Criação de arquiteturas multilocatárias para IA agêntica em AWS

AWS Orientação prescritiva

AWS Orientação prescritiva: Criação de arquiteturas multilocatárias para IA agêntica em AWS

Table of Contents

Criação de arquiteturas multilocatárias para IA agêntica em AWS

Aaron Sempf e Tod Golding, da Amazon Web Services

Julho de 2025 (histórico do documento)

A Inteligência Artificial representa uma mudança de paradigma disruptiva que exige que as organizações repensem como construir, entregar e operar seus sistemas. O modelo agêntico tem equipes explorando novas maneiras de decompor sistemas em um ou mais agentes que criam novos caminhos, possibilidades e valores.

Grande parte da discussão sobre agentes gira em torno das ferramentas, estruturas e padrões usados para criar e implementar agentes. Não devemos apenas adotar boas ferramentas para criar agentes, mas também novos protocolos de integração, estratégias de autenticação e mecanismos de descoberta que possam servir como base de arquiteturas agentes.

Enquanto o número de ferramentas agentes cresce, as equipes também devem considerar como seus agentes lidam com os desafios da arquitetura mais tradicional. Escala, vizinhança ruidosa, resiliência, custo e eficiência operacional são tópicos fundamentais que devem ser avaliados ao projetar, criar e implantar agentes. Independentemente de quão autônomos e inteligentes os agentes possam ser, também devemos garantir que eles alcancem economias de escala, eficiência e agilidade alinhadas às necessidades dos negócios.

O objetivo deste guia é explorar várias dimensões das pegadas dos agentes. Isso inclui analisar vários padrões de implantação e consumo de agentes e destacar diferentes estratégias para criar agentes que atendam às metas arquitetônicas. Isso também significa analisar como os agentes podem ser consumidos em um ambiente multilocatário, introduzindo construções internas que normalmente são necessárias em um ambiente multilocatário.

Público-alvo

Este guia é para arquitetos, desenvolvedores e líderes de tecnologia que desejam criar sistemas multilocatários orientados por IA.

Objetivos

Este guia ajuda você a:

- Entenda as implantações de agentes multilocatários, explorando modelos isolados e agrupados, e como o contexto do inquilino afeta a implementação do agente
- Explore o gerenciamento de agentes, incluindo integração, isolamento de inquilinos e gerenciamento de recursos em ambientes de um e vários fornecedores
- Avalie aspectos de agentes multilocatários, incluindo propriedade de dados, monitoramento e testes

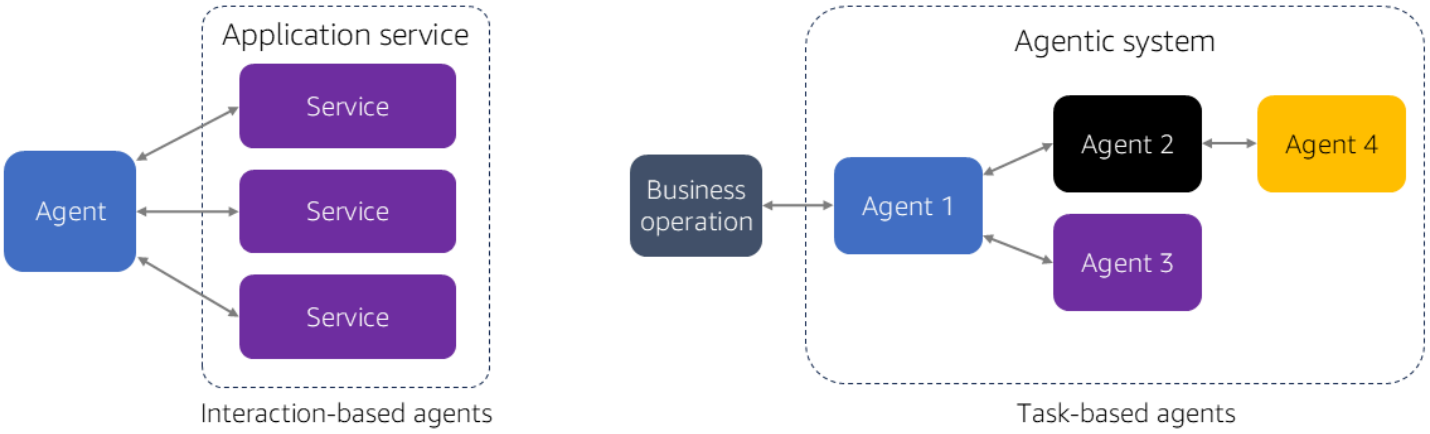
Sobre esta série de conteúdo

Este guia faz parte de uma série sobre IA agente em AWS. Para obter mais informações e ver os outros guias desta série, consulte Agentic AI no site da AWS Prescriptive Guidance.

Fundamentos do agente

Antes de discutirmos os detalhes da arquitetura, devemos descrever as diferentes funções que os agentes desempenham porque “agente” é um termo sobrecarregado que pode ser aplicado a muitos casos de uso. Vamos começar com alguns termos gerais que podem ajudar a categorizá-los.

No nível mais externo, precisamos começar classificando o papel e a natureza dos agentes. Isso é desafiador porque há uma grande variedade de cenários em que os agentes podem ser aplicados a qualquer número de problemas. Para essa discussão, porém, vamos nos concentrar no que significa introduzir um agente em um aplicativo ou sistema. Nesse modelo, enfatizamos como e onde os agentes podem enriquecer melhor a experiência do seu sistema. As opções escolhidas influenciam a forma como seus agentes são criados, integrados e aplicados a diferentes domínios e casos de uso. O diagrama a seguir mostra dois padrões agentes que os construtores usam.

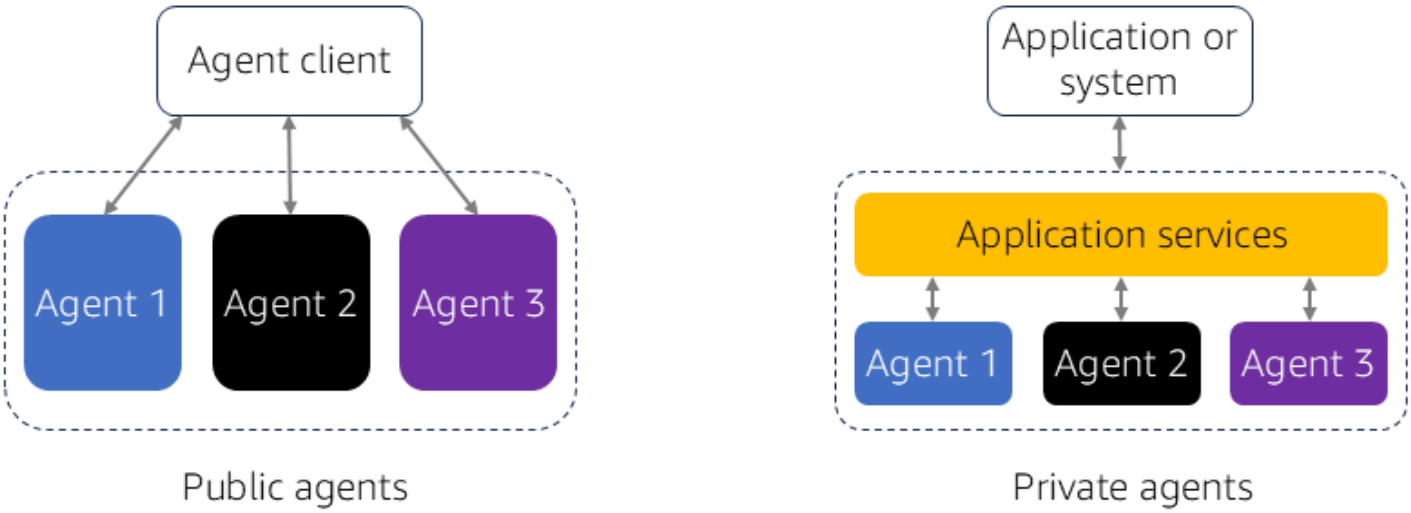


No lado esquerdo do diagrama está um agente baseado em interação. Nesse modo, um agente cria uma visão de um sistema existente para orquestrar interações com os serviços subjacentes para atingir uma meta ou resultado. A chave é que o agente seja adicionado a um sistema como uma abordagem alternativa para direcionar os recursos e capacidades do sistema. Imagine, por exemplo, que um fornecedor independente de software (ISV) tenha um sistema contábil com uma UX usada para realizar operações. O agente baseado em interação simplifica a interação com esses recursos existentes. É menos sobre aprender como alcançar uma meta vagamente definida e mais sobre fornecer uma maneira de orquestrar caminhos conhecidos.

Por outro lado, o sistema baseado em tarefas no lado direito do diagrama representa uma abordagem diferente. Os agentes desse sistema usam seus conhecimentos e habilidades para aprender a concluir tarefas e gerar resultados comerciais. Você poderia argumentar que os dois modelos alcançam resultados comerciais, mas um modelo baseado em tarefas depende

dos próprios agentes para determinar como alcançar um resultado. Esses agentes são menos deterministas e, em vez disso, confiam em sua capacidade de aprender e evoluir. Por outro lado, os agentes baseados em interação são projetados principalmente para orquestrar um conjunto de recursos conhecidos. Essas diferenças afetam a forma como você cria, define o escopo e integra agentes para apoiar seus negócios.

Também precisamos de termos que caracterizem como e onde implantamos agentes. O local onde um agente vive dentro da área de cobertura do seu sistema pode influenciar a forma como ele é construído, definido e protegido. O diagrama a seguir descreve dois modelos distintos que podem ser aplicados aos agentes.

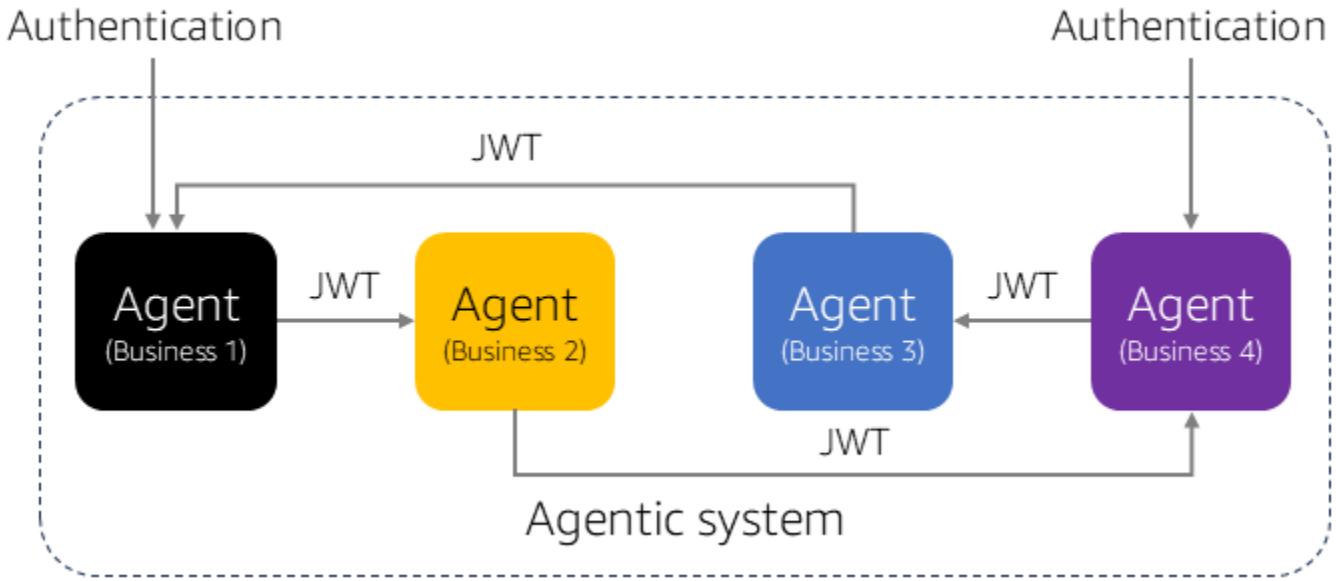


No lado esquerdo do diagrama, há um sistema de implantação com três agentes diferentes. Os agentes são expostos a clientes externos que podem ser outros agentes ou aplicativos. Para esse modelo, os agentes são chamados de agentes públicos.

Por outro lado, o diagrama no lado direito mostra os agentes dentro da implementação da solução. Nesse caso, há uma série de serviços de aplicativos que são consumidos por usuários ou sistemas. Esses usuários interagem com o aplicativo sem saber que os agentes fazem parte da experiência. Os agentes são então invocados e orquestrados pelos serviços do sistema subjacente. Agentes implantados dessa maneira são chamados de agentes privados.

