aws

Guia do Desenvolvedor

AWS Cloud Map

AWS Cloud Map: Guia do Desenvolvedor

Table of Contents

O que é AWS Cloud Map?

AWS Cloud Map é uma solução totalmente gerenciada que você pode usar para mapear nomes lógicos para os serviços e recursos de back-end dos quais seus aplicativos dependem. Também ajuda seus aplicativos a descobrir recursos usando uma das chamadas AWS SDKs de RESTful API ou consultas de DNS. AWS Cloud Map serve somente recursos saudáveis, que podem ser tabelas do Amazon DynamoDB (DynamoDB), filas do Amazon Simple Queue Service (Amazon SQS), quaisquer serviços de aplicativos de nível superior criados usando EC2 instâncias do Amazon Elastic Compute Cloud (Amazon) ou tarefas do Amazon Elastic Container Service (Amazon ECS) e muito mais.

Componentes do AWS Cloud Map

Namespace

Para começar, primeiro você cria um AWS Cloud Map namespace que funciona como uma forma de agrupar serviços para um aplicativo. Um namespace identifica o nome que você deseja usar para localizar seus recursos e também especifica como você deseja localizá-los: usando chamadas de AWS Cloud Map DiscoverInstancesAPI, consultas de DNS em uma VPC ou consultas públicas de DNS. Normalmente, um namespace contém todos os serviços para um aplicativo, como um aplicativo de faturamento. Para obter mais informações, consulte AWS Cloud Map namespaces.

Serviço

Depois de criar um namespace, você cria um AWS Cloud Map serviço para cada tipo de recurso que deseja usar AWS Cloud Map para localizar endpoints. Por exemplo, você pode criar serviços para servidores web e servidores de banco de dados.

Um serviço é um modelo AWS Cloud Map usado quando seu aplicativo adiciona outro recurso, como outro servidor web. Se você optou por localizar recursos usando o DNS ao criar o namespace, um serviço conterá as informações sobre os tipos de registros que você deseja usar para localizar o servidor web. Um serviço também indica se você deseja verificar a integridade do recurso e se deseja usar as verificações de saúde do Amazon Route 53 ou um verificador de saúde terceirizado. Para obter mais informações, consulte AWS Cloud Map serviços.

Instância de serviço

Quando seu aplicativo adiciona um recurso, você pode chamar a ação da AWS Cloud Map RegisterInstanceAPI no código, o que cria uma instância AWS Cloud Map de serviço em um serviço. A instância de serviço contém informações sobre como seu aplicativo pode localizar o recurso, seja usando o DNS ou usando a ação da AWS Cloud Map DiscoverInstancesAPI.

Quando seu aplicativo precisa se conectar a um recurso, ele chama DiscoverInstancesou utiliza consultas DNS públicas ou privadas especificando o namespace e o serviço associados ao recurso. AWS Cloud Map retorna informações sobre como localizar um ou mais recursos. Se você especificou a verificação de saúde ao criar o serviço, AWS Cloud Map retornará somente instâncias íntegras. Para obter mais informações, consulte AWS Cloud Map instâncias de serviço.

Acessando AWS Cloud Map

Você pode acessar AWS Cloud Map das seguintes formas:

- Console de gerenciamento da AWS— Os procedimentos deste guia explicam como usar o Console de gerenciamento da AWS para realizar tarefas.
- AWS SDKs— Se você estiver usando uma linguagem de programação que AWS fornece um SDK para, você pode usar um SDK para acessar. AWS Cloud Map SDKs simplifique a autenticação, integre-se facilmente ao seu ambiente de desenvolvimento e forneça acesso aos AWS Cloud Map comandos. Para obter mais informações, consulte Ferramentas para a Amazon Web Services.
- AWS Command Line Interface— Para obter mais informações, consulte Comece a usar o AWS CLI no Guia AWS Command Line Interface do usuário.
- AWS Tools for Windows PowerShell— Para obter mais informações, consulte Comece a usar o AWS Tools for Windows PowerShell no Guia Ferramentas da AWS para PowerShell do usuário.
- AWS Cloud Map API — Se você estiver usando uma linguagem de programação para a qual um SDK não está disponível, consulte a Referência da AWS Cloud Map API para obter informações sobre ações de API e sobre como fazer solicitações de API.

