aws

Implementando o PostgreSQL gerenciado para aplicativos SaaS multilocatários em AWS

AWS Orientação prescritiva

AWS Orientação prescritiva: Implementando o PostgreSQL gerenciado para aplicativos SaaS multilocatários em AWS

Table of Contents

Implementando o PostgreSQL gerenciado para aplicativos SaaS multilocatários em AWS

Tabby Ward e Thomas Davis, da Amazon Web Services ()AWS

Abril de 2024 (histórico do documento)

Quando você seleciona um banco de dados para armazenar dados operacionais, é fundamental considerar como os dados devem ser estruturados, quais consultas eles responderão, com que rapidez fornecerão respostas e a resiliência da própria plataforma de dados. Além dessas considerações gerais, há implicações de software como serviço (SaaS) para dados operacionais, como isolamento de desempenho, segurança de inquilinos e características e padrões de design exclusivos que são típicos de dados para aplicativos SaaS multilocatários. Este guia discute como esses fatores se aplicam ao uso de um banco de dados PostgreSQL na Amazon Web Services AWS() como armazenamento de dados operacional primário para um aplicativo SaaS multilocatário. Especificamente, o guia se concentra em duas opções AWS gerenciadas do PostgreSQL: Amazon Aurora PostgreSQL Compatible Edition e Amazon Relational Database Service (Amazon RDS) para PostgreSQL.

Resultados de negócios desejados

Esta orientação fornece uma análise detalhada das melhores práticas para aplicativos SaaS multilocatários usando o Aurora PostgreSQL compatível e o Amazon RDS for PostgreSQL. Recomendamos que você use os padrões e conceitos de design fornecidos neste guia para informar e padronizar sua implementação do Aurora compatível com PostgreSQL ou do Amazon RDS for PostgreSQL para seus aplicativos SaaS multilocatários.

Essa orientação prescritiva ajuda a alcançar os seguintes resultados comerciais:

- Escolhendo a melhor opção AWS gerenciada do PostgreSQL para seu caso de uso — Esta orientação compara opções relacionais e não relacionais para uso de banco de dados com aplicativos SaaS. Também discute quais casos de uso são mais adequados para o Aurora PostgreSQL compatível e o Amazon RDS for PostgreSQL. Essas informações ajudarão a selecionar a melhor opção para seu aplicativo SaaS.
- Aplicação das melhores práticas de SaaS por meio da adoção de um modelo de particionamento SaaS — Este guia discute e compara três modelos amplos de particionamento de SaaS que são

aplicáveis a um sistema de gerenciamento de banco de dados (DBMS) PostgreSQL: modelos de pool, bridged e silo e suas variações. Essas abordagens capturam as melhores práticas de SaaS e oferecem flexibilidade ao projetar um aplicativo SaaS. A aplicação de um modelo de particionamento SaaS é uma parte crucial da preservação das melhores práticas.

- Uso efetivo do RLS em modelos de particionamento SaaS de pool — a segurança em nível de linha (RLS) suporta a imposição do isolamento de dados do locatário em uma única tabela do PostgreSQL, restringindo as linhas que podem ser visualizadas com base no usuário ou em uma variável de contexto. Quando você usa o modelo de particionamento de pool, o RLS é necessário para impedir o acesso entre inquilinos.

Seleção de um banco de dados para um aplicativo SaaS

Para muitos aplicativos SaaS multilocatários, a seleção de um banco de dados operacional pode ser resumida em uma escolha entre bancos de dados relacionais e não relacionais, ou uma combinação dos dois. Para tomar sua decisão, considere estes requisitos e características de dados de aplicativos de alto nível:

- Modelo de dados do aplicativo
- Padrões de acesso para os dados
- Requisitos de latência do banco de dados
- Requisitos de integridade de dados e integridade transacional (atomicidade, consistência, isolamento e durabilidade, ou ACID)
- Requisitos de disponibilidade e recuperação entre regiões

A tabela a seguir lista os requisitos e as características dos dados do aplicativo e os discute no contexto das ofertas de AWS banco de dados: Aurora PostgreSQL compatível e Amazon RDS for PostgreSQL (relacional) e Amazon DynamoDB (não relacional). Você pode consultar essa matriz quando estiver tentando decidir entre ofertas de bancos de dados operacionais relacionais e não relacionais.

Bancos de dados	Requisitos e características dos dados do aplicativo SaaS				
	Modelo de dados	Padrões de acesso	Requisitos de latência	Integridade transacional e de dados	Disponibilidade e recuperação entre regiões
Relacional (Compatível com Aurora PostgreSQL e Amazon RDS para PostgreSQL)	Relacional ou altamente normalizado.	Não precisa ser cuidadosamente planejado com antecedência.	De preferência, maior tolerância à latência; pode alcançar latências mais baixas por padrão	Alta integridade transacional e de dados mantida por padrão.	No Amazon RDS, você pode criar uma réplica de leitura para escalabilidade e failover entre

			com o Aurora e implementando réplicas de leitura, armazenamento em cache e recursos semelhantes.		regiões. O Aurora automatiza a principalmente esse processo. Para configurações ativo-ativas em várias Regiões da AWS, você pode usar o encaminhamento de gravação em conjunto com os bancos de dados globais do Aurora.
Não relacional (Amazon DynamoDB)	Geralmente desnormalizado. Esses bancos de dados aproveitam os padrões para modelar many-to-many relacionamentos, itens grandes e dados de séries temporais.	Todos os padrões de acesso (consultas) aos dados devem ser completamente compreendidos antes que um modelo de dados seja produzido.	Latência muito baixa com opções como o Amazon DynamoDB Accelerator (DAX) capazes de melhorar ainda mais o desempenho.	Integridade transacional opcional ao custo do desempenho. As preocupações com a integridade dos dados são transferidas para o aplicativo.	Fácil recuperação entre regiões e configuração ativa-ativa com tabelas globais. (A conformidade com o ACID só é possível em uma única AWS região.)

Alguns aplicativos SaaS multilocatários podem ter modelos de dados exclusivos ou circunstâncias especiais que são melhor atendidas por bancos de dados não incluídos na tabela anterior. Por exemplo, conjuntos de dados de séries temporais, conjuntos de dados altamente conectados ou a manutenção de um livro de transações centralizado podem exigir o uso de um tipo diferente de banco de dados. Analisar todas as possibilidades está além do escopo deste guia. Para obter uma lista abrangente de ofertas de AWS banco de dados e como elas podem atender a diferentes casos de uso em alto nível, consulte a seção Banco de dados do whitepaper Visão geral da Amazon Web Services.

O restante deste guia se concentra nos serviços de banco de dados AWS relacional que oferecem suporte ao PostgreSQL: compatível com Amazon RDS e Aurora PostgreSQL. O DynamoDB exige uma abordagem diferente para otimizar aplicativos SaaS, o que está além do escopo deste guia. Para obter mais informações sobre o DynamoDB, consulte a postagem do blog Particionando dados SaaS AWS multilocatários agrupados com o Amazon DynamoDB.

Escolhendo entre Amazon RDS e Aurora

Na maioria dos casos, recomendamos usar o Aurora compatível com PostgreSQL em vez do Amazon RDS for PostgreSQL. A tabela a seguir mostra os fatores que você deve considerar ao decidir entre essas duas opções.

Componente DBMS	Amazon RDS para PostgreSQL	Compatível com Aurora PostgreSQL
Escalabilidade	Atraso de replicação de minutos, máximo de 5 réplicas de leitura	Atraso de replicação inferior a um minuto (normalmente menos de 1 segundo com bancos de dados globais), máximo de 15 réplicas de leitura
Recuperação de falhas	Pontos de verificação com 5 minutos de intervalo (por padrão) podem diminuir o desempenho do banco de dados	Recuperação assíncrona com threads paralelos para recuperação rápida

Componente DBMS	Amazon RDS para PostgreSQL	Compatível com Aurora PostgreSQL
Failover	60-120 segundos, além do tempo de recuperação de falhas	Normalmente, cerca de 30 segundos (incluindo recuperação de falhas)
Armazenamento	IOPS máximo de 256.000	IOPS restringido somente pelo tamanho e capacidade da instância do Aurora
Alta disponibilidade e recuperação de desastres	Duas zonas de disponibilidade com uma instância em espera, failover entre regiões para ler réplicas ou backups copiados	Três zonas de disponibilidade por padrão, failover entre regiões com bancos de dados globais Aurora, encaminhamento de gravação para configurações ativo-ativas Regiões da AWS
Backup	Durante a janela de backup, pode afetar o desempenho	Backups incrementais automáticos, sem impacto no desempenho
Classes de instâncias de banco	Veja a lista de classes de instância do Amazon RDS	Veja a lista de classes de instância do Aurora

Em todas as categorias descritas na tabela anterior, o Aurora PostgreSQL compatível geralmente é a melhor opção. No entanto, o Amazon RDS for PostgreSQL ainda pode fazer sentido para cargas de trabalho de pequeno a médio porte, porque tem uma seleção maior de classes de instância que podem fornecer uma opção mais econômica às custas do conjunto de recursos mais robusto do Aurora.

Modelos de particionamento SaaS multilocatário para PostgreSQL

O melhor método para realizar a multilocação depende dos requisitos do seu aplicativo SaaS. As seções a seguir demonstram modelos de particionamento para implementar com êxito a multilocação no PostgreSQL.

> **Note**
>
> Os modelos discutidos nesta seção são aplicáveis tanto ao Amazon RDS for PostgreSQL quanto ao Aurora PostgreSQL. As referências ao PostgreSQL nesta seção se aplicam aos dois serviços.

Há três modelos de alto nível que você pode usar no PostgreSQL para particionamento SaaS: silo, ponte e pool. A imagem a seguir resume as compensações entre os modelos de silo e piscina. O modelo de ponte é um híbrido dos modelos de silo e piscina.

Modelo de particionamento	Vantagens	Desvantagens
Silo	• Alinhamento de conformidade • Sem impacto entre inquilinos • Ajuste no nível do inquilino • Disponibilidade em nível de inquilino	• Agilidade comprometida • Sem gerenciamento centralizado • Complexidade de implantação • Custos
Piscina	• Agilidade • Otimização de custo • Gerenciamento centralizado • Implantação simplificada	• Impacto entre inquilinos • Desafios de conformidade • Disponibilidade de tudo ou nada
Ponte	• Algum alinhamento de conformidade	• Alguns desafios de conformidade

Modelo de particionamento	Vantagens	Desvantagens
	• Agilidade • Otimização de custo • Gerenciamento centralizado	• Disponibilidade de tudo ou nada (principalmente) • Impacto entre inquilinos • Complexidade de implantação

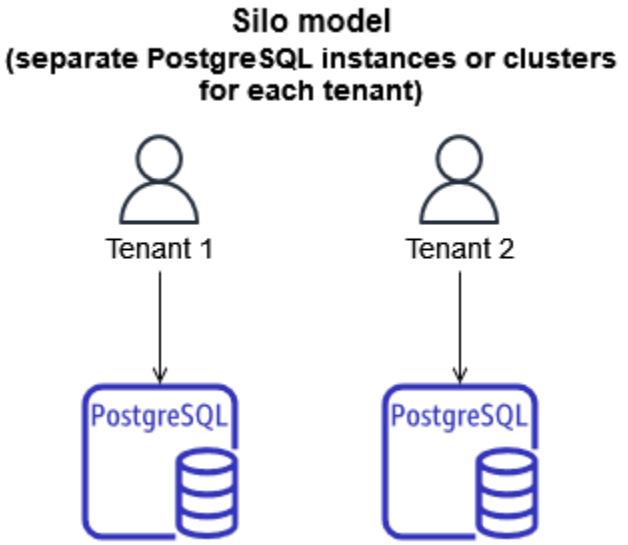
As seções a seguir abordam cada modelo com mais detalhes.

Modelos de particionamento:

- Modelo de silo do PostgreSQL
- Modelo de pool do PostgreSQL
- Modelo de ponte PostgreSQL
- Matriz de decisão

Modelo de silo do PostgreSQL

O modelo de silo é implementado com o provisionamento de uma instância do PostgreSQL para cada locatário em um aplicativo. O modelo de silo se destaca no desempenho do inquilino e no isolamento de segurança e elimina completamente o fenômeno ruidoso do vizinho. O fenômeno do vizinho ruidoso ocorre quando o uso de um sistema por um inquilino afeta o desempenho de outro inquilino. O modelo de silo permite que você adapte o desempenho especificamente a cada inquilino e potencialmente limite as interrupções ao silo de um inquilino específico. No entanto, o que geralmente impulsiona a adoção de um modelo de silo são as rígidas restrições regulatórias e de segurança. Essas restrições podem ser motivadas pelos clientes de SaaS. Por exemplo, os clientes de SaaS podem exigir que seus dados sejam isolados devido a restrições internas, e os provedores de SaaS podem oferecer esse serviço por uma taxa adicional.

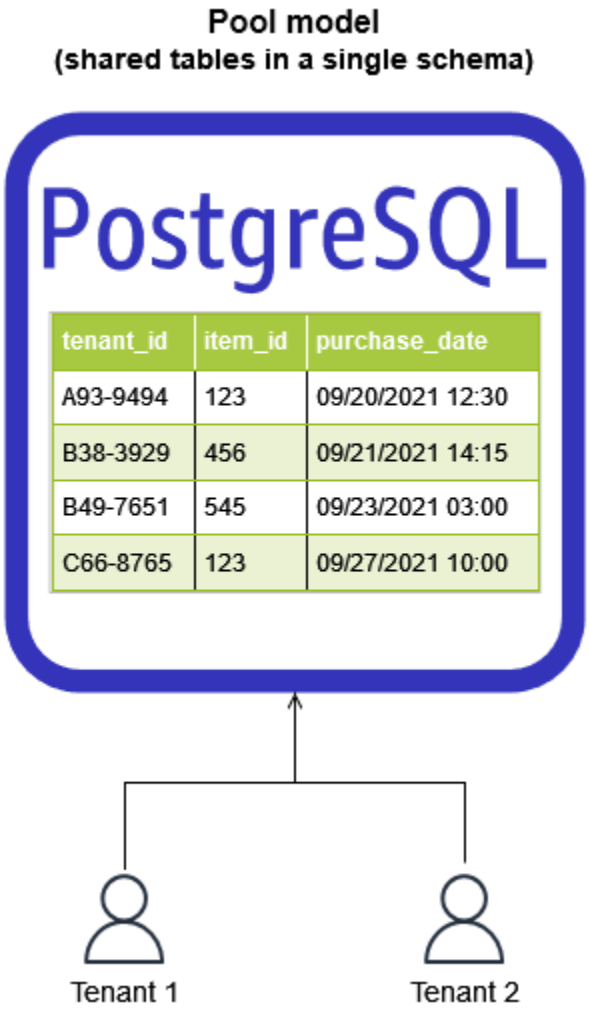


Embora o modelo de silo possa ser necessário em certos casos, ele tem muitas desvantagens. Muitas vezes, é difícil usar o modelo de silo de forma econômica, porque gerenciar o consumo de recursos em várias instâncias do PostgreSQL pode ser complicado. Além disso, a natureza distribuída das cargas de trabalho do banco de dados nesse modelo torna mais difícil manter uma visão centralizada da atividade dos inquilinos. Gerenciar tantas cargas de trabalho operadas de forma independente aumenta a sobrecarga operacional e administrativa. O modelo de silo também torna a integração de inquilinos mais complicada e demorada, porque você precisa provisionar recursos específicos para os inquilinos. Além disso, todo o sistema SaaS pode ser mais difícil de escalar, porque o número cada vez maior de instâncias PostgreSQL específicas para locatários exigirá mais tempo operacional para serem administradas. Uma última consideração é que um aplicativo ou uma camada de acesso a dados precisará manter um mapeamento dos locatários para suas instâncias associadas do PostgreSQL, o que aumenta a complexidade da implementação desse modelo.