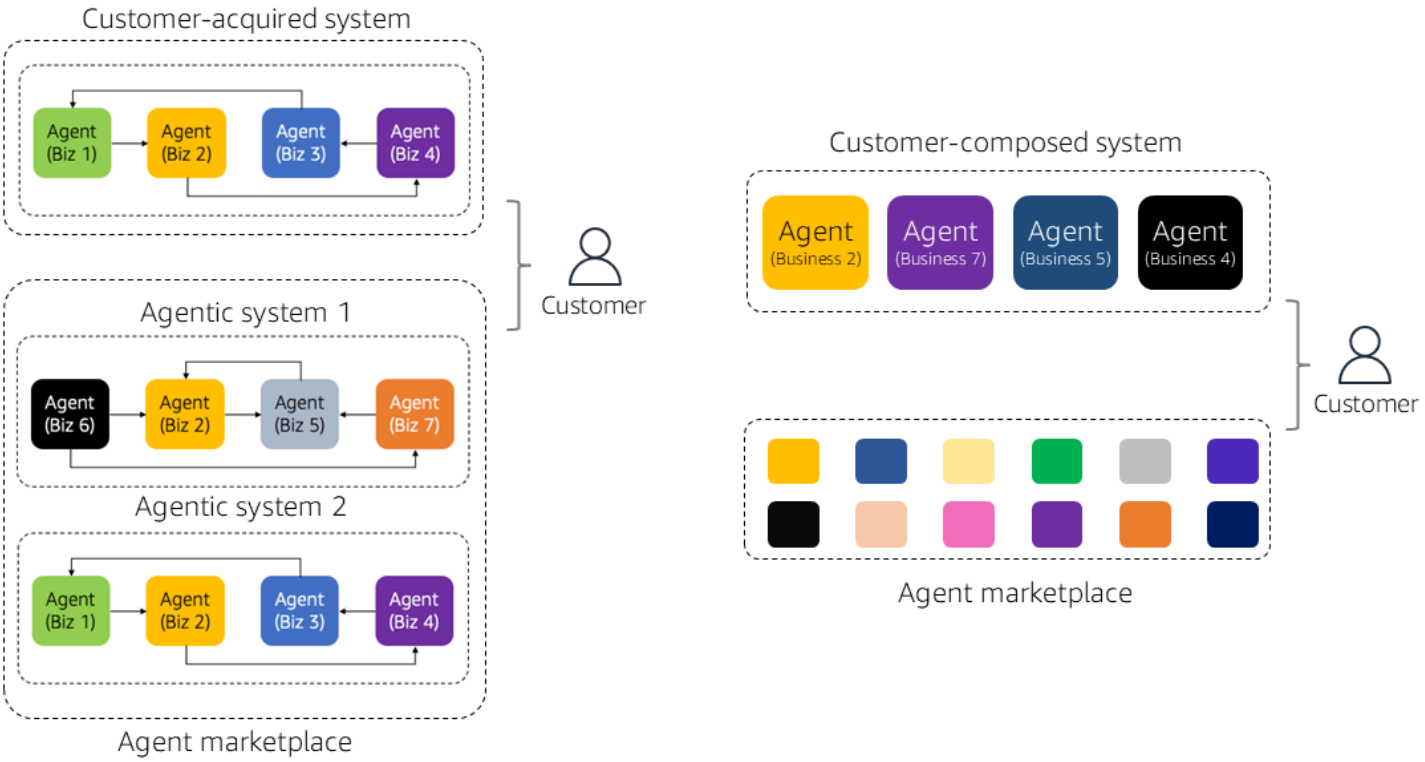
Muito do valor de um agente se concentra no modelo público em que os provedores podem publicar seus agentes com a intenção de integrá-los a outros agentes terceirizados. Os agentes então fariam parte de uma malha ou rede de serviços interconectados que, coletivamente, seriam capazes de lidar com muitos casos de uso. Embora esses agentes possam ser usados em muitos domínios,

o caso de business-to-business uso é uma opção natural. O diagrama a seguir fornece uma visão conceituada de como seria montar um agente de coleta que resolva um problema específico.



O diagrama mostra quatro agentes de negócios que trabalham juntos para atingir um conjunto de objetivos. Quando os agentes são compostos dessa forma, eles representam um sistema agente, e há muitos tipos desses sistemas. Eles podem ser um conjunto pré-empacotado de agentes colaboradores que geralmente são consumidos como uma única unidade. Ou o sistema pode ser montado dinamicamente por clientes que desejam escolher uma combinação de agentes que melhor atenda às suas necessidades.

Ambas as abordagens oferecem caminhos viáveis para a integração de agentes. Alguns agentes são criados com a expectativa de serem integrados a sistemas específicos onde possam maximizar seu valor, alcance e impacto. Essa noção de sistemas agentes também levanta questões sobre como os agentes são adquiridos, e pode haver muitas maneiras de lidar com isso. O diagrama a seguir fornece exemplos de como esses agentes e sistemas podem ser criados por meio de experiências transacionais.

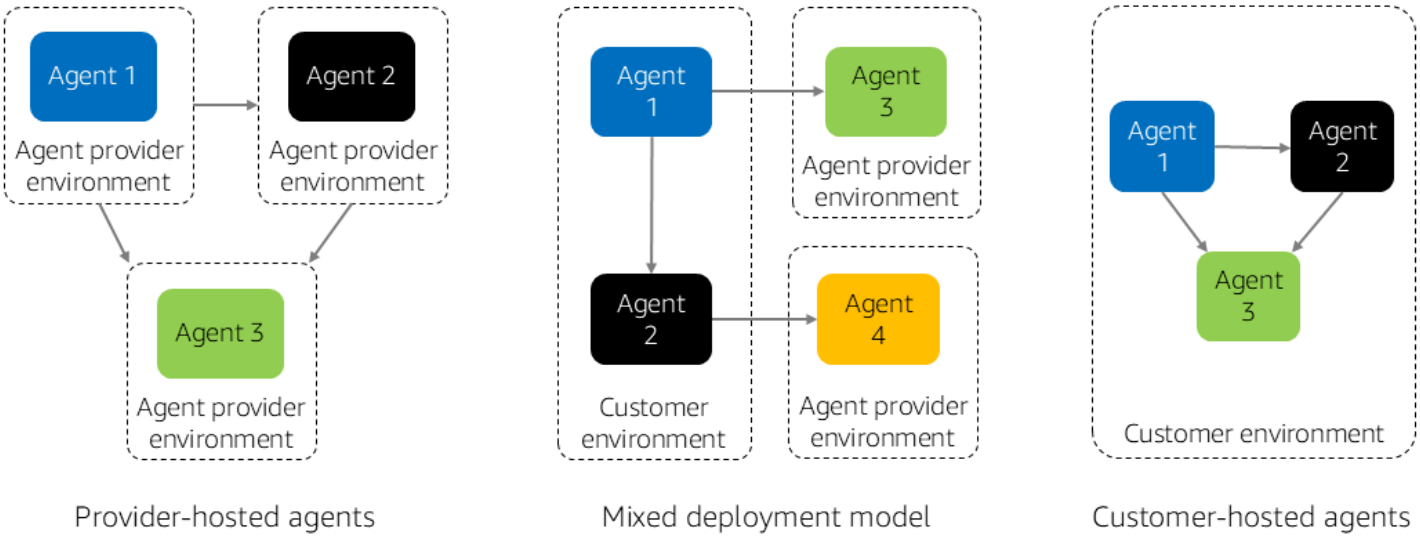


Dois exemplos de experiências de mercado são mostrados. No lado esquerdo, um mercado é usado para adquirir sistemas pré-embalados. Nesse cenário, o mercado descobre e integra sistemas que abordam objetivos mais amplos que exigem a integração e a orquestração de vários agentes.

O exemplo no lado direito mostra um mercado em que os agentes são descobertos e compostos em sistemas agentes. Nesse cenário, os clientes podem criar qualquer sistema de agentes compatíveis e integrados para atender às suas necessidades. A capacidade de montar agentes dessa maneira depende do modelo de compatibilidade e dos requisitos de integração de agentes individuais.

Considerações sobre hospedagem de agentes

Agora que você tem uma ideia dos conceitos mais amplos de agentes, vamos discutir o que significa hospedar e administrar esses agentes. Precisamos pensar em como e onde as computações são executadas, como elas se expandem, como operam e como são gerenciadas. Ao mesmo tempo, alguns padrões que esperamos ver como agentes são mais amplamente aplicados e adotados. O diagrama a seguir mostra um exemplo de permutações prováveis.



Três estratégias distintas são representadas aqui. No lado esquerdo do diagrama, você vê um modelo em que nossos agentes são hospedados, escalados e gerenciados nos ambientes de cada provedor de agentes. Esses agentes são publicados e consumidos como serviços, operando no que é rotulado como modelo de agente como serviço (AaaS). No lado direito, há um modelo em que os agentes de um provedor estão todos hospedados em um ambiente dedicado ao cliente.

No meio do diagrama, há um modelo misto de implantação que combina essas duas estratégias, hospedando alguns agentes localmente no ambiente do cliente e interagindo com alguns agentes hospedados remotamente no ambiente de um provedor.

Uma quarta opção (não mostrada) pode ser quando os agentes são criados como serviços de baixo ou nenhum código que são escalados e gerenciados pelos serviços de infraestrutura do agente. Não abordaremos isso em detalhes porque a arquitetura e a hospedagem dos agentes gerenciados são ditadas principalmente pela organização proprietária dos serviços.

Você pode imaginar a variedade de fatores que podem influenciar a adoção de um desses modelos. Restrições de conformidade, regulatórias e de segurança, por exemplo, podem levar alguém a

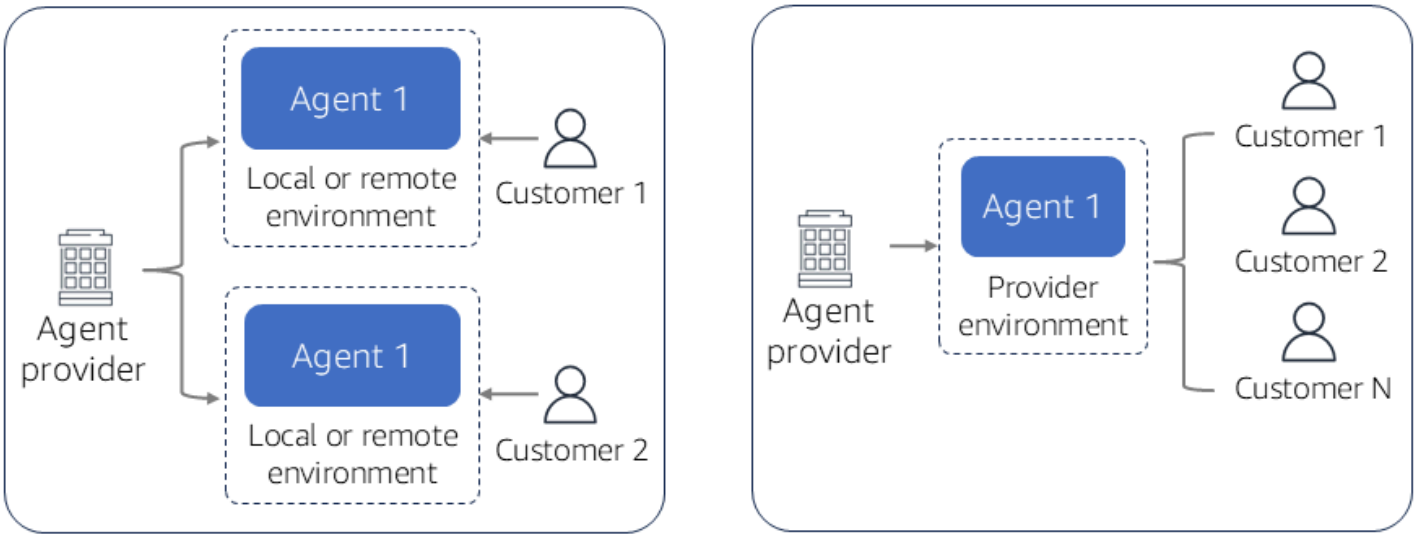
procurar agentes hospedados pelo cliente. Escala, agilidade e eficiência podem impulsionar as organizações mais para o modelo AaaS.

O conceito-chave aqui é que os agentes podem e são implantados e hospedados de várias maneiras. É seu trabalho determinar a melhor forma de aplicar os agentes. A pegada, a segurança e a implantação, entre outros fatores, afetam significativamente a forma como você aborda os agentes de construção e operação. Agentes públicos e privados, por exemplo, podem ter designs e ciclos de vida de lançamento diferentes.

Agentes atendem à multilocação

É fácil pensar nos agentes como elementos básicos em que os agentes são vistos como uma série de componentes autônomos que são montados para atender às necessidades de um domínio ou problema comercial específico. O que fica mais interessante é quando começamos a pensar em como esses agentes são empacotados e consumidos pelos fornecedores. Em muitos aspectos, um agente se torna uma fonte de custo e receita para uma empresa. Os provedores de agentes devem considerar as diferentes personas que consomem seus serviços, o perfil de consumo das personas e as estratégias de monetização que permitem que os provedores de agentes criem modelos de preços e níveis que se alinhem aos consumidores.

Os fornecedores de agentes poderiam oferecer suporte a vários modelos para implantar seus agentes para atender às necessidades do cliente. O diagrama a seguir mostra uma visão conceitual dos dois principais modelos de implantação de agentes.



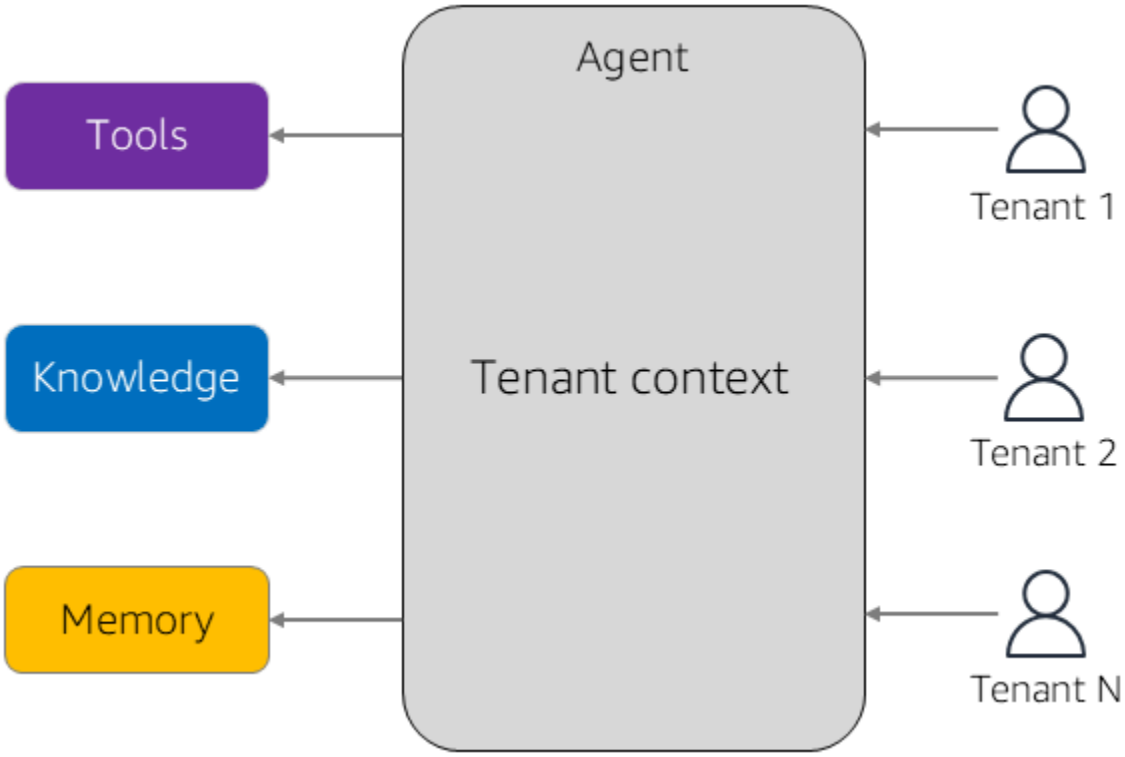
O lado esquerdo do diagrama mostra o modelo de agente dedicado ao cliente. Um provedor de agentes cria um agente implantando uma instância de agente separada para cada cliente integrado. Com essa abordagem, os recursos do agente e sua capacidade de adquirir conhecimento seriam limitados ao escopo do ambiente de um determinado cliente. Isso acaba representando uma experiência por cliente que herda algumas das complexidades e vantagens de oferecer suporte a ambientes dedicados ao cliente.