> **Note**
>
> IPv6 Suporte ao cliente — a partir de 22 de junho de 2023, em todas as novas regiões, todos os comandos enviados pelos `IPv6` clientes são AWS Cloud Map roteados para um novo endpoint dualstack (). `servicediscovery.<region>.api.aws` AWS Cloud Map `IPv6`-somente redes podem ser acessadas tanto para endpoints legacy

(**`servicediscovery.<region>.amazonaws.com`**) quanto dualstack nas seguintes regiões, lançadas antes de 22 de junho de 2023:

- Leste dos EUA (Ohio), us-east-2
- Leste dos EUA (Norte da Virgínia), us-east-1
- Oeste dos EUA (Norte da Califórnia), us-west-1
- Oeste dos EUA (Oregon), us-west-2
- África (Cidade do Cabo), af-south-1
- Ásia-Pacífico (Hong Kong), ap-east-1
- Ásia-Pacífico (Hyderabad), ap-south-2
- Ásia-Pacífico (Jacarta), ap-southeast-3
- Região da Ásia-Pacífico (Melbourne), ap-southeast-4
- Ásia-Pacífico (Mumbai), ap-south-1
- Ásia-Pacífico (Osaka) - ap-northeast-3
- Ásia-Pacífico (Seul), ap-northeast-2
- Ásia-Pacífico (Singapura), ap-southeast-1
- Ásia-Pacífico (Sydney), ap-southeast-2
- Ásia-Pacífico (Tóquio), ap-northeast-1
- Canadá (Central), ca-central-1
- Europa (Frankfurt), eu-central-1
- Europa (Irlanda), eu-west-1
- Europa (Londres), eu-west-2
- Europa (Milão),eu-south-1
- Europa (Paris), eu-west-3
- Europa (Espanha), eu-south-2
- Europa (Estocolmo), eu-north-1
- Europa (Zurique), eu-central-2
- Oriente Médio (Bahrein), me-south-1
- Oriente Médio (EAU), me-central-1
- América do Sul (São Paulo), sa-east-1
- AWS GovCloud (Leste dos EUA) — -1 us-gov-east

- AWS GovCloud (Oeste dos EUA) — -1 us-gov-west

AWS Identity and Access Management

AWS Cloud Map se integra ao AWS Identity and Access Management (IAM), um serviço que sua organização pode usar para realizar as seguintes ações:

- Crie usuários e grupos na AWS conta da sua organização
- Compartilhe os recursos da sua AWS conta entre os usuários da conta de forma eficiente
- Atribuir credenciais de segurança exclusivas a cada usuário
- Controlar detalhadamente o acesso do usuário a serviços e recursos

Por exemplo, você pode usar o IAM com AWS Cloud Map para controlar quais usuários da sua AWS conta podem criar um novo namespace ou registrar instâncias.

Para obter mais informações sobre o IAM, consulte os seguintes recursos:

- Identity and Access Management para AWS Cloud Map
- AWS Identity and Access Management
- Guia do usuário do IAM

AWS Cloud Map Preços

AWS Cloud Map o preço é baseado nos recursos que você registra no registro de serviços e nas chamadas de API que você faz para descobri-los. Com isso, não AWS Cloud Map há pagamentos antecipados e você paga apenas pelo que usa.

Opcionalmente, você pode habilitar a descoberta baseada em DNS para os recursos com endereços IP. Você também pode habilitar a verificação de integridade para seus recursos usando as verificações de integridade do Amazon Route 53, quer esteja descobrindo instâncias usando chamadas à API ou consultas ao DNS. Serão cobrados encargos adicionais relacionados ao uso do DNS Route 53 e da verificação de integridade.

