefícios do EC2 Image Builder:

- Criação simplificada de EC2 imagens: o Image Builder fornece uma maneira simplificada e flexível de criar imagens personalizadas para uso com a Amazon EC2, contêineres e servidores locais. Isso pode ajudar a reduzir o tempo e o esforço necessários para criar e manter imagens personalizadas e permitir que você se concentre em outros aspectos da implantação, como desenvolvimento e teste de aplicativos.
- Pipelines automatizados de criação de EC2 imagens: o Image Builder fornece canais automatizados para criar, testar e implantar imagens personalizadas, o que pode ajudar a simplificar o processo de criação e gerenciamento de imagens. Isso pode ajudar a garantir que suas imagens estejam up-to-date seguras e em conformidade, além de reduzir o tempo e o esforço necessários para a criação e o teste manuais de imagens.
- Integração com os serviços da AWS: o EC2 Image Builder se integra a outros serviços da AWS, como o Amazon Elastic Container Registry (ECR) e o Amazon Elastic Kubernetes Service (EKS), para permitir que você crie imagens personalizadas para uso com contêineres. Isso pode ajudar a simplificar o processo de criação e implantação de contêineres, permitindo que você crie imagens personalizadas que incluam seus aplicativos, bibliotecas e configurações.
- Criação flexível de imagens: o EC2 Image Builder fornece uma maneira flexível de criar imagens personalizadas, permitindo que você especifique o sistema operacional, os aplicativos e os pacotes que você deseja incluir na imagem. Isso pode ajudar a garantir que suas imagens sejam personalizadas de acordo com seus requisitos e casos de uso específicos, além de reduzir o risco de erros ou incompatibilidades durante a implantação.
- Segurança e conformidade de imagem aprimoradas: o EC2 Image Builder permite automatizar testes de imagem, incluindo verificações de vulnerabilidade e conformidade, para garantir que suas imagens estejam seguras e em conformidade. Isso pode ajudar a reduzir o risco de violações de segurança e melhorar a conformidade, além de permitir que você implante seus aplicativos com confiança.

# Estratégias de implantação

Além de selecionar as ferramentas certas para atualizar o código do aplicativo e a infraestrutura de suporte, implementar os processos de implantação corretos é uma parte essencial de uma solução de implantação completa e que funcione bem. Os processos de implantação que você escolhe para atualizar seu aplicativo podem depender do equilíbrio desejado entre controle, velocidade, custo, tolerância ao risco e outros fatores.

Cada serviço de implantação da AWS oferece suporte a várias estratégias de implantação. Esta seção fornecerá uma visão geral das estratégias de implantação de uso geral que podem ser usadas com sua solução de implantação.

## Pré-cozimento versus inicialização AMIs

Se seu aplicativo depende muito da personalização ou implantação de aplicativos em EC2 instâncias da Amazon, você pode otimizar suas implantações por meio de práticas de inicialização e pré-criação.

Instalar seu aplicativo, dependências ou personalizações sempre que uma instância da Amazon é iniciada é chamado de inicialização de uma EC2 instância. Se você tiver um aplicativo complexo ou precisar de grandes downloads, isso pode retardar as implantações e os eventos de escalabilidade.

Uma [Amazon Machine Image](#) (AMI) fornece as informações necessárias para iniciar uma instância (sistemas operacionais, volumes de armazenamento, permissões, pacotes de software etc.). Você pode executar várias instâncias idênticas a partir de uma única AMI. Sempre que uma EC2 instância é executada, você seleciona a AMI que será usada como modelo. O pré-cozimento é o processo de incorporar uma parte significativa dos artefatos do seu aplicativo em uma AMI.

A pré-integração dos componentes do aplicativo em uma AMI pode acelerar o tempo de lançamento e operacionalização de uma instância da Amazon. EC2 As práticas de pré-preparação e inicialização podem ser combinadas durante o processo de implantação para criar rapidamente novas instâncias personalizadas para o ambiente atual.

## Implantações azuis/verdes

Uma estratégia 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 de implantação aumenta a disponibilidade do aplicativo e reduz o risco de implantação ao simplificar o processo de reversão em caso de falha na implantação. Depois que o teste for concluído no ambiente verde, o tráfego ativo do aplicativo é direcionado para o ambiente verde e o ambiente azul é descontinuado.

Vários serviços de implantação da AWS oferecem suporte a estratégias de implantação azul/verde, incluindo Elastic Beanstalk OpsWorks,, CloudFormation e Amazon ECS. CodeDeploy Consulte [Implantações azul/verdes na AWS](#) para obter mais detalhes e estratégias para implementar processos de implantação azul/verde para seu aplicativo.

## Implantações contínuas

Uma implantação contínua é uma estratégia de implantação que substitui lentamente as versões anteriores de um aplicativo por novas versões de um aplicativo, substituindo completamente a infraestrutura na qual o aplicativo está sendo executado. Por exemplo, em uma implantação contínua no Amazon ECS, os contêineres que executam versões anteriores do aplicativo serão one-by-one substituídos por contêineres que executam novas versões do aplicativo.

Uma implantação contínua geralmente é mais rápida do que uma blue/green deployment; however, unlike a blue/green implantação. Em uma implantação contínua, não há isolamento do ambiente entre a versão antiga e a nova do aplicativo. Isso permite que as implantações contínuas sejam concluídas mais rapidamente, mas também aumenta os riscos e complica o processo de reversão se uma implantação falhar.