Modelo de pool do PostgreSQL

O modelo de pool é implementado provisionando uma única instância do PostgreSQL (Amazon RDS ou Aurora) e usando segurança em nível de linha (RLS) para manter o isolamento dos dados do locatário. As políticas de RLS restringem quais linhas em uma tabela são retornadas por`SELECT` consultas ou quais linhas são afetadas por`INSERTUPDATE`, e`DELETE` comandos. O modelo de pool centraliza todos os dados do locatário em um único esquema PostgreSQL, portanto, é significativamente mais econômico e exige menos sobrecarga operacional para ser mantido. O monitoramento dessa solução também é significativamente mais simples devido à sua centralização.

No entanto, o monitoramento dos impactos específicos do inquilino no modelo da piscina geralmente requer alguma instrumentação adicional na aplicação. Isso ocorre porque o PostgreSQL, por padrão, não sabe qual inquilino está consumindo recursos. A integração de inquilinos é simplificada porque nenhuma nova infraestrutura é necessária. Essa agilidade facilita a realização de fluxos de trabalho rápidos e automatizados de integração de inquilinos.



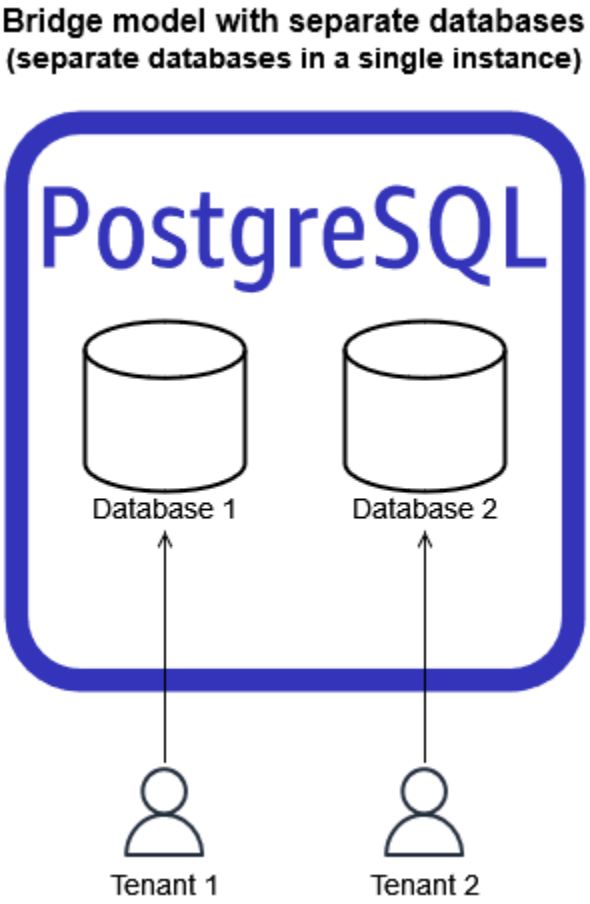
Embora o modelo de piscina seja geralmente mais econômico e mais simples de administrar, ele tem algumas desvantagens. O fenômeno do vizinho barulhento não pode ser completamente eliminado em um modelo de piscina. No entanto, isso pode ser mitigado garantindo que os recursos apropriados estejam disponíveis na instância do PostgreSQL e usando estratégias para reduzir a carga no PostgreSQL, como descarregar consultas para ler réplicas ou para a Amazon ElastiCache. O monitoramento eficaz também desempenha um papel na resposta às questões de isolamento do desempenho do inquilino, porque a instrumentação do aplicativo pode registrar e monitorar a

atividade específica do inquilino. Por fim, alguns clientes de SaaS podem não achar que a separação lógica fornecida pelo RLS é suficiente e podem solicitar medidas adicionais de isolamento.

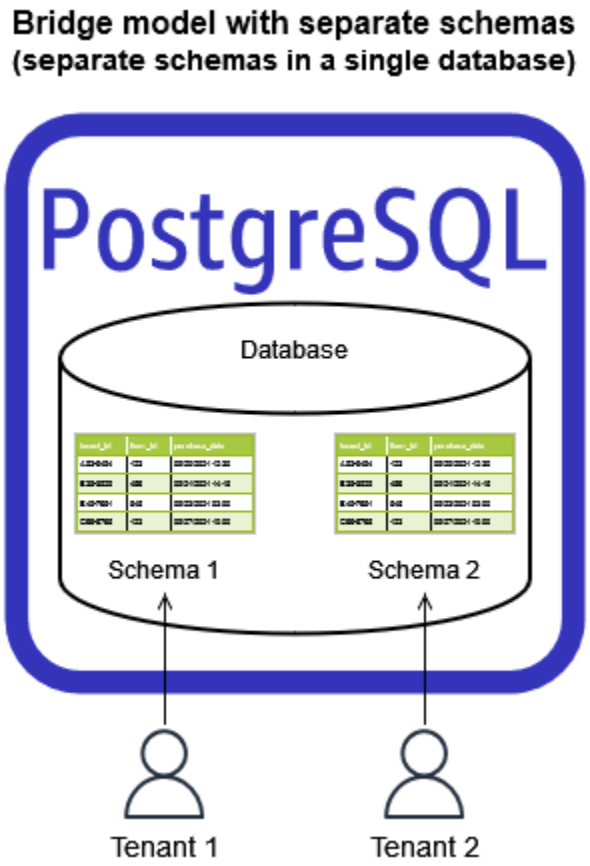
Modelo de ponte PostgreSQL

O modelo de bridge do PostgreSQL é uma combinação das abordagens agrupadas e isoladas. Como no modelo agrupado, você provisiona uma única instância do PostgreSQL para cada inquilino. Para manter o isolamento dos dados do inquilino, você usa construções lógicas do PostgreSQL. No diagrama a seguir, os bancos de dados PostgreSQL são usados para separar dados logicamente.

> **Note**
>
> Um banco de dados PostgreSQL não se refere a uma instância de banco de dados separada do Amazon RDS for PostgreSQL ou do Aurora PostgreSQL. Em vez disso, ele se refere a uma construção lógica do sistema de gerenciamento de banco de dados PostgreSQL para separar dados.



Você também pode implementar o modelo de ponte usando um único banco de dados PostgreSQL, com esquemas específicos do locatário em cada banco de dados, conforme ilustrado no diagrama a seguir.



O modelo de ponte sofre das mesmas preocupações ruidosas de isolamento de desempenho do vizinho e do inquilino do modelo de piscina. Também incorre em algumas despesas operacionais e de provisionamento adicionais, exigindo que bancos de dados ou esquemas separados sejam provisionados por inquilino. Ela exige um monitoramento eficaz para responder rapidamente às preocupações de desempenho dos inquilinos. Também requer instrumentação do aplicativo para monitorar o uso específico do inquilino. No geral, o modelo de ponte pode ser visto como uma alternativa ao RLS, que aumenta um pouco o esforço de integração de inquilinos ao exigir novos bancos de dados ou esquemas PostgreSQL. Assim como no modelo de silo, um aplicativo ou uma camada de acesso a dados precisará manter um mapeamento dos locatários em seus bancos de dados ou esquemas PostgreSQL associados.

Matriz de decisão

Para decidir qual modelo de particionamento SaaS multilocatário você deve usar com o PostgreSQL, consulte a seguinte matriz de decisão. A matriz analisa essas quatro opções de particionamento:

- Silo: uma instância ou cluster de PostgreSQL separado para cada locatário.
- Bridge com bancos de dados separados — Um banco de dados separado para cada locatário em uma única instância ou cluster do PostgreSQL.
- Bridge com esquemas separados — Um esquema separado para cada locatário em um único banco de dados PostgreSQL, em uma única instância ou cluster do PostgreSQL.
- Pool — Tabelas compartilhadas para inquilinos em uma única instância e esquema.

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Caso de uso	O isolamento de dados com controle total do uso de recursos é um requisito fundamental, caso contrário, você tem inquilinos muito grandes e muito sensíveis ao desempenho.	O isolamento de dados é um requisito fundamental, e é necessária uma referência cruzada limitada ou nenhuma referência cruzada dos dados dos inquilinos.	Número moderado de inquilinos com uma quantidade moderada de dados. Esse é o modelo preferido se você precisar cruzar os dados dos inquilinos.	Grande número de inquilinos com menos dados por inquilino.
Nova agilidade de integração de inquilinos	Muito lento. (É necessária uma nova instância ou cluster para cada inquilino.)	Moderadamente lento. (Requer a criação de um novo banco de dados para cada inquilino armazenar objetos do esquema.)	Moderadamente lento. (Requer a criação de um novo esquema para cada inquilino armazenar objetos.)	Opção mais rápida. (É necessária uma configuração mínima.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Esforço e eficiência na configuração do pool de conexões do banco de dados	É necessário um esforço significativo. (Um pool de conexões por inquilino.) Menos eficiente. (Não há compartilhamento de conexão de banco de dados entre inquilinos.)	É necessário um esforço significativo. (Uma configuração de pool de conexões por locatário, a menos que você use o Amazon RDS Proxy.) Menos eficiente. (Sem compartilhamento de conexão de banco de dados entre locatários e número total de conexões. O uso em todos os locatários é limitado com base na classe da instância de instância de banco de dados.)	É necessário menos esforço. (Uma configuração de pool de conexões para todos os inquilinos.) Moderadamente eficiente. (Reutilização da conexão por meio do `SET SCHEMA` comando `SET ROLE` or somente no modo de pool de sessões. `SET`os comandos também causam a fixação da sessão ao usar o Amazon RDS Proxy, mas os pools de conexões do cliente podem ser eliminados e conexões diretas podem ser feitas para	É necessário o mínimo esforço. Mais eficiente. (Um pool de conexões para todos os inquilinos e reutilização eficiente da conexão em todos os inquilinos. Os limites de conexão do banco de dados.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
			cada solicitação de eficiência.)	
Manutenção do banco de dados (gerenciamento de vácuo) e uso de recursos	Gerenciamento mais simples.	Complexidade média. (Pode levar a um alto consumo de recursos, porque um aspirador precisa ser iniciado posteriormente para cada banco de dados`vacuum_naptime`, o que leva ao alto uso da CPU do iniciador automático. Também pode haver uma sobrecarga adicional associada à limpeza das tabelas do catálogo do sistema PostgreSQL para cada banco de dados.)	Tabelas de catálogos do sistema PostgreSQL. (`pg_catalog` Tamanho total em dezenas de GBs, dependendo do número de inquilinos e relações. Provavelmente exigirá modificações nos parâmetros relacionados à aspiração para controlar o inchaço da mesa.)	As tabelas podem ser grandes, dependendo do número de inquilinos e dos dados por inquilino. (Provavelmente exigirá modificações nos parâmetros relacionados à aspiração para controlar o inchaço da mesa.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Esforço de gerenciamento de extensões	Esforço significativo (para cada banco de dados em instâncias separadas).	Esforço significativo (em cada nível de banco de dados).	Esforço mínimo (uma vez no banco de dados comum).	Esforço mínimo (uma vez no banco de dados comum).
Mude o esforço de implantação	Esforço significativo. (Connect a cada instância separada e implemente as alterações.)	Esforço significativo. (Connect a cada banco de dados e esquema e implemente as alterações.)	Esforço moderado. (Connect ao banco de dados comum e implemente alterações para cada esquema.)	Esforço mínimo. (Connect ao banco de dados comum e implemente as alterações.)
Implantação de mudanças — escopo do impacto	Mínimo. (Único inquilino afetado.)	Mínimo. (Único inquilino afetado.)	Mínimo. (Único inquilino afetado.)	Muito grande. (Todos os inquilinos afetados.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Gerenciamento de desempenho e esforço de consultas	Desempenho de consultas gerenciável.	Desempenho de consultas gerenciável.	Desempenho de consultas gerenciável.	É provável que seja necessário um esforço significativo para manter o desempenho da consulta. (Com o tempo, as consultas podem ser executadas mais lentamente devido ao aumento do tamanho das tabelas. Você pode usar o particionamento de tabelas e a fragmentação do banco de dados para manter o desempenho.)
Impacto dos recursos entre inquilinos	Sem impacto. (Sem compartilhamento de recursos entre inquilinos.)	Impacto moderado. (Os inquilinos compartilham recursos comuns, como CPU e memória da instância.)	Impacto moderado. (Os inquilinos compartilham recursos comuns, como CPU e memória da instância.)	Impacto pesado. (Os inquilinos afetam uns aos outros em termos de recursos, conflitos de bloqueio e assim por diante.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Ajuste em nível de inquilino (por exemplo, criação de índices adicionais por inquilino ou ajuste de parâmetros de banco de dados para um determinado inquilino)	Possível.	Um pouco possível. (Alterações no nível do esquema podem ser feitas para cada inquilino, mas os parâmetros do banco de dados são globais em todos os inquilinos.)	Um pouco possível. (Alterações no nível do esquema podem ser feitas para cada inquilino, mas os parâmetros do banco de dados são globais em todos os inquilinos.)	Não é possível. (As mesas são compartilhadas por todos os inquilinos.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Reequilibre os esforços para inquilinos sensíveis ao desempenho	Mínimo. (Não é necessário reequilibrar. Dimensione os recursos do servidor e de I/O para lidar com esse cenário.)	Moderado. (Use a replicação lógica ou `pg_dump` para exportar o banco de dados, mas o tempo de inatividade pode ser longo, dependendo do tamanho dos dados. Você pode usar o recurso de banco de dados transportável no Amazon RDS for PostgreSQL para copiar bancos de dados entre instâncias mais rapidamente.)	Moderado, mas provavelmente envolve um longo tempo de inatividade. (Use a replicação lógica ou `pg_dump` para exportar o esquema, mas o tempo de inatividade pode ser longo, dependendo do tamanho dos dados.)	Significativo, porque todos os inquilinos compartilham as mesmas mesas. (A fragmentação do banco de dados exige a cópia de tudo para outra instância e uma etapa adicional para limpar os dados do locatário.) O mais provável é que exija uma alteração na lógica do aplicativo.

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Tempo de inatividade do banco de dados	Tempo de inatividade de padrão. (Depende do tamanho do catálogo do sistema PostgreSQL.)	É provável que haja um maior tempo de inatividade. (Dependendo do tamanho do catálogo do sistema, o tempo pode variar. (As tabelas do catálogo do sistema PostgreSQL também são duplicadas nos bancos de dados)	É provável que haja um maior tempo de inatividade. (Dependendo do tamanho do catálogo do sistema PostgreSQL, o tempo pode variar.)	Tempo de inatividade de padrão. (Depende do tamanho do catálogo do sistema PostgreSQL.)
Sobrecarga de administração (por exemplo, para análise de registros de banco de dados ou monitoramento de tarefas de backup)	Esforço significativo	Esforço mínimo.	Esforço mínimo.	Esforço mínimo.