Para obter mais informações, consulte Preços do AWS Cloud Map.

AWS Cloud Map e conformidade com a AWS nuvem

Para obter informações sobre AWS Cloud Map conformidade com vários regulamentos de conformidade de segurança e padrões de auditoria, consulte as páginas a seguir:

- AWS Conformidade na nuvem
- AWS Serviços no escopo do Programa de Conformidade

Começando com AWS Cloud Map

Os guias a seguir mostram como configurar para usar AWS Cloud Map e realizar tarefas comuns usando AWS Cloud Map namespaces.

Visão geral do guia	Saiba mais
Inscrevendo-se AWS e se preparando para usar AWS Cloud Map	Configurado para usar AWS Cloud Map
Usando consultas de DNS e chamadas de API para descobrir serviços de back-end.	Saiba como usar a descoberta AWS Cloud Map de serviços com consultas de DNS e chamadas de API
Usando consultas de DNS e chamadas de API para descobrir serviços de back-end usando o. AWS CLI	Saiba como usar a descoberta AWS Cloud Map de serviços com consultas de DNS e chamadas de API usando o AWS CLI
Criação de um aplicativo de amostra e uso de atributos personalizados no código para descobrir recursos.	Saiba como usar a descoberta AWS Cloud Map de serviços com atributos personalizados
Criar um aplicativo de amostra e usar atributos personalizados no código para descobrir recursos usando AWS CLI o.	Saiba como usar a descoberta AWS Cloud Map de serviços com atributos personalizados usando o AWS CLI

Configurado para usar AWS Cloud Map

A visão geral e os procedimentos nas seções a seguir têm como objetivo ajudá-lo a começar a usar AWS e prepará-lo para começar a usar AWS Cloud Map.

Tópicos

- Inscreva-se para AWS
- Acesse a API AWS CLI, AWS Tools for Windows PowerShell, ou o AWS SDKs
- Configurar o AWS Command Line Interface ou AWS Tools for Windows PowerShell
- Baixe um AWS SDK

Inscreva-se para AWS

Inscreva-se para um Conta da AWS

Se você não tiver um Conta da AWS, conclua as etapas a seguir para criar um.

Para se inscrever em um Conta da AWS

1. Abra a https://portal.aws.amazon.com/billing/inscrição.
2. Siga as instruções online.

   Parte do procedimento de inscrição envolve receber uma chamada telefônica ou uma mensagem de texto e inserir um código de verificação pelo teclado do telefone.

   Quando você se inscreve em um Conta da AWS, um Usuário raiz da conta da AWSé criado. O usuário-raiz tem acesso a todos os Serviços da AWS e recursos na conta. Como prática recomendada de segurança, atribua o acesso administrativo a um usuário e use somente o usuário-raiz para executar tarefas que exigem acesso de usuário-raiz.

AWS envia um e-mail de confirmação após a conclusão do processo de inscrição. A qualquer momento, você pode visualizar a atividade atual da sua conta e gerenciar sua conta acessando https://aws.amazon.com/e escolhendo Minha conta.

Criar um usuário com acesso administrativo

Depois de se inscrever em um Conta da AWS, proteja seu Usuário raiz da conta da AWS Centro de Identidade do AWS IAM, habilite e crie um usuário administrativo para que você não use o usuário root nas tarefas diárias.

Proteja seu Usuário raiz da conta da AWS

1. Faça login Console de gerenciamento da AWScomo proprietário da conta escolhendo Usuário raiz e inserindo seu endereço de Conta da AWS e-mail. Na próxima página, insira a senha.

   Para obter ajuda ao fazer login usando o usuário-raiz, consulte Fazer login como usuário-raiz no Guia do usuário do Início de Sessão da AWS .
2. Habilite a autenticação multifator (MFA) para o usuário-raiz.

   Para obter instruções, consulte Habilitar um dispositivo de MFA virtual para seu usuário Conta da AWS raiz (console) no Guia do usuário do IAM.

