Guia do desenvolvedor

AWS Central de Parceiros

AWS Central de Parceiros: Guia do desenvolvedor

Table of Contents

..... vi
Bem-vindo 1
Trabalhando com AWS SDKs 1
Configuração e autenticação 2
Vinculando seu Conta da AWS para o Partner Central 2
Configurar o IAM 3
Autenticação de chamadas de API 3
Assinando suas chamadas com um agente de usuário personalizado 5
Servidor MCP de agentes do Partner Central 7
Visão geral do 7
Capacidades do agente 7
Benefícios principais 8
Exemplos de uso 9
Criação de oportunidades 9
Clonagem de oportunidades 9
Informações sobre o pipeline 10
Resumo da oportunidade 10
Geração de jogos de vendas 10
Criação de perfil de cliente 10
Recomendação de solução 11
Recomendação de financiamento 11
Recomendações da próxima etapa 11
Progressão de oportunidades 12
Experiência agêntica para integração de parceiros 12
Automação do perfil do parceiro 13
Orientação de configuração do vendedor 13
Orientação de conformidade com o PRM 13
Conceitos básicos 14
Introdução 14
Pré-requisitos 14
Etapa 1: configurar permissões do IAM 14
Etapa 2: Conectar seu cliente MCP 21
Assinando suas chamadas com o cabeçalho MCP 23
Etapa 3: verificação da configuração 24

Etapa 4: execução das primeiras tarefas 25
Considerações sobre segurança 27
Próximas etapas 28
Referência da configuração 28
Endpoint 28
Permissões do IAM 29
Ambientes de catálogo 33
Gerenciamento de sessões 34
Upload de arquivo 34
Códigos de erro 35
Limites de taxa 36
Tipos de eventos de streaming (SSE) 36
Próximas etapas 37
Referência de ferramentas 37
Visão geral das ferramentas 37
`Enviar mensagem` 38
`Obter sessão` 46
Tratamento de erros 48
Sobre as APIs 50
Usar as APIs 50
Usando a API da conta 50
Usando a API de vendas 64
Usando a API de benefícios 121
Usando a API Channel 136
Usando a API de medição de receita 138
Regiões aceitas 150
Regiões da API de contas 150
Regiões da API de vendas 150
Regiões da API de benefícios 151
Regiões da API Channel 151
Regiões da API de medição de receita 152
Permissões 153
Acesso à API da conta 153
Acesso à API de vendas 156
Acesso à API de benefícios 162
Acesso à API do canal 165

Acesso à API de medição de receita ... 173
Cotas ... 182
Cotas para a API de contas ... 182
Cotas para a API de vendas ... 186
Cotas para a API de benefícios ... 188
Cotas para a API do canal ... 190
Cotas para a API de medição de receita ... 196
Registro em log ... 197
Registrando a API da conta ... 198
Registrando a API de vendas ... 201
Registrando a API de benefícios ... 203
Registrando a API do canal ... 205
Registrando a API de medição de receita ... 208
Notificações ... 211
Notificações para o AWS API de conta do Partner Central ... 211
Notificações para o AWS API de vendas da Central de Parceiros ... 223
Notificações para o AWS API de benefícios do Partner Central ... 238
Notificações para o AWS API do Partner Central Channel ... 252
Testando em uma sandbox ... 262
Acesso ao ambiente sandbox ... 262
Detalhes importantes sobre o ambiente sandbox ... 262
Testando em um sandbox para a API de contas ... 263
Testando em um sandbox para a API de vendas ... 266
Testando em um sandbox para a API de benefícios ... 270
Testando em um sandbox para a API Channel ... 275
Teste em um sandbox para a API de medição de receita ... 279
Exemplos de código ... 284
Conceitos básicos ... 285
Ações ... 285
Cenários ... 352
Atualizar a entidade associada de uma oportunidade ... 352
Notas da versão ... 358
Histórico do documento ... 391

A referência Central de Parceiros da AWS da API foi reestruturada. Para obter mais informações sobre as operações de API suportadas, consulte a Referência Central de Parceiros da AWS da API.

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.

Bem-vindo ao AWS Guia do desenvolvedor do Partner Central

Este guia do desenvolvedor descreve como AWS os parceiros podem usar as APIs de serviço para integrar e gerenciar atividades de criação, comercialização, venda e expansão com AWS.

Tópicos

- Utilizar AWS API do Partner Central com um AWS SDK
- Configuração e autenticação

Utilizar AWS API do Partner Central com um AWS SDK

AWS kits de desenvolvimento de software (SDKs) estão disponíveis para muitas linguagens de programação populares. Cada SDK fornece uma API, exemplos de código e documentação que permitem que os desenvolvedores criem facilmente aplicações em seu idioma de preferência.

Documentação do SDK	Exemplos de código
AWS SDK para C++	AWS SDK para C++ exemplos de código
AWS CLI	AWS CLI exemplos de código
AWS SDK para Go	AWS SDK para Go exemplos de código
AWS SDK para Java	AWS SDK para Java exemplos de código
AWS SDK para JavaScript	AWS SDK para JavaScript exemplos de código
AWS SDK for Kotlin	AWS SDK for Kotlin exemplos de código
AWS SDK para .NET	AWS SDK para .NET exemplos de código
AWS SDK for PHP	AWS SDK for PHP exemplos de código
AWS Tools for PowerShell	AWS Tools for PowerShell exemplos de código

Documentação do SDK	Exemplos de código
AWS SDK para Python (Boto3)	AWS SDK para Python (Boto3) exemplos de código
AWS SDK for Ruby	AWS SDK for Ruby exemplos de código
AWS SDK para Rust	AWS SDK para Rust exemplos de código
AWS SDK para SAP ABAP	AWS SDK para SAP ABAP exemplos de código
AWS SDK para Swift	AWS SDK para Swift exemplos de código

> **Exemplo de disponibilidade**
>
> Não consegue encontrar o que precisa? Solicite um exemplo de código usando o link Fornecer feedback na parte inferior desta página.

Configuração e autenticação

A configuração e a autenticação com a API do AWS Partner Central envolvem três etapas. Aqui está uma visão geral do processo:

1. Vincule sua conta de AWS Marketplace vendedor ao Partner Central.
2. Configure as permissões usando o IAM.
3. Autentique suas chamadas de API usando o Signature Version 4 (SigV4).

Vinculando seu Conta da AWS para o Partner Central

Vincular seu Conta da AWS ao Partner Central é um pré-requisito para usar a API. Para obter mais informações, consulte Vinculação de contas do AWS Partner Central a contas de AWS Marketplace vendedor. Você deve entrar no Partner Central com uma conta que tenha permissões de líder de aliança ou administrador de nuvem, navegar até a **`Account Linking`** seção e seguir as instruções.

Configurar o IAM

Para usar a API do AWS Partner Central, você precisará de uma função AWS Identity and Access Management(IAM) ou de um usuário do IAM para começar a fazer chamadas. Para obter mais informações, consulte Quando eu uso o IAM? . Siga as etapas para criar funções do IAM e criar um usuário do IAM em seus Conta da AWS guias para isso. Você deve criar esse IAM Role/User na sua conta do Partner Central-linked AWS Marketplace Seller. role/user A criação do IAM não incorre em nenhum custo.

1. Crie um IAM Role/User

   Faça login no AWS Management Console, navegue até o serviço do IAM e siga as etapas para criar uma função do IAM ou um usuário do IAM.
2. Atribuir políticas:

   Anexe políticas gerenciadas ou crie políticas personalizadas conforme necessário. Para modificar ou expandir as permissões, aplique políticas adicionais à função do IAM em vez de copiar e combinar o conteúdo `AWSPartnerCentralOpportunityManagement` com outras permissões. Evite duplicar políticas gerenciadas, pois isso impedirá que você tenha acesso automático aos novos recursos à medida que eles forem lançados, e você precisará atualizar manualmente suas políticas no futuro. Para obter mais detalhes sobre políticas de acesso, consulte a documentação de controle de acesso.

Gerenciamento AWS Marketplace ofertas

Para gerenciar AWS Marketplace ofertas e vinculá-las a oportunidades, os parceiros devem dar permissão à função do IAM para acessar as APIs do catálogo. Certifique-se de role/user que tenha permissões, como `aws-marketplace:ListEntities` `aws-marketplace:SearchAgreements` e.

Autenticação de chamadas de API

AWS A API do Partner Central usa o Signature Version 4 (SigV4) para autenticação. Veja como implementá-lo:

Usar o AWS SDK

AWS Os SDKs lidam automaticamente com a assinatura de solicitações. Forneça suas AWS credenciais e o SDK fará o resto.

1. Para Java, consulte Fornecer credenciais temporárias ao AWS SDK for Java.
2. Para Python (Boto3), consulte Credenciais.
3. Para JavaScript (Node.js), consulte Configuração de credenciais em Node.js.
4. Para o.NET, consulte Resolução de credenciais e perfis.
5. Para outras linguagens de programação e mais exemplos, consulte as Ferramentas para desenvolver AWS.

Autenticação sem usar o AWS SDK

Se um AWS SDK não estiver disponível para a linguagem de programação escolhida, a autenticação envolve a criação manual de uma solicitação canônica, a assinatura da solicitação e o tratamento dos tokens da sessão.AWS oferece orientação abrangente para o uso da assinatura SigV4. No entanto, observe que o uso do AWS SDK é recomendado, pois a assinatura manual de solicitações aumenta a complexidade e exige um gerenciamento cuidadoso dos tokens de segurança.

Cada solicitação do protocolo AWSJSON1_0 DEVE ser enviada para a URL raiz (/) usando o método HTTP “POST” usando os cabeçalhos a seguir.

Cabeçalhos obrigatórios

Cada solicitação de API para o protocolo AWSJSON1_0 DEVE ser enviada para a URL raiz (/) usando o método HTTP “POST” usando os cabeçalhos a seguir.

Cabeçalho	Obrigatório	Description
Content-Type	true	Esse cabeçalho tem um valor estático de `applicati on/x-amz-json-1.0` .
Content-Length	true	O Content-Length cabeçalho padrão definido pela RFC 9110 #section -8.6.
X-Amz-Target	verdadeiro para solicitações	Por exemplo, o valor da operação `CreatePartner` do serviço `PartnerCentralAccount` é `PartnerCe ntralAccount.CreatePartner` .

Assinando suas chamadas com um agente de usuário personalizado

Ao fazer solicitações de API ao AWS Partner Central, recomendamos incluir o `X-Amzn-User-Agent` cabeçalho para ajudar a AWS identificar a origem do aplicativo cliente, monitorar o uso e auditar o desempenho.AWS usa esse cabeçalho para distinguir o tipo de aplicativo cliente que está fazendo a chamada e para coletar informações sobre a taxa de sucesso de diferentes implementações de clientes.

Cabeçalho de agente de usuário personalizado

Nome do cabeçalho: `X-Amzn-User-Agent`

Objetivo: distingue o tipo de cliente que faz a solicitação da API, categorizando a origem da interação.

Formato: `{Integrator}|{Product}`

Exemplo de valor: `AWS|AWS Partner CRM Connector`

Incluir esse cabeçalho em cada solicitação AWS permite analisar padrões de solicitação, rastrear integrações e melhorar a experiência de API para diferentes sistemas clientes.

Usando cabeçalhos personalizados em SDKs

Para incluir o `X-Amzn-User-Agent` cabeçalho nas chamadas do SDK, você pode modificar o comportamento da solicitação do cliente antes de fazer a chamada à API. Abaixo está um exemplo de API de venda usando o AWS SDK para Python (Boto3):

```
import boto3

# Define service and endpoint details
service_name = "partnercentral-selling"
endpoint_url = "https://partnercentral-selling.us-east-1.api.aws"

# Create a boto3 client for Partner Central
partner_central_client = boto3.client(
service_name=service_name,
='us-east-1',
endpoint_url=endpoint_url
)
```

```
# Function to add the custom User-Agent header

def add_version_header(params, **kwargs):
params["headers"]['X-Amzn-User-Agent'] = 'AWS|AWS Partner CRM Connector'

# Register the event to modify the request before the call is made
partner_central_client.meta.events.register(
f'before-call.{service_name}.*', add_version_header
)

# Now, whenever an API call is made using this client, the custom User-Agent header
 will be included
```

Este exemplo demonstra como registrar um evento no SDK do Boto3 para anexar automaticamente o cabeçalho a cada solicitação de API. `X-Amzn-User-Agent` A mesma abordagem pode ser aplicada a outros AWS SDKs modificando seus respectivos mecanismos de interceptação de solicitações.

> **Note**
>
> O formato do `X-Amzn-User-Agent` cabeçalho foi simplificado. Recomendamos usar o novo formato. O formato anterior (`CompanyName|ProductName|CRMName|ProductVersion`) permanece suportado para compatibilidade com versões anteriores. As integrações existentes continuarão funcionando sem modificações.

Servidor MCP de agentes do Partner Central

O servidor MCP dos agentes do Partner Central fornece ferramentas do Partner Central por meio do Model Context Protocol (MCP), permitindo que seus agentes e ferramentas de IA descubram e interajam com o gerenciamento de oportunidades, insights de clientes e programas de financiamento por meio de linguagem natural.

O servidor lida com a autenticação, o gerenciamento de sessões e a aprovação humana em circuito para operações de gravação, mantendo o controle e simplificando os fluxos de trabalho.

Visão geral do

O MCP Server dos agentes do Partner Central é um serviço totalmente gerenciado e AWS hospedado que conecta clientes de MCP-compatible IA aos agentes do Partner Central. Ele usa JSON-RPC 2.0 sobre HTTPS com autenticação SigV4 e oferece suporte a Server-Sent Eventos (SSE) para respostas de streaming em tempo real.

O servidor suporta conversas em vários turnos, anexos de arquivos para análise de documentos e um fluxo de trabalho de aprovação integrado que exige seu consentimento explícito antes da execução de qualquer operação de gravação.

Capacidades do agente

- Insights do pipeline — Obtenha inteligência conversacional sobre seu funil de vendas, incluindo oportunidades em risco, progressão de estágio e análise de perdas fechadas
- Criação de oportunidades — Crie novas oportunidades por meio de conversas em linguagem natural, fazendo upload de notas de reuniões, propostas ou transcrições de chamadas, ou clonando uma oportunidade existente
- Resumo da oportunidade — gere resumos concisos e rápidos de qualquer negócio, abrangendo etapas, gastos, data de fechamento e muito mais
- Geração de oportunidades de vendas — Crie estratégias de vendas personalizadas combinando detalhes de oportunidades, contexto do setor e recomendações de AWS soluções
- Criação de perfil de cliente — Gere perfis de empresas usando informações publicamente disponíveis sobre setor, modelo de negócios, geografia e desenvolvimentos recentes
- Recomendação de solução — Cross-reference suas soluções registradas em relação aos requisitos de oportunidade para encontrar a melhor opção

- Recomendação de financiamento — Avalie oportunidades em relação aos programas de AWS financiamento disponíveis, estime valores e crie solicitações de fundos
- Recomendações da próxima etapa — Obtenha planos de ação priorizados com base em padrões de AWS venda conjunta e orientação de progressão de estágio
- Progressão de oportunidades — faça upload de documentos de apoio, extraia dados relevantes e progrida oportunidades nos estágios do pipeline
- AI-assisted listagem de produtos — gere listagens de AWS Marketplace produtos de alta qualidade a partir de seus ativos digitais existentes, avalie a força da listagem em relação AWS Marketplace aos padrões e receba recomendações em nível de campo para melhorar a capacidade de descoberta. Para obter mais informações, consulte a lista de AI-assisted produtos no Guia do AWS Marketplace vendedor.

Benefícios principais

- Acesso conversacional ao Partner Central — faça perguntas em linguagem natural em vez de navegar por fluxos de trabalho complexos do console
- Mais tempo vendendo — crie oportunidades por meio de uma conversa curta em vez de preencher um formulário de várias etapas, o que reduz a entrada de dados e permite que as equipes de vendas parceiras passem mais tempo vendendo, com o agente recomendando melhorias para que os parceiros enviem oportunidades de maior qualidade e melhorem a higiene dos dutos
- Human-in-the-loop segurança — Todas as operações de gravação exigem sua aprovação explícita antes da execução
- Multi-turn conversas — Refine suas consultas em uma sessão sem repetir o contexto
- Análise de arquivos — Anexe documentos (PDF, DOCX, XLSX, CSV, imagens) para que o agente analise junto com suas perguntas
- Gerenciamento de acesso centralizado — controle o acesso por meio de políticas do IAM com permissões refinadas
- Teste de sandbox — teste fluxos de trabalho em um ambiente sandbox isolado antes de tocar nos dados de produção
- Respostas de streaming — Obtenha feedback via SSE enquanto o agente processa sua solicitação

Exemplos de uso

Todos os AI-generated insights incluem um ID de sessão para rastreabilidade. Os dados são isolados das oportunidades do próprio parceiro conectado. Todo o conteúdo tem rótulos de divulgação claros e é regido pela Política de IA AWS Responsável.

Criação de oportunidades

O agente cria uma nova oportunidade a partir de uma descrição em linguagem natural ou de um documento carregado (PDF, DOCX, XLSX, TXT). Ele extrai o nome do cliente, o escopo do projeto, a data prevista de fechamento e outros campos obrigatórios, enriquece os detalhes do cliente a partir de dados da web disponíveis publicamente, valida o rascunho de acordo com os requisitos de preparação da AWS e cria a oportunidade por meio da API de vendas do Partner Central após a aprovação.

- “Crie uma oportunidade para a Acme Corp: eles querem migrar seu data warehouse para o Redshift, com a data prevista de fechamento no final do terceiro trimestre, com gastos mensais esperados da AWS de 40 mil dólares”
- “Aqui estão minhas notas de ligação da reunião de ontem com GlobalTech — crie uma oportunidade a partir desta transcrição”
- “Use este PDF de proposta para criar uma oportunidade para a migração do SAP do cliente”

Clonagem de oportunidades

O agente cria uma nova oportunidade a partir de uma existente, mescla os detalhes do novo cliente ou projeto que você fornece e valida que pelo menos um campo diferenciador foi alterado antes do envio para que as duplicatas não sejam rejeitadas pela revisão da AWS.

- “Oportunidade de clonagem O1234567890 para um novo cliente — Globex, mesma carga de trabalho, data de fechamento prevista para o final do quarto trimestre”
- “Crie uma oportunidade como a O9876543210, mas para o ambiente de produção do cliente em vez do sandbox”

Informações sobre o pipeline

Obtenha inteligência conversacional sobre seu funil de vendas. O agente analisa a progressão do estágio, os prazos, os negócios paralisados e os padrões de fechamento e perda para revelar o que é mais importante.

- “Quais oportunidades precisam da minha atenção nesta semana?”
- “Quantas oportunidades serão fechadas no próximo mês?”
- “Quais são os principais motivos pelos quais perdemos oportunidades nos últimos 6 meses?”

Resumo da oportunidade

O agente sintetiza o nome da empresa, o setor, o estágio, o AWS gasto mensal esperado, a data prevista de fechamento e outros detalhes importantes em um resumo conciso, sem a necessidade de digitalizar campos individuais do formulário.

- “Me dê um resumo da oportunidade O1234567890"
- “Qual é o status atual e os principais detalhes do acordo com a Acme Corp?”

Geração de jogos de vendas

O agente cria uma estratégia de vendas personalizada combinando os detalhes da oportunidade, o contexto do setor do cliente e as recomendações de AWS soluções relevantes em uma estratégia de vendas pronta para uso.

- “Gere uma jogada de vendas para obter a oportunidade O1234567890"
- “Qual é a melhor abordagem para vender a migração para a nuvem para esse cliente de serviços financeiros?”
- “Crie para mim uma estratégia de vendas para o negócio GlobalTech de análise de dados”

Criação de perfil de cliente

O agente gera um perfil da empresa usando informações publicamente disponíveis — abrangendo classificação do setor, modelo de negócios (B2B/B2C/hybrid), presença geográfica, tamanho da empresa, foco no mercado e desenvolvimentos comerciais recentes. Os perfis são rotulados como “Gerados com dados publicamente disponíveis e insights de AWS IA”.

- “Crie um perfil de cliente para a Acme Corp”
- “O que sabemos sobre o setor e o modelo de negócios desse cliente?”
- “Reúna uma visão geral da empresa para minha próxima reunião com GlobalTech”

Recomendação de solução

O agente faz referência cruzada entre suas soluções registradas e os requisitos da oportunidade, mostrando o nome e a descrição da solução e se ela já está anexada à oportunidade.

- “Quais das nossas soluções melhor correspondem à oportunidade O1234567890?”
- “Nossa solução de análise de dados já está vinculada a esse negócio?”
- “Recomendar soluções para um cliente que deseja migrar suas cargas de trabalho SAP”

Recomendação de financiamento

O agente avalia cada oportunidade em relação aos programas de AWS financiamento disponíveis com base nos detalhes da oportunidade e nos critérios de elegibilidade do programa. Quando uma correspondência é encontrada, ela exibe o nome do programa, a descrição e o raciocínio detalhado. Em seguida, você pode estimar os valores de financiamento, criar rascunhos de solicitação de fundos preenchidos automaticamente ou aprender mais sobre os programas de forma conversacional. A disponibilidade do orçamento do SCA (Acordo de Colaboração Estratégica) aparece quando relevante, e todas as ações respeitam as permissões do IAM.

- “Sou elegível para algum programa de financiamento na oportunidade O6789012345?”
- “Estime o valor do financiamento para um POC com esse cliente”
- “Crie um aplicativo de benefícios MAP para esta oportunidade”

Recomendações da próxima etapa

O agente avalia a oportunidade com base na orientação AWS de progressão de estágio, compara os dados atuais com os critérios de oportunidades bem qualificadas e identifica exatamente quais informações você ainda precisa coletar. O resultado é um plano de ação priorizado baseado em padrões de venda AWS conjunta.

- “O que preciso fazer a seguir para avançar na oportunidade O1234567890?”

- “Essa oportunidade está pronta para ser enviada? Quais campos estão faltando?”
- “Quais são os requisitos para transferir esse negócio de cliente potencial para qualificado?”

Progressão de oportunidades

O agente aceita documentos de apoio (transcrições de reuniões, notas de chamadas, resumos por e-mail), extrai informações relevantes, mapeia-as para campos de oportunidade, avalia os requisitos do estágio e atualiza a oportunidade. Se as lacunas persistirem, ele retornará um detalhamento dos requisitos satisfeitos versus insatisfeitos.

- “Aqui estão minhas notas de chamada — atualize a oportunidade O1234567890 com os detalhes relevantes”
- “Estou anexando a transcrição da reunião. Aproveite esta oportunidade com base no que discutimos.”
- “Analise este resumo do e-mail e me diga quais campos de oportunidade ele satisfaz”

Experiência agêntica para integração de parceiros

O agente automatiza a integração de parceiros no Partner Central e fornece assistência guiada para a configuração do vendedor no Marketplace e a conformidade com o PRM, tudo por meio de linguagem natural.

Capacidades do agente:

- Automação do perfil do parceiro — escaneie seu site, preencha automaticamente seu perfil de parceiro e gerencie a visibilidade, o contato com o líder da aliança, os domínios de treinamento e as conexões da conta
- Orientação de configuração do vendedor — Step-by-step orientação para o registro de vendedores no Marketplace, incluindo formulários fiscais, bancos, conformidade (KYC/BAV/SU), catálogo ESC e funções vinculadas a serviços
- Orientação de conformidade com o PRM — Avalie a prontidão do PRM, recupere códigos de produtos para marcação de receita e verifique as conexões de contas subsidiárias para atribuição consolidada de receita

Automação do perfil do parceiro

O agente escaneia o site da sua empresa para extrair e preencher automaticamente o perfil do seu parceiro e, em seguida, orienta você nas lacunas restantes. Ele pode atualizar campos de perfil, alterar a visibilidade, gerenciar o contato principal da aliança, vincular domínios de certificação de treinamento e lidar com convites de conexão de contas.

- “Você pode ver meu site e preencher meu perfil?”
- “Torne nosso perfil público para que os clientes da AWS possam nos encontrar”
- “Atualize nosso contato principal da aliança para [nome] em [e-mail]”
- “Conecte nossa conta subsidiária na EMEA”

Orientação de configuração do vendedor

O agente percorre o registro de vendedores do Marketplace de ponta a ponta, determinando o formulário fiscal correto com base em seu país e tipo de entidade, explicando a configuração bancária e de desembolso por região e identificando os requisitos de conformidade (KYC, verificação de conta bancária, verificação de usuário secundário) que se aplicam a você.

- “Quero começar a vender no Marketplace — por onde eu começo?”
- “De qual formulário tributário eu preciso? Sou uma empresa na Alemanha”
- “Eu preciso do KYC? Estou vendendo para clientes na França”
- “Como faço para configurar pagamentos se eu estiver fora dos EUA?”

Orientação de conformidade com o PRM

O agente verifica se sua conta está PRM-ready conectada, se existe um anúncio e se o código do produto está disponível para marcação. Para parceiros com contas subsidiárias, ele verifica se todas as contas do vendedor estão devidamente vinculadas para atribuição de receita consolidada.

- “Estou em conformidade com o PRM?”
- “Onde posso encontrar o código do meu produto para marcação?”
- “Diga-me todas as etapas que preciso realizar para estar em conformidade com o PRM”

Conceitos básicos

Pronto para configurar o servidor MCP dos agentes do Partner Central? Acesse o guia de introdução para obter instruções de configuração passo a passo.

Introdução ao Partner Central Agent MCP Server

Este guia explica como configurar o acesso programático ao servidor MCP do Partner Central Agent usando um cliente MCP personalizado. O servidor usa HTTPS direto com autenticação SigV4 — sem necessidade de proxy ou plug-in IDE.

Pré-requisitos

Antes de começar, você deve ter o seguinte:

- Uma conta ativa do Partner Central (migrada para o AWS console)
- Uma AWS conta com permissões do IAM para o Partner Central
- AWS CLI instalada e configurada com credenciais
- Acesso à região us-east-1 (Norte da Virgínia)
- Conectividade HTTPS para `partnercentral-agents-mcp.us-east-1.api.aws`
- Suporte ao TLS 1.2+ em seu cliente HTTP
- Um MCP-compatible cliente que oferece suporte à assinatura de solicitações JSON-RPC 2.0 e SigV4

Etapa 1: configurar permissões do IAM

O servidor MCP do Partner Central Agent exige permissões do IAM em dois níveis: acesso ao protocolo (para se comunicar com o endpoint MCP) e acesso a dados (para realizar operações do Partner Central).

Anexando políticas do IAM

Para anexar uma política à sua identidade do IAM usando o AWS Management Console:

1. Abra o console do IAM.

2. No painel de navegação esquerdo, escolha Usuários, grupos de usuários ou funções, dependendo da identidade à qual você deseja anexar a política e, em seguida, escolha o nome do usuário, grupo ou função específica.
3. Escolha a aba Permissões.
4. Escolha Anexar políticas (ou Adicionar permissões, se for a primeira vez).
5. Na lista de políticas, pesquise e selecione a política gerenciada que você deseja anexar (por exemplo, uma política personalizada criada a partir dos blocos JSON abaixo).
6. Escolha Anexar políticas (ou Avançar e, em seguida, Adicionar permissões) para confirmar.

As permissões entram em vigor imediatamente. Você pode anexar várias políticas à mesma identidade.

Recomendado: use a política gerenciada

A maneira mais simples de conceder acesso ao protocolo MCP é anexar a política `AWSMcpServiceActionsFullAccess` gerenciada à sua identidade do IAM. Essa política inclui todas as permissões necessárias para interagir com o servidor MCP.

Para um controle refinado, você pode usar a chave de `aws:IsMcpServiceAction` condição em suas políticas do IAM para definir o escopo de permissões especificamente para ações de serviço MCP.

Permissões mínimas para acesso ao protocolo MCP

No mínimo, sua identidade do IAM precisa dessa ação para interagir com o servidor MCP:

Ação	Description
`partnercentral:UseSession`	Necessário para criar, atualizar e recuperar sessões de conversação

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "partnercentral:UseSession"
```

```
            ],
            "Resource": "*",
   "Condition": {
                "Bool": {
                    "aws:IsMcpServiceAction": "true"
                }
            }
        }
    ]


}
```

Permissões de acesso a dados

Para realmente realizar as operações do Partner Central por meio do agente, você precisa de permissões adicionais com base no seu caso de uso.

Gerenciamento de oportunidades:

```
{
    "Effect": "Allow",
    "Action": [
        "partnercentral:List*",
        "partnercentral:Get*",
        "partnercentral:CreateOpportunity",
        "partnercentral:UpdateOpportunity",
        "partnercentral:SubmitOpportunity",
        "partnercentral:AssignOpportunity",
        "partnercentral:AssociateOpportunity",
        "partnercentral:DisassociateOpportunity"
    ],
    "Resource": "*"
}
```

Programas de financiamento:

```
{
    "Effect": "Allow",
    "Action": [
        "partnercentral:ListBenefitAllocations",
        "partnercentral:ListBenefitApplications",
        "partnercentral:CreateBenefitApplication",
```

```
        "partnercentral:GetBenefitApplication",
        "partnercentral:UpdateBenefitApplication",
        "partnercentral:SubmitBenefitApplication",
        "partnercentral:AmendBenefitApplication",
        "partnercentral:CancelBenefitApplication",
        "partnercentral:RecallBenefitApplication",
        "partnercentral:AssociateBenefitApplicationResource",
        "partnercentral:DisassociateBenefitApplicationResource"
    ],
    "Resource": "*"
}
```

Acesso ao Marketplace:

```
{
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:DescribeAgreement",
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:ListEntities"
    ],
    "Resource": "*"
}
```

Integração da Central de Parceiros:

```
{
    "Effect": "Allow",
    "Action": [
        "partnercentral:ListPartners",
        "partnercentral:GetPartner",
        "partnercentral:GetProfileVisibility",
        "partnercentral:GetAllianceLeadContact",
        "partnercentral:GetProfileUpdateTask",
        "partnercentral:StartProfileUpdateTask",
        "partnercentral:CancelProfileUpdateTask",
        "partnercentral:PutProfileVisibility",
        "partnercentral:PutAllianceLeadContact",
        "partnercentral:SendEmailVerificationCode",
        "partnercentral:AssociateAwsTrainingCertificationEmailDomain",
        "partnercentral:DisassociateAwsTrainingCertificationEmailDomain",
        "partnercentral:GetAccountConnections",
```

```
        "partnercentral:GetConnectionInvitations",
        "partnercentral:CreateConnectionInvitation",
        "partnercentral:CancelConnectionInvitation",
        "partnercentral:RespondConnectionInvitation",
        "partnercentral:CancelConnection",
        "partnercentral:ManageConnectionPreferences"
    ],
    "Resource": "*"
}
```

Para a configuração do vendedor do Marketplace, adicione também:

```
{
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:StartChangeSet"
    ],
    "Resource": "*"
}
```

Para gerenciamento de funções vinculadas a serviços (revenda Authorizations/CPPO):

```
{
    "Effect": "Allow",
    "Action": [
        "iam:GetRole",
        "iam:CreateServiceLinkedRole"
    ],
    "Resource": "*"
}
```

Política de acesso total

Para desenvolvimento e teste, você pode combinar todas as permissões em uma única política:

```
aws iam create-policy \
    --policy-name PartnerCentralAgentsFullAccess \
    --policy-document '{
        "Version": "2012-10-17",
```

```
        "Statement": [
            {
                "Effect": "Allow",
                "Action": [
                    "partnercentral:UseSession",
                    "partnercentral:List*",
                    "partnercentral:Get*",
                    "partnercentral:CreateOpportunity",
                    "partnercentral:UpdateOpportunity",
                    "partnercentral:SubmitOpportunity",
                    "partnercentral:AssignOpportunity",
                    "partnercentral:AssociateOpportunity",
                    "partnercentral:DisassociateOpportunity",
                    "partnercentral:CreateResourceSnapshot",
                    "partnercentral:CreateResourceSnapshotJob",
                    "partnercentral:StartResourceSnapshotJob",
                    "partnercentral:CreateEngagement",
                    "partnercentral:CreateEngagementInvitation",
                    "partnercentral:RejectEngagementInvitation",
                    "partnercentral:StartEngagementByAcceptingInvitationTask",
                    "partnercentral:StartEngagementFromOpportunityTask",
                    "partnercentral:CreateBenefitApplication",
                    "partnercentral:UpdateBenefitApplication",
                    "partnercentral:SubmitBenefitApplication",
                    "partnercentral:AmendBenefitApplication",
                    "partnercentral:CancelBenefitApplication",
                    "partnercentral:RecallBenefitApplication",
                    "partnercentral:AssociateBenefitApplicationResource",
                    "partnercentral:DisassociateBenefitApplicationResource",
                    "partnercentral:ListPartners",
                    "partnercentral:StartProfileUpdateTask",
                    "partnercentral:CancelProfileUpdateTask",
                    "partnercentral:PutProfileVisibility",
                    "partnercentral:PutAllianceLeadContact",
                    "partnercentral:SendEmailVerificationCode",
                    "partnercentral:AssociateAwsTrainingCertificationEmailDomain",
                    "partnercentral:DisassociateAwsTrainingCertificationEmailDomain",
                    "partnercentral:CreateConnectionInvitation",
                    "partnercentral:CancelConnectionInvitation",
                    "partnercentral:RespondConnectionInvitation",
                    "partnercentral:CancelConnection",
                    "partnercentral:ManageConnectionPreferences"
                ],
                "Resource": "*"
```

```
            },
            {
                "Effect": "Allow",
                "Action": [
                    "aws-marketplace:DescribeEntity",
                    "aws-marketplace:DescribeAgreement",
                    "aws-marketplace:SearchAgreements",
                    "aws-marketplace:ListEntities",
                    "aws-marketplace:DescribeChangeSet",
                    "aws-marketplace:StartChangeSet"
                ],
                "Resource": "*"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "iam:GetRole",
                    "iam:CreateServiceLinkedRole"
                ],
                "Resource": "*"
            }
        ]
    }'
```

Read-only política

Para ambientes de produção ou casos de uso somente para leitura, restrinja as permissões para operações de leitura:

```
aws iam create-policy \
    --policy-name PartnerCentralAgentReadOnly \
    --policy-document '{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Action": [
                    "partnercentral:UseSession",
                    "partnercentral:List*",
                    "partnercentral:Get*",
                    "partnercentral:ListPartners",
                    "partnercentral:GetPartner",
                    "partnercentral:GetProfileVisibility",
```

```
                    "partnercentral:GetAllianceLeadContact",
                    "partnercentral:GetProfileUpdateTask",
                    "partnercentral:GetAccountConnections",
                    "partnercentral:GetConnectionInvitations"
                ],
                "Resource": "*"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "aws-marketplace:DescribeEntity",
                    "aws-marketplace:DescribeAgreement",
                    "aws-marketplace:SearchAgreements",
                    "aws-marketplace:ListEntities",
                    "aws-marketplace:DescribeChangeSet"
                ],
                "Resource": "*"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "iam:GetRole"
                ],
                "Resource": "*"
            }
        ]
    }'
```

Etapa 2: Conectar seu cliente MCP

O Partner Central Agent MCP Server usa HTTPS direto com assinatura de solicitação SigV4. Não há camada de proxy — seu cliente MCP envia solicitações JSON-RPC 2.0 diretamente para o endpoint.

Endpoint

```
https://partnercentral-agents-mcp.us-east-1.api.aws/mcp
```

Autenticação

Todas as solicitações devem ser assinadas com o AWS Signature Version 4 usando:

- Nome do serviço: `partnercentral-agents-mcp`

- Região: `us-east-1`

Inicializar a conexão MCP

Envie uma `initialize` solicitação para estabelecer o protocolo:

```
{
    "jsonrpc": "2.0",
    "id": 1,
    "method": "initialize",
    "params": {
        "protocolVersion": "2025-03-26",
        "capabilities": {},
        "clientInfo": {
            "name": "my-partner-client",
            "version": "1.0.0"
        }
    }
}
```

Resposta esperada:

```
{
    "jsonrpc": "2.0",
    "id": 1,
    "result": {
        "protocolVersion": "2025-03-26",
        "capabilities": {
            "tools": {
                "listChanged": false
            }
        },
        "serverInfo": {
            "name": "PartnerCentralAgentMCPServer",
            "version": "1.0.0"
        }
    }
}
```

Listar as ferramentas disponíveis

```
{
```

```
    "jsonrpc": "2.0",
    "id": 2,
    "method": "tools/list",
    "params": {}
}
```

Assinando suas chamadas com o cabeçalho MCP

Ao fazer solicitações aos agentes MCP do Partner Central, recomendamos incluir o cabeçalho MCP personalizado usando os métodos a seguir para ajudar a AWS identificar a origem do aplicativo cliente, monitorar o uso e auditar o desempenho.AWS usa esse cabeçalho para distinguir o tipo de aplicativo cliente que está fazendo a chamada e para coletar informações sobre a taxa de sucesso de diferentes implementações de clientes.

Método 1: campo _meta (programmatic/stateless MCP)

Para código que constrói diretamente `tools/call` solicitações MCP, forneça o `_meta` campo nas solicitações.

```
{
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "content": [
                {
                    "type": "text",
                    "text": "List my open opportunities with expected close date in Q1
 2026"
                }
            ],
            "catalog": "AWS"
        },
        "_meta": {
            "integrator": "<Integrator's Company Name / Direct>",
            "sourceProduct": "<Integrator's Application Name>"
        }
    }
}
```

Método 2: ClientInfo (agentes personalizados baseados em sessão)

Para clientes MCP personalizados estabelecendo sessões, forneça as informações do cabeçalho MCP dentro do `clientInfo` campo:

```
{
    "method": "initialize",
    "params": {
        "protocolVersion": "2024-11-05",
        "clientInfo": {
            "integrator": "<Integrator's Company Name / Direct>",
            "sourceProduct": "<Integrator's Application Name>"
        }
    }
}
```

Campos em`clientInfo`:

- `integrator`— Nome da empresa ou “Direto” para parceiros

  Exemplo: `AWS`
- `sourceProduct`— Product/agent nome

  Exemplo: `AWS CRM Connector`

Método 3: parâmetro de URL (somente MCP hospedado)

Somente para clientes MCP hospedados nos quais o integrador não pode modificar os campos do protocolo. Use o parâmetro URL:

URL do servidor: `https://mcp.partnercentral.aws?appId=<Integrator's Company Name / Direct>`

Etapa 3: verificação da configuração

Envie uma mensagem simples para confirmar que tudo está funcionando. Use o `Sandbox` catálogo para testar:

```
{
```

```
    "jsonrpc": "2.0",
    "id": 3,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "content": [
                {
                    "type": "text",
                    "text": "Hello, what can you help me with?"
                }
            ],
            "catalog": "Sandbox"
        }
    }
}
```

Se você receber uma resposta `"status": "complete"` e uma resposta de texto do agente, sua configuração está funcionando corretamente. A resposta também incluirá uma `sessionId` que você pode usar para mensagens de acompanhamento.

Etapa 4: execução das primeiras tarefas

Consulte suas oportunidades

```
{
    "jsonrpc": "2.0",
    "id": 4,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
            "content": [
                {
                    "type": "text",
                    "text": "List my open opportunities with expected revenue over
 $50K"
                }
            ],
            "catalog": "AWS"
        }
    }
```

```
}
```

Verifique a elegibilidade do financiamento

```
{
    "jsonrpc": "2.0",
    "id": 5,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
            "content": [
                {
                    "type": "text",
                    "text": "Am I eligible for MAP funding for opportunity
 O1234567890?"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Recupere o histórico da sessão

```
{
    "jsonrpc": "2.0",
    "id": 6,
    "method": "tools/call",
    "params": {
        "name": "getSession",
        "arguments": {
            "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
            "catalog": "AWS"
        }
    }
}
```

Integração de parceiros

```
{
```

```
    "jsonrpc": "2.0",
    "id": 7,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
            "content": [
                {
                    "type": "text",
                    "text": "Help me onboard to Partner Central"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Outras tarefas de integração para experimentar:

- “Oriente-me pelo processo de entrevista tributária”
- “Você pode acessar meu site e preencher meu perfil de parceiro?”
- “O que eu ainda preciso fazer para estar pronto para vender no Marketplace?”

Considerações sobre segurança

- Não passe AWS credenciais pelos parâmetros da ferramenta MCP. A autenticação é feita pela assinatura da solicitação SigV4 na camada de transporte.
- Use o catálogo do Sandbox para testes e desenvolvimento. O `"Sandbox"` catálogo fornece um ambiente isolado que não afeta os dados do parceiro de produção.
- Aplique políticas de IAM com privilégios mínimos na produção. Use a política de somente leitura para monitorar e relatar casos de uso. Conceda permissões de gravação somente quando o usuário precisar atualizar oportunidades ou enviar solicitações de financiamento.
- Revise as operações de gravação com cuidado. O servidor usa a aprovação human-in-the-loop para todas as operações de gravação. Quando uma ação de gravação for proposta, revise os parâmetros antes de aprová-la.
- Os dados da sessão são transitórios. As sessões expiram 48 horas após a criação. Não confie em sessões para armazenamento de dados a longo prazo.

- Os uploads de arquivos vão para um bucket temporário do S3. Os arquivos enviados são armazenados temporariamente e não são retidos permanentemente. Não faça upload de arquivos contendo credenciais, segredos ou outras informações confidenciais.

Próximas etapas

- Referência de configuração — Referência completa para endpoint, ações do IAM, gerenciamento de sessões e códigos de erro
- Referência de ferramentas — documentação detalhada `sendMessage` e `getSession` ferramentas

Referência da configuração

Esta página fornece a referência completa para conexão e configuração do Partner Central Agent MCP Server.

Endpoint

Propriedade	Valor
URL	`https://partnercentral-agents-mcp.us-east-1.api.aws/mcp`
Região	`us-east-1` Apenas (Norte da Virgínia)
Protocolo	JSON-RPC 2.0 sobre HTTPS
Streaming	Server-Sent Eventos (SSE)
Autenticação	AWS Signature versão 4
Nome do serviço SigV4	`partnercentral-agents-mcp`
Requisito de TLS	TLS 1.2 ou superior

Permissões do IAM

Todas as ações do IAM usam o `partnercentral:` prefixo, salvo indicação em contrário.

Acesso ao protocolo MCP

A maneira mais simples de conceder acesso ao protocolo MCP é anexar a política `AWSMcpServiceActionsFullAccess` gerenciada à sua identidade do IAM. Essa política inclui todas as permissões necessárias para interagir com o servidor MCP. Para um controle refinado, você pode usar a chave de `aws:IsMcpServiceAction` condição em suas políticas do IAM para definir o escopo de permissões especificamente para ações de serviço MCP.

Ação	Description
`partnercentral:UseSession`	Crie, atualize e recupere sessões de conversação

Gerenciamento de oportunidades

Ação	Description
`partnercentral:List*`	Listar oportunidades, soluções e outros recursos do Partner Central
`partnercentral:Get*`	Recupere detalhes de oportunidades individuais e outros recursos
`partnercentral:CreateOpportunity`	Crie uma oportunidade
`partnercentral:UpdateOpportunity`	Modifique os campos de oportunidade (estágio, data de fechamento, receita etc.)
`partnercentral:SubmitOpportunity`	Envie uma oportunidade para AWS análise
`partnercentral:AssignOpportunity`	Atribua uma oportunidade a um representante parceiro

Ação	Description
`partnercentral:AssociateOpportunity`	Vincule uma oportunidade a outro recurso (solução etc.)
`partnercentral:DisassociateOpportunity`	Remover um link entre uma oportunidade e outro recurso
`partnercentral:StartEngagementFromOpportunityTask`	Envie oportunidades para iniciar um engajamento a partir de uma oportunidade

Programas de financiamento

Ação	Description
`partnercentral:ListBenefitAllocations`	Liste as alocações de benefícios disponíveis para sua conta
`partnercentral:ListBenefitApplications`	Listar os pedidos de benefícios enviados
`partnercentral:CreateBenefitApplication`	Crie um novo aplicativo de benefícios (MAP, POC, WMP)
`partnercentral:GetBenefitApplication`	Recupere detalhes de um pedido de benefício específico
`partnercentral:UpdateBenefitApplication`	Atualizar um pedido de benefício pendente
`partnercentral:AssociateBenefitApplicationResource`	Vincular um recurso a um aplicativo de benefícios
`partnercentral:DisassociateBenefitApplicationResource`	Remover um link de recurso de um aplicativo de benefícios

Acesso ao Marketplace

Ação	Description
`aws-marketplace:DescribeEntity`	Recuperar detalhes de uma entidade de mercado
`aws-marketplace:SearchAgreements`	Pesquise contratos de mercado
`aws-marketplace:ListEntities`	Listar entidades do mercado

Agente de integração

Gerenciamento de contas de parceiros (**`partnercentral`**):

Ação	Description
`partnercentral:ListPartners`	Listar registros de parceiros para a conta
`partnercentral:GetPartner`	Recupere o registro do parceiro e os detalhes do perfil
`partnercentral:GetProfileVisibility`	Recuperar a visibilidade do perfil atual () PUBLIC/PRIVATE
`partnercentral:GetAllianceLeadContact`	Recupere as informações de contato do líder da aliança
`partnercentral:GetProfileUpdateTask`	Verifique o status de uma atualização pendente do perfil assíncrono
`partnercentral:StartProfileUpdateTask`	Enviar uma atualização do perfil do parceiro (assíncrona)
`partnercentral:CancelProfileUpdateTask`	Cancelar uma tarefa pendente de atualização de perfil
`partnercentral:PutProfileVisibility`	Alterar a visibilidade do perfil para PÚBLICO ou PRIVADO

Ação	Description
`partnercentral:PutAllianceLeadContact`	Atualizar contato do líder da aliança (nome, cargo, e-mail)
`partnercentral:SendEmailVerificationCode`	Envie um código de verificação para alterações baseadas em e-mail domain/contact
`partnercentral:AssociateAwsTrainingCertificationEmailDomain`	Vincule um domínio de e-mail para acompanhamento de certificações de treinamento
`partnercentral:DisassociateAwsTrainingCertificationEmailDomain`	Remover um domínio de e-mail de certificação de treinamento vinculado
`partnercentral:GetAccountConnections`	Listar conexões de contas ativas (por exemplo, links subsidiários)
`partnercentral:GetConnectionInvitations`	Listar convites de conexão enviados e recebidos pendentes
`partnercentral:CreateConnectionInvitation`	Enviar um convite de conexão para outra AWS conta
`partnercentral:CancelConnectionInvitation`	Cancelar um convite de conexão de saída pendente
`partnercentral:RespondConnectionInvitation`	Aceitar ou rejeitar um convite de conexão recebido
`partnercentral:CancelConnection`	Cancelar uma conexão de conta ativa
`partnercentral:ManageConnectionPreferences`	Obtenha ou atualize as preferências de compartilhamento entre contas conectadas

Configuração do vendedor do Marketplace (**`aws-marketplace`**):

Ação	Description
`aws-marketplace:ListEntities`	Listar entidades do Marketplace (por exemplo, entidades vendedoras)
`aws-marketplace:DescribeEntity`	Recupere todos os detalhes de uma entidade do Marketplace
`aws-marketplace:DescribeChangeSet`	Verifique o status de um conjunto de alterações enviado
`aws-marketplace:StartChangeSet`	Envie um conjunto de alterações do Marketplace (por exemplo, atualização do perfil do vendedor)

IAM — função vinculada ao serviço ()**`iam`**:

Ação	Description
`iam:GetRole`	Verifique se a função vinculada ao serviço Marketplace Resale Authorization existe
`iam:CreateServiceLinkedRole`	Criar o perfil do `AWSServiceRoleForMarketplaceResaleAuthorization`

Ambientes de catálogo

Catálogo	Description
`"AWS"`	Ambiente de produção. Todas as operações afetam os dados ativos do parceiro.
`"Sandbox"`	Ambiente de teste isolado. Sem impacto nos dados de produção. Use para desenvolvimento e validação.

Gerenciamento de sessões

Propriedade	Valor
Formato de ID da sessão	UUID v4 com `session-` prefixo (por exemplo,) `session-550e8400-e29b-41d4-a716-446655440000`
Criação de sessão	Automático na primeira `sendMessage` chamada sem um `sessionId`
Expiração da sessão	48 horas a partir da criação da sessão (expiração absoluta, não baseada em inatividade)

Upload de arquivo

Propriedade	Valor
Máximo de arquivos por mensagem	3
Limite de tamanho de imagem	3,75 MB
Limite de tamanho de documentos	4,5 MB
Extensões permitidas	`doc`, `docx`, `pdf`, `png`, `jpeg`, `xlsx`, `csv`, `txt`
Caçamba S3 (produção)	aws-partner-central-marketplace-efemeral-writeonly-files
Caminho de upload	s3://{bucket}/{aws-account-id}/
Requisito de URI do S3	Deve incluir o parâmetro `versionId` de consulta

Carregar fluxo de trabalho

1. Faça upload do arquivo no bucket do S3 apropriado com o prefixo de ID AWS da conta.
2. Observe o URI do S3, incluindo o `versionId` retornado pelo upload.
3. Inclua o arquivo como um bloco de `document` conteúdo na sua `sendMessage` chamada.

Exemplo de formato de URI S3:

```
s3://aws-partner-central-marketplace-ephemeral-writeonly-files/123456789012/my-document.pdf?versionId=abc123
```

Códigos de erro

Código	Name (Nome)	Description
-32001	AUTHENTICATION_FAILURE	A assinatura SigV4 é inválida ou as credenciais expiraram
-31004	TOOL_PERMISSION_DENIED	A identidade do IAM não tem a `partnercentral:` ação necessária para essa operação
-32002	ACCESS_DENIED	Acesso geral negado (conta não cadastrada, incompatibilidade de região etc.)
-32004	LIMIT_EXCEEDED	Limite de taxa ou cota excedido. Tente novamente com um recuo exponencial.
-30001	RESOURCE_NOT_FOUND	O recurso solicitado (sessão, oportunidade etc.) não existe
-32600	INVALID_REQUEST	JSON-RPC Solicitação malformada ou parâmetros inválidos

Código	Name (Nome)	Description
`-32603`	`INTERNAL_ERROR`	Server-side erro. Repetir a solicitação .

Limites de taxa

O servidor impõe limites de taxa por conta:

Operation	Taxa sustentada	Intermitência
`sendMessage`	2 solicitações por minuto	10
Todas as outras operações	10 solicitações por minuto	20

Se você exceder esses limites, o servidor retornará o código de erro `-32004` (LIMIT_EXCEEDED). Implemente um recuo exponencial com instabilidade em seu cliente para lidar com a limitação de taxas com elegância.

Tipos de eventos de streaming (SSE)

Quando `stream` definido como `true` em uma `sendMessage` chamada, o servidor retorna Server-Sent Eventos com estes tipos de eventos:

Tipo de evento	Description
`stream_start`	Conexão de transmissão estabelecida
`assistant-response.start`	O agente começou a gerar uma resposta
`assistant-response.delta`	Fragmento de texto incremental da resposta do agente
`assistant-response.completed`	O agente terminou de gerar a resposta
`server-tool-use`	O agente está invocando uma ferramenta interna (operação de leitura)

Tipo de evento	Description
`server-tool-response`	Resultado de uma invocação de ferramenta interna
`tool_approval_request`	O agente está solicitando aprovação humana para uma operação de gravação
`stream_end`	Fechamento da conexão de transmissão
`done`	Evento final indicando que o fluxo SSE foi concluído

Próximas etapas

- Referência de ferramentas — documentação detalhada `sendMessage` e `getSession` ferramentas

Referência de ferramentas

O Partner Central Agent MCP Server expõe duas ferramentas MCP: `sendMessage` para todas as interações do agente e `getSession` para recuperar o estado da sessão. Todas as operações do Partner Central — consultas de oportunidades, solicitações de financiamento, análise de documentos — são tratadas em linguagem natural por meio de. `sendMessage`

Visão geral das ferramentas

Ferramenta	Description	Categoria
`sendMessage`	Envie mensagens para o agente de IA do Partner Central. Oferece suporte a texto, anexos de arquivos e respostas de aprovação humanas.	Leitura/Gravação

Ferramenta	Description	Categoria
`getSession`	Recupere o estado da sessão, incluindo histórico de conversas, eventos e metadados.	Read-only

`Enviar mensagem`

Ferramenta principal para todas as interações com agentes de IA do Partner Central. Use essa ferramenta para fazer perguntas, solicitar ações, anexar documentos para análise e responder às solicitações de aprovação para operações de gravação.

O agente mantém o contexto da conversa em uma sessão, para que você possa fazer perguntas complementares sem repetir o contexto anterior.

Parâmetros

- `content`(obrigatório) — Matriz de blocos de conteúdo. Cada bloco deve incluir um `type` campo que determine a estrutura do bloco. Você pode incluir vários blocos em uma única mensagem (por exemplo, texto + anexo de documento).

  Tipos de blocos de conteúdo:

Tipo	Campos	Description
`text`	`type`(obrigatório), `text` (obrigatório)	Texto da mensagem do usuário enviado ao agente
`document`	`type`(obrigatório), `filename` (obrigatório), `s3Uri` (obrigatório)	Anexo de arquivo para o agente analisar. `s3Uri`É necessário incluir um `versionId` parâmetro.
`tool_approval_response`	`type`(obrigatório), `toolUseId` (obrigatório),	Resposta a uma solicitação de aprovação feita por humanos

Tipo	Campos	Description
	`decision` (obrigatório), `message` (opcional)	

- `catalog`(obrigatório) — Ambiente de destino para a operação.

  Valores válidos: `"AWS"` (produção), `"Sandbox"` (teste)

- `sessionId`(opcional) — UUID v4 identificando uma sessão existente para continuar. Omita a criação de uma nova sessão. Formato: `session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.

  Padrão: uma nova sessão é criada automaticamente.

- `stream`(opcional) — Ative o streaming de Server-Sent eventos (SSE) para entrega de respostas em tempo real.

  Valores válidos: `true`, `false`

  Padrão: `false`

Resposta

A resposta inclui:

Campo	Description
`sessionId`	Identificador de sessão para mensagens de acompanhamento
`status`	Status da resposta:`"complete"` ,`"requires_approval"` , ou `"error"`
`content`	Conjunto de blocos de conteúdo de resposta do agente

Exemplos

Mensagem de texto básica (nova sessão)

Solicitação:

```
{
    "jsonrpc": "2.0",
    "id": 1,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "content": [
                {
                    "type": "text",
                    "text": "List my open opportunities with expected close date in Q1
 2026"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Resposta:

```
{
    "jsonrpc": "2.0",
    "id": 1,
    "result": {
        "content": [
            {
                "type": "text",
                "text": "I found 12 open opportunities with expected close dates in Q1
 2026. Here's a summary:\n\n1. **O1234567890** - Acme Corp Cloud Migration - $250,000 -
 Qualified stage\n2. **O1234567891** - GlobalTech Data Analytics - $180,000 - Prospect
 stage\n..."
            }
        ],
        "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
        "status": "complete"
    }
```

```
}
```

Follow-up mensagem (sessão existente)

Solicitação:

```
{
    "jsonrpc": "2.0",
    "id": 2,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "content": [
                {
                    "type": "text",
                    "text": "Tell me more about O1234567890. Is it ready for
 submission?"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Anexo de arquivo

Faça o upload de um documento para o S3 primeiro e, em seguida, faça referência a ele na mensagem:

```
{
    "jsonrpc": "2.0",
    "id": 3,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "content": [
                {
                    "type": "text",
```

```
                    "text": "Review this customer proposal and suggest which
 opportunity it aligns with"
                },
                {
                    "type": "document",
                    "filename": "acme-proposal.pdf",
                    "s3Uri": "s3://aws-partner-central-marketplace-ephemeral-writeonly-
files/123456789012/acme-proposal.pdf?versionId=abc123def456"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Restrições de upload de arquivos:

- Máximo de 3 arquivos por mensagem
- Limite de tamanho da imagem: 3,75 MB
- Limite de tamanho do documento: 4,5 MB
- Extensões permitidas: `doc`docx,pdf,png,,jpeg,xlsx,csv, `txt`
- Os arquivos devem ser enviados para `s3://{bucket}/{your-aws-account-id}/`
- O URI do S3 deve incluir o parâmetro de `versionId` consulta

Human-in-the-loop fluxo de trabalho de aprovação

Quando o agente precisa realizar uma operação de gravação (por exemplo, atualizar uma oportunidade, enviar uma solicitação de financiamento), ele retorna um `"requires_approval"` status com os detalhes da ação proposta. Você deve responder com um bloco `tool_approval_response` de conteúdo.