Por outro lado, o diagrama no lado direito do diagrama tem um único agente que é implantado no ambiente do provedor. O agente processa solicitações de vários clientes, evoluindo e aprendendo

com base na experiência coletiva de todos os clientes. Cada novo cliente adicionado simplesmente representaria outro cliente válido do agente. O agente funciona como um modelo de agente como serviço (AaaS), usando construções compartilhadas para atender às necessidades do cliente. Em ambos os casos, os consumidores de agentes podem ser aplicativos, sistemas ou até mesmo outros agentes.

Há duas maneiras de analisar o modelo AaaS. O modelo acima oferece a mesma experiência para todos os clientes. Isso significa que os componentes internos do agente não incluirão nenhum nível de especialização que considere o contexto do cliente solicitante. Geralmente, para esse modo, a suposição é que a natureza do escopo, das metas e do valor de um agente gira em torno de um conjunto compartilhado de recursos, conhecimentos e resultados que são aplicados universalmente a todos os clientes.

A abordagem alternativa ao AaaS é quando o contexto dos clientes influencia a experiência e a implementação do agente. O diagrama a seguir fornece uma visão conceitual da presença de um agente AaaS nesse contexto.



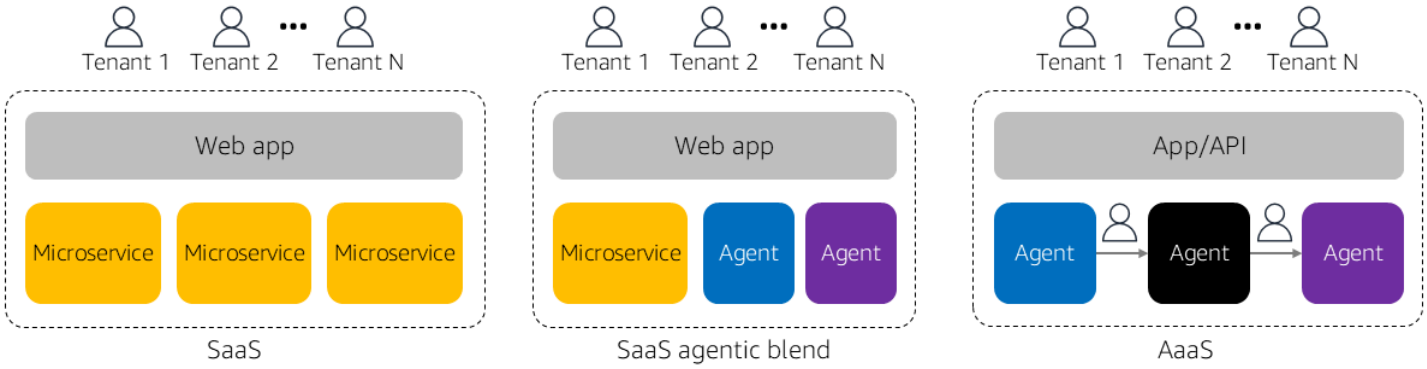
Nessa visão do AaaS, a origem e o contexto das solicitações recebidas afetam significativamente a presença do agente. Os recursos, ações e ferramentas que fazem parte da implementação subjacente do agente podem variar para cada solicitação recebida do inquilino. O valor de um agente está ligado à sua capacidade de usar o contexto do inquilino para chegar a ações e resultados que

são influenciados pelo estado, conhecimento e outros fatores do inquilino. Algumas solicitações podem gerar um resultado exclusivo para cada inquilino, e outras podem levar a resultados mais personalizados por inquilino. Isso adiciona uma nova dimensão à capacidade do agente de aprender, o que pode incluir ser mais contextual e adquirir e aplicar conhecimentos que melhorem os resultados desejados.

Para os provedores, o modelo AaaS oferece muitas vantagens. Com vários clientes consumindo um único agente, o provedor tem uma melhor oportunidade de obter economias de escala, impulsionar a eficiência operacional, controlar os custos e criar uma experiência de gerenciamento unificada. Isso tem o potencial de maior agilidade, inovação e crescimento para o negócio de agentes.

Essas qualidades se sobrepõem aos mesmos princípios que impulsionam a adoção do modelo de software como serviço (SaaS). Essencialmente, o modelo AaaS é construído como um serviço multilocatário que herda muitos dos mesmos atributos de escala, resiliência, isolamento, integração e operação encontrados em um ambiente SaaS. Em muitos aspectos, a experiência de AaaS se baseia fortemente nas estratégias e práticas usadas pelos provedores de SaaS, mas é razoável separar esses termos. Para nossos propósitos, a ênfase está principalmente nas implicações decorrentes dos agentes de construção e operação que exigem suporte multilocatário.

Para um sistema que pode tratar todos os usuários da mesma forma e não exige o gerenciamento de dados persistentes, confidenciais ou específicos do cliente, a noção de locação afetaria minimamente seus agentes. Para sistemas que devem atender a vários clientes e, ao mesmo tempo, preservar o isolamento de dados, a personalização e a percepção do contexto, oferecer suporte a vários locatários pode ser um elemento essencial do design, da estratégia e da meta de um agente. O diagrama a seguir mostra como a multilocação pode ser usada em ambientes agentes.



No lado esquerdo desse diagrama, há uma arquitetura multilocatária clássica. Ele inclui um aplicativo web e uma série de microsserviços que implementam a lógica de negócios. Vários locatários consomem a infraestrutura compartilhada desse ambiente, escalável para atender às mudanças nas

cargas de trabalho de uma população de inquilinos que evolui. O ambiente é operado e gerenciado por meio de um único painel de vidro para todos os inquilinos.

Imagine como esse modelo mental é mapeado para o agente no lado direito desse diagrama. Um agente executa um modelo AaaS que é consumido por um ou mais inquilinos. Os agentes podem ser de vários provedores com o contexto do inquilino fluindo entre eles porque uma única instância de um agente deve processar solicitações de vários inquilinos.

O exemplo no meio desse diagrama é um modelo híbrido em que os agentes fazem parte da experiência geral de SaaS. Algumas partes do sistema são implementadas em um modelo mais tradicional e outras partes do sistema dependem de agentes. É provável que esse padrão seja comum para muitas ofertas de SaaS, especialmente para organizações que estão migrando para uma experiência de agente. É comum que esse modelo persista porque nem todos os sistemas são fornecidos como AaaS puro. Observe também que a multilocação ainda se aplica aos agentes do modelo. Embora os agentes possam estar incorporados em um sistema, eles ainda podem processar solicitações de vários inquilinos.

É natural perguntar se a multilocação realmente importa. Você pode argumentar que um agente processa solicitações, portanto, apoiar a locação pode ter pouco efeito. Porém, à medida que nos aprofundamos nas implicações de agentes multilocatários, a locação pode afetar diretamente a forma como os agentes influenciam a forma como as ferramentas, a memória, os dados e outras partes do agente são acessadas, implantadas e configuradas para oferecer suporte a locatários individuais. A locação também influencia a forma como o dimensionamento, a limitação, os preços, a hierarquização e outros aspectos comerciais se aplicam à arquitetura do seu agente.

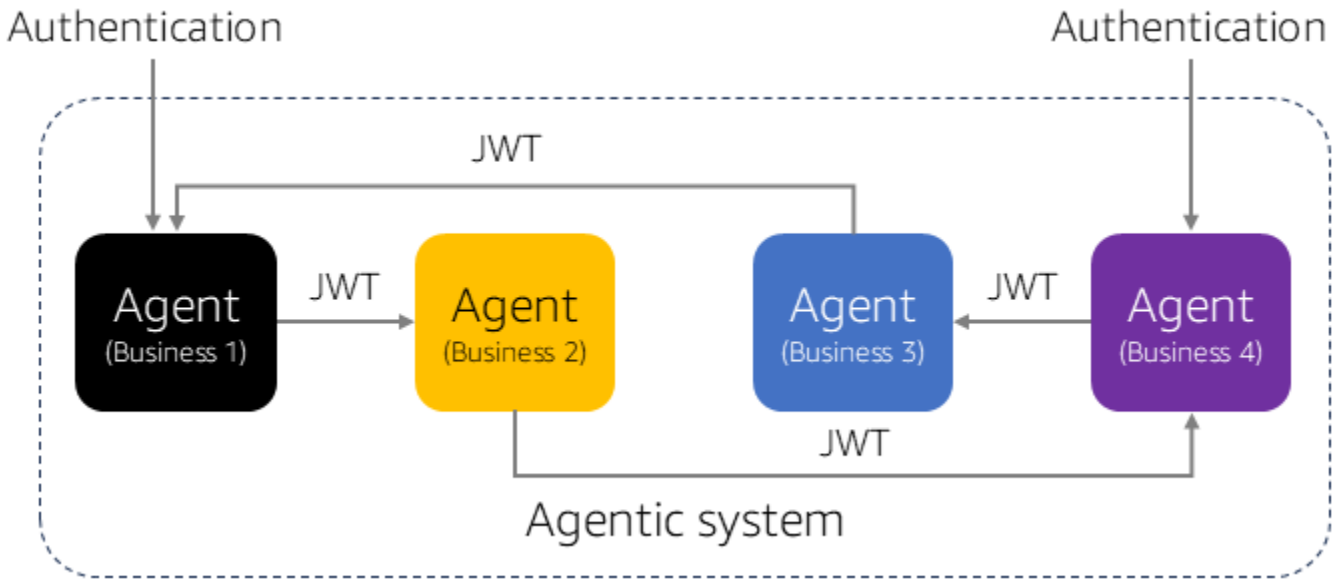
Uma conclusão disso é que existem casos de uso de agentes que exigem suporte multilocatário. O desafio é determinar como a multilocação molda o design geral e a arquitetura de sua experiência como agente. Para alguns agentes, o suporte multilocatário representa uma capacidade diferenciadora, permitindo que os agentes apliquem o contexto específico do inquilino aos agentes que fornecem resultados específicos.

Nas seções subsequentes, você verá como a terminologia e os padrões de design que criamos para descrever arquiteturas SaaS multilocatárias serão úteis. Esses conceitos podem ser adotados pelo modelo AaaS emprestando aspectos úteis, o que introduz novos conceitos específicos do agente onde eles são necessários.

Identidade, contexto do inquilino e sistemas agentes

Adicionar contexto de inquilino a agentes individuais não é particularmente desafiador. Em muitos casos, as equipes podem confiar em mecanismos típicos que vinculam usuários e sistemas aos locatários e passam tokens com reconhecimento de inquilinos aos agentes. Isso é relevante quando consideramos como o contexto e a identidade do inquilino oferecem suporte a vários agentes. Nesse modelo, os inquilinos devem estar vinculados a uma identidade que abranja todos os agentes colaboradores.

Em geral, o domínio agêntico requer um modelo de identidade mais transversal que se alinhe às necessidades atuais e emergentes dos sistemas agentes. Os provedores de agentes exigem mecanismos de identidade que ofereçam suporte a modelos exclusivos de segurança, conformidade e autorização fornecidos com sistemas operacionais. Isso é especialmente desafiador em ambientes em que os sistemas são compostos por clientes ou outros agentes. Cada agente integrado deve conectar sua identidade e o contexto do inquilino às interações do agente. O diagrama a seguir destaca os possíveis desafios de identidade e contexto do inquilino que fazem parte das interações agent-to-agent (a2a).



Este diagrama mostra uma série de agentes criados pelo provedor interagindo como parte do sistema de agentes que abordamos. Agora está adaptado com a identidade e o contexto do inquilino. Esse cenário é um exemplo de um sistema agente que oferece suporte a vários pontos de entrada. Presumimos que cada agente nesse sistema exija seu próprio mecanismo de autenticação para resolver o sistema ou o usuário para um determinado inquilino. À medida que esses agentes interagem, o contexto do inquilino é passado para um token web JSON (JWT) que será usado para autorizar o acesso e injetar o contexto do inquilino no agente.

Conceitualmente, a principal diferença nesse cenário é que os agentes implantam e operam de forma independente, o que significa que cada agente deve ser capaz de resolver sua identidade e autorizar o acesso. A chave é que sua identidade deve ter alguma capacidade distribuída para lidar com as necessidades do sistema agente mais amplo. Também deve haver um alinhamento sobre como os agentes compartilham o contexto do inquilino.

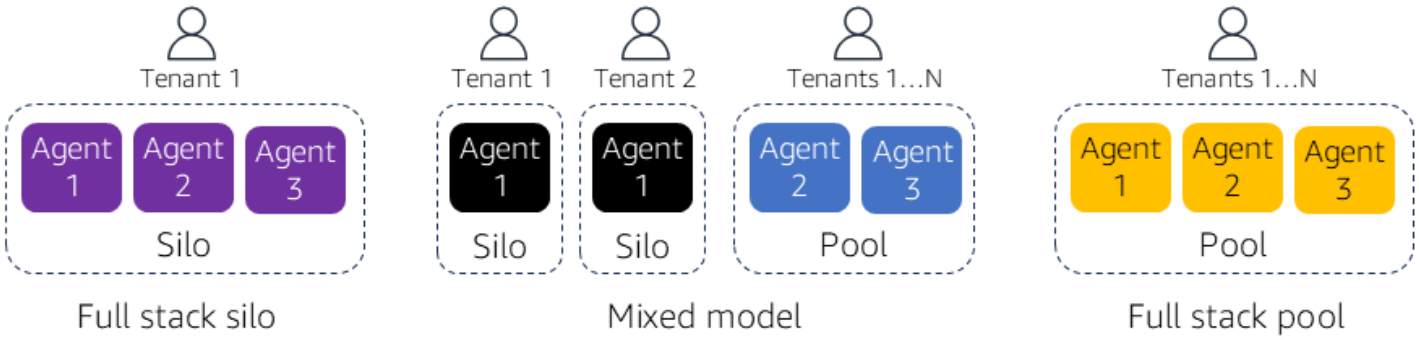
Aplicando valor comercial de SaaS ao AaaS

Geralmente, quando analisamos a execução de qualquer sistema em um as-a-service modelo, consideramos a natureza da experiência e como sua pegada técnica e operacional impulsiona os resultados comerciais. Ao adotar o SaaS, por exemplo, as organizações usam economias de escala, eficiências operacionais, perfis de custo e agilidade para impulsionar o crescimento, as margens e a inovação.