Criar um usuário com acesso administrativo

1. Habilita o Centro de Identidade do IAM.

   Para obter instruções, consulte Habilitar o Centro de Identidade do AWS IAM no Guia do usuário do Centro de Identidade do AWS IAM .
2. No Centro de Identidade do IAM, conceda o acesso administrativo a um usuário.

   Para ver um tutorial sobre como usar o Diretório do Centro de Identidade do IAM como fonte de identidade, consulte Configurar o acesso do usuário com o padrão Diretório do Centro de Identidade do IAM no Guia Centro de Identidade do AWS IAM do usuário.

Iniciar sessão como o usuário com acesso administrativo

- Para fazer login com o seu usuário do Centro de Identidade do IAM, use o URL de login enviado ao seu endereço de e-mail quando o usuário do Centro de Identidade do IAM foi criado.

  Para obter ajuda para fazer login usando um usuário do IAM Identity Center, consulte Como fazer login no portal de AWS acesso no Guia Início de Sessão da AWS do usuário.

Atribuir acesso a usuários adicionais

1. No Centro de Identidade do IAM, crie um conjunto de permissões que siga as práticas recomendadas de aplicação de permissões com privilégio mínimo.

   Para obter instruções, consulte Criar um conjunto de permissões no Guia do usuário do Centro de Identidade do AWS IAM .
2. Atribua usuários a um grupo e, em seguida, atribua o acesso de logon único ao grupo.

   Para obter instruções, consulte Adicionar grupos no Guia do usuário do Centro de Identidade do AWS IAM .

Acesse a API AWS CLI, AWS Tools for Windows PowerShell, ou o AWS SDKs

Para usar a API, o AWS CLI AWS Tools for Windows PowerShell, ou o AWS SDKs, você deve criar chaves de acesso. Essas chaves consistem em um ID da chave de acesso e uma chave de acesso secreta usados para assinar as solicitações programáticas que você faz à AWS.

Os usuários precisam de acesso programático se quiserem interagir com pessoas AWS fora do Console de gerenciamento da AWS. A forma de conceder acesso programático depende do tipo de usuário que está acessando AWS.

Para conceder acesso programático aos usuários, selecione uma das seguintes opções:

Qual usuário precisa de acesso programático?	Para	Por
IAM	(Recomendado) Use as credenciais do console como credenciais temporárias para assinar solicitações programáticas para o AWS CLI, AWS SDKs ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para o AWS CLI, consulte Login para desenvolvimento AWS local no Guia AWS Command Line Interface do usuário. • Para AWS SDKs isso, consulte Login para desenvolvimento AWS local no Guia de referência de ferramentas AWS SDKs e ferramentas.
Identidade da força de trabalho (Usuários gerenciados no Centro de Identidade do IAM)	Use credenciais temporárias para assinar solicitações programáticas para o AWS CLI AWS SDKs, ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para o AWS CLI, consulte Configurando o AWS CLI para uso Centro de Identidade do AWS IAM no Guia do AWS Command Line Interface usuário. • Para AWS SDKs, ferramentas e AWS APIs, consulte a autenticação do IAM Identity Center no Guia de referênci

Qual usuário precisa de acesso programático?	Para	Por
		a de ferramentas AWS SDKs e ferramentas.
IAM	Use credenciais temporári as para assinar solicitações programáticas para o AWS CLI AWS SDKs, ou. AWS APIs	Siga as instruções em Como usar credenciais temporárias com AWS recursos no Guia do usuário do IAM.
IAM	(Não recomendado) Use credenciais de longo prazo para assinar solicitaç ões programáticas para o AWS CLI, AWS SDKs, ou. AWS APIs	Siga as instruções da interface que deseja utilizar. • Para isso AWS CLI, consulte Autenticação usando credenciais de usuário do IAM no Guia do AWS Command Line Interface usuário. • Para ferramentas AWS SDKs e ferramentas, consulte Autenticar usando credenciais de longo prazo no Guia de referência de ferramentas AWS SDKs e ferramentas. • Para isso AWS APIs, consulte Gerenciamento de chaves de acesso para usuários do IAM no Guia do usuário do IAM.