Etapa 1 — O agente solicita aprovação:

```
{
    "jsonrpc": "2.0",
    "id": 4,
    "result": {
        "content": [
            {
```

```
                "type": "text",
                "text": "I'd like to update opportunity O1234567890 with the following
 changes:\n- Target close date: 2026-03-31\n- Expected revenue: $300,000\n- Stage:
 Qualified\n\nPlease approve, reject, or override this action."
            },
            {
                "type": "tool_approval_request",
                "toolUseId": "tool-use-98765",
                "toolName": "update_opportunity_enhanced",
                "parameters": {
                    "opportunityId": "O1234567890",
                    "targetCloseDate": "2026-03-31",
                    "expectedRevenue": 300000,
                    "stage": "Qualified"
                }
            }
        ],
        "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
        "status": "requires_approval"
    }
}
```

Etapa 2 — Aprovar a ação:

```
{
    "jsonrpc": "2.0",
    "id": 5,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "content": [
                {
                    "type": "tool_approval_response",
                    "toolUseId": "tool-use-98765",
                    "decision": "approve"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Etapa 2 (alternativa) — Rejeite a ação:

```
{
    "jsonrpc": "2.0",
    "id": 5,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "content": [
                {
                    "type": "tool_approval_response",
                    "toolUseId": "tool-use-98765",
                    "decision": "reject",
                    "message": "The expected revenue should be $250,000, not $300,000"
                }
            ],
            "catalog": "AWS"
        }
    }
}
```

Etapa 2 (alternativa) — Substituir por uma resposta personalizada:

```
{
    "jsonrpc": "2.0",
    "id": 5,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "content": [
                {
                    "type": "tool_approval_response",
                    "toolUseId": "tool-use-98765",
                    "decision": "override",
                    "message": "Use expected revenue of $250,000 and keep the stage as
 Prospect instead"
                }
            ],
            "catalog": "AWS"
```

```
        }
    }
}
```

Valores da decisão de aprovação:

Decisão	Comportamento
`"approve"`	Execute a ferramenta com os parâmetros propostos
`"reject"`	Não execute a ferramenta. Opcional `message` explica o porquê.
`"override"`	Forneça uma resposta personalizada ou instruções modificadas via `message`

Streaming com SSE

Ative o streaming para receber fragmentos de respostas incrementais à medida que o agente processa sua solicitação:

Solicitação:

```
{
    "jsonrpc": "2.0",
    "id": 6,
    "method": "tools/call",
    "params": {
        "name": "sendMessage",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "content": [
                {
                    "type": "text",
                    "text": "Analyze my pipeline and identify opportunities at risk"
                }
            ],
            "catalog": "AWS",
            "stream": true
```

```
        }
    }
}
```

O servidor responde com um fluxo de eventos SSE:

```
event: stream_start
data: {"sessionId": "session-550e8400-e29b-41d4-a716-446655440000"}

event: assistant-response.start
data: {}

event: server-tool-use
data: {"toolName": "analyze_pipeline", "parameters": {}}

event: server-tool-response
data: {"toolName": "analyze_pipeline", "result": {"opportunitiesAnalyzed": 47,
 "atRisk": 5}}

event: assistant-response.delta
data: {"text": "I analyzed your pipeline of 47 opportunities and identified "}

event: assistant-response.delta
data: {"text": "5 that are at risk of slipping:\n\n"}

event: assistant-response.delta
data: {"text": "1. **O2345678901** - Close date is past due by 15 days\n"}

event: assistant-response.completed
data: {"status": "complete"}

event: stream_end
data: {}
```

Obter sessão

Recupere o estado atual de uma sessão de conversação, incluindo histórico completo de conversas, eventos e metadados. Use isso para inspecionar o estado da sessão, analisar interações passadas ou retomar uma conversa.

Parâmetros

- `sessionId`(obrigatório) — UUID da sessão a ser recuperada. Formato: `session-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.
- `catalog`(obrigatório) — Ambiente ao qual a sessão pertence.

  Valores válidos: `"AWS"`, `"Sandbox"`

Resposta

Campo	Tipo	Description
`sessionId`	string	Identificador da sessão
`createdAt`	string	Registro de data e hora ISO 8601 da criação da sessão
`lastActivity`	string	Registro de data e hora ISO 8601 da última atividade
`sequenceNumber`	integer	Número de sequência do evento atual
`stateType`	string	Estado atual da sessão
`events`	array	Histórico completo de conversas (mensagens de usuários, respostas de agentes, usos de ferramentas)
`variables`	objeto	Variáveis de sessão e metadados
`eventCount`	integer	Número total de eventos na sessão

Exemplo

Solicitação:

```
{
    "jsonrpc": "2.0",
    "id": 7,
    "method": "tools/call",
    "params": {
        "name": "getSession",
        "arguments": {
            "sessionId": "session-550e8400-e29b-41d4-a716-446655440000",
            "catalog": "AWS"
        }
    }
}
```

Resposta:

```
{
    "jsonrpc": "2.0",
    "id": 7,
    "result": {
        "content": [
            {
                "type": "text",
                "text": "{\"sessionId\":\"session-550e8400-e29b-41d4-
a716-446655440000\",\"createdAt\":\"2026-01-15T10:30:00Z\",\"lastActivity\":
\"2026-01-15T11:45:00Z\",\"sequenceNumber\":8,\"stateType\":\"END_TURN\",\"eventCount
\":8,\"events\":[...],\"variables\":{}}"
            }
        ]
    }
}
```

Tratamento de erros

Todos os erros seguem o formato de erro JSON-RPC 2.0:

```
{
    "jsonrpc": "2.0",
    "id": 1,
```

```
    "error": {
        "code": -32001,
        "message": "Authentication failed. Verify your SigV4 credentials and ensure
 they have not expired."
    }
}
```

Consulte the section called “Códigos de erro” a lista completa dos códigos de erro e seus significados.

Estratégia de nova tentativa recomendada

- Para `-32004` (LIMIT_EXCEEDED): tente novamente com recuo exponencial a partir de 1 segundo
- Para `-32603` (INTERNAL_ERROR): tente novamente até 3 vezes com recuo exponencial
- Para `-32001` (AUTHENTICATION_FAILURE): atualize as credenciais e tente novamente
- Para todos os outros erros: não tente novamente automaticamente — inspecione a mensagem de erro e corrija a solicitação

Sobre o AWS APIs do Partner Central

As seções a seguir fornecem informações gerais sobre as APIs.

Tópicos

- Usar o AWS API do Partner Central
- Compatível AWS Regiões
- Permissões
- Cotas para o AWS API do Partner Central
- Registro para o AWS API do Partner Central
- Notificações para o AWS API do Partner Central
- Testando em uma sandbox

Usar o AWS API do Partner Central

As seções a seguir fornecem informações sobre como usar as APIs do AWS Partner Central.

Tópicos

- Usar o AWS API de conta do Partner Central
- Usar o AWS API de vendas da Central de Parceiros
- Usar o AWS API de benefícios do Partner Central
- Usar o AWS API do Partner Central Channel
- Usando a API de medição de receita

Usar o AWS API de conta do Partner Central

Gerenciando a conta do parceiro

As contas de parceiros são registradas e vinculadas a uma AWS conta existente para proporcionar uma IAM-based experiência completa. Essa abordagem elimina a necessidade de credenciais em nível de usuário, garantindo que todos os registros e operações sejam gerenciados por meio da entidade IAM dentro da conta.AWSAWS O Partner Central permite uma integração perfeita com

AWS os serviços, e esse modelo fornece gerenciamento de entidades parceiras, oferece suporte a vários catálogos e estabelece um sistema de direitos em nível de AWS conta.

Os parceiros podem utilizar as APIs de contas de parceiros disponíveis para registrar, gerenciar e atualizar suas contas:

API de registro de parceiros

Os parceiros podem registrar suas contas usando a AWS conta do Partner Engagement. Usando a API Create, os parceiros fornecerão as informações necessárias sobre o líder da aliança, aceitarão os termos da APN e fornecerão um nome legal exclusivo.

API de perfil de parceiro

Os parceiros gerenciam perfis contendo detalhes da empresa (nome, descrição, site, logotipo) com controle de public/private visibilidade, validação assíncrona e rastreamento de status, permitindo uma atualização por vez.

API de gerenciamento de domínio

Os parceiros podem registrar e verificar domínios comerciais por meio da validação de e-mail para estabelecer a identidade organizacional e associar o treinamento e as certificações dos funcionários.

API de conexão de parceiros

Os parceiros conectam suas AWS contas a outras AWS contas para várias finalidades, como colaborar na venda conjunta com outros parceiros, compartilhar dados do AWS Marketplace Offers entre contas, autorizar distributors/channel parceiros a distribute/re vender seus produtos e muito mais.

API de verificação de parceiros

A verificação do parceiro é um pré-requisito obrigatório para o registro da conta do parceiro. Antes que os parceiros possam criar uma conta de parceiro, eles devem concluir os processos de verificação comercial e de identidade. Essas verificações validam que a empresa está legalmente registrada e confirmam a identidade da pessoa que está registrando a conta.AWS

Tópicos

- Trabalhando com o registro de parceiros
- Trabalhando com o perfil do parceiro

- Trabalhando com gerenciamento de domínio
- Trabalhando com conexões de contas de parceiros
- Trabalhando com a verificação de parceiros

Trabalhando com o registro de parceiros

Registro de parceiros

Os parceiros podem registrar suas contas usando a AWS conta do Partner Engagement. Usando a API Create, os parceiros fornecerão as informações necessárias sobre o líder da aliança, aceitarão os termos da APN e fornecerão um nome legal exclusivo.

Durante o registro

1. Criação síncrona da entidade parceira — Os clientes iniciam o processo de registro do parceiro chamando a API Create, que realiza a validação da entrada e cria a entidade parceira na mesma solicitação.
2. Requisito de informações do líder da aliança — Durante o registro do parceiro, o cliente deve fornecer informações de contato do líder da aliança para comunicação relacionada à Rede de AWS Parceiros (APN).
3. Aplicação dos termos e condições — os clientes devem aceitar os Termos e Condições da APN durante o registro. A aceitação é necessária para concluir o registro e acessar todos os recursos. Os termos se aplicam a todos os benefícios e funcionalidades do programa APN.
4. Tag-On-Create Support — Como parte da Experiência de Autorização AWS Consistente (CAE) e da Tag-Based Autorização, os clientes podem adicionar tags no momento da criação da entidade parceira.
5. Name-Based Validação legal — O registro reforçará a exclusividade com base no nome legal do parceiro, garantindo que não haja registros duplicados na mesma entidade.

Registro postal

1. Suporte ao gerenciamento de tags — após o registro, os parceiros podem marcar, desmarcar e listar as tags de uma entidade parceira existente.
2. Buscar entidade parceira - Depois que uma entidade parceira é criada, os clientes podem usar a API List para recuperar a ID do parceiro e, em seguida, usar a API Get para buscar os detalhes de uma entidade parceira específica.

Resumo de APIs

1. CreatePartner API: os clientes iniciam o processo de registro do parceiro chamando a CreatePartner API, que realiza a validação da entrada e cria a entidade parceira na mesma solicitação.
2. ListPartners API: após o registro, os clientes podem usar a API List para recuperar a lista de entidades parceiras.
3. GetPartner API: após o registro, os clientes podem usar a API Get para recuperar detalhes de uma entidade parceira específica.
4. PutAllianceLeadContact API: atualiza as principais informações de contato do líder da aliança. Essa operação permite que os parceiros transfiram a designação principal do líder da aliança ou modifiquem os detalhes de contato, mantendo a continuidade organizacional.
5. GetAllianceLeadContact API: atualiza as principais informações de contato do líder da aliança. Essa operação permite que os parceiros transfiram a designação principal do líder da aliança ou modifiquem os detalhes de contato, mantendo a continuidade organizacional.

Trabalhando com o perfil do parceiro

Gerenciamento de perfis

Os parceiros gerenciam perfis contendo detalhes da empresa (nome, descrição, site, logotipo) com controle de public/private visibilidade, validação assíncrona e rastreamento de status, permitindo uma atualização por vez.

1. Informações do perfil — Os perfis dos parceiros devem incluir detalhes essenciais da empresa, como nome de exibição, descrição, URL do site, logotipo IndustrySegments PrimarySolutionType e.
2. Support multilíngue — Os perfis de parceiros oferecem suporte a vários idiomas. Cada versão localizada do perfil contém um atributo para localidade.
3. Controle de visibilidade do perfil — Os parceiros podem configurar a visibilidade do perfil (público ou privado).
4. Armazenamento do perfil — As informações do perfil serão armazenadas no objeto da conta do parceiro.
5. Gerenciamento do status do perfil — As atualizações do perfil são validadas de forma assíncrona antes da publicação. Os parceiros podem verificar o status mais recente de atualização do perfil

(IN_PROGRESS, SUCCEED, FAILED e CANCELLED). Somente perfis aprovados se tornam visíveis publicamente.

6. Controle de simultaneidade de solicitações — Somente uma solicitação de atualização de perfil é permitida por vez. Se uma atualização de perfil já estiver em andamento, os clientes devem cancelar explicitamente a atualização contínua por meio da CancelPartnerProfileUpdateTask API antes de iniciar uma nova.
7. Cancelamento de tarefas de atualização — os parceiros podem cancelar uma atualização de perfil em andamento chamando a CancelPartnerProfileUpdateTask API. O cancelamento só é permitido enquanto a atualização estiver no status IN_PROGRESS e deve ser concluído antes de iniciar uma nova atualização.

Resumo de APIs

1. StartPutProfileTask API: essa API aceita atualizações de perfil de parceiro e executa a validação básica de entrada síncrona.
2. GetProfileUpdateTask API: essa API permite que os parceiros busquem o status atual de atualização do perfil. Ele deve ser usado após o cliente iniciar uma atualização de perfil via StartPutPartnerProfile API, permitindo que os parceiros verifiquem se a solicitação foi IN_PROGRESS, FAILED, SUCCEEDED ou CANCELLED.
3. CancelProfileTask API: permite que um parceiro cancele explicitamente uma tarefa de atualização de perfil que está atualmente no status IN_PROGRESS. Isso inclui as fases de verificação automática e revisão manual. O cancelamento de uma solicitação permite que o parceiro inicie uma nova atualização de perfil sem esperar pela conclusão do processo de verificação atual.
4. PutProfileVisibility API: ao chamar essa API, os parceiros podem controlar a visibilidade de um perfil, independentemente da localidade. Isso permite que os parceiros tornem os perfis públicos ou privados sem modificar o conteúdo.
5. GetProfileVisibility API: ao chamar a UpdatePartnerProfileVisibility API, os parceiros podem controlar a visibilidade de um perfil, independentemente da localidade. Isso permite que os parceiros tornem os perfis públicos ou privados sem modificar o conteúdo.

Trabalhando com gerenciamento de domínio

Gerenciamento de domínio

Os parceiros podem registrar e verificar domínios comerciais por meio da validação de e-mail para estabelecer a identidade organizacional e associar o treinamento e as certificações dos funcionários.

Resumo de APIs

1. SendEmailVerificationCode API: inicia o processo de verificação de e-mail enviando um código de verificação para o endereço de e-mail especificado. Usado tanto para criação de novos contatos quanto para atualizações de endereços de e-mail.
2. AssociateAwsTrainingCertificationEmailDomain API: associa um domínio de e-mail ao treinamento e à certificação da AWS para a conta do parceiro, permitindo a verificação automática das certificações dos funcionários.
3. DisassociateAwsTrainingCertificationEmailDomain API: remove a associação entre um domínio de e-mail e o treinamento e a certificação da AWS para a conta do parceiro.

Trabalhando com conexões de contas de parceiros

Os parceiros podem gerenciar conexões entre suas próprias contas ou conexões com AWS contas de outros parceiros usando a API AWS Partner Connection. O processo envolve duas fases:

1. Fase de convite de conexão: iniciando e respondendo às solicitações de conexão
2. Fase de conexão ativa: gerenciamento de conexões estabelecidas e suas atividades comerciais associadas

Criando e enviando um convite de conexão

Etapa 1: criar um convite de conexão

A primeira etapa para estabelecer uma conexão de parceiro é criar e enviar um convite de conexão usando a `CreateConnectionInvitation` ação. Ao criar um convite, você deve especificar:

- Catálogo: o AWS catálogo (AWS ou Sandbox) em que a conexão será gerenciada
- ConnectionType: O tipo de relacionamento que você deseja estabelecer. Escolha uma das seguintes opções:
  - **`OPPORTUNITY_COLLABORATION`**: use esse tipo quando quiser enviar uma solicitação de conexão para a conta de outro parceiro para que você possa enviar a ele convites de engajamento de oportunidades para colaboração de oportunidades
  - **`SUBSIDIARY`**: Use esse tipo quando quiser conectar várias contas de vendedor do AWS Marketplace que você possui à sua conta “principal” que você vinculou à APN ou na qual você está registrado como Parceiro

- ReceiverIdentifier: ID do perfil público do parceiro do destinatário (para o tipo de conexão Opportunity Collaboration) ou ID AWS da conta (para o tipo de conexão da subsidiária)
- E-mail: Seu endereço de e-mail de contato para comunicação
- Nome: Seu nome como remetente
- Mensagem: uma descrição explicando a finalidade da solicitação de conexão

Quando criado, o convite entra no estado Pendente, aguardando a ação do destinatário.

> **Important**
>
> Ao enviar um convite de conexão, você concorda em revelar o ID da sua AWS conta ao destinatário, caso ele aceite o convite. Essa divulgação é necessária para estabelecer a conexão no nível da conta.

Etapa 2: monitore seus convites enviados

Você pode acompanhar o status dos convites enviados usando a `ListConnectionInvitations` ação com o `ParticipantType` parâmetro definido como. `SENDER` Isso permite a você:

- Veja todos os convites enviados
- Filtrar por status (`PENDING`,`ACCEPTED`,`REJECTED`,`EXPIRED`)
- Filtrar por tipo de conexão
- Confira convites específicos para outros parceiros

Etapa 3: cancelar um convite (opcional)

Se precisar retirar um convite pendente antes que ele seja aceito, você pode usar a `CancelConnectionInvitation` ação. Observe que:

- Você só pode cancelar convites no estado Pendente
- Depois que um convite é aceito e uma conexão é estabelecida, ela não pode ser cancelada
- Para encerrar uma conexão estabelecida, você deve usar a `CancelConnection` ação em vez disso

Recebendo e respondendo a convites de conexão

Etapa 1: Listar os convites recebidos

Para ver os convites de conexão que você recebeu, use a `ListConnectionInvitations` ação com o ParticipantType parâmetro definido como. `RECEIVER` Você pode filtrar por:

- Tipo de conexão
- Status (`PENDING`,`ACCEPTED`,`REJECTED`)
- Identificador do remetente

Etapa 2: revisar os detalhes do convite

Use a `GetConnectionInvitation` ação para ver detalhes completos de um convite específico, incluindo:

- Nome do remetente e e-mail de contato
- Mensagem de convite explicando o propósito
- Tipo de conexão que está sendo solicitado
- Data de expiração
- Informações do perfil público do remetente

Etapa 3: aceitar ou rejeitar o convite

Depois de analisar os detalhes do convite, você tem duas opções:

Opção A: aceitar o convite

Use a `AcceptConnectionInvitation` ação para estabelecer a conexão. Quando você aceita:

- Uma nova conexão é criada no estado Ativo
- O status do convite muda para Aceito
- O ID AWS da sua conta é revelado ao remetente
- Você pode começar a conduzir atividades comerciais habilitadas pelo tipo de conexão.

> **Important**
>
> Consentimento importante: ao aceitar um convite de conexão, você concorda em revelar o ID da sua AWS conta ao remetente. Essa divulgação é necessária para estabelecer a conexão no nível da conta.

Opção B: rejeitar o convite

Use a `RejectConnectionInvitation` ação se você não quiser estabelecer a conexão. Opcionalmente, você pode fornecer um motivo para a rejeição. Uma vez rejeitado:

- O status do convite muda para Rejeitado
- Nenhuma conexão é estabelecida
- O convite não aparecerá mais na sua lista de convites pendentes

Expiração automática

Se nenhuma ação for tomada dentro do período de expiração, o sistema fará a transição automática do convite para o estado Expirado. Convites expirados não podem ser aceitos nem rejeitados.

Gerenciando conexões ativas

Visualizando suas conexões

Depois que uma conexão é estabelecida, ambas as partes podem visualizá-la e gerenciá-la usando as seguintes ações:

`ListConnections`: recupere uma lista de suas conexões com opções de filtragem:

- Filtrar por tipo e status da conexão (por exemplo,`OPPORTUNITY_COLLABORATION:ACTIVE`)
- Filtrar pelo identificador de outra parte
- Exibir informações resumidas para cada conexão

`GetConnection`: recupere detalhes completos de uma conexão específica, incluindo:

- Todos os tipos de conexão associados a esse relacionamento de parceiro
- Status de cada tipo de conexão (ativa ou encerrada)

- Informações de contato do convite original
- AWS IDs de conta e IDs de perfil público de ambas as partes
- Carimbos de data e hora de criação e encerramento para cada tipo de conexão

Adicionando tipos de conexão a uma conexão existente

Se você já tiver uma conexão ativa com um parceiro e quiser estabelecer um novo tipo de relacionamento, poderá enviar um novo convite de conexão com um tipo de conexão diferente. Após a aceitação:

- O novo tipo de conexão é adicionado à conexão existente
- Ambos os tipos de conexão permanecem gerenciáveis de forma independente
- O mesmo ID de conexão é mantido para o relacionamento

> **Note**
>
> Só pode haver uma conexão ativa de qualquer tipo entre duas contas em um determinado catálogo a qualquer momento.

Encerrando tipos de conexão

Qualquer uma das partes pode encerrar um tipo de conexão específico usando a `CancelConnection` ação. Ao encerrar:

- Especifique o ID da conexão e o tipo de conexão a ser encerrada
- Forneça um motivo para a rescisão
- O status do tipo de conexão muda para `CANCELED`
- Outros tipos de conexão dentro da mesma conexão permanecem inalterados

Encerramento completo da conexão

Quando todos os tipos de conexão em uma conexão são encerrados, a própria conexão passa para o estado Terminado. Uma conexão completamente encerrada:

- Não pode ser reativado, mas você pode enviar uma nova solicitação de conexão para continuar os negócios com essa conta

- Permanece no sistema para manutenção de registros históricos
- Pode ser visualizado usando a `GetConnection` API
- Requer um novo convite para restabelecer o relacionamento

Monitorando eventos de conexão

Os parceiros devem monitorar eventos relacionados à conexão usando a Amazon EventBridge para se manterem informados sobre:

- Novos convites de conexão recebidos
- Alterações de status nos convites enviados (`ACCEPTED`,,`REJECTED`) `EXPIRED`
- Terminações de conexão iniciadas pela outra parte

Ao receber um evento, use a ação Get API apropriada para obter os detalhes mais recentes:

- `GetConnectionInvitation`para atualizações de convites
- `GetConnection`para atualizações de conexão

Eventos possíveis

- Convite de conexão criado: notifica a conta do destinatário sobre os convites de conexão recebidos.
- Convite de conexão cancelado: notifica a conta do destinatário quando um remetente cancela um convite de conexão recebido.
- Convite de conexão expirado: notifica as contas do remetente e do destinatário quando um convite de conexão expira sem ação.
- Convite de conexão rejeitado: notifica a conta do remetente quando o convite de conexão é rejeitado pelo destinatário.
- Convite de conexão aceito: notifica a conta do remetente quando o convite de conexão é aceito e uma conexão é estabelecida.
- Conexão cancelada: notifica ambas as partes conectadas quando toda a conexão é totalmente encerrada.

Entendendo os estados de conexão

Estados do convite de conexão

Estado	Description	Pode fazer a transição para
`PENDING`	Estado inicial quando criado; aguardando a ação do receptor	`ACCEPTED`, `REJECTED`, `EXPIRED`, `CANCELED`
`ACCEPTED`	Receptor aceito; conexão criada	Nenhum (estado terminal)
`REJECTED`	O destinatário recusou; contém o motivo opcional da rejeição	Nenhum (estado terminal)
`EXPIRED`	System-expired devido a nenhuma ação dentro do período de expiração	Nenhum (estado terminal)
`CANCELED`	O remetente retirou o convite antes da aceitação	Nenhum (estado terminal)

Estados do tipo de conexão

Estado	Description
`ACCEPTED`	O tipo de conexão é operacional; as atividades comerciais associadas estão ativadas
`CANCELED`	Tipo de conexão encerrado por qualquer uma das partes

Práticas recomendadas

1. Comunicação clara: forneça mensagens detalhadas e claras em seus convites de conexão, explicando o propósito e a colaboração esperada.

2. Respostas oportunas: monitore e responda prontamente aos convites de conexão para evitar a expiração automática.
3. Revisão regular: revise periodicamente suas conexões ativas para garantir que elas permaneçam relevantes às suas necessidades comerciais atuais.
4. Rescisão adequada: ao encerrar um relacionamento comercial, use a CancelConnection ação com um motivo claro para manter a comunicação profissional.
5. Monitoramento de eventos: configure EventBridge regras para receber automaticamente notificações de alterações no status da conexão, para ajudar a se manter informado sobre atualizações importantes.
6. Gerenciamento do tipo de conexão: considere quais tipos de conexão são necessários antes de estabelecer conexões, pois cada tipo permite diferentes recursos comerciais.
7. Conscientização sobre segurança: lembre-se de que estabelecer uma conexão revela os IDs das AWS contas entre as partes. Conecte-se somente com parceiros confiáveis.

Tipos de conexão e suas finalidades

Tipo de conexão	Finalidade	Caso de uso
`OPPORTUNITY_COLLABORATION`	Permite a colaboração em oportunidades de venda conjunta com outros parceiros	Quando você quiser que outro parceiro tenha acesso à sua oportunidade ou vice-versa, use esse tipo de conexão. Essa conexão permite que você envie convites de engajamento de oportunidades para parceiros conectados.
`SUBSIDIARY`	Estabelece conexões entre suas várias contas de vendedor do AWS Marketplace e sua conta principal do Partner	Use quando você tiver várias contas de vendedor do AWS Marketplace que deseja conectar à sua conta principal , na qual você está registrado como parceiro e vendedor.

Trabalhando com a verificação de parceiros

Verificação do parceiro gerente

A verificação do parceiro é um pré-requisito obrigatório para o registro da conta do parceiro. Antes que os parceiros possam criar uma conta de parceiro, eles devem concluir os processos de verificação comercial e de identidade. Essas verificações validam que a empresa está legalmente registrada e confirmam a identidade da pessoa que está registrando a conta.AWS

Os parceiros podem utilizar as APIs de verificação disponíveis para iniciar e monitorar os processos de verificação:

Durante a verificação

1. Iniciação da verificação comercial — os parceiros iniciam a verificação comercial chamando a `StartVerification` API com detalhes do registro comercial, incluindo nome legal, ID de registro, código do país e jurisdição de incorporação.
2. Início da verificação de identidade — os parceiros iniciam a verificação de identidade chamando a `StartVerification` API com as informações do registrante. A API retorna um URL de preenchimento seguro em que o registrante conclui o fluxo de trabalho de verificação de identidade usando uma identificação emitida pelo governo.
3. Monitoramento do status da verificação — os parceiros usam a `GetVerification` API para recuperar o status atual e os detalhes dos processos de verificação. A API retorna o tipo de verificação, o status, os registros de data e hora e detalhes específicos da verificação.
4. Conclusão da verificação — Tanto a verificação comercial quanto a verificação de identidade devem atingir `SUCCEEDED` o status antes que os parceiros possam prosseguir com a criação da conta do parceiro. As verificações falhadas ou expiradas devem ser resolvidas antes do registro da conta.

Pós-verificação

1. Criação de conta de parceiro — após a verificação bem-sucedida, os parceiros continuam criando sua conta de parceiro usando a `CreatePartner` API descrita na documentação da conta de parceiro.
2. Recuperação do registro de verificação — Os parceiros podem recuperar os detalhes da verificação a qualquer momento usando a `GetVerification` API com a ID de verificação retornada durante o início.

Resumo de APIs

1. **`StartVerification`**API: os parceiros iniciam os processos de verificação chamando a `StartVerification` API. Para verificação comercial, a API valida os detalhes do registro comercial. Para verificação de identidade, a API gera um URL de preenchimento com limite de tempo para que o registrante conclua a verificação de identidade.
2. **`GetVerification`**API: depois de iniciar a verificação, os parceiros usam a `GetVerification` API para recuperar o status e os detalhes da verificação. A API retorna o status atual, os registros de data e hora e as informações específicas da verificação.

Usar o AWS API de vendas da Central de Parceiros

Essa referência da API do AWS Partner Central foi projetada para ajudar AWS os parceiros a integrar sistemas de gerenciamento de relacionamento com o cliente (CRM) com o Partner Central. A API automatiza as interações com o Partner Central, o que ajuda a garantir engajamentos efetivos em atividades comerciais conjuntas.

A API fornece a funcionalidade padrão AWS da API. Acesse-o usando ações de API ou usando um AWS SDK personalizado para sua linguagem de programação ou plataforma. Para obter mais informações, consulte Introdução AWS e Ferramentas para desenvolver AWS.

Recursos oferecidos por AWS API do Partner Central

1. Gerenciamento de oportunidades: facilita o gerenciamento de oportunidades de coselling com AWS o uso de ações de API`CreateOpportunity`, como,`UpdateOpportunity`, `ListOpportunitiesGetOpportunity`, e. `AssignOpportunity`
2. AWS gerenciamento de referências: facilita o recebimento de referências compartilhadas AWS usando ações como`ListEngagementInvitations`,`GetEngagementInvitation`, e. `StartEngagementByAcceptingInvitation` `RejectEngagementInvitation`
3. Associação de entidades: associe entidades relacionadas, como AWS produtos, soluções de parceiros, AWS Marketplace soluções, AWS Marketplace produtos e ofertas AWS Marketplace privadas, a oportunidades usando as ações `AssociateOpportunity` `DisassociateOpportunity` e.
4. Exibir detalhes da AWS oportunidade: use a `GetAWSOpportunitySummary` ação para recuperar resumos em tempo real das AWS oportunidades vinculadas às suas oportunidades.
5. Listar soluções: fornece APIs de lista para listar soluções que os parceiros oferecem usando`ListSolutions`.

6. Assinatura de eventos: os parceiros podem se inscrever para receber atualizações em tempo real sobre oportunidades ouvindo eventos como Oportunidade criada, Oportunidade atualizada, Convite de engajamento aceito, Convite de engajamento rejeitado e Convite de engajamento criado usando a Amazon EventBridge.

Regiões e endpoints compatíveis

A API AWS Partner Central está disponível na região Leste dos EUA (Norte da Virgínia).

Região	Endpoint
us-east-1	partnercentral-selling.us-east-1.api.aws

Os parceiros podem testar e validar integrações de API em um ambiente seguro de sandbox. Isso permite que você teste as ações da API sem afetar os dados ativos. Para obter mais informações, consulte Testando em uma caixa de areia para o AWS API de vendas da Central de Parceiros.

Venda de entidades de API

AWS As entidades do Partner Central representam os principais componentes de negócios usados em contratos de covenda entre parceiros e.AWS Essas entidades encapsulam informações relacionadas a oportunidades, soluções, produtos e ofertas, permitindo a colaboração e o gerenciamento tranquilos das atividades de vendas conjuntas. As seções a seguir fornecem descrições das entidades principais na API de vendas do Partner Central.

Oportunidade

Uma oportunidade é uma venda ou negócio em potencial que uma empresa identifica e busca ativamente. É um cliente potencial qualificado ou um lead com uma necessidade específica que os produtos ou serviços da empresa podem atender. Normalmente, as oportunidades são rastreadas em um funil de vendas ou sistema de CRM e formam a base da receita futura. O gerenciamento eficaz de oportunidades envolve nutrir leads durante o processo de vendas, desde a qualificação inicial até o fechamento. Para obter mais informações, consulte Trabalhando com suas oportunidades e tipos de dados

Soluções para parceiros

Representa uma solução de parceiro (chamada de oferta no AWS Partner Central), que é um produto de software ou uma prática de consultoria criada e fornecida por AWS parceiros. As

soluções de parceiros ajudam os clientes a enfrentar desafios comerciais específicos ou atingir metas específicas usando AWS serviços. Para obter mais informações, consulte O que é uma solução?

AWS produto

Representa um AWS serviço ou produto específico.AWS oferece uma ampla variedade de produtos e serviços projetados para fornecer soluções de infraestrutura escaláveis, confiáveis e econômicas. Os parceiros podem obter a lista mais recente de AWS produtos na página de importação em massa no Partner Central (Iniciar importação >AWS Produtos e soluções). Para obter mais informações, você pode aprender sobre AWS os Produtos ou ver a lista de todos os AWS Produtos.

AWS Marketplace oferta privada

AWS Marketplace a oferta privada é um recurso que permite que AWS Marketplace os vendedores ofereçam preços e condições específicos para AWS clientes individuais. Por meio de ofertas privadas, os vendedores podem negociar preços personalizados, cronogramas de pagamento e termos de licença de usuário final.AWS os clientes podem obter soluções de software que atendam aos seus requisitos específicos e, ao mesmo tempo, possivelmente se beneficiem de condições ou preços mais favoráveis em comparação com as ofertas padrão. O processo de oferta privada envolve o vendedor criar uma oferta exclusiva com termos personalizados, que é então compartilhada de forma privada com o AWS cliente designado para análise e aceitação. Para obter mais informações, consulte Ofertas privadas em AWS Marketplace.

convite de noivado

O convite de engajamento se refere a uma solicitação formal de parceiros AWS para colaborar em uma indicação específica. Isso permite que AWS o parceiro trabalhe em conjunto para impulsionar a oportunidade. O convite pode ser aceito ou rejeitado pelo parceiro.

Tópicos

- Trabalhando com suas oportunidades
- Trabalhando com oportunidades de AWS
- Trabalhando com atualizações de oportunidades
- Trabalhando com oportunidades de vários parceiros
- Associando, desassociando e atribuindo oportunidades
- Trabalhando com seus leads

- Trabalho com insights de dimensionamento de negócios
- Práticas recomendadas

Trabalhando com suas oportunidades

O que é uma oportunidade?

Durante os estágios iniciais do processo de vendas, um representante de vendas avalia se um indivíduo interessado (chamado lead) tem potencial para se tornar um cliente. Essa fase de avaliação e validação é chamada de Qualificação. Quando um lead é considerado qualificado e tem uma probabilidade maior de se converter em cliente, ele é classificado como uma oportunidade.

Trabalhando com suas oportunidades

Os parceiros podem gerenciar oportunidades criadas em seus sistemas de CRM e sincronizá-las com o AWS Partner Central usando a API de vendas do AWS Partner Central. Isso permite que os parceiros rastreiem e gerenciem oportunidades do início ao encerramento.

Criando uma oportunidade

A primeira etapa no gerenciamento de oportunidades é criar uma oportunidade usando a `CreateOpportunity` ação. Isso cria uma oportunidade com o `Lifecycle.ReviewStatus` conjunto para`Pending Submission`. No entanto, a oportunidade ainda não foi enviada AWS para validação.

Depois que a oportunidade for criada, você deverá associar pelo menos uma Solução,AWS Marketplace Solução ou AWS Marketplace Produto de Parceiro à oportunidade usando a `AssociateOpportunity` ação. Essa ação define claramente o que a oportunidade está tentando vender. Você pode ver a lista completa de soluções disponíveis em sua conta usando a `ListSolutions` API e associar entre uma e dez soluções.

Opcionalmente, você também pode associar AWS produtos relevantes à oportunidade usando a `AssociateOpportunity` ação. Essa etapa ajuda as equipes de AWS vendas a entender quais AWS produtos devem ser vendidos em conjunto com a oportunidade.

Depois que a solução ou o produto necessário forem associados e, opcionalmente,AWS os produtos estiverem vinculados, você poderá começar a se engajar na oportunidade usando a `StartEngagementFromOpportunityTask` ação. É quando você envia a oportunidade de iniciar um compromisso.

Processo de revisão

Depois de iniciar o engajamento na oportunidade usando a `StartEngagementFromOpportunityTask` ação, a oportunidade entra na fase de AWS validação e `Lifecycle.ReviewStatus` é definida como`Submitted`. Nenhuma alteração pode ser feita na oportunidade até que o processo de análise seja concluído.

Durante essa fase de validação,AWS garante que os detalhes da oportunidade sejam precisos e completos. Enquanto a validação está em andamento, o. `Lifecycle.ReviewStatus` é definido como`In-Review`.

Se houver alterações ou detalhes adicionais exigidos do parceiro,AWS`Lifecycle.ReviewStatus` defina `Action Required` o como e todas as atualizações necessárias serão comunicadas por meio do `Lifecycle.ReviewComments` campo.

Depois que a oportunidade passa pela validação, ela `Lifecycle.ReviewStatus` muda para`Approved`, tornando a oportunidade pronta para atividades de venda conjunta.

Os parceiros devem monitorar o `Opportunity Updated` evento usando a Amazon EventBridge. Isso os notificará sobre quaisquer alterações de status ou feedback de AWS. Ao receber o evento, os parceiros podem usar a ação da `GetOpportunity` API para buscar os detalhes mais recentes da oportunidade e verificar o `Lifecycle.ReviewStatus` campo.

Resolvendo problemas de validação

Se `Lifecycle.ReviewStatus` estiver definido como`Action Required`, os parceiros precisarão abordar os problemas destacados por AWS. Para resolver isso, os parceiros podem atualizar a oportunidade usando a ação `UpdateOpportunity` da API.

Durante o `Action Required` estado, somente alguns campos são editáveis. Estes campos incluem:

1. `Customer.Account.Address.City`
2. `Customer.Account.Address.Country`
3. `Customer.Account.Address.PostalCode`
4. `Customer.Account.Address.StateOrRegion`
5. `Customer.Account.Address.StreetAddress`
6. `Customer.Account.WebsiteUrl`

7. `LifeCycle.TargetCloseDate`
8. `Project.ExpectedCustomerSpend.Amount`
9. `Project.ExpectedCustomerSpend.Currency`
10. `Project.CustomerBusinessProblem`
11. `PartnerOpportunityIdentifier`

Depois de fazer as alterações necessárias, a oportunidade entra novamente na fase de validação e o processo se repete até que a oportunidade `Lifecycle.ReviewStatus` seja definida como ou. `Approved` `Disqualified`

Post-Approval Atualizações

Quando a oportunidade surgir`Approved`, os parceiros podem continuar atualizando a oportunidade conforme necessário usando a `UpdateOpportunity` ação, facilitando atividades de venda conjunta contínuas.

Os parceiros devem continuar monitorando os `Opportunity Updated` eventos por meio da Amazon EventBridge para se manterem atualizados sobre quaisquer alterações. Para obter mais informações sobre o rastreamento de AWS atualizações, consulte a seção “Trabalhando com atualizações de oportunidades”. Os parceiros também podem atualizar campos selecionados com base nas regras de validação comercial.

Trabalhando com oportunidades de AWS

1. Recebendo o AWS Oportunidade

As oportunidades são compartilhadas com os parceiros quando um executivo de AWS vendas associa um parceiro a uma oportunidade no sistema AWS de CRM. Elas são chamadas de AWS Oportunidades, distintas das oportunidades criadas na própria conta do parceiro.

Quando uma AWS oportunidade tem um parceiro anexado,AWS cria um convite de engajamento contendo um subconjunto de dados da AWS oportunidade. Os parceiros receberão um evento criado por convite de engajamento.

2. Analisando o convite de noivado

O convite de engajamento contém informações essenciais `Project.TitleProject.CustomerUseCase`,

como`Lifecycle.Stage`,`Project.CustomerBusinessProblem`,, e alguns campos adicionais. Os parceiros podem usar esses dados para decidir se querem aproveitar a oportunidade.

No entanto, os seguintes campos da AWS Oportunidade não estão incluídos no convite de engajamento:

```
Customer.Account.Address.StreetAddress
Customer.Account.Address.City
Customer.Account.Address.PostalCode
Customer.Contact
Customer.Account.AWSAccountId
Project.OtherSolutionDescription
RelatedEntityIdentifiers
```

3. Como lidar com o convite de noivado

Ao receber um evento criado por convite de engajamento, os parceiros podem usar a `GetEngagementInvitation` ação para recuperar detalhes da AWS oportunidade.

Se o parceiro decidir não aproveitar a oportunidade, ele poderá rejeitar o convite usando a `RejectEngagementInvitation` ação, junto com a necessária`RejectionReason`. Uma vez rejeitado, o acesso à oportunidade é perdido.

Para aceitar o convite e continuar, os parceiros devem usar a `StartEngagementByAcceptingInvitationTask` ação. Essa é uma ação assíncrona que executa sequencialmente as seguintes tarefas:

1. Aceita o convite de noivado.
2. Cria uma nova oportunidade na conta do parceiro usando dados da AWS oportunidade.
3. Inclui detalhes adicionais necessários para identificar o cliente na conta do parceiro.

Após a conclusão, um evento de oportunidade criada é acionado com o ID de oportunidade correspondente. Os parceiros podem então usar essa ID na `GetOpportunity` ação para recuperar todos os detalhes da oportunidade.

Se o parceiro não estiver usando eventos, ele poderá ligar `StartEngagementByAcceptingInvitationTask` novamente com a mesma carga para verificar o status mais recente.

4. Gerenciando o AWS Oportunidade Post-Acceptance

Quando a oportunidade estiver na conta do parceiro, ela poderá ser gerenciada e atualizada como qualquer outra oportunidade no sistema. Os parceiros podem usar ações como `UpdateOpportunity` fazer as alterações necessárias.

Para obter mais detalhes sobre como rastrear AWS atualizações e gerenciar atualizações de oportunidades, consulte a Trabalhando com atualizações de oportunidades seção.

Trabalhando com atualizações de oportunidades

Atualizando oportunidades

Os parceiros podem usar a `UpdateOpportunity` ação para atualizar oportunidades, com regras específicas que determinam quais campos são atualizados e quando são atualizados:

1. As atualizações não podem ser feitas se `Lifecycle.ReviewStatus` for `Submitted` ou`In-Review`.
2. Antes do envio, os parceiros podem fazer atualizações, mas não as AWS processarão a menos que a `StartEngagementFromOpportunityTask` ação seja invocada.
3. Quando a oportunidade está ativa `Submitted` ou com `In-review` status, todas as atualizações são bloqueadas.
4. Se a oportunidade estiver em `Action Required` status,AWS abrirá campos selecionados para atualizações. Estes campos incluem:
   - `Customer.Account.Address.City`
   - `Customer.Account.Address.Country`
   - `Customer.Account.Address.PostalCode`
   - `Customer.Account.Address.StateOrRegion`
   - `Customer.Account.Address.StreetAddress`
   - `Customer.Account.WebsiteUrl`
   - `LifeCycle.TargetCloseDate`
   - `Project.ExpectedCustomerSpend.Amount`
   - `Project.ExpectedCustomerSpend.CurrencyCode`
   - `Project.ExpectedCustomerSpend.EstimationURL`
   - `Project.ExpectedCustomerSpend.Frequency`

- `Project.ExpectedCustomerSpend.TargetCompany`
- `Project.CustomerBusinessProblem`
- `PartnerOpportunityIdentifier`

5. Após o processo de revisão (ou seja, quando `Lifecycle.ReviewStatus` está definido como`Approved`), os seguintes campos não podem ser atualizados:
   - `Customer.Account.Address.Country`
   - `Customer.Account.Address.PostalCode`
   - `Customer.Account.Industry`
   - `Customer.Account.WebsiteUrl`
   - `Project.CustomerBusinessProblem`
   - `PartnerOpportunityIdentifier`
   - `Project.Title`
6. Para todos os outros campos, as atualizações podem ser feitas usando a `UpdateOpportunity` ação. No entanto, restrições adicionais podem ser aplicadas com base nas regras de negócios do programa ou tipo de oportunidade específico. Para obter mais detalhes, consulte as validações em nível de campo.
7. Para todas as atualizações feitas por meio da interface do usuário e da API, o evento Opportunity Updated é gerado.

Recebendo atualizações de AWS sobre oportunidades

AWS normalmente atualiza AWS oportunidades e, cada vez que uma atualização é feita, um evento de oportunidade atualizada é gerado.

Para recuperar as atualizações mais recentes do AWS, os parceiros precisam invocar duas ações separadas

1. `GetOpportunity`para recuperar detalhes da oportunidade do parceiro.
2. `GetAWSOpportunitySummary`para recuperar resumos em tempo real dos dados da AWS oportunidade.

A maioria das atualizações regulares do AWS estará disponível por meio da `GetAWSOpportunitySummary` resposta. No entanto,AWS pode ocasionalmente atualizar diretamente os atributos na oportunidade do parceiro.

Para consumir atualizações do AWS

1. Invoque a `GetAWSOpportunitySummary` ação para recuperar os detalhes mais recentes da AWS oportunidade.
2. Se as mudanças precisarem ser refletidas na oportunidade do parceiro, use a `UpdateOpportunity` ação para copiar os dados relevantes na oportunidade do parceiro.

Os parceiros podem optar por automatizar esse processo como um mecanismo de atualização direta ou implementar um processo de revisão manual para validar e atualizar os dados.

Trabalhando com oportunidades de vários parceiros

AWS Partner s podem trabalhar juntos em oportunidades AWS como participantes ativos.

Tópicos

- Engajamentos e fotos
- Criação de uma política personalizada para ResourceSnapshotJobRole
- Convidando parceiros para uma oportunidade
- Recuperando detalhes do convite de engajamento
- Respondendo a um convite de noivado
- Analisando e atualizando oportunidades com vários parceiros
- Usando instantâneos para receber atualizações de parceiros
- Visualizando membros do engajamento
- Monitorando o status do trabalho de captura instantânea de recursos

Engajamentos e fotos

Um contrato é um recurso de propriedade da empresa AWS para colaboração entre várias AWS Partner contas e AWS sobre uma oportunidade para o cliente. Os contratos facilitam o compartilhamento seguro de informações entre parceiros e a colaboração, ao mesmo tempo em que mantêm a propriedade e o controle individuais das oportunidades. Os engajamentos abrangem oportunidades que se originam de ambos AWS e dos parceiros, como um participante AWS ativo. Os parceiros podem convidar AWS ou outros parceiros para participar de um compromisso usando`EngagementInvitations`. Quando os parceiros convidados aceitam, eles recebem sua própria oportunidade dentro do mesmo compromisso.

Os membros do engajamento compartilham o progresso por meio de instantâneos — cópias pontuais e imutáveis de campos específicos de um recurso subjacente, como uma oportunidade. Os instantâneos são criados dentro do engajamento e compartilhados com os membros. Quando o recurso subjacente muda, o proprietário pode criar uma nova revisão de instantâneo para refletir as atualizações.AWS fornece trabalhos instantâneos — trabalhos de propriedade do cliente que criam automaticamente novas revisões quando o recurso é alterado. Depois de compartilhados, os instantâneos permanecem permanentemente acessíveis aos membros do engajamento.

Criação de uma política personalizada para ResourceSnapshotJobRole

Colaborar em oportunidades com vários parceiros exige compartilhar instantâneos de suas oportunidades com outros parceiros em um compromisso. Para manter o acesso aos detalhes mais recentes da oportunidade, você precisa criar um papel personalizado chamado`ResourceSnapshotJobRole`. Essa função permite que o sistema crie instantâneos de suas oportunidades em seu nome e recupere instantâneos de outros parceiros no mesmo contrato. Sem essa função, todas as atualizações que você fizer em sua oportunidade não serão visíveis para outros parceiros no engajamento, e eles continuarão vendo instantâneos desatualizados dos dados de sua oportunidade.

> **Note**
>
> Você pode usar um nome diferente de`ResourceSnapshotJobRole`. Se você usar um nome diferente, substitua todas as instâncias das `ResourceSnapshotJobRole` políticas a seguir pelo nome da sua política.

Para criar essa função, acesse seu console do IAM e crie a `ResourceSnapshotJobRole` função com a seguinte política de confiança personalizada:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "resource-snapshot-job.partnercentral-selling.amazonaws.com"
      },
```

```
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Depois de criar essa função, você precisa anexar a política AWSPartnerCentralSellingResourceSnapshotJobExecutionRolePolicyAWS gerenciada ou anexar a seguinte política de permissões:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "partnercentral:CreateResourceSnapshot"
            ],
            "Resource": [
                "arn:aws:partnercentral:*::catalog/AWS/engagement/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "partnercentral:GetOpportunity"
            ],
            "Resource": [
                "arn:aws:partnercentral:*:111122223333:catalog/AWS/opportunity/*"
            ]
        }
    ]
}
```

Substitua `{account}` pelo seu ID de conta da AWS.

Depois de criar a `ResourceSnapshotJobRole` função e anexar as permissões, você precisa definir a `ResourceSnapshotJobRole` para sua organização. Use a PutSellingSystemSettingsação para

definir a função que você criou como a da `ResourceSnapshotJobRole` sua empresa (identificada pela sua AWS conta):

```
aws-cli/2.13.5 Python/3.11.4 Linux/4.14.255-314-253.539.amzn2.x86_64
Host: partner-central.aws.amazon.com
Content-Type: application/json

{
  "ResourceSnapshotJobRoleIdentifier": "arn:aws:iam::{account}:role/
ResourceSnapshotJobRole"
}
```

`{account}`Substitua pelo ID da sua AWS conta e `ResourceSnapshotJobRole` pelo nome da função que você criou. Essa função será assumida implicitamente pelos trabalhos de instantâneos de recursos para acessar suas oportunidades e criar instantâneos para compartilhar com outros parceiros no compromisso.

Convidando parceiros para uma oportunidade

Os parceiros podem colaborar em oportunidades provenientes de ambos os parceiros e AWS iniciando um convite de engajamento. Você pode convidar no máximo nove parceiros para uma oportunidade.

Para convidar parceiros para uma oportunidade

1. Siga as etapas em Encontrar e se conectar com parceiros no Guia de vendas do AWS Partner Central. Você deve se conectar com os parceiros antes de convidá-los para oportunidades. Essa conexão concede acesso mútuo aos IDs de conta uns dos outros, garantindo que os convites cheguem à conta do parceiro pretendido.
2. Use a StartEngagementFromOpportunityTaskação para criar um engajamento e associar uma oportunidade a ele. Essa ação assíncrona executa as seguintes tarefas sequencialmente:
    1. Cria um engajamento.
    2. Associa a oportunidade ao engajamento.
    3. Invoca a CreateEngagementInvitationação para.AWS
    4. Envia a oportunidade para.AWS
    5. Inicia o `ResourceSnapshotJob` para criar instantâneos.
3. Convide outros parceiros para participarem.

a. Para obter o ID de engajamento associado à sua oportunidade, use a ListEngagementFromOpportunityTasksação, filtrada pelo `TaskIdentifier` retornado na etapa anterior.

b. Use o ID de engajamento e o ID da conta do parceiro receptor para enviar um convite com`CreateEngagementInvitation`.

Um convite bem-sucedido envia um evento criado pelo convite de engajamento para o parceiro receptor.

Recuperando detalhes do convite de engajamento

Ao receber um evento criado por convite de engajamento, use a GetEngagementInvitationação para recuperar os detalhes do convite. A resposta inclui informações essenciais sobre a oportunidade do cliente associada ao contrato.

Revise os detalhes do convite `InvitationMessageProject.Title`, especialmente os `Project.CustomerBusinessProblem` campos`Project.CustomerUseCase`,, e. Essas informações fornecem contexto sobre a oportunidade do cliente e as expectativas de colaboração do parceiro convidativo.

Use esses detalhes para avaliar se você deseja aproveitar a oportunidade aceitando ou recusando o convite de engajamento.

Respondendo a um convite de noivado

Quando um parceiro envia a você um convite de engajamento, ele cria um evento criado por convite de engajamento que notifica você sobre o convite. Você tem a opção de aceitar ou rejeitar o convite. Essa decisão determina seu envolvimento na oportunidade de vários parceiros.

Para rejeitar um convite de engajamento:

Use a RejectEngagementInvitationação para rejeitar o convite. Você deve fornecer um `RejectionReason` parâmetro explicando sua decisão. Depois de rejeitado, você perde o acesso aos detalhes do convite, e um evento de rejeição do convite de engajamento notifica o parceiro remetente de que você rejeitou o convite.

Para aceitar um convite de noivado:

Para aceitar o convite e continuar com a oportunidade de vários parceiros, use a StartEngagementByAcceptingInvitationTaskação. Essa ação assíncrona executa as seguintes tarefas sequencialmente:

1. Aceita o convite de noivado.
2. Cria uma nova oportunidade em sua conta de parceiro usando dados da oportunidade do parceiro remetente. Você receberá um evento criado por uma oportunidade.
3. Inclui detalhes adicionais necessários para identificar o cliente em sua conta.
4. Publica um evento aceito por convite de noivado que notifica o parceiro de que você aceitou o convite e foi adicionado ao compromisso.

Convite de noivado expirado

Se você não responder a um convite de engajamento dentro de quinze dias, ele expirará e você não poderá participar da oportunidade com vários parceiros. Um evento de convite de engajamento expirado notifica o parceiro remetente de que o convite expirou.

> **Note**
>
> Se você ainda quiser colaborar, o parceiro remetente deve iniciar um novo convite de engajamento.

Analisando e atualizando oportunidades com vários parceiros

Quando um parceiro inicia um engajamento em uma oportunidade, o status de revisão da oportunidade recém-criada na conta do parceiro receptor é determinado pelo status da oportunidade do parceiro remetente.

Oportunidade com status de avaliação enviada:

Se a oportunidade do parceiro remetente estiver `Lifecycle.ReviewStatus` definida como`Submitted`, o seguinte processo ocorrerá:

1. A nova oportunidade criada na conta do parceiro receptor será `Lifecycle.ReviewStatus` definida como`Submitted`.
2. O `Submitted` status permanece até que a oportunidade do parceiro remetente seja`Approved`.

3. Depois que a oportunidade do parceiro remetente for validada `Lifecycle.ReviewStatus` e definida como`Approved`, as oportunidades dos outros parceiros herdarão automaticamente o status. `Approved`

Esse processo garante detalhes consistentes das oportunidades em oportunidades com vários parceiros, eliminando a necessidade de várias avaliações.

Depois de concluído, um evento criado pela oportunidade é acionado com o ID da oportunidade correspondente. Os parceiros podem usar essa ID com a `GetOpportunity` ação para recuperar os detalhes completos da oportunidade.

Oportunidade com status de revisão aprovado:

Se o parceiro remetente iniciar um engajamento em uma oportunidade com `Lifecycle.ReviewStatus` definido como`Approved`, o seguinte processo ocorrerá:

1. A nova oportunidade criada na conta do parceiro receptor será `Lifecycle.ReviewStatus` definida como`Approved`.
2. Um evento criado pela oportunidade é acionado com o ID da oportunidade correspondente.
3. Os parceiros podem usar a GetOpportunityação com o ID da oportunidade para recuperar todos os detalhes da oportunidade.

No entanto, a oportunidade aprovada pode não ter alguns detalhes específicos do parceiro que não foram copiados da oportunidade do parceiro remetente. O parceiro receptor deve atualizar esses detalhes usando a UpdateOpportunityação quando o estágio da oportunidade for atualizado para `Qualified` ou em um estágio posterior:

- `Project.CustomerUseCase`
- `Project.DeliveryModels`
- `Solution`

- `PrimaryNeedsFromAws`
- `LifeCycle.TargetCloseDate`
- `Project.ExpectedCustomerSpend.Amount`
- `Project.ExpectedCustomerSpend.Currency`

- `Marketing.source`
- `Marketing.AwsFundingUsed`

Depois que a oportunidade é criada na conta do parceiro receptor, ela pode ser gerenciada e atualizada como qualquer outra oportunidade no sistema do parceiro. Os parceiros podem usar a UpdateOpportunityação para fazer alterações ou fornecer mais informações sobre seu envolvimento na oportunidade.

Usando instantâneos para receber atualizações de parceiros

Em um contrato, os parceiros mantêm e atualizam suas oportunidades de forma independente. Quando alguém revisa os recursos de um engajamento, como adicionar uma oportunidade, um evento criado por um instantâneo do recurso de engajamento é publicado.

Para se manter informado sobre as mudanças e acessar as informações mais atuais das oportunidades de outros parceiros, você pode usar as seguintes ações:

- ListEngagementResourceAssociations— Use essa ação para recuperar o ID de engajamento associado à oportunidade.
- ListResourceSnapshots— use essa ação para recuperar uma lista abrangente de todos os instantâneos de oportunidades associados ao contrato, fornecendo uma visão geral dos instantâneos disponíveis em todos os parceiros.
- GetResourceSnapshot— use essa ação para obter resumos em tempo real de instantâneos de oportunidades específicas, permitindo que você visualize as informações mais atualizadas sem acessar diretamente a oportunidade de outro parceiro.

Se você identificar mudanças ou atualizações relevantes da oportunidade de outro parceiro que devem ser refletidas na sua, use a UpdateOpportunityação. Essa ação facilita a incorporação seletiva de dados pertinentes à sua oportunidade, garantindo alinhamento e consistência em todo o engajamento.

Visualizando membros do engajamento

Um engajamento em uma oportunidade com vários parceiros pode ter até dez parceiros. Sempre que um novo parceiro aceita o convite de engajamento, um evento adicionado ao membro do engajamento é publicado, notificando todos os membros atuais sobre a mudança.

Para ver todos os membros colaborando em um compromisso, siga estas etapas:

1. Use a ListEngagementResourceAssociationsação para recuperar o ID de engajamento associado à oportunidade.
2. Forneça o ID da etapa 1 até a ListEngagementMembersação para obter os detalhes dos parceiros dos membros do engajamento.

> **Note**
>
> Somente membros de um compromisso podem invocar a `ListEngagementMembers` ação.

Monitorando o status do trabalho de captura instantânea de recursos

Ao trabalhar com oportunidades de vários parceiros, você deve criar uma função que permita criar instantâneos de suas oportunidades para outros parceiros e recuperar instantâneos de oportunidades de outros parceiros em um compromisso. Sem essa função, todas as atualizações que você fizer em sua oportunidade não estarão disponíveis para outros parceiros no contrato, e eles continuarão vendo fotos desatualizadas de sua oportunidade.

Para rastrear o status dos trabalhos de snapshot de recursos associados a uma oportunidade com vários parceiros em um contrato, siga estas etapas:

1. Use a ListEngagementResourceAssociationsação para recuperar o ID de engajamento associado à oportunidade.
2. Forneça o ID da etapa 1 até a ListResourceSnapshotJobsação para gerar uma lista de todos os trabalhos de snapshot pertencentes ao chamador no compromisso. Recupere o ID do trabalho a partir da resposta.
3. Forneça o ID do trabalho para a GetResourceSnapshotJobação para rastrear o status do trabalho e ver se ele está em execução.

A `ResourceSnapshotJob` operação publica métricas para a Amazon CloudWatch para suas operações assíncronas. CloudWatch processa os dados em métricas legíveis e quase em tempo real para ajudá-lo a monitorar o desempenho e a integridade do serviço.

As seguintes métricas estão disponíveis:

- Falhas — contabiliza problemas internos durante a execução do trabalho. Use essa métrica para monitorar a estabilidade do serviço e definir alarmes para limites de falha.

- Erros — rastreia problemas relacionados ao cliente durante o processamento do trabalho. Essa métrica ajuda a identificar problemas com as entradas ou configurações do cliente.
- Invocações — Representa o número total de chamadas de tarefas, incluindo tentativas bem-sucedidas e fracassadas. Use isso para acompanhar as tendências de uso do serviço ao longo do tempo.

> **Note**
>
> As métricas são reportadas CloudWatch somente quando há atividade no `ResourceSnapshotJob` serviço. Se não houver trabalhos ou dados para uma métrica específica, essa métrica não será relatada.