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Disponibilidade em nível de inquilino	Mais alto. (Cada inquilino falha e se recupera de forma independente.)	Maior escopo de impacto. (Todos os inquilinos falham e se recuperam juntos em caso de problemas de hardware ou recursos.)	Maior escopo de impacto. (Todos os inquilinos falham e se recuperam juntos em caso de problemas de hardware ou recursos.)	Maior escopo de impacto. (Todos os inquilinos falham e se recuperam juntos em caso de problemas de hardware ou recursos.)
Esforço de backup e recuperação em nível de locatário	Menor esforço. (O backup de cada locatário pode ser feito e restaurado de forma independente.)	Esforço moderado. (Use exportação e importação lógicas para cada inquilino. Alguma codificação e automação são necessárias.)	Esforço moderado. (Use exportação e importação lógicas para cada inquilino. Alguma codificação e automação são necessárias.)	Esforço significativo. (Todos os inquilinos compartilham as mesmas mesas.)
Esforço de point-in-time recuperação em nível de inquilino	Esforço mínimo. (Use a recuperação pontual usando snapshots ou use o retrocesso no Amazon Aurora.)	Esforço moderado. (Use restauração de instantâneos, seguida de exportação/importação. No entanto, essa será uma operação lenta.)	Esforço moderado. (Use restauração de instantâneos, seguida de exportação/importação. No entanto, essa será uma operação lenta.)	Esforço e complexidade significativos.

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Nome do esquema uniforme	Mesmo nome do esquema para cada locatário.	Mesmo nome do esquema para cada locatário.	Esquema diferente para cada locatário.	Esquema comum.
Personalização por inquilino (por exemplo, colunas de tabela adicionais para um inquilino específico)	Possível.	Possível.	Possível.	Complicado (porque todos os inquilinos compartilham as mesmas mesas).

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Eficiência do gerenciamento de catálogos na camada de mapeamento relacional de objetos (ORM) (por exemplo, Ruby)	Eficiente (porque a conexão com o cliente é específica para um inquilino).	Eficiente (porque a conexão do cliente é específica para um banco de dados).	Moderadamente eficiente. (Dependendo do ORM usado, do modelo de segurança do usuário/função e `dasearch_path` configuração, o cliente às vezes armazena em cache os metadados de todos os locatários, levando a um alto uso da memória da conexão de banco de dados.)	Eficiente (porque todos os inquilinos compartilham as mesmas mesas).

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Esforço consolidado de relatórios de inquilinos	Esforço significativo. (Você precisa usar wrappers de dados externos [FDWs] para consolidar dados em todos os locatários ou extrair, transformar e carregar [ETL] em outro banco de dados de relatórios.)	Esforço significativo. (Você precisa usar FDWs para consolidar dados em todos os inquilinos ou ETL em outro banco de dados de relatórios.)	Esforço moderado. (Você pode agregar dados em todos os esquemas usando uniões.)	Esforço mínimo. (Todos os dados do inquilino estão nas mesmas tabelas, portanto, os relatórios são simples.)
Instância somente leitura específica do inquilino para geração de relatórios (por exemplo, com base na assinatura)	Menor esforço. (Crie uma réplica de leitura.)	Esforço moderado. (Você pode usar a replicação lógica ou oAWS Database Migration Service [AWS DMS] para configurar.)	Esforço moderado. (Você pode usar a replicação lógica ouAWS DMS configurar.)	Complicado (porque todos os inquilinos compartilham as mesmas mesas).

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Isolamento de dados	Melhor.	Melhor. (Você pode gerenciar permissões em nível de banco de dados usando funções do PostgreSQL.)	Melhor. (Você pode gerenciar permissões em nível de esquema usando funções do PostgreSQL.)	Pior. (Como todos os inquilinos compartilham as mesmas tabelas, você precisa implementar recursos como segurança em nível de linha [RLS] para isolamento de inquilinos.)
Chave de criptografia de armazenamento específica do locatário	Possível. (Cada cluster do PostgreSQL pode ter sua própria chaveAWS Key Management Service [AWS KMS] para criptografia de armazenamento.)	Não é possível. (Todos os inquilinos compartilham a mesma chave KMS para criptografia de armazenamento.)	Não é possível. (Todos os inquilinos compartilham a mesma chave KMS para criptografia de armazenamento.)	Não é possível. (Todos os inquilinos compartilham a mesma chave KMS para criptografia de armazenamento.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
UsandoAWS Identity and Access Managemen t (IAM) para autenticação de banco de dados para cada inquilino	Possível.	Possível.	Possível (com usuários separados do PostgreSQ L para cada esquema).	Não é possível (porque as mesas são compartilhadas por todos os inquilinos).
Custo da infraestrutura	Mais alto (porque nada é compartilhado).	Moderado.	Moderado.	Mais baixo.
Duplicação de dados e uso do armazenamento	Maior agregado entre todos os inquilino s. (As tabelas do catálogo do sistema PostgreSQL e os dados estáticos e comuns do aplicativo são duplicados em todos os locatários.)	Maior agregado entre todos os inquilino s. (As tabelas do catálogo do sistema PostgreSQL e os dados estáticos e comuns do aplicativo são duplicados em todos os locatários.)	Moderado. (Os dados estáticos e comuns do aplicativo podem estar em um esquema comum e acessados por outros locatário s.)	Mínimo. (Sem duplicação de dados. Os dados estáticos e comuns do aplicativo podem estar no mesmo esquema.)

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Monitoramento centrado no inquilino (descubra rapidamente qual inquilino está causando problemas)	Menor esforço. (Como cada inquilino é monitorado separadamente, é fácil verificar a atividade de um inquilino específico.)	Esforço moderado. (Como todos os inquilinos compartilham o mesmo recurso físico, você precisa aplicar uma filtragem adicional para verificar a atividade de um inquilino específico.)	Esforço moderado. (Como todos os inquilinos compartilham o mesmo recurso físico, você precisa aplicar uma filtragem adicional para verificar a atividade de um inquilino específico.)	Esforço significativo. (Como todos os inquilinos compartilham todos os recursos, incluindo tabelas, você precisa usar a captura de variáveis de vinculação para verificar a qual inquilino uma consulta SQL específica pertence.)
Gerenciamento centralizado e monitoramento de saúde/atividades	Esforço significativo (para configurar o monitoramento central e um centro de comando central).	Esforço moderado (porque todos os inquilinos compartilham a mesma instância).	Esforço moderado (porque todos os inquilinos compartilham a mesma instância).	Esforço mínimo (porque todos os inquilinos compartilham os mesmos recursos, incluindo o esquema).

	Silo	Bridge com bancos de dados separados	Ponte com esquemas separados	Piscina
Chances de envolver o identificador de objeto (OID) e o ID da transação (XID)	Mínimo.	Alto. (Como o OID, o XID é um único contador de cluster do PostgreSQL e pode haver problemas de limpeza eficaz em bancos de dados físicos).	Moderado. (Como o OID, o XID é um único contador em todo o cluster do PostgreSQL).	Alto. (Por exemplo, uma única tabela pode atingir o limite do TOAST OID de 4 bilhões, dependendo do número de out-of-line colunas.)

Recomendações de segurança no nível de linha

A segurança em nível de linha (RLS) é necessária para manter o isolamento dos dados do locatário em um modelo agrupado com o PostgreSQL. O RLS centraliza a aplicação de políticas de isolamento no nível do banco de dados e elimina a carga de manter esse isolamento dos desenvolvedores de software. A forma mais comum de implementar o RLS é habilitar esse recurso no PostgreSQL DBMS. O RLS envolve a filtragem do acesso às linhas de dados com base em um valor em uma coluna especificada. Você pode usar dois métodos para filtrar o acesso aos dados:

- Uma coluna de dados especificada em uma tabela é comparada ao valor do usuário atual do PostgreSQL. Os valores na coluna que são equivalentes ao usuário conectado do PostgreSQL podem ser acessados por esse usuário.
- Uma coluna de dados especificada em uma tabela é comparada ao valor de uma variável de tempo de execução definida pelo aplicativo. Os valores na coluna que são equivalentes à variável de tempo de execução são acessíveis durante essa sessão.

A segunda opção é a preferida, pois a primeira opção requer a criação de um novo usuário do PostgreSQL para cada locatário. Em vez disso, um aplicativo SaaS que usa PostgreSQL deve ser responsável por definir um contexto específico do inquilino em tempo de execução ao consultar o PostgreSQL. Isso terá o efeito de aplicar o RLS. Você também pode ativar o RLS com table-by-table base nisso. Como prática recomendada, você deve habilitar o RLS em todas as tabelas que contêm dados de inquilinos.

O exemplo a seguir cria duas tabelas e ativa o RLS. Este exemplo compara uma coluna de dados com o valor da variável de tempo de execuçãoapp.current_tenant.

```
-- Create a table for our tenants with indexes on the primary key and the tenant’s name
CREATE TABLE tenant (
    tenant_id UUID DEFAULT uuid_generate_v4() PRIMARY KEY,
    name VARCHAR(255) UNIQUE,
    status VARCHAR(64) CHECK (status IN ('active', 'suspended', 'disabled')),
    tier VARCHAR(64) CHECK (tier IN ('gold', 'silver', 'bronze'))
);

-- Create a table for users of a tenant
CREATE TABLE tenant_user (
    user_id UUID DEFAULT uuid_generate_v4() PRIMARY KEY,
    tenant_id UUID NOT NULL REFERENCES tenant (tenant_id) ON DELETE RESTRICT,
```

```
    email VARCHAR(255) NOT NULL UNIQUE,
    given_name VARCHAR(255) NOT NULL CHECK (given_name <> ''),
    family_name VARCHAR(255) NOT NULL CHECK (family_name <> '')
);

-- Turn on RLS
ALTER TABLE tenant ENABLE ROW LEVEL SECURITY;

-- Restrict read and write actions so tenants can only see their rows
-- Cast the UUID value in tenant_id to match the type current_setting
-- This policy implies a WITH CHECK that matches the USING clause
CREATE POLICY tenant_isolation_policy ON tenant
USING (tenant_id = current_setting('app.current_tenant')::UUID);

-- And do the same for the tenant users
ALTER TABLE tenant_user ENABLE ROW LEVEL SECURITY;

CREATE POLICY tenant_user_isolation_policy ON tenant_user
USING (tenant_id = current_setting('app.current_tenant')::UUID);
```

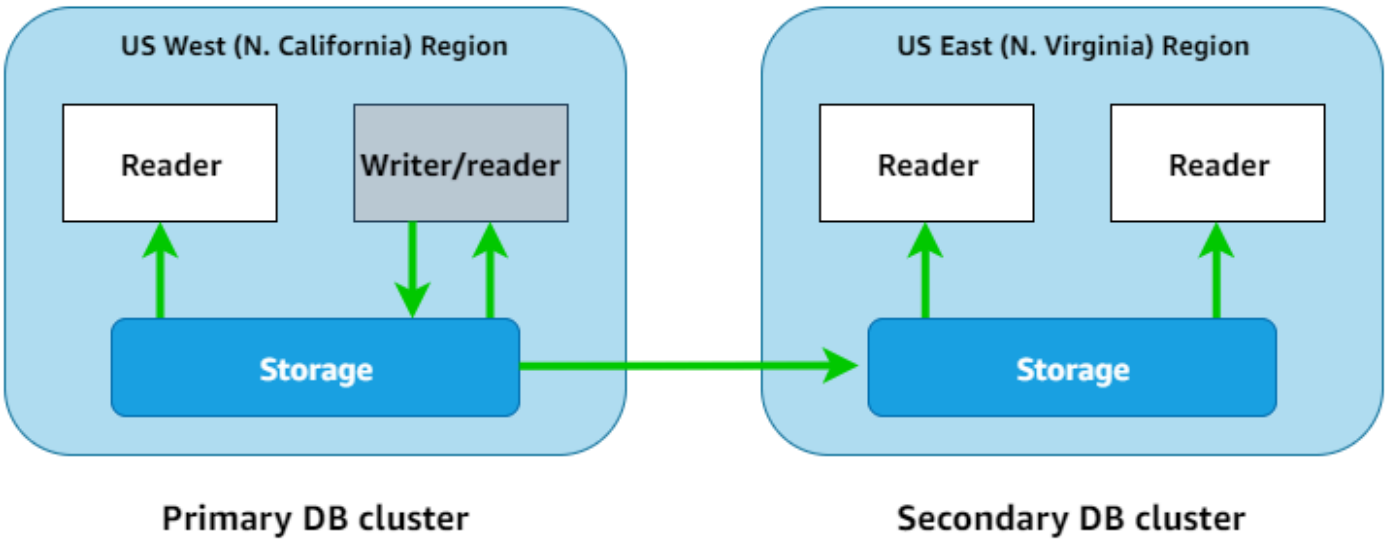
Para obter mais informações, consulte a postagem de blog Isolamento de dados de vários inquilinos com o PostgreSQL Row Level Security. A equipe daAWS SaaS Factory também tem alguns exemplos GitHub para auxiliar na implementação do RLS.

Disponibilidade do PostgreSQL para o modelo de pool

Os modelos de pool, por sua natureza, têm apenas uma única instância do PostgreSQL. Portanto, projetar seu aplicativo para alta disponibilidade é crucial. Uma falha ou paralisação de um banco de dados agrupado faz com que seu aplicativo seja degradado ou fique inacessível para todos os seus inquilinos.

As instâncias de banco de dados do Amazon RDS for PostgreSQL podem se tornar redundantes em duas zonas de disponibilidade ao ativar o recurso de alta disponibilidade. Para obter mais informações, consulte Alta disponibilidade (Multi-AZ) para o Amazon RDS na documentação do Amazon RDS. Para o failover entre regiões, você pode criar uma réplica de leitura em uma região diferente. AWS (Essa réplica de leitura precisa ser promovida como parte de um processo de failover.) Além disso, você pode replicar backups replicados em todas AWS as regiões para recuperação. Para obter mais informações, consulte Replicação de backups automatizados para outra AWS região na documentação do Amazon RDS.

Compatível com o Aurora PostgreSQL, faz backup automático dos dados de uma forma que possa sustentar a falha de várias zonas de disponibilidade. (Consulte Alta disponibilidade do Amazon Aurora na documentação do Aurora.) Para tornar o Aurora mais resiliente e se recuperar mais rapidamente, você pode criar réplicas de leitura do Aurora em outras zonas de disponibilidade. Você pode usar os bancos de dados globais do Aurora para replicar dados em cinco AWS regiões adicionais para recuperação entre regiões e failover automático. (Consulte Uso de bancos de dados globais do Amazon Aurora na documentação do Aurora.) Além disso, você pode habilitar o encaminhamento de gravação com os bancos de dados globais do Aurora para obter alta disponibilidade em vários. Regiões da AWS



Independentemente de você estar usando o Amazon RDS for PostgreSQL ou compatível com o Aurora PostgreSQL, recomendamos que você implemente recursos de alta disponibilidade para mitigar o impacto de qualquer interrupção em todos os aplicativos SaaS multilocatários que usam um modelo de pool.

Práticas recomendadas

Esta seção lista algumas das conclusões de alto nível deste guia. Para discussões detalhadas sobre cada ponto, siga os links para as seções correspondentes.

CompareAWS as opções do PostgreSQL gerenciado

AWSoferece duas formas principais de executar o PostgreSQL em um ambiente gerenciado. (Nesse contexto, gerenciado significa que a infraestrutura do PostgreSQL e o DBMS são parcialmente ou totalmente suportados por umAWS serviço.) As opções gerenciadas do PostgreSQLAWS têm o benefício de automatizar backups, failover, otimização e alguma administração do PostgreSQL. Como opções gerenciadas,AWS oferece o Amazon Aurora Edição compatível com PostgreSQL e o Amazon Relational Database Service (Amazon RDS) para PostgreSQL do Amazon Aurora Relational Database Service (Amazon RDS). Você pode selecionar a melhor opção entre esses dois modelos analisando seu caso de uso do PostgreSQL. Para obter mais informações, consulte a seção Como escolher entre o Amazon RDS e o Aurora deste guia.