Configurar o AWS Command Line Interface ou AWS Tools for Windows PowerShell

O AWS Command Line Interface (AWS CLI) é uma ferramenta unificada para gerenciar AWS serviços. Para obter informações sobre como instalar e configurar o AWS CLI, consulte Instalando ou atualizando para a versão mais recente do AWS CLI no Guia do AWS Command Line Interface Usuário.

Se você tem experiência com o Windows PowerShell, talvez prefira usar AWS Tools for Windows PowerShell. Para obter mais informações, consulte Configuração do AWS Tools for Windows PowerShell no Guia do usuário do Ferramentas da AWS para PowerShell .

Baixe um AWS SDK

Se você estiver usando uma linguagem de programação que AWS fornece um SDK para, recomendamos que você use um SDK em vez da AWS Cloud Map API. Usar um SDK tem vários benefícios. SDKs simplifique a autenticação, integre-se facilmente ao seu ambiente de desenvolvimento e forneça acesso aos AWS Cloud Map comandos. Para obter mais informações, consulte Ferramentas para a Amazon Web Services.

Saiba como usar a descoberta AWS Cloud Map de serviços com consultas de DNS e chamadas de API

O tutorial a seguir simula uma arquitetura de microsserviços com dois serviços de back-end. O primeiro serviço poderá ser descoberto usando uma consulta de DNS. O segundo serviço poderá ser descoberto usando somente a AWS Cloud Map API.

> **Note**
>
> Os detalhes dos recursos, como nomes de domínio e endereços IP, são apenas para fins de simulação. Eles não podem ser resolvidos pela internet.

Para obter uma end-to-end AWS CLI versão deste tutorial, consulteSaiba como usar a descoberta AWS Cloud Map de serviços com consultas de DNS e chamadas de API usando o AWS CLI.

Pré-requisitos

Os pré-requisitos a seguir devem ser atendidos para concluir o tutorial com êxito.

- Antes de começar, conclua as etapas em Configurado para usar AWS Cloud Map.
- Se você ainda não instalou o AWS Command Line Interface, siga as etapas em Instalando ou atualizando a versão mais recente do AWS CLI para instalá-lo.

  O tutorial requer um terminal de linha de comando ou um shell para executar os comandos. No Linux e no macOS, use o gerenciador de pacotes e de shell de sua preferência.

  > **Note**
  >
  > No Windows, alguns comandos da CLI do Bash que você costuma usar com o Lambda (como `zip`) não são compatíveis com os terminais integrados do sistema operacional. Para obter uma versão do Ubuntu com o Bash integrada no Windows, instale o Subsistema do Windows para Linux.

- O tutorial requer um ambiente local com o comando `dig` DNS lookup utility.

Etapa 1: criar um AWS Cloud Map namespace

Nesta etapa, você cria um AWS Cloud Map namespace público. AWS Cloud Map cria uma zona hospedada do Route 53 em seu nome com esse mesmo nome. Isso permite que você descubra as instâncias de serviço criadas nesse namespace usando registros DNS públicos ou usando AWS Cloud Map chamadas de API.

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Escolha Create namespace (Criar namespace).
3. Para Nome do namespace, especifique. `cloudmap-tutorial.com`

   > **Note**
   >
   > Se você fosse usar isso na produção, você gostaria de garantir que especificou o nome de um domínio que você possuía ou ao qual tinha acesso. Mas, para os propósitos deste tutorial, não é necessário que seja um domínio real que esteja sendo usado.

4. (Opcional) Para a descrição do namespace, especifique uma descrição para o que você pretende usar o namespace.
5. Em Descoberta de instâncias, selecione chamadas de API e consultas públicas de DNS.
6. Deixe o resto dos valores padrão e escolha Criar namespace.