Para garantir o bom funcionamento da `ResourceSnapshotJob` operação:

- Use as métricas para verificar se o serviço funciona conforme o esperado.
- Crie CloudWatch alarmes para monitorar métricas e acionar ações (como enviar notificações) quando as métricas excederem os intervalos aceitáveis.
- Configure o monitoramento proativo para identificar e resolver problemas.

Para obter mais informações sobre o uso de CloudWatch métricas, consulte o Guia CloudWatch do usuário da Amazon.

Associando, desassociando e atribuindo oportunidades

As oportunidades podem ser associadas ou dissociadas de soluções,AWS produtos,AWS Marketplace soluções,AWS Marketplace produtos,AWS Marketplace ofertas e AWS Marketplace conjuntos de ofertas de parceiros em qualquer estágio do ciclo de vida da oportunidade.

Associando oportunidades a outras entidades

As entidades associadas são recuperadas do `GetOpportunity` método dentro do `RelatedEntityIdentifiers` objeto. O `RelatedEntityIdentifiers` pode ser atualizado usando`AssociateOpportunity`. Observe que esse campo não pode ser atualizado usando o `CreateOpportunity` método `UpdateOpportunity` ou.

Soluções

Antes de iniciar um contrato com AWS a `StartEngagementFromOpportunityTask` ação, é obrigatório associar pelo menos uma e até dez soluções de parceiros. Uma AWS recomendação pode ou não conter uma solução de parceiro.

Para visualizar suas soluções existentes, use a `ListSolutions` ação.

Os parceiros podem criar, atualizar e gerenciar suas soluções na `Build` seção do AWS Partner Central.

> **Important**
>
> Para avançar do estágio de oportunidade para Comprometido ou Lançado, você deve associar uma solução válida ou uma lista de AWS Marketplace produtos à oportunidade.

> **Note**
>
> As soluções de parceiros legados (`S-XXXX`identificadores) que usam o tipo de `Solutions` entidade ainda são suportadas. No entanto, recomendamos associar o tipo de `AwsMarketplaceSolutions` entidade às oportunidades, pois o tipo de `Solution` entidade será descontinuado no futuro.

AWS produtos

Até 20 AWS produtos podem ser associados a uma oportunidade. Para ver uma lista de AWS produtos disponíveis, use a lista de AWS produtos hospedados em GitHub. A associação com AWS produtos é feita exclusivamente usando a `AssociateOpportunity` ação. Para substituir ou remover um produto, use a `DisassociateOpportunity` ação.

Soluções

As oportunidades podem ser associadas a soluções da sua conta de vendedor.
Use a `AssociateOpportunity` ação com `RelatedEntityType` definido como`AwsMarketplaceSolutions`.

A entidade deve estar no status Público ou Limitado. O serviço rejeita entidades preliminares ou inativas.

Para associar uma solução, é necessário um ARN. O exemplo a seguir mostra o formato ARN de uma solução:

```
arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/Solution/soln-
ab1c2de3fgh4i
```

Você também pode associar soluções de uma AWS Marketplace conta subsidiária conectada à sua conta principal por meio do Partner Account Connection. Insira o ARN completo, incluindo o ID da conta da subsidiária.

A `ListSolutions` ação retorna `AwsMarketplaceSolutionArn` em cada resumo da solução e oferece suporte à filtragem por ARN.

AWS Marketplace Produtos

As oportunidades podem ser associadas aos AWS Marketplace produtos da sua conta de vendedor. Use a `AssociateOpportunity` ação com `RelatedEntityType` definido como`AwsMarketplaceProducts`.

A entidade deve estar no status Público ou Limitado. O serviço rejeita entidades preliminares ou inativas.

Para associar um produto, é necessário um ARN. O ARN inclui o tipo de produto (por exemplo,, `AmiProductSaaSProduct`,`ContainerProduct`). O exemplo a seguir mostra o formato ARN de um produto AMI:

```
arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/AmiProduct/prod-
ab1c2de3fgh4i
```

Você também pode associar produtos de uma AWS Marketplace conta subsidiária conectada à sua conta principal por meio do Partner Account Connection. Insira o ARN completo, incluindo o ID da conta da subsidiária.

AWS Marketplace ofertas e conjuntos de ofertas

AWS Marketplace ofertas

As oportunidades podem ser vinculadas a uma oferta AWS Marketplace privada. Para ver as ofertas disponíveis, use o ListEntities da API do AWS Marketplace Catálogo. Atualmente, você só pode associar ofertas da conta do AWS Marketplace vendedor vinculada ao AWS Partner Central.

Para associar uma oferta privada, o ARN é necessário. Amostra:

```
arn:aws:aws-marketplace:us-east-1:123123123123:AWSMarketplace/Offer/offer-
dtn3example1tg
```

AWS Marketplace conjuntos de ofertas

As oportunidades também podem ser associadas aos conjuntos de AWS Marketplace ofertas. Os conjuntos de ofertas permitem o agrupamento de várias ofertas de mercado relacionadas para um pacote abrangente de soluções. Para ver os conjuntos de ofertas disponíveis, use o ListEntities da API de AWS Marketplace catálogo.

Para associar um conjunto de ofertas, é necessário um ARN. Amostra:

```
arn:aws:aws-marketplace:us-east-1:123123123123:AWSMarketplace/OfferSet/offerset-
dtn3example1tg
```

Importante: uma oportunidade pode ser associada a uma oferta OU a um conjunto de ofertas, mas não a ambos. Somente um desses tipos de entidade pode ser associado a uma oportunidade por vez.

Desassociando oportunidades de outras entidades

Use a `DisassociateOpportunity` ação para desvincular a oportunidade dos seguintes tipos de entidade:

- Soluções
- AWS Produtos
- AWS Marketplace Soluções
- AWS Marketplace Produtos
- AWS Marketplace Ofertas
- AWS Marketplace Conjuntos de ofertas

Dependendo do estado da oportunidade, regras de validação diferentes se aplicam quando você desvincula esses objetos relacionados.

Atribuição de oportunidades

Use `AssignOpportunity` para alterar o proprietário da oportunidade. Você pode definir qualquer um dos usuários do Partner Central como proprietário da oportunidade.

Trabalhando com seus leads

O que é um lead?

Nas vendas business-to-business (B2B), um lead representa um cliente em potencial que expressou interesse nos produtos ou serviços de uma empresa, mas ainda não foi totalmente qualificado como uma oportunidade de venda. Os leads são o estágio inicial do funil de vendas e exigem nutrição e qualificação antes que possam ser convertidos em oportunidades de venda conjunta ativa com eles.AWS

Os leads compartilhados AWS com parceiros são baseados em sinais de engajamento do cliente, como participação em webinars, participação em campanhas ou consultas sobre soluções de parceiros. Esses leads incluem um contexto valioso, como informações da empresa do cliente, problemas comerciais e detalhes do engajamento, para ajudar os parceiros a priorizar e qualificar oportunidades em potencial.

Trabalhando com convites principais

Os parceiros recebem convites de leads AWS quando clientes em potencial expressam interesse em soluções ou serviços de parceiros. O processo de convite de leads permite que os parceiros avaliem os leads antes de se comprometerem com o engajamento, garantindo a alocação eficiente de recursos e maiores taxas de conversão.

Recebendo convites de leads

AWS cria convites para leads e os compartilha com parceiros por meio da API de vendas. Quando um novo convite de lead está disponível,AWS envia um convite de engajamento com o contexto do lead para parceiros qualificados.

Os parceiros devem monitorar o `Engagement Invitation Created` evento usando a Amazon EventBridge. Essa notificação de evento inclui o `payloadType` campo definido como`LeadInvitation`, permitindo que os parceiros diferenciem os convites para leads dos convites para oportunidades. Ao receber o evento, os parceiros podem recuperar os detalhes do convite para avaliar o lead.

Listando convites para leads

Os parceiros podem ver todos os convites para leads usando a ação da `ListEngagementInvitations` API. Essa ação recupera convites em que o responsável pela chamada é remetente ou destinatário.

Para filtrar especificamente os convites para leads, os parceiros devem usar o `payloadType` filtro com o valor. `LeadInvitation` Esse filtro garante que somente os convites para leads sejam retornados, excluindo os convites para oportunidades dos resultados.

Os convites principais podem estar nos seguintes estados:

- Pendente: aguardando a aceitação ou rejeição do parceiro
- Aceito: o parceiro aceitou o convite e obteve acesso aos detalhes completos do lead
- Rejeitado: o parceiro recusou o convite
- Expirado: o convite ultrapassou a data de validade sem ação

Avaliando convites de leads

Antes de aceitar um convite principal, os parceiros devem recuperar informações detalhadas do convite usando a ação da `GetEngagementInvitation` API. Isso fornece um contexto essencial para tomar decisões informadas de aceitação ou rejeição.

A carga útil do convite principal inclui:

Informações do cliente (pré-aceitação disponível):

- Nome da empresa, setor e país
- URL do site
- Segmento de mercado (corporativo, grande, médio, pequeno, micro)
- AWS nível de maturidade (Avaliação, Single-Account, Multi-Account)

Interações com o cliente:

- Tipo de fonte e informações da campanha
- Ação tomada pelo cliente (preenchimento do formulário, participação no webinar etc.)
- Descrição do problema comercial fornecida pelo cliente
- Categoria de caso de uso
- Título do contato (fornece o contexto da Autoridade para a qualificação do BANT)

> **Important**
>
> As informações de contato do cliente (nome, e-mail, número de telefone) não estão visíveis na fase de convite. Os detalhes de contato ficam disponíveis somente após a aceitação do convite, garantindo a privacidade do cliente e evitando o comportamento de cultivo de chumbo.

Aceitando convites de leads

Quando um parceiro decide buscar um lead, ele deve aceitar o convite usando a ação da `AcceptEngagementInvitation` API. Essa ação adiciona o parceiro ao contrato e concede acesso aos detalhes completos do lead, incluindo informações de contato do cliente.

Após a aceitação:

- O parceiro é adicionado como membro do compromisso
- As informações de contato do cliente se tornam visíveis por meio do contexto do lead de engajamento
- O lead é classificado como Partner Qualified Lead (PQL) em sistemas AWS
- O lead aparece na lista de leads ativos do parceiro
- Os parceiros podem começar a nutrir a liderança e estabelecer contato com o cliente

Os parceiros podem aceitar vários convites de leads de forma programática ligando `AcceptEngagementInvitation` para cada convite, permitindo o processamento eficiente em massa de leads de alta qualidade.

Rejeitando convites principais

Se um lead não estiver alinhado com os recursos, a capacidade ou o foco comercial do parceiro, os parceiros poderão recusar o convite usando a ação da `RejectEngagementInvitation` API. Ao rejeitar um convite de lead, os parceiros devem fornecer um motivo de rejeição para ajudar a AWS melhorar o encaminhamento e a correspondência de leads. Os motivos comuns de rejeição incluem:

- Falta de cobertura geográfica
- Experiência técnica insuficiente para o caso de uso
- Restrições de capacidade atuais

- Incompatibilidade entre o setor de clientes
- Desalinhamento orçamentário

Após a rejeição:

- O lead é removido da fila de convites pendentes do parceiro
- As informações de contato do cliente nunca são compartilhadas com o parceiro
- O lead é classificado como um lead rejeitado pelo parceiro (PRL) nos sistemas AWS
- O convite permanece visível na lista de convites rejeitados do parceiro para referência

Os leads rejeitados podem ser transferidos para outros parceiros, desde que os requisitos de consentimento do cliente sejam atendidos.

Gerenciando leads aceitos

Depois de aceitar um convite para um lead, os parceiros podem acessar informações completas sobre o lead, cultivar relacionamentos com o cliente e acompanhar o lead nas etapas de qualificação.

Visualizando detalhes do lead

Os parceiros acessam todos os detalhes do lead usando a ação `GetEngagement` da API. O contrato contém um contexto de lead com informações abrangentes sobre o cliente e suas interações com AWS.

O contexto do Lead inclui:

Perfil completo do cliente:

- Todas as informações da empresa do convite original
- Detalhes de contato completos para todas as interações com o cliente (nome, e-mail, telefone, cargo comercial)
- AWS nível de maturidade e segmento de mercado

Histórico de interações:

- Vários pontos de contato com o cliente consolidados em um único contrato

- Cada interação inclui informações de origem, ação do cliente, problema comercial e detalhes de contato
- As interações são listadas cronologicamente para fornecer uma visão completa da jornada do cliente

Status de qualificação:

- Estado atual da qualificação do lead (status não qualificado, qualificado, desqualificado ou personalizado)
- Os parceiros podem atualizar esse status à medida que avançam no processo de qualificação

Essa visão abrangente permite que os parceiros entendam o histórico completo de engajamento do cliente em várias AWS campanhas e pontos de contato, em vez de tratar cada interação como um lead separado.

Listando leads ativos

Os parceiros podem recuperar todos os leads ativos usando a ação da `ListEngagements` API com o `contextType` filtro definido como. `Lead` Isso retorna compromissos em que o parceiro é membro e o engajamento contém um contexto de lead.

A resposta da lista inclui informações importantes, como:

- ID de engajamento e título
- Nome da empresa do cliente
- Pontuação de engajamento atual
- Status de qualificação
- Número de interações
- Carimbos de data e hora de criação e modificação

Os parceiros podem usar essa lista para criar painéis, monitorar a integridade do pipeline de leads e identificar leads que precisam de atenção.

Enriquecendo leads com insights de prospecção

Os parceiros podem enriquecer os leads aceitos com AWS insights para melhor priorizá-los e qualificá-los antes de investir em divulgação. O enriquecimento é realizado de forma assíncrona por

meio de tarefas de prospecção, que aumentam o engajamento do lead com sinais derivados para apoiar as decisões AWS de qualificação.

Iniciando uma tarefa de prospecção

Os parceiros iniciam o enriquecimento usando a ação da `StartProspectingFromEngagementTask` API. Essa ação aceita até 100 identificadores de engajamento em uma única solicitação, permitindo que os parceiros enriqueçam leads em lotes. A tarefa é executada de forma assíncrona e retorna imediatamente um `TaskId` e `TaskArn` que os parceiros usam para monitorar o progresso. A inicial `TaskStatus` é uma das `PENDINGIN_PROGRESS`,`COMPLETED`, ou`FAILED`.

Opcionalmente, os parceiros podem fornecer um `TaskName` para rotular a tarefa e um `ClientToken` para garantir a idempotência em todas as novas tentativas.

Sondagem para obter resultados

Como a prospecção é assíncrona, os parceiros fazem uma pesquisa para conclusão usando a ação da `GetProspectingFromEngagementTask` API com o retorno no início. `TaskId` A resposta relata o status geral da tarefa e um resultado por engajamento para cada identificador enviado.

Cada engajamento é processado de forma independente, para que os compromissos individuais possam ser bem-sucedidos ou fracassados sem afetar os outros na mesma tarefa. Para cada engajamento, o resultado inclui:

- Identificador de engajamento: o engajamento que foi processado.
- Status:`PENDING`,`IN_PROGRESS`,`COMPLETED`, ou`FAILED`.
- Identificador de contexto de prospecção: preenchido quando o engajamento é processado com sucesso. Esse identificador faz referência ao contexto de prospecção enriquecido adicionado ao contrato e pode ser usado em operações subsequentes.
- Código e mensagem do motivo: Preenchidos somente quando um compromisso falha, fornecendo um código de falha enumerado e uma descrição legível por humanos com as etapas de recuperação sugeridas.

Monitoramento de várias tarefas

Para monitorar o enriquecimento de muitos leads, os parceiros usam a ação da `ListProspectingFromEngagementTasks` API. Essa ação lista todas as tarefas de prospecção iniciadas pela conta do chamador e oferece suporte a filtros opcionais por identificador de tarefa,

nome da tarefa ou intervalo de horário de início, com classificação configurável. A resposta é paginada; use o `NextToken` valor de cada resposta para recuperar as páginas subsequentes.

Quando o enriquecimento é concluído com sucesso, os AWS insights são adicionados ao contrato como um contexto de prospecção. Os parceiros podem recuperar os detalhes aprimorados `GetEngagement` e usá-los para definir o status de qualificação do lead e decidir quais leads devem avançar para a conversão.

Atualizando as informações do lead

À medida que os parceiros cultivam leads e coletam informações adicionais, eles podem atualizar os detalhes dos leads usando a ação da `UpdateEngagementContext` API. Essa ação permite que os parceiros modifiquem o contexto do lead dentro do engajamento.

Os parceiros podem atualizar:

Status de qualificação: os parceiros devem atualizar o status da qualificação à medida que avançam no processo interno de qualificação:

- Não qualificado: status padrão após aceitar um lead; indica que o lead precisa ser avaliado
- Qualificado: o lead atende aos critérios de qualificação e mostra alto potencial de conversão em uma oportunidade
- Desqualificado: o lead não atende aos critérios ou não é adequado para as soluções do parceiro
- Estados personalizados: os parceiros podem definir seus próprios estados de qualificação de acordo com seus processos internos

Metadados de leads: os parceiros podem adicionar ou atualizar informações adicionais relevantes para seus processos de qualificação e nutrição.

A atualização do status de qualificação ajuda os parceiros a acompanhar a progressão dos leads, gerar relatórios precisos do pipeline e identificar leads prontos para serem convertidos em oportunidades.

Monitoramento AWS Atualizações

AWS pode atualizar as informações do lead com base em novos sinais de engajamento do cliente ou em uma pontuação de engajamento refinada. Quando o engajamento éAWS atualizado, os parceiros recebem um `Engagement Updated` evento via Amazon EventBridge.

Essas atualizações podem incluir:

- Pontuações de engajamento refinadas com base na atividade de novos clientes
- Interações adicionais com clientes a partir de novas campanhas ou pontos de contato
- Informações atualizadas do cliente

Os parceiros devem monitorar esses eventos e ligar `GetEngagement` para obter os detalhes mais recentes do lead. Isso garante que os parceiros tenham as informações mais atuais para priorizar e nutrir leads.

O `Engagement Updated` evento inclui o `contextTypes` campo, permitindo que os parceiros filtrem especificamente os leads (engajamentos com o contexto do lead).

Convertendo um lead em uma oportunidade

Quando um lead é qualificado e demonstra uma intenção de compra séria, os parceiros podem convertê-lo em uma oportunidade de colaboração formal com ele.AWS Essa conversão marca a transição da nutrição de leads (principalmente orientada por parceiros) para o gerenciamento de oportunidades (colaborativo com).AWS

Abordagem recomendada: API de conveniência

Os parceiros podem simplificar o processo de criação e vinculação de oportunidades usando a ação `StartOpportunityFromEngagementTask` conveniente da API. Essa API de tarefas orquestra várias ações automaticamente:

1. Cria um rascunho de oportunidade com informações preliminares do contexto do lead
2. Vincula a oportunidade ao engajamento da fonte por meio de um resumo do recurso
3. Adiciona o CustomerProject contexto ao engajamento

Esse método conveniente reduz o número de chamadas de API necessárias e garante o rastreamento adequado da atribuição. Os parceiros que usam essa abordagem ainda precisam:

- Detalhes completos da oportunidade usando `UpdateOpportunity`
- Soluções de parceiros associados usando `AssociateOpportunity`
- Envie a oportunidade usando `StartEngagementFromOpportunityTask` quando estiver pronto

A API de conveniência é particularmente útil para parceiros com fluxos de trabalho automatizados de lead to opportunity que desejam minimizar a complexidade da integração.

Abordagem alternativa: Step-by-step API

Criando uma oportunidade de rascunho

A primeira etapa na conversão de um lead é criar um rascunho de oportunidade usando a ação da `CreateOpportunity` API. Isso cria uma oportunidade com o `Lifecycle.ReviewStatus` conjunto para`Pending Submission`. Nesse estágio, a oportunidade ainda não foi enviada AWS para validação.

Os parceiros devem preencher a oportunidade com informações coletadas durante a formação de leads, incluindo:

- Detalhes da conta do cliente
- Informações do projeto e problema comercial
- Gastos esperados do cliente
- Data de fechamento prevista
- Quaisquer outros detalhes relevantes coletados durante a qualificação

O rascunho da oportunidade permite que os parceiros preparem informações completas antes de enviá-las AWS, garantindo maiores taxas de aprovação e validação mais rápida.

Vinculando oportunidades ao engajamento de leads

Depois de criar a oportunidade de rascunho, os parceiros devem vinculá-la ao engajamento do lead de origem usando a ação da `CreateResourceSnapshot` API. Essa etapa é fundamental para:

- Rastreamento de atribuição: estabelece a cadeia de proveniência desde o convite do lead até o fechamento da oportunidade, permitindo o cálculo preciso do ROI para campanhas de marketing e fontes de leads.
- Métricas de conversão: permite que AWS nossos parceiros meçam as taxas de conversão de leads em oportunidades e identifiquem fontes de leads bem-sucedidas.
- Preservação do contexto: mantém o histórico completo da jornada do cliente, incluindo todas as interações originais e sinais de engajamento.

Ao criar o instantâneo do recurso, os parceiros especificam:

- O ID do engajamento (engajamento do lead de origem)
- O identificador da oportunidade

- O tipo de recurso (Oportunidade)

Essa ação adiciona automaticamente um CustomerProject contexto ao engajamento, sinalizando que o lead foi convertido em uma oportunidade ativa. O engajamento agora contém o contexto original do lead (preservando o histórico) e o novo CustomerProject contexto (indicando uma oportunidade ativa).

Preenchendo os detalhes da oportunidade

Depois que a oportunidade for criada e vinculada, os parceiros devem preencher os requisitos adicionais da oportunidade antes do envio. Consulte a ação `AssociateOpportunity` da API para obter detalhes.

Atualização dos detalhes do projeto: os parceiros podem usar a ação da `UpdateOpportunity` API para refinar as informações do projeto, os problemas comerciais do cliente, os gastos esperados e outros detalhes relevantes coletados durante a qualificação do lead.

Enviando a oportunidade

Depois que todas as informações necessárias forem preenchidas, os parceiros enviarão a oportunidade de AWS validação usando a API de `StartEngagementFromOpportunityTask` conveniência. Isso faz a transição da oportunidade do status de rascunho para o processo de AWS revisão.

Requisito de validação: o contrato deve ter um CustomerProject contexto antes do envio. Esse contexto é criado automaticamente quando usado `CreateResourceSnapshot` para vincular a oportunidade ao engajamento. Se esse contexto estiver ausente, o envio falhará.

Após o envio:

- A oportunidade `Lifecycle.ReviewStatus` muda para `Submitted`
- O lead é classificado como Partner Sales Qualified Lead (PSQL) em sistemas AWS
- AWS inicia a validação para garantir que os detalhes da oportunidade sejam precisos e completos
- Nenhuma alteração pode ser feita na oportunidade até que o processo de análise seja concluído

A oportunidade então segue o fluxo de trabalho de validação padrão descrito na documentação “Trabalhando com suas oportunidades”, progredindo por estados como Ação necessária In-Review, Aprovada ou Desqualificada.

Trabalho com insights de dimensionamento de negócios

O que é Deal Sizing?

O dimensionamento de negócios é um recurso das oportunidades de engajamento do cliente (ACE) da APN na Central de AWS parceiros que usa IA para fornecer estimativas automatizadas do tamanho do negócio e recomendações de AWS serviços quando os parceiros criam ou atualizam oportunidades.

Estimativas de MRR previstas pela IA

Os insights de dimensionamento de negócios fornecem a mediana das faixas de receita recorrente mensal (MRR) previstas com base nas AWS oportunidades históricas, nos casos de uso e nas descrições comerciais dos parceiros. Os parceiros devem revisar e atualizar o MRR total à medida que os parceiros coletam mais informações sobre o negócio e o progresso no ciclo de vendas.

AI-Recommended AWS Produtos

AI-recommended os produtos são identificados com base nas descrições dos problemas comerciais do cliente e nos detalhes da oportunidade. A IA recomenda AWS produtos relevantes alinhados aos requisitos técnicos e aos casos de uso. Os parceiros devem analisar essas sugestões e personalizar a seleção de produtos para atender às necessidades específicas do cliente.

Insights aprimorados com URLs da calculadora de preços

Os parceiros também podem importar URLs da Calculadora de AWS Preços para preencher automaticamente as seleções de serviços e receber informações aprimoradas, incluindo indicadores de elegibilidade do Migration Acceleration Program (MAP), recomendações de otimização e análise de possíveis economias de custos.

Entendendo Source-Separated os dados

O Deal Sizing fornece informações de duas fontes distintas para oferecer uma visibilidade abrangente do valor da oportunidade:

- AWS Fonte: AI-recommended produtos baseados em padrões de oportunidades históricas semelhantes
- Fonte do parceiro: Suas próprias estimativas importadas da Calculadora de AWS Preços

Lidando com a variação entre fontes

Quando as estimativas fornecidas pelo parceiro diferem significativamente das AWS recomendações, os parceiros devem:

1. Analise os detalhes da oportunidade para garantir que todas as informações relevantes sejam capturadas
2. Verifique se as configurações de URL da Calculadora de Preços correspondem aos requisitos reais
3. Considere AWS as recomendações como pontos de dados informados por oportunidades históricas semelhantes
4. Tome decisões informadas com base no contexto e nos requisitos específicos do cliente

Acessando insights de dimensionamento de negócios

Os parceiros acessam os dados de dimensionamento de negócios por meio da ação de `GetAwsOpportunitySummary` API, que retorna um `AwsOpportunitySummary` objeto estendido contendo previsões de dimensionamento de negócios, recomendações de produtos e insights de otimização.

Quando os dados de dimensionamento de negócios se tornam disponíveis

Os insights de dimensionamento de negócios ficam disponíveis após a criação de uma oportunidade com detalhes suficientes do cliente e do projeto. O sistema analisa os atributos da oportunidade e gera:

1. MRR previsto por IA: calculado automaticamente com base nas características da oportunidade
2. AWS recomendações de produtos: serviços relevantes para o problema comercial e os requisitos técnicos do cliente
3. Informações aprimoradas (quando o URL da Calculadora de Preços é fornecido): elegibilidade do MAP, recomendações de otimização e análise de economia de custos

> **Note**
>
> Os insights de dimensionamento de negócios não estão disponíveis para todas as oportunidades. A elegibilidade depende de fatores como o histórico do parceiro e a integridade dos detalhes da oportunidade fornecidos.

Fornecendo insumos de dimensionamento de negócios

Os parceiros fornecem informações para o dimensionamento de negócios por meio de ações `CreateOpportunity` de `UpdateOpportunity` API. O sistema usa atributos de oportunidade para gerar recomendações.

Fornecimento do valor total do contrato (TCV)

Os parceiros que estimam o valor total do contrato em vez da receita mensal recorrente podem fornecer o valor do negócio diretamente. A AWS converte automaticamente o TCV em AWS MRR usando AI-powered modelos de conversão.

Para o dimensionamento de negócios no TCV, forneça `ExpectedContractDuration` como `Months` (os valores válidos variam de 1 a 144 meses) e `TargetCompany` como `Self` e `Frequency` como. `None`

```
{
  "Project": {
    "ExpectedContractDuration": { "Term": "Months", "Value": "36" },
    "ExpectedCustomerSpend": [
      {
        "Amount": "1200000.00",
        "CurrencyCode": "USD",
        "Frequency": "None",
        "TargetCompany": "Self"
      }
    ]
  }
}
```

A AWS deriva a estimativa do AWS MRR e retorna as duas entradas em: `GetOpportunity`

```
{
  "Project": {
    "ExpectedContractDuration": { "Term": "Months", "Value": "36" },
    "ExpectedCustomerSpend": [
      {
        "Amount": "5600.00",
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      },
```

```
      {
        "Amount": "1200000.00",
        "CurrencyCode": "USD",
        "Frequency": "None",
        "TargetCompany": "Self"
      }
    ]
  }
}
```

No exemplo acima, a `ExpectedCustomerSpend` primeira matriz contém a estimativa do AWS MRR (`TargetCompany: "AWS"`,`Frequency: "Monthly"`) e a segunda matriz é o valor total do contrato do parceiro (`TargetCompany: "Self"`,`Frequency: "None"`).

Regras de validação do TCV

Todos os erros de validação do TCV retornam o código de erro `INVALID_VALUE` com`Reason: "BUSINESS_VALIDATION_FAILED"`. O fator distintivo é o `FieldName` e`Message`:

Condição	Campo	Mensagem
Entrada automática sem `ExpectedContractDuration`	`project.expectedContractDuration`	ExpectedContractDuration é necessário quando uma entrada Self está presente
`ExpectedContractDuration` sem entrada automática (somente para criar)	`project.expectedCustomerSpend`	Uma entrada automática é necessária quando ExpectedContractDuration está presente
Várias entradas próprias	`project.expectedCustomerSpend`	Somente uma entrada automática é permitida
Inválido TargetCompany quando a duração está presente	`project.expectedCustomerSpend.targetCompany`	TargetCompany deve ser 'AWS' ou 'Self'
Incompatibilidade de moedas entre AWS e Self	`project.expectedCustomerSpend.currencyCode`	CurrencyCode deve corresponder entre as entradas

Condição	Campo	Mensagem
Entrada automática + EstimationUrl em conjunto	`project.expectedCustomerSpend.estimationUrl`	Entrada automática e EstimationUrl não pode coexistir

> **Note**
>
> Ativado`UpdateOpportunity`, se `ExpectedContractDuration` estiver presente sem uma entrada Self, mas o valor do negócio do parceiro existir no armazenamento, o sistema reconstrói a entrada Self automaticamente — nenhum erro é gerado.

Alternando entre TCV e MRR manual

Para voltar do modo TCV para o MRR manual, `ExpectedContractDuration` defina explicitamente `null` e forneça somente uma entrada: `AWS`

```
{
  "Project": {
    "ExpectedContractDuration": null,
    "ExpectedCustomerSpend": [
      {
        "Amount": "8000.00",
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  }
}
```

> **Note**
>
> Omitir `ExpectedContractDuration` da carga útil (não enviar o campo) significa “sem alteração” — os dados existentes do TCV são preservados. Enviar explicitamente `null` significa “limpar o modo TCV”.

URLs da calculadora de preços e TCV

Se um parceiro fornecer uma `Self` entrada (modo TCV) e uma`EstimationUrl`, a API retornará um erro de validação. Entrada automática e EstimationUrl não pode coexistir na mesma solicitação.

Fornecendo estimativas manuais de MRR

Os parceiros podem inserir manualmente as estimativas de MRR usando o `Project.ExpectedCustomerSpend[].Amount` campo:

> **Note**
>
> O `CurrencyCode` campo aceita `USD` `EUR` e. Se a partição da AWS for `aws-eusc` (AWS European Sovereign Cloud), o código da moeda deverá ser`EUR`.

```
{
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": "25000.00",
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  }
}
```

Esse campo está disponível para todos os parceiros e pode ser usado para aceitar previsões de IA definindo o mesmo valor ou Partner-provided substituindo por valores.

Fornecendo URLs da calculadora de preços

Opcionalmente, os parceiros podem fornecer URLs da Calculadora de AWS Preços por meio do `Project.ExpectedCustomerSpend[].EstimationUrl` campo para importar automaticamente as configurações do serviço e receber informações aprimoradas, incluindo indicadores de elegibilidade do MAP, recomendações de otimização e análise de economia de custos.

```
{
```

```
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": null,
        "CurrencyCode": "USD",
        "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  }
}
```

Quando um URL válido da Calculadora de Preços é fornecido, o sistema calcula automaticamente o valor do MRR a partir da configuração da calculadora e preenche o campo Valor e os dados detalhados de gastos em nível de produto na fonte do Parceiro de. `AwsProductsSpendInsightsBySource` Os parceiros devem definir o valor como nulo ao fornecer um EstimationUrl - o sistema o calculará automaticamente.

URLs em formatos inválidos e URLs que não puderem ser resolvidos serão rejeitados com um. `ValidationException`

Como quantidade e EstimationUrl trabalho em conjunto

Você tem flexibilidade na forma como fornece estimativas mensais de receita recorrente (MRR) para oportunidades. A API lida de forma inteligente com várias combinações de valores manuais e URLs da Calculadora de AWS Preços para garantir que suas oportunidades sempre possam ser criadas com sucesso.

Usando apenas uma quantidade manual

Quando você fornece somente o campo Valor, a API salva sua estimativa manual e define EstimationUrl como nula. Essa abordagem é útil quando você tem uma estimativa de MRR específica a partir de conversas com clientes ou de seus próprios cálculos.

Exemplo de solicitação:

```
{
    "Project": {
        "ExpectedCustomerSpend": [{
            "Amount": "25000.00",
```

```
            "CurrencyCode": "USD",
            "Frequency": "Monthly",
            "TargetCompany": "AWS"
        }]
    }
}
```

Usando apenas um AWS URL da calculadora de preços

Quando você fornece somente o EstimationUrl campo:

- URL válido — A API calcula o valor do MRR a partir da configuração da calculadora e salva o URL e o valor calculado.
- URL inválida — A API retorna um ValidationException com detalhes sobre por que a URL é inválida.

Exemplo de solicitação:

```
{
    "Project": {
        "ExpectedCustomerSpend": [{
            "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
            "CurrencyCode": "USD",
            "Frequency": "Monthly",
            "TargetCompany": "AWS"
        }]
    }
}
```

Exemplo de resposta de erro (URL inválido):

```
{
    "ErrorList": [{
        "Code": "INVALID_VALUE",
        "FieldName": "project.expectedCustomerSpend.estimationUrl",
        "Message": "Invalid Estimation URL: https://calculator.aws/#/estimate?
id=invalid"
    }],
    "Message": "The provided Estimation URL is invalid",
    "Reason": "INVALID_VALUE"
}
```

Fornecendo tanto o valor quanto EstimationUrl

Quando você fornece os dois campos, a API prioriza a garantia de que sua oportunidade seja criada com sucesso:

1. Se ambos os valores permanecerem inalterados — nenhuma atualização será executada nesses campos
2. O URL é inválido — a API salva o valor fornecido e o define EstimationUrl como nulo, permitindo que a criação da oportunidade continue.
3. O URL é válido e o valor calculado corresponde ao valor fornecido — Ambos os valores são salvos.
4. O URL é válido, mas o valor calculado não corresponde ao valor fornecido — a API salva o valor fornecido e o define EstimationUrl como nulo sem retornar um erro.

> **Note**
>
> Quando os dois valores não coincidem com os existentes, a API primeiro tenta calcular o valor a partir do URL. Seu valor fornecido manualmente sempre tem precedência quando há um problema de incompatibilidade ou validação.

Principal benefício: URLs inválidos nunca bloqueiam a criação de oportunidades

Essa abordagem garante que URLs inválidos ou inacessíveis da Calculadora de AWS Preços nunca impeçam você de criar ou atualizar oportunidades. Seu valor fornecido manualmente sempre tem precedência quando há um problema de conflito ou validação, dando a você controle total sobre os dados da oportunidade.

Entendendo o modelo AwsOpportunitySummary de dados estendido

Esta seção descreve a estrutura de resposta retornada pela ação `GetAwsOpportunitySummary` da API.

> **Note**
>
> AI-forecasted O MRR é devolvido em `Project.ExpectedCustomerSpend[].Amount`

A ação da `GetAwsOpportunitySummary` API retorna insights de dimensionamento de negócios por meio da nova `Insights.AwsProductsSpendInsightsBySource` estrutura que fornece uma análise abrangente de gastos com atribuição de origem.

Separação de fontes:AWS versus dados de parceiros

O Deal Sizing separa os insights por fonte para fornecer transparência e evitar a dupla contagem:

- AWS Fonte: recomendações de produtos de IA da AWS
- Fonte do parceiro: dados de gastos e detalhes do produto importados dos URLs da calculadora de preços fornecidos pelo parceiro

Essa separação permite que os parceiros:

1. Compare suas estimativas com as AWS recomendações
2. Valide suas suposições de dimensionamento
3. Evite inflar os totais combinando acidentalmente as duas fontes
4. Mantenha a visibilidade da proveniência dos dados

Visão geral da estrutura

```
{
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "<AWS | Partner>": {
        "CurrencyCode": "string",
        "Frequency": "string",
        "TotalAmount": "string",
        "TotalOptimizedAmount": "string",
        "TotalPotentialSavingsAmount": "string",
        "TotalAmountByCategory": { },
        "AwsProducts": [ ]
      }
    }
  },
  "Project": {
    "ExpectedCustomerSpend": [ ]
  },
  "RelatedEntityIds": {
    "AwsProducts": [ ]
```

```
  }
}
```

Referência de campo

AwsProductsSpendInsightsBySource Mapa

Source-separated gaste insights que forneçam análises independentes para AWS recomendações e estimativas de parceiros.

O `Insights.AwsProductsSpendInsightsBySource` campo é um mapa contendo até duas chaves:

Chave	Tipo	Description
AWS	AwsProductsSpendInsights	AI-generated insights, incluindo produtos recomenda dos da AWS
Parceiro	AwsProductsSpendInsights	Partner-sourced insights derivados dos URLs da calculadora de preços, incluindo otimizações e custos detalhados do serviço

> **Note**
>
> Somente fontes com dados disponíveis são incluídas na resposta. As novas oportunidades geralmente começam com apenas a AWS fonte preenchida.

AwsProductInsights Objeto

Análise abrangente de gastos para uma única fonte (AWS ou parceiro), incluindo valores totais, economias de otimização, detalhamentos de categorias de programas e insights detalhados em nível de produto.

Cada objeto de origem contém os seguintes campos:

Campo	Tipo	Description
CurrencyCode	string	Código monetário ISO 4217. Os valores suportados são “USD” e “EUR”.
Frequência	string	Indica com que frequência se espera que o cliente gaste o valor projetado. Somente o valor Mensal é permitido para o campo Frequência, representando o gasto mensal recorrente.
TotalAmount	string	Gasto total estimado para essa fonte antes das otimizações
TotalOptimizedAmount	string	Gasto total estimado após a aplicação das otimizações recomendadas
TotalPotentialSavingsAmount	string	Economias quantificadas alcançáveis por meio da implementação de otimizaçõ es
TotalAmountByCategory	objeto	Valores de gastos mapeados para AWS programas e caminhos de modernização
AwsProducts	array	Product-level detalhes, incluindo recomendações de custos e otimização

Limitação do estado atual: Para a AWS fonte, TotalAmount, TotalOptimizedAmount, e TotalPotentialSavingsAmount pode ser omitida quando as recomendações

de gastos por produto não estão disponíveis. Os parceiros devem consultar `Project.ExpectedCustomerSpend[].Amount` a estimativa total de AWS MRR nesses casos.

TotalAmountByCategory Objeto

Os gastos do Maps são equivalentes a AWS programas e iniciativas estratégicas:

Chave de categoria	Description
Elegível para MA	Gastos em serviços elegíveis para financiamento do Migration Acceleration Program
Elegível para CA	Serviços de suporte à alocação e rastreamento de custos
Pathway - Mude para a nuvem nativa	Serviços alinhados com a modernização nativa da nuvem
Pathway - Mude para Managed Services	Serviços de apoio à adoção de serviços gerenciados
Pathway - Mude para o código aberto	Alternativas de serviço de código aberto
Pathway - Mova para contêineres	Container-based opções de serviço
Pathway - Mude para o modo sem servidor	Serviços de computação sem servidor
Pathway - Migre para bancos de dados	Serviços de modernização de banco de dados
Pathway - Mude para a análise	Serviços de análise e processamento de dados

> **Note**
>
> Os valores totais das categorias podem exceder TotalAmount porque os produtos individuais podem pertencer a várias categorias.

AwsProducts Matriz

Lista de AWS serviços com indicadores de elegibilidade do programa (MAP, caminhos de modernização), estimativas de custo e recomendações de otimização.

Cada objeto do produto contém:

Campo	Tipo	Obrigatório	Descrição
ProductCode	string	Sim	AWS Identificador de produto do Partner Central usado para associação de oportunidades
ServiceCode	string	Não	Código de serviço da calculadora de preços (links para o URL original da calculado ra)
Categorias	array	Sim	Lista de categoria s de programas e caminhos para os quais este produto é elegível
Valor	string	Não	Custo básico do serviço antes das otimizações (pode ser nulo para as recomendações fornecidas)AWS
OptimizedAmount	string	Não	Custo do serviço após a aplicação das otimizações (pode ser nulo para AWS

Campo	Tipo	Obrigatório	Descrição
			as recomendações fornecidas)
PotentialSavingsAmount	string	Não	Service-specific redução de custos por meio de otimizações (pode ser nula para AWS recomendações fornecidas)
Otimizações	array	Não	Lista de recomendações de otimização específicas para este produto

Tratamento nulo: para produtos AWS de origem, o valor e os PotentialSavingsAmount campos podem ser nulos quando as recomendações de gastos por produto não estão disponíveis. OptimizedAmount Em `Project.ExpectedCustomerSpend[0].Amount` vez disso, o sistema fornece MRR total por meio de.

Objeto de otimização

Cada recomendação de otimização contém:

Campo	Tipo	Description
Description	string	Human-readable explicação da estratégia de otimização
SavingsAmount	string	Economia de custos quantificada alcançável com a implementação dessa otimização

Exemplos

Exemplo 1:AWS Somente recomendações (estado atual)

Este exemplo mostra a resposta típica quando somente recomendações de IA estão disponíveis e os valores de gastos por produto ainda não podem ser recomendados.


```
{
  "Catalog": "AWS",
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "AWS": {
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "AwsProducts": [
          {
            "ProductCode": "AmazonEC2",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "AWSLambda",
            "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
 Native"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "AmazonRDS",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
 to Managed Services"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          }
        ]
      }
    }
```

```
  },
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": "28000.00",
        "CurrencyCode": "USD",
        "EstimationUrl": null,
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  },
  "RelatedEntityIds": {
    "AwsProducts": [
      "AmazonEC2",
      "AWSLambda",
      "AmazonRDS"
    ]
  }
}
```

Pontos-chave: a AWS fonte mostra produtos recomendados com categorias, mas sem quantidades por produto. Estimativa total de MRR disponível via`Project.ExpectedCustomerSpend[].Amount`.

Exemplo 2: Importação da calculadora de preços de parceiros

Este exemplo mostra insights aprimorados quando um parceiro fornece um URL válido da Calculadora de Preços.

```
{
  "Catalog": "AWS",
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "Partner": {
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TotalAmount": "32500.00",
        "TotalOptimizedAmount": "30000.00",
        "TotalPotentialSavingsAmount": "2500.00",
        "TotalAmountByCategory": {
          "MAP Eligible": "22500.00",
          "Cost Allocation Tagging": "32500.00",
```

```
          "Pathway - Move to Cloud Native": "5000.00",
          "Pathway - Move to Managed Services": "7500.00"
        },
        "AwsProducts": [
          {
            "ServiceCode": "ec2",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": "15000.00",
            "OptimizedAmount": "13500.00",
            "PotentialSavingsAmount": "1500.00",
            "Optimizations": [
              {
                "Description": "Convert to 3-year reserved instances",
                "SavingsAmount": "1000.00"
              },
              {
                "Description": "Migrate to Graviton-based instances",
                "SavingsAmount": "500.00"
              }
            ]
          },
          {
            "ServiceCode": "lambda",
            "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
            "Amount": "5000.00",
            "OptimizedAmount": "5000.00",
            "PotentialSavingsAmount": "0.00",
            "Optimizations": []
          },
          {
            "ServiceCode": "AmazonRDS",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
            "Amount": "7500.00",
            "OptimizedAmount": "6500.00",
            "PotentialSavingsAmount": "1000.00",
            "Optimizations": [
              {
                "Description": "Optimize Multi-AZ for dev/test environments",
                "SavingsAmount": "1000.00"
              }
            ]
          },
```

```
          {
            "ServiceCode": "AmazonS3",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": "5000.00",
            "OptimizedAmount": "5000.00",
            "PotentialSavingsAmount": "0.00",
            "Optimizations": []
          }
        ]
      }
    }
  },
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": "32500.00",
        "CurrencyCode": "USD",
        "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  },
  "RelatedEntityIds": {
    "AwsProducts": []
  }
}
```

Pontos principais: a fonte do parceiro contém detalhes completos dos gastos com valores por produto. As recomendações de otimização fornecem estratégias práticas de redução de custos. Os detalhamentos das categorias mostram a elegibilidade para o MAP ($22.500 de um total de $32.500). Os produtos incluem ServiceCode links para a configuração da Calculadora de Preços.

Exemplo 3: Ambas as fontes presentes (cenário de comparação)

Este exemplo demonstra como as AWS recomendações e as estimativas dos parceiros aparecem juntas, permitindo a comparação.

```
{
  "Catalog": "AWS",
  "Insights": {
    "AwsProductsSpendInsightsBySource": {
      "AWS": {
```

```
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "AwsProducts": [
          {
            "ProductCode": "AmazonEC2",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "AWSLambda",
            "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "EBS",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          },
          {
            "ProductCode": "RDS",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
            "Amount": null,
            "OptimizedAmount": null,
            "PotentialSavingsAmount": null,
            "Optimizations": []
          }
        ]
      },
      "Partner": {
        "CurrencyCode": "USD",
        "Frequency": "Monthly",
        "TotalAmount": "32500.00",
        "TotalOptimizedAmount": "30000.00",
```

```
        "TotalPotentialSavingsAmount": "2500.00",
        "TotalAmountByCategory": {
          "MAP Eligible": "22500.00",
          "Cost Allocation Tagging": "32500.00",
          "Pathway - Move to Cloud Native": "5000.00",
          "Pathway - Move to Managed Services": "7500.00"
        },
        "AwsProducts": [
          {
            "ServiceCode": "ec2",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": "15000.00",
            "OptimizedAmount": "13500.00",
            "PotentialSavingsAmount": "1500.00",
            "Optimizations": [
              {
                "Description": "Convert to 3-year reserved instances",
                "SavingsAmount": "1000.00"
              }
            ]
          },
          {
            "ServiceCode": "lambda",
            "Categories": ["Cost Allocation Tagging", "Pathway - Move to Cloud
Native"],
            "Amount": "5000.00",
            "OptimizedAmount": "5000.00",
            "PotentialSavingsAmount": "0.00",
            "Optimizations": []
          },
          {
            "ServiceCode": "amazonRDS",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging", "Pathway - Move
to Managed Services"],
            "Amount": "7500.00",
            "OptimizedAmount": "6500.00",
            "PotentialSavingsAmount": "1000.00",
            "Optimizations": [
              {
                "Description": "Optimize Multi-AZ for dev/test environments",
                "SavingsAmount": "1000.00"
              }
            ]
          },
```

```
          {
            "ServiceCode": "amazonS3",
            "Categories": ["MAP Eligible", "Cost Allocation Tagging"],
            "Amount": "5000.00",
            "OptimizedAmount": "5000.00",
            "PotentialSavingsAmount": "0.00",
            "Optimizations": []
          }
        ]
      }
    }
  },
  "Project": {
    "ExpectedCustomerSpend": [
      {
        "Amount": "28000.00",
        "CurrencyCode": "USD",
        "EstimationUrl": "https://calculator.aws/#/estimate?id=abc123",
        "Frequency": "Monthly",
        "TargetCompany": "AWS"
      }
    ]
  },
  "RelatedEntityIds": {
    "AwsProducts": [
      "AmazonEC2",
      "AWSLambda",
      "EBS",
      "RDS"
    ]
  }
}
```

Pontos-chave: Ambas as fontes apresentam:AWS recomendações (total de 28 mil dólares) e calculadora de parceiros (total de 32,5 mil dólares).AWS as recomendações mostram 4 produtos; a calculadora de parceiros mostra 4 produtos (3 sobrepostos). Os parceiros podem comparar AWS as recomendações com suas estimativas detalhadas.AWS total disponível via`Project.ExpectedCustomerSpend[].Amount`.

Práticas recomendadas

Tratamento de valores nulos

1. EstimationUrl Campo: EstimationUrl Espere ser nulo para a maioria dos parceiros. Esse é um comportamento padrão e não deve ser tratado como um erro. Exiba as opções de entrada manual do MRR como o caminho principal.
2. Categorias ausentes: se a AWS fonte não TotalAmountByCategory estiver presente, isso indica que as recomendações por produto não estão disponíveis. Os detalhamentos de categorias estão disponíveis para Partner-sourced dados quando os URLs da Calculadora de Preços são fornecidos.

Recomendações de otimização

1. Exiba insights acionáveis: mostre descrições de otimização e valores de economia de forma proeminente para ajudar os parceiros a entender as oportunidades de redução de custos.
2. Economia agregada: soma os PotentialSavingsAmount produtos para mostrar o potencial total de otimização.
3. Priorize por impacto: classifique as otimizações por SavingsAmount para ajudar os parceiros a se concentrarem nas recomendações de maior impacto.

Monitoramento de atualizações

1. Pesquisa periódica: faça pesquisas **`GetAwsOpportunitySummary`** periódicas (recomendado: a cada 5 minutos) durante os fluxos de trabalho de criação de oportunidades.
2. Lidar com a geração assíncrona: os insights de dimensionamento de negócios podem não estar imediatamente disponíveis após a criação da oportunidade. Implemente os estados de carregamento apropriados e a lógica de repetição.

Padrões de integração de API

1. Aproveite as integrações existentes: os parceiros que já estão ligando `GetAwsOpportunitySummary` só precisam consumir campos adicionais na resposta, sem a necessidade de novas chamadas de API.
2. Degradação elegante: crie interfaces de usuário para lidar com cenários em que os dados de dimensionamento de negócios ainda não estejam disponíveis ou estejam incompletos.

Qualidade e precisão dos dados

1. Valide os URLs da calculadora de preços: certifique-se de que os URLs sejam válidos antes do envio para evitar respostas nulas. EstimationUrl
2. Forneça detalhes detalhados de oportunidades: informações mais completas sobre oportunidades (detalhes do cliente, descrição do projeto, requisitos técnicos) geram recomendações de IA mais precisas.
3. Analise a variação: quando as estimativas do parceiro diferirem significativamente das AWS recomendações, analise os detalhes da oportunidade para garantir que todo o contexto relevante seja capturado.

Práticas recomendadas

Reagindo aos eventos

Ao lidar com eventos da API do AWS Partner Central, certifique-se de que sua lógica de processamento seja idempotente para lidar com eventos duplicados. Em vez de fazer GetOpportunitychamadas imediatas para cada evento, considere agrupar ou buscar seletivamente os detalhes com base nas necessidades do seu aplicativo. Para operações ininterruptas, cuidado com as cotas.

Implementando um bloqueio otimista

O bloqueio otimista evita a substituição não intencional de dados durante atualizações simultâneas. Aqui está um cenário típico:

1. O parceiro recupera uma oportunidade de seu sistema de CRM.
2. O usuário `A` atualiza a oportunidade no AWS Partner Central.
3. O usuário `B` atualiza a mesma oportunidade ao mesmo tempo por meio da integração do CRM.
4. Se os dados mudarem, o sistema CRM tentará fazer o upload dos dados, mas retornará um`ConflictException`.
5. O usuário analisa o erro e resolve manualmente os dados conflitantes.

Para evitar esse cenário, todas as UpdateOpportunitysolicitações devem incluir o `LastModifiedDate` parâmetro, que você pode obter das GetOpportunityações anteriores CreateOpportunityUpdateOpportunity, e. A atualização será bem-sucedida somente se

`LastModifiedDate` corresponder ao nosso sistema. Caso contrário, você deve buscar a versão mais recente `LastModifiedDate` usando GetOpportunitye tentar novamente a atualização.

Sincronizando dados entre CRM e AWS Central de Parceiros

É essencial manter seu sistema sincronizado com os dados mais recentes do Partner Central. A seguir estão duas estratégias para garantir que seu sistema reflita os dados mais recentes:

Usando eventos (recomendado)

1. Carregue dados usando ListOpportunities.
2. Inscreva-se em eventos de oportunidades.
3. Responda a novas oportunidades ou mudanças.
   - Quando você recebe `Opportunity Created``Opportunity Updated`, ou `Engagement Invitation Accepted` eventos, use `GetOpportunity` para buscar os dados mais recentes.
   - Ao receber `Engagement Invitation Rejected` eventos, remova as oportunidades correspondentes.

Usando a ListOpportunities enquete

1. Carregue dados usando ListOpportunities.
2. Escolha uma frequência de pesquisa, garantindo que ela não seja muito frequente para evitar esgotar sua cota de leitura diária.
3. Identifique os dados mais recentes `LastModifiedDate` de seus dados armazenados, garantindo que sejam originários de AWS.
4. Use o timestamp no `AfterLastModifiedDate` filtro ao ligar. ListOpportunities

```
{
   "FilterList": [
       {
           "Name": "AfterLastModifiedDate",
           "ValueList": [ "2023-05-01T20:37:46Z" ] // Replace with actual timestamp
 of your last synced data
        }
   ]
}
```

5. AWS retornará oportunidades criadas ou atualizadas após o valor indicado no timestamp.

6. Repita todas as páginas retornadas usando `NextToken` e atualize os dados do seu sistema usando GetOpportunity.

```
{
   "NextToken": "AAMA-EFRSN...PZa942D",
   "FilterList": [
       {
           "Name": "AfterLastModifiedDate",
           "ValueList": [ "2023-05-01T20:37:46Z" ] // Replace with actual timestamp
 of your last synced data
        }
   ]
}
```

Usar o AWS API de benefícios do Partner Central

A API de benefícios do AWS Partner Central simplifica a forma como os parceiros descobrem, se inscrevem e gerenciam benefícios em todos os programas de AWS parceria. Ao centralizar o gerenciamento de benefícios em uma interface única e intuitiva, o serviço cria uma meritocracia transparente que recompensa os parceiros e simplifica as operações.

A API fornece funcionalidade de AWS API padrão. Acesse-o usando ações de API ou usando um AWS SDK personalizado para sua linguagem de programação ou plataforma. Para obter mais informações, consulte Introdução AWS e Ferramentas para desenvolver AWS.

O que é um benefício?

Na Rede de AWS Parceiros (APN), um benefício representa uma vantagem ou capacidade que um parceiro pode obter AWS para apoiar o crescimento de seus negócios e o sucesso de seus clientes. Os benefícios são projetados para recompensar os parceiros com base em suas qualificações e contribuições, ao mesmo tempo em que fornecem valor tangível que ajuda os parceiros a ampliar suas AWS práticas.

Os benefícios podem assumir várias formas, incluindo:

- Incentivos financeiros — créditos, desembolsos em dinheiro e programas de financiamento, como o Migration Acceleration Program (MAP) ou o Marketing Development Funds (MDF)
- Benefícios de acesso — pré-visualize o acesso a novos serviços, ferramentas ou recursos especializados

- Reconhecimento — Emblemas digitais, certificações, designações de competência e status de nível de parceiro
- Recursos técnicos — Assistência na arquitetura da solução, suporte técnico e ProServe contratos
- Suporte de marketing — Co-marketing oportunidades, status de parceiro em destaque e publicações de histórias de sucesso

Descobrindo os benefícios disponíveis

Os parceiros iniciam sua jornada de benefícios descobrindo quais benefícios estão disponíveis para eles e entendendo os requisitos de elegibilidade. A API Benefits fornece recursos abrangentes de descoberta por meio das APIs Benefit Discovery.

Listando os benefícios disponíveis

Os parceiros podem ver todos os benefícios disponíveis usando a ação `ListBenefits` da API. Essa ação recupera um catálogo de benefícios com recursos de filtragem opcionais para ajudar os parceiros a encontrar oportunidades relevantes rapidamente.

Para descobrir os benefícios com eficiência, os parceiros podem filtrar por:

- Programa - Filtre por programas de AWS parceiros específicos, como MAP (Migration Acceleration Program), MDF (Marketing Development Funds), ISV Workload Migration, Innovation Sandbox, Proof of Concept ou Partner Initiative Funding
- Tipo de atendimento - Filtre pela forma como os benefícios são entregues: `CREDITSCASH`,`DISCOUNT`,, `ACCESSRECOGNITION`, ou `RESOURCE`
- Status - Filtrar por status de benefício: `ACTIVE` ou `DEPRECATED`

A `ListBenefits` API retorna resumos de benefícios contendo informações essenciais, como ID do benefício, nome, descrição, programas associados e tipos de atendimento. Essa visão resumida permite que os parceiros examinem rapidamente os benefícios disponíveis e identifiquem os mais relevantes para seus objetivos de negócios.

Exemplo de abordagem de filtragem:

Parceiros focados em incentivos financeiros podem filtrar por tipos de atendimento `CREDITS` e `CASH` ver oportunidades de financiamento. Os parceiros que buscam capacitação técnica podem filtrar por tipo de atendimento `RESOURCE` ou `ACCESS` encontrar benefícios de suporte técnico e acesso a ferramentas.

Visualizando informações detalhadas sobre benefícios

Depois que um parceiro identifica um benefício de interesse, ele pode recuperar detalhes completos usando a ação da `GetBenefit` API. Isso fornece informações abrangentes, incluindo:

- Critérios de elegibilidade — Requisitos detalhados que os parceiros devem atender para se qualificarem para o benefício
- Esquema de solicitação de benefícios — As informações e a documentação específicas que os parceiros precisam fornecer ao se inscrever
- Detalhes do subsídio - O que os parceiros receberão após a aprovação
- Programas associados - Quais programas de AWS parceiros o benefício apoia

O esquema de solicitação de benefícios é particularmente importante, pois define a estrutura e os campos obrigatórios para os aplicativos de benefícios. Os parceiros devem analisar esse esquema cuidadosamente antes de criar um aplicativo de benefícios para garantir que coletem todas as informações necessárias com antecedência.

> **Important**
>
> A determinação da elegibilidade para cada benefício é feita pelo cliente ou pela camada de interface do usuário. Para benefícios relacionados ao financiamento, a elegibilidade geralmente se baseia no desempenho do scorecard do parceiro e na participação no programa.

Tópicos

- Trabalhando com aplicativos de benefícios
- Trabalhando com alocações de benefícios

Trabalhando com aplicativos de benefícios

Um aplicativo de benefícios modela a solicitação de um parceiro por um benefício específico. Ele captura todas as informações necessárias para avaliar e processar a solicitação de acordo com as condições específicas do benefício. O ciclo de vida do pedido de benefícios avança em vários estados, desde a criação do rascunho até a aprovação ou rejeição final.

Criação de aplicativos de benefícios

Os parceiros iniciam o processo de solicitação de benefícios criando um aplicativo de benefícios usando a ação da `CreateBenefitApplication` API. Na criação, o aplicativo entra em `PENDING_SUBMISSION` status, permitindo que os parceiros preparem as informações completas antes de enviá-las para análise.