Selecione um modelo de particionamento SaaS multilocatário

Você pode escolher entre três modelos de particionamento SaaS que são aplicáveis ao PostgreSQL: silo, bridge e pool. Cada modelo tem vantagens e desvantagens, e você deve escolher o modelo ideal, dependendo do seu caso de uso. O Amazon RDS for PostgreSQL e o Aurora PostgreSQL são compatíveis com os três modelos. A escolha de um modelo é fundamental para manter o isolamento dos dados do locatário em seus aplicativos SaaS. Para uma discussão detalhada desses modelos, consulte a seção Modelos de particionamento SaaS multilocatário para PostgreSQL neste guia.

Use segurança em nível de linha para modelos de particionamento SaaS de pool

A segurança em nível de linha (RLS) é necessária para manter o isolamento dos dados do locatário em um modelo de pool com o PostgreSQL. Isso ocorre porque não há separação lógica entre infraestrutura, bancos de dados PostgreSQL ou esquemas por inquilino em um modelo de pool. O RLS centraliza a aplicação de políticas de isolamento no nível do banco de dados e elimina a carga de manter esse isolamento dos desenvolvedores de software. Você pode usar o RLS para limitar

as operações do banco de dados a um inquilino específico. Para mais informações e um exemplo, consulte a seção Recomendações de segurança em nível de linha deste guia.

Perguntas frequentes

Esta seção fornece respostas às perguntas mais frequentes sobre a implementação do PostgreSQL gerenciado em aplicativos SaaS multilocatários.

Quais opções gerenciadas do PostgreSQLAWS oferecem?

AWSoferece o Amazon Aurora compatível com PostgreSQL e o Amazon Relational Database Service (Amazon RDS) para PostgreSQL. AWStambém tem um amplo catálogo de ofertas de bancos de dados gerenciados.

Qual serviço é ideal para aplicativos SaaS?

Você pode usar aplicativos compatíveis com Aurora PostgreSQL e Amazon RDS for PostgreSQL para aplicativos SaaS e todos os modelos de particionamento SaaS discutidos neste guia. Esses dois serviços têm diferenças em escalabilidade, recuperação de falhas, failover, opções de armazenamento, alta disponibilidade, recuperação de desastres, backup e as classes de instância disponíveis para cada opção. A escolha ideal dependerá do seu caso de uso específico. Use a matriz de decisão neste guia para escolher a melhor opção para seu caso de uso.

Quais requisitos exclusivos devo considerar se eu decidir usar um banco de dados PostgreSQL com um aplicativo SaaS multilocatário?

Como acontece com qualquer armazenamento de dados usado com um aplicativo SaaS, a consideração mais importante é o método para manter o isolamento dos dados do locatário. Conforme discutido neste guia, há várias maneiras de obter isolamento de dados de locatários com ofertasAWS gerenciadas do PostgreSQL. Além disso, você deve considerar o isolamento de desempenho por inquilino para qualquer implementação do PostgreSQL.

Quais modelos posso usar para manter o isolamento dos dados do locatário com o PostgreSQL?

Você pode usar os modelos de silo, ponte e pool como estratégias de particionamento SaaS para manter o isolamento dos dados dos inquilinos. Para uma discussão sobre esses modelos e como

eles podem ser aplicados ao PostgreSQL, consulte a seção Modelos de particionamento SaaS multilocatário para PostgreSQL neste guia.

Como faço para manter o isolamento dos dados do inquilino com um único banco de dados PostgreSQL que é compartilhado entre vários locatários?

O PostgreSQL oferece suporte a um recurso de segurança em nível de linha (RLS) que você pode usar para impor o isolamento de dados do locatário em um único banco de dados ou instância PostgreSQL. Além disso, você pode provisionar bancos de dados PostgreSQL separados por locatário em uma única instância ou criar esquemas por inquilino para atingir essa meta. Para ver as vantagens e desvantagens dessas abordagens, consulte a seção Recomendações de segurança em nível de linha neste guia.

Next steps (Próximas etapas)

AWSoferece duas opções para operar o PostgreSQL gerenciado: compatível com Aurora PostgreSQL e Amazon RDS for PostgreSQL. Recomendamos que você avalie os dois serviços e escolha a opção que melhor suporta seu caso de uso específico para seus aplicativos SaaS multilocatários. A conformidade com um modelo de particionamento SaaS pode garantir que um aplicativo SaaS que usa o PostgreSQL siga estritamente as melhores práticas para manter a locação. Os modelos de particionamento de silo, ponte e pool SaaS oferecem suporte a muitos casos de uso de SaaS. Esses modelos oferecem vantagens variadas entre fatores como isolamento de desempenho, sobrecarga operacional e segurança do inquilino.

Próximas etapas:

- Avalie a compatibilidade com Aurora PostgreSQL e o Amazon RDS for PostgreSQL e escolha a melhor opção para seu aplicativo SaaS.
- Selecione um modelo de particionamento SaaS que atenda aos requisitos de sua aplicação: silo, ponte ou pool.
- Implemente o PostgreSQL de acordo com o modelo de particionamento SaaS selecionado.

Recursos

Referências

- Estratégias de armazenamento SaaS: criando um modelo de armazenamento multilocatário em (whitepaper) AWS AWS
- Recuperação de desastres entre regiões usando o Amazon Aurora Global Database para Amazon Aurora PostgreSQL (publicação no blog) AWS
- Isolamento de dados multilocatários com o PostgreSQL Row Level Security (postagem no blog) AWS
- Trabalhar com Amazon Aurora PostgreSQL (documentação do Aurora)
- PostgreSQL no Amazon RDS (documentação do Amazon RDS)

Parceiros

- Amazon Aurora para parceiros do PostgreSQL
- Parceiros do Amazon RDS for PostgreSQL

Histórico do documento

A tabela a seguir descreve alterações significativas feitas neste guia. Se desejar receber notificações sobre futuras atualizações, inscreva-se em um feed RSS.

Alteração	Descrição	Data
Atualização	Atualizações para refletir a disponibilidade do encaminhamento de gravação no Aurora.	29 de abril de 2024
Atualização	A tabela de comparação do Amazon RDS e do Aurora foi atualizada.	21 de outubro de 2022
—	Publicação inicial	30 de setembro de 2021

AWS Glossário de orientação prescritiva

A seguir estão os termos comumente usados em estratégias, guias e padrões fornecidos pela Orientação AWS Prescritiva. Para sugerir entradas, use o link Fornecer feedback no final do glossário.

Números

7 Rs

Sete estratégias comuns de migração para mover aplicações para a nuvem. Essas estratégias baseiam-se nos 5 Rs identificados pela Gartner em 2011 e consistem em:

- Refatorar/rearquitetar: mova uma aplicação e modifique sua arquitetura aproveitando ao máximo os recursos nativos de nuvem para melhorar a agilidade, a performance e a escalabilidade. Isso normalmente envolve a portabilidade do sistema operacional e do banco de dados. Exemplo: migrar seu banco de dados Oracle on-premises para o Amazon Aurora Edição Compatível com PostgreSQL.
- Redefinir a plataforma (mover e redefinir [mover e redefinir (lift-and-reshape]): mova uma aplicação para a nuvem e introduza algum nível de otimização a fim de aproveitar os recursos da nuvem. Exemplo: migre seu banco de dados Oracle local para o Amazon Relational Database Service (Amazon RDS) para Oracle na nuvem. AWS
- Recomprar (drop and shop): mude para um produto diferente, normalmente migrando de uma licença tradicional para um modelo SaaS. Exemplo: migrar seu sistema de gerenciamento de relacionamento com o cliente (CRM) para o Salesforce.com.
- Redefinir a hospedagem (mover sem alterações [lift-and-shift])mover uma aplicação para a nuvem sem fazer nenhuma alteração a fim de aproveitar os recursos da nuvem. Exemplo: migre seu banco de dados Oracle local para o Oracle em uma instância do EC2 na nuvem. AWS
- Realocar (mover o hipervisor sem alterações [hypervisor-level lift-and-shift]): mover a infraestrutura para a nuvem sem comprar novo hardware, reescrever aplicações ou modificar suas operações existentes. Esse cenário de migração é específico do VMware Cloud on AWS, que oferece suporte à compatibilidade de máquinas virtuais (VM) e à portabilidade da carga de trabalho entre seu ambiente local e. AWSÉ possível usar as tecnologias VMware Cloud Foundation de seus datacenters on-premises ao migrar sua infraestrutura para o VMware

Cloud na AWS. Exemplo: realocar o hipervisor que hospeda seu banco de dados Oracle para o VMware Cloud on. AWS

- Reter (revisitar): mantenha as aplicações em seu ambiente de origem. Isso pode incluir aplicações que exigem grande refatoração, e você deseja adiar esse trabalho para um momento posterior, e aplicações antigas que você deseja manter porque não há justificativa comercial para migrá-las.
- Retirar: desative ou remova aplicações que não são mais necessárias em seu ambiente de origem.

A

ABAC

Consulte controle de acesso baseado em atributos.

serviços abstratos

Veja os serviços gerenciados.

ACID

Veja atomicidade, consistência, isolamento, durabilidade.

migração ativa-ativa

Um método de migração de banco de dados no qual os bancos de dados de origem e de destino são mantidos em sincronia (por meio de uma ferramenta de replicação bidirecional ou operações de gravação dupla), e ambos os bancos de dados lidam com transações de aplicações conectadas durante a migração. Esse método oferece suporte à migração em lotes pequenos e controlados, em vez de exigir uma substituição única. É mais flexível, mas exige mais trabalho do que a migração ativa-passiva.

migração ativa-passiva

Um método de migração de banco de dados no qual os bancos de dados de origem e de destino são mantidos em sincronia, mas somente o banco de dados de origem manipula as transações das aplicações conectadas enquanto os dados são replicados no banco de dados de destino. O banco de dados de destino não aceita nenhuma transação durante a migração.

função agregada

Uma função SQL que opera em um grupo de linhas e calcula um único valor de retorno para o grupo. Exemplos de funções agregadas incluem `SUM` e. `MAX`

AI

Veja inteligência artificial.

AIOps

Veja as operações de inteligência artificial.

anonimização

O processo de excluir permanentemente informações pessoais em um conjunto de dados. A anonimização pode ajudar a proteger a privacidade pessoal. Dados anônimos não são mais considerados dados pessoais.

antipadrões

Uma solução frequentemente usada para um problema recorrente em que a solução é contraproducente, ineficaz ou menos eficaz do que uma alternativa.

controle de aplicativos

Uma abordagem de segurança que permite o uso somente de aplicativos aprovados para ajudar a proteger um sistema contra malware.

portfólio de aplicações

Uma coleção de informações detalhadas sobre cada aplicação usada por uma organização, incluindo o custo para criar e manter a aplicação e seu valor comercial. Essas informações são fundamentais para o processo de descoberta e análise de portfólio e ajudam a identificar e priorizar as aplicações a serem migradas, modernizadas e otimizadas.

inteligência artificial (IA)

O campo da ciência da computação que se dedica ao uso de tecnologias de computação para desempenhar funções cognitivas normalmente associadas aos humanos, como aprender, resolver problemas e reconhecer padrões. Para obter mais informações, consulte O que é inteligência artificial?

operações de inteligência artificial (AIOps)

O processo de usar técnicas de machine learning para resolver problemas operacionais, reduzir incidentes operacionais e intervenção humana e aumentar a qualidade do serviço. Para obter

mais informações sobre como as AIOps são usadas na estratégia de migração para a AWS , consulte o guia de integração de operações.

criptografia assimétrica

Um algoritmo de criptografia que usa um par de chaves, uma chave pública para criptografia e uma chave privada para descriptografia. É possível compartilhar a chave pública porque ela não é usada na descriptografia, mas o acesso à chave privada deve ser altamente restrito.

atomicidade, consistência, isolamento, durabilidade (ACID)

Um conjunto de propriedades de software que garantem a validade dos dados e a confiabilidade operacional de um banco de dados, mesmo no caso de erros, falhas de energia ou outros problemas.

controle de acesso por atributo (ABAC)

A prática de criar permissões minuciosas com base nos atributos do usuário, como departamento, cargo e nome da equipe. Para obter mais informações, consulte ABAC AWS na documentação AWS Identity and Access Management (IAM).

fonte de dados autorizada

Um local onde você armazena a versão principal dos dados, que é considerada a fonte de informações mais confiável. Você pode copiar dados da fonte de dados autorizada para outros locais com o objetivo de processar ou modificar os dados, como anonimizá-los, redigi-los ou pseudonimizá-los.

Availability Zone (zona de disponibilidade)

Um local distinto dentro de um Região da AWS que está isolado de falhas em outras zonas de disponibilidade e fornece conectividade de rede barata e de baixa latência a outras zonas de disponibilidade na mesma região.

AWS Estrutura de adoção da nuvem (AWS CAF)

Uma estrutura de diretrizes e melhores práticas AWS para ajudar as organizações a desenvolver um plano eficiente e eficaz para migrar com sucesso para a nuvem. AWS O CAF organiza a orientação em seis áreas de foco chamadas perspectivas: negócios, pessoas, governança, plataforma, segurança e operações. As perspectivas de negócios, pessoas e governança têm como foco habilidades e processos de negócios; as perspectivas de plataforma, segurança e operações concentram-se em habilidades e processos técnicos. Por exemplo, a perspectiva das pessoas tem como alvo as partes interessadas que lidam com recursos humanos (RH), funções de pessoal e gerenciamento de pessoal. Nessa perspectiva, o AWS CAF fornece orientação para

desenvolvimento, treinamento e comunicação de pessoas para ajudar a preparar a organização para a adoção bem-sucedida da nuvem. Para obter mais informações, consulte o site da AWS CAF e o whitepaper da AWS CAF.

AWS Estrutura de qualificação da carga de trabalho (AWS WQF)

Uma ferramenta que avalia as cargas de trabalho de migração do banco de dados, recomenda estratégias de migração e fornece estimativas de trabalho. AWS O WQF está incluído com AWS Schema Conversion Tool ()AWS SCT. Ela analisa esquemas de banco de dados e objetos de código, código de aplicações, dependências e características de performance, além de fornecer relatórios de avaliação.

B

bot ruim

Um bot destinado a perturbar ou causar danos a indivíduos ou organizações.