Estratégias de implantação contínua podem ser usadas com a maioria das soluções de implantação. [Consulte Políticas de CloudFormation atualização para obter mais informações sobre implantações contínuas com CloudFormation; Atualizações contínuas com o Amazon ECS para obter mais detalhes sobre implantações contínuas com o Amazon ECS; Atualizações de configuração do ambiente contínuo do Elastic Beanstalk para obter mais detalhes sobre implantações contínuas com o Elastic Beanstalk; e Usando uma implantação contínua em para obter mais detalhes sobre implantações contínuas com. AWS OpsWorks](#) OpsWorks

## Implantações do Canary

As [implantações do Canary](#) são um tipo de estratégia de implantação azul/verde que é mais avessa ao risco. Essa estratégia envolve uma abordagem em fases na qual o tráfego é transferido para uma nova versão do aplicativo em dois incrementos. O primeiro incremento é uma pequena porcentagem

do tráfego, chamada de grupo canário. Esse grupo é usado para testar a nova versão e, se for bem-sucedido, o tráfego será transferido para a nova versão no segundo incremento.

As implantações do Canary podem ser implementadas em duas etapas ou linearmente. Na abordagem em duas etapas, o novo código do aplicativo é implantado e exposto para teste. Após a aceitação, ele é lançado para o resto do ambiente ou de forma linear. A abordagem linear envolve o aumento incremental do tráfego para a nova versão do aplicativo até que todo o tráfego flua para a nova versão.

## Implantações no local

Uma [implantação local](#) é uma estratégia de implantação que atualiza a versão do aplicativo sem substituir nenhum componente da infraestrutura. Em uma implantação local, a versão anterior do aplicativo em cada recurso computacional é interrompida, o aplicativo mais recente é instalado e a nova versão do aplicativo é iniciada e validada. Isso permite que as implantações de aplicativos prossigam com o mínimo de perturbação na infraestrutura subjacente.

Uma implantação local permite que você implante seu aplicativo sem criar uma nova infraestrutura; no entanto, a disponibilidade do seu aplicativo pode ser afetada durante essas implantações. Essa abordagem também minimiza os custos de infraestrutura e a sobrecarga de gerenciamento associados à criação de novos recursos.

Consulte [Visão geral de uma implantação local](#) para obter mais detalhes sobre o uso de estratégias de implantação local com CodeDeploy.

## Combinando serviços de implantação

Não existe uma solução de implantação “única para todos” na AWS. No contexto da criação de uma solução de implantação, é importante considerar o tipo de aplicativo, pois isso pode ditar quais serviços da AWS são mais apropriados. Para fornecer funcionalidade completa para provisionar, configurar, implantar, escalar e monitorar seu aplicativo, geralmente é necessário combinar vários serviços de implantação.

Um padrão comum para aplicativos na AWS é usar CloudFormation (e suas extensões) para gerenciar a infraestrutura de uso geral e usar uma solução de implantação mais especializada para gerenciar atualizações de aplicativos. No caso de um aplicativo em contêineres, CloudFormation poderia ser usado para criar a infraestrutura do aplicativo, e o Amazon ECS e o Amazon EKS poderiam ser usados para provisionar, implantar e monitorar contêineres.

Os serviços de implantação da AWS também podem ser combinados com serviços de implantação de terceiros. Isso permite que as organizações integrem facilmente os serviços de implantação da AWS em seus 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 serviços existentes como parte de um pipeline completo.



## Conclusão

A AWS fornece várias ferramentas para simplificar e automatizar o provisionamento da infraestrutura e a implantação de aplicativos; cada serviço de implantação oferece recursos diferentes para gerenciar aplicativos. Para criar uma arquitetura de implantação bem-sucedida, avalie os recursos disponíveis de cada serviço em relação às necessidades do seu aplicativo e da sua organização.

# Colaboradores

Os colaboradores deste documento incluem:

- Manikandan Chandrasekaran, tecnólogo principal
- Anil Nadiminti, arquiteto sênior de soluções
- Bryant Bost, consultor da AWS ProServe

## Outras fontes de leitura

Para obter informações adicionais, consulte:

- [Página de whitepapers da AWS](#)
- [Introdução DevOps à AWS — Estratégias de implantação](#)

# Revisões do documento

Para receber notificação sobre atualizações nesse whitepaper, inscreva-se no feed RSS.

Alteração	Descrição	Data
<a href="#">Whitepaper atualizado</a>	Atualizado por completo para os mais recentes serviços e estratégias de implantação	31 de maio de 2024
<a href="#">Atualização secundária</a>	A seção de implantações em azul/verde foi revisada para maior clareza.	8 de abril de 2021
<a href="#">Whitepaper atualizado</a>	Atualizado com os serviços e recursos mais recentes.	3 de junho de 2020
<a href="#">Publicação inicial</a>	Whitepaper publicado pela primeira vez	1 de março de 2015

# Avisos

Os clientes são responsáveis por fazer uma avaliação independente das informações contidas neste documento. Este documento: (a) serve apenas para fins informativos, (b) representa as práticas e ofertas atuais de produtos da AWS, que estão sujeitas a alterações sem aviso prévio, e (c) não cria nenhum compromisso ou garantia por parte da AWS e de seus afiliados, fornecedores ou licenciadores. Os produtos ou serviços da AWS são fornecidos “no estado em que se encontram”, sem garantias, representações ou condições de qualquer tipo, expressas ou implícitas. As responsabilidades e as obrigações da AWS para com os clientes são controladas por contratos da AWS, e este documento não faz parte nem modifica nenhum contrato entre a AWS e seus clientes.

© 2024 Amazon Web Services, Inc. ou suas afiliadas. Todos os direitos reservados.

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.