Ao criar um pedido de benefícios, os parceiros devem fornecer:

- Identificador do benefício - O ID ou ARN do benefício que está sendo solicitado
- Tipos de atendimento — Como o parceiro espera receber o benefício (`CREDITS`,,`CASH`,`DISCOUNT`, `ACCESSRECOGNITION`, ou`RESOURCE`)
- Detalhes do aplicativo de benefícios - Um documento JSON contendo informações específicas do benefício, conforme definido pelo esquema de solicitação do benefício
- Token de cliente - Um token de idempotência exclusivo para evitar envios duplicados

Opcionalmente, os parceiros podem fornecer:

- Nome e descrição - Human-readable identificadores para o aplicativo
- Contatos do parceiro - Informações de contato da pessoa que gerencia essa solicitação de benefício (máximo de 1 contato)
- Anexos de arquivo - Documentação de apoio, como planos de projeto, SOWs, comprovante de custo ou pesquisas de satisfação do cliente (máximo de 10 arquivos)
- Recursos associados - Links para oportunidades relacionadas ou alocações de benefícios existentes (máximo de 10 recursos)
- Tags - Key-value pares para organização e rastreamento de recursos (máximo de 200 tags)

A `CreateBenefitApplication` API realiza uma validação flexível, verificando somente tipos e padrões de campo básicos. A validação completa da lógica de negócios ocorre durante o envio, permitindo que os parceiros salvem as inscrições incompletas e retornem posteriormente para concluí-las.

Prática recomendada: os parceiros devem usar o esquema de solicitação de benefícios `GetBenefit` para entender exatamente quais informações são necessárias antes de criar um aplicativo. Isso reduz a probabilidade de falhas no envio devido a dados ausentes ou inválidos.

Atualizando aplicativos de benefícios

Os parceiros podem modificar os rascunhos dos pedidos de benefícios usando a ação `UpdateBenefitApplication` da API. Isso permite que os parceiros refinem as informações, adicionem documentação ou corrijam erros antes do envio.

Ao atualizar um pedido de benefícios, os parceiros devem fornecer:

- Identificador - O ID ou ARN do pedido de benefício a ser atualizado
- Revisão - O número da revisão atual para um bloqueio otimista
- Detalhes do aplicativo de benefícios - O documento JSON completo e atualizado

Os parceiros devem enviar o objeto completo da solicitação de benefícios, mesmo que somente campos específicos estejam mudando. A melhor prática é primeiro recuperar os detalhes mais recentes do aplicativo usando`GetBenefitApplication`, modificar os campos necessários e, em seguida, enviar a carga útil atualizada completa para. `UpdateBenefitApplication`

Bloqueio otimista: o campo de revisão garante que as atualizações sejam aplicadas somente se o aplicativo não tiver sido alterado desde a última vez que foi recuperado. Se a revisão não corresponder ao valor atual do banco de dados, a atualização será rejeitada com um erro de conflito. Isso evita que os parceiros substituam acidentalmente as alterações feitas por outros processos ou sistemas.

As atualizações só podem ser realizadas enquanto o aplicativo estiver em `PENDING_SUBMISSION` status. Depois de enviadas, as inscrições entram no processo de análise e não podem mais ser atualizadas por meio dessa API.

Associando recursos a aplicativos de benefícios

Os parceiros podem vincular aplicativos de benefícios a recursos relacionados usando a ação `AssociateBenefitApplicationResource` da API. Isso cria conexões valiosas entre os benefícios e o contexto comercial no qual eles estão sendo usados.

Os tipos de recurso aceitos incluem:

- OPORTUNIDADE - Vincule o aplicativo de benefícios a uma oportunidade específica do cliente em. Isso é particularmente útil para benefícios específicos de oportunidades, como financiamento de MAP ou créditos de POC vinculados ao engajamento do cliente.

- BENEFIT_ALLOCATION - Vincule às alocações de benefícios existentes, permitindo que os parceiros cadeiem benefícios ou demonstrem como os benefícios anteriores estão sendo aproveitados

A associação de recursos só pode ocorrer antes do envio. Depois que uma solicitação de benefício é enviada (o status muda para`IN_REVIEW`), nenhuma outra associação é permitida. Os parceiros podem associar até 10 recursos a cada solicitação de benefícios.

Para desassociar um recurso, os parceiros usam a ação da `DisassociateBenefitApplicationResource` API. Assim como a associação, a dissociação só pode ocorrer no `PENDING_SUBMISSION` status.

> **Important**
>
> Os parceiros devem ter as permissões apropriadas para acessar o aplicativo de benefícios e ler o recurso específico que está sendo associado. A API valida essas permissões durante a operação de associação.

Envio de solicitações de benefícios

Quando uma solicitação de benefícios está completa e pronta para AWS análise, os parceiros a enviam usando a ação da `SubmitBenefitApplication` API. O envio aciona uma transição de estado de `PENDING_SUBMISSION` para `IN_REVIEW` e inicia o fluxo de trabalho de aprovação específico do benefício.

Após o envio, a API realiza uma validação abrangente, incluindo:

- Validação de campos obrigatórios - Garante que todos os campos obrigatórios nos detalhes do pedido de benefícios estejam presentes
- Validação do formato do campo - Verifica se os valores do campo correspondem aos padrões e tipos de dados esperados
- Validação de regras de negócios - aplica lógica e restrições de negócios específicas de benefícios
- Validação de recursos - confirma que todos os recursos associados são válidos e acessíveis
- Validação de arquivo - Verifica se o processamento de todos os anexos de arquivos foi concluído com êxito

Se a validação falhar, a API retornará um ValidationException com códigos de erro e mensagens detalhados indicando quais campos precisam ser corrigidos. Os parceiros devem resolver esses problemas usando `UpdateBenefitApplication` antes de tentar o envio novamente.

Requisito de processamento de arquivos: Os pedidos de benefícios não podem ser enviados se algum arquivo anexado ainda estiver em `PENDING` status. Os parceiros devem aguardar a conclusão do processamento do arquivo (o status muda para`SUCCEEDED`) antes de enviá-lo. Se os arquivos falharem no processamento (status`FAILED`), os parceiros devem fazer o upload das versões corrigidas.

Após o envio bem-sucedido:

- O status do aplicativo muda para `IN_REVIEW`
- A equipe do proprietário do benefício é notificada sobre o novo envio
- Os parceiros não podem mais modificar os detalhes do aplicativo por meio de APIs de atualização padrão
- O aplicativo entra em um fluxo de trabalho de aprovação definido que pode incluir estágios de aprovação comercial, aprovação técnica e aprovação financeira

Os parceiros podem acompanhar o progresso do envio recuperando a inscrição e monitorando o campo Estágio, que indica o estágio de aprovação atual (por exemplo, “Aprovação comercial”, “Aprovação técnica”, “Aprovação financeira”).

Gerenciando inscrições enviadas

Depois de enviados, os parceiros têm opções limitadas para gerenciar os pedidos de benefícios, mas a API fornece ações específicas para cenários comuns.

Relembrando aplicativos de benefícios

Se um parceiro descobrir um erro ou precisar fazer alterações após o envio, ele poderá cancelar o aplicativo usando a ação da `RecallBenefitApplication` API. O Recall faz a transição do aplicativo de volta ao `PENDING_SUBMISSION` status, permitindo atualizações e reenvio.

Ao cancelar uma inscrição, os parceiros devem fornecer:

- Identificador - O ID ou ARN do pedido de benefício a ser retirado
- Motivo - Uma explicação opcional para o recall (máximo de 1000 caracteres)

O campo do motivo permite uma melhor rastreabilidade e ajuda a AWS entender problemas comuns que levam a recalls, informando futuras melhorias no processo de solicitação de benefícios.

Após o recall, os parceiros podem usar `UpdateBenefitApplication` para fazer as alterações necessárias e reenviar usando `SubmitBenefitApplication` quando estiverem prontos.

> **Important**
>
> Normalmente, o recall só está disponível durante os estágios iniciais de revisão. As inscrições que progrediram para estágios posteriores de aprovação ou foram aprovadas podem não ser elegíveis para recall.

Alterando os pedidos de benefícios

Para pequenas correções após o envio, os parceiros podem usar a ação da `AmendBenefitApplication` API para atualizar campos específicos sem precisar recuperar todo o aplicativo. Isso é particularmente útil quando AWS os revisores solicitam esclarecimentos ou correções durante o processo de revisão.

As emendas usam uma Patch-style abordagem JSON em que os parceiros especificam:

- Path - Uma expressão JSONPath identificando o campo a ser atualizado (por exemplo,) `$.CreditDisbursementDetails.AwsAccountIdForCredits`
- Valor - O novo valor para o campo
- Operação - A operação a ser executada (atualmente somente `REPLACE` é suportada)

Os parceiros podem enviar até 10 emendas em uma única chamada de API. Cada emenda deve incluir um número de revisão atualizado para um bloqueio otimista.

As emendas devem ser usadas para pequenas correções. Para mudanças substanciais, os parceiros devem usar `RecallBenefitApplication` para devolver o aplicativo ao status de rascunho para obter atualizações abrangentes.

Cancelamento de solicitações de benefícios

Se um parceiro não precisar mais de um benefício ou quiser retirar sua inscrição, ele poderá cancelá-la usando a ação da `CancelBenefitApplication` API. O cancelamento faz a transição da solicitação para o `CANCELED` status, encerrando permanentemente o processo de inscrição.

Ao cancelar uma inscrição, os parceiros devem fornecer:

- Identificador - O ID ou ARN do pedido de benefício a ser cancelado
- Motivo - Uma explicação opcional para o cancelamento (máximo de 1000 caracteres)

Depois de cancelados, os aplicativos não podem ser reativados. Se o parceiro decidir posteriormente que precisa do benefício, ele deve criar um novo pedido de benefício.

Os motivos comuns para o cancelamento incluem:

- A oportunidade do cliente foi perdida ou atrasada
- As restrições de capacidade do parceiro foram alteradas
- As prioridades de negócios mudaram
- O benefício não é mais necessário para a finalidade pretendida

Exibindo detalhes da solicitação de benefícios

Os parceiros podem recuperar informações completas sobre um aplicativo de benefícios usando a ação da `GetBenefitApplication` API. Isso fornece uma visão abrangente do aplicativo, incluindo:

Metadados do aplicativo:

- Identificador exclusivo (ID) e nome de recurso da Amazon (ARN)
- Identificador de benefício associado
- Nome e descrição do aplicativo
- Status atual (`PENDING_SUBMISSIONIN_REVIEW`,`ACTION_REQUIRED`,`APPROVED`,`REJECTED`, ou`CANCELED`)
- Estágio de processamento atual

Informações de status:

- Motivo do status - Human-readable explicação do status atual
- Códigos de motivo de status - Códigos estruturados indicando problemas ou requisitos específicos (por exemplo, “Lista de verificação incompleta”, “Falta o plano do projeto”, “Falta o resultado do projeto”)

Conteúdo do aplicativo:

- Documento JSON completo com detalhes do pedido de benefícios
- Informações de contato do parceiro
- Anexos de arquivo com status de processamento
- Recursos associados (oportunidades ou alocações)
- Tags de recursos

Informações de auditoria:

- Carimbo de data e hora de criação
- Carimbo de data/hora da última modificação
- Número da revisão atual

Os códigos de motivo do status são particularmente valiosos quando um aplicativo está em `ACTION_REQUIRED` status. Esses códigos fornecem orientação específica e acionável sobre o que os parceiros precisam abordar para que a inscrição prossiga.

Listando aplicativos de benefícios

Os parceiros podem visualizar todos os seus aplicativos de benefícios usando a ação `ListBenefitApplications` da API. Isso retorna uma lista paginada de resumos de aplicativos com poderosos recursos de filtragem.

Os parceiros podem filtrar os aplicativos por:

- Programa - Visualize aplicativos para programas específicos (MAP, MDF, Sandbox, POC, etc.)
- Tipo de atendimento - Filtrar por método de entrega (`CREDITS`,`CASH`,`ACCESS`, etc.)
- Identificador de benefícios - Visualize os aplicativos para um benefício específico
- Status - Filtrar por status do aplicativo (`PENDING_SUBMISSIONIN_REVIEW`,`APPROVED`,,`REJECTED`,`CANCELED`)
- Estágio - Filtrar por estágio de aprovação (aprovação comercial, aprovação técnica, aprovação financeira)
- ARNs de recursos associados - Encontre aplicativos vinculados a oportunidades ou alocações específicas

A resposta da lista inclui informações resumidas essenciais, como:

- ID, ARN e nome do aplicativo
- ID do benefício associado
- Programas e tipos de atendimento
- Status e estágio atuais
- Carimbos de data e hora de criação e modificação
- Recursos associados
- Número da revisão atual
- Campos selecionados dos detalhes do pedido de benefícios (conforme definido pelo proprietário do benefício)

Os parceiros podem configurar a classificação dos resultados usando o parâmetro Classificar, com opções para classificar por data de criação, status, programa ou identificador de benefícios em ordem crescente ou decrescente.

Criação de painéis: a `ListBenefitApplications` API foi projetada para oferecer suporte à criação de painéis de parceiros. Ao filtrar e classificar os aplicativos, os parceiros podem criar visualizações como:

- Aplicativos que exigem ação (`ACTION_REQUIRED`status)
- Inscrições enviadas recentemente (classificadas por data de criação, `IN_REVIEW` status)
- Candidaturas aprovadas com alocação pendente (`APPROVED`status)
- Aplicativos por programa (filtrados por programas específicos)

Trabalhando com alocações de benefícios

Uma alocação de benefícios representa um benefício que foi concedido a um parceiro. Quando uma solicitação de benefício é aprovada,AWS cria uma ou mais alocações de benefícios que fornecem aos parceiros os créditos reais, o financiamento, o acesso ou o cumprimento de outros benefícios.

Entendendo as alocações de benefícios

As alocações de benefícios podem representar vários tipos de atendimento:

- Desembolsos financeiros (CASH) - Pagamentos financeiros diretos a parceiros com rastreamento de desembolsos
- AWS Créditos (CRÉDITOS) - Códigos de crédito aplicáveis a AWS contas com rastreamento de uso
- Alocações de consumíveis - valores de Pre-approved financiamento ou carteiras com rastreamento de utilização
- Concessões de acesso (ACCESS) - Acesso a sistemas, ferramentas ou recursos
- Reconhecimento (RECONHECIMENTO) - Emblemas digitais, certificações ou designações de status
- Recursos (RESOURCE) - Suporte técnico, serviços de consultoria ou ProServe contratos

Cada alocação de benefícios tem um ciclo de vida rastreado por meio de seu campo Status:

- ATIVO - A alocação está disponível para uso
- INATIVO - A alocação está temporariamente indisponível
- CUMPRIDA - A alocação foi totalmente usada ou entregue

As alocações de benefícios também incluem informações temporais que definem quando elas se tornam efetivas (StartsAt) e quando expiram (ExpiresAt), permitindo que os parceiros entendam os prazos de utilização dos benefícios.

Listando alocações de benefícios

Os parceiros podem ver todas as alocações de benefícios usando a ação da `ListBenefitAllocations` API. Isso retorna uma lista paginada de resumos de alocação com recursos abrangentes de filtragem.

Os parceiros podem filtrar as alocações por:

- Identificador de benefícios - Visualize as alocações para um benefício específico
- Identificador do aplicativo de benefícios - Visualize as alocações resultantes de um aplicativo específico
- Tipo de atendimento - Filtrar por tipo de alocação (`CREDITS`,, `CASHACCESS`, etc.)
- Status - Filtrar por status de alocação (`ACTIVE`,`INACTIVE`,`FULFILLED`)

A resposta da lista inclui resumos de alocação contendo:

- ID de alocação, ARN e nome
- ID do benefício associado e ID do pedido de benefício (se aplicável)
- Tipo de atendimento
- Status atual
- Carimbo de data e hora de criação
- Data de início

Essa exibição de lista permite que os parceiros criem painéis de rastreamento de alocação e identifiquem:

- Alocações ativas disponíveis para uso imediato
- Alocações que se aproximam do vencimento
- Alocações preenchidas para rastreamento histórico
- Valor total da alocação por programa ou tipo de atendimento

Exibindo informações detalhadas de alocação

Os parceiros podem recuperar detalhes completos da alocação usando a ação da `GetBenefitAllocation` API. Isso fornece informações abrangentes sobre a alocação, incluindo detalhes específicos do atendimento que variam de acordo com o tipo de alocação.

Informações básicas de alocação:

- Identificador exclusivo (ID) e nome de recurso da Amazon (ARN)
- Nome e status da alocação
- Motivo do status (explicação do status atual)
- ID do benefício associado e ID do pedido de benefício
- Tipo de atendimento
- Carimbos de data e hora de criação, atualização, início e expiração

Detalhes de atendimento (Type-Specific):

O FulfillmentDetail campo contém diferentes estruturas de informações com base no tipo de atendimento:

Detalhes do desembolso de dinheiro (tipo CASH)

Para pagamentos diretos em dinheiro, os detalhes de atendimento incluem:

- Valor desembolsado - O valor total do financiamento desembolsado, incluindo valor do valor e código da moeda
- Detalhes da emissão (se aplicável) - Informações sobre pedidos de compra ou eventos de emissão:
  - ID de emissão - Identificador exclusivo para o desembolso
  - Valor da emissão - Valor em moeda local
  - Emitido em - Data e hora do desembolso

Essa estrutura suporta desembolsos únicos e cenários de financiamento pré-aprovados, nos quais várias emissões podem ocorrer em relação a uma única alocação.

Detalhes do crédito (tipo de CRÉDITOS)

Para AWS créditos, os detalhes do processamento incluem:

- Valor alocado - Valor total de crédito alocado
- Valor emitido - Valor total do crédito emitido (pode ser menor do que o alocado para emissão em fases)
- Códigos de crédito - Conjunto de detalhes de códigos de crédito individuais:
  - AWS ID da conta - Conta em que o crédito é aplicado
  - Valor - Valor monetário do código de crédito
  - AWS Código de crédito - A sequência real do código de crédito
  - Status - Status do código de crédito (`ACTIVE`,`INACTIVE`,`FULFILLED`)
  - Emitido em - Quando o código de crédito foi emitido
  - Expira em - Quando o código de crédito expira

Esse rastreamento de crédito detalhado permite que os parceiros monitorem a utilização do crédito em várias contas e entendam quais créditos estão disponíveis, parcialmente usados ou totalmente consumidos.

Detalhes da alocação de consumíveis () Pre-Approved Amounts/Wallets

Para grupos de financiamento pré-aprovados, os detalhes do atendimento incluem:

- Quantia alocada - Valor total na alocação
- Quantidade restante - Quantidade não utilizada ainda disponível
- Quantidade utilizada - Quantidade já consumida
- Detalhes da emissão (se aplicável) - Informações da ordem de compra para a alocação

As alocações de consumíveis permitem que os parceiros retirem fundos conforme necessário para atividades elegíveis, com rastreamento em tempo real do saldo restante.

Detalhes de acesso (tipo de acesso)

Para concessões de acesso, os detalhes de atendimento incluem:

- Descrição - Explicação detalhada do acesso que está sendo concedido e como utilizá-lo

As alocações de acesso podem fornecer acesso a serviços de pré-visualização, ferramentas especializadas ou recursos restritos. O campo de descrição fornece instruções sobre como os parceiros podem ativar e usar o acesso concedido.

Aplicando alocações a novos pedidos de benefícios

O ApplicableBenefitIds campo em alocações de benefícios identifica outros benefícios que podem alavancar essa alocação. Isso permite a cadeia de benefícios, na qual os parceiros podem usar as alocações existentes para solicitar benefícios adicionais.

Por exemplo, um parceiro pode:

1. Receba uma alocação de financiamento pré-aprovada do MAP
2. Crie aplicativos de benefícios para projetos individuais de migração de clientes
3. Associe esses aplicativos à alocação pré-aprovada
4. Retire fundos da alocação à medida que os projetos são aprovados

Essa abordagem simplifica o processo de solicitação de benefícios para parceiros com financiamento pré-aprovado, reduzindo o ciclo de revisão para solicitações de projetos individuais.

Usar o AWS API do Partner Central Channel

AWS As APIs do Partner Central Channel permitem que você gerencie programaticamente seu negócio de revenda autorizado AWS. Essas APIs permitem que provedores de AWS soluções,AWS distribuidores e vendedores AWS de distribuição:

- Contas AWS de gerenciamento de relatórios usadas para programas de revenda para o Partner Central
- Envie convites para AWS contas para aceitar associações parceiras
- Crie relacionamentos necessários para integrar AWS contas de clientes finais revendidas e aplicar descontos de parceiros

Com as APIs do AWS Partner Central Channel, você pode criar automação personalizada para:

- Integre novas AWS contas de propriedade de parceiros aos programas de revenda da APN
- Integre novas contas de clientes finais aos programas de revenda
- Acelere a integração de clientes e a qualificação de descontos

Trabalhando com contas de gerenciamento de programas (PMAs)

Uma Conta de Gerenciamento de Programa (PMA) associa sua conta AWS de gerenciamento ao registro do Programa de Canais. As contas de gerenciamento do programa são ativadas pelo autoatendimento usando os apertos de mão do canal. Quando você cria e ativa um PMA, os benefícios e descontos do programa de canais são aplicados a todas as faturas entregues à conta de AWS gerenciamento associada. Use essas APIs para relatar e gerenciar AWS contas que lidam com revendas e descontos de canais:

- `CreateProgramManagementAccount`: Adicione novas contas para gerenciamento de faturamento de clientes
- `UpdateProgramManagementAccount`: Modificar contas existentes
- `DeleteProgramManagementAccount`: Remover contas dos programas de revenda
- `ListProgramManagementAccounts`: Visualize contas existentes e seu status

Trabalhando com apertos de mão de canais

Os apertos de mão do canal permitem o consentimento mútuo e seguro para operações críticas de gerenciamento de canais.AWS O Partner Central usa controles de canal para verificar o consentimento das contas de AWS gerenciamento para três propósitos distintos: ativar contas de gerenciamento de programas (PMAs), estabelecer períodos de serviço e encerrar períodos de serviço antecipadamente. Existem três tipos:

- `PROGRAM_MANAGEMENT_ACCOUNT`: Os apertos de mão da conta de gerenciamento do programa verificam o consentimento ao ativar os PMAs para aplicar os benefícios do programa de canais
- `START_SERVICE_PERIOD`: Os apertos de mão de criação do período de serviço estabelecem acordos mútuos para compromissos de transferência de faturamento com períodos mínimos de aviso prévio ou prazos fixos
- `REVOKE_SERVICE_PERIOD`: Os apertos de mão de rescisão do período de serviço permitem a rescisão antecipada dos períodos de serviço ativos quando ambas as partes consentem. Todos os tipos de handshake exigem a aceitação da conta de AWS gerenciamento de destino para entrarem em vigor.

Use essas APIs para gerenciar os handshakes do canal para ativação do PMA e gerenciamento do período de serviço:

- `CreateChannelHandshake`: Crie apertos de mão para convites do PMA, estabelecimento do período de serviço ou rescisão do período de serviço
- `AcceptChannelHandshake`: Aceite apertos de mão de AWS contas alvo (pode ser usado por parceiros ou clientes, dependendo do tipo de aperto de mão)
- `RejectChannelHandshake`: Rejeite apertos de mão das contas alvo AWS
- `CancelChannelHandshake`: cancele os apertos de mão pendentes antes que eles sejam aceitos, rejeitados ou expirem
- `ListChannelHandshakes`: monitore e visualize apertos de mão com filtragem por tipo e status

Trabalhando com relacionamentos de canais

Os relacionamentos permitem que você gerencie clientes finais, organizações internas ou vendedores posteriores. Cada relacionamento inclui:

- A conta de AWS gerenciamento da AWS organização associada

- Metadados necessários da AWS Partner Network para qualificação de descontos em canais

Quando você cria um relacionamento, todo o uso da AWS organização associada recebe os benefícios relacionados ao Programa de Canais. Use essas APIs para denunciar vendedores e clientes finais:

- `CreateRelationship`: integre contas internas de novos clientes finais, vendedores de distribuição ou contas internas AWS
- `GetRelationship`: Exibir detalhes específicos do relacionamento
- `ListRelationships`: Exibir todos os relacionamentos
- `UpdateRelationship`: Modificar relacionamentos existentes
- `DeleteRelationship`: Remover relacionamentos

Usando a API de medição de receita

Tópicos

- Trabalhando com seus IDs de atribuição de receita
- Trabalhando com suas ações de receita do Marketplace

Trabalhando com seus IDs de atribuição de receita

Um ID de atribuição de receita é um identificador em nível de negócio que mapeia sua receita de produtos do AWS Marketplace para oportunidades específicas do AWS Marketplace Offers Partner Central. and/or AWS Ele se baseia na implementação do Partner Revenue Measurement (PRM) em nível de produto: o PRM captura a receita total atribuída a um produto do Marketplace, e um ID de atribuição de receita mapeia essa receita para negócios específicos de acordo com as porcentagens mensais de alocação de custos que você especificar.

O Revenue Attribution Service expõe APIs para criar, recuperar, listar e atualizar IDs de atribuição de receita e gerenciar suas entradas mensais de alocação de custos de forma assíncrona.

O que é um ID de atribuição de receita?

Uma ID de atribuição de receita tem duas camadas:

- A atribuição é um recurso nomeado pelo parceiro e descrito pelo parceiro, identificado por um ID de `ra-` prefixo (por exemplo,) e um ARN. `ra-aabbccddee001` Cada atribuição tem como escopo uma `Catalog` (para produção, `AWS Sandbox` para teste) e a conta do parceiro.
- Uma ou mais entradas mensais de alocação de custos. Cada entrada mapeia uma única combinação (ID da oferta ou ID da oportunidade, mês de cobrança) para uma porcentagem de alocação de custos. As entradas são gerenciadas em lotes de forma assíncrona por meio de um padrão de tarefa de alocação.

Um ID de atribuição de receita pode ser usado de duas maneiras:

- Como uma sobreposição em nível de negócio em sua implementação existente de PRM em nível de produto. Crie um ID de atribuição de receita, associe suas ofertas and/or ACE de ofertas do Marketplace aplicáveis e AWS mapeie sua receita de produtos já medida a essas ofertas específicas. Nenhuma alteração nas tags de recursos ou nas cadeias de caracteres do agente do usuário existentes é necessária.
- Como um identificador independente usado diretamente como um valor de tag de recurso (`aws-apn-id=<RA ID>`) ou em uma string de agente de usuário (`APN_1.1/pc_<RA ID>$`) para atribuir o AWS consumo a um negócio específico desde o início.

Trabalhando com seus IDs de atribuição de receita

Os parceiros podem gerenciar IDs de atribuição de receita e suas entradas mensais de alocação de custos por meio do Revenue Attribution Service. O ciclo de vida progride por meio de dois fluxos independentes: o fluxo de registro de atribuição (criar, atualizar, recuperar, listar) e o fluxo de alocações (iniciar uma tarefa em lote, pesquisar resultados, recuperar entradas individuais, listar entradas para uma atribuição).

Criação de um ID de atribuição de receita

A primeira etapa é criar uma ID de atribuição de receita usando a ação da `CreateRevenueAttribution` API. O retornado `Id` e `Arn` pode ser usado imediatamente como um valor de tag de recurso ou em uma string de agente de usuário para atribuir o consumo.AWS

Ao criar um ID de atribuição de receita, os parceiros devem fornecer:

- **`Catalog`**— `AWS` para produção ou `Sandbox` teste.
- **`Name`**— um nome legível por humanos para a atribuição. Deve ser exclusivo no Catálogo e na conta do parceiro. Máximo de 128 caracteres.

Opcionalmente, os parceiros podem fornecer:

- **`Description`**— descrição em texto livre da atribuição. Máximo de 1024 caracteres.
- **`MarketplaceProduct`**— o produto do AWS Marketplace a ser associado a essa atribuição. Forneça `ProductIdentifier` (o ID do produto do Marketplace de 25 caracteres) e `TenancyModel` (`MULTI_TENANTouSINGLE_TENANT`). Se omitida na criação, a atribuição se aplica a todo o consumo medido pelo ID de atribuição de receita, independentemente do produto do Marketplace comprado pelo cliente. Isso é útil quando uma única implantação abrange mais de uma listagem do Marketplace.
- **`Tags`**— até 200 pares de valores-chave para organização de recursos. As chaves de tag devem ser exclusivas na solicitação.

Melhor prática: forneça `MarketplaceProduct.ProductIdentifier` sempre que seu negócio estiver ancorado em uma listagem específica do Marketplace. Isso permite AWS validar se o produto é de propriedade da conta do parceiro chamador ou de uma conta subsidiária conectada por meio da Conexão de Conta de Parceiro (PAC) e mostra o resolvido `ProductCode` e `ProductType` (por exemplo,, `SaaSAMI`,`ML`) na resposta. Se o produto não for de propriedade de uma conta autorizada, a API retornará `ValidationException` com o motivo`PRODUCT_NOT_FOUND_OR_NOT_OWNED`.

A resposta retorna o novo `Id` (formato`ra-[a-z0-9]{13}`)`Arn`, `MarketplaceProduct` os atributos resolvidos e o `Version` número inicial (que começa em 1 e aumenta em cada atualização subsequente).

Adicionar entradas mensais de alocação de custos

Depois que um ID de atribuição de receita é criado, os parceiros associam o AWS Marketplace Offers and/or ACE Opportunities a ele enviando um lote de entradas mensais de alocação de custos por meio da ação da API. `StartRevenueAttributionAllocationsTask` Essa é uma operação assíncrona que aceita até 250 alterações de alocação () `CREATE` and/or `UPDATE` por tarefa.

Cada entrada de alocação especifica:

- ID da oferta ou ID da oportunidade — o identificador exclusivo do negócio que está sendo associado. Se uma oferta do Marketplace já estiver vinculada a uma oportunidade ACE, os parceiros só precisarão fornecer o ID da oferta; atribuem AWS automaticamente a receita à oportunidade vinculada.
- Mês de cobrança — o mês civil ao qual a alocação de custos se aplica (por exemplo,`2026-04`).

- % de alocação de custos — a parte do AWS consumo total do produto atribuível a essa oferta neste mês de cobrança. Necessário para produtos SaaS multilocatários, incluindo componentes hospedados por parceiros para implantações híbridas, nas quais os clientes compartilham a infraestrutura.
- ID AWS da conta do cliente — a AWS conta do cliente que consome o produto. Necessário para produtos SaaS multilocatários.

A porcentagem total de alocação de custos em todas as entradas de alocação de uma determinada ID de atribuição de receita e do mesmo mês de cobrança não deve exceder 100%. Se o total exceder 100%, a entrada ofensiva será rejeitada durante a validação comercial assíncrona com um código de erro por registro.

Validação síncrona: `StartRevenueAttributionAllocationsTask` executa a validação básica da forma de forma síncrona (tipos de campo, padrões, tamanho do lote). Se a validação básica for aprovada, a API retornará um `TaskId` com status`IN_PROGRESS`. A validação comercial (verificações de limite, regras de imutabilidade, pesquisas de dependências em relação ao Marketplace e ao ACE) é executada de forma assíncrona e os resultados por registro são descobertos por meio de. `GetRevenueAttributionAllocationsTask`

Para monitorar uma tarefa enviada, os parceiros fazem uma pesquisa `GetRevenueAttributionAllocationsTask` com o ID de recurso de atribuição de receita ou ARN. A resposta avança de `IN_PROGRESS` para `COMPLETED` (independentemente de os registros individuais terem sido bem-sucedidos ou falhados) e retorna:

- **`TaskStatus`**—`IN_PROGRESS`,`COMPLETED`, ou `FAILED` (falha na tarefa de nível superior).
- **`RecordResults`**— para cada registro de entrada: o atribuído `AllocationId` (se bem-sucedido), o status por registro (`SUCCEEDED`ou`FAILED`) e um código de erro estruturado e uma mensagem se o registro falhar.

Os parceiros devem recuperar os resultados das tarefas imediatamente. Após a conclusão da tarefa, os resultados por registro podem ser reconciliados e qualquer entrada com falha pode ser corrigida e reenviada em uma nova tarefa.

Editando entradas mensais de alocação de custos

As entradas de alocação de custos são gerenciadas por mês de cobrança com estas regras:

- Os parceiros podem adicionar uma nova entrada mensal para o mês de cobrança atual ou futuro a qualquer momento.
- Os parceiros podem atualizar uma entrada mensal para o mês de cobrança atual ou futuro a qualquer momento.
- Os parceiros podem atualizar a entrada do mês passado até o dia 7 do mês atual. Essa janela permite que os parceiros analisem o uso real no AWS Cost Explorer antes de finalizar a alocação do mês anterior.

Após o 7º dia do mês atual, a atribuição do mês de cobrança anterior não poderá mais ser modificada. As atualizações em uma entrada do mês atual ou do futuro se aplicam a partir do próximo ciclo de cobrança mensal. A atribuição histórica dos meses anteriores não é recalculada retroativamente.

As atualizações nas entradas de alocação são enviadas por meio da mesma ação de `StartRevenueAttributionAllocationsTask` API `Operation` definida `UPDATE` para cada entrada afetada. O padrão de tarefa assíncrona lida com atualizações e criações de maneira uniforme.

Atualização de um ID de atribuição de receita

Os parceiros podem atualizar o ID `Description` de atribuição de receita existente usando a ação da `UpdateRevenueAttribution` API. Caso contrário, o registro de atribuição em si é imutável —`Name`,`MarketplaceProduct`, e os valores de tag-on-create não podem ser alterados após a criação. Para remarcar o recurso, use as operações de AWS marcação padrão no ARN da atribuição.

Ao atualizar um ID de atribuição de receita, os parceiros devem fornecer:

- **`Catalog`**— `AWS` ou`Sandbox`.
- **`Identifier`**— o `ra-` ID da atribuição a ser atualizada.
- **`Version`**— a versão atual da atribuição para bloqueio otimista.

Opcionalmente, os parceiros podem fornecer:

- **`Description`**— a descrição atualizada em texto livre. Máximo de 1024 caracteres.
- **`ClientToken`**— um token de idempotência para a atualização.

Bloqueio otimista: o `Version` campo garante que as atualizações sejam aplicadas somente se a atribuição não tiver sido alterada desde a última vez que foi recuperada. Se o envio `Version` não corresponder à versão atual do recurso, a atualização será rejeitada com `ConflictException` uma mensagem incluindo os números da versão enviada e atual. A melhor prática é recuperar a versão mais recente `GetRevenueAttribution` antes de cada atualização.

Visualizando detalhes do ID de atribuição de receita

Os parceiros podem recuperar informações completas para um único ID de atribuição de receita usando a ação da `GetRevenueAttribution` API. Isso retorna:

Metadados do recurso:

- Identificador exclusivo (`Id`) e nome de recurso da Amazon (`Arn`).
- O `Catalog` em que a atribuição existe.
- `Name` e `Description`.
- O recuperado `Version` e o `LatestVersion` (use o mais recente para atualizações subsequentes).

Atributos do produto do Marketplace (se `MarketplaceProduct` foram definidos na criação):

- `ProductId`— o ID do produto do Marketplace fornecido pelo parceiro.
- `ProductCode`— o código do produto AWS Marketplace foi resolvido a partir de`ProductId`.
- `ProductType`—`SaaS`,`AMI`, `ContainerML`,`Data`, ou`Professional Services`.
- `TenancyModel`— `MULTI_TENANT` ou`SINGLE_TENANT`.

Informações de auditoria:

- `CreatedDate` e `LastModifiedDate`.

Opcionalmente, os parceiros podem fornecer uma `Version` informação específica na solicitação para recuperar uma versão histórica da atribuição. Omita `Version` a devolução do mais recente.

Para recuperar uma entrada específica de alocação de custos mensal, os parceiros usam a ação de `GetRevenueAttributionAllocation` API com a entrada. `AllocationId` Isso retorna a ID da oferta ou a ID da oportunidade, o mês de cobrança, a porcentagem de alocação de custos, a ID da AWS conta do cliente e o status e as informações de auditoria da entrada.

Listando IDs de atribuição de receita

Os parceiros podem ver todas as IDs de atribuição de receita em suas contas usando a ação de `ListRevenueAttributions` API. Isso retorna uma lista paginada de resumos de atribuição com recursos de filtragem e classificação.

Os parceiros podem filtrar os resultados por:

- **`Catalog`**— `AWS` ou `Sandbox` (obrigatório).
- **`Identifiers`**— uma lista de até 100 `ra-` IDs específicos para recuperar.
- **`CreatedAfter`**/**`CreatedBefore`**— filtrar por intervalo de data e hora de criação (inclusive).

Os parceiros podem configurar a classificação dos resultados usando o `Sort` parâmetro:

- `SortBy`— `CreatedDate` ou`LastModifiedDate`.
- `SortOrder`— `ASCENDING` ou`DESCENDING`.

A resposta inclui um resumo para cada atribuição:`Arn`,,,`Id`, `MarketplaceProduct` atributos resolvidos `CatalogName`,, `CreatedDate``LastModifiedDate`, os mais recentes `Version` e `TotalRevenueAttributionAssociationCount` (o número de entradas ativas de alocação de custos mensais vinculadas à atribuição).

Use `MaxResults` (padrão 25, máximo 100) e `NextToken` para paginar por meio de grandes conjuntos de resultados. A resposta inclui `NextToken` se páginas adicionais estiverem disponíveis.

Listando entradas mensais de alocação de custos

Os parceiros podem listar as entradas mensais de alocação de custos para uma ID de atribuição de receita específica usando a ação da `ListRevenueAttributionAllocations` API. Isso retorna uma lista paginada de resumos de alocação com recursos de filtragem.

Os parceiros podem filtrar os resultados por:

- **`MarketplaceOfferId`**— listar somente as entradas associadas a uma oferta específica do Marketplace.
- **`AwsPartnerCentralOpportunityId`**— liste somente as entradas associadas a uma oportunidade específica do Partner Central.
- **`BillingMonth`**— listar somente as entradas de um mês civil específico.

A resposta inclui informações resumidas para cada entrada: a`AllocationId`, a ID da oferta ou ID da oportunidade associada, a ID da AWS conta do cliente, o mês de cobrança, a porcentagem de alocação de custos, o nome e o status da entrada e os registros de data e hora da auditoria.

Criação de painéis: a combinação de `ListRevenueAttributions` e foi `ListRevenueAttributionAllocations` projetada para apoiar a criação de painéis de parceiros. Os parceiros podem criar visualizações como:

- Todas as IDs de atribuição de receita criadas nos últimos 30 dias, classificadas por data de criação.
- Todas as entradas mensais de alocação de custos para uma oferta específica do Marketplace em vários IDs de atribuição de receita.
- Todas as entradas de alocação de custos para o mês de cobrança atual, para validar que os totais não excedam 100% por atribuição.
- Todos os IDs de atribuição de receita que tenham pelo menos uma entrada de alocação ativa (). `TotalRevenueAttributionAssociationCount > 0`

Trabalhando com suas ações de receita do Marketplace

Uma participação na receita do Marketplace é uma entrada para um produto do AWS Marketplace que declara a parte da receita total do produto coletada por meio do AWS Marketplace.AWS usa essa entrada para correlacionar a Marketplace-collected receita de um parceiro com outros sinais de receita.

O Marketplace Revenue Share Service expõe APIs para criar, recuperar e listar recursos do Marketplace Revenue Share e para gerenciar suas alocações (as porcentagens de participação na receita declaradas pelo parceiro e as janelas de data efetiva). As operações de AWS marcação padrão são suportadas no recurso Marketplace Revenue Share.

O que é uma participação na receita do Marketplace?

Uma participação na receita do Marketplace tem duas camadas:

- O compartilhamento — uma entrada para um único produto do AWS Marketplace, identificado pelo Marketplace`ProductId`. Há exatamente uma participação na receita do Marketplace por produto.
- Uma ou mais alocações — cada alocação declara uma `RevenueSharePercent` janela de data efetiva (`EffectiveDate`, opcionalmente). `ExpirationDate` Um compartilhamento pode ter

várias alocações ao longo do tempo, mas, a qualquer momento, somente uma `ACTIVE` alocação pode ser aplicada.

As alocações têm as seguintes propriedades principais:

- Status — `ACTIVE` ou`INACTIVE`. Na criação, o status padrão é. `ACTIVE` Somente as `ACTIVE` alocações participam dos cálculos de receita e das verificações de sobreposição.
- Invariante de sobreposição — dentro de um único compartilhamento, duas `ACTIVE` alocações não podem ter intervalos sobrepostos. `[EffectiveDate, ExpirationDate]` No máximo, uma `ACTIVE` alocação aberta (não`ExpirationDate`) é permitida por ação a qualquer momento.

Trabalhando com suas ações de receita do Marketplace

Os parceiros gerenciam o Marketplace Revenue Share e suas alocações por meio do Marketplace Revenue Share Service. O ciclo de vida progride por meio de dois fluxos: o fluxo de compartilhamento (criar, recuperar, listar, marcar) e o fluxo de alocações (criar, atualizar, excluir automaticamente, recuperar, listar).

Criando uma participação na receita do Marketplace

A primeira etapa é criar uma participação na receita do Marketplace para um produto do AWS Marketplace usando a ação da `CreateMarketplaceRevenueShare` API. O compartilhamento é identificado pelo Marketplace `ProductId` e serve como contêiner para todas as alocações subsequentes.

Ao criar uma participação na receita do Marketplace, os parceiros devem fornecer:

- **`Catalog`**— `AWS` para produção ou `Sandbox` teste.
- **`ProductId`**— o ID do produto Marketplace de 25 caracteres. Padrão:`[a-z0-9]{25}`.

Opcionalmente, os parceiros podem fornecer:

- **`Tags`**— até 50 pares de valores-chave para organização de recursos no momento da criação. Use `TagResource` para adicionar tags após a criação.
- **`ClientToken`**— um símbolo de idempotência para evitar criações duplicadas ao tentar novamente. Máximo de 64 caracteres. Os SDKs geram automaticamente esse token.

Validação do produto: na criação, o serviço verifica com o AWS Marketplace se o produto é de propriedade da conta do chamador ou de uma conta subsidiária conectada ao chamador via Partner Account Connection (PAC). Se o produto não existir ou a conta proprietária não estiver autorizada para o chamador, a API retornará `ValidationException` com o motivo. `PRODUCT_NOT_FOUND_OR_NOT_OWNED`

Uma ação por produto: uma segunda `CreateMarketplaceRevenueShare` chamada para o mesmo `ProductId` nas mesmas `Catalog` devoluções`ConflictException`.

A resposta retorna o compartilhamento `Arn` (formato`arn:{partition}:partnercentral:{region}:{account-id}:catalog/{catalog}/marketplace-revenue-share/{productId}`), o `CatalogProductId`, o `Version` número inicial (que começa em 1 e aumenta em cada mutação de alocação) e quaisquer tags aplicadas na criação.

Adicionando alocações

Depois que uma participação na receita do Marketplace é criada, os parceiros declaram uma porcentagem de participação na receita e uma janela de data efetiva criando uma alocação por meio da ação da API. `CreateMarketplaceRevenueShareAllocation`

Ao criar uma alocação, os parceiros devem fornecer:

- **`Catalog`**— `AWS` ou`Sandbox`.
- **`MarketplaceRevenueShareIdentifier`**— o identificador da ação principal.
- **`RevenueSharePercent`**— a porcentagem da receita do produto coletada por meio do Marketplace fornecida como decimal (por exemplo, `0.45` para 45%).
- **`EffectiveDate`**— a data do calendário em que essa alocação se torna efetiva. Formato: `YYYY-MM-DD`.

A resposta retorna a alocação `Id` (formato`mrsa-[A-Za-z0-9]{13}`), o status da alocação (`ACTIVE`na criação) e o compartilhamento principal aumentado `Version` (um único número inteiro de versão por compartilhamento é adicionado em cada mutação de alocação, usado para bloqueio otimista em atualizações de alocação subsequentes).

Atualizando alocações

Os parceiros podem atualizar uma alocação usando a ação da `UpdateMarketplaceRevenueShareAllocation` API. Os campos atualizáveis dependem se a alocação está datada do futuro, está em vigor no momento ou está datada do passado.

Ao atualizar uma alocação, os parceiros devem fornecer:

- **`Catalog`**— `AWS` ou`Sandbox`.
- **`MarketplaceRevenueShareIdentifier`**— o identificador da ação principal.
- **`AllocationId`**— o `mrsa-` ID da alocação a ser atualizada.
- **`MarketplaceRevenueShareVersion`**— a versão atual da ação principal para um bloqueio otimista na alocação. Os conflitos de colisão retornam`ConflictException`.

Opcionalmente, os parceiros podem fornecer:

- **`RevenueSharePercent`**— a porcentagem atualizada de participação na receita. Editável somente em alocações com data futura.
- **`EffectiveDate`**— a data efetiva atualizada. Editável somente em alocações com data futura.
- **`ExpirationDate`**— a data de expiração atualizada. Editável em alocações com data futura (qualquer valor> `EffectiveDate`) e em alocações atualmente efetivas ou abertas (qualquer valor). ≥ `today`
- **`Status`**— `ACTIVE` ou`INACTIVE`. A configuração para `INACTIVE` excluir temporariamente a alocação. Editável somente em alocações com data futura.
- **`ClientToken`**— um símbolo de idempotência.

Visualizando detalhes do Marketplace Revenue Share

Os parceiros podem recuperar informações completas de uma única participação na receita do Marketplace usando a ação de `GetMarketplaceRevenueShare` API. Isso retorna o compartilhamento`Arn`,`Catalog`,`ProductId`, `Version` (a versão atual do compartilhamento principal), `CreatedDate``LastModifiedDate`, e todas as tags aplicadas ao compartilhamento.

Os parceiros podem recuperar uma única alocação usando a ação da `GetMarketplaceRevenueShareAllocation` API com a alocação. `Id` Isso retorna:

- A alocação `Id` e a participação da controladora. `MarketplaceRevenueShareArn`
- `RevenueSharePercent`, `EffectiveDate`, `ExpirationDate`.
- `Status`(`ACTIVE`ou`INACTIVE`).
- O compartilhamento principal é `MarketplaceRevenueShareVersion` para um bloqueio otimista nas atualizações subsequentes.

- `CreatedDate` e `LastModifiedDate`.

Listando ações de receita do Marketplace

Os parceiros podem ver todas as ações de receita do Marketplace de propriedade de suas contas usando a ação de `ListMarketplaceRevenueShares` API. Isso retorna uma lista paginada de resumos de compartilhamentos.

Os parceiros podem filtrar os resultados por:

- **`Catalog`**— `AWS` ou `Sandbox` (obrigatório).
- **`ProductIds`**— uma lista de IDs de produtos específicos do Marketplace a serem recuperados.

A resposta inclui um resumo de cada compartilhamento: `ArnCatalog`,`ProductId`,, o mais recente `VersionCreatedDate`, `LastModifiedDate` e.

Use `MaxResults` e `NextToken` para paginar por meio de grandes conjuntos de resultados.

Listando alocações

Os parceiros podem listar as alocações para uma participação na receita específica do Marketplace usando a ação de `ListMarketplaceRevenueShareAllocations` API. Isso retorna uma lista paginada de resumos de alocação com recursos de filtragem.

Os parceiros podem filtrar os resultados por:

- **`Status`**— `ACTIVE` ou`INACTIVE`.
- Intervalo de datas efetivas —`EffectiveDateOnOrAfter`/`EffectiveDateOnOrBefore`para recuperar alocações cuja data efetiva esteja dentro de uma janela específica.

A resposta inclui informações resumidas para cada alocação:`Id`,,,`RevenueSharePercent`, `EffectiveDate` `ExpirationDate` `Status`MarketplaceRevenueShareVersion, e registros de data e hora de auditoria.

Marcação de compartilhamentos de receita do Marketplace

As operações de AWS marcação padrão são suportadas no recurso Marketplace Revenue Share:

- **`TagResource`**— adicione ou sobrescreva tags em um Marketplace Revenue Share. Máximo de 50 tags por compartilhamento.

- **`UntagResource`**— remova tags específicas por chave.
- **`ListTagsForResource`**— recupere todas as tags atualmente aplicadas a um Marketplace Revenue Share.

As alocações não oferecem suporte a tags. Marque o compartilhamento principal para organizar as alocações relacionadas.

Cada operação de marcação aceita o compartilhamento `Arn` (formato`arn:{partition}:partnercentral:{region}:{account-id}:catalog/{catalog}/marketplace-revenue-share/{productId}`) como identificador do recurso. A AWS criação de tags padrão também é suportada por meio do `Tags` campo ativado. `CreateMarketplaceRevenueShare`

Compatível AWS Regiões

As seções a seguir fornecem informações sobre as regiões da AWS suportadas.

Tópicos

- Compatível AWS regiões para o AWS API de conta do Partner Central
- Compatível AWS regiões para o AWS API de vendas da Central de Parceiros
- Compatível AWS regiões para o AWS API de benefícios do Partner Central
- Compatível AWS regiões para o AWS API do Partner Central Channel
- Compatível AWS regiões para o AWS API de medição de receita do Partner Central

Compatível AWS regiões para o AWS API de conta do Partner Central

Você pode acessar o serviço de conta do AWS Partner Central de qualquer região da AWS Leste dos EUA (Norte da Virgínia) com o seguinte endpoint.

```
partnercentral-account.us-east-1.api.aws
```

Compatível AWS regiões para o AWS API de vendas da Central de Parceiros

Você pode acessar o serviço de vendas do AWS Partner Central de qualquer região da AWS Leste dos EUA (Norte da Virgínia) com o seguinte endpoint.

```
partnercentral-selling.us-east-1.api.aws
```

Compatível AWS regiões para o AWS API de benefícios do Partner Central

Você pode acessar o serviço de benefícios do AWS Partner Central de qualquer região da AWS Leste dos EUA (Norte da Virgínia) com o seguinte endpoint.

```
partnercentral-benefits.us-east-1.api.aws
```

Compatível AWS regiões para o AWS API do Partner Central Channel

Você pode acessar o serviço de gerenciamento de canais do AWS Partner Central de qualquer região comercial da AWS com o seguinte endpoint.

```
partnercentral-channel.global.api.aws
```

Assinando solicitações com Sigv4a

Como a API do AWS Partner Central Channel usa um endpoint global, as solicitações são assinadas com Signature Version 4a (SigV4a), uma extensão do SigV4 que oferece suporte à assinatura com criptografia assimétrica, permitindo assinaturas verificáveis em várias regiões. AWS

Usando o AWS SDK ou a CLI

Se você usa um AWS SDK ou a AWS CLI, a assinatura Sigv4a é processada automaticamente quando você chama o endpoint global. Não é exigida nenhuma configuração adicional.

Assinando solicitações manualmente

Se você assinar solicitações manualmente sem um SDK, esteja ciente das seguintes diferenças em relação ao SigV4 padrão:

- O Sigv4a deriva um par de chaves do Algoritmo de Assinatura Digital de Curva Elíptica (ECDSA) de sua chave de acesso secreta existente AWS , em vez de usar a derivação de chave. HMAC-based
- O escopo da credencial é usado * para o componente de região em vez de um nome de região específico, porque a assinatura é válida em todas as regiões.

Para obter mais informações sobre como o SigV4a funciona, consulte AWS Signature versão 4 para solicitações de API. Para exemplos de código de assinatura SigV4a, consulte o projeto sigv4a-signing-examples em. GitHub

Compatível AWS regiões para o AWS API de medição de receita do Partner Central

Você pode acessar o serviço AWS Partner Central Revenue Measurement de qualquer região comercial da AWS com o seguinte endpoint.

```
partnercentral-prm.global.api.aws
```

Assinando solicitações com Sigv4a

Como o serviço de medição de receita usa um endpoint global, as solicitações são assinadas com o Signature Version 4a (SIGV4a), uma extensão do SigV4 que oferece suporte à assinatura com criptografia assimétrica, permitindo assinaturas verificáveis em várias regiões. AWS

Usando o AWS SDK ou a CLI

Se você usa um AWS SDK ou a AWS CLI, a assinatura Sigv4a é processada automaticamente quando você chama o endpoint global. Não é exigida nenhuma configuração adicional.

Assinando solicitações manualmente

Se você assinar solicitações manualmente sem um SDK, esteja ciente das seguintes diferenças em relação ao SigV4 padrão:

- O Sigv4a deriva um par de chaves do Algoritmo de Assinatura Digital de Curva Elíptica (ECDSA) de sua chave de acesso secreta existente AWS , em vez de usar a derivação de chave. HMAC-based
- O escopo da credencial é usado * para o componente de região em vez de um nome de região específico, porque a assinatura é válida em todas as regiões.

Para obter mais informações sobre como o SigV4a funciona, consulte AWS Signature versão 4 para solicitações de API. Para exemplos de código de assinatura SigV4a, consulte o projeto sigv4a-signing-examples em. GitHub

Permissões

As seções a seguir fornecem informações sobre permissões.

Tópicos

- Acesso à API da conta
- Acesso à API de vendas
- Acesso à API de benefícios
- Acesso à API do canal
- Acesso à API de medição de receita

Acesso à API da conta

O controle de acesso e as permissões são gerenciados pelo AWS Identity and Access Management(IAM). Esta seção fornece orientação para configurar as permissões necessárias para interagir com a API da conta.

Pré-requisitos

Antes de configurar as permissões, verifique se vocêConta da AWS está vinculado e criou as funções e os usuários do IAM necessários. Para obter mais informações, consulte Configuração e autenticação.

Utilizar AWS políticas gerenciadas

AWS fornece políticas gerenciadas que concedem as permissões necessárias para interagir com a API da conta. Para fornecer o acesso necessário para gerenciar os recursos da conta, anexe a `AWSPartnerCentralFullAccess` política às suas identidades do IAM. Para obter mais informações, consulte políticas AWS gerenciadas para usuários.

Atribuição de políticas a funções e usuários do IAM

Siga estas etapas para atribuir políticas às funções e aos usuários do IAM:

1. Faça login no AWS Management Console.
2. Navegue até o serviço IAM.
3. Selecione funções ou usuários e escolha a função ou o usuário do IAM ao qual você deseja anexar uma política.

4. Anexe a `AWSPartnerCentralFullAccess` política à função ou ao usuário do IAM selecionado.

Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

Gerenciando permissões usando chaves de condição

As chaves de condição nas políticas do IAM fornecem permissões em nível de recurso para quando aplicar as políticas de declaração. Você pode usar chaves de condição para especificar condições que determinam quando determinadas permissões são permitidas ou negadas.

Para obter mais informações, consulte Elementos da política JSON do IAM: operadores de condição.

Visão geral das chaves de condição

Chave de condição	Description	Ações aplicáveis	Valores válidos
Central de parceiros: catálogo	filtra o acesso pelo tipo da entidade de catálogo associada	todas as ações	AWS, caixa de areia

Resumo das permissões necessárias

Resumo das permissões necessárias

Ação	Description
central de parceiros: AcceptConnectionInvitation	permite aceitar convites de conexão
central de parceiros: AssociateAwsTrainingCertificationEmailDomain	permite associar domínios de e-mail AWS de certificação de treinamento
central de parceiros: CancelConnection	permite cancelar conexões
central de parceiros: CancelConnectionInvitation	permite cancelar convites de conexão
central de parceiros: CancelProfileUpdateTask	permite cancelar tarefas de atualização de perfil

Ação	Description
central de parceiros: CreateConnectionInvitation	permite criar convites de conexão
central de parceiros: CreatePartner	permite criar parceiros
central de parceiros: DisassociateAwsTrainingCertificationEmailDomain	permite desassociar domínios de e-mail de certificação AWS de treinamento
central de parceiros: GetAllianceLeadContact	permite recuperar detalhes de contato do líder da aliança
central de parceiros: GetConnection	permite recuperar detalhes da conexão
central de parceiros: GetConnectionInvitation	permite recuperar detalhes do convite de conexão
central de parceiros: GetConnectionPreferences	permite recuperar as preferências de conexão
central de parceiros: GetPartner	permite recuperar detalhes do parceiro
central de parceiros: GetProfileUpdateTask	permite recuperar detalhes da tarefa de atualização do perfil
central de parceiros: GetProfileVisibility	permite recuperar as configurações de visibilidade do perfil
central de parceiros: GetVerification	permite recuperar detalhes de verificação
central de parceiros: ListConnectionInvitations	permite listar convites de conexão
central de parceiros: ListConnections	permite listar conexões
central de parceiros: ListPartners	permite listar parceiros
central de parceiros: PutAllianceLeadContact	permite atualizar os detalhes de contato do líder da aliança

Ação	Description
central de parceiros: PutProfileVisibility	permite atualizar as configurações de visibilidade do perfil
central de parceiros: RejectConnectionInvitation	permite rejeitar convites de conexão
central de parceiros: SendEmailVerificationCode	permite o envio de códigos de verificação por e-mail
central de parceiros: StartProfileUpdateTask	permite iniciar tarefas de atualização de perfil
central de parceiros: StartVerification	permite iniciar processos de verificação
central de parceiros: UpdateConnectionPreferences	permite atualizar as preferências de conexão

Acesso à API de vendas

O controle de acesso e as permissões são gerenciados pelo AWS Identity and Access Management(IAM). Esta seção fornece orientação para configurar as permissões necessárias para interagir com a API, incluindo as permissões necessárias para listar AWS Marketplace entidades.

Pré-requisitos

Antes de configurar as permissões, verifique se vocêConta da AWS está vinculado ao Partner Central e criou as funções e os usuários do IAM necessários. Para obter mais informações, consulte Configuração e autenticação.

Utilizar AWS políticas gerenciadas

AWS fornece políticas gerenciadas que concedem as permissões necessárias para interagir com a API. Para fornecer o acesso necessário para gerenciar oportunidades, anexe a `AWSPartnerCentralOpportunityManagement` política às suas identidades do IAM. Para obter mais informações, consulte políticas AWS gerenciadas para usuários do AWS Partner Central.