BCP

Veja o planejamento de continuidade de negócios.

gráfico de comportamento

Uma visualização unificada e interativa do comportamento e das interações de recursos ao longo do tempo. É possível usar um gráfico de comportamento com o Amazon Detective para examinar tentativas de login malsucedidas, chamadas de API suspeitas e ações similares. Para obter mais informações, consulte Dados em um gráfico de comportamento na documentação do Detective.

sistema big-endian

Um sistema que armazena o byte mais significativo antes. Veja também endianness.

classificação binária

Um processo que prevê um resultado binário (uma de duas classes possíveis). Por exemplo, seu modelo de ML pode precisar prever problemas como “Este e-mail é ou não é spam?” ou “Este produto é um livro ou um carro?”

filtro de bloom

Uma estrutura de dados probabilística e eficiente em termos de memória que é usada para testar se um elemento é membro de um conjunto.

blue/green deployment (implantação azul/verde)

Uma estratégia de implantação em que você cria dois ambientes separados, mas idênticos. Você executa a versão atual do aplicativo em um ambiente (azul) e a nova versão do aplicativo no outro ambiente (verde). Essa estratégia ajuda você a reverter rapidamente com o mínimo de impacto.

bot

Um aplicativo de software que executa tarefas automatizadas pela Internet e simula a atividade ou interação humana. Alguns bots são úteis ou benéficos, como rastreadores da Web que indexam informações na Internet. Alguns outros bots, conhecidos como bots ruins, têm como objetivo perturbar ou causar danos a indivíduos ou organizações.

botnet

Redes de bots infectadas por malware e sob o controle de uma única parte, conhecidas como pastor de bots ou operador de bots. As redes de bots são o mecanismo mais conhecido para escalar bots e seu impacto.

ramo

Uma área contida de um repositório de código. A primeira ramificação criada em um repositório é a ramificação principal. Você pode criar uma nova ramificação a partir de uma ramificação existente e, em seguida, desenvolver recursos ou corrigir bugs na nova ramificação. Uma ramificação que você cria para gerar um recurso é comumente chamada de ramificação de recurso. Quando o recurso estiver pronto para lançamento, você mesclará a ramificação do recurso de volta com a ramificação principal. Para obter mais informações, consulte Sobre filiais (GitHub documentação).

acesso em vidro quebrado

Em circunstâncias excepcionais e por meio de um processo aprovado, um meio rápido para um usuário obter acesso a um Conta da AWS que ele normalmente não tem permissão para acessar. Para obter mais informações, consulte o indicador Implementar procedimentos de quebra de vidro na orientação do Well-Architected AWS .

estratégia brownfield

A infraestrutura existente em seu ambiente. Ao adotar uma estratégia brownfield para uma arquitetura de sistema, você desenvolve a arquitetura de acordo com as restrições dos sistemas e da infraestrutura atuais. Se estiver expandindo a infraestrutura existente, poderá combinar as estratégias brownfield e greenfield.

cache do buffer

A área da memória em que os dados acessados com mais frequência são armazenados.

capacidade de negócios

O que uma empresa faz para gerar valor (por exemplo, vendas, atendimento ao cliente ou marketing). As arquiteturas de microsserviços e as decisões de desenvolvimento podem ser orientadas por recursos de negócios. Para obter mais informações, consulte a seção Organizados de acordo com as capacidades de negócios do whitepaper Executar microsserviços conteinerizados na AWS.

planejamento de continuidade de negócios (BCP)

Um plano que aborda o impacto potencial de um evento disruptivo, como uma migração em grande escala, nas operações e permite que uma empresa retome as operações rapidamente.

C

CAF

Consulte Estrutura de adoção da AWS nuvem.

implantação canária

O lançamento lento e incremental de uma versão para usuários finais. Quando estiver confiante, você implanta a nova versão e substituirá a versão atual em sua totalidade.

CCoE

Veja o Centro de Excelência em Nuvem.

CDC

Veja a captura de dados de alterações.

captura de dados de alterações (CDC)

O processo de rastrear alterações em uma fonte de dados, como uma tabela de banco de dados, e registrar metadados sobre a alteração. É possível usar o CDC para várias finalidades, como auditar ou replicar alterações em um sistema de destino para manter a sincronização.

engenharia do caos

Introduzir intencionalmente falhas ou eventos disruptivos para testar a resiliência de um sistema. Você pode usar AWS Fault Injection Service (AWS FIS) para realizar experimentos que estressam suas AWS cargas de trabalho e avaliar sua resposta.

CI/CD

Veja a integração e a entrega contínuas.

classificação

Um processo de categorização que ajuda a gerar previsões. Os modelos de ML para problemas de classificação predizem um valor discreto. Os valores discretos são sempre diferentes uns dos outros. Por exemplo, um modelo pode precisar avaliar se há ou não um carro em uma imagem.

criptografia no lado do cliente

Criptografia de dados localmente, antes que o alvo os AWS service (Serviço da AWS) receba.

Centro de Excelência da Nuvem (CCoE)

Uma equipe multidisciplinar que impulsiona os esforços de adoção da nuvem em toda a organização, incluindo o desenvolvimento de práticas recomendas de nuvem, a mobilização de recursos, o estabelecimento de cronogramas de migração e a liderança da organização em transformações em grande escala. Para obter mais informações, consulte as postagens do CCoE no blog AWS Cloud Enterprise Strategy.

computação em nuvem

A tecnologia de nuvem normalmente usada para armazenamento de dados remoto e gerenciamento de dispositivos de IoT. A computação em nuvem geralmente está conectada à tecnologia de computação de ponta.

modelo operacional em nuvem

Em uma organização de TI, o modelo operacional usado para criar, amadurecer e otimizar um ou mais ambientes de nuvem. Para obter mais informações, consulte Criar seu modelo operacional de nuvem.

estágios de adoção da nuvem

As quatro fases pelas quais as organizações normalmente passam quando migram para a AWS nuvem:

- Projeto: executar alguns projetos relacionados à nuvem para fins de prova de conceito e aprendizado
- Fundação: realizar investimentos fundamentais para escalar sua adoção da nuvem (por exemplo, criar uma zona de pouso, definir um CCoE, estabelecer um modelo de operações)
- Migração: migrar aplicações individuais
- Reinvenção: otimizar produtos e serviços e inovar na nuvem

Esses estágios foram definidos por Stephen Orban na postagem do blog The Journey Toward Cloud-First & the Stages of Adoption no blog AWS Cloud Enterprise Strategy. Para obter informações sobre como eles se relacionam com a estratégia de AWS migração, consulte o guia de preparação para migração.

CMDB

Consulte o banco de dados de gerenciamento de configuração.

repositório de código

Um local onde o código-fonte e outros ativos, como documentação, amostras e scripts, são armazenados e atualizados por meio de processos de controle de versão. Os repositórios de nuvem comuns incluem GitHub ou AWS CodeCommit. Cada versão do código é chamada de ramificação. Em uma estrutura de microsserviços, cada repositório é dedicado a uma única peça de funcionalidade. Um único pipeline de CI/CD pode usar vários repositórios.

cache frio

Um cache de buffer que está vazio, não está bem preenchido ou contém dados obsoletos ou irrelevantes. Isso afeta a performance porque a instância do banco de dados deve ler da memória principal ou do disco, um processo que é mais lento do que a leitura do cache do buffer.

dados frios

Dados que raramente são acessados e geralmente são históricos. Ao consultar esse tipo de dados, consultas lentas geralmente são aceitáveis. Mover esses dados para níveis ou classes de armazenamento de baixo desempenho e menos caros pode reduzir os custos.

visão computacional (CV)

Um campo da IA que usa aprendizado de máquina para analisar e extrair informações de formatos visuais, como imagens e vídeos digitais. Por exemplo, AWS Panorama oferece dispositivos que adicionam CV às redes de câmeras locais, e a Amazon SageMaker fornece algoritmos de processamento de imagem para CV.

desvio de configuração

Para uma carga de trabalho, uma alteração de configuração em relação ao estado esperado. Isso pode fazer com que a carga de trabalho se torne incompatível e, normalmente, é gradual e não intencional.

banco de dados de gerenciamento de configuração (CMDB)

Um repositório que armazena e gerencia informações sobre um banco de dados e seu ambiente de TI, incluindo componentes de hardware e software e suas configurações. Normalmente, os dados de um CMDB são usados no estágio de descoberta e análise do portfólio da migração.

pacote de conformidade

Um conjunto de AWS Config regras e ações de remediação que você pode montar para personalizar suas verificações de conformidade e segurança. Você pode implantar um pacote de conformidade como uma entidade única em uma Conta da AWS região ou em uma organização usando um modelo YAML. Para obter mais informações, consulte Pacotes de conformidade na documentação. AWS Config

integração contínua e entrega contínua (CI/CD)

O processo de automatizar os estágios de origem, criação, teste, preparação e produção do processo de lançamento do software. O CI/CD é comumente descrito como um pipeline. O CI/CD pode ajudar você a automatizar processos, melhorar a produtividade, melhorar a qualidade do código e entregar com mais rapidez. Para obter mais informações, consulte Benefícios da entrega contínua. CD também pode significar implantação contínua. Para obter mais informações, consulte Entrega contínua versus implantação contínua.

CV

Veja visão computacional.

D

dados em repouso

Dados estacionários em sua rede, por exemplo, dados que estão em um armazenamento.

classificação de dados

Um processo para identificar e categorizar os dados em sua rede com base em criticalidade e confidencialidade. É um componente crítico de qualquer estratégia de gerenciamento de riscos de

segurança cibernética, pois ajuda a determinar os controles adequados de proteção e retenção para os dados. A classificação de dados é um componente do pilar de segurança no AWS Well-Architected Framework. Para obter mais informações, consulte Classificação de dados.

desvio de dados

Uma variação significativa entre os dados de produção e os dados usados para treinar um modelo de ML ou uma alteração significativa nos dados de entrada ao longo do tempo. O desvio de dados pode reduzir a qualidade geral, a precisão e a imparcialidade das previsões do modelo de ML.

dados em trânsito

Dados que estão se movendo ativamente pela sua rede, como entre os recursos da rede.

malha de dados

Uma estrutura arquitetônica que fornece propriedade de dados distribuída e descentralizada com gerenciamento e governança centralizados.

minimização de dados

O princípio de coletar e processar apenas os dados estritamente necessários. Praticar a minimização de dados no Nuvem AWS pode reduzir os riscos de privacidade, os custos e a pegada de carbono de sua análise.

perímetro de dados

Um conjunto de proteções preventivas em seu AWS ambiente que ajudam a garantir que somente identidades confiáveis acessem recursos confiáveis das redes esperadas. Para obter mais informações, consulte Construindo um perímetro de dados em. AWS

pré-processamento de dados

A transformação de dados brutos em um formato que seja facilmente analisado por seu modelo de ML. O pré-processamento de dados pode significar a remoção de determinadas colunas ou linhas e o tratamento de valores ausentes, inconsistentes ou duplicados.

proveniência dos dados

O processo de rastrear a origem e o histórico dos dados ao longo de seu ciclo de vida, por exemplo, como os dados foram gerados, transmitidos e armazenados.

titular dos dados

Um indivíduo cujos dados estão sendo coletados e processados.

data warehouse

Um sistema de gerenciamento de dados que oferece suporte à inteligência comercial, como análises. Os data warehouses geralmente contêm grandes quantidades de dados históricos e geralmente são usados para consultas e análises.

linguagem de definição de dados (DDL)

Instruções ou comandos para criar ou modificar a estrutura de tabelas e objetos em um banco de dados.

linguagem de manipulação de dados (DML)

Instruções ou comandos para modificar (inserir, atualizar e excluir) informações em um banco de dados.

DDL

Consulte a linguagem de definição de banco de dados.

deep ensemble

A combinação de vários modelos de aprendizado profundo para gerar previsões. Os deep ensembles podem ser usados para produzir uma previsão mais precisa ou para estimar a incerteza nas previsões.

Aprendizado profundo

Um subcampo do ML que usa várias camadas de redes neurais artificiais para identificar o mapeamento entre os dados de entrada e as variáveis-alvo de interesse.

defense-in-depth

Uma abordagem de segurança da informação na qual uma série de mecanismos e controles de segurança são cuidadosamente distribuídos por toda a rede de computadores para proteger a confidencialidade, a integridade e a disponibilidade da rede e dos dados nela contidos. Ao adotar essa estratégia AWS, você adiciona vários controles em diferentes camadas da AWS Organizations estrutura para ajudar a proteger os recursos. Por exemplo, uma defense-in-depth abordagem pode combinar autenticação multifatorial, segmentação de rede e criptografia.

administrador delegado

Em AWS Organizations, um serviço compatível pode registrar uma conta de AWS membro para administrar as contas da organização e gerenciar as permissões desse serviço. Essa conta

é chamada de administrador delegado para esse serviço Para obter mais informações e uma lista de serviços compatíveis, consulte Serviços que funcionam com o AWS Organizations na documentação do AWS Organizations .

implantação

O processo de criar uma aplicação, novos recursos ou correções de código disponíveis no ambiente de destino. A implantação envolve a implementação de mudanças em uma base de código e, em seguida, a criação e execução dessa base de código nos ambientes da aplicação

ambiente de desenvolvimento

Veja o ambiente.

controle detectivo

Um controle de segurança projetado para detectar, registrar e alertar após a ocorrência de um evento. Esses controles são uma segunda linha de defesa, alertando você sobre eventos de segurança que contornaram os controles preventivos em vigor. Para obter mais informações, consulte Controles detectivos em Como implementar controles de segurança na AWS.

mapeamento do fluxo de valor de desenvolvimento (DVSM)

Um processo usado para identificar e priorizar restrições que afetam negativamente a velocidade e a qualidade em um ciclo de vida de desenvolvimento de software. O DVSM estende o processo de mapeamento do fluxo de valor originalmente projetado para práticas de manufatura enxuta. Ele se concentra nas etapas e equipes necessárias para criar e movimentar valor por meio do processo de desenvolvimento de software.

gêmeo digital

Uma representação virtual de um sistema real, como um prédio, fábrica, equipamento industrial ou linha de produção. Os gêmeos digitais oferecem suporte à manutenção preditiva, ao monitoramento remoto e à otimização da produção.

tabela de dimensões

Em um esquema em estrela, uma tabela menor que contém atributos de dados sobre dados quantitativos em uma tabela de fatos. Os atributos da tabela de dimensões geralmente são campos de texto ou números discretos que se comportam como texto. Esses atributos são comumente usados para restringir consultas, filtrar e rotular conjuntos de resultados.

desastre

Um evento que impede que uma workload ou sistema cumpra seus objetivos de negócios em seu local principal de implantação. Esses eventos podem ser desastres naturais, falhas técnicas ou o resultado de ações humanas, como configuração incorreta não intencional ou ataque de malware.

recuperação de desastres (DR)

A estratégia e o processo que você usa para minimizar o tempo de inatividade e a perda de dados causados por um desastre. Para obter mais informações, consulte Recuperação de desastres de cargas de trabalho em AWS: Recuperação na nuvem no AWS Well-Architected Framework.

DML

Consulte linguagem de manipulação de banco de dados.

design orientado por domínio

Uma abordagem ao desenvolvimento de um sistema de software complexo conectando seus componentes aos domínios em evolução, ou principais metas de negócios, atendidos por cada componente. Esse conceito foi introduzido por Eric Evans em seu livro, Design orientado por domínio: lidando com a complexidade no coração do software (Boston: Addison-Wesley Professional, 2003). Para obter informações sobre como usar o design orientado por domínio com o padrão strangler fig, consulte Modernizar incrementalmente os serviços web herdados do Microsoft ASP.NET (ASMX) usando contêineres e o Amazon API Gateway.

DR

Veja a recuperação de desastres.

detecção de deriva

Rastreando desvios de uma configuração básica. Por exemplo, você pode usar AWS CloudFormation para detectar desvios nos recursos do sistema ou AWS Control Tower para detectar mudanças em seu landing zone que possam afetar a conformidade com os requisitos de governança.

DVSM

Veja o mapeamento do fluxo de valor do desenvolvimento.

E

EDA

Veja a análise exploratória de dados.

computação de borda

A tecnologia que aumenta o poder computacional de dispositivos inteligentes nas bordas de uma rede de IoT. Quando comparada à computação em nuvem, a computação de ponta pode reduzir a latência da comunicação e melhorar o tempo de resposta.