É provável que os agentes entregues como AaaS almejem resultados comerciais semelhantes. Ao oferecer suporte a vários inquilinos, um agente pode alinhar o consumo de recursos às atividades dos inquilinos. Isso gera economias de escala que vêm com os ambientes SaaS tradicionais. O AaaS também permite que as organizações gerenciem, operem e implantem agentes de uma forma que possibilite lançamentos frequentes e aumente a agilidade dos fornecedores de agentes. A chave é que o modelo AaaS não depende da tecnologia. Ela cria e impulsiona estratégias de negócios que promovem o crescimento, agilizam a adoção e simplificam as operações.

Modelos de implantação de agentes

Em uma experiência básica de AaaS, um provedor pode implantar agentes usando vários padrões. Há uma infinidade de fatores que influenciam a forma como os agentes são implantados para atender às necessidades de clientes, desempenho, conformidade, geografia e segurança. Diferentes estratégias de implantação afetam a forma como um agente é projetado, implementado e consumido. É aqui que podemos introduzir termos clássicos de multilocatário para rotular diferentes estratégias de implantação. O diagrama a seguir mostra diferentes permutações para a implantação de agentes em um ambiente AaaS.



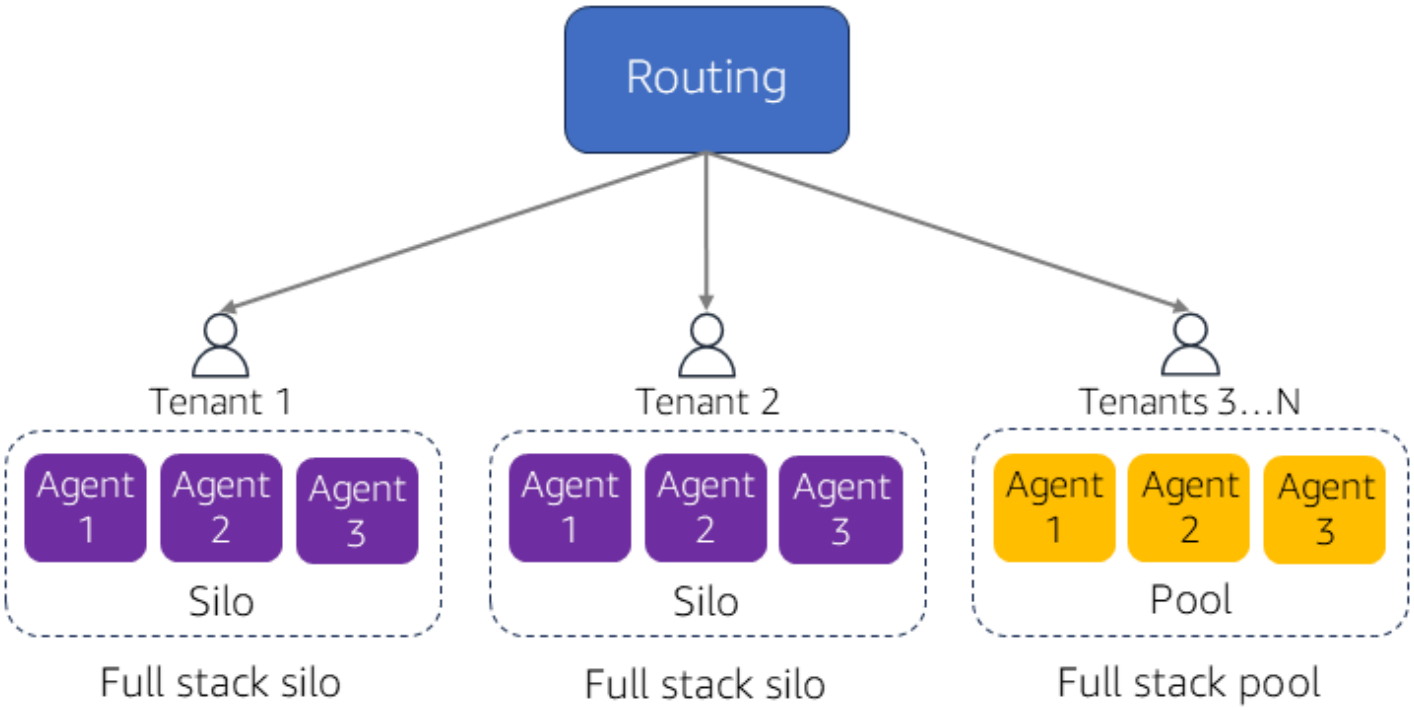
Esse diagrama representa três modos de implantação do agente. No lado esquerdo, há um modelo em silos, em que cada inquilino tem uma experiência totalmente isolada e um conjunto dedicado de agentes. Nesse cenário, os agentes não compartilham ambientes de computação, recursos ou execução entre os locatários.

O exemplo intermediário ilustra um modelo híbrido, em que os inquilinos usam uma combinação de agentes isolados e agrupados. Por exemplo, o Agente 1 é implantado em modo isolado — cada locatário recebe uma instância dedicada — enquanto os agentes 2 e 3 operam em um modelo agrupado, compartilhando recursos entre os locatários.

No lado direito, há um modelo totalmente agrupado, em que todos os agentes são compartilhados entre os locatários, oferecendo uma implantação clássica de vários locatários. Nesse cenário, os locatários utilizam a infraestrutura comum de computação, memória e serviços para a execução do agente.

A ideia é que os agentes possam operar em diferentes modelos de implantação, com recursos computacionais e dependentes dedicados (em silos) ou compartilhados (agrupados) entre os locatários. Essas estratégias de implantação não são mutuamente exclusivas. Os serviços de agentes geralmente atendem a um espectro de necessidades do cliente, combinando os dois modelos para equilibrar desempenho, isolamento, custo e escalabilidade. O diagrama a seguir

mostra um sistema agente que oferece suporte a várias configurações de implantação no mesmo ambiente operacional.



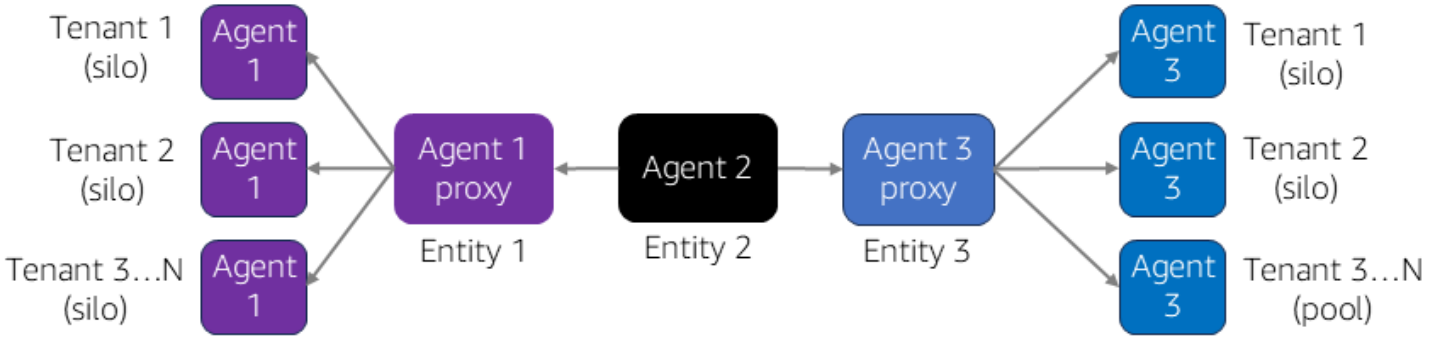
Neste diagrama, um provedor de agentes tem três agentes que são implantados por meio do agente como serviço (AaaS). Eles oferecem suporte a dois tipos de inquilinos. No lado esquerdo, dois inquilinos têm requisitos de conformidade e desempenho que eles atendem por meio de um modelo de silo completo. O inquilino restante no lado direito funciona em um modelo agrupado em que os inquilinos compartilham recursos.

Se o objetivo for agilidade e eficiência operacional, tente limitar os efeitos associados ao suporte a modelos de implantação por locatário. Isso significa implementar o roteamento e outros mecanismos de experiência que permitam que os agentes sejam gerenciados, operados e implantados por meio de um único painel de vidro.

Se você criar um agente em um ambiente com pouco ou nenhum código, não haverá a noção de agentes isolados ou agrupados. Em vez disso, os agentes podem ser totalmente gerenciados por outro agente. Modelos agrupados e em silos se aplicam mais a ambientes em que uma organização controla a construção e a área útil do agente. Nesse caso, as equipes devem considerar qual modelo de implantação oferecer suporte.

Superficialmente, esses modelos de implantação não afetam diretamente o funcionamento de um agente em um sistema mais amplo. Um agente pode não ter conhecimento direto de outros agentes que estão implantados em um silo ou modelo agrupado. Em vez disso, essas estratégias

de implantação podem ser implementadas como parte de uma construção de roteamento em um ambiente. O diagrama a seguir mostra um exemplo de como modelos agrupados e em silos podem ser implementados usando uma estratégia de roteamento.



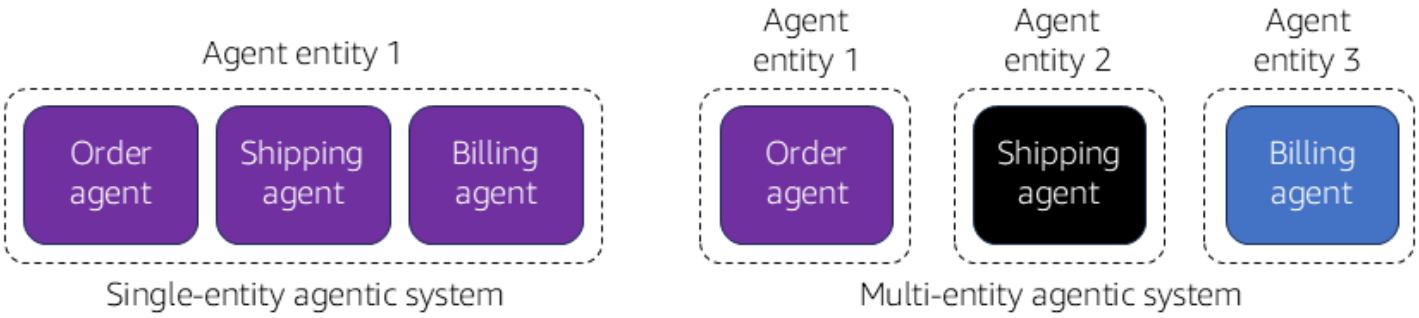
Este exemplo inclui três agentes de três fornecedores diferentes. Cada fornecedor de agentes tem a opção de implementar sua própria estratégia de implantação. Por exemplo, o agente 1 usa um proxy para distribuir solicitações de entrada para um conjunto de agentes inquilinos isolados. O Agente 2 não requer roteamento e oferece suporte a todas as solicitações do inquilino por meio de um agente agrupado. O Agente 3 é uma implantação de modelo híbrido em que alguns inquilinos são isolados e outros agrupados.

Se e como você optar por oferecer suporte a esses modelos de implantação depende da natureza da sua solução. Talvez você não precise oferecer suporte a nenhum dos modelos. No entanto, você pode ter casos em que deve considerar apoiar essa estratégia, como com conformidade, vizinhança ruidosa, desempenho ou hierarquização.

Apresentando e aplicando o contexto do inquilino

Se construirmos agentes que apoiem a multilocação, devemos começar considerando como configurar o contexto do inquilino, que será usado para aplicar políticas, estratégias e mecanismos específicos do inquilino na implementação do agente.

No nível mais básico, você pode introduzir o contexto do inquilino nos agentes por meio das ferramentas e mecanismos comuns que usamos nas arquiteturas clássicas de vários locatários. Isso pode ser feito por meio de uma chave de API ou de vários outros mecanismos de validação. OAuth Muitos exemplos disso se concentram em resolver um sistema ou usuário autenticado em uma chave JSON web token (JWT) que contém o contexto do inquilino. O JWT é então propagado pelo sistema. Isso fica mais interessante quando consideramos como compor sistemas agentes. O diagrama a seguir mostra um exemplo de duas variedades de ambientes agentes.



Neste diagrama, o modelo no lado esquerdo representa um sistema agente em que todos os agentes pertencem, são gerenciados e hospedados por uma única entidade. Quando você tem controle total de toda a experiência, pode usar estratégias típicas para passar os inquilinos por cada agente.