AWSPartnerCentralOpportunityManagement política

Essa política concede acesso total às ações de gerenciamento de oportunidades do Partner Central.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "OpportunityManagement",
            "Effect": "Allow",
            "Action": [
                "partnercentral:AcceptEngagementInvitation",
                "partnercentral:AssignOpportunity",
                "partnercentral:AssociateOpportunity",
                "partnercentral:CreateEngagement",
                "partnercentral:CreateEngagementInvitation",
                "partnercentral:CreateOpportunity",
                "partnercentral:CreateResourceSnapshot",
                "partnercentral:CreateResourceSnapshotJob",
                "partnercentral:DeleteResourceSnapshotJob",
                "partnercentral:DisassociateOpportunity",
                "partnercentral:GetAwsOpportunitySummary",
                "partnercentral:GetEngagement",
                "partnercentral:GetEngagementInvitation",
                "partnercentral:GetOpportunity",
                "partnercentral:GetResourceSnapshot",
                "partnercentral:GetResourceSnapshotJob",
                "partnercentral:ListEngagementByAcceptingInvitationTasks",
                "partnercentral:ListEngagementFromOpportunityTasks",
                "partnercentral:ListEngagementInvitations",
                "partnercentral:ListEngagementMembers",
                "partnercentral:ListEngagementResourceAssociations",
                "partnercentral:ListEngagements",
                "partnercentral:ListOpportunities",
                "partnercentral:ListResourceSnapshotJobs",
                "partnercentral:ListResourceSnapshots",
                "partnercentral:ListSolutions",
                "partnercentral:RejectEngagementInvitation",
                "partnercentral:StartEngagementByAcceptingInvitationTask",
                "partnercentral:StartEngagementFromOpportunityTask",
                "partnercentral:StartResourceSnapshotJob",
                "partnercentral:StopResourceSnapshotJob",
                "partnercentral:SubmitOpportunity",
                "partnercentral:UpdateOpportunity"
            ],
```

```
            "Resource": "*"
        },
        {
            "Sid": "ListingAWSMarketplaceEntities",
            "Effect": "Allow",
            "Action": ["aws-marketplace:ListEntities"],
            "Resource": "*"
        },
        {
            "Sid": "AWSMarketplaceOffersAccess",
            "Effect": "Allow",
            "Action": ["aws-marketplace:DescribeEntity"],
            "Resource": [
              "arn:aws:aws-marketplace:*:*:AWSMarketplace/Offer/*",
              "arn:aws:aws-marketplace:*:*:AWSMarketplace/OfferSet/*"
            ]
        }
    ]
}
```

Políticas personalizadas

Se as políticas gerenciadas não atenderem às suas necessidades, crie políticas personalizadas do IAM que concedam as permissões necessárias para seu caso de uso. O exemplo a seguir é uma política personalizada que concede permissões para listar AWS Marketplace entidades:

Example Exemplo de política personalizada

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "partnercentral:ListOpportunities",
                "aws-marketplace:ListEntities"
            ],
            "Resource": "*"
```

```
        }
    ]
}
```

Política permissiva personalizada

Essa política fornece amplo acesso às ações de venda do Partner Central, incluindo recursos que podem ser adicionados no futuro sem exigir atualizações de políticas. Ao usar a ação curinga`partnercentral:*`, essa política concede automaticamente acesso aos novos recursos de venda do Partner Central à medida que eles se tornam disponíveis, reduzindo a necessidade de atualizações manuais. Essa política também inclui permissões para interagir com AWS Marketplace entidades, o que ajuda a garantir que o acesso seja mantido tanto para ações de venda quanto para ações do Marketplace.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Action":
            [
                "partnercentral:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action":
            [
                "aws-marketplace:ListEntities",
                "aws-marketplace:DescribeEntity"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action":
            [
```

```
                "aws-marketplace:SearchAgreements",
                "aws-marketplace:DescribeAgreement"
            ],
            "Resource": "*",
            "Condition":
            {
                "StringEquals":
                {
                    "aws-marketplace:PartyType": "Proposer"
                }
            }
        }
    ]
}
```

Atribuição de políticas a funções e usuários do IAM

Siga estas etapas para atribuir políticas às funções e aos usuários do IAM:

1. Faça login no AWS Management Console.
2. Navegue até o serviço IAM.
3. Selecione funções ou usuários e escolha a função ou o usuário do IAM ao qual você deseja anexar uma política.
4. Anexe a `AWSPartnerCentralOpportunityManagement` política ou sua política personalizada à função ou usuário do IAM selecionado.

Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

Gerenciando permissões usando chaves de condição

As chaves de condição nas políticas do IAM fornecem permissões em nível de recurso para quando aplicar as políticas de declaração. Você pode usar chaves de condição para especificar condições que determinam quando determinadas permissões são permitidas ou negadas.

Para obter mais informações, consulte Elementos da política JSON do IAM: operadores de condição.

Visão geral das chaves de condição

Chave de condição	Description	Ações aplicáveis	Valores válidos
Central de parceiros: catálogo	filtra o acesso pelo tipo da entidade de catálogo associada	todas as ações	AWS, caixa de areia
mercado da AWS: PartyType	filtra o acesso com base no tipo de parte (por exemplo, proponente)	SearchAgreements, DescribeAgreement	proponente

Resumo das permissões necessárias

Resumo das permissões necessárias

Ação	Description
central de parceiros: CreateOpportunity	permite criar oportunidades
central de parceiros: UpdateOpportunity	permite atualizar oportunidades
central de parceiros: ListOpportunities	permite listar oportunidades
central de parceiros: GetOpportunity	permite recuperar detalhes da oportunidade
central de parceiros: ListSolutions	permite listar soluções
central de parceiros: AssociateOpportunity	permite associar oportunidades com outras entidades
central de parceiros: DisassociateOpportunity	permite dissociar oportunidades de outras entidades
central de parceiros: AcceptEngagementInvitation	permite aceitar convites de noivado
central de parceiros: RejectEngagementInvitation	permite rejeitar convites de engajamento

Ação	Description
central de parceiros: GetEngagementInvitation	permite recuperar detalhes do convite de engajamento
central de parceiros: ListEngagementInvitations	permite listar convites de engajamento
central de parceiros: SubmitOpportunity	permite o envio de oportunidades
central de parceiros: GetAwsOpportunitySummary	permite recuperar o resumo da AWS oportunidade
central de parceiros: StartProspectingFromEngagementTask	Cria uma tarefa de prospecção a partir de contratos
central de parceiros: GetProspectingFromEngagementTask	Retorna detalhes da tarefa de prospecção
central de parceiros: ListProspectingFromEngagementTasks	Retorna uma lista de tarefas de prospecção
mercado da AWS: ListEntities	permite listar AWS Marketplace entidades
mercado da AWS: DescribeEntity	permite descrever AWS Marketplace entidades
mercado da AWS: SearchAgreements	permite pesquisar contratos em AWS Marketplace
mercado da AWS: DescribeAgreement	permite descrever acordos em AWS Marketplace

Acesso à API de benefícios

O controle de acesso e as permissões são gerenciados pelo AWS Identity and Access Management(IAM). Esta seção fornece orientação para configurar as permissões necessárias para interagir com a API de benefícios.

Pré-requisitos

Antes de configurar as permissões, verifique se vocêConta da AWS está vinculado e criou as funções e os usuários do IAM necessários. Para obter mais informações, consulte Configuração e autenticação.

Utilizar AWS políticas gerenciadas

AWS fornece políticas gerenciadas que concedem as permissões necessárias para interagir com a API de benefícios. Para fornecer o acesso necessário para gerenciar recursos de benefícios, anexe a `AWSPartnerCentralFullAccess` política às suas identidades do IAM. Para obter mais informações, consulte políticas AWS gerenciadas para usuários.

Atribuição de políticas a funções e usuários do IAM

Siga estas etapas para atribuir políticas às funções e aos usuários do IAM:

1. Faça login no AWS Management Console.
2. Navegue até o serviço IAM.
3. Selecione funções ou usuários e escolha a função ou o usuário do IAM ao qual você deseja anexar uma política.
4. Anexe a `AWSPartnerCentralFullAccess` política à função ou ao usuário do IAM selecionado.

Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

Gerenciando permissões usando chaves de condição

As chaves de condição nas políticas do IAM fornecem permissões em nível de recurso para quando aplicar as políticas de declaração. Você pode usar chaves de condição para especificar condições que determinam quando determinadas permissões são permitidas ou negadas.

Para obter mais informações, consulte Elementos da política JSON do IAM: operadores de condição.

Visão geral das chaves de condição

Chave de condição	Description	Ações aplicáveis	Valores válidos
Central de parceiros: catálogo	filtra o acesso pelo tipo da entidade de catálogo associada	todas as ações	AWS, caixa de areia

Resumo das permissões necessárias

Resumo das permissões necessárias

Ação	Description
central de parceiros: AmendBenefitApplication	permite alterar os aplicativos de benefícios
central de parceiros: AssociateBenefitApplicationResource	permite associar recursos a aplicativos de benefícios
central de parceiros: CancelBenefitApplication	permite o cancelamento de solicitações de benefícios
central de parceiros: CreateBenefitApplication	permite criar aplicativos de benefícios
central de parceiros: DisassociateBenefitApplicationResource	permite desassociar recursos dos aplicativos de benefícios
central de parceiros: GetBenefit	permite recuperar detalhes do benefício
central de parceiros: GetBenefitAllocation	permite recuperar detalhes da alocação de benefícios
central de parceiros: GetBenefitApplication	permite recuperar detalhes do aplicativo de benefícios
central de parceiros: ListBenefitAllocations	permite listar alocações de benefícios
central de parceiros: ListBenefitApplications	permite listar aplicativos de benefícios
central de parceiros: ListBenefits	permite listar benefícios
central de parceiros: RecallBenefitApplication	permite o recall de aplicativos de benefícios
central de parceiros: SubmitBenefitApplication	permite o envio de solicitações de benefícios
central de parceiros: UpdateBenefitApplication	permite atualizar os aplicativos de benefícios

Acesso à API do canal

O controle de acesso e as permissões são gerenciados pelo AWS Identity and Access Management(IAM). Esta seção fornece orientação para configurar as permissões necessárias para interagir com a API do AWS Partner Central Channel.

Configuração da conta de gerenciamento de canais

Os recursos de gerenciamento de canais no AWS Partner Central exigem que as funções do IAM sejam implantadas em dois tipos de AWS contas:

1. AWS Central de parceiros vinculada Conta da AWS:

   Esse é o Conta da AWS associado à sua conta do AWS Partner Network e do Partner Central. Há apenas uma Central de AWS Parceiros vinculada Conta da AWS por parceiro. Como uma atividade de configuração única, você criará uma função do IAM Conta da AWS usando uma política gerenciada por canais do Partner Central.
2. Conta de gerenciamento do programa Conta da AWS:

   Isso éConta da AWS usado como uma Bill-Transfer conta para gerenciar o faturamento e o pagamento do consumo do cliente final. Isso Conta da AWSé relatado como uma conta de gerenciamento do programa no gerenciamento de canais do AWS Partner Central. Você pode ter várias contas de gerenciamento do programa e precisará criar uma função do IAM em cada Conta da AWS relatório como uma conta de gerenciamento do programa.

Para cada Conta da AWS tipo, você precisará associar diferentes funções do IAM a políticas específicas de permissão e confiança.

Acesso para o AWS Central de parceiros vinculada Conta da AWS

No link do AWS Partner Central Conta da AWS, crie uma função do IAM para gerenciar os recursos do canal, que serão concedidos aos usuários do Partner Central. Essa função do IAM permite que os usuários visualizem e gerenciem recursos de gerenciamento de canais no AWS Partner Central. O nome da função deve começar com`PartnerCentralRoleFor`, e o nome recomendado é`PartnerCentralRoleForChannel`. Para configurar a função, faça o seguinte:

- Adicione a seguinte política de confiança personalizada para permitir que administradores de nuvem e líderes de alianças do AWS Partner Central mapeiem as funções do IAM para os usuários do Partner Central:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "partnercentral-account-management.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Associe a política [AWSPartnerCentralChannelManagement](#)gerenciada à `PartnerCentralRoleForChannel` função.

Acesso à conta de gerenciamento do programa AWS conta (s)

Em cada uma delas Conta da AWS relatada como uma conta de gerenciamento do programa no AWS Partner Central, crie as seguintes funções do IAM, com os nomes obrigatórios abaixo:

- `PartnerCentralChannelHandshakeApprovalManagement`: utilize essa função para gerenciar os apertos de mão entre o Partner Central e o gerenciamento do seu programa.Conta da AWS Essa função do IAM pode ser utilizada para responder e ativar suas contas de gerenciamento do programa no AWS Partner Central. Ao criar essa função, associe a política `AWSPartnerCentralChannelHandshakeApprovalManagement` gerenciada.
- `PartnerCentralChannelBillingTransferReadOnly`: utilize essa função entre contas para permitir a visibilidade do status da transferência de faturamento a partir do AWS Partner Central. Essa função compartilhará o status da transferência de faturamento do seu Conta da AWS para o AWS Partner Central para visualizar centralmente o status da transferência de faturamento. Se essa função for criada, os usuários no AWS Partner Central poderão ver o status da transferência de faturamento no AWS Partner Central. No entanto, essa função não concede aos usuários do AWS Partner Central permissão para acessar as transferências de faturamento no console AWS Billing and Cost Management.
  - Política de confiança:
    - No JSON abaixo, substitua o por seu ID real `{PartnerCentralLinkedAccountID}` de Conta da AWS 12 dígitos vinculado ao AWS Partner Central ao criar a função.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::{PartnerCentralLinkedAccountId}:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Política de permissão:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "BillingTransferReadOnly",
      "Effect": "Allow",
      "Action": [
        "organizations:DescribeResponsibilityTransfer",
        "organizations:ListHandshakesForOrganization",
        "organizations:ListInboundResponsibilityTransfers",
        "organizations:ListOutboundResponsibilityTransfers"
    ],
      "Resource": "*"
    }
  ]
}
```

- `PartnerCentralChannelBillingTransferManagement`(opcional): utilize essa função entre contas para permitir a criação e o gerenciamento de transferências de faturamento do AWS Partner Central. Essa função permitirá que os usuários do AWS Partner Central acessem a função do IAM que contém a política `AWSPartnerCentralChannelManagement` gerenciada para gerenciar as transferências de faturamento na conta Conta da AWS de gerenciamento do programa. Usando essa função entre contas, os usuários do AWS Partner Central poderão acessar e gerenciar as transferências de faturamento no console Billing and AWS Cost Management usando o botão AWS“Manage AWS in Billing” no gerenciamento de canais do Partner Central.

- Política de confiança:
  - No JSON abaixo, substitua o por seu ID real `{PartnerCentralLinkedAccountID}` de Conta da AWS 12 dígitos vinculado ao AWS Partner Central ao criar a função.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::{PartnerCentralLinkedAccountId}:root"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- Política de permissão:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "BillingTransferManagement",
      "Effect": "Allow",
      "Action": [
        "billingconductor:CreateBillingGroup",
        "billingconductor:ListPricingPlans",
        "organizations:AcceptHandshake",
        "organizations:CancelHandshake",
        "organizations:DeclineHandshake",
        "organizations:DescribeResponsibilityTransfer",
        "organizations:InviteOrganizationToTransferResponsibility",
        "organizations:ListHandshakesForAccount",
        "organizations:ListHandshakesForOrganization",
        "organizations:ListInboundResponsibilityTransfers",
        "organizations:ListOutboundResponsibilityTransfers",
        "organizations:TerminateResponsibilityTransfer",
        "organizations:UpdateResponsibilityTransfer"
    ],
      "Resource": "*"
    }
```

```
    ]
}
```

Utilizar AWS políticas gerenciadas

AWS fornece políticas gerenciadas que concedem as permissões necessárias para interagir com a API do canal. Para fornecer o acesso necessário para gerenciar todos os recursos do canal, anexe a `AWSPartnerCentralChannelManagement` política às suas identidades do IAM. Para fornecer o acesso necessário para visualizar e responder às solicitações de handshake do canal, anexe a `AWSPartnerCentralChannelHandshakeApprovalManagement` política às suas identidades do IAM. Para obter mais informações, consulte políticas AWS gerenciadas para usuários do AWS Partner Central.

AWSPartnerCentralChannelManagement política

Essa política concede acesso total às ações de gerenciamento de canais do Partner Central.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ChannelManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateProgramManagementAccount",
        "partnercentral:UpdateProgramManagementAccount",
        "partnercentral:DeleteProgramManagementAccount",
        "partnercentral:ListProgramManagementAccounts",
        "partnercentral:GetProgramManagementAccount",
        "partnercentral:CreateRelationship",
        "partnercentral:UpdateRelationship",
        "partnercentral:DeleteRelationship",
        "partnercentral:GetRelationship",
        "partnercentral:ListRelationships",
        "partnercentral:CreateChannelHandshake",
        "partnercentral:AcceptChannelHandshake",
        "partnercentral:RejectChannelHandshake",
        "partnercentral:CancelChannelHandshake",
        "partnercentral:ListChannelHandshakes"
      ],
      "Resource": "*",
```

```
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    },
    {
      "Sid": "ChannelBillingTransferRoleAccess",
      "Effect": "Allow",
      "Action": [
        "sts:AssumeRole"
      ],
      "Resource": [
        "arn:aws:iam::*:role/PartnerCentralChannelBillingTransferManagement",
        "arn:aws:iam::*:role/PartnerCentralChannelBillingTransferReadOnly"
      ]
    },
    {
      "Sid": "TaggingAccess",
      "Effect": "Allow",
      "Action": [
        "partnercentral:TagResource",
        "partnercentral:UntagResource",
        "partnercentral:ListTagsForResource"
      ],
      "Resource": [
        "arn:aws:partnercentral:*:*:catalog/*/program-management-account/*",
        "arn:aws:partnercentral:*:*:catalog/*/channel-handshake/*"
      ],
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    }
  ]
}
```

AWSPartnerCentralChannelHandshakeApprovalManagement política

Essa política concede acesso para visualizar e responder às solicitações de handshake do canal. Essa política deve ser aplicada às funções nas AWS contas que recebem solicitações de handshake do canal.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ChannelHandshakeManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:ListChannelHandshakes",
        "partnercentral:AcceptChannelHandshake",
        "partnercentral:RejectChannelHandshake"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "AWS",
            "Sandbox"
          ]
        }
      }
    }
  ]
}
```

Atribuição de políticas a funções e usuários do IAM

Siga estas etapas para atribuir políticas às funções e aos usuários do IAM:

1. Faça login no AWS Management Console.
2. Navegue até o serviço IAM.
3. Selecione funções ou usuários e escolha a função ou o usuário do IAM ao qual você deseja anexar uma política.
4. Anexe a `AWSPartnerCentralChannelManagement` política ou sua política personalizada à função ou usuário do IAM selecionado.

Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

Gerenciando permissões usando chaves de condição

As chaves de condição nas políticas do IAM fornecem permissões em nível de recurso para quando aplicar as políticas de declaração. Você pode usar chaves de condição para especificar condições que determinam quando determinadas permissões são permitidas ou negadas.

Para obter mais informações, consulte Elementos da política JSON do IAM: operadores de condição.

Visão geral das chaves de condição

Chave de condição	Description	Ações aplicáveis	Valores válidos
Central de parceiros: catálogo	Filtra o acesso pelo tipo da entidade de catálogo associada	Todas as ações de gerenciamento de canais	AWS, Sandbox
central de parceiros: ChannelHandshakeType	Filtra o acesso com base no tipo de handshake do canal	CreateChannelHandshake, AcceptChannelHandshake, RejectChannelHandshake, CancelChannelHandshake, ListChannelHandshakes	CONTA_DE_GESTÃO_PROGRAMA, PERIODO_SERVIÇO_INICIAL, PERIODO_DE_SERVIÇO_REVOGAR

Resumo das permissões necessárias

Resumo das permissões necessárias

Ação	Description
central de parceiros: CreateProgramManagementAccount	permite criar contas de gerenciamento de programas
central de parceiros: UpdateProgramManagementAccount	permite atualizar as contas de gerenciamento do programa

Ação	Description
central de parceiros: DeleteProgramManagementAccount	permite excluir contas de gerenciamento do programa
central de parceiros: ListProgramManagementAccounts	permite listar contas de gerenciamento do programa
central de parceiros: GetProgramManagementAccount	permite recuperar detalhes da conta de gerenciamento do programa
central de parceiros: CreateRelationship	permite criar relacionamentos
central de parceiros: UpdateRelationship	permite atualizar relacionamentos
central de parceiros: DeleteRelationship	permite excluir relacionamentos
central de parceiros: GetRelationship	permite recuperar detalhes do relacionamento
central de parceiros: ListRelationships	permite listar relacionamentos
central de parceiros: CreateChannelHandshake	permite criar apertos de mão de canais
central de parceiros: AcceptChannelHandshake	permite aceitar apertos de mão do canal
central de parceiros: RejectChannelHandshake	permite rejeitar apertos de mão do canal
central de parceiros: CancelChannelHandshake	permite cancelar apertos de mão do canal
central de parceiros: ListChannelHandshakes	permite listar os apertos de mão do canal

Acesso à API de medição de receita

O controle de acesso e as permissões são gerenciados pelo AWS Identity and Access Management(IAM). Esta seção fornece orientação para configurar as permissões necessárias para interagir com a API de Medição de Receita, incluindo as permissões necessárias para listar AWS Marketplace entidades.

Pré-requisitos

Antes de configurar as permissões, verifique se vocêConta da AWS está vinculado ao Partner Central e criou as funções e os usuários do IAM necessários. Para obter mais informações, consulte Configuração e autenticação.

Utilizar AWS políticas gerenciadas

AWS fornece políticas gerenciadas que concedem as permissões necessárias para interagir com a API de Medição de Receita. Duas políticas gerenciadas estão disponíveis, dependendo da sua personalidade:

- Parceiros — anexe a `AWSPartnerCentralRevenueAttributionManagement` política às suas identidades do IAM.
- Clientes — anexe a `AWSRevenueAttributionManagement` política às suas identidades do IAM.

Para obter mais informações, consulte políticas AWS gerenciadas para usuários do AWS Partner Central.

AWSPartnerCentralRevenueAttributionManagement política

Essa política fornece o acesso necessário às atividades de gerenciamento de atribuição de receita para AWS contas registradas no AWS Partner Central.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RevenueMeasurementCreation",
            "Effect": "Allow",
            "Action": [
                "partnercentral:CreateRevenueAttribution",
                "partnercentral:CreateMarketplaceRevenueShare",
                "partnercentral:CreateMarketplaceRevenueShareAllocation"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
```

```
                ]
            }
        }
    },
    {
        "Sid": "RevenueMeasurementListing",
        "Effect": "Allow",
        "Action": [
            "partnercentral:ListRevenueAttributions",
            "partnercentral:ListRevenueAttributionAllocations",
            "partnercentral:ListMarketplaceRevenueShares",
            "partnercentral:ListMarketplaceRevenueShareAllocations"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    },
    {
        "Sid": "RevenueAttributionManagement",
        "Effect": "Allow",
        "Action": [
            "partnercentral:GetRevenueAttribution",
            "partnercentral:UpdateRevenueAttribution",
            "partnercentral:GetRevenueAttributionAllocation",
            "partnercentral:StartRevenueAttributionAllocationsTask",
            "partnercentral:GetRevenueAttributionAllocationsTask"
        ],
        "Resource": "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*",
        "Condition": {
            "StringEquals": {
                "partnercentral:Catalog": [
                    "AWS",
                    "Sandbox"
                ]
            }
        }
    },
    {
```

```
            "Sid": "MarketplaceRevenueShareManagement",
            "Effect": "Allow",
            "Action": [
                "partnercentral:GetMarketplaceRevenueShare",
                "partnercentral:GetMarketplaceRevenueShareAllocation",
                "partnercentral:UpdateMarketplaceRevenueShareAllocation"
            ],
            "Resource": "arn:aws:partnercentral:*:*:catalog/*/marketplace-revenue-
share/*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "TaggingAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:TagResource",
                "partnercentral:UntagResource",
                "partnercentral:ListTagsForResource"
            ],
            "Resource": [
                "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*",
                "arn:aws:partnercentral:*:*:catalog/*/marketplace-revenue-share/*"
            ],
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "OpportunityAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:ListOpportunities",
```

```
                "partnercentral:GetOpportunity"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "PartnerResourceAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:ListPartners",
                "partnercentral:GetPartner"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "ListingAWSMarketplaceEntities",
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:ListEntities"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AWSMarketplaceEntityAccess",
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:DescribeEntity"
            ],
            "Resource": [
```

```
                "arn:aws:aws-marketplace:*:*:AWSMarketplace*/Offer/*"
            ]
        },
        {
            "Sid": "AmazonQPartnerAssistantAccess",
            "Effect": "Allow",
            "Action": [
                "q:StartConversation",
                "q:SendMessage",
                "q:GetConversation",
                "q:ListConversations",
                "q:PassRequest"
            ],
            "Resource": "*"
        }
    ]
}
```

AWSRevenueAttributionManagement política

Essa política fornece o acesso necessário às atividades de gerenciamento de atribuição de receita para AWS contas que não estão registradas no AWS Partner Central.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RevenueAttributionCreation",
            "Effect": "Allow",
            "Action": [
                "partnercentral:CreateRevenueAttribution"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
```

```
            "Sid": "RevenueAttributionListing",
            "Effect": "Allow",
            "Action": [
                "partnercentral:ListRevenueAttributions"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "RevenueAttributionManagement",
            "Effect": "Allow",
            "Action": [
                "partnercentral:GetRevenueAttribution",
                "partnercentral:UpdateRevenueAttribution"
            ],
            "Resource": "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "TaggingAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:TagResource",
                "partnercentral:UntagResource",
                "partnercentral:ListTagsForResource"
            ],
            "Resource": [
                "arn:aws:partnercentral:*:*:catalog/*/revenue-attribution/*"
            ],
            "Condition": {
```

```
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "AWS",
                        "Sandbox"
                    ]
                }
            }
        }
    ]
}
```

Atribuição de políticas a funções e usuários do IAM

Siga estas etapas para atribuir políticas às funções e aos usuários do IAM:

1. Faça login no AWS Management Console.
2. Navegue até o serviço IAM.
3. Selecione funções ou usuários e escolha a função ou o usuário do IAM ao qual você deseja anexar uma política.
4. Anexe a `AWSPartnerCentralRevenueAttributionManagement` política (para parceiros) ou a `AWSRevenueAttributionManagement` política (para clientes) à função ou usuário do IAM selecionado.

Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

Gerenciando permissões usando chaves de condição

As chaves de condição nas políticas do IAM fornecem permissões em nível de recurso para quando aplicar políticas de declaração. Você pode usar chaves de condição para especificar condições que determinam quando determinadas permissões são permitidas ou negadas.

Para obter mais informações, consulte Elementos da política JSON do IAM: operadores de condição.

Visão geral das chaves de condição

Chave de condição	Description	Ações aplicáveis	Valores válidos
Central de parceiros: catálogo	Filtra o acesso pelo tipo da entidade de catálogo associada	Todas as ações	AWS, Caixa de areia

Resumo das permissões necessárias

Resumo das permissões necessárias

Ação	Description
central de parceiros: CreateRevenueAttribution	Permite criar um novo registro de atribuição de receita
central de parceiros: UpdateRevenueAttribution	Permite atualizar um registro de atribuição de receita existente
central de parceiros: GetRevenueAttribution	Permite recuperar os detalhes de uma atribuiçã o de receita específica
central de parceiros: ListRevenueAttributions	Permite listar atribuições de receita com filtros opcionais
central de parceiros: GetRevenueAttribut ionAllocation	Permite recuperar uma única alocação por seu ID
central de parceiros: ListRevenueAttribu tionAllocations	Permite listar alocações comprometidas com suporte de filtragem
central de parceiros: StartRevenueAttrib utionAllocationsTask	Permite enviar um lote de até 250 alterações de alocação para processamento assíncrono
central de parceiros: GetRevenueAttribut ionAllocationsTask	Permite recuperar o status de uma tarefa de alocação enviada anteriormente
central de parceiros: CreateMarketplaceR evenueShare	Permite criar um novo recurso de compartil hamento de receita no mercado
central de parceiros: GetMarketplaceReve nueShare	Permite recuperar detalhes de uma participa ção específica na receita do mercado
central de parceiros: ListMarketplaceRev enueShares	Permite listar compartilhamentos de receita do mercado com filtros opcionais

Ação	Description
central de parceiros: CreateMarketplaceRevenueShareAllocation	Permite criar uma nova alocação de participação na receita do mercado
central de parceiros: GetMarketplaceRevenueShareAllocation	Permite recuperar detalhes da alocação de participação na receita de um mercado específico
central de parceiros: UpdateMarketplaceRevenueShareAllocation	Permite atualizar uma alocação de participação na receita de um mercado existente
central de parceiros: ListMarketplaceRevenueShareAllocations	Permite listar alocações sob uma participação na receita do mercado
central de parceiros: TagResource	Permite adicionar ou sobrescrever tags para um recurso
central de parceiros: UntagResource	Permite remover tags de um recurso
central de parceiros: ListTagsForResource	Permite listar tags associadas a um recurso

Cotas para o AWS API do Partner Central

As seções a seguir fornecem informações sobre cotas de serviço.

Tópicos

- Cotas para o AWS API de conta do Partner Central
- Cotas para o AWS API de vendas da Central de Parceiros
- Cotas para o AWS API de benefícios do Partner Central
- Cotas para o AWS API do Partner Central Channel
- Cotas para o AWS API de medição de receita do Partner Central

Cotas para o AWS API de conta do Partner Central

A API da conta do AWS Partner Central tem as seguintes cotas.

Cotas adicionais

Cotas adicionais

Nome de exibição	Catálogo	Description	Valor padrão
Convites de conexão aberta por conta	AWS	O número máximo de convites de conexão aberta que você pode manter com contas de parceiros no catálogo AWS	1.000
Conexões ativas por conta	AWS	O número máximo de conexões ativas que você pode manter com contas de parceiros no AWS catálogo	1.000
Domínios de e-mail por parceiro	AWS	O número máximo de domínios de e-mail que podem ser associados a uma conta de parceiro para certificação de AWS treinamento no catálogo AWS	50
Taxa de convites de conexão por conta	AWS	O número máximo de convites de conexão por dia que você pode enviar no catálogo AWS	50
Taxa de tarefas de atualização de perfil por conta	AWS	O número máximo de tarefas de atualização de perfil por dia que	10

Nome de exibição	Catálogo	Description	Valor padrão
		você pode iniciar no catálogo AWS	
Taxa de atualizaç ões de visibilidade do perfil por conta	AWS	O número máximo de atualizações de visibilidade do perfil por dia que você pode iniciar no catálogo AWS	5
Taxa de códigos de verificação de e-mail por endereço de e-mail	AWS	O número máximo de códigos de verificação de e-mail por dia que você pode solicitar no AWS catálogo	5
Convites de conexão aberta por conta	Sandbox	O número máximo de convites de conexão aberta que você pode manter com contas de parceiros no catálogo do Sandbox	1.000
Conexões ativas por conta	Sandbox	O número máximo de conexões ativas que você pode manter com contas de parceiros no catálogo do Sandbox	1.000

Nome de exibição	Catálogo	Description	Valor padrão
Domínios de e-mail por parceiro	Sandbox	O número máximo de domínios de e-mail que podem ser associados a uma conta de parceiro para certificação de AWS treinamento no catálogo do Sandbox	50
Taxa de convites de conexão por conta	Sandbox	O número máximo de convites de conexão por dia que você pode enviar no catálogo do Sandbox	50
Taxa de tarefas de atualização de perfil por conta	Sandbox	O número máximo de tarefas de atualização de perfil por dia que você pode iniciar no catálogo do Sandbox	10
Taxa de atualizaç ões de visibilidade do perfil por conta	Sandbox	O número máximo de atualizações de visibilidade do perfil por dia que você pode iniciar no catálogo do Sandbox	5

Nome de exibição	Catálogo	Description	Valor padrão
Taxa de códigos de verificação de e-mail por endereço de e-mail	Sandbox	O número máximo de códigos de verificação de e-mail por dia que você pode solicitar no catálogo do Sandbox. Nenhum e-mail é enviado do catálogo do Sandbox.	N/A

Entendendo e gerenciando cotas

Limitação de intervalo

Quando um limite de taxa de API for atingido, o serviço responderá com um ThrottlingException. Para lidar melhor com a limitação de taxa, a AWS recomenda implementar estratégias exponenciais de recuo e repetição em seu aplicativo.

Solicitar um aumento de cota

Se as cotas padrão não atenderem aos seus requisitos, você poderá solicitar um aumento de cota na página Service Quotas. O console Service Quotas é uma interface baseada em navegador que você pode usar para visualizar e gerenciar suas cotas de serviço. Você pode acessar as Cotas de Serviço de qualquer AWS Management Console página escolhendo-a na barra de navegação superior ou pesquisando por Cotas de Serviço no.AWS Management Console

Cotas para o AWS API de vendas da Central de Parceiros

AWS A API de vendas do Partner Central impõe cotas para garantir o uso justo e proteger o serviço contra uso indevido. Abaixo estão as cotas detalhadas para várias operações e associações de API por oportunidade.

Cotas de operação da API

Tipo	Operação de API	Cota (por conta de parceiro)
Leia as ações	GetOpportunity	10 por segundo; 100.000 por 24 horas

Tipo	Operação de API	Cota (por conta de parceiro)
	GetAwsOpportunitySummary	
	ListOpportunities	
	ListSolutions	
	GetEngagementInvitation	
	ListEngagementInvitations	
Prospecção, leitura, ações	GetProspectingFrom EngagementTask	100 por segundo
Prospecção, leitura, ações	ListProspectingFro mEngagementTasks	50 por segundo
Escrever ações	CreateOpportunity	1 por segundo; 10.000 por 24 horas
	UpdateOpportunity	
	AssociateOpportunity	
	DisassociateOpportunity	
	RejectEngagementInvitation	
	AssignOpportunity	
	StartEngagementFro mOpportunityTask	
	StartEngagementByAcceptingI nvitationTask	
Prospecção, escrita, ações	StartProspectingFr omEngagementTask	2 por segundo
Engajamento, escreva ações	CreateEngagement	15 por segundo

Cotas de associação por oportunidade

Entidade relacionada	Quota
AWS produtos	20 por oportunidade
Soluções para parceiros	10 por oportunidade
AWS Marketplace soluções	10 por oportunidade
AWS Marketplace produtos	10 por oportunidade
AWS Marketplace ofertas privadas	1 por oportunidade

Entendendo e gerenciando cotas

Limitação de intervalo

Quando um limite de taxa de API for atingido, o serviço responderá com um ThrottlingException. Para lidar melhor com a limitação de taxa, a AWS recomenda implementar estratégias exponenciais de recuo e repetição em seu aplicativo.

Janela de tempo para cotas

As cotas diárias são redefinidas em um período contínuo de 24 horas. Suas solicitações seriam limitadas, por exemplo, se você tivesse realizado 10.000 ações de gravação nas últimas 24 horas e estivesse tentando realizar a 10.001ª solicitação. Certifique-se de que os padrões de uso do seu aplicativo levem isso em consideração para evitar a limitação não intencional.

Solicitar um aumento de cota

Se as cotas padrão não atenderem aos seus requisitos, você poderá solicitar um aumento de cota na página Service Quotas. O console Service Quotas é uma interface baseada em navegador que você pode usar para visualizar e gerenciar suas cotas de serviço. Você pode acessar as Cotas de Serviço de qualquer AWS Management Console página escolhendo-a na barra de navegação superior ou pesquisando por Cotas de Serviço no.AWS Management Console

Cotas para o AWS API de benefícios do Partner Central

A API de benefícios do AWS Partner Central tem as seguintes cotas.

Cotas de solicitações

Cotas de solicitações

Operações de API	Cota (por AWS conta)
ListBenefits	20 por segundo
GetBenefit	20 por segundo
AmendBenefitApplication	10 por segundo
CancelBenefitApplication	10 por segundo
CreateBenefitApplication	10 por segundo
GetBenefitApplication	20 por segundo
ListBenefitApplications	30 por segundo
RecallBenefitApplication	10 por segundo
SubmitBenefitApplication	10 por segundo
UpdateBenefitApplication	10 por segundo
GetBenefitAllocation	20 por segundo
ListBenefitAllocations	20 por segundo
AssociateBenefitApplicationResource	10 por segundo
DisassociateBenefitApplicationResource	10 por segundo

Cotas adicionais

Cotas adicionais

Nome de exibição	Catálogo	Description	Valor padrão
Aplicativos de benefícios	AWS	O número máximo de aplicativos de	10.000

Nome de exibição	Catálogo	Description	Valor padrão
		benefícios que você pode criar no AWS catálogo	
Aplicativos de benefícios	Sandbox	O número máximo de aplicativos de benefícios que você pode criar no catálogo do Sandbox	10.000

Entendendo e gerenciando cotas

Limitação de intervalo

Quando um limite de taxa de API for atingido, o serviço responderá com um ThrottlingException. Para lidar melhor com a limitação de taxa, a AWS recomenda implementar estratégias exponenciais de recuo e repetição em seu aplicativo.

Solicitar um aumento de cota

Se as cotas padrão não atenderem aos seus requisitos, você poderá solicitar um aumento de cota na página Service Quotas. O console Service Quotas é uma interface baseada em navegador que você pode usar para visualizar e gerenciar suas cotas de serviço. Você pode acessar as Cotas de Serviço de qualquer AWS Management Console página escolhendo-a na barra de navegação superior ou pesquisando por Cotas de Serviço no.AWS Management Console

Cotas para o AWS API do Partner Central Channel

A API do AWS Partner Central Channel tem as seguintes cotas.

Cotas de solicitações

Cotas de solicitações

Operações de API	Cota (por AWS conta)
CreateProgramManagementAccount	3 por segundo

Operações de API	Cota (por AWS conta)
UpdateProgramManagementAccount	3 por segundo
ListProgramManagementAccounts	5 por segundo
DeleteProgramManagementAccount	3 por segundo
CreateChannelHandshake	3 por segundo
ListChannelHandshakes	5 por segundo
AcceptChannelHandshake	3 por segundo
RejectChannelHandshake	3 por segundo
CancelChannelHandshake	3 por segundo
CreateRelationship	3 por segundo
GetRelationship	5 por segundo
ListRelationships	5 por segundo
UpdateRelationship	3 por segundo
DeleteRelationship	3 por segundo

Cotas adicionais

Cotas adicionais

Nome de exibição	Catálogo	Description	Valor padrão
Contas de gerenciamento do programa	AWS	O número máximo de contas de gerenciamento de programas que você pode criar no AWS catálogo	50

Nome de exibição	Catálogo	Description	Valor padrão
Relacionamentos por conta de gerenciamento do programa	AWS	O número máximo de relacionamentos que você pode estabelecer por conta de gerenciamento do programa no AWS catálogo	1.000
Apertos de mão da conta de gerenciamento do programa por conta	AWS	O número máximo de apertos de mão ativos da conta de gerenciamento de programas no catálogo AWS	100
Início do período de serviço: apertos de mão por conta	AWS	O número máximo de apertos de mão ativos para estabelecer períodos de serviço no catálogo AWS	200
Revogue os apertos de mão do período de serviço por conta	AWS	O número máximo de apertos de mão ativos para revogar períodos de serviço no catálogo AWS	200
Taxa de apertos de mão da conta de gerenciamento de programas por dia	AWS	O número máximo de apertos de mão da conta de gerenciamento do programa que você pode enviar para a mesma AWS conta por dia no catálogo AWS	5

Nome de exibição	Catálogo	Description	Valor padrão
Taxa de apertos de mão do período de início do serviço por dia	AWS	O número máximo de apertos de mão do período de início do serviço que você pode enviar para a mesma AWS conta por dia no catálogo AWS	5
Taxa de apertos de mão do período de revogação do serviço por dia	AWS	O número máximo de apertos de mão do período de revogação do serviço que você pode enviar para a mesma AWS conta por dia no catálogo AWS	5
Contas de gerenciamento do programa	Sandbox	O número máximo de contas de gerenciamento de programas que você pode criar no catálogo do Sandbox	50
Relacionamentos por conta de gerenciamento do programa	Sandbox	O número máximo de relacionamentos que você pode estabelecer por conta de gerenciamento do programa no catálogo do Sandbox	1.000

Nome de exibição	Catálogo	Description	Valor padrão
Apertos de mão da conta de gerenciamento do programa por conta	Sandbox	O número máximo de apertos de mão ativos da conta de gerenciamento de programas no catálogo do Sandbox	100
Início do período de serviço: apertos de mão por conta	Sandbox	O número máximo de apertos de mão ativos para estabelecer períodos de serviço no catálogo do Sandbox	200
Revogue os apertos de mão do período de serviço por conta	Sandbox	O número máximo de apertos de mão ativos para revogar períodos de serviço no catálogo do Sandbox	200
Taxa de apertos de mão da conta de gerenciamento de programas por dia	Sandbox	O número máximo de apertos de mão da conta de gerenciamento do programa que você pode enviar para a mesma AWS conta por dia no catálogo do Sandbox	5

Nome de exibição	Catálogo	Description	Valor padrão
Taxa de apertos de mão do período de início do serviço por dia	Sandbox	O número máximo de apertos de mão do período de início do serviço que você pode enviar para a mesma AWS conta por dia no catálogo do Sandbox	5
Taxa de apertos de mão do período de revogação do serviço por dia	Sandbox	O número máximo de apertos de mão do período de revogação do serviço que você pode enviar para a mesma AWS conta por dia no catálogo do Sandbox	5

Entendendo e gerenciando cotas

Limitação de intervalo

Quando um limite de taxa de API for atingido, o serviço responderá com um ThrottlingException. Para lidar melhor com a limitação de taxa, a AWS recomenda implementar estratégias exponenciais de recuo e repetição em seu aplicativo.

Solicitar um aumento de cota

Se as cotas padrão não atenderem aos seus requisitos, você poderá solicitar um aumento de cota na página Service Quotas. O console Service Quotas é uma interface baseada em navegador que você pode usar para visualizar e gerenciar suas cotas de serviço. Você pode acessar as Cotas de Serviço de qualquer AWS Management Console página escolhendo-a na barra de navegação superior ou pesquisando por Cotas de Serviço no.AWS Management Console

Cotas para o AWS API de medição de receita do Partner Central

AWS A API de medição de receita do Partner Central impõe cotas para garantir o uso justo e proteger o serviço contra uso indevido. Abaixo estão as cotas detalhadas para várias operações de API.

Cotas de operação da API

Tipo	Operação de API	Cota (por conta de parceiro)
Leia as ações	GetRevenueAttribution	50 por segundo
	GetRevenueAttributionAllocation	
	GetRevenueAttributionAllocationsTask	
	GetMarketplaceRevenueShare	
	GetMarketplaceRevenueShareAllocation	
Listar ações	ListRevenueAttributions	10 por segundo
	ListRevenueAttributionAllocations	
	ListMarketplaceRevenueShares	
	ListMarketplaceRevenueShareAllocations	
	ListTagsForResource	
Escrever ações	CreateRevenueAttribution	2 por segundo
	UpdateRevenueAttribution	

Tipo	Operação de API	Cota (por conta de parceiro)
	CreateMarketplaceRevenueShare	
	CreateMarketplaceRevenueShareAllocation	
	UpdateMarketplaceRevenueShareAllocation	
	TagResource	
	UntagResource	
Ações de gravação assíncrona	StartRevenueAttributionAllocationsTask	1 por segundo

Entendendo e gerenciando cotas

Limitação de intervalo

Quando um limite de taxa de API for atingido, o serviço responderá com um ThrottlingException. Para lidar melhor com a limitação de taxa,AWS recomenda implementar estratégias exponenciais de recuo e repetição em seu aplicativo.

Solicitar um aumento de cota

Se as cotas padrão não atenderem aos seus requisitos, você poderá solicitar um aumento de cota na página Service Quotas. O console Service Quotas é uma interface baseada em navegador que você pode usar para visualizar e gerenciar suas cotas de serviço. Você pode acessar as Cotas de Serviço de qualquer AWS Management Console página escolhendo-a na barra de navegação superior ou pesquisando por Cotas de Serviço no.AWS Management Console

Registro para o AWS API do Partner Central

As seções a seguir fornecem informações sobre registro em log.

Tópicos

- Registrando o AWS API de conta do Partner Central
- Registrando o AWS API de vendas da Central de Parceiros
- Registrando o AWS API de benefícios do Partner Central
- Registrando o AWS API do Partner Central Channel
- Registrando o AWS API de atribuição de receita do Partner Central

Registrando o AWS API de conta do Partner Central

AWS CloudTrailé um serviço que permite governança, conformidade, auditoria operacional e auditoria de risco de seu.Conta da AWS Com AWS CloudTrail, você pode registrar, monitorar continuamente e reter as atividades da conta relacionadas às ações em toda a sua AWS infraestrutura.AWS A atividade da API da conta do Partner Central é registrada como eventos em CloudTrail. Você pode criar uma trilha, uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3.

Visão geral do

A API de conta do AWS Partner Central está integrada com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS Partner Central. CloudTrail captura todas as chamadas de API para a API da conta do AWS Partner Central como eventos. As chamadas capturadas incluem chamadas do AWS Partner Central e de chamadas de código para as operações da API da conta do AWS Partner Central.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para a API de contas do AWS Partner Central. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação feita à API da Conta do AWS Partner Central, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Noções básicas AWS Entradas do arquivo de log da Partner Central Account API

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3. Quando sua trilha acompanha os eventos da API da Conta do AWS Partner Central, CloudTrail processa os eventos como arquivos de log em todas as regiões. Cada arquivo de log pode conter um ou mais eventos.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `CreatePartner` ação na API de conta do AWS Partner Central:


```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/CloudTrailTestUser",
        "accountId": "123456789012",
        "accessKeyId": "ABCDEFGHIJKLMNOP1234",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLE52AGFT725JGDZ",
                "arn": "arn:aws:iam::123456789012:role/ExampleRole",
                "accountId": "123456789012",
                "userName": "ExampleRole"
            },
            "attributes": {
                "creationDate": "2025-11-07T19:45:28Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2025-11-07T19:45:58Z",
    "eventSource": "partnercentral-account.amazonaws.com",
    "eventName": "CreatePartner",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "PostmanRuntime/7.18.0",
    "requestParameters": {
        "catalog": "AWS",
        "clientToken": "abcdef12-3456-7890-bcde-f123456789ab",
        "legalName": "ExampleLegalName",
        "primarySolutionType": "VALUE_ADDED_RESALE_AWS_SERVICES",
        "allianceLeadContact": {
            "firstName": "ExampleFirstName",
            "lastName": "ExampleLastName",
            "email": "example@domain.com",
            "businessTitle": "ExampleBusinessTitle"
        },
        "emailVerificationCode": "ExampleVerificationCode",
```

```
        "tags": [
            {
                "key": "office",
                "value": "1"
            }
        ]
    },
    "responseElements": {
        "catalog": "AWS",
        "arn": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/partner/
EXAMPLE_PARTNER_ID",
        "id": "EXAMPLE_PARTNER_ID",
        "legalName": "ExampleLegalName",
        "createdAt": "2025-11-07T19:45:57.867007329Z",
        "profile": {
            "primarySolutionType": "VALUE_ADDED_RESALE_AWS_SERVICES"
        },
        "allianceLeadContact": {
            "firstName": "ExampleFirstName",
            "lastName": "ExampleLastName",
            "email": "example@domain.com",
            "businessTitle": "ExampleBusinessTitle"
        }
    },
    "requestID": "12345678-1234-5678-9abc-def012345678",
    "eventID": "87654321-4321-8765-cba9-fed098765432",
    "readOnly": false,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::PartnerCentral::Partner",
            "ARN": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/partner/
EXAMPLE_PARTNER_ID"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "partnercentral-account.partner.us-east-1.api.aws"
    }
```

```
}
```

Neste exemplo, a `CreatePartner` ação foi chamada pelo usuário do IAM chamado CloudTrailTestUser. A solicitação foi feita em 7 de novembro de 2025 às 19:45:58 UTC. A solicitação criou um novo parceiro com ID `EXAMPLE_PARTNER_ID` para o catálogo da AWS com o nome legal "ExampleLegalName".

Campos em AWS Entradas do arquivo de log da Partner Central Account API

Cada entrada em um arquivo de CloudTrail log contém informações sobre quem fez uma solicitação, os recursos utilizados na solicitação e os elementos de resposta retornados pela API de conta do AWS Partner Central. A lista de campos em uma entrada de registro, como `eventVersionuserIdentity`, `eeventTime`, fornece informações detalhadas sobre a ação. Por exemplo, o `sourceIPAddress` campo mostra o endereço IP do qual a solicitação foi feita.

Registrando o AWS API de vendas da Central de Parceiros

AWS CloudTrailé um serviço que permite governança, conformidade, auditoria operacional e auditoria de risco de seu.Conta da AWS Com AWS CloudTrail, você pode registrar, monitorar continuamente e reter as atividades da conta relacionadas às ações em toda a sua AWS infraestrutura.AWS A atividade da API do Partner Central é registrada como eventos em CloudTrail. Você pode criar uma trilha, uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3.

Visão geral do

A API do AWS Partner Central é integrada com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS Partner Central. CloudTrail captura todas as chamadas de API para o AWS Partner Central como eventos. As chamadas capturadas incluem chamadas do AWS Partner Central e de chamadas de código para as operações da API do AWS Partner Central.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para o AWS Partner Central. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita ao AWS Partner Central, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Noções básicas AWS Entradas do arquivo de registro da API de vendas do Partner Central

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3. Quando sua trilha acompanha os eventos do AWS Partner Central, CloudTrail processa os eventos como arquivos de log em todas as regiões. Cada arquivo de log pode conter um ou mais eventos.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `ListOpportunities` ação no AWS Partner Central:


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGHIJKLMNOP12345",
        "arn": "arn:aws:iam::123456789010:user/CloudTrailTestUser",
        "accountId": "123456789010",
        "accessKeyId": "ABCDEFGHIJKLMNOP1234",
        "userName": "CloudTrailTestUser"
    },
    "eventTime": "2023-10-17T21:49:23Z",
    "eventSource": "partnercentral-selling.amazonaws.com",
    "eventName": "ListOpportunities",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "PostmanRuntime/7.18.0",
    "requestParameters": {
        "MaxResults": 20
    },
    "responseElements": null,
    "requestID": "fEXAMPLE-cb3e-4e21-86fd-6b3EXAMPLEd1",
    "eventID": "7EXAMPLE-97d6-4139-91e3-01aEXAMPLE48",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789010"
}
```


Neste exemplo, a `ListOpportunities` ação foi chamada pelo usuário do IAM chamado CloudTrailTestUser. A ação foi chamada no Região da AWS us-east-1 e a solicitação foi feita em 17 de outubro de 2023 às 21:49:23 UTC.

Campos em AWS Entradas do arquivo de registro da API de vendas do Partner Central

Cada entrada em um arquivo de CloudTrail log contém informações sobre quem fez uma solicitação, os recursos utilizados na solicitação e os elementos de resposta retornados pelo AWS Partner Central. A lista de campos em uma entrada de registro, como `eventVersionuserIdentity`, `eeventTime`, fornece informações detalhadas sobre a ação. Por exemplo, o `sourceIPAddress` campo mostra o endereço IP a partir do qual a solicitação foi feita.

Registrando o AWS API de benefícios do Partner Central

AWS CloudTrailé um serviço que permite a governança, a conformidade, a auditoria operacional e a auditoria de risco do seu.Conta da AWS Com AWS CloudTrail, você pode registrar, monitorar continuamente e reter as atividades da conta relacionadas às ações em toda a sua AWS infraestrutura.AWS A atividade da API de benefícios do Partner Central é registrada como eventos em CloudTrail. Você pode criar uma trilha, uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3.

Visão geral do

A API de benefícios do AWS Partner Central está integrada com AWS CloudTrail um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS Partner Central. CloudTrail captura todas as chamadas de API para a API de benefícios do AWS Partner Central como eventos. As chamadas capturadas incluem chamadas do AWS Partner Central e de chamadas de código para as operações da API de benefícios do AWS Partner Central.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para a API de benefícios do AWS Partner Central. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação feita à API de benefícios do AWS Partner Central, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Noções básicas AWS Entradas do arquivo de log da API de benefícios do Partner Central

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3. Quando sua trilha acompanha os eventos da API de benefícios do AWS Partner Central, CloudTrail processa os eventos como arquivos de log em todas as regiões. Cada arquivo de log pode conter um ou mais eventos.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `ListBenefitApplications` ação na API de benefícios do AWS Partner Central:


```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:user/CloudTrailTestUser",
        "accountId": "123456789012",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "EX_PRINCIPAL_ID",
                "arn": "arn:aws:iam::123456789012:role/TestRole",
                "accountId": "123456789012",
                "userName": "TestRole"
            },
            "attributes": {
                "creationDate": "2025-10-21T03:08:23Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2025-10-21T03:10:59Z",
    "eventSource": "partnercentral-benefits.amazonaws.com",
    "eventName": "ListBenefitApplications",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "python-requests/2.32.4",
    "requestParameters": {
        "catalog": "AWS"
    },
```

```
    "responseElements": null,
    "requestID": "12345678-1234-5678-9abc-def012345678",
    "eventID": "87654321-4321-8765-cba9-fed098765432",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}
```

Neste exemplo, a `ListBenefitApplications` ação foi chamada pela usuária do IAM chamada Alice. A solicitação foi feita em 21 de outubro de 2025 às 03:10:59 UTC. A solicitação listou os aplicativos de benefícios para o catálogo da AWS.

Campos em AWS Entradas do arquivo de log da API de benefícios do Partner Central

Cada entrada em um arquivo de CloudTrail log contém informações sobre quem fez uma solicitação, os recursos utilizados na solicitação e os elementos de resposta retornados pela API de benefícios do AWS Partner Central. A lista de campos em uma entrada de registro, como `eventVersionuserIdentity`, `eeventTime`, fornece informações detalhadas sobre a ação. Por exemplo, o `sourceIPAddress` campo mostra o endereço IP a partir do qual a solicitação foi feita.

Registrando o AWS API do Partner Central Channel

AWS CloudTrailé um serviço que permite a governança, a conformidade, a auditoria operacional e a auditoria de risco do seu.Conta da AWS Com AWS CloudTrail, você pode registrar, monitorar continuamente e reter as atividades da conta relacionadas às ações em toda a sua AWS infraestrutura.AWS A atividade da API do Partner Central Channel é registrada como eventos em CloudTrail. Você pode criar uma trilha, uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3.

Visão geral do

A API do AWS Partner Central Channel é integrada com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS Partner Central. CloudTrail captura todas as chamadas de API para a API do AWS Partner Central Channel como eventos. As chamadas capturadas incluem chamadas do AWS Partner Central e de chamadas de código para as operações da API do AWS Partner Central Channel.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para a API do AWS Partner Central Channel. Se você não

configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação feita à API do AWS Partner Central Channel, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

Noções básicas AWS Entradas do arquivo de log da API Partner Central Channel

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3. Quando sua trilha acompanha os eventos da API do AWS Partner Central Channel, CloudTrail processa os eventos como arquivos de log em todas as regiões. Cada arquivo de log pode conter um ou mais eventos.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a `CreateProgramManagementAccount` ação na API do AWS Partner Central Channel:


```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAEXAMPLE52AGFT725JGDZ:example-user-Isengard",
        "arn": "arn:aws:sts::123456789012:assumed-role/Admin/example-user-Isengard",
        "accountId": "123456789012",
        "accessKeyId": "ASIAEXAMPLE52AGL6IXQLZ5",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROAEXAMPLE52AGFT725JGDZ",
                "arn": "arn:aws:iam::123456789012:role/ExampleRole",
                "accountId": "123456789012",
                "userName": "ExampleRole"
            },
            "attributes": {
                "creationDate": "2025-10-21T17:06:47Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2025-10-21T17:07:26Z",
    "eventSource": "partnercentral-channel.amazonaws.com",
    "eventName": "CreateProgramManagementAccount",
```

```
    "awsRegion": "us-east-1",
    "sourceIPAddress": "127.0.0.1",
    "userAgent": "PostmanRuntime/7.18.0",
    "requestParameters": {
        "catalog": "AWS",
        "program": "SOLUTION_PROVIDER",
        "displayName": "ExampleDisplayName",
        "accountId": "987654321098",
        "clientToken": "abcdef12-3456-7890-bcde-f123456789ab"
    },
    "responseElements": {
        "programManagementAccountDetail": {
            "id": "pma-example123456789",
            "arn": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/program-
management-account/pma-example123456789"
        }
    },
    "requestID": "12345678-1234-5678-9abc-def012345678",
    "eventID": "87654321-4321-8765-cba9-fed098765432",
    "readOnly": false,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS::PartnerCentralChannel::ProgramManagementAccount",
            "ARN": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/program-
management-account/pma-example123456789"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management",
    "tlsDetails": {
        "clientProvidedHostHeader": "partnercentral-channel.global.api.aws"
    }
}
```


Neste exemplo, a `CreateProgramManagementAccount` ação foi chamada pela função do IAM nomeada por ExampleRole meio de uma sessão de função assumida. A solicitação foi feita em 21 de outubro de 2025 às 17:07:26 UTC. A solicitação criou uma nova conta de gerenciamento do programa com ID `pma-example123456789` para o programa Solution Provider.

Campos em AWS Entradas do arquivo de log da API Partner Central Channel

Cada entrada em um arquivo de CloudTrail log contém informações sobre quem fez uma solicitação, os recursos utilizados na solicitação e os elementos de resposta retornados pela API do AWS Partner Central Channel. A lista de campos em uma entrada de registro, como `eventVersionuserIdentity`, `eeventTime`, fornece informações detalhadas sobre a ação. Por exemplo, o `sourceIPAddress` campo mostra o endereço IP a partir do qual a solicitação foi feita.

Registrando o AWS API de atribuição de receita do Partner Central

AWS CloudTrailé um serviço que permite a governança, a conformidade, a auditoria operacional e a auditoria de risco do seu.Conta da AWS Com AWS CloudTrail, você pode registrar, monitorar continuamente e reter as atividades da conta relacionadas às ações em toda a sua AWS infraestrutura.AWS A atividade da API de medição de receita do Partner Central é registrada como eventos em CloudTrail. Você pode criar uma trilha, uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3.

Visão geral do

A API de Medição de Receita do AWS Partner Central está integrada com AWS CloudTrail um serviço que fornece um registro das ações realizadas por um usuário, função ou AWS serviço no AWS Partner Central. CloudTrail captura todas as chamadas de API para o AWS Partner Central Revenue Attribution como eventos. As chamadas capturadas incluem chamadas do console do AWS Partner Central e de chamadas de código para as operações da API de Medição de Receita do AWS Partner Central.