Criptografia

Um processo de computação que transforma dados de texto simples, legíveis por humanos, em texto cifrado.

chave de criptografia

Uma sequência criptográfica de bits aleatórios que é gerada por um algoritmo de criptografia. As chaves podem variar em tamanho, e cada chave foi projetada para ser imprevisível e exclusiva.

endianismo

A ordem na qual os bytes são armazenados na memória do computador. Os sistemas big-endian armazenam o byte mais significativo antes. Os sistemas little-endian armazenam o byte menos significativo antes.

endpoint

Veja o endpoint do serviço.

serviço de endpoint

Um serviço que pode ser hospedado em uma nuvem privada virtual (VPC) para ser compartilhado com outros usuários. Você pode criar um serviço de endpoint com AWS PrivateLink e conceder permissões a outros diretores Contas da AWS ou a AWS Identity and Access Management (IAM). Essas contas ou entidades principais podem se conectar ao serviço de endpoint de maneira privada criando endpoints da VPC de interface. Para obter mais informações, consulte Criar um serviço de endpoint na documentação do Amazon Virtual Private Cloud (Amazon VPC).

planejamento de recursos corporativos (ERP)

Um sistema que automatiza e gerencia os principais processos de negócios (como contabilidade, MES e gerenciamento de projetos) para uma empresa.

criptografia envelopada

O processo de criptografar uma chave de criptografia com outra chave de criptografia. Para obter mais informações, consulte Criptografia de envelope na documentação AWS Key Management Service (AWS KMS).

environment (ambiente)

Uma instância de uma aplicação em execução. Estes são tipos comuns de ambientes na computação em nuvem:

- ambiente de desenvolvimento: uma instância de uma aplicação em execução que está disponível somente para a equipe principal responsável pela manutenção da aplicação. Ambientes de desenvolvimento são usados para testar mudanças antes de promovê-las para ambientes superiores. Esse tipo de ambiente às vezes é chamado de ambiente de teste.
- ambientes inferiores: todos os ambientes de desenvolvimento para uma aplicação, como aqueles usados para compilações e testes iniciais.
- ambiente de produção: uma instância de uma aplicação em execução que os usuários finais podem acessar. Em um pipeline de CI/CD, o ambiente de produção é o último ambiente de implantação.
- ambientes superiores: todos os ambientes que podem ser acessados por usuários que não sejam a equipe principal de desenvolvimento. Isso pode incluir um ambiente de produção, ambientes de pré-produção e ambientes para testes de aceitação do usuário.

epic

Em metodologias ágeis, categorias funcionais que ajudam a organizar e priorizar seu trabalho. Os epics fornecem uma descrição de alto nível dos requisitos e das tarefas de implementação. Por exemplo, os épicos de segurança AWS da CAF incluem gerenciamento de identidade e acesso, controles de detetive, segurança de infraestrutura, proteção de dados e resposta a incidentes. Para obter mais informações sobre epics na estratégia de migração da AWS , consulte o guia de implementação do programa.

ERP

Consulte planejamento de recursos corporativos.

análise exploratória de dados (EDA)

O processo de analisar um conjunto de dados para entender suas principais características. Você coleta ou agrega dados e, em seguida, realiza investigações iniciais para encontrar padrões,

detectar anomalias e verificar suposições. O EDA é realizado por meio do cálculo de estatísticas resumidas e da criação de visualizações de dados.

F

tabela de fatos

A tabela central em um esquema em estrela. Ele armazena dados quantitativos sobre operações comerciais. Normalmente, uma tabela de fatos contém dois tipos de colunas: aquelas que contêm medidas e aquelas que contêm uma chave externa para uma tabela de dimensões.

falham rapidamente

Uma filosofia que usa testes frequentes e incrementais para reduzir o ciclo de vida do desenvolvimento. É uma parte essencial de uma abordagem ágil.

limite de isolamento de falhas

No Nuvem AWS, um limite, como uma zona de disponibilidade, Região da AWS um plano de controle ou um plano de dados, que limita o efeito de uma falha e ajuda a melhorar a resiliência das cargas de trabalho. Para obter mais informações, consulte Limites de isolamento de AWS falhas.

ramificação de recursos

Veja a filial.

recursos

Os dados de entrada usados para fazer uma previsão. Por exemplo, em um contexto de manufatura, os recursos podem ser imagens capturadas periodicamente na linha de fabricação.

importância do recurso

O quanto um recurso é importante para as previsões de um modelo. Isso geralmente é expresso como uma pontuação numérica que pode ser calculada por meio de várias técnicas, como Shapley Additive Explanations (SHAP) e gradientes integrados. Para obter mais informações, consulte Interpretabilidade do modelo de aprendizado de máquina com:AWS.

transformação de recursos

O processo de otimizar dados para o processo de ML, incluindo enriquecer dados com fontes adicionais, escalar valores ou extrair vários conjuntos de informações de um único

campo de dados. Isso permite que o modelo de ML se beneficie dos dados. Por exemplo, se a data “2021-05-27 00:15:37” for dividida em “2021”, “maio”, “quinta” e “15”, isso poderá ajudar o algoritmo de aprendizado a aprender padrões diferenciados associados a diferentes componentes de dados.

FGAC

Veja o controle de acesso refinado.

controle de acesso refinado (FGAC)

O uso de várias condições para permitir ou negar uma solicitação de acesso.

migração flash-cut

Um método de migração de banco de dados que usa replicação contínua de dados por meio da captura de dados alterados para migrar dados no menor tempo possível, em vez de usar uma abordagem em fases. O objetivo é reduzir ao mínimo o tempo de inatividade.

G

bloqueio geográfico

Veja as restrições geográficas.

restrições geográficas (bloqueio geográfico)

Na Amazon CloudFront, uma opção para impedir que usuários em países específicos acessem distribuições de conteúdo. É possível usar uma lista de permissões ou uma lista de bloqueios para especificar países aprovados e banidos. Para obter mais informações, consulte Restringir a distribuição geográfica do seu conteúdo na CloudFront documentação.

Fluxo de trabalho do GitFlow

Uma abordagem na qual ambientes inferiores e superiores usam ramificações diferentes em um repositório de código-fonte. O fluxo de trabalho do Gitflow é considerado legado, e o fluxo de trabalho baseado em troncos é a abordagem moderna e preferida.

estratégia greenfield

A ausência de infraestrutura existente em um novo ambiente. Ao adotar uma estratégia greenfield para uma arquitetura de sistema, é possível selecionar todas as novas tecnologias sem a

restrição da compatibilidade com a infraestrutura existente, também conhecida como brownfield. Se estiver expandindo a infraestrutura existente, poderá combinar as estratégias brownfield e greenfield.

barreira de proteção

Uma regra de alto nível que ajuda a gerenciar recursos, políticas e conformidade em todas as unidades organizacionais (UOs). Barreiras de proteção preventivas impõem políticas para garantir o alinhamento a padrões de conformidade. Elas são implementadas usando políticas de controle de serviço e limites de permissões do IAM. Barreiras de proteção detectivas detectam violações de políticas e problemas de conformidade e geram alertas para remediação. Eles são implementados usando AWS Config, AWS Security Hub, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector e verificações personalizadas AWS Lambda .

H

HA

Veja a alta disponibilidade.

migração heterogênea de bancos de dados

Migrar seu banco de dados de origem para um banco de dados de destino que usa um mecanismo de banco de dados diferente (por exemplo, Oracle para Amazon Aurora). A migração heterogênea geralmente faz parte de um esforço de redefinição da arquitetura, e converter o esquema pode ser uma tarefa complexa. O AWS fornece o AWS SCT para ajudar nas conversões de esquemas.

alta disponibilidade (HA)

A capacidade de uma workload operar continuamente, sem intervenção, em caso de desafios ou desastres. Os sistemas AH são projetados para realizar o failover automático, oferecer consistentemente desempenho de alta qualidade e lidar com diferentes cargas e falhas com impacto mínimo no desempenho.

modernização de historiador

Uma abordagem usada para modernizar e atualizar os sistemas de tecnologia operacional (OT) para melhor atender às necessidades do setor de manufatura. Um historiador é um tipo de banco de dados usado para coletar e armazenar dados de várias fontes em uma fábrica.

migração homogênea de bancos de dados

Migrar seu banco de dados de origem para um banco de dados de destino que compartilha o mesmo mecanismo de banco de dados (por exemplo, Microsoft SQL Server para Amazon RDS para SQL Server). A migração homogênea geralmente faz parte de um esforço de redefinição da hospedagem ou da plataforma. É possível usar utilitários de banco de dados nativos para migrar o esquema.

dados quentes

Dados acessados com frequência, como dados em tempo real ou dados translacionais recentes. Esses dados normalmente exigem uma camada ou classe de armazenamento de alto desempenho para fornecer respostas rápidas às consultas.

hotfix

Uma correção urgente para um problema crítico em um ambiente de produção. Devido à sua urgência, um hotfix geralmente é feito fora do fluxo de trabalho normal de DevOps lançamento.

período de hipercuidados

Imediatamente após a substituição, o período em que uma equipe de migração gerencia e monitora as aplicações migradas na nuvem para resolver quaisquer problemas. Normalmente, a duração desse período é de 1 a 4 dias. No final do período de hipercuidados, a equipe de migração normalmente transfere a responsabilidade pelas aplicações para a equipe de operações de nuvem.

I

IaC

Veja a infraestrutura como código.

Política baseada em identidade

Uma política anexada a um ou mais diretores do IAM que define suas permissões no Nuvem AWS ambiente.

aplicação ociosa

Uma aplicação que tem um uso médio de CPU e memória entre 5 e 20% em um período de 90 dias. Em um projeto de migração, é comum retirar essas aplicações ou retê-las on-premises.

IIoT

Veja a Internet das Coisas industrial.

infraestrutura imutável

Um modelo que implanta uma nova infraestrutura para cargas de trabalho de produção em vez de atualizar, corrigir ou modificar a infraestrutura existente. Infraestruturas imutáveis são inerentemente mais consistentes, confiáveis e previsíveis do que infraestruturas mutáveis. Para obter mais informações, consulte as melhores práticas de implantação usando infraestrutura imutável no Well-Architected AWS Framework.

VPC de entrada (admissão)

Em uma arquitetura de AWS várias contas, uma VPC que aceita, inspeciona e roteia conexões de rede de fora de um aplicativo. A Arquitetura de referência de segurança da AWS recomenda configurar sua conta de rede com VPCs de entrada, saída e inspeção para proteger a interface bidirecional entre a aplicação e a Internet em geral.

migração incremental

Uma estratégia de substituição na qual você migra a aplicação em pequenas partes, em vez de realizar uma única substituição completa. Por exemplo, é possível mover inicialmente apenas alguns microsserviços ou usuários para o novo sistema. Depois de verificar se tudo está funcionando corretamente, mova os microsserviços ou usuários adicionais de forma incremental até poder descomissionar seu sistema herdado. Essa estratégia reduz os riscos associados a migrações de grande porte.

Indústria 4.0

Um termo que foi introduzido por Klaus Schwab em 2016 para se referir à modernização dos processos de fabricação por meio de avanços em conectividade, dados em tempo real, automação, análise e IA/ML.

infraestrutura

Todos os recursos e ativos contidos no ambiente de uma aplicação.

Infraestrutura como código (IaC)

O processo de provisionamento e gerenciamento da infraestrutura de uma aplicação por meio de um conjunto de arquivos de configuração. A IaC foi projetada para ajudar você a centralizar

o gerenciamento da infraestrutura, padronizar recursos e escalar rapidamente para que novos ambientes sejam reproduzíveis, confiáveis e consistentes.

Internet das Coisas Industrial (IIoT)

O uso de sensores e dispositivos conectados à Internet nos setores industriais, como manufatura, energia, automotivo, saúde, ciências biológicas e agricultura. Para obter mais informações, consulte Construir uma estratégia de transformação digital para a Internet das Coisas Industrial (IIoT).

VPC de inspeção

Em uma arquitetura de AWS várias contas, uma VPC centralizada que gerencia as inspeções do tráfego de rede entre VPCs (na mesma ou em diferentes Regiões da AWS), a Internet e as redes locais. A Arquitetura de referência de segurança da AWS recomenda configurar sua conta de rede com VPCs de entrada, saída e inspeção para proteger a interface bidirecional entre a aplicação e a Internet em geral.

Internet das Coisas (IoT)

A rede de objetos físicos conectados com sensores ou processadores incorporados que se comunicam com outros dispositivos e sistemas pela Internet ou por uma rede de comunicação local. Para obter mais informações, consulte O que é IoT?

interpretabilidade

Uma característica de um modelo de machine learning que descreve o grau em que um ser humano pode entender como as previsões do modelo dependem de suas entradas. Para obter mais informações, consulte Interpretabilidade do modelo de machine learning com a AWS.

IoT

Consulte Internet das Coisas.

Biblioteca de informações de TI (ITIL)

Um conjunto de práticas recomendadas para fornecer serviços de TI e alinhar esses serviços a requisitos de negócios. A ITIL fornece a base para o ITSM.

Gerenciamento de serviços de TI (ITSM)

Atividades associadas a design, implementação, gerenciamento e suporte de serviços de TI para uma organização. Para obter informações sobre a integração de operações em nuvem com ferramentas de ITSM, consulte o guia de integração de operações.

ITIL

Consulte a biblioteca de informações de TI.

ITSM

Veja o gerenciamento de serviços de TI.

L

controle de acesso baseado em etiqueta (LBAC)

Uma implementação do controle de acesso obrigatório (MAC) em que os usuários e os dados em si recebem explicitamente um valor de etiqueta de segurança. A interseção entre a etiqueta de segurança do usuário e a etiqueta de segurança dos dados determina quais linhas e colunas podem ser vistas pelo usuário.

zona de pouso

Uma landing zone é um AWS ambiente bem arquitetado, com várias contas, escalável e seguro. Um ponto a partir do qual suas organizações podem iniciar e implantar rapidamente workloads e aplicações com confiança em seu ambiente de segurança e infraestrutura. Para obter mais informações sobre zonas de pouso, consulte Configurar um ambiente da AWS com várias contas seguro e escalável.

migração de grande porte

Uma migração de 300 servidores ou mais.

LBAC

Veja controle de acesso baseado em etiquetas.

privilégio mínimo

A prática recomendada de segurança de conceder as permissões mínimas necessárias para executar uma tarefa. Para obter mais informações, consulte Aplicar permissões de privilégios mínimos na documentação do IAM.

mover sem alterações (lift-and-shift)

Veja 7 Rs.

sistema little-endian

Um sistema que armazena o byte menos significativo antes. Veja também endianness.

ambientes inferiores

Veja o ambiente.

M

machine learning (ML)

Um tipo de inteligência artificial que usa algoritmos e técnicas para reconhecimento e aprendizado de padrões. O ML analisa e aprende com dados gravados, por exemplo, dados da Internet das Coisas (IoT), para gerar um modelo estatístico baseado em padrões. Para obter mais informações, consulte Machine learning.

ramificação principal

Veja a filial.

malware

Software projetado para comprometer a segurança ou a privacidade do computador. O malware pode interromper os sistemas do computador, vazar informações confidenciais ou obter acesso não autorizado. Exemplos de malware incluem vírus, worms, ransomware, cavalos de Tróia, spyware e keyloggers.

serviços gerenciados

Serviços da AWS para o qual AWS opera a camada de infraestrutura, o sistema operacional e as plataformas, e você acessa os endpoints para armazenar e recuperar dados. O Amazon Simple Storage Service (Amazon S3) e o Amazon DynamoDB são exemplos de serviços gerenciados. Eles também são conhecidos como serviços abstratos.

sistema de execução de manufatura (MES)

Um sistema de software para rastrear, monitorar, documentar e controlar processos de produção que convertem matérias-primas em produtos acabados no chão de fábrica.

MAP

Consulte Migration Acceleration Program.

mecanismo

Um processo completo no qual você cria uma ferramenta, impulsiona a adoção da ferramenta e, em seguida, inspeciona os resultados para fazer ajustes. Um mecanismo é um ciclo que se reforça e se aprimora à medida que opera. Para obter mais informações, consulte Construindo mecanismos no AWS Well-Architected Framework.

conta-membro

Todos, Contas da AWS exceto a conta de gerenciamento, que fazem parte de uma organização em AWS Organizations. Uma conta só pode ser membro de uma organização de cada vez.

MES

Veja o sistema de execução de manufatura.