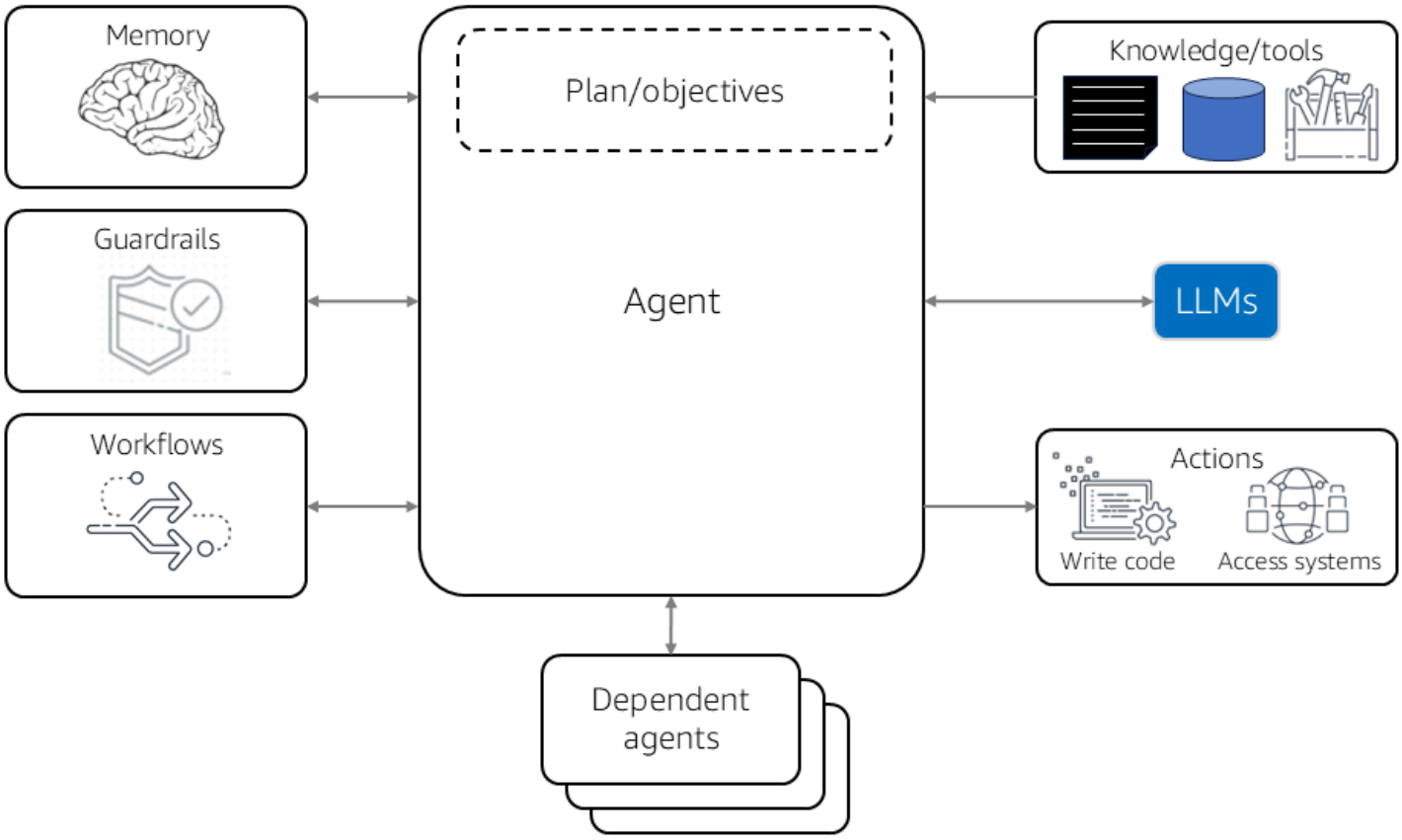
O modelo no lado direito, que pode ser mais comum, representa um sistema de agentes que abrange várias entidades. Os agentes são criados, gerenciados e operados de forma independente, portanto, cada um tem seus próprios esquemas de autenticação e autorização. O desafio aqui é que precisamos de uma maneira universal de resolver e compartilhar o contexto do inquilino entre esses agentes. Isso depende de um modelo mais distribuído em que cada agente deve ser capaz de autenticar sistemas ou usuários e resolvê-los para um inquilino de acordo com os mecanismos aplicados.

Construindo agentes com reconhecimento de inquilinos

A multilocação influencia a forma como implementamos agentes individuais. À medida que um agente processa solicitações, considere como o contexto do inquilino afeta a forma como um agente acessa os dados, toma decisões e invoca ações. Para entender melhor como e onde a multilocação afeta o perfil do seu agente, primeiro determine como as construções podem fazer parte de qualquer agente.

O desafio é que o escopo, a natureza e o design dos agentes são tudo menos concretos porque os fornecedores fazem suas próprias escolhas sobre o design de uma experiência de agente. Em última análise, o objetivo de um agente é que ele é um serviço de aprendizado autônomo que pode acessar uma variedade de ferramentas, fontes de dados e memória para determinar a melhor forma de resolver uma tarefa.

É menos importante saber exatamente quais estratégias e padrões um agente usa. Em um modelo multilocatário, é mais importante identificar como várias partes de um agente são configuradas, acessadas e aplicadas. Considere um ambiente de agente em potencial que depende de uma série de recursos e mecanismos para atingir seus objetivos. O diagrama a seguir mostra um exemplo desse agente.

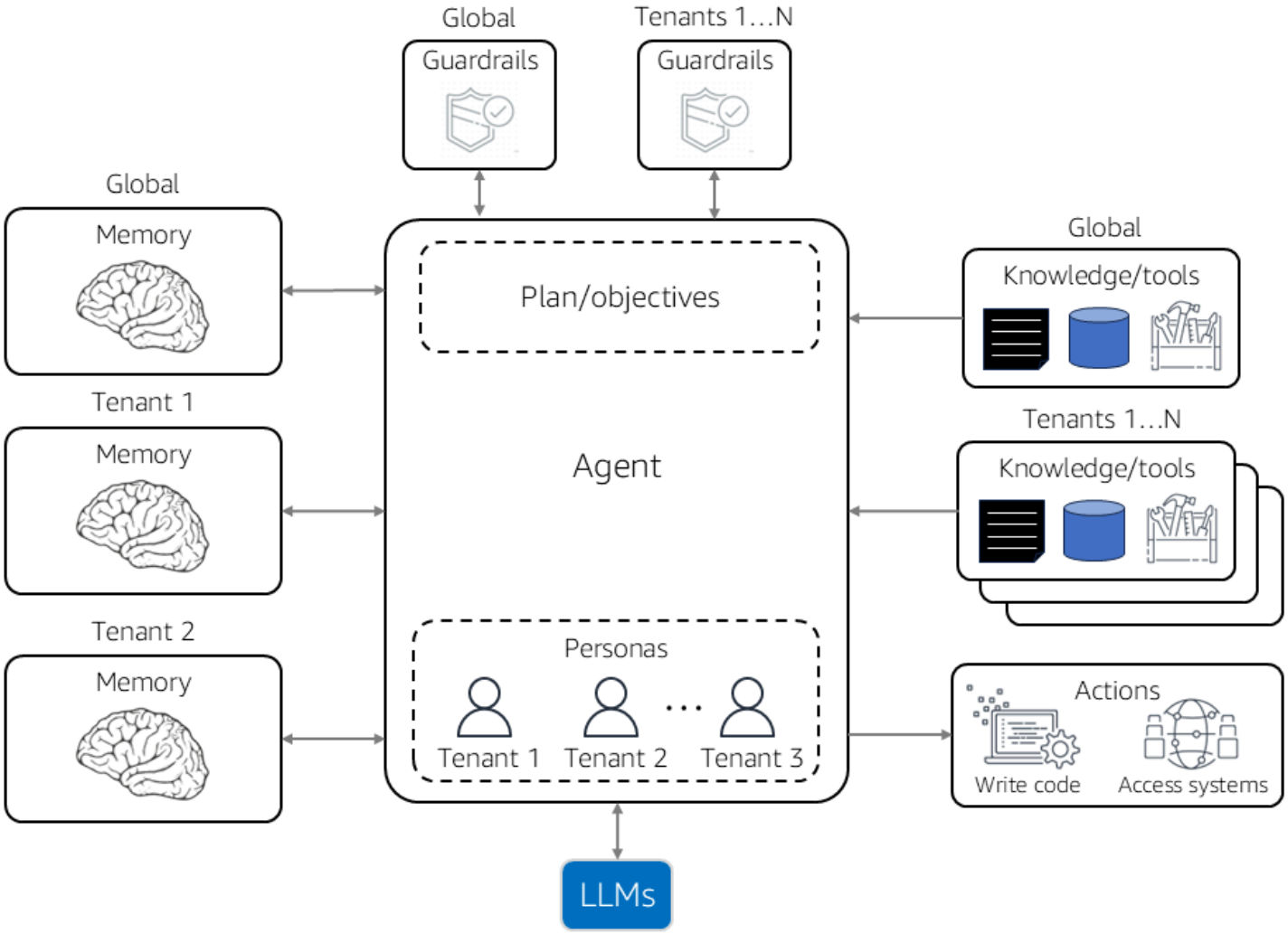


Este diagrama representa uma ampla gama de possibilidades agentes, mostrando várias ferramentas e mecanismos que podem ser combinados para atingir uma meta. No lado esquerdo do diagrama, observe como um agente depende da memória como parte de seu contexto, das grades de proteção para definir as políticas que orientam suas atividades e dos fluxos de trabalho direcionados a tarefas específicas. Alguns podem argumentar que os fluxos de trabalho não devem ser incluídos nesse contexto, mas pode haver cenários em que os fluxos de trabalho sejam parte integrante de uma experiência de agente.

O lado direito do diagrama mostra como informações, como conhecimento e ferramentas, podem fornecer informações e contextos adicionais que aprimoram as capacidades do agente. Em seguida, o agente gera ações, como escrever código ou acessar sistemas. A parte inferior do diagrama mostra como os agentes dependem de um ou mais agentes internos ou terceirizados que podem ser orquestrados como parte de um sistema mais amplo.

Agora podemos pensar no que significa introduzir a multilocação. A locação nos obriga a considerar como e onde um agente introduz estratégias e mecanismos que ditam comportamentos e ações. Isso adiciona outra dimensão à forma como pensamos sobre os agentes em termos de conhecimento, aprendizado, ferramentas e memória.

Vamos agora considerar como modificar esse modelo para oferecer suporte à multilocação. O diagrama a seguir mostra um exemplo de um modelo multiagente.



Neste diagrama, apresentamos as personas de inquilinos que se destinam a moldar a forma como um agente integra o contexto do inquilino. Por exemplo, no lado esquerdo do diagrama, a memória do agente é alterada para oferecer suporte à memória específica do inquilino. O mesmo acontece no lado direito do diagrama, onde o agente oferece suporte ao conhecimento e às ferramentas específicas do inquilino. O mesmo suporte também é aplicado às grades de proteção.

Esse pode ser um exemplo extremo, pois nem todos os aspectos de um agente multilocatário exigem recursos por inquilino. A questão é que você deve considerar como adaptar seu agente para inquilinos específicos pode aumentar sua eficácia. Essa abordagem permite que seu agente aumente seu impacto e valor, forneça um contexto mais relevante em suas respostas e desenvolva capacidades especializadas. O agente será então capaz de aprender, adaptar e realizar tarefas que são exclusivamente adequadas para diferentes personalidades.

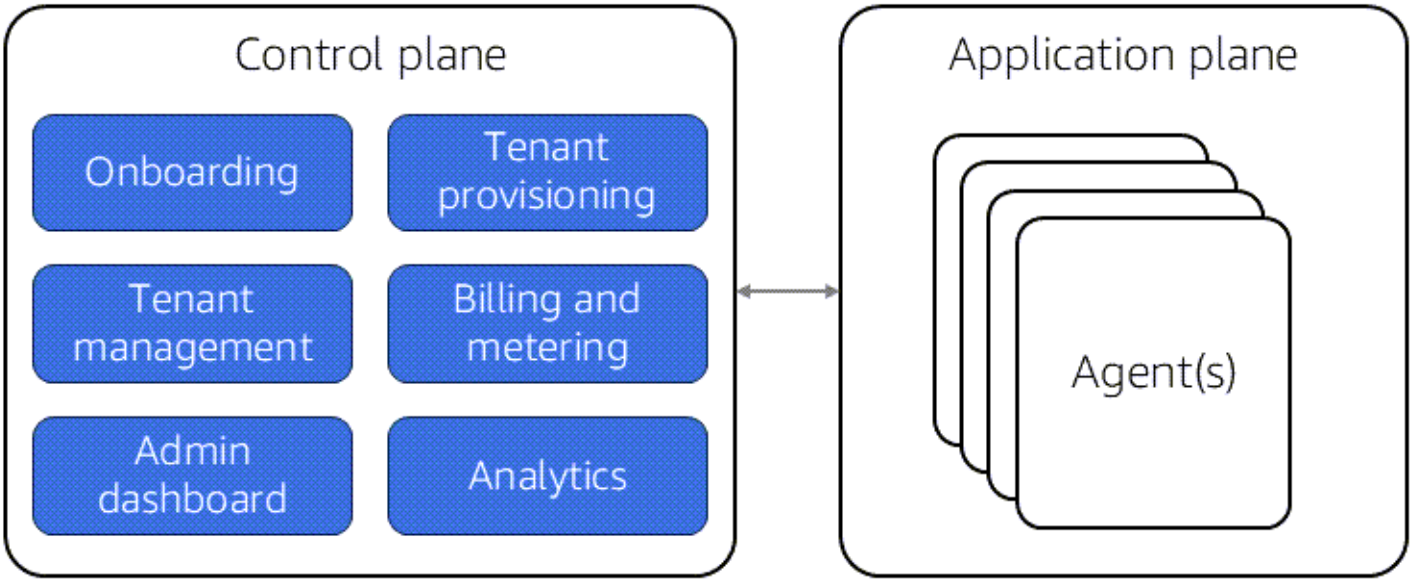
A ideia principal é que o contexto do inquilino afeta diretamente a forma como você cria agentes. Ele também pode moldar as interações dos inquilinos com entidades externas, incluindo outros

agentes. A criação de um agente multilocatário apresenta desafios tradicionais, como vizinhos barulhentos, isolamento de inquilinos, hierarquização, limitação e gerenciamento de custos. O design e a arquitetura do seu agente devem abordar esses conceitos básicos de multilocatário, que exploraremos na próxima seção.

Empregando planos de controle em ambientes agênticos

As melhores práticas multilocatárias geralmente dividem as implementações em duas partes distintas: um plano de controle e um plano de aplicação. O plano de controle fornece um único painel para acessar os mecanismos operacionais, de gerenciamento e de orquestração que abrangem os inquilinos do ambiente. O plano do aplicativo é onde residem a lógica de negócios, os recursos e os recursos funcionais.