Se você criar uma trilha, poderá habilitar a entrega contínua de CloudTrail eventos para um bucket do Amazon S3, incluindo eventos para o AWS Partner Central Revenue Measurement. Se você não configurar uma trilha, ainda poderá ver os eventos mais recentes no CloudTrail console no Histórico de eventos.

Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita àAWS Partner Central Revenue Measurement, o endereço IP do qual a solicitação foi feita, quem fez a solicitação, quando ela foi feita e detalhes adicionais.

As seguintes ações da API de Medição de Receita do AWS Partner Central estão registradas: CloudTrail

- `CreateRevenueAttribution`

- UpdateRevenueAttribution
- GetRevenueAttribution
- ListRevenueAttributions
- TagResource
- UntagResource
- ListTagsForResource
- CreateMarketplaceRevenueShare
- GetMarketplaceRevenueShare
- ListMarketplaceRevenueShares
- CreateMarketplaceRevenueShareAllocation
- GetMarketplaceRevenueShareAllocation
- UpdateMarketplaceRevenueShareAllocation
- ListMarketplaceRevenueShareAllocations
- StartRevenueAttributionAllocationsTask
- GetRevenueAttributionAllocationsTask
- ListRevenueAttributionAllocations
- GetRevenueAttributionAllocation

Noções básicas AWS Entradas do arquivo de registro da API de medição de receita do Partner Central

Uma trilha é uma configuração que permite a entrega de eventos como arquivos de log para um bucket do Amazon S3. Quando sua trilha acompanha os eventos de medição de receita do AWS Partner Central, CloudTrail processa os eventos como arquivos de log em todas as regiões. Cada arquivo de log pode conter um ou mais eventos.

O exemplo a seguir mostra uma entrada de CloudTrail registro que demonstra a CreateRevenueAttribution ação na Atribuição de Receita do AWS Partner Central:

```
{
    "eventVersion": "1.11",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAEXAMPLEID",
```

```
        "arn": "arn:aws:iam::123456789012:user/CloudTrailTestUser",
        "accountId": "123456789012",
        "accessKeyId": "AKIAEXAMPLEKEY",
        "userName": "CloudTrailTestUser"
    },
    "eventTime": "2026-06-02T11:53:54Z",
    "eventSource": "partnercentral-prm.amazonaws.com",
    "eventName": "CreateRevenueAttribution",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.1",
    "userAgent": "aws-sdk-java/2.20.0",
    "requestParameters": {
        "catalog": "AWS",
        "name": "my-revenue-attribution",
        "marketplaceProduct": {
            "tenancyModel": "MULTI_TENANT"
        }
    },
    "responseElements": {
        "id": "ra-0123456789abcdef0",
        "arn": "arn:aws:partnercentral:us-east-1:123456789012:catalog/AWS/revenue-
attribution/ra-0123456789abcdef0",
        "name": "my-revenue-attribution",
        "revision": "1"
    },
    "requestID": "5ce2f907-3850-412a-b1a4-r012r1234567",
    "eventID": "80b39b72-eefe-4578-8db0-01ee2e34ee56",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```


Neste exemplo, a `CreateRevenueAttribution` ação foi chamada pelo usuário do IAM chamado CloudTrailTestUser. A ação foi chamada em Região da AWS us-east-1 e a solicitação foi feita em 2 de junho de 2026 às 11:53:54 UTC. Um novo recurso de atribuição de receita foi criado com o ID. `ra-0123456789abcdef0`

Campos em AWS Entradas do arquivo de registro da API de medição de receita do Partner Central

Cada entrada em um arquivo de CloudTrail registro contém informações sobre quem fez uma solicitação, os recursos utilizados na solicitação e os elementos de resposta retornados pela AWS

Partner Central Revenue Measurement. A lista de campos em uma entrada de registro, como `eventVersionuserIdentity`, `eeventTime`, fornece informações detalhadas sobre a ação. Por exemplo, o `sourceIPAddress` campo mostra o endereço IP a partir do qual a solicitação foi feita.

Notificações para o AWS API do Partner Central

As seções a seguir fornecem informações sobre notificações.

Tópicos

- Notificações para o AWS API de conta do Partner Central
- Notificações para o AWS API de vendas da Central de Parceiros
- Notificações para o AWS API de benefícios do Partner Central
- Notificações para o AWS API do Partner Central Channel

Notificações para o AWS API de conta do Partner Central

As notificações de conexão da conta do parceiro permitem que os parceiros se mantenham informados sobre os eventos do ciclo de vida da conexão que afetam diretamente seus relacionamentos comerciais e fluxos de trabalho operacionais. Esses eventos são essenciais para que os parceiros:

- Reaja prontamente à nova colaboração
- Mantenha o conhecimento do status do portfólio de conexões
- Tome as medidas apropriadas quando as conexões forem estabelecidas ou encerradas
- Garanta a continuidade dos negócios sabendo quando os relacionamentos mudam

Tópicos

- Preencha o pré-requisito para monitorar eventos
- Configure EventBridge a Amazon para monitorar eventos
- Saiba mais sobre eventos da API de conexões de contas

Preencha o pré-requisito para monitorar eventos

Os usuários precisam das permissões apropriadas do IAM para acessar e gerenciar eventos publicados pela API de conexões da conta. Para obter mais informações sobre as ações, os recursos

e as chaves de condição disponíveis para EventBridge, consulte Usando as condições da política do IAM na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Uma das chaves de condição é`events:detail-type`, que pode ser usada para definir permissões para tipos específicos de eventos.

O exemplo de política a seguir demonstra como personalizar e definir o escopo das permissões para os eventos propostos. A `AllowPutRuleForPartnercentralAccountEvents` declaração permite a criação de regras, mas somente para eventos da `aws.partnercentral-account` fonte.

Para exemplos detalhados de políticas do IAM, consulte a documentação da AWS sobre EventBridge permissões.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnerCentralAccountEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "events:source": "aws.partnercentral-account.connection"
        }
      }
    }
  ]
}
```

Configure EventBridge a Amazon para monitorar eventos

Para monitorar os eventos da API de conexões da conta, você cria uma EventBridge regra que corresponde aos eventos que você deseja capturar. Você pode usar o AWS Management Console ou os AWS SDKs para criar e gerenciar regras. As seções a seguir explicam como criar regras usando os dois métodos. Independentemente do método usado, você deve criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

AWS Management Console configuração

Para configurar uma EventBridge regra usando o AWS Management Console, siga as etapas em Criação de regras que reagem a eventos na Amazon EventBridge no Guia EventBridge do usuário

da Amazon. Ao criar a regra, você deve definir o barramento de eventos como padrão e criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

Veja a seguir um exemplo de uma regra de evento:

```
{
    "source": ["aws.partnercentral-account"],
    "detail": {
        "catalog": ["AWS"]
    }
}
```

AWS Configuração do SDK

Você pode usar os AWS SDKs para criar e gerenciar EventBridge regras de forma programática. Para obter mais informações, consulte PutRulea Amazon EventBridge API Reference.

O exemplo a seguir usa o AWS SDK para Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='MyConnectionInvitationReceivedRule',
    EventPattern=
    '{
        "source": ["aws.partnercentral-account"],
        "detail-type": ["Partner Connection Invitation Received"],
        "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Saiba mais sobre eventos da API de conexões de contas

As seções a seguir descrevem os tipos de eventos da API de conexões de conta, os cenários que os acionam e os exemplos de eventos.

Tipos de eventos

A seguir estão os tipos de eventos e seus gatilhos.

- Convite de conexão de parceiro recebido: notifica o receptor de que outro parceiro deseja estabelecer uma conexão
- Convite de conexão de parceiro aceito: notifica o remetente de que o convite foi aceito e que a conexão agora está ativa
- Convite de conexão de parceiro rejeitado: notifica o remetente de que o convite foi recusado
- Conexão de parceiro cancelada: notifica o outro participante conectado quando uma conexão ativa é encerrada
- Convite de conexão de parceiro cancelado: notifica o destinatário de que o remetente retirou o convite antes que qualquer ação fosse tomada
- Convite de conexão de parceiro expirado: notifica o remetente e o destinatário de que o convite expirou

Eventos de exemplo

As seções a seguir fornecem exemplos dos eventos listados anteriormente na seção anterior.

Tópicos

- Convite de conexão de parceiro recebido
- Convite de conexão de parceiro aceito
- Convite de conexão de parceiro rejeitado
- Conexão de parceiro cancelada
- Convite de conexão de parceiro cancelado
- O convite de conexão de parceiro expirou

Convite de conexão de parceiro recebido

Contexto de acionamento

Esse evento é gerado quando um convite de conexão é criado e mantido com sucesso no armazenamento de dados do PAC por meio da CreateConnectionInvitation API. O evento é enviado imediatamente após a conclusão bem-sucedida da criação do convite e persiste no armazenamento de dados do PAC, não apenas quando a chamada da API é feita.

O evento será enviado automaticamente quando um novo convite de conexão for criado. Um evento por convite criado sem eventos duplicados para o mesmo convite.

Destinatários

A AWS conta do destinatário no convite de conexão.

Tratamento esperado

Ações típicas do cliente:

- Notificação imediata: acione alertas para as equipes de negócios sobre novas oportunidades de parceria
- Automação do fluxo de trabalho: atualize automaticamente os sistemas de gerenciamento de parceiros com convites pendentes
- Apoio à decisão: encaminhe convites para os tomadores de decisão apropriados com base no tipo de conexão
- Registro de auditoria: registre o recebimento do convite para acompanhamento de conformidade e parceria
- Automação de respostas: para parceiros confiáveis, é possível aceitar automaticamente determinados tipos de convite

Padrões comuns de integração:

- Função Lambda → Atualizar o painel do portal do parceiro
- Fila SQS → Processar convites em lote para análise
- Tópico do SNS → Email/Slack notificações para equipes de negócios
- Destino da API → Webhook para sistemas CRM/partner de gerenciamento externos

Exemplo

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Received",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "account": "<corresponding partner AWS account>",
  "resources": [ "<<Connection Invitation ARN>>" ],
```

```
  "detail": {
    "catalog": "AWS",
    "connectionInvitation" :{
        "arn": "<<Connection Invitation ARN>>",
        "id": "pacinv-****",
        "connectionType": "OpportunityCollaboration",
        "inviterEmail": "abc@def.com",
        "invitationMessage": "We'd like to collaborate on a joint solution for cloud
 security. Please accept this connection invitation to proceed.",
        "senderCompanyName": "<<Sender Partner Account Name>>",
        "senderProfileId": "pprofile-****",
        "expiresAt": "<ISO 8601 date time>"
    }
  }
}
```

Convite de conexão de parceiro aceito

Contexto de acionamento

Esse evento é gerado quando um convite de conexão é aceito com sucesso e uma conexão é criada e mantida no armazenamento de dados do PAC por meio da AcceptConnectionInvitation API. O evento é enviado imediatamente após a conclusão bem-sucedida da aceitação do convite e o estabelecimento da conexão.

O evento será enviado automaticamente quando um convite de conexão for aceito. Aceito um evento por convite, sem eventos duplicados para a mesma aceitação.

Destinatários

Ambas as AWS contas envolvidas no convite de conexão: a conta do remetente que originalmente enviou o convite e a conta do destinatário que aceitou o convite.

Tratamento esperado

Ações típicas do cliente:

- Ativação de parceria: acione fluxos de trabalho para permitir atividades comerciais de camada 2
- Atualizações de status: atualize os sistemas de gerenciamento de parceiros com o status da conexão ativa
- Notificação: Alerte as equipes de negócios sobre o estabelecimento bem-sucedido da parceria

- Provisionamento de acesso: conceda automaticamente as permissões apropriadas para colaboração
- Analytics: acompanhe as taxas de conversão de parcerias e as métricas de sucesso

Padrões comuns de integração:

- Função Lambda → Ativar permissões de compartilhamento de dados e atualizar o portal do parceiro
- Fila SQS → Batch processa ativações de conexão para configuração de negócios
- Tópico do SNS → Email/Slack notificações para equipes técnicas e comerciais
- Destino da API → Webhook para sistemas de CRM para atualizar o status da parceria

Exemplo

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type":  "Partner Connection Invitation Accepted",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "resources": [ "<<Connection Invitation ARN>>" , "<<Connection ARN>>" ],
  "account": "<corresponding partner AWS account>",
  "detail": {
    "catalog": "AWS",
    "connectionInvitation" :{
        "arn": "<<Connection Invitation ARN>>",
        "id": "pacinv-****",
        "connectionType": "OpportunityCollaboration"
    },
    "connection": {
        "arn": "<<Connection ARN>>",
         "id": "pac-*****",
         "Participant1AccountId": "<<Sender AWS AccountId>>",
         "Participant2AccountId": "<<Receiver AWS AccountId>>",
         "status": "ACTIVE"
     }
  }
}
```

Convite de conexão de parceiro rejeitado

Contexto de acionamento

Esse evento é gerado quando um convite de conexão é rejeitado com sucesso por meio da RejectConnectionInvitation API. O evento é enviado imediatamente após a rejeição do convite ser processada e persistida no armazenamento de dados do PAC.

O evento será enviado automaticamente quando um convite de conexão for rejeitado. Um evento por convite rejeitado sem eventos duplicados para a mesma rejeição.

Destinatários

A AWS conta que originalmente enviou o convite de conexão (a conta do remetente).

Tratamento esperado

Ações típicas do cliente:

- Atualizações de status: atualize os sistemas de gerenciamento de parceiros com o status de rejeição
- Follow-up Ações: acionar fluxos de trabalho para abordagens alternativas de parceria
- Notificação: alerte as equipes de negócios sobre a decisão de parceria
- Limpeza: remova referências de convites pendentes dos sistemas internos

Padrões comuns de integração:

- Função Lambda → Atualize o portal do parceiro e acione fluxos de trabalho de acompanhamento
- Fila SQS → Rejeições de processos em lote para análise de negócios
- Tópico do SNS → Email/Slack notificações para equipes de desenvolvimento de negócios
- Destino da API → Webhook para sistemas de CRM para atualizar o status da oportunidade

Exemplo

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Rejected",
  "source": "aws.partnercentral-account",
```

```
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "account": "<corresponding partner AWS account>",
  "resources": [ "<<Connection Invitation ARN>>" ],
  "detail": {
    "catalog": "AWS",
    "connectionInvitation" :{
        "arn": "<<Connection Invitation ARN>>",
        "id": "pacinv-****",
        "connectionType": "OpportunityCollaboration",
        "invitationMessage": "We'd like to collaborate on a joint solution for cloud
 security. Please accept this connection invitation to proceed.",
        "receiverProfileId": "pprofile-****"
    }
  }
}
```

Conexão de parceiro cancelada

Contexto de acionamento

Esse evento é gerado quando uma conexão ativa é cancelada com sucesso e o status ativo é atualizado para encerrado no armazenamento de dados do PAC por meio da API. CancelConnection O evento é enviado imediatamente após o cancelamento da conexão ser processado e o status da conexão ser atualizado.

O evento será enviado automaticamente quando a conexão for encerrada. Um evento por conexão cancelado sem eventos duplicados para o mesmo cancelamento.

Destinatários

A AWS conta do outro participante na conexão (não daquele que iniciou o cancelamento).

Tratamento esperado

Ações típicas do cliente:

- Revogação de acesso: revogue imediatamente as permissões e desative o compartilhamento de dados
- Atualizações de status: atualize os sistemas de gerenciamento de parceiros com o status da conexão encerrada
- Notificação: Alerte as equipes técnicas e comerciais sobre o término da parceria

- Limpeza: remova as referências de conexão e limpe os recursos compartilhados
- Análise: acompanhe a duração da conexão e os padrões de encerramento

Padrões comuns de integração:

- Função Lambda → Revogar permissões e atualizar o status do portal do parceiro
- Fila SQS → Terminações de conexão de processamento em lote para fluxos de trabalho de limpeza
- Tópico do SNS → Email/Slack notificações para equipes técnicas e comerciais
- Destino da API → Webhook para sistemas de CRM para atualizar o status da parceria

Exemplo

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Cancelled",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "resources": [ "<<Connection ARN>>" ],
  "account": "<corresponding partner AWS account>",
  "detail": {
    "catalog": "AWS",
    "connection" :{
        "arn": "<<Connection Invitation ARN>>",
        "id": "pac-****",
        "connectionType": "OpportunityCollaboration",
        "canceledBy": "<<AWS account ID>>"
    }
  }
}
```

Convite de conexão de parceiro cancelado

Contexto de acionamento

Esse evento é gerado quando um convite de conexão pendente é cancelado com sucesso por meio da CancelConnectionInvitation API. O evento é enviado imediatamente após o cancelamento do convite ser processado e o status do convite ser atualizado para CANCELADO.

O evento será enviado automaticamente quando um convite de conexão for cancelado. Um evento por convite de conexão foi cancelado sem eventos duplicados para o mesmo cancelamento.

Destinatários

A AWS conta que deveria receber o convite de conexão (a conta do destinatário).

Tratamento esperado

Ações típicas do cliente:

- Atualizações de status: atualize os sistemas de gerenciamento de parceiros para remover referências de convites pendentes
- Notificação: alerte as equipes de negócios de que uma possível oportunidade de parceria foi retirada
- Limpeza: remova as referências de convite dos sistemas internos de rastreamento
- Analytics: acompanhe os padrões de cancelamento de convites para obter informações sobre parcerias

Padrões comuns de integração:

- Função Lambda → Atualizar o portal do parceiro e remover notificações de convites pendentes
- Fila SQS → Cancelamentos de processos em lote para análise de negócios
- Tópico do SNS → Email/Slack notificações para equipes de desenvolvimento de negócios
- Destino da API → Webhook para sistemas de CRM para atualizar o status da oportunidade

Exemplo

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Cancelled",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "account": "<corresponding partner AWS account>",
  "resources": [ "<<Connection Invitation ARN>>" ],
  "detail": {
    "catalog": "AWS",
```

```
    "connectionInvitation" :{
        "arn": "<<Connection Invitation ARN>>",
        "id": "pacinv-****",
        "connectionType": "OpportunityCollaboration",
        "invitationMessage": "We'd like to collaborate on a joint solution for cloud
 security. Please accept this connection invitation to proceed.",
        "senderProfileId": "pprofile-****"
    }
  }
}
```

O convite de conexão de parceiro expirou

Contexto de acionamento

Esse evento é gerado quando um convite de conexão pendente expira automaticamente após atingir seu registro de ExpiresAt data e hora sem ser aceito ou rejeitado pelo destinatário. O evento é acionado automaticamente pelo sistema quando o prazo de expiração do convite é atingido.

O evento será enviado automaticamente quando o convite de conexão expirar. Um evento por convite expirou sem eventos duplicados com a mesma expiração.

Destinatários

Ambas as AWS contas envolvidas no convite de conexão: a conta do remetente que originalmente enviou o convite e a conta do destinatário que recebeu o convite.

Tratamento esperado

Ações típicas do cliente:

- Atualizações de status: atualize os sistemas de gerenciamento de parceiros para marcar o convite como expirado
- Follow-up Ações: acionar fluxos de trabalho para reengajamento ou abordagens alternativas de parceria
- Notificação: alerte as equipes de negócios sobre oportunidades de parceria perdidas
- Limpeza: remova referências de convite expiradas dos sistemas internos de rastreamento

Padrões comuns de integração:

- Função Lambda → Atualize o portal do parceiro e acione fluxos de trabalho de acompanhamento

- Fila SQS → Expirações de processos em lote para análise de negócios
- Tópico do SNS → Email/Slack notificações para equipes de desenvolvimento de negócios
- Destino da API → Webhook para sistemas de CRM para atualizar o status da oportunidade

Exemplo

```
{
  "version": "1",
  "id": "<event id>",
  "detail-type": "Partner Connection Invitation Expired",
  "source": "aws.partnercentral-account",
  "time": "<ISO 8601 date time>",
  "region": "us-east-1",
  "resources": [ "<<Connection Invitation ARN>>" ],
  "detail": {
    "catalog": "AWS",
    "connectionInvitation": {
        "arn": "<<Connection Invitation ARN>>",
        "id": "pacinv-****",
        "connectionType": "OpportunityCollaboration",
        "inviterEmail": "abc@def.com",
        "invitationMessage": "We'd like to collaborate on a joint solution for cloud
 security. Please accept this connection invitation to proceed.",
        "senderProfileId": "pprofile-****",
        "receiverProfileId": "pprofile-****"
    }
  }
}
```

Notificações para o AWS API de vendas da Central de Parceiros

Os eventos da API de vendas fornecem notificações em tempo real sobre mudanças no status ou nos detalhes das oportunidades. Esses eventos ajudam a manter seus sistemas sincronizados com o AWS Partner Central e ajudam a garantir respostas e atualizações oportunas.

Tópicos

- Preencha o pré-requisito para monitorar eventos
- Configure EventBridge a Amazon para monitorar eventos
- Saiba mais sobre a venda de eventos de API

Preencha o pré-requisito para monitorar eventos

Os usuários precisam das permissões apropriadas do IAM para acessar e gerenciar eventos publicados pela API de vendas do Partner Central. Para obter mais informações sobre as ações, os recursos e as chaves de condição disponíveis para EventBridge, consulte Usando as condições da política do IAM na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Uma das chaves de condição é`events:detail-type`, que pode ser usada para definir permissões para tipos específicos de eventos.

O exemplo de política a seguir demonstra como personalizar e definir o escopo das permissões para os eventos propostos. A `AllowPutRuleForPartnercentralSellingEvents` declaração permite a criação de regras, mas somente para eventos da `aws.partnercentral-selling` fonte.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": {
    "Sid": "AllowPutRuleForPartnercentralSellingEvents",
    "Effect": "Allow",
    "Action": "events:PutRule",
    "Resource": "*",
    "Condition": {
      "ForAllValues:StringEquals": {
        "events:source": "aws.partnercentral-selling"
      }
    }
  }
}
```

Configure EventBridge a Amazon para monitorar eventos

Para monitorar a venda de eventos de API, você cria uma EventBridge regra que corresponde aos eventos que você deseja capturar. Você pode usar o AWS Management Console ou os AWS SDKs para criar e gerenciar regras. As seções a seguir explicam como criar regras usando os dois métodos. Independentemente do método usado, você deve criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

AWS Management Console configuração

Para configurar uma EventBridge regra usando o AWS Management Console, siga as etapas em Criação de regras que reagem a eventos na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Ao criar a regra, você deve definir o barramento de eventos como padrão e criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

Veja a seguir um exemplo de uma regra de evento:

```
{
    "source": ["aws.partnercentral-selling"],
    "detail": {
        "catalog": ["AWS"]
    }
}
```

AWS Configuração do SDK

Você pode usar os AWS SDKs para criar e gerenciar EventBridge regras de forma programática. Para obter mais informações, consulte PutRulea Amazon EventBridge API Reference.

O exemplo a seguir usa o AWS SDK para Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='MyOpportunityCreatedRule',
    EventPattern=
    '{
        "source": ["aws.partnercentral-selling"],
        "detail-type": ["Opportunity Created"],
        "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Saiba mais sobre a venda de eventos de API

As seções a seguir descrevem os tipos de eventos da API de venda, os cenários que os acionam e os exemplos de eventos.

Tipos de eventos

A seguir estão os tipos de eventos e seus gatilhos.

- Oportunidade criada: acionada quando uma nova oportunidade é criada.
- Oportunidade atualizada: acionada quando uma oportunidade (oportunidade ou seu resumo de AWS oportunidade correspondente) é atualizada.
- Convite de engajamento criado: acionado quando um convite de AWS referência é criado.
- Convite de engajamento aceito: acionado quando um parceiro aceita um convite de AWS engajamento, confirmando seu interesse em colaborar com a AWS oportunidade.
- Convite de noivado rejeitado: acionado quando um parceiro rejeita um convite de AWS noivado.
- O convite de engajamento expirou: Este evento é acionado quando um parceiro rejeita um. EngagementInvitation Ele notifica o parceiro de envio e recebimento de que o convite expirou.
- Membro de engajamento adicionado: esse evento é acionado quando um novo membro se junta ao Engajamento após aceitar um convite. Ele notifica todos os membros atuais do Engajamento sobre o novo membro que está sendo adicionado ao Engajamento.
- Criação de instantâneo do recurso de engajamento: esse evento é acionado quando a nova revisão do instantâneo dos recursos (como oportunidades) associados a um compromisso é criada. Ele notifica todos os membros atuais do Engajamento sobre as alterações no recurso Snapshot associado.
- Engajamento criado: acionado quando um novo engajamento é criado.
- Engajamento atualizado: acionado quando um engajamento é atualizado.

Cenários de eventos

A tabela a seguir descreve como os eventos operam em diferentes cenários. As seguintes suposições são feitas para esses cenários:

- AWS também é parceira nesses cenários.
- `ResourceSnapshotJob`está configurado corretamente pelo parceiro.

- Os campos atualizados sobre a oportunidade também estão no modelo de oportunidade do Resource Snapshot.

Ação de você como parceiro	Eventos recebidos	Tipo de participante
Você cria uma oportunidade	Oportunidade criada	
Você usa `StartEngagementFromOpportunityTask` para enviar uma oportunidade	Oportunidade atualizada, engajamento criado, resumo do recurso de engajamento criado, membro do engajamento adicionado, convite de engajamento criado	
AWS aprova sua oportunidade enviada	Convite de engajamento aceito, membro de engajamento adicionado, oportunidade atualizada, resumo do recurso de engajamento criado	
AWS rejeita sua oportunidade enviada	Convite de engajamento rejeitado, oportunidade atualizada, resumo do recurso de engajamento criado	
Você associa uma AWS Marketplace oferta a uma oportunidade	Oportunidade atualizada	
Você associa uma solução a uma oportunidade	Oportunidade atualizada, resumo dos recursos de engajamento criado	
Você atualiza uma oportunidade	Oportunidade atualizada, instantâneo do recurso de engajamento criado (opcional se `ResourceSnapshotJob` estiver configurado e	

Ação de você como parceiro	Eventos recebidos	Tipo de participante
	as atualizações de campo estiverem no modelo)	
AWS atualiza sua oportunidade	Oportunidade atualizada, resumo dos recursos de engajamento criado	
Você convida outro parceiro para um compromisso	Convite de engajamento criado	Remetente
Seu convite para participar de um compromisso é aceito por um parceiro	Convite de noivado aceito, membro de engajamento adicionado	Remetente
Seu convite para participar de um compromisso é rejeitado por um parceiro	Convite de compromisso rejeitado	Remetente
Seu convite para participar de um compromisso não é atendido por um parceiro por 15 dias	O convite de noivado expirou	Remetente
Você recebe um convite de outro parceiro para participar de um compromisso	Convite de engajamento criado	Receptor
Você aceita o convite de outro parceiro para participar de um compromisso via `StartEngagementByAcceptingInvitationTask`	Convite de engajamento aceito, membro de engajamento adicionado, oportunidade criada, oportunidade atualizada, instantâneo do recurso de engajamento criado	Receptor
Você rejeita o convite de outro parceiro para participar de um compromisso	Convite de compromisso rejeitado	Receptor

Ação de você como parceiro	Eventos recebidos	Tipo de participante
Você não age sob convite para participar de um compromisso de outro parceiro por 15 dias	O convite de noivado expirou	Receptor
Você cria um compromisso	Engajamento criado	Remetente
Você atualiza um compromisso	Engajamento atualizado	Remetente, destinatário

Eventos de exemplo

As seções a seguir fornecem exemplos dos eventos listados anteriormente na seção anterior.

Tópicos

- Oportunidade criada
- Oportunidade atualizada
- Convite de engajamento criado
- Convite de compromisso aceito
- Convite de compromisso rejeitado
- O convite de noivado expirou
- Membro de engajamento adicionado
- Instantâneo do recurso de engajamento criado
- Engajamento criado
- Engajamento atualizado

Oportunidade criada

Os parceiros usam esse evento para:

- Notifique seus usuários de que uma nova oportunidade foi criada.
- Acione fluxos de trabalho automatizados, como atualizar os sistemas de CRM ou notificar as equipes de vendas sobre a criação de novas oportunidades.

O exemplo a seguir mostra um evento típico de Oportunidade Criada.

```
{
    "version": "1",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "source": "aws.partnercentral-selling",
    "detail-type": "Opportunity Created",
    "time": "2023-04-16T15:23:45Z",
    "region": "us-east-1",
    "account": "123456789012",
    "detail": {
        "schemaVersion": "<version number>",
        "catalog": "<Sandbox | AWS>",
        "opportunity": {
            "identifier": "String"
        }
    }
}
```

Oportunidade atualizada

Os parceiros usam esse evento para:

- Notifique seus usuários de que uma oportunidade existente foi atualizada.
- Acione fluxos de trabalho automatizados, como atualizar sistemas de CRM ou notificar equipes de vendas.

O exemplo a seguir mostra um evento típico de atualização de oportunidade.

```
{
    "version": "1",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "source": "aws.partnercentral-selling",
    "detail-type": "Opportunity Updated",
    "time": "2023-04-16T15:23:45Z",
    "region": "us-east-1",
    "account": "123456789012",
    "detail": {
        "schemaVersion": "<version number>",
        "catalog": "<Sandbox | AWS>",
        "opportunity": {
```

```
            "identifier": "String"
        }
    }
}
```

Convite de engajamento criado

Os parceiros usam esse evento para:

- Notifique seus usuários sobre as novidades que EngagementInvitation eles receberam.
- Acione fluxos de trabalho automatizados para aceitar ou rejeitar o convite, como atualizar sistemas de CRM ou notificar equipes de vendas.

O exemplo a seguir mostra um evento típico criado por convite de engajamento.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Invitation Created",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-15T12:34:56Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-
v7p8z56whnauo"
    ],
    "detail": {
      "catalog": "AWS",
      "engagementInvitation": {
        "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/
engi-v7p8z56whnauo",
        "id": "engi-v7p8z56whnauo",
        "engagementId": "eng-12345678901234",
        "senderAccountId": "string",
        "receiverAccountId": "string",
        "senderCompanyName": "string",
        "expirationDate": "string",
        "participantType": "Enum", //Sender/Receiver
  "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
      }
    }
```

```
}
```

Convite de compromisso aceito

Os parceiros usam esse evento para:

- Notifique seus usuários de que o EngagementInvitation foi aceito.
- Atualize seus registros internos para refletir o novo parceiro no Engajamento.
- Acione fluxos de trabalho automatizados para sincronizar dados ou notificar outros membros da equipe sobre o novo membro do Engagement.

O exemplo a seguir mostra um evento típico aceito por convite de noivado.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Invitation Accepted",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-16T15:23:45Z",
    "region": "us-east-1",
    "resources": ["arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/
engi-v7p8z56whnauo"],
    "detail": {
        "catalog": "AWS",
        "engagementInvitation": {
            "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-
invitation/engi-v7p8z56whnauo",
            "id": "engi-v7p8z56whnauo",
            "engagementId": "eng-12356whnauo",
            "participantType": "Enum", //Sender/Receiver
            "senderAccountId": "string",
            "receiverAccountId": "string",
            "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
        }
    }
}
```

Convite de compromisso rejeitado

Os parceiros usam esse evento para:

- Notifique seus usuários de que o EngagementInvitation foi rejeitado.
- Atualize seus registros internos para refletir o convite rejeitado.
- Acione todos os fluxos de trabalho necessários, como notificar a equipe de vendas sobre a rejeição.

O exemplo a seguir mostra um evento típico de rejeição de convite de engajamento.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Invitation Rejected",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-17T09:48:27Z",
    "region": "us-east-1",
    "resources": ["arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/
engi-v7p8z56whnauo"],
    "detail": {
        "catalog": "AWS",
        "engagementInvitation": {
            "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-
invitation/engi-v7p8z56whnauo",
            "id": "engi-v7p8z56whnauo",
            "engagementId": "eng-12356whnauo",
            "participantType": "Enum", //Sender/Receiver
            "senderAccountId": "string",
            "receiverAccountId": "string",
            "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
        }
    }
}
```

O convite de noivado expirou

Os parceiros usam esse evento para:

- Notifique seus usuários de que o EngagementInvitation expirou e não pode mais ser usado.
- Atualize seus registros internos para refletir o convite expirado.
- Acione todos os fluxos de trabalho necessários, como notificar a equipe de vendas sobre o convite expirado.

O exemplo a seguir mostra um evento típico de convite de engajamento expirado.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Invitation Expired",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-18T18:20:15Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/engi-
v7p8z56whnauo"
    ],
    "detail": {
      "catalog" : "AWS",
      "engagementInvitation": {
        "arn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement-invitation/
engi-v7p8z56whnauo",
        "id": "engi-v7p8z56whnauo",
        "engagementId": "eng-12356whnauo",
        "participantType": "Enum", //Sender/Receiver
        "senderAccountId": "string",
        "receiverAccountId": "string",
        "payloadType": "LeadInvitation" //OpportunityInvitation|LeadInvitation
      }
    }
}
```

Membro de engajamento adicionado

Os parceiros usam esse evento para:

- Notifique seus usuários sobre as mudanças na associação do Engagement.
- Atualize seus registros internos para refletir o novo membro do Engagement.
- Acione todos os fluxos de trabalho necessários, como notificar os membros da equipe sobre as mudanças.

O exemplo a seguir mostra um evento típico de Engagement Member Added.

```
{
```

```
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Member Added",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-16T15:23:45Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-v7p8z56whnauo"
    ],
    "detail": {
      "catalog" : "AWS",
      "engagement": {
        "id": "eng-v7p8z56whnauo"
      },
      "engagementMember": {
        "accountId": "string",
        "companyName": "string"
      }
    }
}
```

Instantâneo do recurso de engajamento criado

Os parceiros usam esse evento para monitorar alterações nos recursos associados a um compromisso e acionar fluxos de trabalho automatizados, como atualizar sistemas de CRM ou notificar equipes de vendas. O modelo de snapshot determina o tipo de atualização:

`OpportunitySummaryView`

- Notifique todos os membros de um compromisso de que o compromisso foi atualizado.

`AwsOpportunitySummaryFullView`

- Acompanhe as atualizações feitas especificamente pela AWS para oportunidades dentro de um compromisso.
- Essa notificação e esse instantâneo são visíveis somente para o proprietário da oportunidade.
- Inclui atualizações de dimensionamento de negócios, que não estão disponíveis por meio do `OpportunitySummaryView` modelo.

O exemplo a seguir mostra um evento típico do Engagement Resource Snapshot Created.

```
{
```

```
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Resource Snapshot Created",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-18T18:20:15Z",
    "region": "us-east-1",
    "resources": [
      "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-v7p8z56whnauo"
    ],
    "detail": {
      "catalog" : "AWS",
      "resourceSnapshot": {
        "arn": "arn:aws:partnercentral-selling:us-east-1::catalog/AWS/engagement/eng-
v7p8z56whnauo/resource/Opportunity/o-12312/template/temp-name/snapshot/snapshot-19232",
        "engagementId": "eng-v7p8z56whnauo",
        "resourceType": "Opportunity",
        "resourceId" : "O123211231",
        "createdBy": "123123123123",
        "targetMemberAccounts": ["123123123123", "aws"],
        "resourceSnapshotTemplateName": "temp-name"
      }
    }
}
```

Engajamento criado

Os parceiros usam esse evento para:

- Notifique seus usuários de que um novo Engajamento foi criado.
- Acione fluxos de trabalho automatizados, como atualizar os sistemas de CRM ou notificar as equipes de vendas sobre novas criações de engajamento.

O exemplo a seguir mostra um evento típico de Engagement Created.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Created",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-18T18:20:15Z",
```

```
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-567a1j5hxp35lo"
    ],
    "detail": {
        "catalog": "AWS",
        "engagement": {
            "engagementArn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/
eng-567a1j5hxp35lo",
            "engagementId": "eng-567a1j5hxp35lo",
            "CreatedAt": "2023-04-18T18:20:15Z",
            "CreatedBy": "aws",
            "ContextTypes": ["Lead"]

        }
    }
}
```

Engajamento atualizado

Os parceiros usam esse evento para:

- Notifique seus usuários de que um Engajamento existente foi atualizado.
- Acione fluxos de trabalho automatizados, como atualizar sistemas de CRM ou notificar equipes de vendas.

O exemplo a seguir mostra um evento típico de Engagement Updated.

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Engagement Updated",
    "source": "aws.partnercentral-selling",
    "account": "123456789012",
    "time": "2023-04-18T18:20:15Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/eng-567a1j5hxp35lo"
    ],
    "detail": {
        "catalog": "AWS",
        "engagement": {
```

```
            "engagementArn": "arn:aws:partnercentral:us-east-1::catalog/AWS/engagement/
eng-567a1j5hxp35lo",
            "engagementId": "eng-567a1j5hxp35lo",
            "LastModifiedAt": "2023-04-18T18:20:15Z",
            "LastModifiedBy": "aws",
            "ContextTypes": ["Lead", "CustomerProject"]
        }
    }
}
```

Notificações para o AWS API de benefícios do Partner Central

O Serviço de Benefícios oferece EventBridge eventos para BenefitApplications BenefitAllocations e. Esses eventos permitem que os clientes criem arquiteturas abrangentes orientadas por eventos que respondem às mudanças em todo o ciclo de vida dos recursos, desde a aplicação inicial até a alocação e o atendimento final.

Tópicos

- Preencha o pré-requisito para monitorar eventos
- Configure EventBridge a Amazon para monitorar eventos
- Saiba mais sobre os eventos da API de benefícios

Preencha o pré-requisito para monitorar eventos

Os usuários precisam das permissões apropriadas do IAM para acessar e gerenciar eventos publicados pela API de benefícios. Para obter mais informações sobre as ações, os recursos e as chaves de condição disponíveis para EventBridge, consulte Usando as condições da política do IAM na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Uma das chaves de condição é`events:detail-type`, que pode ser usada para definir permissões para tipos específicos de eventos.

O exemplo de política a seguir demonstra como personalizar e definir o escopo das permissões para os eventos propostos. A `AllowPutRuleForPartnercentralBenefitsEvents` declaração permite a criação de regras, mas somente para eventos da `aws.partnercentral-benefits` fonte.

Para exemplos detalhados de políticas do IAM, consulte a documentação da AWS sobre EventBridge permissões.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnerCentralBenefitsEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "events:source": "aws.partnercentral-benefits"
        }
      }
    }
  ]
}
```

Configure EventBridge a Amazon para monitorar eventos

Para monitorar eventos da API de benefícios, você cria uma EventBridge regra que corresponde aos eventos que você deseja capturar. Você pode usar o AWS Management Console ou os AWS SDKs para criar e gerenciar regras. As seções a seguir explicam como criar regras usando os dois métodos. Independentemente do método usado, você deve criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

AWS Management Console configuração

Para configurar uma EventBridge regra usando o AWS Management Console, siga as etapas em Criação de regras que reagem a eventos na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Ao criar a regra, você deve definir o barramento de eventos como padrão e criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

Veja a seguir um exemplo de uma regra de evento:

```
{
    "source": ["aws.partnercentral-benefits"],
    "detail": {
        "catalog": ["AWS"]
    }
}
```

AWS Configuração do SDK

Você pode usar os AWS SDKs para criar e gerenciar EventBridge regras de forma programática. Para obter mais informações, consulte PutRulea Amazon EventBridge API Reference.

O exemplo a seguir usa o AWS SDK para Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='MyBenefitApplicationCreatedRule',
    EventPattern=
    '{
        "source": ["aws.partnercentral-benefits"],
        "detail-type": ["Benefit Application Created"],
        "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Saiba mais sobre os eventos da API de benefícios

As seções a seguir descrevem os tipos de eventos da API de benefícios, os cenários que os acionam e os exemplos de eventos.

Tipos de eventos

A seguir estão os tipos de eventos e seus gatilhos.

Aplicações de benefícios

- Aplicativo de benefícios criado: quando um aplicativo de benefícios é criado
- Solicitação de benefícios em análise: Quando uma solicitação de benefício é enviada ou quando a solicitação passa da Ação necessária para A análise, esse evento é acionado.
- Ação de solicitação de benefícios necessária: Esse evento ocorre quando um revisor interno atualiza o status da inscrição para indicar que o parceiro precisa fornecer informações adicionais, esclarecimentos ou modificações antes que a análise possa continuar.

- Solicitação de benefícios rejeitada: esse evento ocorre quando um revisor interno atualiza o status da solicitação para indicar que a solicitação não atende aos critérios ou requisitos do benefício e foi recusada
- Solicitação de benefícios aprovada: Esse evento ocorre quando um revisor interno atualiza o status da solicitação para indicar que a solicitação foi aprovada para alocação de benefícios.
- Estágio de solicitação de benefícios alterado: Esse evento ocorre quando um revisor interno atualiza o estágio da solicitação durante processos de revisão, como análise comercial,AWS revisão, análise financeira etc.

Alocações de benefícios

- Alocação de benefícios ativada: quando uma alocação de benefícios é criada ou atualizada de volta para Ativada por um serviço interno.
- Alocação de benefícios inativada: esse evento ocorre quando um revisor interno atualiza o status da alocação ou quando uma alocação de benefícios expira automaticamente.
- Alocação de benefícios cumprida: esse evento ocorre quando um revisor interno atualiza o status da alocação para indicar que a alocação foi totalmente utilizada.

Eventos de exemplo

As seções a seguir fornecem exemplos dos eventos listados anteriormente na seção anterior.

Tópicos

- Aplicativo de benefícios criado
- Solicitação de benefícios em análise
- Ação de solicitação de benefícios necessária
- Pedido de benefício rejeitado
- Solicitação de benefícios aprovada
- Estágio de solicitação de benefícios alterado
- Alocação de benefícios ativada
- Alocação de benefícios inativada
- Alocação de benefícios cumprida

Aplicativo de benefícios criado

Contexto de acionamento

O evento Benefit Application Created é publicado na Amazon EventBridge quando um novo Pedido de Benefícios é criado com sucesso por meio do Serviço de Benefícios. Esse evento ocorre quando a chamada da CreateBenefitApplication API é concluída com êxito, indicando que um parceiro iniciou uma solicitação para um benefício específico.

Esse evento representa o início do ciclo de vida da solicitação de benefícios e fornece aos clientes uma notificação imediata quando novas solicitações são enviadas para suas contas.

Destinatários

A conta proprietária receberá o evento para a alocação de benefícios.

Exemplo

```
{
  "id": "7gh88765-k0jh-d08g-i292-2ff1796m030n",
  "detail-type": "Benefit Application Created",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-10T15:45:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "PENDING_SUBMISSION",
      "revision": "hh7e8400-e29b-41d4-a716-446655440012"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
```

```
}
```

Solicitação de benefícios em análise

Contexto de acionamento

O evento Benefit Application In Review é publicado na Amazon EventBridge quando um aplicativo de benefícios passa para o status IN_REVIEW por meio do Benefits Service. Esse evento ocorre quando:

- Um parceiro envia sua inscrição por meio da SubmitBenefitApplication chamada de API
- Um aplicativo faz a transição de ACTION_REQUIRED de volta para IN_REVIEW depois que o parceiro responde ao feedback
- Uma solicitação é recuperada e, em seguida, reenviada.

Esse evento indica que a inscrição entrou no processo formal de revisão e está sendo avaliada por equipes AWS internas.

Destinatários

A conta proprietária receberá o evento para a alocação de benefícios.

Exemplo

```
{
  "id": "8hi99876-l1ki-e19h-j303-3gg2807n141o",
  "detail-type": "Benefit Application In Review",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-10T16:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "IN_REVIEW",
```

```
      "stage": "BUSINESS_REVIEW",
      "revision": "ii8f9511-f30c-52e5-b827-557766551123"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Ação de solicitação de benefícios necessária

Contexto de acionamento

O evento Benefit Application Action Required é publicado na Amazon EventBridge quando um aplicativo de benefícios faz a transição para o status ACTION_REQUIRED por meio do Benefits Service. Esse evento ocorre quando um revisor interno atualiza o status do aplicativo por meio da UpdateBenefitApplicationInternal API para indicar que o parceiro precisa fornecer informações adicionais, esclarecimentos ou modificações antes que a análise possa continuar.

Esse evento representa um ponto crítico no ciclo de vida do aplicativo em que a intervenção do parceiro é necessária para levar o aplicativo adiante no processo de análise.

Destinatários

A conta proprietária receberá o evento para a alocação de benefícios

Exemplo

```
{
  "id": "9ij00987-m2lj-f20i-k414-4hh3918o252p",
  "detail-type": "Benefit Application Action Required",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-12T10:30:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
```

```
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "ACTION_REQUIRED",
      "stage": "BUSINESS_REVIEW",
      "revision": "jj9g0622-g41d-63f6-c938-668877662234"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```


Pedido de benefício rejeitado

Contexto de acionamento

O evento Solicitação de Benefício Rejeitada é publicado na Amazon EventBridge quando uma Solicitação de Benefício passa para o status REJEITADA por meio do Serviço de Benefícios. Esse evento ocorre quando um revisor interno atualiza o status do aplicativo por meio da UpdateBenefitApplicationInternal API para indicar que o aplicativo não atende aos critérios ou requisitos de benefícios e foi recusado.

Esse evento representa um estado terminal no ciclo de vida do aplicativo em que o aplicativo foi formalmente recusado e não prosseguirá com a alocação de benefícios.

Destinatários

A conta proprietária receberá o evento para a alocação de benefícios.

Exemplo


```
{
  "id": "0kl11098-n3mk-g31j-l525-5ii4029p363q",
  "detail-type": "Benefit Application Rejected",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-15T14:20:00Z",
  "region": "us-east-1",
  "resources": [
```

```
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "REJECTED",
      "revision": "kk0h1733-h52e-74g7-d049-779988773345"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Solicitação de benefícios aprovada

Contexto de acionamento

O evento Solicitação de Benefício Aprovada é publicado na Amazon EventBridge quando uma Solicitação de Benefício passa para o status APROVADO por meio do Serviço de Benefícios. Esse evento ocorre quando um revisor interno atualiza o status do aplicativo por meio da UpdateBenefitApplicationInternal API para indicar que o aplicativo atende a todos os critérios e requisitos de benefícios e foi aprovado para alocação de benefícios.

Esse evento representa uma conclusão bem-sucedida do processo de análise da inscrição e normalmente aciona a criação de uma alocação de benefícios correspondente.

Destinatários

A conta proprietária receberá o evento para a alocação de benefícios.

Exemplo

```
{
  "id": "1lm22109-o4nl-h42k-m636-6jj5130q474r",
  "detail-type": "Benefit Application Approved",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
```

```
  "time": "2025-02-14T16:45:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "APPROVED",
      "revision": "ll1i2844-i63f-85h8-e150-880099884456"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Estágio de solicitação de benefícios alterado

Contexto de acionamento

O evento Benefit Application Stage Changed é publicado na Amazon EventBridge quando o estágio de análise de um pedido de benefícios é atualizado por meio do Serviço de Benefícios. Esse evento ocorre quando um revisor interno atualiza o estágio do aplicativo para um dos possíveis valores de Enum (FINANCE_REVIEW, BUSINESS_REVIEW, TECH_REVIEW e _REVIEW).AWS

Diferentemente das mudanças de status, as mudanças de estágio são atualizações informativas somente para leitura que fornecem visibilidade da progressão do fluxo de trabalho de revisão interna sem exigir a ação do parceiro.

Destinatários

A conta proprietária receberá o evento para a alocação de benefícios.

Exemplo

```
{
  "id": "2mn33210-p5om-i53l-n747-7kk6241r585s",
```

```
  "detail-type": "Benefit Application Stage Changed",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-02-13T11:15:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-application/
benappl-12345abcdef123"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitApplication": {
      "id": "benappl-12345abcdef123",
      "fulfillmentTypes": ["CREDITS"],
      "status": "IN_REVIEW",
      "stage": "FINANCIAL_REVIEW",
      "revision": "mm2j3955-j74g-96i9-f261-991100995567"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```


Alocação de benefícios ativada

Contexto de acionamento

O evento Benefit Allocation Activated é publicado na Amazon EventBridge quando uma alocação de benefícios passa para o status ATIVO por meio do Benefits Service. Esse evento ocorre quando um revisor interno atualiza o status da alocação por meio da UpdateBenefitAllocationInternal API para reativar uma alocação anteriormente inativa.

Esse evento geralmente ocorre quando uma alocação que foi temporariamente inativada (devido à expiração, mudanças na política ou motivos administrativos) é restaurada ao status de ativa, disponibilizando novamente os benefícios para utilização pelo parceiro.

Destinatários

A AWS conta pertencente ao parceiro receberá a mensagem. Em outras palavras, a AWS Conta que possui o recurso.

Evento de exemplo

```
{
  "id": "3no44321-q6pn-j64m-o858-8ll7352s696t",
  "detail-type": "Benefit Allocation Activated",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-03-01T08:15:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-allocation/benalloc-
abc123def456"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitAllocation": {
      "id": "benalloc-abc123def456",
      "fulfillmentType": "CREDITS",
      "status": "ACTIVE"
    },
    "benefitApplication": {
      "id": "benappl-12345abcdef123"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Alocação de benefícios inativada

Contexto de acionamento

O evento de alocação de benefícios inativada é publicado na Amazon EventBridge quando uma alocação de benefícios passa para o status INATIVO por meio do Serviço de benefícios. Esse evento ocorre quando um revisor interno atualiza o status da alocação por meio da UpdateBenefitAllocationInternal API ou quando uma alocação de benefícios expira automaticamente.

Esse evento geralmente ocorre quando uma alocação é temporariamente suspensa (devido à expiração, mudanças na política, problemas de conformidade ou motivos administrativos), tornando os benefícios indisponíveis para utilização pelo parceiro até serem reativados.

Destinatários

A AWS conta pertencente ao parceiro receberá a mensagem. Em outras palavras, a AWS Conta que possui o recurso.

Evento de exemplo

```
{
  "id": "4op55432-r7qo-k75n-p969-9mm8463t707u",
  "detail-type": "Benefit Allocation Inactivated",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-06-30T23:59:59Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-allocation/benalloc-
abc123def456"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitAllocation": {
      "id": "benalloc-abc123def456",
      "fulfillmentType": "CREDITS",
      "status": "INACTIVE"
    },
    "benefitApplication": {
      "id": "benappl-12345abcdef123"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Alocação de benefícios cumprida

Contexto de acionamento

O evento Benefit Allocation Fulfilled é publicado na Amazon EventBridge quando uma alocação de benefícios passa para o status FULFILLED por meio do Benefits Service. Esse evento ocorre quando

um revisor interno atualiza o status da alocação por meio da UpdateBenefitAllocationInternal API para indicar que a alocação foi totalmente utilizada ou que os termos do benefício foram atendidos.

Esse evento representa a conclusão do ciclo de vida do benefício, indicando que o parceiro utilizou com sucesso o benefício alocado ou que a alocação chegou à sua conclusão natural.

Destinatários

A AWS conta pertencente ao parceiro receberá a mensagem. Em outras palavras, a AWS Conta que possui o recurso.

Evento de exemplo

```
{
  "id": "5pq66543-s8rp-l86o-q070-0nn9574u818v",
  "detail-type": "Benefit Allocation Fulfilled",
  "source": "aws.partnercentral-benefits",
  "account": "123456789012",
  "time": "2025-12-15T17:30:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:partnercentral:us-east-1:123456789012:benefit-allocation/benalloc-
abc123def456"
  ],
  "detail": {
    "catalog": "AWS",
    "benefitAllocation": {
      "id": "benalloc-abc123def456",
      "fulfillmentType": "CREDITS",
      "status": "FULFILLED"
    },
    "benefitApplication": {
      "id": "benappl-12345abcdef123"
    },
    "benefit": {
      "id": "ben-xyz789uvw012",
      "name": "AWS Partner Credits",
      "program": ["AWS Partner Network"]
    }
  }
}
```

Notificações para o AWS API do Partner Central Channel

Os eventos da API do canal fornecem notificações em tempo real sobre mudanças no status ou nos detalhes das atividades relacionadas ao canal. Esses eventos ajudam a manter seus sistemas sincronizados com o AWS Partner Central e ajudam a garantir respostas e atualizações oportunas.

Pré-requisitos

Você precisa das permissões apropriadas do IAM para acessar e gerenciar eventos da API do AWS Partner Central Channel. Para obter informações sobre as ações, os recursos e as chaves de condição disponíveis para EventBridge, consulte Usando as condições da política do IAM na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Uma das chaves de condição é`events:detail-type`, que você pode usar para definir o escopo de permissões para tipos específicos de eventos.

O exemplo de política a seguir demonstra como personalizar e definir o escopo das permissões para os eventos. A `AllowPutRuleForPartnercentralChannelEvents` declaração permite a criação de regras, mas somente para eventos da `aws.partnercentral-channel` fonte.

Para exemplos detalhados de políticas do IAM, consulte a documentação da AWS sobre EventBridge permissões.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPutRuleForPartnerCentralChannelEvents",
      "Effect": "Allow",
      "Action": "events:PutRule",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "events:source": "aws.partnercentral-channel"
        }
      }
    }
  ]
}
```

Configure EventBridge a Amazon para monitorar eventos

Para monitorar os eventos da API do canal, você cria uma EventBridge regra que corresponde aos eventos que você deseja capturar. Você pode usar o AWS Management Console ou os AWS SDKs para criar e gerenciar regras. As seções a seguir explicam como criar regras usando os dois métodos. Independentemente do método usado, você deve criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

AWS Management Console configuração

Para configurar uma EventBridge regra usando o AWS Management Console, siga as etapas em Criação de regras que reagem a eventos na Amazon EventBridge no Guia EventBridge do usuário da Amazon. Ao criar a regra, você deve definir o barramento de eventos como padrão e criar a regra na `us-east-1` região Leste dos EUA (Norte da Virgínia).

Veja a seguir um exemplo de uma regra de evento:

```
{
    "source": ["aws.partnercentral-channel"],
    "detail": {
        "catalog": ["AWS"]
    }
}
```

AWS Configuração do SDK

Você pode usar os AWS SDKs para criar e gerenciar EventBridge regras de forma programática. Para obter mais informações, consulte PutRulea Amazon EventBridge API Reference.

O exemplo a seguir usa o AWS SDK para Python (Boto3):

```
import boto3

client = boto3.client('events', region_name='us-east-1')

response = client.put_rule(
    Name='ChannelHandshakeCreatedRule',
    EventPattern=
    '{
        "source": ["aws.partnercentral-channel"],
        "detail-type": ["Channel Handshake Created"],
```

```
        "detail": {"catalog": ["AWS"]}
    }',
    State='ENABLED'
)
print('Rule ARN:', response['RuleArn'])
```

Saiba mais sobre os eventos da API do canal

As seções a seguir descrevem os tipos de eventos da API do canal, os cenários que os acionam e os exemplos de eventos.

Tipos de eventos

A seguir estão os tipos de eventos e seus gatilhos.

- Aperto de mão de canal criado: acionado quando um novo aperto de mão de canal é criado.
- Aperto de mão de canal aceito: acionado quando um novo aperto de mão de canal é aceito.
- Aperto de mão do canal rejeitado: acionado quando um novo aperto de mão do canal é rejeitado.

Canal Handshake criado

Os exemplos a seguir mostram eventos típicos do Channel Handshake Created para diferentes tipos de handshake.

Inicie o período de serviço Handshake:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Created",
    "source": "aws.partnercentral-channel",
    "account": "789789789789",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-abc123def456g"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
```

```
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "START_SERVICE_PERIOD",
        "id": "ch-abc123def456g",
        "receiverAccountId": "789789789789",
        "ownerAccountId": "123123123123",
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "startServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}
```

Revogar o Handshake do Período de Serviço:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Created",
    "source": "aws.partnercentral-channel",
    "account": "789789789789",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-def456ghi789j"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "REVOKE_SERVICE_PERIOD",
        "id": "ch-def456ghi789j",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "789789789789",
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
```

```
        "handshakeDetail": {
            "revokeServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}
```

Handshake da conta de gerenciamento de programas:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Created",
    "source": "aws.partnercentral-channel",
    "account": "456456456456",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-ghi789jkl012m"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "pma-abc123def456g",
        "handshakeType": "PROGRAM_MANAGEMENT_ACCOUNT",
        "id": "ch-ghi789jkl012m",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "456456456456",
        "senderAccountId": "123123123123",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "programManagementAccountHandshakeDetail": {
                "program": "SOLUTION_PROVIDER"
            }
        }
    }
}
```

Handshake de canal aceito

Os exemplos a seguir mostram eventos típicos do Channel Handshake Accepted para diferentes tipos de handshake.

Inicie o período de serviço Handshake:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Accepted",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-abc123def456g"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "START_SERVICE_PERIOD",
        "id": "ch-abc123def456g",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "789789789789",
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "startServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}
```

Revogar o Handshake do Período de Serviço:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Accepted",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-def456ghi789j"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "REVOKE_SERVICE_PERIOD",
        "id": "ch-def456ghi789j",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "789789789789",
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "revokeServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}
```


Handshake da conta de gerenciamento de programas:


```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Accepted",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
```

```
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-ghi789jkl012m"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "pma-abc123def456g",
        "handshakeType": "PROGRAM_MANAGEMENT_ACCOUNT",
        "id": "ch-ghi789jkl012m",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "456456456456",
        "senderAccountId": "123123123123",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "programManagementAccountHandshakeDetail": {
                "program": "SOLUTION_PROVIDER"
            }
        }
    }
}
```

Canal Handshake rejeitado

Os exemplos a seguir mostram eventos típicos do Channel Handshake Rejected para diferentes tipos de handshake.