Etapa 2: criar os AWS Cloud Map serviços

Nesta etapa, você cria dois serviços. O primeiro serviço poderá ser descoberto usando chamadas públicas de DNS e API. O segundo serviço poderá ser descoberto usando somente chamadas de API.

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação esquerdo, escolha Namespaces para listar os namespaces que você criou.
3. Na lista de namespaces, selecione o **`cloudmap-tutorial.com`** namespace e escolha Exibir detalhes.
4. Na seção Serviços, escolha Criar serviço e faça o seguinte para criar o primeiro serviço.

    a. Em Nome do serviço, digite `public-service`. O nome do serviço será aplicado aos registros DNS AWS Cloud Map criados. O formato usado é *`<service-name>`*`.`*`<namespace-name>`*.

    b. Para Configuração do Service Discovery, selecione API e DNS.

    c. Na seção Configuração de DNS, em Política de roteamento, selecione Roteamento de respostas de vários valores.

    > **Note**
    >
    > O console traduzirá isso para MULTIVALUE depois de selecionado. Para obter mais informações sobre as opções de roteamento disponíveis, consulte Como escolher uma política de roteamento no Guia do desenvolvedor do Route 53.

    d. Deixe o restante dos valores padrão e escolha Criar serviço, que o levará de volta à página de detalhes do namespace.
5. Na seção Serviços, escolha Criar serviço e faça o seguinte para criar o segundo serviço.

a. Em Nome do serviço, digite `backend-service`.

b. Para Configuração do Service Discovery, selecione somente API.

c. Deixe o resto dos valores padrão e escolha Criar serviço.

Etapa 3: registrar as instâncias do AWS Cloud Map serviço

Nesta etapa, você cria duas instâncias de serviço, uma para cada serviço em nosso namespace.

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Na lista de namespaces, selecione o namespace que você criou na etapa 1 e escolha Exibir detalhes.
3. Na página de detalhes do namespace, na lista de serviços, selecione o `public-service` serviço e escolha Exibir detalhes.
4. Na seção Instâncias de serviço, escolha Registrar instância de serviço e faça o seguinte para criar a primeira instância de serviço.

   a. Para ID da instância de serviço, especifique`first`.

   b. Para IPv4 endereço, especifique`192.168.2.1`.

   c. Deixe o resto dos valores padrão e escolha Registrar instância de serviço.
5. Usando o breadcrumb na parte superior da página, selecione cloudmap-tutorial.com para voltar à página de detalhes do namespace.
6. Na página de detalhes do namespace, na lista de serviços, selecione o serviço de back-end e escolha Exibir detalhes.
7. Na seção Instâncias de serviço, escolha Registrar instância de serviço e faça o seguinte para criar a segunda instância de serviço.

   a. Em ID da instância de serviço, especifique `second` para indicar que essa é a segunda instância de serviço.

   b. Em Tipo de instância, selecione Informações de identificação para outro recurso.

   c. Para atributos personalizados, adicione um par de valores-chave `service-name` como chave e `backend` como valor.

   d. Escolha Registrar instância de serviço.

Etapa 4: descobrir as instâncias do AWS Cloud Map serviço

Agora que o AWS Cloud Map namespace, os serviços e as instâncias de serviço foram criados, você pode verificar se tudo está funcionando descobrindo as instâncias. Use o `dig` comando para verificar as configurações públicas de DNS e a AWS Cloud Map API para verificar o serviço de back-end. Para obter mais informações sobre o `dig` comando, consulte dig - DNS lookup utility.

1. Faça login no Console de gerenciamento da AWS e abra o console do Route 53 em https://console.aws.amazon.com/route53/.
2. No painel de navegação, escolha Hosted zones (Zonas hospedadas).
3. Selecione a zona hospedada do cloudmap-tutorial.com. Isso exibe os detalhes da zona hospedada em um painel separado. Anote os servidores de nomes associados à sua zona hospedada, pois os usaremos na próxima etapa.
4. Usando o comando dig e um dos servidores de nomes do Route 53 para sua zona hospedada, consulte os registros DNS da sua instância de serviço.