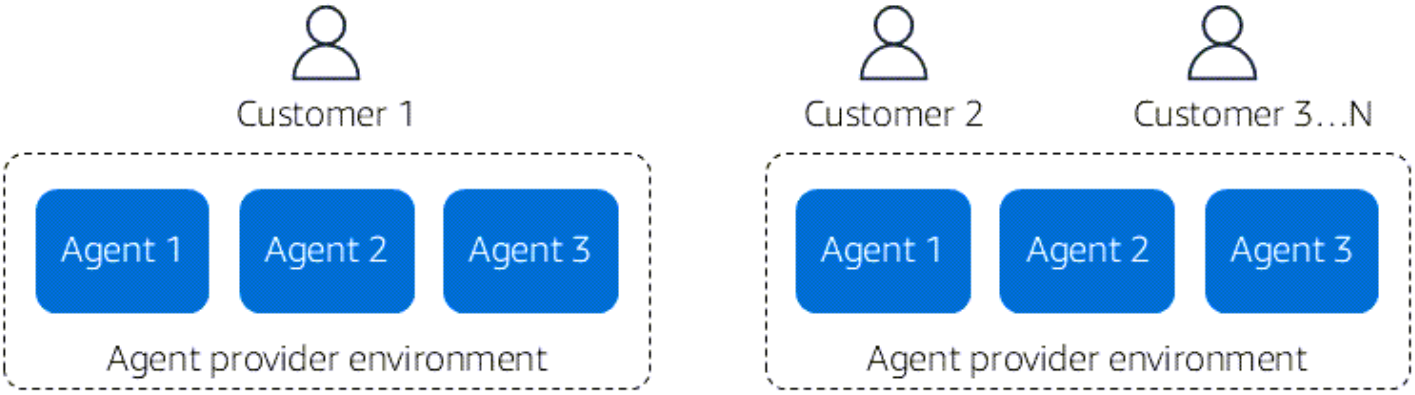
Essa divisão de responsabilidade também se aplica aos modelos agênticos. Um agente multilocatário exige um grau de gerenciamento, operação e insights centralizados, e faz sentido atender continuamente a essas necessidades por meio de um plano de controle. O diagrama a seguir mostra uma visão conceitual de como esses planos são divididos em um ambiente de agente como serviço (AaaS).



Este diagrama mostra a separação tradicional dos planos de controle e aplicação. A novidade é que o plano de controle agora gerencia os agentes que compõem um ambiente AaaS. O plano de controle interage com todos os agentes porque presumimos que os agentes são criados, gerenciados e implantados por um único provedor.

Esse modelo introduz camadas adicionais de complexidade, especialmente no ciclo de vida do agente e na coordenação de terceiros, mas mantém a separação fundamental das preocupações. O plano de controle ainda fornece os mesmos recursos principais ao orquestrar a configuração dos agentes, fornecer observabilidade ao inquilino e ao agente, coletar dados de consumo e medição para cobrança e gerenciar as políticas do inquilino.

Esse cenário se torna mais complexo se você considerar um sistema multiagente que incorpora agentes de vários fornecedores. O diagrama a seguir mostra um exemplo desse modelo.



Este diagrama mostra quatro agentes de diferentes fornecedores que fazem parte de um sistema multiagente. Provedores terceirizados ainda operam e implantam cada agente, que é configurado para permitir o acesso autorizado de um ou mais provedores. Os agentes, no entanto, permanecem sob o controle do provedor, então cada agente mantém seu próprio plano de controle.

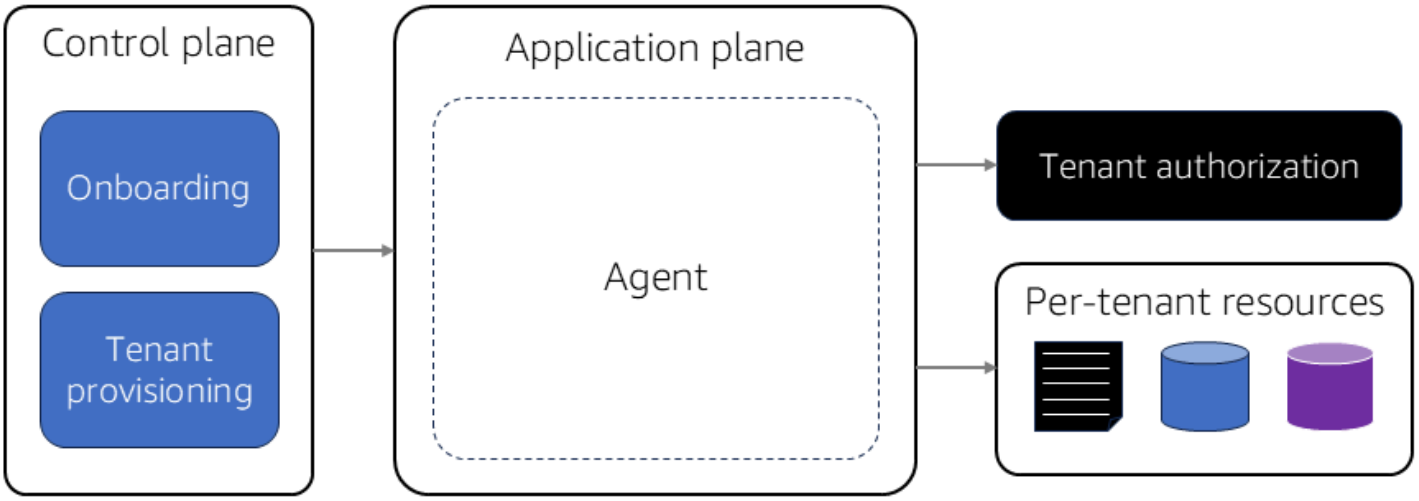
Essencialmente, esses agentes multilocatários se comportam como serviços terceirizados que se integram a outros agentes. Dessa forma, eles devem ter seu próprio plano de controle para fornecer a operação, a configuração e o gerenciamento centralizados dos recursos de um agente.

Presumimos que os agentes são serviços independentes executados em uma experiência hospedada pelo provedor. Mas isso pode não estar claro em um cenário em que um agente consumidor impõe mais restrições sobre como e onde hospedar um agente.

Integração de inquilinos a agentes

Normalmente, a integração é uma parte vital de qualquer ambiente de AaaS. A forma como você cria, configura e provisiona inquilinos geralmente envolve muitas partes móveis, integrações e ferramentas. A experiência de integração do agente pode exigir os mesmos serviços encontrados em um plano de controle AaaS, que inclui identidade do inquilino, hierarquização, provisionamento de recursos por inquilino e configuração de políticas de inquilino.

Sua abordagem à integração de agentes é influenciada pelo modelo de presença e locação de seu ambiente de agência. Cada agente isolado e agrupado tem suas próprias nuances, e a escolha de usar um único agente ou vários agentes também afeta o processo de integração. O diagrama a seguir mostra uma visão conceitual de como a integração afeta a configuração de um agente.



Cada vez que você integra um agente, o plano de controle deve tomar as medidas necessárias para permitir que o inquilino acesse o agente. A forma de apresentar inquilinos varia de acordo com o modelo de autorização do agente, mas suponha que você crie uma identidade de inquilino que associe solicitações de agentes a inquilinos individuais. Esse contexto do inquilino determina a experiência do agente aplicando-a a rotas, escopos e controle de acesso.

A integração também pode exigir que você configure todos os recursos por inquilino usados por um agente. É aqui que o serviço de provisionamento de inquilinos do plano de controle conecta seu agente aos dados e recursos específicos do inquilino que o agente consulta.

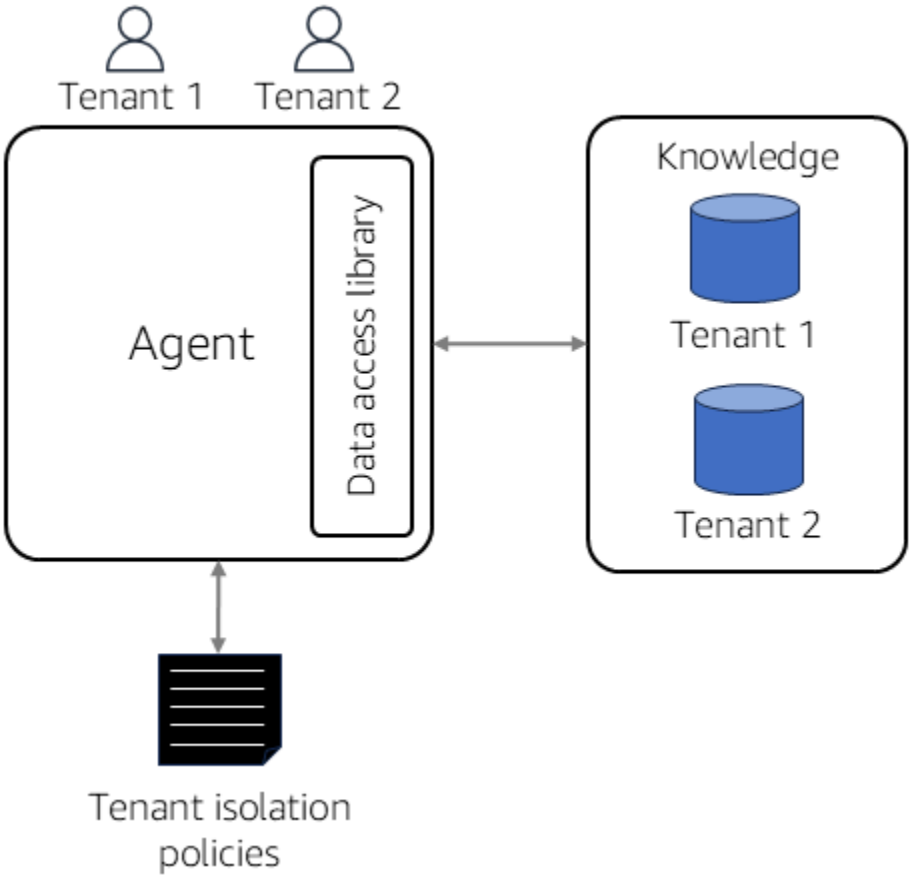
Se o seu sistema depende da integração de agentes terceirizados, você também deve atender às necessidades desses agentes durante o processo de integração. A forma como isso funciona depende dos mecanismos de segurança e integração para autorizar o acesso entre agentes. Idealmente, as etapas necessárias para orquestrar e configurar a agent-to-agent autenticação e a autorização são abordadas por meio da integração automática.

Impondo o isolamento do inquilino

O isolamento de inquilinos é um conceito que se aplica a todas as configurações de vários inquilinos. Isso significa que suas políticas e estratégias garantem que um inquilino não possa acessar os recursos de outros inquilinos. Para agentes multilocatários, talvez seja necessário introduzir construções e mecanismos que ajudem a impor os requisitos de isolamento de locatários de um agente.

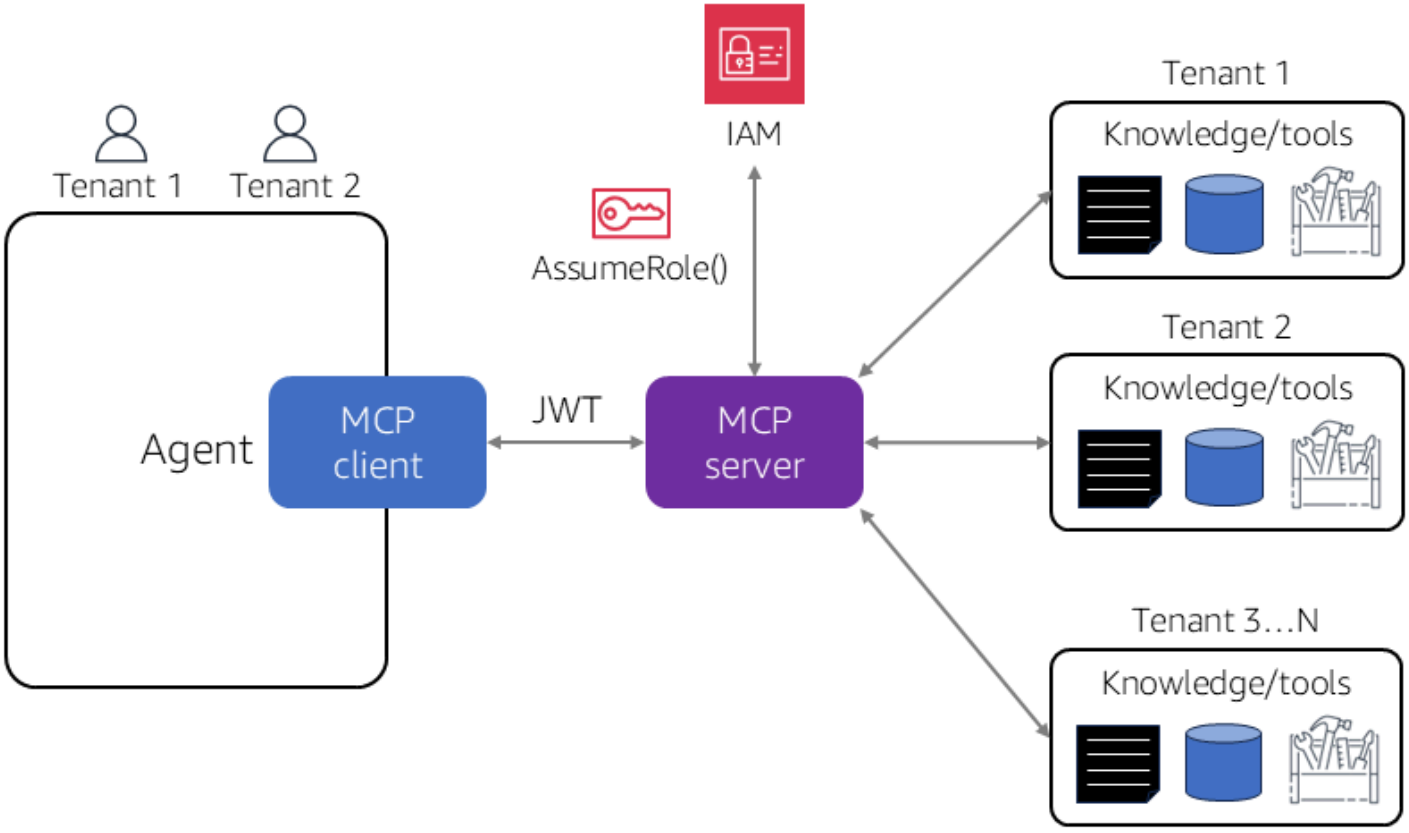
Aplicar o isolamento de inquilinos é como outras estratégias que usam sistemas multilocatários tradicionais. Geralmente, ao criar uma arquitetura AaaS, identifique qualquer área em seu sistema em que uma solicitação ou ação possa acessar recursos para determinar se a solicitação ultrapassa os limites do inquilino. Por exemplo, microsserviços podem ter dependências em tabelas dedicadas por inquilino do Amazon DynamoDB. Isso exige que você introduza políticas que garantam que a tabela de um inquilino não possa ser acessada por outro inquilino.