Inicie o período de serviço Handshake:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Rejected",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-abc123def456g"
    ],
    "detail": {
```

```
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "START_SERVICE_PERIOD",
        "id": "ch-abc123def456g",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "789789789789",
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "startServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}
```

Revogar o Handshake do Período de Serviço:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Rejected",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-def456ghi789j"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "rs-jkl012mno345p",
        "handshakeType": "REVOKE_SERVICE_PERIOD",
        "id": "ch-def456ghi789j",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "789789789789",
```

```
        "senderAccountId": "456456456456",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "revokeServicePeriodHandshakeDetail": {
                "servicePeriodType": "MINIMUM_NOTICE_PERIOD",
                "minimumNoticeDays": "14",
                "endDate": null,
                "startDate": "2025-02-02T00:00:00Z",
                "note": "Test note example"
            }
        }
    }
}
```

Handshake da conta de gerenciamento de programas:

```
{
    "version": "0",
    "id": "01234567-0123-0123-0123-0123456789ab",
    "detail-type": "Channel Handshake Rejected",
    "source": "aws.partnercentral-channel",
    "account": "123123123123",
    "time": "2025-02-01T00:00:00Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:partnercentral:us-east-1:123123123123:catalog/AWS/channel-handshake/
ch-ghi789jkl012m"
    ],
    "detail": {
        "requestId": "12345678-1234-1234-1234-123456789abc",
        "catalog": "AWS",
        "associatedResourceIdentifier": "pma-abc123def456g",
        "handshakeType": "PROGRAM_MANAGEMENT_ACCOUNT",
        "id": "ch-ghi789jkl012m",
        "ownerAccountId": "123123123123",
        "receiverAccountId": "456456456456",
        "senderAccountId": "123123123123",
        "senderDisplayName": "TestDisplayName",
        "handshakeDetail": {
            "programManagementAccountHandshakeDetail": {
                "program": "SOLUTION_PROVIDER"
            }
        }
```

```
    }
}
```

Testando em uma sandbox

Os parceiros podem usar um sandbox para testar e validar suas interações com a API do AWS Partner Central em um ambiente seguro e isolado, garantindo uma operação tranquila antes de promover sua solução para o ambiente de produção.AWS oferece um sandbox dinâmico para usuários da API do AWS Partner Central que retorna respostas semelhantes às do ambiente de produção.AWS não fornece uma interface de usuário para o ambiente sandbox. Portanto, os parceiros precisam confiar nas respostas programáticas para testar suas soluções.

Acesso ao ambiente sandbox

Os parceiros obtêm acesso ao ambiente de teste assim que vinculam sua conta Conta da AWSà Partner Central. Para obter mais informações, consulte Vinculando sua AWS conta à conta do AWS Partner Central. Cada solicitação inclui um `catalog` parâmetro que determina o ambiente de dados. Quando `catalog` definido como`AWS`, ele faz referência aos dados de produção e, quando definido como`Sandbox`, faz referência aos dados do sandbox.

Detalhes importantes sobre o ambiente sandbox

1. Atualização de dados: uma vez por ano,AWS atualiza os dados no ambiente sandbox (normalmente no início do ano). Após essa atualização, você pode perder alguns dos dados em seu ambiente de teste.
2. Escopo do teste: o ambiente sandbox é normalmente usado para testes funcionais e não para testar escalabilidade ou desempenho.

Tópicos

- Testando em uma caixa de areia para o AWS API de conta do Partner Central
- Testando em uma caixa de areia para o AWS API de vendas da Central de Parceiros
- Testando em uma caixa de areia para o AWS API de benefícios do Partner Central
- Testando em uma caixa de areia para o AWS API do Partner Central Channel
- Testando em uma caixa de areia para o AWS API de medição de receita do Partner Central

Testando em uma caixa de areia para o AWS API de conta do Partner Central

Como usar o sandbox

Para usar o sandbox, conclua as seguintes etapas:

1. Crie uma função do IAM:

   Crie uma função do IAM Conta da AWS vinculada à sua conta do AWS Partner Central.

2. Atribuir política:

   Atribua a política a seguir à função do IAM. Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccountManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:AcceptConnectionInvitation",
        "partnercentral:AssociateAwsTrainingCertificationEmailDomain",
        "partnercentral:CancelConnection",
        "partnercentral:CancelConnectionInvitation",
        "partnercentral:CancelProfileUpdateTask",
        "partnercentral:CreateConnectionInvitation",
        "partnercentral:CreatePartner",
        "partnercentral:DisassociateAwsTrainingCertificationEmailDomain",
        "partnercentral:GetAllianceLeadContact",
        "partnercentral:GetConnection",
        "partnercentral:GetConnectionInvitation",
        "partnercentral:GetConnectionPreferences",
        "partnercentral:GetPartner",
        "partnercentral:GetProfileUpdateTask",
        "partnercentral:GetProfileVisibility",
  "partnercentral:GetVerification",
        "partnercentral:ListConnectionInvitations",
        "partnercentral:ListConnections",
        "partnercentral:ListPartners",
        "partnercentral:PutAllianceLeadContact",
```

```
        "partnercentral:PutProfileVisibility",
        "partnercentral:RejectConnectionInvitation",
        "partnercentral:SendEmailVerificationCode",
        "partnercentral:StartProfileUpdateTask",
  "partnercentral:StartVerification",
        "partnercentral:UpdateConnectionPreferences"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "Sandbox"
          ]
        }
      }
    },
    {
      "Sid": "TaggingAccess",
      "Effect": "Allow",
      "Action": [
        "partnercentral:TagResource",
        "partnercentral:UntagResource",
        "partnercentral:ListTagsForResource"
      ],
      "Resource": [
        "arn:aws:partnercentral:*:*:catalog/Sandbox/partner/*"
      ],
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "Sandbox"
          ]
        }
      }
    }
  ]
}
```

3. Use as credenciais da função do IAM:

   Use as credenciais (chave secreta e chave de acesso) dessa função do IAM em sua solução para realizar as ações da API.

Teste AWS actions

Durante a fase de teste, muitas vezes é necessário simular AWS ações. Essa simulação permite que os parceiros testem minuciosamente o fluxo completo de sua integração com AWS os serviços. Cada solicitação inclui um parâmetro de catálogo, que determina o ambiente de dados.

Simulando a criação de um parceiro

Para simular a criação de um parceiro, na carga útil da `CreatePartner` ação, inclua `"catalog": "Sandbox"` na carga útil.

Por exemplo:

```
{
  "ClientToken": "client-generated-uuid",
  "Catalog": "Sandbox",
  "LegalName": "Your Name",
  "PrimarySolutionType": "SOLUTION_TYPE",
  "AllianceLeadContact": {
    "FirstName": "Example First Name",
    "LastName": "Example Last Name",
    "BusinessTitle": "Example Title",
    "Email": "example@domain.com"
  },
  "EmailVerificationCode": "123456"
}
```

Nota: No Sandbox, a verificação de e-mail está desativada. Você pode usar qualquer um de seis dígitos EmailVerificationCode para testar. Além disso, não use “Tags” na solicitação.

Notas de teste adicionais

1. Para testar outras ações, defina `"catalog": "Sandbox"` a carga útil.
2. Para testar as conexões da conta do parceiro no sandbox, você deve executar a `StartProfileUpdateTask` ação após criar o parceiro no catálogo do sandbox.
3. Para passar para a produção, altere o valor do catálogo de `Sandbox` para`AWS`.

Eventos de teste no ambiente sandbox

Os parceiros podem consumir eventos do ambiente sandbox para ajudar a testar as implementações baseadas em eventos. Configure EventBridge o mesmo Conta da AWS com regras para ouvir

eventos de sandbox especificando `catalog: Sandbox` nos detalhes do evento. Para obter mais informações, consulte Notificações para o AWS API de conta do Partner Central.

Exemplo de regra de evento:

```
{
  "source": ["aws.partnercentral-account"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

As regras de eventos que especificam `catalog` como `Sandbox` corresponderão somente aos eventos provenientes do sandbox, gerados pelas ações que você executa no ambiente sandbox.

Testando em uma caixa de areia para o AWS API de vendas da Central de Parceiros

Como usar o sandbox

1. Crie uma função do IAM:

   Crie uma função do IAM Conta da AWS vinculada à sua conta do AWS Partner Central.

2. Atribuir política:

   Atribua a política a seguir à função do IAM. Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "partnercentral:CreateOpportunity",
                "partnercentral:UpdateOpportunity",
                "partnercentral:ListOpportunities",
                "partnercentral:GetOpportunity",
                "partnercentral:GetAwsOpportunitySummary",
                "partnercentral:ListSolutions",
```

```
                "partnercentral:AssociateOpportunity",
                "partnercentral:DisassociateOpportunity",
                "partnercentral:AssignOpportunity",
                "partnercentral:SubmitOpportunity",
                "partnercentral:AcceptEngagementInvitation",
                "partnercentral:CreateEngagementInvitation",
                "partnercentral:RejectEngagementInvitation",
                "partnercentral:GetEngagementInvitation",
                "partnercentral:ListEngagementInvitations",
                "partnercentral:StartEngagementFromOpportunityTask",
                "partnercentral:StartEngagementByAcceptingInvitationTask",
                "partnercentral:CreateResourceSnapshotJob",
                "partnercentral:StartResourceSnapshotJob",
                "partnercentral:CreateEngagement"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": "Sandbox"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:ListEntities",
                "aws-marketplace:DescribeEntity"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:SearchAgreements",
                "aws-marketplace:DescribeAgreement"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws-marketplace:PartyType": "Proposer"
                }
            }
        }
    ]
```

```
}
```

3. Use as credenciais da função do IAM:

   Use as credenciais (chave secreta e chave de acesso) dessa função do IAM em sua solução para realizar as ações da API.

Teste AWS actions

Durante a fase de teste, muitas vezes é necessário simular AWS ações. Essa simulação permite que os parceiros testem minuciosamente o fluxo completo de sua integração com AWS os serviços.

Simulando a criação de um AWS oportunidade originada

Para simular a criação de uma oportunidade AWS originada, na carga útil da `CreateOpportunity` ação, inclua e. `"Catalog": "Sandbox"` `"Origin": "AWS Referral"`

Por exemplo:

```
{
    "Catalog": "Sandbox",
    "Origin": "AWS Referral",
    "OpportunityIdentifier": "O123456",
    "Title": "Test Opportunity",
    ...
}
```

Simulando atualizações em uma oportunidade originada por um parceiro

Para simular atualizações ou outras ações em uma oportunidade originada pelo parceiro, use a `UpdateOpportunity` ação com `Lifecycle.ReviewStatus: “Approved”` e/ou `“Action Required”` na carga `"Catalog": "Sandbox"` útil.

Se você estiver retirando a carga da `GetOpportunity` ação para fazer a atualização, certifique-se de alterar a ID para `Identifier` e remover os seguintes campos:

- `CreatedDate`
- `OpportunityTeam`
- `RelatedEntityIdentifiers`

Por exemplo:

```
{
    "Catalog": "Sandbox",
    "Identifier": "O123456",
    "Title": "Updated Test Opportunity",
    "Lifecycle": {
        "ReviewStatus": "Approved"
    }
    ...
}
```

Eventos de teste no ambiente sandbox

Os parceiros podem consumir eventos de oportunidade do ambiente sandbox para ajudar a testar as implementações baseadas em eventos. Configure EventBridge o mesmo Conta da AWS com regras para ouvir eventos de sandbox especificando `catalog: sandbox` nos detalhes do evento. Para obter mais informações, consulte Notificações para o AWS API de vendas da Central de Parceiros.

Exemplo de regra de evento:

```
{
    "source": ["aws.partnercentral-selling"],
    "detail": {
        "catalog": ["Sandbox"]
    }
}
```

As regras de eventos que especificam `catalog` como `Sandbox` corresponderão somente aos eventos provenientes do sandbox, gerados pelas ações que você executa no ambiente sandbox.

Notas de teste adicionais

1. `AssociateOpportunity`Ação de teste:
   a. Use a solução padrão `"S-1234567"` para testar a `AssociateOpportunity` ação.
   b. Para testar as ofertas do Marketplace, use um ID de oferta real da sua conta.
2. Para passar para a produção, altere o `catalog` valor de `Sandbox` para`AWS`.

Como obter ajuda

Se você encontrar desafios para integrar seu CRM ou precisar testar um cenário específico não abordado aqui, entre em contato com AWS o suporte apresentando um caso por meio das seguintes etapas:

1. Faça login no AWS Partner Central com suas credenciais do AWS Partner Network.
2. No Support Center do Partner Central, escolha `Open New Case` registrar um novo caso. Preencha os campos da seguinte forma:
   a. Tipo de estojo de suporte:`AWS Partner Central`.
   b. Pergunta sobre:`Partner Central Tools or ACE leads and opportunities`.
   c. Seja específico: selecione o tipo de caso de integração de CRM mais adequado.
   d. Assunto: Inclua uma breve descrição da solicitação.
   e. Descrição: forneça uma descrição detalhada dos problemas, perguntas, erros e etapas de solução de problemas.
   f. Anexos: inclua registros e capturas de tela, quando aplicável.

Testando em uma caixa de areia para o AWS API de benefícios do Partner Central

Como usar o sandbox

Para usar o sandbox, conclua as seguintes etapas:

1. Crie uma função do IAM:

   Crie uma função do IAM Conta da AWS vinculada à sua conta do AWS Partner Central.
2. Atribuir política:

   Atribua a política a seguir à função do IAM. Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

```
{
  "Version": "2012-10-17",
    "Statement": [{
            "Sid": "BenefitsManagement",
            "Effect": "Allow",
```

```
            "Action": [
                "partnercentral:ListBenefits",
                "partnercentral:GetBenefit",
                "partnercentral:CreateBenefitApplication",
                "partnercentral:AmendBenefitApplication",
                "partnercentral:UpdateBenefitApplication",
                "partnercentral:SubmitBenefitApplication",
                "partnercentral:GetBenefitApplication",
                "partnercentral:CancelBenefitApplication",
                "partnercentral:RecallBenefitApplication",
                "partnercentral:ListBenefitApplications",
                "partnercentral:AssociateBenefitApplicationResource",
                "partnercentral:DisasssociateBenefitApplicationResource",
                "partnercentral:ListBenefitAllocations",
                "partnercentral:GetBenefitAllocation",
                "partnercentral:TagResource",
                "partnercentral:UntagResource",
                "partnercentral:ListTagsForResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "AWSPartnerOpportunityAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:GetAwsOpportunitySummary",
                "partnercentral:GetOpportunity",
                "partnercentral:ListOpportunities"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
```

```
        },
        {
            "Sid": "ListingAWSMarketplaceEntities",
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:ListEntities"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AWSMarketplaceOffersAccess",
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:DescribeEntity"
            ],

            "Resource": [
                "arn:aws:aws-marketplace:::AWSMarketplace/Offer/*"
            ]
        },
        {
            "Sid": "S3PutObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject"
            ],
            "Resource": ["arn:aws:s3:::aws-partner-central-marketplace-ephemeral-
writeonly-files/*"]
        },
        {
            "Sid": "TaggingAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:TagResource",
                "partnercentral:UntagResource",
                "partnercentral:ListTagsForResource"
            ],
            "Resource": [
                "arn:aws:partnercentral:us-east-1:*:catalog/Sandbox/benefit-
application/*",
                "arn:aws:partnercentral:us-east-1:*:catalog/Sandbox/benefit-
allocation/*"
            ],
            "Condition": {
```

```
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        }
    ]
}
```

3. Use as credenciais da função do IAM:

   Use as credenciais (chave secreta e chave de acesso) dessa função do IAM em sua solução para realizar as ações da API.

Teste AWS actions

Durante a fase de teste, muitas vezes é necessário simular AWS ações. Essa simulação permite que os parceiros testem minuciosamente o fluxo completo de sua integração com AWS os serviços. Cada solicitação inclui um parâmetro de catálogo, que determina o ambiente de dados.

Simulando a criação de um aplicativo de benefícios

Para simular a criação de um aplicativo de benefícios, na carga útil da `CreateBenefitApplication` ação, inclua `"catalog": "Sandbox"` na carga útil.

Por exemplo:

```
{
  "Catalog": "Sandbox",
  "ClientToken": "String",
  "BenefitIdentifier": "String",
  "FulfillmentTypes": ["String"],
  "BenefitApplicationDetails": JsonDocument,
  "Name": "String",
  "Description": "String",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
```

```
  "PartnerContacts": [
    {
      "BusinessTitle": "string",
      "Email": "string",
      "FirstName": "string",
      "LastName": "string",
      "Phone": "string"
    }
  ],
  "FileDetails": [
    {
      "FileURI": "String",
      "BusinessUseCase": "String"
    }
  ],
  "AssociatedResources": [
    {
      "ResourceType": "BENEFIT_ALLOCATION | OPPORTUNITY",
      "ResourceArn": "ARN"
    }
  ]
}
```

Notas de teste adicionais

1. Para testar outras ações, defina `"catalog": "Sandbox"` a carga útil.
2. Os IDs de benefícios no catálogo Sandbox e no catálogo da AWS são diferentes. Faça uma chamada de `GetBenefit` API com `"catalog": "Sandbox"` para obter o Identificador de Benefícios na Sandbox.
3. A alocação de benefícios no catálogo Sandbox e no catálogo da AWS é diferente.
4. Para passar para a produção, altere o valor do catálogo de `Sandbox` para`AWS`.

Eventos de teste no ambiente sandbox

Os parceiros podem consumir eventos do ambiente sandbox para ajudar a testar as implementações baseadas em eventos. Configure EventBridge o mesmo Conta da AWS com regras para ouvir eventos de sandbox especificando `catalog: Sandbox` nos detalhes do evento. Para obter mais informações, consulte Notificações para o AWS API de benefícios do Partner Central.

Exemplo de regra de evento:

```
{
  "source": ["aws.partnercentral-benefits"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

As regras de eventos que especificam `catalog` como `Sandbox` corresponderão somente aos eventos provenientes do sandbox, gerados pelas ações que você executa no ambiente sandbox.

Testando em uma caixa de areia para o AWS API do Partner Central Channel

As contas no sandbox não estão qualificadas para os benefícios do canal e não são validadas de acordo com todos os requisitos do programa.

Como usar o sandbox

Para usar o sandbox, conclua as seguintes etapas:

1. Crie uma função do IAM:

   Crie uma função do IAM Conta da AWS vinculada à sua conta do AWS Partner Central.

2. Atribuir política:

   Atribua a política a seguir à função do IAM. Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ChannelManagement",
      "Effect": "Allow",
      "Action": [
        "partnercentral:CreateProgramManagementAccount",
        "partnercentral:UpdateProgramManagementAccount",
        "partnercentral:DeleteProgramManagementAccount",
        "partnercentral:ListProgramManagementAccounts",
        "partnercentral:GetProgramManagementAccount",
```

```
        "partnercentral:CreateRelationship",
        "partnercentral:UpdateRelationship",
        "partnercentral:DeleteRelationship",
        "partnercentral:GetRelationship",
        "partnercentral:ListRelationships",
        "partnercentral:CreateChannelHandshake",
        "partnercentral:AcceptChannelHandshake",
        "partnercentral:RejectChannelHandshake",
        "partnercentral:CancelChannelHandshake",
        "partnercentral:ListChannelHandshakes"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "Sandbox"
          ]
        }
      }
    },
    {
      "Sid": "TaggingAccess",
      "Effect": "Allow",
      "Action": [
        "partnercentral:TagResource",
        "partnercentral:UntagResource",
        "partnercentral:ListTagsForResource"
      ],
      "Resource": [
        "arn:aws:partnercentral:*:*:catalog/Sandbox/program-management-account/*",
        "arn:aws:partnercentral:*:*:catalog/Sandbox/channel-handshake/*"
      ],
      "Condition": {
        "StringEquals": {
          "partnercentral:Catalog": [
            "Sandbox"
          ]
        }
      }
    }
  ]
}
```

3. Use as credenciais da função do IAM:

Use as credenciais (chave secreta e chave de acesso) dessa função do IAM em sua solução para realizar as ações da API.

Teste AWS actions

Durante a fase de teste, muitas vezes é necessário simular AWS ações. Essa simulação permite que os parceiros testem minuciosamente o fluxo completo de sua integração com AWS os serviços. Cada solicitação inclui um parâmetro de catálogo, que determina o ambiente de dados.

Simulando a criação de uma conta de gerenciamento de programas

Para simular a criação de uma Conta de Gerenciamento do Programa, na carga útil da `CreateProgramManagementAccount` ação, inclua `"catalog": "Sandbox"` na carga útil.

Por exemplo:

```
{
  "catalog": "Sandbox",
  "accountId": "123456789012",
  "displayName": "Test PMA Name",
  "clientToken": "123abc456def789ghi",
  "program": "SOLUTION_PROVIDER",
  "tags": [
    {
      "key": "testkey",
      "value": "testvalue"
    }
  ]
}
```

Simulando atualizações em um relacionamento

Para simular atualizações em um relacionamento, use a `UpdateRelationship` ação com `"catalog": "Sandbox"` na carga útil.

Por exemplo:

```
{
  "catalog": "Sandbox",
  "displayName": "Updated Test PMA",
  "identifier": "rs-abc123def456g",
```

```
  "programManagementAccountIdentifier": "pma-123abc456def7"
}
```

Notas de teste adicionais

1. Para testar outras ações, defina `"catalog": "Sandbox"` a carga útil.
2. Para passar para a produção, altere o valor do catálogo de `Sandbox` para`AWS`.

Eventos de teste no ambiente sandbox

Os parceiros podem consumir eventos do ambiente sandbox para ajudar a testar as implementações baseadas em eventos. Configure EventBridge o mesmo Conta da AWS com regras para ouvir eventos de sandbox especificando `catalog: Sandbox` nos detalhes do evento. Para obter mais informações, consulte Notificações para o AWS API do Partner Central Channel.

Exemplo de regra de evento:

```
{
  "source": ["aws.partnercentral-channel"],
  "detail": {
    "catalog": ["Sandbox"]
  }
}
```

As regras de eventos que especificam `catalog` como `Sandbox` corresponderão somente aos eventos provenientes do sandbox, gerados pelas ações que você executa no ambiente sandbox.

Como obter ajuda

Se você encontrar desafios nos testes ou precisar testar um cenário específico não abordado aqui, entre em contato com o suporte apresentando um caso por meio das seguintes etapas:

1. Faça login no AWS Partner Central com suas credenciais do AWS Partner Network.
2. No Support Center do Partner Central, escolha `Open New Case` registrar um novo caso. Preencha os campos da seguinte forma:
   a. Tipo de estojo de suporte: `Channel Program`
   b. Pergunta sobre: `General Program Inquiry`
   c. Seja específico: selecione o tipo de caso de canal mais adequado.

d. Assunto: Inclua uma breve descrição da solicitação.
e. Descrição: forneça uma descrição detalhada dos problemas, perguntas, erros e etapas de solução de problemas.
f. Anexos: inclua registros e capturas de tela, quando aplicável.

Testando em uma caixa de areia para o AWS API de medição de receita do Partner Central

Como usar o sandbox

Para usar o sandbox, conclua as seguintes etapas:

1. Crie uma função do IAM:

   Crie uma função do IAM Conta da AWS vinculada à sua conta do AWS Partner Central.
2. Atribuir política:

   Atribua a política a seguir à função do IAM. Para obter mais informações, consulte Adicionar e remover permissões de identidade do IAM.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RevenueMeasurementCreation",
            "Effect": "Allow",
            "Action": [
                "partnercentral:CreateRevenueAttribution",
                "partnercentral:CreateMarketplaceRevenueShare",
                "partnercentral:CreateMarketplaceRevenueShareAllocation"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        },
```

```
        {
            "Sid": "RevenueMeasurementListing",
            "Effect": "Allow",
            "Action": [
                "partnercentral:ListRevenueAttributions",
                "partnercentral:ListRevenueAttributionAllocations",
                "partnercentral:ListMarketplaceRevenueShares",
                "partnercentral:ListMarketplaceRevenueShareAllocations"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "RevenueAttributionManagement",
            "Effect": "Allow",
            "Action": [
                "partnercentral:GetRevenueAttribution",
                "partnercentral:UpdateRevenueAttribution",
                "partnercentral:GetRevenueAttributionAllocation",
                "partnercentral:StartRevenueAttributionAllocationsTask",
                "partnercentral:GetRevenueAttributionAllocationsTask"
            ],
            "Resource": "arn:aws:partnercentral:*:*:catalog/Sandbox/revenue-
attribution/*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "MarketplaceRevenueShareManagement",
            "Effect": "Allow",
            "Action": [
                "partnercentral:GetMarketplaceRevenueShare",
                "partnercentral:GetMarketplaceRevenueShareAllocation",
```

```
                "partnercentral:UpdateMarketplaceRevenueShareAllocation"
            ],
            "Resource": "arn:aws:partnercentral:*:*:catalog/Sandbox/marketplace-
revenue-share/*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "TaggingAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:TagResource",
                "partnercentral:UntagResource",
                "partnercentral:ListTagsForResource"
            ],
            "Resource": [
                "arn:aws:partnercentral:*:*:catalog/Sandbox/revenue-attribution/*",
                "arn:aws:partnercentral:*:*:catalog/Sandbox/marketplace-revenue-
share/*"
            ],
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "OpportunityAccess",
            "Effect": "Allow",
            "Action": [
                "partnercentral:ListOpportunities",
                "partnercentral:GetOpportunity"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "partnercentral:Catalog": [
```

```
                        "Sandbox"
                    ]
                }
            }
        },
        {
            "Sid": "ListingAWSMarketplaceEntities",
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:ListEntities"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AWSMarketplaceEntityAccess",
            "Effect": "Allow",
            "Action": [
                "aws-marketplace:DescribeEntity"
            ],
            "Resource": [
                "arn:aws:aws-marketplace:*:*:AWSMarketplace*/Offer/*"
            ]
        }
    ]
}
```

3. Use as credenciais da função do IAM:

   Use as credenciais (chave secreta e chave de acesso) dessa função do IAM em sua solução para realizar as ações da API.

Teste AWS actions

Durante a fase de teste, muitas vezes é necessário simular AWS ações. Essa simulação permite que os parceiros testem minuciosamente o fluxo completo de sua integração com AWS os serviços. Cada solicitação inclui um parâmetro de catálogo, que determina o ambiente de dados.

Simulando a criação de um ID de atribuição de receita

Para simular a criação de um ID de atribuição de receita, na carga útil da `CreateRevenueAttribution` ação, `"Catalog": "Sandbox"` inclua na carga útil.

Por exemplo:

```
{
  "Catalog": "Sandbox",
  "ClientToken": "String",
  "Name": "String",
  "Description": "String",
  "TenancyModel": "MULTI_TENANT | SINGLE_TENANT",
  "ProductIdentifier": "String",
  "Tags": [
    {
      "Key": "String",
      "Value": "String"
    }
  ]
}
```

Notas de teste adicionais

1. Para testar outras ações, defina `"Catalog": "Sandbox"` a carga útil.
2. Para passar para a produção, altere o valor do catálogo de `Sandbox` para`AWS`.

Exemplos de código para o Partner Central usando AWS SDKs

Os exemplos de código a seguir mostram como usar o Partner Central com um kit AWS de desenvolvimento de software (SDK).

Ações são trechos de código de programas maiores e devem ser executadas em contexto. Embora as ações mostrem como chamar perfis de serviço individuais, você pode ver as ações no contexto em seus cenários relacionados.

Cenários são exemplos de código que mostram como realizar tarefas específicas chamando várias funções dentro de um serviço ou combinadas com outros Serviços da AWS.

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Exemplos de código

- Exemplos básicos de uso do Partner Central AWS SDKs
  - Ações para o Partner Central usando AWS SDKs
    - Use AssignOpportunitycom um AWS SDK
    - Use AssociateOpportunitycom um AWS SDK
    - Use CreateOpportunitycom um AWS SDK
    - Use DisassociateOpportunitycom um AWS SDK
    - Use GetAwsOpportunitySummarycom um AWS SDK
    - Use GetEngagementInvitationcom um AWS SDK
    - Use GetOpportunitycom um AWS SDK
    - Use ListEngagementInvitationscom um AWS SDK
    - Use ListOpportunitiescom um AWS SDK
    - Use ListSolutionscom um AWS SDK
    - Use RejectEngagementInvitationcom um AWS SDK
    - Use StartEngagementByAcceptingInvitationTaskcom um AWS SDK
    - Use StartEngagementFromOpportunityTaskcom um AWS SDK
    - Use UpdateOpportunitycom um AWS SDK

- Cenários para o uso do Partner Central AWS SDKs
  - Atualizar a entidade associada de uma oportunidade

Exemplos básicos de uso do Partner Central AWS SDKs

Os exemplos de código a seguir mostram como usar os conceitos básicos do AWS Partner Central com AWS SDKs.

Exemplos

- Ações para o Partner Central usando AWS SDKs
  - Use AssignOpportunitycom um AWS SDK
  - Use AssociateOpportunitycom um AWS SDK
  - Use CreateOpportunitycom um AWS SDK
  - Use DisassociateOpportunitycom um AWS SDK
  - Use GetAwsOpportunitySummarycom um AWS SDK
  - Use GetEngagementInvitationcom um AWS SDK
  - Use GetOpportunitycom um AWS SDK
  - Use ListEngagementInvitationscom um AWS SDK
  - Use ListOpportunitiescom um AWS SDK
  - Use ListSolutionscom um AWS SDK
  - Use RejectEngagementInvitationcom um AWS SDK
  - Use StartEngagementByAcceptingInvitationTaskcom um AWS SDK
  - Use StartEngagementFromOpportunityTaskcom um AWS SDK
  - Use UpdateOpportunitycom um AWS SDK

Ações para o Partner Central usando AWS SDKs

Os exemplos de código a seguir demonstram como realizar ações individuais do Partner Central com AWS SDKs. Cada exemplo inclui um link para GitHub, onde você pode encontrar instruções para configurar e executar o código.

Esses trechos chamam a API da Central de Parceiros e são trechos de código de programas maiores que devem ser executados no contexto. É possível ver as ações em contexto em Cenários para o uso do Partner Central AWS SDKs.

Os exemplos a seguir incluem apenas as ações mais utilizadas. Consulte uma lista completa na Referência de API da Central de Parceiros da AWS.

Exemplos

- Use AssignOpportunitycom um AWS SDK
- Use AssociateOpportunitycom um AWS SDK
- Use CreateOpportunitycom um AWS SDK
- Use DisassociateOpportunitycom um AWS SDK
- Use GetAwsOpportunitySummarycom um AWS SDK
- Use GetEngagementInvitationcom um AWS SDK
- Use GetOpportunitycom um AWS SDK
- Use ListEngagementInvitationscom um AWS SDK
- Use ListOpportunitiescom um AWS SDK
- Use ListSolutionscom um AWS SDK
- Use RejectEngagementInvitationcom um AWS SDK
- Use StartEngagementByAcceptingInvitationTaskcom um AWS SDK
- Use StartEngagementFromOpportunityTaskcom um AWS SDK
- Use UpdateOpportunitycom um AWS SDK

Use **`AssignOpportunity`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `AssignOpportunity`.

Java

SDK para Java 2.x

Reatribua uma oportunidade existente a outro usuário.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
```

```
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssignOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssignOpportunityResponse;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssigneeContact;

/*
Purpose
PC-API-07 Assigning a new owner
*/

public class AssignOpportunity {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     String assigneeFirstName = "John";

     String assigneeLastName = "Doe";

     String assigneeEmail = "test@test.com";

     String businessTitle = "PartnerAccountManager";

     AssignOpportunityResponse response = getResponse(opportunityId,
 assigneeFirstName, assigneeLastName, assigneeEmail, businessTitle);

     ReferenceCodesUtils.formatOutput(response);
    }
```

```
 static AssignOpportunityResponse getResponse(String opportunityId, String
 assigneeFirstName, String assigneeLastName, String assigneeEmail, String
 businessTitle) {

  AssignOpportunityRequest assignOpportunityRequest =
 AssignOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(opportunityId)
          .assignee(AssigneeContact.builder()
            .firstName(assigneeFirstName)
            .lastName(assigneeLastName)
            .email(assigneeEmail)
            .businessTitle(businessTitle)
            .build())
          .build();

        AssignOpportunityResponse response =
 client.assignOpportunity(assignOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte AssignOpportunitya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Reatribua uma oportunidade existente a outro usuário.

```
#!/usr/bin/env python

"""
Purpose
PC-API-07 Assigning a new owner
"""
import logging
import boto3
import utils.helpers as helper
```

```
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def assign_opportunity(identifier):
    assign_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "Identifier": identifier,
        "Assignee": {
            "BusinessTitle": "OpportunityOwner",
            "Email": "test@test.com",
            "FirstName": "John",
            "LastName": "Doe"
        }
    }
    try:
        # Perform an API call
        response =
 partner_central_client.assign_opportunity(**assign_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    identifier = "O4236468"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Assigning a new owner to an opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(assign_opportunity(identifier))

if __name__ == "__main__":
```

```
    usage_demo()
```

- Para obter detalhes da API, consulte a AssignOpportunityReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`AssociateOpportunity`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `AssociateOpportunity`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação em contexto no seguinte exemplo de código:

- Atualizar a entidade associada de uma oportunidade

Java

SDK para Java 2.x

Crie uma associação formal entre uma oportunidade e várias entidades relacionadas.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityResponse
```

```
/*
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
*/

public class AssociateOpportunity {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     String entityType = "Solutions";

     String entityIdentifier = "S-0000000";

     AssociateOpportunityResponse response = getResponse(opportunityId,
 entityType, entityIdentifier );

     ReferenceCodesUtils.formatOutput(response);
    }

static AssociateOpportunityResponse getResponse(String opportunityId, String
 entityType, String entityIdentifier) {

        AssociateOpportunityRequest associateOpportunityRequest =
 AssociateOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .opportunityIdentifier(opportunityId)
          .relatedEntityType(entityType)
          .relatedEntityIdentifier(entityIdentifier)
          .build();

        AssociateOpportunityResponse response =
 client.associateOpportunity(associateOpportunityRequest);
```

```
        return response;
 }
}
```

- Para obter detalhes da API, consulte AssociateOpportunitya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Crie uma associação formal entre uma oportunidade e várias entidades relacionadas.

```
#!/usr/bin/env python

"""
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
"""
import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def associate_opportunity(entity_type, entity_identifier, opportunityIdentifier):
    associate_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : entity_type,
```

```
        "RelatedEntityIdentifier" : entity_identifier
    }
    try:
        # Perform an API call
        response =
 partner_central_client.associate_opportunity(**associate_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    #entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
    entity_type = "Solutions"
    entity_identifier = "S-0059717"
    opportunityIdentifier = "O5465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Associate Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(associate_opportunity(entity_type,
 entity_identifier, opportunityIdentifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a AssociateOpportunityReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`CreateOpportunity`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `CreateOpportunity`.

.NET

SDK para .NET

Crie uma oportunidade.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// PDX-License-Identifier: Apache-2.0

using System;
using Newtonsoft.Json;
using Amazon;
using Amazon.Runtime;
using Amazon.PartnerCentralSelling;
using Amazon.PartnerCentralSelling.Model;

namespace AWSExample
{
    class Program
    {
        static readonly string catalogToUse = "AWS";
        static async Task Main(string[] args)
        {
            // Initialize credentials from .aws/credentials file
            var credentials = new
 Amazon.Runtime.CredentialManagement.SharedCredentialsFile();
            if (credentials.TryGetProfile("default", out var profile))
            {
                AWSCredentials awsCredentials =
 profile.GetAWSCredentials(credentials);

                var client = new
 AmazonPartnerCentralSellingClient(awsCredentials);

                var request = new CreateOpportunityRequest
                {
                    Catalog = catalogToUse,
                    Origin = "Partner Referral",
                    Customer = new Customer
                    {
                        Account = new Account
                        {
                            Address = new Address
                            {
```

```
                                CountryCode = "US",
                                PostalCode = "99502",
                                StateOrRegion = "Alaska"
                            },
                            CompanyName = "TestCompanyName",
                            Duns = "123456789",
                            WebsiteUrl = "www.test.io",
                            Industry = "Automotive"
                        },
                        Contacts = new List<Contact>
                        {
                            new Contact
                            {
                                Email = "test@test.io",
                                FirstName = "John  ",
                                LastName = "Doe",
                                Phone = "+14444444444",
                                BusinessTitle = "test title"
                            }
                        }
                    },
                    LifeCycle = new LifeCycle
                    {
                        ReviewStatus = "Submitted",
                        TargetCloseDate = "2024-12-30"
                    },
                    Marketing = new Marketing
                    {
                        Source = "None"
                    },
                    OpportunityType = "Net New Business",
                    PrimaryNeedsFromAws = new List<string> { "Co-Sell -
Architectural Validation" },
                    Project = new Project
                    {
                        Title = "Moin Test UUID",
                        CustomerBusinessProblem = "Sandbox is not working as
expected",
                        CustomerUseCase = "AI Machine Learning and Analytics",
                        DeliveryModels = new List<string> { "SaaS or PaaS" },
                        ExpectedCustomerSpend = new List<ExpectedCustomerSpend>
                        {
                            new ExpectedCustomerSpend
                            {
```

```
                                Amount = "2000.0",
                                CurrencyCode = "USD",
                                Frequency = "Monthly",
                                TargetCompany = "Ibexlabs"
                            }
                        },
                        SalesActivities = new List<string> { "Initialized
 discussions with customer" }
                    }
                };

                try
                {
                    var response = await client.CreateOpportunityAsync(request);
                    Console.WriteLine(response.HttpStatusCode);
                    string formattedJson = JsonConvert.SerializeObject(response,
 Formatting.Indented);
                    Console.WriteLine(formattedJson);
                }
                catch (ValidationException ex)
                {
                    Console.WriteLine("Validation error: " + ex.Message);
                }
                catch (AmazonPartnerCentralSellingException e)
                {
                    Console.WriteLine("Failed:");
                    Console.WriteLine(e.RequestId);
                    Console.WriteLine(e.ErrorCode);
                    Console.WriteLine(e.Message);
                }
            }
            else
            {
                Console.WriteLine("Profile not found.");
            }
        }
    }
}
```

- Para obter detalhes da API, consulte CreateOpportunitya Referência AWS SDK para .NET da API.

Java

SDK para Java 2.x

Crie uma oportunidade.

```
package org.example;

import java.time.Instant;
import java.util.ArrayList;
import java.util.List;

import static org.example.utils.Constants.*;

import org.example.entity.Root;
import org.example.utils.ReferenceCodesUtils;
import org.example.utils.StringSerializer;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import software.amazon.awssdk.services.partnercentralselling.model.Account;
import software.amazon.awssdk.services.partnercentralselling.model.Address;
import software.amazon.awssdk.services.partnercentralselling.model.Contact;
import
 software.amazon.awssdk.services.partnercentralselling.model.CreateOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.CreateOpportunityResponse;
import software.amazon.awssdk.services.partnercentralselling.model.Customer;
import
 software.amazon.awssdk.services.partnercentralselling.model.ExpectedCustomerSpend;
import software.amazon.awssdk.services.partnercentralselling.model.LifeCycle;
import software.amazon.awssdk.services.partnercentralselling.model.Marketing;
import software.amazon.awssdk.services.partnercentralselling.model.MonetaryValue;
import
 software.amazon.awssdk.services.partnercentralselling.model.NextStepsHistory;
import software.amazon.awssdk.services.partnercentralselling.model.Project;
import
 software.amazon.awssdk.services.partnercentralselling.model.SoftwareRevenue;

import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
```

```
import com.google.gson.ToNumberPolicy;

public class CreateOpportunity {

 static final Gson GSON = new GsonBuilder()
   .setObjectToNumberStrategy(ToNumberPolicy.LAZILY_PARSED_NUMBER)
   .registerTypeAdapter(String.class, new StringSerializer())
   .create();

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

 public static void main(String[] args) {

  String inputFile = "CreateOpportunity2.json";

  if (args.length > 0)
   inputFile = args[0];

  CreateOpportunityResponse response = createOpportunity(inputFile);

  client.close();
 }

 static CreateOpportunityResponse createOpportunity(String inputFile) {

  String inputString = ReferenceCodesUtils.readInputFileToString(inputFile);

  Root root = GSON.fromJson(inputString, Root.class);

  List<NextStepsHistory> nextStepsHistories = new ArrayList<NextStepsHistory>();
  if ( root.lifeCycle != null && root.lifeCycle.nextStepsHistories != null) {
   for (org.example.entity.NextStepsHistory nextStepsHistoryJson :
 root.lifeCycle.nextStepsHistories) {
    NextStepsHistory nextStepsHistory = NextStepsHistory.builder()
      .time(Instant.parse(nextStepsHistoryJson.time))
      .value(nextStepsHistoryJson.value)
                 .build();
    nextStepsHistories.add(nextStepsHistory);
   }
```

```
 }

 LifeCycle lifeCycle = null;
 if ( root.lifeCycle != null ) {
  lifeCycle = LifeCycle.builder()
   .closedLostReason(root.lifeCycle.closedLostReason)
   .nextSteps(root.lifeCycle.nextSteps)
   .nextStepsHistory(nextStepsHistories)
   .reviewComments(root.lifeCycle.reviewComments)
   .reviewStatus(root.lifeCycle.reviewStatus)
   .reviewStatusReason(root.lifeCycle.reviewStatusReason)
   .stage(root.lifeCycle.stage)
   .targetCloseDate(root.lifeCycle.targetCloseDate)
   .build();
 }

 Marketing marketing = null;
 if ( root.marketing != null ) {
  marketing = Marketing.builder()
    .awsFundingUsed(root.marketing.awsFundingUsed)
    .campaignName(root.marketing.campaignName)
    .channels(root.marketing.channels)
    .source(root.marketing.source)
    .useCases(root.marketing.useCases)
    .build();

 }

 Address address = null;
 if ( root.customer != null && root.customer.account != null &&
root.customer.account.address != null ) {
  address = Address.builder()
   .city(root.customer.account.address.city)
               .postalCode(root.customer.account.address.postalCode)
               .stateOrRegion(root.customer.account.address.stateOrRegion)
               .countryCode(root.customer.account.address.countryCode)
               .streetAddress(root.customer.account.address.streetAddress)
               .build();
 }

 Account account = null;
 if ( root.customer != null && root.customer.account!= null) {
  account = Account.builder()
            .address(address)
```

```
            .awsAccountId(root.customer.account.awsAccountId)
               .duns(root.customer.account.duns)
               .industry(root.customer.account.industry)
               .otherIndustry(root.customer.account.otherIndustry)
               .companyName(root.customer.account.companyName)
               .websiteUrl(root.customer.account.websiteUrl)
               .build();
 }

 List<Contact> contacts = new ArrayList<Contact>();
 if ( root.customer != null && root.customer.contacts != null) {
  for (org.example.entity.Contact jsonContact : root.customer.contacts) {
   Contact contact = Contact.builder()
                .email(jsonContact.email)
                .firstName(jsonContact.firstName)
                .lastName(jsonContact.lastName)
                .phone(jsonContact.phone)
                .businessTitle(jsonContact.businessTitle)
                .build();
   contacts.add(contact);
  }
 }

 Customer customer = Customer.builder()
   .account(account)
   .contacts(contacts)
   .build();

 Contact oportunityTeamContact = null;
 if (root.opportunityTeam != null && root.opportunityTeam.get(0) != null ) {
  oportunityTeamContact = Contact.builder()
               .firstName(root.opportunityTeam.get(0).firstName)
               .lastName(root.opportunityTeam.get(0).lastName)
               .email(root.opportunityTeam.get(0).email)
               .phone(root.opportunityTeam.get(0).phone)
               .businessTitle(root.opportunityTeam.get(0).businessTitle)
               .build();
 }

 List<ExpectedCustomerSpend> expectedCustomerSpends = new
ArrayList<ExpectedCustomerSpend>();
 if ( root.project != null && root.project.expectedCustomerSpend != null) {
  for (org.example.entity.ExpectedCustomerSpend expectedCustomerSpendJson :
root.project.expectedCustomerSpend) {
```

```
    ExpectedCustomerSpend expectedCustomerSpend = null;
    expectedCustomerSpend = ExpectedCustomerSpend.builder()
      .amount(expectedCustomerSpendJson.amount)
      .currencyCode(expectedCustomerSpendJson.currencyCode)
      .frequency(expectedCustomerSpendJson.frequency)
      .targetCompany(expectedCustomerSpendJson.targetCompany)
      .build();
    expectedCustomerSpends.add(expectedCustomerSpend);
   }
        }

        Project project = null;
        if ( root.project != null) {
         project = Project.builder()
                .title(root.project.title)
                .customerBusinessProblem(root.project.customerBusinessProblem)
                .customerUseCase(root.project.customerUseCase)
                .deliveryModels(root.project.deliveryModels)
                .expectedCustomerSpend(expectedCustomerSpends)
                .salesActivities(root.project.salesActivities)
                .competitorName(root.project.competitorName)
                .otherSolutionDescription(root.project.otherSolutionDescription)
                .build();
        }

        SoftwareRevenue softwareRevenue = null;
        if ( root.softwareRevenue != null) {
         MonetaryValue monetaryValue = null;
         if ( root.softwareRevenue.value != null) {
          monetaryValue = MonetaryValue.builder()
             .amount(root.softwareRevenue.value.amount)
             .currencyCode(root.softwareRevenue.value.currencyCode)
             .build();
         }
         softwareRevenue = SoftwareRevenue.builder()
           .deliveryModel(root.softwareRevenue.deliveryModel)
           .effectiveDate(root.softwareRevenue.effectiveDate)
           .expirationDate(root.softwareRevenue.expirationDate)
           .value(monetaryValue)
           .build();
        }

  // Building the Actual CreateOpportunity Request
```

```
  CreateOpportunityRequest createOpportunityRequest =
 CreateOpportunityRequest.builder()
    .catalog(CATALOG_TO_USE)
    .clientToken(root.clientToken)
    .primaryNeedsFromAwsWithStrings(root.primaryNeedsFromAws)
    .opportunityType(root.opportunityType)
    .lifeCycle(lifeCycle)
    .marketing(marketing)
    .nationalSecurity(root.nationalSecurity)
    .origin(root.origin)
    .customer(customer)
    .project(project)
    .partnerOpportunityIdentifier(root.partnerOpportunityIdentifier)
    .opportunityTeam(oportunityTeamContact)
    .softwareRevenue(softwareRevenue)
    .build();

  CreateOpportunityResponse response =
 client.createOpportunity(createOpportunityRequest);
  System.out.println("Successfully created: " + response);

  return response;
    }

}
```

- Para obter detalhes da API, consulte CreateOpportunitya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Crie uma oportunidade.

```
#!/usr/bin/env python
import boto3
import logging
import sys
import os
sys.path.append(os.path.dirname(os.path.dirname(os.path.abspath(__file__))))
```

```
import utils.helpers as helper
import utils.stringify_details as sd
from botocore.client import ClientError
from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

def create_opportunity(partner_central_client):
    create_opportunity_request = helper.remove_nulls(sd.stringify_json("src/
create_opportunity/createOpportunity.json"))
    try:
        # Perform an API call
        response =
 partner_central_client.create_opportunity(**create_opportunity_request)

        helper.pretty_print_datetime(response)

        # Retrieve the opportunity details
        get_response = partner_central_client.get_opportunity(
            Identifier=response["Id"],
            Catalog=CATALOG_TO_USE
        )
        helper.pretty_print_datetime(get_response)
        return response
    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Create Opportunity.")
    print("-" * 88)

    partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
    )

    create_opportunity(partner_central_client)

if __name__ == "__main__":
```

```
    usage_demo()
```

- Para obter detalhes da API, consulte a CreateOpportunityReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`DisassociateOpportunity`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `DisassociateOpportunity`.

Exemplos de ações são trechos de código de programas maiores e devem ser executados em contexto. É possível ver essa ação em contexto no seguinte exemplo de código:

- Atualizar a entidade associada de uma oportunidade

Java

SDK para Java 2.x

Remova uma associação existente entre uma oportunidade e entidades relacionadas.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityReque
import
 software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityRespo

/*
```

```
Purpose
PC-API -14 Removing a Solution
PC-API -15 Removing an offer
PC-API -16 Removing a product
entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
*/

public class DisassociateOpportunity {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     String entityType = "Solutions";

     String entityIdentifier = "S-0000000";

     DisassociateOpportunityResponse response = getResponse(opportunityId,
 entityType, entityIdentifier );

     ReferenceCodesUtils.formatOutput(response);
    }

 static DisassociateOpportunityResponse getResponse(String opportunityId, String
 entityType, String entityIdentifier) {

  DisassociateOpportunityRequest disassociateOpportunityRequest =
 DisassociateOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .opportunityIdentifier(opportunityId)
          .relatedEntityType(entityType)
          .relatedEntityIdentifier(entityIdentifier)
          .build();

        DisassociateOpportunityResponse response =
 client.disassociateOpportunity(disassociateOpportunityRequest);
```

```
        return response;
 }
}
```

- Para obter detalhes da API, consulte DisassociateOpportunitya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Remova uma associação existente entre uma oportunidade e entidades relacionadas.

```
#!/usr/bin/env python

"""
Purpose
PC-API -14 Removing a Solution
PC-API -15 Removing an offer
PC-API -16 Removing a product
"""
import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def disassociate_opportunity(entity_type, entity_identifier,
 opportunityIdentifier):
    disassociate_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : entity_type,
```

```
        "RelatedEntityIdentifier" : entity_identifier
    }
    try:
        # Perform an API call
        response =
 partner_central_client.disassociate_opportunity(**disassociate_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    #entity_type = Solutions | AWSProducts | AWSMarketplaceOffers
    entity_type = "Solutions"
    entity_identifier = "S-0049999"
    opportunityIdentifier = "O4397574"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get updated Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(disassociate_opportunity(entity_type,
 entity_identifier, opportunityIdentifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a DisassociateOpportunityReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`GetAwsOpportunitySummary`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `GetAwsOpportunitySummary`.

Java

SDK para Java 2.x

Recupera um resumo de uma AWS oportunidade.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.GetAwsOpportunitySummaryRequ
import
 software.amazon.awssdk.services.partnercentralselling.model.GetAwsOpportunitySummaryResp

/*
 * Purpose
 * PC-API-25 Retrieves a summary of an AWS Opportunity.
 */

public class GetAwsOpportunitySummary {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     GetAwsOpportunitySummaryResponse response = getResponse(opportunityId);

     ReferenceCodesUtils.formatOutput(response);
```

```
    }

 public static GetAwsOpportunitySummaryResponse getResponse(String opportunityId)
 {

  GetAwsOpportunitySummaryRequest getOpportunityRequest =
 GetAwsOpportunitySummaryRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .relatedOpportunityIdentifier(opportunityId)
          .build();

  GetAwsOpportunitySummaryResponse response =
 client.getAwsOpportunitySummary(getOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte GetAwsOpportunitySummarya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Recupera um resumo de uma AWS oportunidade.

```
#!/usr/bin/env python

"""
Purpose
PC-API-25 Retrieves a summary of an AWS Opportunity.
 LifeCycle.ReviewStatus=Approved
"""
import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE
```

```
serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "RelatedOpportunityIdentifier": identifier
    }
    try:
        # Perform an API call
        response =
 partner_central_client.get_aws_opportunity_summary(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    identifier = "O5465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get AWS Opportunity summary.")
    print("-" * 88)

    helper.pretty_print_datetime(get_opportunity(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a GetAwsOpportunitySummaryReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`GetEngagementInvitation`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `GetEngagementInvitation`.

Java

SDK para Java 2.x

Recupera os detalhes de um convite de engajamento compartilhado AWS por um parceiro.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationReque
import
 software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationRespo

/*
 * Purpose
 * PC-API-22 Get engagement invitation opportunity
 */

public class GetEngagementInvitation {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
```

```
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     GetEngagementInvitationResponse response = getResponse(opportunityId);

     ReferenceCodesUtils.formatOutput(response);
    }

 static GetEngagementInvitationResponse getResponse(String opportunityId) {

  GetEngagementInvitationRequest getOpportunityRequest =
 GetEngagementInvitationRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(opportunityId)
          .build();

  GetEngagementInvitationResponse response =
 client.getEngagementInvitation(getOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte GetEngagementInvitationa Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Recupera os detalhes de um convite de engajamento compartilhado AWS por um parceiro.

```
#!/usr/bin/env python

"""
Purpose
PC-API-22  GetOpportunityEngagementInvitation - Retrieves details of a specific
 engagement invitation.
```

```
This operation allows partners to view the invitation and its associated
 information,
such as the customer, project, and lifecycle details.
"""
import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def get_opportunity_engagement_invitation(identifier):
    get_opportunity_engagement_invitation_request ={
        "Catalog": CATALOG_TO_USE,
     "Identifier": identifier
    }
    try:
        # Perform an API call
        response =
 partner_central_client.get_engagement_invitation(**get_opportunity_engagement_invitation
        return response

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))

def usage_demo():
    identifier = "arn:aws:partnercentral-selling:us-east-1:aws:catalog/Sandbox/
engagement-invitation/engi-0000000IS0Qga"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Given the ARN identifier, retrieve details of Opportunity Engagement
 Invitation.")
    print("-" * 88)
```

```
 helper.pretty_print_datetime(get_opportunity_engagement_invitation(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a GetEngagementInvitationReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`GetOpportunity`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `GetOpportunity`.

.NET

SDK para .NET

Tenha uma oportunidade.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// PDX-License-Identifier: Apache-2.0

using System;
using Newtonsoft.Json;
using Amazon;
using Amazon.Runtime;
using Amazon.PartnerCentralSelling;
using Amazon.PartnerCentralSelling.Model;

namespace AWSExample
{
    class Program
    {
        static readonly string catalogToUse = "AWS";
        static readonly string identifier = "O1111111";
        static async Task Main(string[] args)
        {
```

```
            // Initialize credentials from .aws/credentials file
            var credentials = new
 Amazon.Runtime.CredentialManagement.SharedCredentialsFile();
            if (credentials.TryGetProfile("default", out var profile))
            {
                AWSCredentials awsCredentials =
 profile.GetAWSCredentials(credentials);

                var client = new
 AmazonPartnerCentralSellingClient(awsCredentials);

                var request = new GetOpportunityRequest
                {
                    Catalog = catalogToUse,
                    Identifier = identifier
                };

                try {
                    var response = await client.GetOpportunityAsync(request);
                    Console.WriteLine(response.HttpStatusCode);
                    string formattedJson = JsonConvert.SerializeObject(response,
 Formatting.Indented);
                    Console.WriteLine(formattedJson);
                } catch(ValidationException ex) {
                    Console.WriteLine("Validation error: " + ex.Message);
                } catch (AmazonPartnerCentralSellingException e) {
                    Console.WriteLine("Failed:");
                    Console.WriteLine(e.RequestId);
                    Console.WriteLine(e.ErrorCode);
                    Console.WriteLine(e.Message);
                }
            }
            else
            {
                Console.WriteLine("Profile not found.");
            }
        }
    }
}
```

- Para obter detalhes da API, consulte GetOpportunitya Referência AWS SDK para .NET da API.

Go

SDK para Go V2

Tenha uma oportunidade.

```
package main

import (
 "context"
 "encoding/json"
 "fmt"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/config"
 "github.com/aws/aws-sdk-go-v2/service/partnercentralselling"
)

func main() {
 config, err := config.LoadDefaultConfig(context.TODO())

 if err != nil {
  log.Fatal(err)
 }

 config.Region = "us-east-1"

 client := partnercentralselling.NewFromConfig(config)

 output, err := client.GetOpportunity(context.TODO(),
 &partnercentralselling.GetOpportunityInput{
  Identifier: aws.String("O1111111"),
  Catalog:    aws.String("AWS"),
 })

 if err != nil {
  log.Fatal(err)
 }
 log.Println("printing opportuniy...\n")

 jsonOutput, err := json.MarshalIndent(output, "", "    ")
```

```
 fmt.Println(string(jsonOutput))
}
```

- Para obter detalhes da API, consulte GetOpportunitya Referência AWS SDK para Go da API.

Java

SDK para Java 2.x

Tenha uma oportunidade.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityResponse;

/*
 * Purpose
 * PC-API-08 Get updated Opportunity
 */

public class GetOpportunity {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();
```

```
    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     GetOpportunityResponse response = getResponse(opportunityId);

     ReferenceCodesUtils.formatOutput(response);
    }

 public static GetOpportunityResponse getResponse(String opportunityId) {

        GetOpportunityRequest getOpportunityRequest =
 GetOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(opportunityId)
          .build();

        GetOpportunityResponse response =
 client.getOpportunity(getOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte GetOpportunitya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Tenha uma oportunidade.

```
#!/usr/bin/env python

"""
Purpose
PC-API -08 Get updated Opportunity given opportunity id
"""
import logging
```

```
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "Identifier": identifier
    }
    try:
        # Perform an API call
        response =
 partner_central_client.get_opportunity(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    identifier = "O5465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get updated Opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(get_opportunity(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a GetOpportunityReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`ListEngagementInvitations`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `ListEngagementInvitations`.

Java

SDK para Java 2.x

Recupera uma lista de convites de engajamento enviados ao parceiro.

```
package org.example;

import java.util.ArrayList;
import java.util.List;

import org.example.utils.ReferenceCodesUtils;
import static org.example.utils.Constants.*;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.ListEngagementInvitationsReq
import
 software.amazon.awssdk.services.partnercentralselling.model.ListEngagementInvitationsRes
import
 software.amazon.awssdk.services.partnercentralselling.model.ParticipantType;
import
 software.amazon.awssdk.services.partnercentralselling.model.EngagementInvitationSummary;

public class ListEngagementInvitations {
```

```
 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

     List<EngagementInvitationSummary> opportunitySummaries = getResponse();
        ReferenceCodesUtils.formatOutput(opportunitySummaries);
    }

    static List<EngagementInvitationSummary> getResponse() {

  List<EngagementInvitationSummary> opportunitySummaries = new
 ArrayList<EngagementInvitationSummary>();

  ListEngagementInvitationsRequest listOpportunityRequest =
 ListEngagementInvitationsRequest.builder()
                .catalog(CATALOG_TO_USE)
                .participantType(ParticipantType.RECEIVER)
          .maxResults(5)
          .build();

  ListEngagementInvitationsResponse response =
 client.listEngagementInvitations(listOpportunityRequest);

     opportunitySummaries.addAll(response.engagementInvitationSummaries());

     client.close();

        return opportunitySummaries;
  }
 }
```

- Para obter detalhes da API, consulte ListEngagementInvitationsa Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Recupera uma lista de convites de engajamento enviados ao parceiro.

```
#!/usr/bin/env python

"""
Purpose
PC-API-21 ListEngagementInvitations - Retrieves a list of engagement invitations
 based on specified criteria.
This operation allows partners to view all invitations to engagement.
"""
import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def list_engagement_invitations():
    list_engagement_invitations_request ={
        "Catalog": CATALOG_TO_USE,
        "MaxResults": 20
    }
    try:
        # Perform an API call
        response =
 partner_central_client.list_engagement_invitations(**list_engagement_invitations_request
        return response

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))
```

```
def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Retrieve list of Engagement Invitations.")
    print("-" * 88)

    helper.pretty_print_datetime(list_engagement_invitations())

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a ListEngagementInvitationsReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`ListOpportunities`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `ListOpportunities`.