   ```
   dig @hosted-zone-nameserver public-service.cloudmap-tutorial.com
   ```

   O `ANSWER SECTION` na saída deve exibir o IPv4 endereço que você associou ao seu `public-service` serviço.

   ```
   ;; ANSWER SECTION:
   public-service.cloudmap-tutorial.com. 300 IN A 192.168.2.1
   ```

5. Usando o AWS CLI, consulte os atributos de suas segundas instâncias de serviço.

   ```
   aws servicediscovery discover-instances --namespace-name cloudmap-tutorial.com --
   service-name backend-service --region region
   ```

   A saída exibe os atributos que você associou ao serviço como pares de valores-chave.

   ```
   {
       "Instances": [
           {
               "InstanceId": "second",
               "NamespaceName": "cloudmap-tutorial.com",
               "ServiceName": "backend-service",
               "HealthStatus": "UNKNOWN",
               "Attributes": {
   ```

```
                "service-name": "backend"
            }
        }
    ],
    "InstancesRevision": 71462688285136850
}
```

Etapa 5: limpar os recursos

Depois de concluir o tutorial, você pode excluir os recursos. AWS Cloud Map exige que você as limpe na ordem inversa, primeiro as instâncias do serviço, depois os serviços e, finalmente, o namespace. AWS Cloud Map limpará os recursos do Route 53 em seu nome quando você seguir essas etapas.

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Na lista de namespaces, selecione o **`cloudmap-tutorial.com`** namespace e escolha Exibir detalhes.
3. Na página de detalhes do namespace, na lista de serviços, selecione o `public-service` serviço e escolha Exibir detalhes.
4. Na seção Instâncias de serviço, selecione a `first` instância e escolha Cancelar registro.
5. Usando o breadcrumb na parte superior da página, selecione cloudmap-tutorial.com para voltar à página de detalhes do namespace.
6. Na página de detalhes do namespace, na lista de serviços, selecione o serviço público e escolha Excluir.
7. Repita as etapas de 3 a 6 para o. `backend-service`
8. No painel de navegação à esquerda, escolha Namespaces.
9. Selecione o **`cloudmap-tutorial.com`** namespace e escolha Excluir.

> **Note**
>
> Embora AWS Cloud Map limpe os recursos do Route 53 em seu nome, você pode navegar até o console do Route 53 para verificar se a zona `cloudmap-tutorial.com` hospedada foi excluída.

Saiba como usar a descoberta AWS Cloud Map de serviços com consultas de DNS e chamadas de API usando o AWS CLI

Este tutorial demonstra como usar a descoberta AWS Cloud Map de serviços usando a AWS Command Line Interface (CLI). Você criará uma arquitetura de microsserviços com dois serviços de back-end — um detectável usando consultas de DNS e outro detectável usando somente a API. AWS Cloud Map

Para obter um tutorial que inclui etapas AWS Cloud Map do console, consulteSaiba como usar a descoberta AWS Cloud Map de serviços com consultas de DNS e chamadas de API.

Pré-requisitos

Os pré-requisitos a seguir devem ser atendidos para concluir o tutorial com êxito.

- Antes de começar, conclua as etapas em Configurado para usar AWS Cloud Map.
- Se você ainda não instalou o AWS Command Line Interface, siga as etapas em Instalando ou atualizando a versão mais recente do AWS CLI para instalá-lo.

  O tutorial requer um terminal de linha de comando ou um shell para executar os comandos. No Linux e no macOS, use o gerenciador de pacotes e de shell de sua preferência.

  > **Note**
  >
  > No Windows, alguns comandos da CLI do Bash que você costuma usar com o Lambda (como `zip`) não são compatíveis com os terminais integrados do sistema operacional. Para obter uma versão do Ubuntu com o Bash integrada no Windows, instale o Subsistema do Windows para Linux.

- O tutorial requer um ambiente local com o comando `dig` DNS lookup utility.