Nesse caso, considere o isolamento do inquilino por meio de uma lente de agente e suas interações com qualquer um de seus recursos por inquilino. O diagrama a seguir mostra um exemplo conceitual de como os agentes aplicam políticas de isolamento de inquilinos para controlar o acesso aos recursos dos inquilinos.



No lado direito desse diagrama, o agente tem conhecimento por inquilino armazenado em bancos de dados vetoriais separados. Conforme o agente processa uma solicitação, ele examina o contexto do inquilino que fez a solicitação. Com base nisso, o agente aplica uma política de isolamento apropriada para garantir que os inquilinos sejam impedidos de acessar dados ou recursos fora dos limites designados.

Se seu agente usa um Model Context Protocol (MCP), ele também pode implementar seu modelo de isolamento de inquilinos. O diagrama a seguir mostra um exemplo de como introduzir o MCP e aplicar políticas de isolamento.



O MCP é um protocolo padronizado que um agente usa para se integrar com quaisquer ferramentas, dados e recursos. Neste exemplo, um cliente MCP e um servidor MCP interagem com o conhecimento e as ferramentas específicos do inquilino mostrados no lado direito do diagrama. O contexto do inquilino flui do cliente para o servidor, e o servidor usa esse contexto para adquirir credenciais com escopo de inquilino do serviço (IAM). AWS Identity and Access Management As credenciais controlam o acesso aos recursos de cada inquilino, garantindo que um inquilino possa acessar os recursos de outro inquilino.

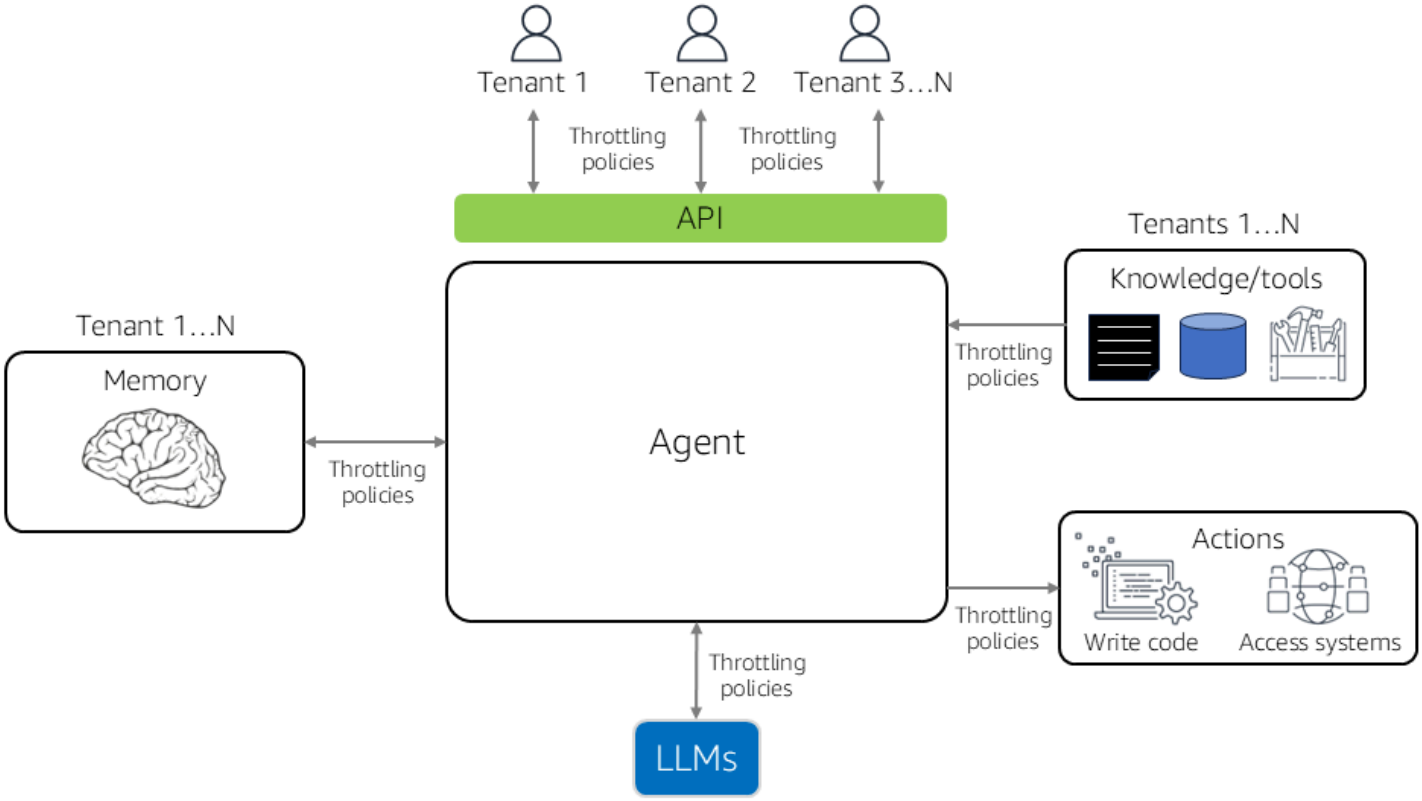
À medida que os agentes incorporam a multilocação, eles devem introduzir mecanismos que apliquem políticas de isolamento de inquilinos à medida que processam as solicitações. Em alguns casos, o IAM pode ajudar a limitar o acesso aos recursos do inquilino. Em outros casos, talvez seja necessário introduzir outras ferramentas ou estruturas para aplicar políticas de isolamento de inquilinos.

Vizinho e agentes barulhentos

Em um ambiente AaaS multilocatário em que vários inquilinos compartilham um agente, pense em onde e como introduzir políticas que evitem condições ruidosas de vizinhos. As políticas podem

introduzir uma limitação de uso geral que se aplica a todo o consumo, ou você pode ter políticas baseadas em inquilinos ou níveis que apliquem a limitação com base em uma determinada pessoa. Você pode impor maiores restrições de consumo aos inquilinos do nível básico do que aos inquilinos do nível premium.

Essa noção de limitação pode ser aplicada em vários pontos da arquitetura. O diagrama a seguir mostra um exemplo de algumas áreas possíveis para introduzir políticas de vizinhança ruidosa.



Em nossa análise anterior da implementação de vários agentes, examinamos diferentes recursos que seu agente pode usar, destacando o potencial de recursos por inquilino em um agente. Cada ponto de contato é uma área em potencial para introduzir políticas de limitação, o que ajuda a garantir que os inquilinos não excedam os limites de consumo do seu sistema ou as políticas de classificação por níveis do inquilino.

Os melhores lugares para introduzir proteções barulhentas para vizinhos são em pontos da arquitetura em que os inquilinos compartilham recursos. Esses componentes compartilhados ou agrupados, como modelos de computação, memória e grandes linguagens APIs, são os mais suscetíveis à degradação do desempenho se um único locatário consumir de forma desproporcional.

Um local natural para aplicar a limitação é no ponto de entrada do agente, às vezes chamado de “borda externa”. Aqui, você pode introduzir limites globais ou tenant-tier-based de taxa antes que o

agente comece a processar a solicitação. A limitação também pode ser aplicada mais profundamente no caminho de execução, como quando o agente chama um LLM, acessa a memória ou invoca ferramentas compartilhadas.

Essas políticas ajudam você a impor o uso justo, manter a resiliência dos agentes sob carga e preservar uma experiência consistente entre os locatários. Dependendo de suas metas, você pode se concentrar na proteção geral do sistema (resiliência) ou no gerenciamento granular da experiência do inquilino (por exemplo, com direitos baseados em níveis).

Dados, operações e testes

Agentes e propriedade de dados

Uma análise da implementação do agente destaca cenários em que um agente depende dos dados de um determinado inquilino. Nesse caso, considere o ciclo de vida dos dados e, mais importante, onde eles são armazenados. Isso é especialmente importante para setores e casos de uso em que a natureza dos dados influencia a forma como um agente os acessa.

Os provedores de AaaS devem avaliar como resolver problemas de dados em um ambiente multilocatário, o que pode afetar a integração, o isolamento e as operações de um agente. As nuances e estratégias aplicáveis variam de acordo com as ferramentas, tecnologias e dados que você consome. Você pode abordar isso de várias maneiras, o que é algo que você deve conhecer ao criar qualquer oferta de AaaS.

Operações de agentes multilocatários

Ao criar ambientes de agentes, pense em como operar e gerenciar seus agentes. Como provedor, você precisa de métricas, dados, insights e registros que permitam monitorar a saúde, a escala e a atividade de um agente. Isso é mais pronunciado em um ambiente de agente multilocatário, no qual você desejará entender como locatários individuais consomem recursos do agente.

Isso é ainda mais significativo em configurações de vários agentes, quando você precisa de insights sobre as interações dos agentes. Ser capaz de traçar o perfil e rastrear atividades entre agentes pode ser essencial para solucionar problemas que afetam a escala, a precisão e a eficácia do seu sistema.

As equipes de operações também podem traçar o perfil das interações do LLM para ter uma melhor noção das cargas que os agentes exercem. LLMs Esses dados são essenciais para a implementação do agente de refino. Também pode dar às equipes operacionais uma visão de como os agentes e a locação afetam o perfil geral de custos de um sistema.

Treinamento e teste de agentes multilocatários

Um desafio associado aos agentes de construção é que se espera que eles aprendam e evoluam. Isso também significa que devemos testar nosso agente, refiná-lo e melhorar sua precisão antes

de colocá-lo em produção. Há muitas áreas em que você pode inspecionar e avaliar se seu agente está avaliando e categorizando corretamente a intenção ou escolhendo e invocando ferramentas e ações apropriadas. A lista de variáveis é extensa, mas, em última análise, trata-se de garantir que seu agente encontre resultados que atinjam suas metas.

Examinar todas as partes móveis e os princípios associados aos agentes de teste está além do escopo deste documento, mas observe que as estratégias de teste aumentam a complexidade dos ambientes de AaaS multilocatários. Por exemplo, se um agente tem dados, memória e outras construções que são aplicadas contextualmente a cada inquilino, os resultados de um agente podem ser moldados pelos recursos por inquilino.

Se você usa um agente para simular um cenário, talvez seja necessário expandir sua simulação para casos de uso específicos do inquilino. Da mesma forma, você deve refinar os procedimentos de validação para permitir casos em que os critérios de validação sejam diferentes para cada inquilino.

Considerações e discussão

Onde o SaaS se encaixa?

Especialistas do setor debatem ativamente sobre como os agentes influenciam o cenário de software como serviço (SaaS). Embora seja verdade que os agentes estão mudando o software de muitos sistemas, é exagero sugerir que os agentes tornem os modelos de entrega obsoletos. Alguns provedores de SaaS provavelmente serão interrompidos pela adoção de agentes, e alguns podem repensar totalmente sua proposta de valor, adotando um modelo de agente como serviço (AaaS). Outros podem encontrar um equilíbrio introduzindo agentes seletivamente para atender a necessidades específicas.

Esse tópico é interessante porque adotar os melhores princípios de SaaS pode representar a próxima evolução do SaaS. Isso pode significar que o SaaS está evoluindo, ou pode significar que os princípios fundamentais do SaaS estão sendo empacotados e implementados em um modelo baseado em agentes. Provavelmente, é menos importante decidir onde a terminologia finalmente chega, mas parece improvável que o SaaS como conceito desapareça. É mais provável que os agentes moldem a presença do SaaS.

Em última análise, devemos decidir quais estratégias podem ser aplicadas ao AaaS, o que significa permitir que as organizações adotem arquiteturas agênticas e estratégias de negócios para que os fornecedores possam maximizar a eficiência, o valor e o impacto de seus sistemas agentes. Agentes não são caixas pretas. Os agentes consomem recursos, escalam as operações, dependem dos dados e geram custos — todos fatores que os fornecedores devem abordar. Os fornecedores de agentes devem avaliar como os princípios de multilocação podem moldar as ofertas de serviços e otimizar os modelos operacionais.

Discussão

O cenário de agências continua evoluindo com designs que variam com base em domínios, casos de uso pretendidos e setores-alvo. Parte dessa evolução inclui refinar ainda mais nossa visão de estratégias, padrões e compensações que os arquitetos consideram ao projetar e construir agentes.

Uma estratégia abrangente de agentes deve estar alinhada aos objetivos comerciais e técnicos. Isso inclui definir mercados-alvo e personas, estabelecer estratégias de preços e gerenciamento de recursos e determinar como os agentes se encaixam em sistemas maiores. Essas considerações

são particularmente importantes ao fornecer AaaS, onde escala, eficiência de custos e inovação são os principais objetivos.

As capacidades operacionais são igualmente importantes. O ambiente deve oferecer suporte ao monitoramento da atividade do agente, das métricas de integridade e dos padrões de uso. Isso se torna mais complexo em sistemas multiagentes, nos quais as operações devem ser coordenadas entre agentes independentes.

No geral, essa discussão sobre agentes apenas esboça as várias considerações arquitetônicas que poderiam fazer parte dos sistemas agentes. Além de selecionar ferramentas e LLMs estruturas adequadas, o sucesso depende da criação de uma arquitetura que atenda aos requisitos comerciais de escalabilidade, eficiência, implantação e multilocação.

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
Publicação inicial	—	14 de julho de 2025

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: migrar seu banco de dados Oracle on-premises 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: migrar seu banco de dados Oracle on-premises 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. Você migra servidores de uma plataforma on-premises para um serviço de nuvem para a mesma plataforma. Exemplo: Migrar um Microsoft Hyper-V aplicativo para o. 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 atributo.

serviços abstraídos

Veja 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 em que 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.