Transporte de telemetria de enfileiramento de mensagens (MQTT)

Um protocolo de comunicação leve machine-to-machine (M2M), baseado no padrão de publicação/assinatura, para dispositivos de IoT com recursos limitados.

microsserviço

Um serviço pequeno e independente que se comunica por meio de APIs bem definidas e normalmente pertence a equipes pequenas e autônomas. Por exemplo, um sistema de seguradora pode incluir microsserviços que mapeiam as capacidades comerciais, como vendas ou marketing, ou subdomínios, como compras, reclamações ou análises. Os benefícios dos microsserviços incluem agilidade, escalabilidade flexível, fácil implantação, código reutilizável e resiliência. Para obter mais informações, consulte Integração de microsserviços usando serviços sem AWS servidor.

arquitetura de microsserviços

Uma abordagem à criação de aplicações com componentes independentes que executam cada processo de aplicação como um microsserviço. Esses microsserviços se comunicam por meio de uma interface bem definida usando APIs leves. Cada microsserviço nessa arquitetura pode ser atualizado, implantado e escalado para atender à demanda por funções específicas de uma aplicação. Para obter mais informações, consulte Implementação de microsserviços em. AWS

Programa de Aceleração da Migração (MAP)

Um AWS programa que fornece suporte de consultoria, treinamento e serviços para ajudar as organizações a criar uma base operacional sólida para migrar para a nuvem e ajudar a

compensar o custo inicial das migrações. O MAP inclui uma metodologia de migração para executar migrações legadas de forma metódica e um conjunto de ferramentas para automatizar e acelerar cenários comuns de migração.

migração em escala

O processo de mover a maior parte do portfólio de aplicações para a nuvem em ondas, com mais aplicações sendo movidas em um ritmo mais rápido a cada onda. Essa fase usa as práticas recomendadas e lições aprendidas nas fases anteriores para implementar uma fábrica de migração de equipes, ferramentas e processos para agilizar a migração de workloads por meio de automação e entrega ágeis. Esta é a terceira fase da estratégia de migração para a AWS.

fábrica de migração

Equipes multifuncionais que simplificam a migração de workloads por meio de abordagens automatizadas e ágeis. As equipes da fábrica de migração geralmente incluem operações, analistas e proprietários de negócios, engenheiros de migração, desenvolvedores e DevOps profissionais que trabalham em sprints. Entre 20 e 50% de um portfólio de aplicações corporativas consiste em padrões repetidos que podem ser otimizados por meio de uma abordagem de fábrica. Para obter mais informações, consulte discussão sobre fábricas de migração e o guia do Cloud Migration Factory neste conjunto de conteúdo.

metadados de migração

As informações sobre a aplicação e o servidor necessárias para concluir a migração. Cada padrão de migração exige um conjunto de metadados de migração diferente. Exemplos de metadados de migração incluem a sub-rede, o grupo de segurança e AWS a conta de destino.

padrão de migração

Uma tarefa de migração repetível que detalha a estratégia de migração, o destino da migração e a aplicação ou o serviço de migração usado. Exemplo: rehospede a migração para o Amazon EC2 AWS com o Application Migration Service.

Avaliação de Portfólio para Migração (MPA)

Uma ferramenta on-line que fornece informações para validar o caso de negócios para a migração para a AWS nuvem. O MPA fornece avaliação detalhada do portfólio (dimensionamento correto do servidor, preços, comparações de TCO, análise de custos de migração), bem como planejamento de migração (análise e coleta de dados de aplicações, agrupamento de aplicações, priorização de migração e planejamento de ondas). A ferramenta MPA (requer login) está disponível gratuitamente para todos os AWS consultores e consultores parceiros da APN.

Avaliação de Preparação para Migração (MRA)

O processo de obter insights sobre o status de prontidão de uma organização para a nuvem, identificar pontos fortes e fracos e criar um plano de ação para fechar as lacunas identificadas, usando o CAF. AWS Para mais informações, consulte o guia de preparação para migração. A MRA é a primeira fase da estratégia de migração para a AWS.

estratégia de migração

A abordagem usada para migrar uma carga de trabalho para a AWS nuvem. Para obter mais informações, consulte a entrada de 7 Rs neste glossário e consulte Mobilize sua organização para acelerar migrações em grande escala.

ML

Veja o aprendizado de máquina.

modernização

Transformar uma aplicação desatualizada (herdada ou monolítica) e sua infraestrutura em um sistema ágil, elástico e altamente disponível na nuvem para reduzir custos, ganhar eficiência e aproveitar as inovações. Para obter mais informações, consulte Estratégia para modernizar aplicativos no Nuvem AWS.

avaliação de preparação para modernização

Uma avaliação que ajuda a determinar a preparação para modernização das aplicações de uma organização. Ela identifica benefícios, riscos e dependências e determina o quão bem a organização pode acomodar o estado futuro dessas aplicações. O resultado da avaliação é um esquema da arquitetura de destino, um roteiro que detalha as fases de desenvolvimento e os marcos do processo de modernização e um plano de ação para abordar as lacunas identificadas. Para obter mais informações, consulte Avaliar a preparação para modernização de aplicações na AWS Cloud.

aplicações monolíticas (monólitos)

Aplicações que são executadas como um único serviço com processos fortemente acoplados. As aplicações monolíticas apresentam várias desvantagens. Se um recurso da aplicação apresentar um aumento na demanda, toda a arquitetura deverá ser escalada. Adicionar ou melhorar os recursos de uma aplicação monolítica também se torna mais complexo quando a base de código cresce. Para resolver esses problemas, é possível criar uma arquitetura de microsserviços. Para obter mais informações, consulte Decompor monólitos em microsserviços.

MAPA

Consulte Avaliação do portfólio de migração.

MQTT

Consulte Transporte de telemetria de enfileiramento de mensagens.

classificação multiclasse

Um processo que ajuda a gerar previsões para várias classes (prevendo um ou mais de dois resultados). Por exemplo, um modelo de ML pode perguntar “Este produto é um livro, um carro ou um telefone?” ou “Qual categoria de produtos é mais interessante para este cliente?”

infraestrutura mutável

Um modelo que atualiza e modifica a infraestrutura existente para cargas de trabalho de produção. Para melhorar a consistência, confiabilidade e previsibilidade, o AWS Well-Architected Framework recomenda o uso de infraestrutura imutável como uma prática recomendada.

O

OAC

Veja o controle de acesso de origem.

CARVALHO

Veja a identidade de acesso de origem.

OCM

Veja o gerenciamento de mudanças organizacionais.

migração offline

Um método de migração no qual a workload de origem é desativada durante o processo de migração. Esse método envolve tempo de inatividade prolongado e geralmente é usado para workloads pequenas e não críticas.

OI

Veja a integração de operações.

OLA

Veja o contrato em nível operacional.

migração online

Um método de migração no qual a workload de origem é copiada para o sistema de destino sem ser colocada offline. As aplicações conectadas à workload podem continuar funcionando durante a migração. Esse método envolve um tempo de inatividade nulo ou mínimo e normalmente é usado para workloads essenciais para a produção.

OPC-UA

Consulte Comunicação de processo aberto — Arquitetura unificada.

Comunicação de processo aberto — Arquitetura unificada (OPC-UA)

Um protocolo de comunicação machine-to-machine (M2M) para automação industrial. O OPC-UA fornece um padrão de interoperabilidade com esquemas de criptografia, autenticação e autorização de dados.

acordo de nível operacional (OLA)

Um acordo que esclarece o que os grupos funcionais de TI prometem oferecer uns aos outros para apoiar um acordo de serviço (SLA).

análise de prontidão operacional (ORR)

Uma lista de verificação de perguntas e melhores práticas associadas que ajudam você a entender, avaliar, prevenir ou reduzir o escopo de incidentes e possíveis falhas. Para obter mais informações, consulte Operational Readiness Reviews (ORR) no Well-Architected AWS Framework.

tecnologia operacional (OT)

Sistemas de hardware e software que funcionam com o ambiente físico para controlar operações, equipamentos e infraestrutura industriais. Na manufatura, a integração dos sistemas OT e de tecnologia da informação (TI) é o foco principal das transformações da Indústria 4.0.

integração de operações (OI)

O processo de modernização das operações na nuvem, que envolve planejamento de preparação, automação e integração. Para obter mais informações, consulte o guia de integração de operações.

trilha organizacional

Uma trilha criada por ela AWS CloudTrail registra todos os eventos de todos Contas da AWS em uma organização em AWS Organizations. Essa trilha é criada em cada Conta da AWS que faz parte da organização e monitora a atividade em cada conta. Para obter mais informações, consulte Criação de uma trilha para uma organização na CloudTrail documentação.

gerenciamento de alterações organizacionais (OCM)

Uma estrutura para gerenciar grandes transformações de negócios disruptivas de uma perspectiva de pessoas, cultura e liderança. O OCM ajuda as organizações a se prepararem e fazerem a transição para novos sistemas e estratégias, acelerando a adoção de alterações, abordando questões de transição e promovendo mudanças culturais e organizacionais. Na estratégia de AWS migração, essa estrutura é chamada de aceleração de pessoas, devido à velocidade de mudança necessária nos projetos de adoção da nuvem. Para obter mais informações, consulte o guia do OCM.

controle de acesso de origem (OAC)

Em CloudFront, uma opção aprimorada para restringir o acesso para proteger seu conteúdo do Amazon Simple Storage Service (Amazon S3). O OAC oferece suporte a todos os buckets S3 Regiões da AWS, criptografia do lado do servidor com AWS KMS (SSE-KMS) e solicitações dinâmicas ao bucket S3. `PUT` `DELETE`

Identidade do acesso de origem (OAI)

Em CloudFront, uma opção para restringir o acesso para proteger seu conteúdo do Amazon S3. Quando você usa o OAI, CloudFront cria um principal com o qual o Amazon S3 pode se autenticar. Os diretores autenticados podem acessar o conteúdo em um bucket do S3 somente por meio de uma distribuição específica. CloudFront Veja também OAC, que fornece um controle de acesso mais granular e aprimorado.

OU

Veja a análise de prontidão operacional.

NÃO

Veja a tecnologia operacional.

VPC de saída (egresso)

Em uma arquitetura de AWS várias contas, uma VPC que gerencia conexões de rede que são iniciadas de dentro de um aplicativo. A Arquitetura de referência de segurança da AWS

recomenda configurar sua conta de rede com VPCs de entrada, saída e inspeção para proteger a interface bidirecional entre a aplicação e a Internet em geral.

P

limite de permissões

Uma política de gerenciamento do IAM anexada a entidades principais do IAM para definir as permissões máximas que o usuário ou perfil podem ter. Para obter mais informações, consulte Limites de permissões na documentação do IAM.

informações de identificação pessoal (PII)

Informações que, quando visualizadas diretamente ou combinadas com outros dados relacionados, podem ser usadas para inferir razoavelmente a identidade de um indivíduo. Exemplos de PII incluem nomes, endereços e informações de contato.

PII

Veja as informações de identificação pessoal.

manual

Um conjunto de etapas predefinidas que capturam o trabalho associado às migrações, como a entrega das principais funções operacionais na nuvem. Um manual pode assumir a forma de scripts, runbooks automatizados ou um resumo dos processos ou etapas necessários para operar seu ambiente modernizado.

PLC

Consulte controlador lógico programável.

AMEIXA

Veja o gerenciamento do ciclo de vida do produto.

política

Um objeto que pode definir permissões (consulte a política baseada em identidade), especificar as condições de acesso (consulte a política baseada em recursos) ou definir as permissões máximas para todas as contas em uma organização em AWS Organizations (consulte a política de controle de serviços).

persistência poliglota

Escolher de forma independente a tecnologia de armazenamento de dados de um microsserviço com base em padrões de acesso a dados e outros requisitos. Se seus microsserviços tiverem a mesma tecnologia de armazenamento de dados, eles poderão enfrentar desafios de implementação ou apresentar baixa performance. Os microsserviços serão implementados com mais facilidade e alcançarão performance e escalabilidade melhores se usarem o armazenamento de dados mais bem adaptado às suas necessidades. Para obter mais informações, consulte Habilitar a persistência de dados em microsserviços.

avaliação do portfólio

Um processo de descobrir, analisar e priorizar o portfólio de aplicações para planejar a migração. Para obter mais informações, consulte Avaliar a preparação para a migração.

predicado

Uma condição de consulta que retorna `true` ou`false`, normalmente localizada em uma `WHERE` cláusula.

pressão de predicados

Uma técnica de otimização de consulta de banco de dados que filtra os dados na consulta antes da transferência. Isso reduz a quantidade de dados que devem ser recuperados e processados do banco de dados relacional e melhora o desempenho das consultas.

controle preventivo

Um controle de segurança projetado para evitar que um evento ocorra. Esses controles são a primeira linha de defesa para ajudar a evitar acesso não autorizado ou alterações indesejadas em sua rede. Para obter mais informações, consulte Controles preventivos em Como implementar controles de segurança na AWS.

principal (entidade principal)

Uma entidade AWS que pode realizar ações e acessar recursos. Essa entidade geralmente é um usuário raiz para um Conta da AWS, uma função do IAM ou um usuário. Para obter mais informações, consulte Entidade principal em Termos e conceitos de perfis na documentação do IAM.

Privacidade por design

Uma abordagem em engenharia de sistemas que leva em consideração a privacidade em todo o processo de engenharia.

zonas hospedadas privadas

Um contêiner que armazena informações sobre como você quer que o Amazon Route 53 responda a consultas ao DNS para um domínio e seus subdomínios dentro de uma ou mais VPCs. Para obter mais informações, consulte Como trabalhar com zonas hospedadas privadas na documentação do Route 53.

controle proativo

Um controle de segurança projetado para impedir a implantação de recursos não compatíveis. Esses controles examinam os recursos antes de serem provisionados. Se o recurso não estiver em conformidade com o controle, ele não será provisionado. Para obter mais informações, consulte o guia de referência de controles na AWS Control Tower documentação e consulte Controles proativos em Implementação de controles de segurança em AWS.

gerenciamento do ciclo de vida do produto (PLM)

O gerenciamento de dados e processos de um produto em todo o seu ciclo de vida, desde o design, desenvolvimento e lançamento, passando pelo crescimento e maturidade, até o declínio e a remoção.

ambiente de produção

Veja o ambiente.

controlador lógico programável (PLC)

Na fabricação, um computador altamente confiável e adaptável que monitora as máquinas e automatiza os processos de fabricação.

pseudonimização

O processo de substituir identificadores pessoais em um conjunto de dados por valores de espaço reservado. A pseudonimização pode ajudar a proteger a privacidade pessoal. Os dados pseudonimizados ainda são considerados dados pessoais.

publicar/assinar (pub/sub)

Um padrão que permite comunicações assíncronas entre microsserviços para melhorar a escalabilidade e a capacidade de resposta. Por exemplo, em um MES baseado em microsserviços, um microsserviço pode publicar mensagens de eventos em um canal no qual outros microsserviços possam se inscrever. O sistema pode adicionar novos microsserviços sem alterar o serviço de publicação.

Q

plano de consulta

Uma série de etapas, como instruções, usadas para acessar os dados em um sistema de banco de dados relacional SQL.

regressão de planos de consultas

Quando um otimizador de serviço de banco de dados escolhe um plano menos adequado do que escolhia antes de uma determinada alteração no ambiente de banco de dados ocorrer. Isso pode ser causado por alterações em estatísticas, restrições, configurações do ambiente, associações de parâmetros de consulta e atualizações do mecanismo de banco de dados.

R

Matriz RACI

Veja responsável, responsável, consultado, informado (RACI).

ransomware

Um software mal-intencionado desenvolvido para bloquear o acesso a um sistema ou dados de computador até que um pagamento seja feito.