.NET

SDK para .NET

Liste oportunidades.

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// PDX-License-Identifier: Apache-2.0

using System;
using Newtonsoft.Json;
using Amazon;
using Amazon.Runtime;
using Amazon.PartnerCentralSelling;
using Amazon.PartnerCentralSelling.Model;

namespace AWSExample
{
```

```
    class Program
    {
        static readonly string catalogToUse = "Sandbox";
        static async Task Main(string[] args)
        {
            // Initialize credentials from .aws/credentials file
            var credentials = new
 Amazon.Runtime.CredentialManagement.SharedCredentialsFile();
            if (credentials.TryGetProfile("default", out var profile))
            {
                AWSCredentials awsCredentials =
 profile.GetAWSCredentials(credentials);

                //var config = new AmazonPartnerCentralSellingConfig()
                //{
                //    ServiceURL = "https://partnercentral-selling.us-
east-1.api.aws",
                //};
                //var client = new
 AmazonPartnerCentralSellingClient(awsCredentials, config);
                var client = new
 AmazonPartnerCentralSellingClient(awsCredentials);
                var request = new ListOpportunitiesRequest
                {
                    Catalog = catalogToUse,
                    MaxResults = 2
                };

                try {
                    var response = await client.ListOpportunitiesAsync(request);
                    Console.WriteLine(response.HttpStatusCode);
                    foreach (var opportunity in response.OpportunitySummaries)
                    {
                        Console.WriteLine("Opportunity id: " + opportunity.Id);
                    }
                    string formattedJson =
 JsonConvert.SerializeObject(response.OpportunitySummaries, Formatting.Indented);
                    Console.WriteLine(formattedJson);
                } catch(ValidationException ex) {
                    Console.WriteLine("Validation error: " + ex.Message);
                } catch (AmazonPartnerCentralSellingException e) {
                    Console.WriteLine("Failed:");
                    Console.WriteLine(e.RequestId);
                    Console.WriteLine(e.ErrorCode);
```

```
                    Console.WriteLine(e.Message);
                }
            }
            else
            {
                Console.WriteLine("Profile not found.");
            }
        }
    }
}
```

- Para obter detalhes da API, consulte ListOpportunitiesa Referência AWS SDK para .NET da API.

Go

SDK para Go V2

Liste oportunidades.

```
package main

import (
 "context"
 "encoding/json"
 "fmt"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/config"
 "github.com/aws/aws-sdk-go-v2/service/partnercentralselling"
)

func main() {
 config, err := config.LoadDefaultConfig(context.TODO())

 if err != nil {
  log.Fatal(err)
 }

 config.Region = "us-east-1"
```

```
 client := partnercentralselling.NewFromConfig(config)

 output, err := client.ListOpportunities(context.TODO(),
 &partnercentralselling.ListOpportunitiesInput{
  MaxResults: aws.Int32(2),
  Catalog:    aws.String("AWS"),
 })

 if err != nil {
  log.Fatal(err)
 }

 jsonOutput, err := json.MarshalIndent(output, "", "    ")
 fmt.Println(string(jsonOutput))
}
```

- Para obter detalhes da API, consulte ListOpportunitiesa Referência AWS SDK para Go da API.

Java

SDK para Java 2.x

Liste oportunidades.

```
package org.example;

import java.util.ArrayList;
import java.util.List;

import org.example.utils.ReferenceCodesUtils;
import static org.example.utils.Constants.*;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.ListOpportunitiesRequest;
```

```
import
 software.amazon.awssdk.services.partnercentralselling.model.ListOpportunitiesResponse;
import
 software.amazon.awssdk.services.partnercentralselling.model.OpportunitySummary;

/*
 * Purpose
 * PC-API-18 Getting list of Opportunities
 */

public class ListOpportunititesPaging {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {
     List<OpportunitySummary> opportunitySummaries = getResponse();
        ReferenceCodesUtils.formatOutput(opportunitySummaries);
    }

    private static List<OpportunitySummary> getResponse() {
     List<OpportunitySummary> opportunitySummaries = new
 ArrayList<OpportunitySummary>();

  ListOpportunitiesRequest listOpportunityRequest =
 ListOpportunitiesRequest.builder()
                .catalog(CATALOG_TO_USE)
          .maxResults(5)
          .build();

     ListOpportunitiesResponse response =
 client.listOpportunities(listOpportunityRequest);

     opportunitySummaries.addAll(response.opportunitySummaries());

     while (response.nextToken() != null && response.nextToken().length() > 0) {
      listOpportunityRequest = ListOpportunitiesRequest.builder()
                  .catalog(CATALOG_TO_USE)
              .maxResults(5)
              .nextToken(response.nextToken())
```

```
                .build();
       response = client.listOpportunities(listOpportunityRequest);
       opportunitySummaries.addAll(response.opportunitySummaries());
      }

      client.close();

         return opportunitySummaries;
 }
}
```

- Para obter detalhes da API, consulte ListOpportunitiesa Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Liste oportunidades.

```
#!/usr/bin/env python

"""
Purpose
PC-API -18 Getting list of Opportunities
"""
import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def get_list_of_opportunities():
```

```
    opportunity_list = []

    list_opportunities_request ={
        "Catalog": CATALOG_TO_USE,
     "MaxResults": 20
    }
    try:
        # Perform an API call
        response =
 partner_central_client.list_opportunities(**list_opportunities_request)
        opportunity_list.extend(response["OpportunitySummaries"])

        while "NextToken" in response and response["NextToken"] is not None:
            list_opportunities_request["NextToken"] = response["NextToken"]
            response =
 partner_central_client.list_opportunities(**list_opportunities_request)
            opportunity_list.extend(response["OpportunitySummaries"])

        return opportunity_list

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Getting list of Opportunities.")
    print("-" * 88)

    helper.pretty_print_datetime(get_list_of_opportunities())

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a ListOpportunitiesReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`ListSolutions`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `ListSolutions`.

Java

SDK para Java 2.x

Recupera uma lista de soluções de parceiros que o parceiro registrou na Central de Parceiros.

```
package org.example;

import java.util.ArrayList;
import java.util.List;

import static org.example.utils.Constants.*;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.ListSolutionsRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.ListSolutionsResponse;
import software.amazon.awssdk.services.partnercentralselling.model.SolutionBase;

/*
 * Purpose
 * PC-API-10 Getting list of solutions
 */

public class ListSolutions {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
```

```
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {
     List<SolutionBase> solutionSummaries = getResponse();
        ReferenceCodesUtils.formatOutput(solutionSummaries);
    }

    static List<SolutionBase> getResponse() {
  List<SolutionBase> solutionSummaries = new ArrayList<SolutionBase>();

  ListSolutionsRequest listSolutionsRequest = ListSolutionsRequest.builder()
    .catalog(CATALOG_TO_USE)
          .maxResults(5)
          .build();

     ListSolutionsResponse response = client.listSolutions(listSolutionsRequest);

     solutionSummaries.addAll(response.solutionSummaries());

        return solutionSummaries;
 }
}
```

- Para obter detalhes da API, consulte ListSolutionsa Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Recupera uma lista de soluções de parceiros que o parceiro registrou na Central de Parceiros.

```
#!/usr/bin/env python

"""
Purpose
PC-API-10 Getting list of solutions
"""
import logging
```

```
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def get_list_of_solutions():
    list_solutions_request ={
        "Catalog": CATALOG_TO_USE,
     "MaxResults": 20
    }
    try:
        # Perform an API call
        response =
 partner_central_client.list_solutions(**list_solutions_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Getting list of solutions.")
    print("-" * 88)

    helper.pretty_print_datetime(get_list_of_solutions())

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a ListSolutionsReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`RejectEngagementInvitation`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `RejectEngagementInvitation`.

Java

SDK para Java 2.x

Rejeita e EngagementInvitation isso éAWS compartilhado.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.RejectEngagementInvitationRe
import
 software.amazon.awssdk.services.partnercentralselling.model.RejectEngagementInvitationRe

/*
 * Purpose
 * PC-API-05 AWS Originated(AO) rejection
 */

public class RejectEngagementInvitation {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
```

```
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

  RejectEngagementInvitationResponse response = getResponse(opportunityId);

     ReferenceCodesUtils.formatOutput(response);
    }

 static RejectEngagementInvitationResponse getResponse(String invitationId) {

        RejectEngagementInvitationRequest rejectOpportunityRequest =
 RejectEngagementInvitationRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(invitationId)
          .rejectionReason("Unable to support")
          .build();

  RejectEngagementInvitationResponse response =
 client.rejectEngagementInvitation(rejectOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte RejectEngagementInvitationa Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Rejeita e EngagementInvitation isso éAWS compartilhado.

```
#!/usr/bin/env python

"""
Purpose
```

```
PC-API-05 AWS Originated AO rejection - RejectOpportunityEngagementInvitation -
 Rejects a engagement invitation.
This action indicates that the partner does not wish to participate in the
 engagement and
provides a reason for the rejection.
Upon rejection, a OpportunityEngagementInvitationRejected event is triggered.
Subsequently, the invitation will no longer be available for the partner to act
 on.
"""
import json
import logging
import boto3
import utils.helpers as helper

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def reject_opportunity_engagement_invitation(identifier, reject_reason):
    reject_opportunity_engagement_invitation_request ={
        "Catalog": CATALOG_TO_USE,
     "Identifier": identifier,
        "RejectionReason": reject_reason
    }
    try:
        # Perform an API call
        response =
 partner_central_client.reject_engagement_invitation(**reject_opportunity_engagement_invi
        return response

    except Exception as err:
        # Catch all client exceptions
        print(json.dumps(err.response))

def usage_demo():
    identifier = "arn:aws:partnercentral:us-east-1::catalog/Sandbox/engagement-
invitation/engi-0000002isviga"
    reject_reason = "Customer problem unclear"
```

```
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Given the ARN identifier and reject reason, reject the Opportunity
 Engagement Invitation.")
    print("-" * 88)


 helper.pretty_print_datetime(reject_opportunity_engagement_invitation(identifier,
 reject_reason))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a RejectEngagementInvitationReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`StartEngagementByAcceptingInvitationTask`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `StartEngagementByAcceptingInvitationTask`.

Java

SDK para Java 2.x

Inicia o noivado aceitando um EngagementInvitation.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
```

```
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.StartEngagementByAcceptingIn
import
 software.amazon.awssdk.services.partnercentralselling.model.StartEngagementByAcceptingIn
import
 software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationReque
import
 software.amazon.awssdk.services.partnercentralselling.model.GetEngagementInvitationRespo
import
 software.amazon.awssdk.services.partnercentralselling.model.InvitationStatus;

/*
Purpose
PC-API-04: Start Engagement By Accepting InvitationTask for AWS Originated(AO)
 opportunity
*/

public class StartEngagementByAcceptingInvitationTask {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

 static String clientToken = "test-a30d161";

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     StartEngagementByAcceptingInvitationTaskResponse response =
 getResponse(opportunityId);

     if ( response == null) {
      System.out.println("Opportunity is not AWS Originated.");
     } else {
      ReferenceCodesUtils.formatOutput(response);
     }
    }
```

```
   private static GetEngagementInvitationResponse getInvitation(String
invitationId) {

    GetEngagementInvitationRequest getRequest =
GetEngagementInvitationRequest.builder()
         .catalog(Constants.CATALOG_TO_USE)
         .identifier(invitationId)
         .build();

 GetEngagementInvitationResponse response =
client.getEngagementInvitation(getRequest);

       return response;
}

static StartEngagementByAcceptingInvitationTaskResponse getResponse(String
invitationId) {

 if ( getInvitation(invitationId).status().equals(InvitationStatus.PENDING)) {
  StartEngagementByAcceptingInvitationTaskRequest acceptOpportunityRequest =
    StartEngagementByAcceptingInvitationTaskRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(invitationId)
          .clientToken(clientToken)
          .build();

  StartEngagementByAcceptingInvitationTaskResponse response =
client.startEngagementByAcceptingInvitationTask(acceptOpportunityRequest);
        return response;
 }
 return null;
}
}
```

- Para obter detalhes da API, consulte StartEngagementByAcceptingInvitationTaska Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Inicia o noivado aceitando um EngagementInvitation.

```
#!/usr/bin/env python

"""
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
"""
import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request ={
     "Identifier": identifier,
        "Catalog": CATALOG_TO_USE
    }
    try:
        # Perform an API call
        response =
 partner_central_client.get_engagement_invitation(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)
```

```
def start_engagement_by_accepting_invitation_task(identifier):

    response = get_opportunity(identifier)

    if ( response['Status'] == 'PENDING') :
        accept_opportunity_engagement_invitation_request ={
            "Catalog": CATALOG_TO_USE,
         "Identifier" : identifier,
            "ClientToken": "test-123456"
        }
        try:
            # Perform an API call
            response =
 partner_central_client.start_engagement_by_accepting_invitation_task(**accept_opportunit
            return response

        except ClientError as err:
            # Catch all client exceptions
            print(err.response)
            return None
    else:
        return None

def usage_demo():
    identifier = "arn:aws:partnercentral:us-east-1::catalog/Sandbox/engagement-
invitation/engi-0000002isusga"
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Get updated Opportunity.")
    print("-" * 88)


 helper.pretty_print_datetime(start_engagement_by_accepting_invitation_task(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a [StartEngagementByAcceptingInvitationTask](#)Referência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`StartEngagementFromOpportunityTask`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `StartEngagementFromOpportunityTask`.

Java

SDK para Java 2.x

Inicia o processo de engajamento com base em uma oportunidade existente aceitando o convite de engajamento e criando uma oportunidade correspondente no sistema do parceiro.

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import software.amazon.awssdk.services.partnercentralselling.model.AwsSubmission;
import
 software.amazon.awssdk.services.partnercentralselling.model.SalesInvolvementType;
import
 software.amazon.awssdk.services.partnercentralselling.model.StartEngagementFromOpportuni
import
 software.amazon.awssdk.services.partnercentralselling.model.StartEngagementFromOpportuni
import software.amazon.awssdk.services.partnercentralselling.model.Visibility;

/*
 * Purpose
 * PC-API-01 Partner Originated (PO) opp submission(Start Engagement From
 Opportunity Task for AO Originated Opportunity)
 */
```

```
public class StartEngagementFromOpportunityTask {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

    public static void main(String[] args) {

     String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

     StartEngagementFromOpportunityTaskResponse response =
 getResponse(opportunityId);

     ReferenceCodesUtils.formatOutput(response);
    }

 static StartEngagementFromOpportunityTaskResponse getResponse(String
 opportunityId) {

  StartEngagementFromOpportunityTaskRequest submitOpportunityRequest =
 StartEngagementFromOpportunityTaskRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(opportunityId)
          .clientToken("test-annjqwesdsd99")

 .awsSubmission(AwsSubmission.builder().involvementType(SalesInvolvementType.CO_SELL).vis
          .build();

  StartEngagementFromOpportunityTaskResponse response =
 client.startEngagementFromOpportunityTask(submitOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte StartEngagementFromOpportunityTaska Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Inicia o processo de engajamento com base em uma oportunidade existente aceitando o convite de engajamento e criando uma oportunidade correspondente no sistema do parceiro.

```
#!/usr/bin/env python

"""
Purpose
PC-API -11 Associating a product
PC-API -12 Associating a solution
PC-API -13 Associating an offer
"""
import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def start_engagement_from_opportunity_task(identifier):

    start_engagement_from_opportunity_task_request ={
            "AwsSubmission": {
                "InvolvementType": "Co-Sell",
                "Visibility": "Full"
            },
            "Catalog": CATALOG_TO_USE,
         "Identifier" : identifier,
            "ClientToken": "test-annjqwesdsd99"
    }
    try:
            # Perform an API call
```

```
            response =
 partner_central_client.start_engagement_from_opportunity_task(**start_engagement_from_op
            return response

    except ClientError as err:
            # Catch all client exceptions
            print(err.response)
            return None

def usage_demo():
    identifier = "05465588"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Start Engagement from Opportunity Task.")
    print("-" * 88)


 helper.pretty_print_datetime(start_engagement_from_opportunity_task(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a StartEngagementFromOpportunityTaskReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Use **`UpdateOpportunity`**com um AWS SDK

Os exemplos de código a seguir mostram como usar o `UpdateOpportunity`.

Java

SDK para Java 2.x

Atualize uma oportunidade.

```
package org.example;

import java.time.Instant;
import java.util.ArrayList;
import java.util.List;

import static org.example.utils.Constants.*;

import org.example.entity.Root;
import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;
import org.example.utils.StringSerializer;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import software.amazon.awssdk.services.partnercentralselling.model.Account;
import software.amazon.awssdk.services.partnercentralselling.model.Address;
import software.amazon.awssdk.services.partnercentralselling.model.Contact;
import software.amazon.awssdk.services.partnercentralselling.model.Customer;
import
 software.amazon.awssdk.services.partnercentralselling.model.ExpectedCustomerSpend;
import
 software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.GetOpportunityResponse;
import software.amazon.awssdk.services.partnercentralselling.model.LifeCycle;
import software.amazon.awssdk.services.partnercentralselling.model.Marketing;
import
 software.amazon.awssdk.services.partnercentralselling.model.NextStepsHistory;
import software.amazon.awssdk.services.partnercentralselling.model.Project;
import software.amazon.awssdk.services.partnercentralselling.model.ReviewStatus;
import
 software.amazon.awssdk.services.partnercentralselling.model.UpdateOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.UpdateOpportunityResponse;

import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.ToNumberPolicy;
```

```
/*
 * Purpose
 * PC-API-02/06 Update opportunity when LifeCycle.ReviewStatus is not Submitted
 or In-Review
 */

public class UpdateOpportunity {

 static final Gson GSON = new GsonBuilder()
   .setObjectToNumberStrategy(ToNumberPolicy.LAZILY_PARSED_NUMBER)
   .registerTypeAdapter(String.class, new StringSerializer())
   .create();

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

 static String OPPORTUNITY_ORIGIN = ORIGIN_PARTNER_ORIGINATED;

 public static void main(String[] args) {

  String inputFile = "updateOpportunity.json";

  if (args.length > 0)
   inputFile = args[0];

  UpdateOpportunityResponse response = updateOpportunity(inputFile);

  client.close();
 }

 public static GetOpportunityResponse getResponse(String opportunityId) {

        GetOpportunityRequest getOpportunityRequest =
 GetOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .identifier(opportunityId)
          .build();

        GetOpportunityResponse response =
 client.getOpportunity(getOpportunityRequest);
```

```
        System.out.println(opportunityId + ":" + response);
        return response;
}

public static UpdateOpportunityResponse updateOpportunity(String inputFile) {

 String inputString = ReferenceCodesUtils.readInputFileToString(inputFile);

 Root root = GSON.fromJson(inputString, Root.class);
 GetOpportunityResponse response = getResponse(root.identifier);

 if (response != null
   && response.lifeCycle() != null
   && response.lifeCycle().reviewStatus() != null
   && response.lifeCycle().reviewStatus() != ReviewStatus.SUBMITTED
   && response.lifeCycle().reviewStatus() != ReviewStatus.IN_REVIEW) {

  List<NextStepsHistory> nextStepsHistories = new ArrayList<NextStepsHistory>();
  if ( root.lifeCycle != null && root.lifeCycle.nextStepsHistories != null) {
   for (org.example.entity.NextStepsHistory nextStepsHistoryJson :
root.lifeCycle.nextStepsHistories) {
    NextStepsHistory nextStepsHistory = NextStepsHistory.builder()
      .time(Instant.parse(nextStepsHistoryJson.time))
      .value(nextStepsHistoryJson.value)
                  .build();
    nextStepsHistories.add(nextStepsHistory);
   }
  }

  LifeCycle lifeCycle = null;
  if ( root.lifeCycle != null ) {
   lifeCycle = LifeCycle.builder()
    .closedLostReason(root.lifeCycle.closedLostReason)
    .nextSteps(root.lifeCycle.nextSteps)
    .nextStepsHistory(nextStepsHistories)
    .reviewComments(root.lifeCycle.reviewComments)
    .reviewStatus(root.lifeCycle.reviewStatus)
    .reviewStatusReason(root.lifeCycle.reviewStatusReason)
    .stage(root.lifeCycle.stage)
    .targetCloseDate(root.lifeCycle.targetCloseDate)
    .build();
  }

  Marketing marketing = null;
```

```
  if ( root.marketing != null ) {
   marketing = Marketing.builder()
      .awsFundingUsed(root.marketing.awsFundingUsed)
      .campaignName(root.marketing.campaignName)
      .channels(root.marketing.channels)
      .source(root.marketing.source)
      .useCases(root.marketing.useCases)
      .build();

  }

  Address address = null;
  if (root.customer != null && root.customer.account != null &&
root.customer.account.address != null) {
   address =
Address.builder().postalCode(root.customer.account.address.postalCode)
     .stateOrRegion(root.customer.account.address.stateOrRegion)
     .countryCode(root.customer.account.address.countryCode).build();
  }

  Account account = null;
  if (root.customer != null && root.customer.account != null) {
   account = Account.builder().address(address).duns(root.customer.account.duns)

.industry(root.customer.account.industry).companyName(root.customer.account.companyName)
     .websiteUrl(root.customer.account.websiteUrl).build();
  }

  List<Contact> contacts = new ArrayList<Contact>();
  if ( root.customer != null && root.customer.contacts != null) {
   for (org.example.entity.Contact jsonContact : root.customer.contacts) {
    Contact contact = Contact.builder()
                  .email(jsonContact.email)
                  .firstName(jsonContact.firstName)
                  .lastName(jsonContact.lastName)
                  .phone(jsonContact.phone)
                  .businessTitle(jsonContact.businessTitle)
                  .build();
    contacts.add(contact);
   }
  }

  Customer customer =
Customer.builder().account(account).contacts(contacts).build();
```

```
  List<ExpectedCustomerSpend> expectedCustomerSpends = new
ArrayList<ExpectedCustomerSpend>();
  if ( root.project != null && root.project.expectedCustomerSpend != null) {
   for (org.example.entity.ExpectedCustomerSpend expectedCustomerSpendJson :
root.project.expectedCustomerSpend) {
    ExpectedCustomerSpend expectedCustomerSpend = null;
    expectedCustomerSpend = ExpectedCustomerSpend.builder()
      .amount(expectedCustomerSpendJson.amount)
      .currencyCode(expectedCustomerSpendJson.currencyCode)
      .frequency(expectedCustomerSpendJson.frequency)
      .targetCompany(expectedCustomerSpendJson.targetCompany)
      .build();
    expectedCustomerSpends.add(expectedCustomerSpend);
   }
        }

  Project project = null;
  if (root.project != null) {
   project = Project.builder().title(root.project.title)
     .customerBusinessProblem(root.project.customerBusinessProblem)

.customerUseCase(root.project.customerUseCase).deliveryModels(root.project.deliveryModel
     .expectedCustomerSpend(expectedCustomerSpends)

.salesActivities(root.project.salesActivities).competitorName(root.project.competitorNam
     .otherSolutionDescription(root.project.otherSolutionDescription).build();
  }

  // Building the Actual CreateOpportunity Request
  UpdateOpportunityRequest updateOpportunityRequest =
UpdateOpportunityRequest.builder().catalog(root.catalog)

.identifier(root.identifier).lastModifiedDate(Instant.parse(root.lastModifiedDate))

.primaryNeedsFromAwsWithStrings(root.primaryNeedsFromAws).opportunityType(root.opportuni
    .lifeCycle(lifeCycle)
    .customer(customer)
    .project(project)
    .partnerOpportunityIdentifier(root.partnerOpportunityIdentifier)
    .marketing(marketing)
    .nationalSecurity(root.nationalSecurity)
    .opportunityType(root.opportunityType)
    .build();
```

```
   UpdateOpportunityResponse updateResponse =
 client.updateOpportunity(updateOpportunityRequest);
   System.out.println("Successfully updated opportunity: " + updateResponse);

   return updateResponse;
  } else {
   System.out.println("Opportunity cannot be updated.");
   return null;
  }
    }
}
```

- Para obter detalhes da API, consulte UpdateOpportunitya Referência AWS SDK for Java 2.x da API.

Python

SDK para Python (Boto3)

Atualize uma oportunidade.

```
#!/usr/bin/env python

"""
Purpose
PC-API-2  Updating Partner Originated Opportunity
"""
import logging
import boto3
import sys
import os
sys.path.append(os.path.dirname(os.path.dirname(os.path.abspath(__file__))))
import utils.helpers as helper
from botocore.client import ClientError
import utils.stringify_details as sd
from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
```

```
        service_name=serviceName,
        region_name='us-east-1'
)

def get_opportunity(identifier):
    get_opportunity_request ={
     "Identifier": identifier,
        "Catalog": CATALOG_TO_USE
    }
    try:
        # Perform an API call
        response =
 partner_central_client.get_opportunity(**get_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def update_opportunity():
    update_opportunity_request_orig = sd.stringify_json("src/update_opportunity/
update_opportunity_technical_validation.json")
    update_opportunity_request =
 helper.remove_nulls(update_opportunity_request_orig)

    try:
        # Perform an API call
        response =
 partner_central_client.update_opportunity(**update_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def update_opportunity_if_eligible(identifier):
    response = get_opportunity(identifier)
    if response is not None:
        return update_opportunity()
    else:
        print("Failed to retrieve opportunity details")

def usage_demo():
    identifier = "O5465588"
```

```
    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Updating opportunity.")
    print("-" * 88)

    helper.pretty_print_datetime(update_opportunity_if_eligible(identifier))

if __name__ == "__main__":
    usage_demo()
```

- Para obter detalhes da API, consulte a UpdateOpportunityReferência da API AWS SDK for Python (Boto3).

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Cenários para o uso do Partner Central AWS SDKs

Os exemplos de código a seguir mostram como implementar cenários comuns no Partner Central com AWS SDKs. Esses cenários mostram como realizar tarefas específicas chamando várias funções na Central de Parceiros ou combinadas com outros Serviços da AWS. Cada cenário inclui um link para o código-fonte completo, onde podem ser encontradas instruções sobre como configurar e executar o código.

Os cenários têm como alvo um nível intermediário de experiência para ajudar você a compreender ações de serviço em contexto.

Exemplos

- Atualizar a entidade associada de uma oportunidade

Atualizar a entidade associada de uma oportunidade

Os exemplos de código a seguir mostram como:

- Desassociar uma entidade antiga.
- Associar uma entidade nova.

Java

SDK para Java 2.x

> **Note**
>
> Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no repositório Cenários.

Atualizar a entidade associada de uma oportunidade

```
package org.example;

import static org.example.utils.Constants.*;

import org.example.utils.Constants;
import org.example.utils.ReferenceCodesUtils;

import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.http.apache.ApacheHttpClient;
import software.amazon.awssdk.regions.Region;
import
 software.amazon.awssdk.services.partnercentralselling.PartnerCentralSellingClient;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityRequest;
import
 software.amazon.awssdk.services.partnercentralselling.model.AssociateOpportunityResponse
import
 software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityReque
import
 software.amazon.awssdk.services.partnercentralselling.model.DisassociateOpportunityRespo

/*
Purpose
PC-API -17 Replacing a solution
*/

public class ReplaceSolution {

 static PartnerCentralSellingClient client =
 PartnerCentralSellingClient.builder()
            .region(Region.US_EAST_1)
```

```
            .credentialsProvider(DefaultCredentialsProvider.create())
            .httpClient(ApacheHttpClient.builder().build())
            .build();

   public static void main(String[] args) {

    String opportunityId = args.length > 0 ? args[0] : OPPORTUNITY_ID;

    String entityType = "Solutions";
    String originalEntityIdentifier = "S-0000000";
    String newEntityIdentifier = "S-0011111";

    disassociateOppornititityResponse(opportunityId, entityType,
originalEntityIdentifier );
    AssociateOpportunityResponse associateOpportunityResponse =
associateOpportunityResponse(opportunityId, entityType, newEntityIdentifier );

    ReferenceCodesUtils.formatOutput(associateOpportunityResponse);
   }

private static AssociateOpportunityResponse associateOpportunityResponse(String
opportunityId, String entityType, String entityIdentifier) {

       AssociateOpportunityRequest associateOpportunityRequest =
AssociateOpportunityRequest.builder()
   .catalog(Constants.CATALOG_TO_USE)
         .opportunityIdentifier(opportunityId)
         .relatedEntityType(entityType)
         .relatedEntityIdentifier(entityIdentifier)
         .build();

       AssociateOpportunityResponse response =
client.associateOpportunity(associateOpportunityRequest);

       return response;
}

private static DisassociateOpportunityResponse
disassociateOppornititityResponse(String opportunityId, String entityType, String
entityIdentifier) {
 PartnerCentralSellingClient client = PartnerCentralSellingClient.builder()
           .region(Region.US_EAST_1)
           .credentialsProvider(DefaultCredentialsProvider.create())
           .httpClient(ApacheHttpClient.builder().build())
```

```
            .build();

        DisassociateOpportunityRequest disassociateOpportunityRequest =
 DisassociateOpportunityRequest.builder()
    .catalog(Constants.CATALOG_TO_USE)
          .opportunityIdentifier(opportunityId)
          .relatedEntityType(entityType)
          .relatedEntityIdentifier(entityIdentifier)
          .build();

        DisassociateOpportunityResponse response =
 client.disassociateOpportunity(disassociateOpportunityRequest);

        return response;
 }
}
```

- Para obter detalhes da API, consulte os tópicos a seguir na Referência da API AWS SDK for Java 2.x.
  - AssociateOpportunity
  - DisassociateOpportunity

Python

SDK para Python (Boto3)

> **Note**
>
> Tem mais sobre GitHub. Encontre o exemplo completo e saiba como configurar e executar no AWS Code Examples Repository.

Atualizar a entidade associada de uma oportunidade

```
#!/usr/bin/env python

"""
Purpose
PC-API -17 Replacing a solution
```

```
"""
import logging
import boto3
import utils.helpers as helper
from botocore.client import ClientError

from utils.constants import CATALOG_TO_USE

serviceName = "partnercentral-selling"

partner_central_client = boto3.client(
        service_name=serviceName,
        region_name='us-east-1'
)

def replace_solution(original_entity_identifier, new_entity_identifier,
 opportunityIdentifier):
    disassociate_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : "Solutions",
        "RelatedEntityIdentifier" : original_entity_identifier
    }

    associate_opportunity_request ={
        "Catalog": CATALOG_TO_USE,
     "OpportunityIdentifier" : opportunityIdentifier,
        "RelatedEntityType" : "Solutions",
        "RelatedEntityIdentifier" : new_entity_identifier
    }
    try:
        # Perform an API call
        response =
 partner_central_client.disassociate_opportunity(**disassociate_opportunity_request)
        response =
 partner_central_client.associate_opportunity(**associate_opportunity_request)
        return response

    except ClientError as err:
        # Catch all client exceptions
        print(err.response)

def usage_demo():
    original_entity_identifier = "S-0049999"
```

```
    new_entity_identifier = "S-0050014"
    opportunityIdentifier = "O4397574"

    logging.basicConfig(level=logging.INFO, format="%(levelname)s: %(message)s")

    print("-" * 88)
    print("Replacing a solution.")
    print("-" * 88)

    helper.pretty_print_datetime(replace_solution(original_entity_identifier,
 new_entity_identifier, opportunityIdentifier))

if __name__ == "__main__":
    usage_demo()
```

- Consulte detalhes da API nos tópicos a seguir na Referência de API do AWS SDK para Python (Boto3).
  - AssociateOpportunity
  - DisassociateOpportunity

Para obter uma lista completa dos guias do desenvolvedor do AWS SDK e exemplos de código, consulteUtilizar AWS API do Partner Central com um AWS SDK. Este tópico também inclui informações sobre como começar e detalhes sobre versões anteriores do SDK.

Notas da versão

Esta página contém um resumo das mudanças significativas na documentação da API do AWS Partner Central, começando com a atualização mais recente.

Histórico de revisões

Alteração	Descrição	Data
ARCO 6.0	• O AWS Partner Central adicionou novas APIs de medição de receita de parceiros.	2026-06-30
ARCO 5.3	• Atualizações do ACE Resonate: `GetAwsOpportunitySummary` resposta atualizada.	2026-06-16
ARCO 5.2	• Suporte monetário EUR: `CreateOpportunity` `UpdateOpportunity` ,, e `CreateEngagementInvitation` agora aceita, `EUR` além do código `USD` de moeda MRR. Quando a partição da AWS é `aws-eusc` (AWS European Sovereign Cloud), o código da moeda deve ser`EUR`. • EUR em operações de leitura: `GetOpportunity` `ListOpportunities` ,, e `GetAwsOpportunitySummary` agora retorne `EUR` ao	2026-03-30

Alteração	Descrição	Data
	`CurrencyCode` campo para oportunidades na `aws-eusc` partição.	
ARCO 5.1	• Suporte monetário EUR: `CreateOpportunity` `UpdateOpportunity` ,, e `CreateEngagementInvitation` agora aceita, `EUR` além do código `USD` de moeda MRR. Quando a partição da AWS é `aws-eusc` (AWS European Sovereign Cloud), o código da moeda deve ser`EUR`. • EUR em operações de leitura: `GetOpportunity` `ListOpportunities` ,, e `GetAwsOpportunitySummary` agora retorne `EUR` ao `CurrencyCode` campo para oportunidades na `aws-eusc` partição.	2026-03-30

Alteração	Descrição	Data
ARCO 5	• API de conta introduzida: foi adicionada uma nova API de conta ao AWS Partner Central, permitindo que os parceiros gerenciem as informações da conta, o registro, os perfis, o gerenciamento de domínio e as conexões de parceiros. • Documentação de referênci a da API da conta: foi adicionada uma documenta ção abrangente de referênci a da API para a API da conta, incluindo registro de parceiros, gerenciam ento de perfil, verificação de domínio e fluxos de trabalho de conexão de conta. • Registro da API da conta: foi adicionado suporte de registro e documentação para a API da conta para ajudar a auditar e monitorar o uso da API. • Notificações da API da conta: foi adicionado suporte à notificação de eventos para a API da conta, permitindo que os parceiros recebam atualizaç ões em tempo real sobre as atividades relacionadas à	2025-11-30

Alteração	Descrição	Data
	conta, incluindo conexões da conta do parceiro. • Cotas da API da conta: foi adicionada a documentação de cotas de serviço para a API da conta. • Teste de sandbox para a API de contas: foram adicionados recursos de teste de sandbox para a API de contas, permitindo que os parceiros testem fluxos de trabalho de gerenciamento de contas em um ambiente que não seja de produção. • API de benefícios introduzida: foi adicionada uma nova API de benefícios ao AWS Partner Central, permitindo que os parceiros descubram, se inscrevam e gerenciem benefícios em todos os programas de AWS parceiros por meio de uma interface centralizada. • Documentação de referência da API de benefícios: foi adicionada uma documentação de referência abrangente da API de benefícios para a API de benefícios, alocações e processos de atendimento.	

Alteração	Descrição	Data
	• Registro da API de benefícios: foi adicionado o suporte de registro e documentação para a API de benefícios para ajudar a auditar e monitorar o uso da API. • Notificações da API de benefícios: foi adicionado o suporte à notificação de eventos para a API de benefícios, permitindo que os parceiros recebam atualizações em tempo real sobre solicitações e alocações de benefícios. • Cotas da API de benefícios: foi adicionada a documentação de cotas de serviço para a API de benefícios. • Teste de sandbox para API de benefícios: foram adicionados recursos de teste de sandbox para a API de benefícios, permitindo que os parceiros testem fluxos de trabalho de gerenciamento de benefícios em um ambiente não produtivo. • API de vendas atualizada: ações aprimoradas de engajamento da API de vendas para apoiar o fluxo	

Alteração	Descrição	Data
	de trabalho de gerenciamento de leads. • Atualizações da documentação da API de venda: documentação aprimorada da API de vendas com estrutura aprimorada e fluxos de trabalho atualizados para gerenciamento de leads. • Venda de atualizações de notificações da API: documentação aprimorada de notificação de eventos para eventos de engajamento e convite de engajamento. • Expansão da estrutura da API: estrutura de documentação expandida para oferecer suporte a quatro APIs distintas (conta, benefícios, vendas e canal) com seções dedicadas para as permissões, registros, notificações, cotas e testes de sandbox de cada API. • Documentação aprimorada de uso da API: foram adicionados guias de uso abrangentes para APIs de contas e benefícios com fluxos de trabalho detalhados e padrões de integração.	

Alteração	Descrição	Data
	• Documentação das regiões suportadas: foi adicionada documentação para AWS as regiões compatíveis com as APIs de contas e benefícios. • Novo filtro para`ListOpportunities` : `TargetCloseDate` filtro adicionado para permitir filtrar oportunidades de acordo com a data de fechamento esperada, uso `AfterTargetCloseDate` e `BeforeTargetCloseDate` campos em YYYY-MM-DD formato.	

Alteração	Descrição	Data
ARCO 4	• API de canal introduzida: foi adicionada uma nova API de canal ao AWS Partner Central, permitindo que os parceiros gerenciem relacionamentos entre canais e colaborações entre parceiros. • Documentação de referência da API do canal: foi adicionada uma documentação de referência abrangente da API do canal, incluindo todas as ações e modelos de dados disponíveis. • Permissões da API do canal: foram adicionadas políticas de permissão do IAM e documentação de controle de acesso específicas à API do canal. • Registro da API do canal: foi adicionado suporte de registro e documentação para a API do canal para ajudar a auditar e monitorar o uso da API. • Notificações da API do canal: foi adicionado suporte à notificação de eventos para a API do canal, permitindo que os parceiros recebam atualizações em tempo real sobre as	2025-11-19

Alteração	Descrição	Data
	atividades relacionadas ao canal. • Cotas da API do canal: foi adicionada a documentação de cotas de serviço para a API do canal. • Teste de sandbox para a API do canal: foram adicionados recursos de teste de sandbox para a API do canal, permitindo que os parceiros testem os fluxos de trabalho do canal em um ambiente não produtivo. • Reorganização da estrutura da API: documentação reorganizada para distingui r entre a API de venda e a API de canal, com seções dedicadas para as permissões, registros, notificações e cotas de cada API. • Documentação das regiões suportadas: foi adicionada documentação para AWS regiões compatíveis com as APIs de vendas e de canal.	

Alteração	Descrição	Data
ARCO 3	• Políticas de permissão de API atualizadas: as políticas foram atualizadas para refletir as novas ações introduzidas nesta versão. Certifique-se de que suas funções e permissões do IAM sejam ajustadas adequadamente. • Detalhes de uso do cabeçalho personalizado: foram adicionados detalhes sobre como usar o cabeçalho personalizado `X-Amzn-User-Agent` para ajudar a auditar e medir o uso da API. • Ações substituídas por ações assíncronas: `AcceptOpportunityEngagementInvitation` Substituído pela ação assíncrona `StartEngagementByAcceptingInvitationTask` e o formato da carga foi alterado em relação ao RC2. • Renomeação de entidades e ações: Alterado `OpportunityEngagementInvitation` para `EngagementInvitation`. Os	2024-10-17

Alteração	Descrição	Data
	atributos dessas ações também foram alterados para se alinharem à nova entidade. As seguintes ações foram substituídas: • `GetOpportunityEngagementInvitation` é agora `GetEngagementInvitation`. Vários atributos no convite foram alterados, adicionados ou removidos. • `ListOpportunityEngagementInvitations` é agora `ListEngagementInvitations`. • `RejectOpportunityEngagementInvitation` é agora `RejectEngagementInvitation`. • `AssignOpportunity` atualização de parâmetros: `AssignOpportunity` agora leva `Assignee` em vez de `AssigneeEmail`. Atualize suas implementações adequadamente. • `SubmitOpportunity` alteração de funcionalidade: agora	

Alteração	Descrição	Data
	`SubmitOpportunity` é executada pela ação assíncrona. `StartEngagementFromOpportunityTask` `SubmitOpportunity` agora aceita `InvolvementType` `Visibility` e. • Renomeação e expansão de atributos: `EstimatedAwsMonthlyRevenue` alterado para`ExpectedCustomerSpend` . `ExpectedCustomerSpend` agora inclui novos atributos, como `Frequency` `TargetCompany` e. • Resumo da nova ação para AWS oportunidade: a visibilidade da oportunidade agora é fornecida por meio de uma ação separada chamada`GetAwsOpportunitySummary` . • Atualizações do fluxo de trabalho para trabalhar com oportunidades: O fluxo de trabalho para trabalhar com suas oportunidades foi atualizado para usar as novas ações introduzidas nesta versão. • Atualizações de fluxo de trabalho para trabalhar com	

Alteração	Descrição	Data
	oportunidades: o fluxo de trabalho para trabalhar com oportunidades foi atualizado para usoEngagementInvitations , o que melhora as oportunidades de tratamento oferecidas pela AWS. • Atualizações do fluxo de trabalho para rastrear atualizações: O fluxo de trabalho para rastrear atualizações agora está trabalhando com atualizações de oportunidades, aumentando a clareza e a eficiência no monitoramento do status da oportunidade. • Novos filtros para listar ações: novos filtros estão disponíveis para ListOpportunities ações ListSolutions e. • PoisListSolutions , os novos filtros são FilterCategory FilterIdentifier , FilterStatus e. • PoisListOpportunities , os novos filtros são FilterCustomerCompanyName FilterIde	

Alteração	Descrição	Data
	`ntifier` , `FilterLastModifiedDate` `FilterLifeCycleAppprovalStatus` ,`FilterLifeCycleStage` ,, `FilterOrigin` e. • Remoção do `AccessControl` subobjeto: O `AccessControl` subobjeto no modelo de oportunidade foi removido. `NationalSecurity` agora é um atributo de nível superior. • Atributos realocados para `GetAwsOpportunitySummary` : Atributos como `AWSOpportunityTeam` insights (`EngagementScore` ,`NextBestAction`), origem e agora `InvolvementType` estão disponíveis por meio de, `GetAwsOpportunitySummary` em vez de `GetOpportunity` . • Introdução do `InvolvementType` campo: introduziu o `InvolvementType` campo para diferenciar entre oportunidades de	

Alteração	Descrição	Data
	venda conjunta e oportunidades somente de visibilidade. • Atualizações do nome do evento: os eventos foram atualizados para refletir as mudanças. Por exemplo, o convite de engajamento de oportunidade criado agora é um convite de engajamento criado. • Atualizações da carga útil de resposta: todas as cargas de resposta agora incluem um ARN and/or de ID. • Solicitar atualizações de carga útil: todas as cargas de solicitação agora usam identificadores como entrada, que aceitam um ID ou ARN. • Atualização da entidade de engajamento: o atributo de título na entidade de engajamento foi renomeado `EngagementTitle` . • Remoção do prefixo do filtro: todos os filtros não incluem mais o prefixo do filtro, simplificando o mecanismo de filtragem. • `StartEngagementFromOpportunity` atualizaç	

Alteração	Descrição	Data
	ão da ação: o atributo `AwsInvolvement` foi alterado para`AwsSubmission` . • Atualização dos detalhes do projeto de convite: `Invitation.ProjectDetails.TargetCompletionDate` foi renomeado`Invitation.ProjectDetails.EstimatedCompletionDate` . • Atualização da estimativa de receita: `Invitation.PartnerRevenueEstimate` é agora`ExpectedCustomerSpend` . O tipo de dados mudou de um objeto para uma matriz contendo um objeto. • Atualização da conta do destinatário: o campo `ReceiverAccount` foi renomeado`AccountReceiver` . • Novo atributo do convite de engajamento: o `SenderContact` atributo foi introduzido para convites de engajamento. • `Customer.Contact` agora é uma	

Alteração	Descrição	Data
	matriz em vez de um objeto. `OpportunityOwner` , `PartnerAccountManager` são valores suportados como `BusinessTitle` . • `PartnerOpportunityTeam` agora é denominado `OpportunityTeam` . • `APNPrograms` é agora `ApnPrograms` • Os status dos convites de engajamento foram atualizados. • `ListEngagementInvitations` agora requer `ParticipantType` como `RECEIVER`.	

Alteração	Descrição	Data
ARCO 2	• Introduziu uma nova entidade chamada OpportunityEngagement Invitation. • Introduziu uma nova ação: ListOpportunityEngagementInvitations. • Introduziu uma nova ação: GetOpportunityEngagementInvitation. • Apresentou um novo evento: “Foi criado um convite de engajamento de oportunidade”. • AcceptOpportunity Substituído por AcceptOpportunityEngagementInvitation. • RejectOpportunity Substituído por RejectOpportunityEngagementInvitation. • Substituiu “Oportunidade aceita” por “Convite de engajamento de oportunidade aceito”. • Substituiu “Oportunidade rejeitada” por “Convite de engajamento de oportunidade rejeitado”. • `PrimaryNeedsFromAWS` alterado para PrimaryNeedsFromAws.	2024-08-07

Alteração	Descrição	Data
	• AWSOpportunityTeam alterado paraAwsOpportunityTeam . • AWSSalesLifeCycle alterado paraAwsSalesLifeCycle . • PrimaryNeedsFromAWSChangeReason alterado paraPrimaryNeedsFromAwsChangeReason . • AWSAccountId alterado paraAwsAccountId . • ExpectedMonthlyAWSRevenue alterado paraExpectedMonthlyAwsRevenue . • AWSFundingUsed alterado paraAwsFundingUsed . • AWSMarketplaceOffers alterado paraAwsMarketplaceOffers . • Countryalterado paraCountryCode . • PartnerOpportunityContact alterado paraPartnerOpportunityTeam . • Campo atualizado Contact.Title	

Alteração	Descrição	Data
	para`Contact.BusinessTitle` . • Campo atualizado `ContactInfo` para`OwnerInfo` . • Mudou a AWS equipe de um mapa para uma lista. • Detalhes adicionados sobre a lista de aceitos Rejection Reasons. • Forneceu informações sobre como usar o Token do Cliente. • Detalhes incluídos sobre como usar UpdateOpportunity. • Orientações adicionais sobre o uso de AWS produtos e soluções de parceiros. • Forneceu detalhes sobre a OpportunityEngagementInvitation entidade. • Atualizado StateOrRegion para incluir uma lista revisada de valores aceitos. • Implementou várias correções de bugs para melhorar o desempenho e as mensagens de erro.	

Alteração	Descrição	Data
ARCO 1	• [Interrupção] Renomeação de ApprovalStatus para ReviewStatus: Renomeamos o ApprovalStatus campo ReviewStatus para melhor refletir sua finalidade. Além disso, estamos introduzindo um novo status, Envio pendente, para indicar uma oportunidade preliminar. Agora, ReviewStatus aceitará os seguintes valores: Envio pendente, Enviado, Em análise, Aprovado, Rejeitado e Ação necessária. O status anterior de Rascunho será substituído pelo Envio pendente. • [Interrupção] Introdução da SubmitOpportunity Ação: Uma nova ação SubmitOpportunity,, foi introduzida. Ao contrário do comportamento atual em que CreateOpportunity também envia a oportunidade, agora você pode criar uma oportunidade no estado Envio pendente sem vincular imediatamente uma Solução ou AWS Produto. Para concluir o envio, use a AssociateOpportunity	2024-06-21

Alteração	Descrição	Data
	ação para vincular uma solução ou produto, seguida por SubmitOpportunity. Essa separação fornece melhor desempenho de resposta da API e uma fase de preparação flexível antes do envio. • [Quebra] Requisito de um parâmetro de catálogo e descontinuação do Sandbox Endpoint: O endpoint 'gamma' (partnercentral-selling-gamma.us-eas t-1.amazonaws.com) ficará indisponível após /2024. 7/30 Todas as ações da API agora exigem um parâmetro Catalog para especificar se a operação é executada no Sandbox ou no AWS catálogo. As regras de notificação agora serão filtradas pelo catálogo em vez do EnvironmentName. • [Quebra] Renomear OpportunityIdentifier para Identifier: Actions AssignOpportunity, AcceptOpportunity, e RejectOpportunity agora use Identifier em vez de OpportunityIdentifier para	

Alteração	Descrição	Data
	manter a consistência com UpdateOpportunity. • Mudanças do tipo Enum para String for APNPrograma e UseCase: Estaremos convertendo CustomerUseCase, Project.APNPrograms Marketing.ActivityUseCases, e Project.CustomerUseCase do tipo de enumeração para o tipo de string para reduzir os riscos associados à alteração de valores. Esses campos aceitarão valores somente de listas predefinidas, mas não estão disponíveis de forma nativa nos SDKs. • Tratamento aprimorado de erros: o tratamento de erros nas APIs foi significativamente aprimorado para facilitar a depuração e a resolução mais rápida de problemas. O novo tratamento de erros estrutura os erros em dois estágios: 1. Falha na validação da solicitação: verifica os tipos e o formato dos dados, ou 2. Falha na validação comercial: todos os problemas em	

Alteração	Descrição	Data
	uma carga serão listados em uma resposta, especific ando o campo que contém erros. Esse feedback consolidado permite que você solucione e corrija erros com mais eficiência. Códigos e formatos de erros foram padronizados. • Identificador de origem do cliente: capacidade dos parceiros incluírem o X-Amzn-User-Agent cabeçalho em suas solicitaç ões, o que ajuda a identific ar a origem do tráfego. Use o formato [Nome da solução, incluindo provedor da solução] | [Versão] Por exemplo:AWS Partner CRM Connector | v2.0.1 • Correção de bug, já implantada, sem alteração no SDK] Criação da oportunidade e ID AWS da conta: os parceiros agora podem iniciar uma oportunid ade enquanto alteram a ID da AWS conta de nula para um valor válido sem encontrar erros. Anteriorm ente, um bug impedia que os parceiros realizassem	

Alteração	Descrição	Data
	essas ações separadam ente. • Correção de bug, já implantada, sem alteração no SDK] Associação de soluções com novas oportunidades: os parceiros podem associar uma solução a uma oportunid ade recém-criada. Antes, quando os parceiros criavam uma nova oportunid ade e associavam simultane amente uma solução, as informações da solução estavam sendo perdidas. Esse problema foi resolvido. • Correção de bug, já implantada, sem alteração no SDK] Exibição dos detalhes do proprietário da oportunidade: o problema em que algumas oportunid ades exibiam os detalhes do proprietário em um campo combinado LastName "" em vez dos campos esperados de “E-mail”, "FirstName“e" LastName "foi corrigido. • Correção de bug, já implantada, sem alteração no SDK] Customer. Account.WebsiteUrl Campo opcional: o Customer.	

Alteração	Descrição	Data
	Account.WebsiteUrl campo se tornou opcional quando o AccessControls.NationalSecurity campo é “Sim”, alinhando assim o comportamento da API com o comportamento da interface de usuário existente. Se o AccessControls.NationalSecurity campo for “Não” ou nulo, o WebsiteUrl campo será obrigatório como uma validação comercial. • Correção de bug, já implantada, sem alteração no SDK] Requisito de ID de AWS conta para parceiros de consultoria: a inconsistência na interface do usuário que mostrava o campo ID da AWS conta como opcional para parceiros de consultoria foi resolvida. Agora é um campo obrigatório condicional, conforme a documentação. • Correção de bug. Para ser implantado 6/19, nenhuma alteração no SDK] Estágio de uma Co-sell oportunidade para fechada e perdida: os parceiros	

Alteração	Descrição	Data

	agora podem atualizar com sucesso o estágio de uma Co-sell oportunidade para encerrada e perdida, desde que um motivo fechado seja fornecido. Isso resolve o erro anterior dizendo “Campo obrigatório ausente: o motivo da perda é obrigatório ao fechar a oportunidade”.
	• Documentação] Melhorias na documentação: Fiz várias melhorias na documentação para refletir as mudanças.
	• Problemas conhecidos com uma correção pendente e uma lista dos próximos recursos:
	  • Ao executar Associate Opportunity ações com soluções ou produtos, as mensagens de erro das Ofertas são mostradas incorretamente.
	  • Capacidade de criar oportunidades remotamente e simular o processo de AWS validação no Sandbox
	  • O código postal retorna nulo quando está no	

Alteração	Descrição	Data
	formato XXXXX-XXXX dos EUA. • As próximas etapas e o histórico da próxima etapa não estão disponíveis para oportunidades marcadas como “Não preciso de suporte de”AWS. • Não é possível obter oportunidades histórica s com descrições de problemas comerciais do cliente com mais de 255 caracteres. • Capacidade de excluir oportunidades que estão pendentes de envio. • LeadSource não está disponível no modelo de dados.	

Alteração	Descrição	Data
Versão beta 5	• Alterado LifeCycle.Approval Status para LifeCycle .ReviewStatus • Alterado Insights. ReviewComments para LifeCycle.ReviewComments • Adicionado LifeCycle .ReviewStatusReason (novo atributo) • Novo valor adicionado “Pendente” a PartnerAc ceptance.Status • SDKs lançados para dotnet, java, javascript, python, ruby. • Guia de referência da API atualizado para refletir as alterações acima.	2024-04-19

Alteração	Descrição	Data
Versão beta 4	• Teste de sandbox adicionado: capacidade de testar a notificação de evento de oportunidade no sandbox. • Documentação adicionada: introduziu exemplos de regras de notificação de eventos de oportunidade. • Lançamentos de SDK adicionados: SDKs lançados para Java (V2), Javascript (V2), (V3) e Python DotNet (V3). • Formato de data atualizado: atualizado para seguir os padrões ISO.	2024-03-04

Alteração	Descrição	Data
Versão beta 3	• Arquivo de registro de alterações adicionado: introduziu um arquivo de registro de alterações para melhor rastreamento de atualizações e alterações. • Guia de referência de API atualizado (PDF): lançou uma nova versão do Guia de referência de API em formato PDF. • Guia de referência de API atualizado (versão da Web): a versão web do Guia de referência de API foi atualizada. • DotNet SDK adicionado: Lançamos o DotNet SDK, expandindo nosso suporte para diferentes ambientes de programação. • SDK do Python Boto 3 atualizado (V2): Lançada a versão 2 do SDK do Python Boto 3, introduzindo novos recursos e melhorias.	2024-01-24

Alteração	Descrição	Data
Versão beta 2	• Guia de referência de API adicionado (versão Web): Lançou uma nova versão do Guia de referência de API (versão Web). Nota: Esta versão requer que um servidor web local seja executado após a extração para uma funcionalidade adequada.	2024-01-07
Versão beta 1	• Boto3 adicionado (versão inicial): lançou a versão inicial do Boto3, um SDK Python para nossos serviços. • Guia do programa beta adicionado: introduzimos um guia para os participa ntes do nosso programa de testes beta. • Guia de referência de API adicionado (PDF): Carregamos a primeira versão do nosso Guia de referência de API em formato PDF, fornecendo documentação abrangente para nossa API.	2023-12-15

Alteração	Descrição	Data
Versão beta 0	Essa versão inicial fornece um conjunto de funcionalidades para gerenciar e otimizar engajamentos e oportunidades de parceiros no AWS Partner Central.	2023-11-15

Histórico do documento

Veja a seguir uma lista dos lançamentos da documentação para essa referência de API.

Alteração	Descrição	Data
Documentação adicionada para APIs de medição de receita de parceiros	AWS O Partner Central adicionou novas APIs para a Medição da Receita de Parceiros (PRM).	30 de junho de 2026
Documentação atualizada para ACE Resonate	AWS Resposta atualizada da API da Partner Central para GetAwsOpportunitySummary ações de suporte ao Cosell Motion.	16 de junho de 2026
Documentação adicionada para o Partner Central MCP Server	AWS O servidor MCP dos agentes centrais de parceiros fornece ferramentas centrais de parceiros por meio do Model Context Protocol (MCP).	16 de março de 2026
Lançamento do RC 5	Lançada a versão Candidate 5 da API do AWS Partner Central. Introduziu uma nova API de conta no AWS Partner Central, permitindo que os parceiros gerenciem as informações da conta, o registro, os perfis, o gerenciamento de domínio e as conexões de parceiros. Introduziu uma nova API de benefícios no AWS Partner Central, permitindo que	30 de novembro de 2025

	os parceiros descubram, se inscrevam e gerenciem benefícios em todos os programas de AWS parceiros por meio de uma interface centralizada. Ações aprimoradas de engajamento da API de vendas para apoiar o fluxo de trabalho de gerenciamento de leads. Foram adicionados guias de uso abrangentes e documentação de API para APIs de contas e benefícios com fluxos de trabalho detalhados e padrões de integração. Documentação aprimorada da API de vendas para gerenciamento de engajamento com contexto de leads.	
Lançamento do RC 4	Lançada a versão Candidate 4 da API do AWS Partner Central. Introduziu a API de canais para gerenciar relacionamentos entre canais e colaborações entre parceiros, permitindo que os parceiros se qualifiquem para os benefícios do programa ao realizar transações com clientes finais usando o Billing Transfer. A documentação foi reorganizada para oferecer suporte a várias APIs do AWS Partner Central.	19 de novembro de 2025

Documentação adicionada para oportunidades com vários parceiros	Multi-partner opportunities (em versão prévia) é uma experiência integrada no AWS Partner Central que permite que AWS os parceiros encontrem e se conectem com outros parceiros e vendam juntos ofertas com clientes.	4 de dezembro de 2024
Lançamento do RC 3	Lançada a versão Candidate 3 da API AWS Partner Central	17 de outubro de 2024
Lançamento do RC 2	Lançada a versão Candidate 2 da API AWS Partner Central	7 de agosto de 2024
Versão RC 1	Lançada a versão Candidate 1 da API AWS Partner Central	21 de junho de 2024
Lançamento do Beta 5	Lançada a versão Beta 5 da API AWS Partner Central	19 de abril de 2024
Versão beta 4	Lançada a versão beta 4 da API AWS Partner Central	4 de março de 2024
Lançamento do Beta 3	Lançada a versão Beta 3 da API AWS Partner Central	24 de janeiro de 2024
Versão beta 2	Lançada a versão Beta 2 da API AWS Partner Central	7 de janeiro de 2024
Versão beta 1	Lançada a versão beta 1 da API AWS Partner Central	15 de dezembro de 2023
Lançamento do Beta 0	AWS Versão inicial da API Partner Central	15 de novembro de 2023