AGGREGATE FUNCTION

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 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 aplicações

Uma abordagem de segurança que permite o uso somente de aplicações aprovadas 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 AIOps é usado na estratégia de AWS migração, 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.

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 malicioso

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

BCP

Veja 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 da aplicação em um ambiente (azul) e a nova versão da aplicação no outro ambiente (verde). Essa estratégia ajuda você a reverter rapidamente com o mínimo de impacto.

bot

Uma aplicação de software que executa tarefas automatizadas na internet e simula a atividade ou interação humana. Alguns bots são úteis ou benéficos, como crawlers da web que indexam informações na internet. Outros bots, conhecidos como bots maliciosos, têm como objetivo causar disrupção ou danos a indivíduos ou organizações.

botnet

Redes de bots infectadas por malware e sob o controle de uma única parte, conhecidas como bot herder ou operador de bots. Os botnets 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 de emergência

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 Implement break-glass procedures nas orientações do AWS Well-Architected.

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

Veja AWS Cloud Adoption Framework.

implantação canário

O lançamento lento e incremental de uma versão para usuários finais. Quando estiver confiante, você implanta a nova versão e substitui a versão atual por completo.

CCoE

Veja Centro de Excelência da Nuvem.

CDC

Veja captura de dados de alteração.

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 integração e 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 em 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 publicações CCo E no blog de estratégia Nuvem AWS corporativa.

computação em nuvem

A tecnologia de nuvem normalmente usada para armazenamento de dados remoto e gerenciamento de dispositivos de IoT. A computação em nuvem é normalmente conectada à tecnologia de computação de borda.

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 ao migrar para a Nuvem AWS:

- Projeto: executar alguns projetos relacionados à nuvem para fins de prova de conceito e aprendizado
- Fundação — Fazer investimentos fundamentais para escalar sua adoção da nuvem (por exemplo, criar uma landing zone, definir um CCo E, 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 de estratégia Nuvem AWS empresarial. Para obter

informações sobre como eles se relacionam com a estratégia de AWS migração, consulte o guia de preparação para migração.

CMDB

Veja 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 o GitHub ou o Bitbucket Cloud. 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 de IA que usa machine learning para analisar e extrair informações de formatos visuais, como vídeos e imagens digitais. Por exemplo, a Amazon SageMaker AI fornece algoritmos de processamento de imagem para CV.

desvio de configuração

Em uma workload, uma alteração de configuração em relação ao estado esperado. Isso pode fazer com que a workload 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. CI/CD é comumente descrito como um pipeline. CI/CD pode ajudá-lo 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.

data mesh

Um framework de arquitetura 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 compatível com business intelligence, como analytics. 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

Veja 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 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 normalmente são 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 (RD)

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

DML

Veja 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 recuperação de desastres.

Detecção da oscilação

Rastreamento de desvios de uma configuração de linha de base. 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 mapeamento do fluxo de valor de desenvolvimento.

E

EDA

Veja análise exploratória de dados.

EDI

Veja intercâmbio eletrônico de dados.

computação de borda

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

intercâmbio eletrônico de dados (EDI)

A troca automatizada de documentos comerciais entre organizações. Para obter mais informações, consulte O que é EDI (Intercâmbio eletrônico de dados)?.

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 endpoint de 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 empresariais (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).

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 CI/CD pipeline, 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

Veja planejamento de recursos empresariais.

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. Ela armazena dados quantitativos sobre as operações comerciais. Normalmente, uma tabela de fatos contém dois tipos de colunas: as que contêm medidas e as que contêm uma chave externa para uma tabela de dimensões.

Antecipar-se à falha

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

delimitação de isolamento contra 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 AWS Fault Isolation Boundaries.

ramificação de recursos

Veja ramificação.

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.

prompt few shot

Fornecer a um LLM um pequeno número de exemplos que demonstram a tarefa e o resultado desejado antes de solicitar que ele execute uma tarefa semelhante. Essa técnica é uma aplicação do aprendizado em contexto, em que os modelos aprendem com exemplos (shots) incorporados aos prompts. Prompts few-shot podem ser eficazes para tarefas que exigem formatação, raciocínio ou conhecimento de domínio específicos. Veja também prompts zero-shot.

FGAC

Veja 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 via captura de dados de alteração para migrar os dados no menor tempo possível, em vez de usar uma abordagem em fases. O objetivo é reduzir ao mínimo o tempo de inatividade.

FM

Veja modelo de base.

modelo de base (FM)

Uma grande rede neural de aprendizado profundo que vem treinando em grandes conjuntos de dados generalizados e não rotulados. FMs são capazes de realizar uma ampla variedade de tarefas gerais, como entender a linguagem, gerar texto e imagens e conversar em linguagem natural. Para obter mais informações, consulte O que são modelos de base?.

G

IA generativa

Um subconjunto de modelos de IA que foram treinados em grandes quantidades de dados e que podem usar um simples prompt de texto para criar novos artefatos e conteúdo, como imagens, vídeos, texto e áudio. Para obter mais informações, consulte O que é IA generativa?.

bloqueio geográfico

Veja 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 trunk-based é a abordagem moderna e preferencial.

golden image

Um snapshot de um sistema ou software usado como modelo para implantar novas instâncias desse sistema ou software. Por exemplo, na manufatura, uma golden image pode ser usada para provisionar software em vários dispositivos e ajudar a melhorar a velocidade, a escalabilidade e a produtividade nas operações de fabricação de dispositivos.

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 governar recursos, políticas e conformidade em todas as unidades organizacionais ()OUs. 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 CSPM, Amazon GuardDuty AWS Trusted Advisor, Amazon Inspector e verificações personalizadas AWS Lambda .

H

HA

Veja 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.

dados de hold-out

Uma parte dos dados históricos rotulados que são retidos de um conjunto de dados usado para treinar um modelo de machine learning. Você pode usar dados de hold-out para avaliar a performance do modelo comparando as predições do modelo com os dados de retenção.

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.

eu

IaC

Veja 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 Internet das Coisas Industrial.

infraestrutura imutável

Um modelo que implanta uma nova infraestrutura para workloads de produção em vez de atualizar, aplicar patches ou modificar a infraestrutura existente. Infraestruturas imutáveis são inerentemente mais consistentes, confiáveis e preditivas do que infraestruturas mutáveis. Para obter mais informações, consulte a prática recomendada Implantar usando infraestrutura imutável no AWS Well-Architected 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 AWS Segurança recomenda configurar sua conta de rede com entrada, saída e inspeção VPCs para proteger a interface bidirecional entre seu aplicativo 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 manufatura por meio de avanços em conectividade, dados em tempo real, automação, analytics 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 industrial das coisas (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 Criando uma estratégia de transformação digital industrial da Internet das Coisas (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 AWS Segurança recomenda configurar sua conta de rede com entrada, saída e inspeção VPCs para proteger a interface bidirecional entre seu aplicativo 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 aprendizado de máquina com AWS.

IoT

Veja 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

Veja biblioteca de informações de TI.

ITSM

Veja 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.

grande modelo de linguagem (LLM)

Um modelo de IA de aprendizado profundo pré-treinado em uma grande quantidade de dados. Um LLM pode realizar várias tarefas, como responder a perguntas, resumir documentos, traduzir texto para outros idiomas e completar frases. Para obter mais informações, consulte O que são LLMs.

migração de grande porte

Uma migração de 300 servidores ou mais.

LBAC

Veja controle de acesso baseado em rótulo.

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.

LLM

Veja grande modelo de linguagem.

ambientes inferiores

Veja 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 ramificação.

Malware

Software projetado para comprometer a segurança ou a privacidade do computador. O malware pode interromper os sistemas do computador, vazar informações sensíveis ou obter acesso não autorizado. Exemplos de malware incluem vírus, worms, ransomware, cavalos de Troia, 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 abstraídos.

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

Veja Programa de Aceleração da Migração.

mecanismo

Um processo completo em que 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 de 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 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 de forma bem definida APIs e normalmente é de propriedade de equipes pequenas e independentes. 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 leveza. APIs 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 migrar para a Nuvem AWS. 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 workload para a Nuvem AWS. Para obter mais informações, veja a entrada 7 Rs neste glossário e consulte Mobilize sua organização para acelerar migrações em grande escala.

ML

Veja machine learning.

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 Strategy for modernizing applications in the 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 Evaluating modernization readiness for applications in the Nuvem AWS.

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.

MPA

Veja Avaliação do Portfólio para Migração.

MQTT

Veja 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 workloads 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 controle de acesso de origem.

OAI

Veja identidade de acesso de origem.

OCM

Veja gerenciamento de alterações 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 integração de operações.

Ola

Veja acordo de 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

Veja Open Process Communications - Unified Architecture.

Open Process Communications - Unified Architecture (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 práticas recomendadas 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 AWS Well-Architected Framework.

tecnologia operacional (TO)

Sistemas de hardware e software que trabalham com o ambiente físico para controlar operações, equipamentos e infraestrutura industriais. Na manufatura, a integração dos sistemas de tecnologia da informação (TI) e tecnologia operacional (TO) 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 exigida 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.

ORR

Veja análise de prontidão operacional.

OT

Veja 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 AWS Segurança recomenda configurar sua conta de rede com entrada, saída e inspeção VPCs para proteger a interface bidirecional entre seu aplicativo 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 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

Veja controlador lógico programável.

PLM

Veja gerenciamento do ciclo de vida do produto.

política

Um objeto que pode definir permissões (veja política baseada em identidade), especificar condições de acesso (veja políta baseada em recurso) ou definir as permissões máximas para todas as contas em uma organização no AWS Organizations (veja 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.

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 cláusula `WHERE`.

pushdown de predicados

Uma técnica de otimização de consultas 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 a performance 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 desenvolvimento.

zonas hospedadas privadas

Um contêiner que contém informações sobre como você deseja que o Amazon Route 53 responda às consultas de DNS para um domínio e seus subdomínios em um 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 desenvolvido para evitar a implantação de recursos não conformes. Esses controles verificam 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 a concepção, o desenvolvimento e o lançamento, passando pelo crescimento e maturidade, até o declínio e a remoção.

ambiente de produção

Veja ambiente.

controlador lógico programável (PLC)

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

encadeamento de prompts

Uso da saída de um prompt do LLM como entrada para o próximo prompt para gerar respostas melhores. Essa técnica é usada para dividir uma tarefa complexa em subtarefas, ou para refinar ou expandir iterativamente uma resposta preliminar. Isso ajuda a melhorar a precisão e a relevância das respostas de um modelo e permite resultados mais granulares e personalizados.

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.

publish/subscribe (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 em que outros microsserviços possam assinar. 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, aprovador, consultado, informado (RACI).

RAG

Veja geração aumentada via recuperação.

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, aprovador, consultado, informado (RACI).

RCAC

Veja 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.

Redefinir arquitetura

Veja 7 Rs.

objetivo de ponto de recuperação (RPO).

O máximo período de tempo aceitável desde o último ponto de recuperação de dados. Isso 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 máximo atraso 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 informações, consulte Specify which Regiões da AWS your account can use.

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 uma aplicação de resistir ou se recuperar de interrupções. Alta disponibilidade e recuperação de desastres são considerações comuns ao planejar a resiliência na Nuvem AWS. Para obter mais informações, consulte Nuvem AWS Resilience.

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.

Retirada

Veja 7 Rs.

Geração Aumentada de Recuperação (RAG)

Uma tecnologia de IA generativa em que um LLM faz referência a uma fonte de dados autorizada que está fora de suas fontes de dados de treinamento antes de gerar uma resposta. Por exemplo, um modelo RAG pode realizar uma pesquisa semântica na base de conhecimento ou nos dados personalizados de uma organização. Para obter mais informações, consulte O que é RAG (geração aumentada via recuperação)?.

alternância

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

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 objetivo de ponto de recuperação.

RTO

Veja objetivo de 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 Console de gerenciamento da AWS 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 política de controle de serviço.

secret

Em AWS Secrets Manager, informações confidenciais ou restritas, como uma senha ou credenciais de usuário, que você armazena de forma criptografada. 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 What's in a Secrets Manager secret? na documentação do Secrets Manager.

segurança desde a concepção

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

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 primários de controles de segurança: preventivos, detectivos, responsivos e proativos.

hardening 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 aplicação de patches em uma instância do Amazon EC2 ou a alternância 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 em AWS Organizations. SCPs defina barreiras ou estabeleça limites nas ações que um administrador pode delegar a usuários ou funções. Você pode usar SCPs como listas de permissão ou listas 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 avaliação de um aspecto de performance de um serviço, como taxa de erro, disponibilidade ou throughput.

objetivo de nível de serviço (SLO)

Uma métrica alvo que representa a integridade de um serviço, conforme avaliado 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 sistema de gerenciamento de eventos e informações de segurança.

ponto único de falha (SPOF)

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

SLA

Veja cacordo de serviço.

SLI

Veja indicador de nível de serviço.

SLO

Veja objetivo de 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 Phased approach to modernizing applications in the Nuvem AWS.

SPOF

Veja ponto único de falha.

esquema em estrela

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 ser usada 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. Cada sub-rede fica alocada em uma única zona de disponibilidade.

controle supervisório 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 a performance. Você pode usar o Amazon CloudWatch Synthetics para criar esses testes.

prompt do sistema

Uma técnica para fornecer contexto, instruções ou orientações a um LLM a fim de direcionar seu comportamento. Os prompts do sistema ajudam a definir o contexto e a estabelecer regras para interações com os usuários.

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 da . 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 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 você pode usar para interconectar sua rede com VPCs a rede local. 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.

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 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 da VPC

Uma conexão entre duas VPCs que permite rotear o 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 backend.

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.

WORM

Veja gravação única e várias leituras.

WQF

Veja AWS Workload Qualification Framework.

gravação única e várias leituras (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, normalmente malware, que tira proveito de uma vulnerabilidade zero-day.

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.

prompt zero shot

Fornecer a um LLM instruções para realizar uma tarefa, mas sem exemplos (shots) que possam ajudar a orientá-lo. O LLM deve usar seu conhecimento pré-treinado para lidar com a tarefa. A

eficácia dos prompts zero-shot depende da complexidade da tarefa e da qualidade do prompt. Veja também prompts few-shot.

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.