Matriz RASCI

Veja responsável, responsável, consultado, informado (RACI).

RCAC

Veja o controle de acesso por linha e coluna.

réplica de leitura

Uma cópia de um banco de dados usada somente para leitura. É possível encaminhar consultas para a réplica de leitura e reduzir a carga no banco de dados principal.

rearquiteta

Veja 7 Rs.

objetivo de ponto de recuperação (RPO)

Período de tempo aceitável máximo desde o último ponto de recuperação de dados. Determina o que é considerado uma perda aceitável de dados entre o último ponto de recuperação e a interrupção do serviço.

objetivo de tempo de recuperação (RTO)

O atraso máximo aceitável entre a interrupção e a restauração do serviço.

refatorar

Veja 7 Rs.

Região

Uma coleção de AWS recursos em uma área geográfica. Cada um Região da AWS é isolado e independente dos outros para fornecer tolerância a falhas, estabilidade e resiliência. Para obter mais informações, consulte Especificar o que Regiões da AWS sua conta pode usar.

regressão

Uma técnica de ML que prevê um valor numérico. Por exemplo, para resolver o problema de “Por qual preço esta casa será vendida?” um modelo de ML pode usar um modelo de regressão linear para prever o preço de venda de uma casa com base em fatos conhecidos sobre a casa (por exemplo, a metragem quadrada).

redefinir a hospedagem

Veja 7 Rs.

versão

Em um processo de implantação, o ato de promover mudanças em um ambiente de produção.

realocar

Veja 7 Rs.

redefinir a plataforma

Veja 7 Rs.

recomprar

Veja 7 Rs.

resiliência

A capacidade de um aplicativo de resistir ou se recuperar de interrupções. Alta disponibilidade e recuperação de desastres são considerações comuns ao planejar a resiliência no. Nuvem AWS Para obter mais informações, consulte Nuvem AWS Resiliência.

política baseada em recurso

Uma política associada a um recurso, como um bucket do Amazon S3, um endpoint ou uma chave de criptografia. Esse tipo de política especifica quais entidades principais têm acesso permitido, ações válidas e quaisquer outras condições que devem ser atendidas.

matriz responsável, accountable, consultada, informada (RACI)

Uma matriz que define as funções e responsabilidades de todas as partes envolvidas nas atividades de migração e nas operações de nuvem. O nome da matriz é derivado dos tipos de responsabilidade definidos na matriz: responsável (R), responsabilizável (A), consultado (C) e informado (I). O tipo de suporte (S) é opcional. Se você incluir suporte, a matriz será chamada de matriz RASCI e, se excluir, será chamada de matriz RACI.

controle responsivo

Um controle de segurança desenvolvido para conduzir a remediação de eventos adversos ou desvios em relação à linha de base de segurança. Para obter mais informações, consulte Controles responsivos em Como implementar controles de segurança na AWS.

reter

Veja 7 Rs.

aposentar-se

Veja 7 Rs.

rotação

O processo de atualizar periodicamente um segredo para dificultar o acesso das credenciais por um invasor.

controle de acesso por linha e coluna (RCAC)

O uso de expressões SQL básicas e flexíveis que tenham regras de acesso definidas. O RCAC consiste em permissões de linha e máscaras de coluna.

RPO

Veja o objetivo do ponto de recuperação.

RTO

Veja o objetivo do tempo de recuperação.

runbook

Um conjunto de procedimentos manuais ou automatizados necessários para realizar uma tarefa específica. Eles são normalmente criados para agilizar operações ou procedimentos repetitivos com altas taxas de erro.

S

SAML 2.0

Um padrão aberto que muitos provedores de identidade (IdPs) usam. Esse recurso permite o login único federado (SSO), para que os usuários possam fazer login AWS Management Console ou chamar as operações da AWS API sem que você precise criar um usuário no IAM para todos em sua organização. Para obter mais informações sobre a federação baseada em SAML 2.0, consulte Sobre a federação baseada em SAML 2.0 na documentação do IAM.

SCADA

Veja controle de supervisão e aquisição de dados.

SCP

Veja a política de controle de serviços.

secret

Em AWS Secrets Manager, informações confidenciais ou restritas, como uma senha ou credenciais de usuário, que você armazena de forma criptografada. Ele consiste no valor secreto e em seus metadados. O valor secreto pode ser binário, uma única string ou várias strings. Para obter mais informações, consulte Secret na documentação do Secrets Manager.

controle de segurança

Uma barreira de proteção técnica ou administrativa que impede, detecta ou reduz a capacidade de uma ameaça explorar uma vulnerabilidade de segurança. Existem quatro tipos principais de controles de segurança: preventivos, detectivos, responsivos e proativos.

fortalecimento da segurança

O processo de reduzir a superfície de ataque para torná-la mais resistente a ataques. Isso pode incluir ações como remover recursos que não são mais necessários, implementar a prática recomendada de segurança de conceder privilégios mínimos ou desativar recursos desnecessários em arquivos de configuração.

sistema de gerenciamento de eventos e informações de segurança (SIEM)

Ferramentas e serviços que combinam sistemas de gerenciamento de informações de segurança (SIM) e gerenciamento de eventos de segurança (SEM). Um sistema SIEM coleta, monitora e analisa dados de servidores, redes, dispositivos e outras fontes para detectar ameaças e violações de segurança e gerar alertas.

automação de resposta de segurança

Uma ação predefinida e programada projetada para responder ou remediar automaticamente um evento de segurança. Essas automações servem como controles de segurança responsivos ou detectivos que ajudam você a implementar as melhores práticas AWS de segurança. Exemplos de ações de resposta automatizada incluem a modificação de um grupo de segurança da VPC, a correção de uma instância do Amazon EC2 ou a rotação de credenciais.

Criptografia do lado do servidor

Criptografia dos dados em seu destino, por AWS service (Serviço da AWS) quem os recebe.

política de controle de serviços (SCP)

Uma política que fornece controle centralizado sobre as permissões de todas as contas em uma organização no AWS Organizations. As SCPs definem barreiras de proteção ou estabelecem limites para as ações que um administrador pode delegar a usuários ou perfis. É possível usar SCPs como listas de permissão ou de negação para especificar quais serviços ou ações são permitidos ou proibidos. Para obter mais informações, consulte Políticas de controle de serviço na AWS Organizations documentação.

service endpoint (endpoint de serviço)

O URL do ponto de entrada para um AWS service (Serviço da AWS). Você pode usar o endpoint para se conectar programaticamente ao serviço de destino. Para obter mais informações, consulte Endpoints do AWS service (Serviço da AWS) na Referência geral da AWS.

acordo de serviço (SLA)

Um acordo que esclarece o que uma equipe de TI promete fornecer aos clientes, como tempo de atividade e performance do serviço.

indicador de nível de serviço (SLI)

Uma medida de um aspecto de desempenho de um serviço, como taxa de erro, disponibilidade ou taxa de transferência.

objetivo de nível de serviço (SLO)

Uma métrica alvo que representa a integridade de um serviço, conforme medida por um indicador de nível de serviço.

modelo de responsabilidade compartilhada

Um modelo que descreve a responsabilidade com a qual você compartilha AWS pela segurança e conformidade na nuvem. AWS é responsável pela segurança da nuvem, enquanto você é responsável pela segurança na nuvem. Para obter mais informações, consulte o Modelo de responsabilidade compartilhada.

SIEM

Veja informações de segurança e sistema de gerenciamento de eventos.

ponto único de falha (SPOF)

Uma falha em um único componente crítico de um aplicativo que pode interromper o sistema.

SLA

Veja o contrato de nível de serviço.

ESGUIO

Veja o indicador de nível de serviço.

SLO

Veja o objetivo do nível de serviço.

split-and-seed modelo

Um padrão para escalar e acelerar projetos de modernização. À medida que novos recursos e lançamentos de produtos são definidos, a equipe principal se divide para criar novas equipes

de produtos. Isso ajuda a escalar os recursos e os serviços da sua organização, melhora a produtividade do desenvolvedor e possibilita inovações rápidas. Para obter mais informações, consulte Abordagem em fases para modernizar aplicativos no. Nuvem AWS

CUSPE

Veja um único ponto de falha.

esquema de estrelas

Uma estrutura organizacional de banco de dados que usa uma grande tabela de fatos para armazenar dados transacionais ou medidos e usa uma ou mais tabelas dimensionais menores para armazenar atributos de dados. Essa estrutura foi projetada para uso em um data warehouse ou para fins de inteligência comercial.

padrão strangler fig

Uma abordagem à modernização de sistemas monolíticos que consiste em reescrever e substituir incrementalmente a funcionalidade do sistema até que o sistema herdado possa ser desativado. Esse padrão usa a analogia de uma videira que cresce e se torna uma árvore estabelecida e, eventualmente, supera e substitui sua hospedeira. O padrão foi apresentado por Martin Fowler como forma de gerenciar riscos ao reescrever sistemas monolíticos. Para ver um exemplo de como aplicar esse padrão, consulte Modernizar incrementalmente os serviços Web herdados do Microsoft ASP.NET (ASMX) usando contêineres e o Amazon API Gateway.

sub-rede

Um intervalo de endereços IP na VPC. Uma sub-rede deve residir em uma única zona de disponibilidade.

controle de supervisão e aquisição de dados (SCADA)

Na manufatura, um sistema que usa hardware e software para monitorar ativos físicos e operações de produção.

symmetric encryption (criptografia simétrica)

Um algoritmo de criptografia que usa a mesma chave para criptografar e descriptografar dados.

testes sintéticos

Testar um sistema de forma que simule as interações do usuário para detectar possíveis problemas ou monitorar o desempenho. Você pode usar o Amazon CloudWatch Synthetics para criar esses testes.

T

tags

Pares de valores-chave que atuam como metadados para organizar seus recursos. AWS As tags podem ajudar você a gerenciar, identificar, organizar, pesquisar e filtrar recursos. Para obter mais informações, consulte Marcar seus recursos do AWS.

variável-alvo

O valor que você está tentando prever no ML supervisionado. Ela também é conhecida como variável de resultado. Por exemplo, em uma configuração de fabricação, a variável-alvo pode ser um defeito do produto.

lista de tarefas

Uma ferramenta usada para monitorar o progresso por meio de um runbook. Uma lista de tarefas contém uma visão geral do runbook e uma lista de tarefas gerais a serem concluídas. Para cada tarefa geral, ela inclui o tempo estimado necessário, o proprietário e o progresso.

ambiente de teste

Veja o ambiente.

treinamento

O processo de fornecer dados para que seu modelo de ML aprenda. Os dados de treinamento devem conter a resposta correta. O algoritmo de aprendizado descobre padrões nos dados de treinamento que mapeiam os atributos dos dados de entrada no destino (a resposta que você deseja prever). Ele gera um modelo de ML que captura esses padrões. Você pode usar o modelo de ML para obter previsões de novos dados cujo destino você não conhece.

gateway de trânsito

Um hub de trânsito de rede que pode ser usado para interconectar as VPCs e as redes on-premises. Para obter mais informações, consulte O que é um gateway de trânsito na AWS Transit Gateway documentação.

fluxo de trabalho baseado em troncos

Uma abordagem na qual os desenvolvedores criam e testam recursos localmente em uma ramificação de recursos e, em seguida, mesclam essas alterações na ramificação principal. A

ramificação principal é então criada para os ambientes de desenvolvimento, pré-produção e produção, sequencialmente.

Acesso confiável

Conceder permissões a um serviço que você especifica para realizar tarefas em sua organização AWS Organizations e em suas contas em seu nome. O serviço confiável cria um perfil vinculado ao serviço em cada conta, quando esse perfil é necessário, para realizar tarefas de gerenciamento para você. Para obter mais informações, consulte Usando AWS Organizations com outros AWS serviços na AWS Organizations documentação.

tuning (ajustar)

Alterar aspectos do processo de treinamento para melhorar a precisão do modelo de ML. Por exemplo, você pode treinar o modelo de ML gerando um conjunto de rótulos, adicionando rótulos e repetindo essas etapas várias vezes em configurações diferentes para otimizar o modelo.

equipe de duas pizzas

Uma pequena DevOps equipe que você pode alimentar com duas pizzas. Uma equipe de duas pizzas garante a melhor oportunidade possível de colaboração no desenvolvimento de software.

U

incerteza

Um conceito que se refere a informações imprecisas, incompletas ou desconhecidas que podem minar a confiabilidade dos modelos preditivos de ML. Há dois tipos de incertezas: a incerteza epistêmica é causada por dados limitados e incompletos, enquanto a incerteza aleatória é causada pelo ruído e pela aleatoriedade inerentes aos dados. Para obter mais informações, consulte o guia Como quantificar a incerteza em sistemas de aprendizado profundo.

tarefas indiferenciadas

Também conhecido como trabalho pesado, trabalho necessário para criar e operar um aplicativo, mas que não fornece valor direto ao usuário final nem oferece vantagem competitiva. Exemplos de tarefas indiferenciadas incluem aquisição, manutenção e planejamento de capacidade.

ambientes superiores

Veja o ambiente.

V

aspiração

Uma operação de manutenção de banco de dados que envolve limpeza após atualizações incrementais para recuperar armazenamento e melhorar a performance.

controle de versões

Processos e ferramentas que rastreiam mudanças, como alterações no código-fonte em um repositório.

emparelhamento de VPC

Uma conexão entre duas VPCs que permite rotear tráfego usando endereços IP privados. Para ter mais informações, consulte O que é emparelhamento de VPC? na documentação da Amazon VPC.

vulnerabilidade

Uma falha de software ou hardware que compromete a segurança do sistema.

W

cache quente

Um cache de buffer que contém dados atuais e relevantes que são acessados com frequência. A instância do banco de dados pode ler do cache do buffer, o que é mais rápido do que ler da memória principal ou do disco.

dados mornos

Dados acessados raramente. Ao consultar esse tipo de dados, consultas moderadamente lentas geralmente são aceitáveis.

função de janela

Uma função SQL que executa um cálculo em um grupo de linhas que se relacionam de alguma forma com o registro atual. As funções de janela são úteis para processar tarefas, como calcular uma média móvel ou acessar o valor das linhas com base na posição relativa da linha atual.

workload

Uma coleção de códigos e recursos que geram valor empresarial, como uma aplicação voltada para o cliente ou um processo de back-end.

workstreams

Grupos funcionais em um projeto de migração que são responsáveis por um conjunto específico de tarefas. Cada workstream é independente, mas oferece suporte aos outros workstreams do projeto. Por exemplo, o workstream de portfólio é responsável por priorizar aplicações, planejar ondas e coletar metadados de migração. O workstream de portfólio entrega esses ativos ao workstream de migração, que então migra os servidores e as aplicações.

MINHOCA

Veja escrever uma vez, ler muitas.

WQF

Consulte o AWS Workload Qualification Framework.

escreva uma vez, leia muitas (WORM)

Um modelo de armazenamento que grava dados uma única vez e evita que os dados sejam excluídos ou modificados. Os usuários autorizados podem ler os dados quantas vezes forem necessárias, mas não podem alterá-los. Essa infraestrutura de armazenamento de dados é considerada imutável.

Z

exploração de dia zero

Um ataque, geralmente malware, que tira proveito de uma vulnerabilidade de dia zero.

vulnerabilidade de dia zero

Uma falha ou vulnerabilidade não mitigada em um sistema de produção. Os agentes de ameaças podem usar esse tipo de vulnerabilidade para atacar o sistema. Os desenvolvedores frequentemente ficam cientes da vulnerabilidade como resultado do ataque.

aplicação zumbi

Uma aplicação que tem um uso médio de CPU e memória inferior a 5%. Em um projeto de migração, é comum retirar essas aplicações.

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.