Crie um AWS Cloud Map namespace

Primeiro, você criará um AWS Cloud Map namespace público. AWS Cloud Map criará uma zona hospedada do Route 53 com o mesmo nome, permitindo a descoberta de serviços por meio de registros DNS e chamadas de API.

1. Crie o namespace DNS público:

```
aws servicediscovery create-public-dns-namespace \
    --name cloudmap-tutorial.com \
    --creator-request-id cloudmap-tutorial-request-1 \
    --region us-east-2
```

O comando retorna um ID de operação que você pode usar para verificar o status da criação do namespace:

```
{
    "OperationId": "gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9xmplyzd"
}
```

2. Verifique o status da operação para confirmar que o namespace foi criado com sucesso:

```
aws servicediscovery get-operation \
    --operation-id gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9xmplyzd \
    --region us-east-2
```

3. Quando a operação for bem-sucedida, obtenha o ID do namespace:

```
aws servicediscovery list-namespaces \
    --region us-east-2 \
    --query "Namespaces[?Name=='cloudmap-tutorial.com'].Id" \
    --output text
```

Esse comando retorna o ID do namespace, que você precisará para as etapas subsequentes:

```
ns-abcd1234xmplefgh
```

Crie os AWS Cloud Map serviços

Agora, crie dois serviços em seu namespace. O primeiro serviço poderá ser descoberto usando chamadas de DNS e de API, enquanto o segundo poderá ser descoberto usando somente chamadas de API.

1. Crie o primeiro serviço com a descoberta de DNS ativada:

```
aws servicediscovery create-service \
```

```
    --name public-service \
    --namespace-id ns-abcd1234xmplefgh \
    --dns-config "RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=300}]" \
    --region us-east-2
```

O comando retorna detalhes sobre o serviço criado:

```
{
    "Service": {
        "Id": "srv-abcd1234xmplefgh",
        "Arn": "arn:aws:servicediscovery:us-east-2:123456789012:service/srv-
abcd1234xmplefgh",
        "Name": "public-service",
        "NamespaceId": "ns-abcd1234xmplefgh",
        "DnsConfig": {
            "NamespaceId": "ns-abcd1234xmplefgh",
            "RoutingPolicy": "MULTIVALUE",
            "DnsRecords": [
                {
                    "Type": "A",
                    "TTL": 300
                }
            ]
        },
        "CreateDate": 1673613600.000,
        "CreatorRequestId": "public-service-request"
    }
}
```

2. Crie o segundo serviço com a descoberta somente de API:

```
aws servicediscovery create-service \
    --name backend-service \
    --namespace-id ns-abcd1234xmplefgh \
    --type HTTP \
    --region us-east-2
```

O comando retorna detalhes sobre o serviço criado:

```
{
    "Service": {
        "Id": "srv-ijkl5678xmplmnop",
```

```
        "Arn": "arn:aws:servicediscovery:us-east-2:123456789012:service/srv-
ijkl5678xmplmnop",
        "Name": "backend-service",
        "NamespaceId": "ns-abcd1234xmplefgh",
        "Type": "HTTP",
        "CreateDate": 1673613600.000,
        "CreatorRequestId": "backend-service-request"
    }
}
```

Registre as instâncias do AWS Cloud Map serviço

Em seguida, registre as instâncias de serviço para cada um dos seus serviços. Essas instâncias representam os recursos reais que serão descobertos.

1. Registre a primeira instância com um IPv4 endereço para descoberta de DNS:

   ```
   aws servicediscovery register-instance \
       --service-id srv-abcd1234xmplefgh \
       --instance-id first \
       --attributes AWS_INSTANCE_IPV4=192.168.2.1 \
       --region us-east-2
   ```

   O comando retorna um ID de operação:

   ```
   {
       "OperationId": "4yejorelbukcjzpnr6tlmrghsjwpngf4-k9xmplyzd"
   }
   ```

2. Verifique o status da operação para confirmar que a instância foi registrada com sucesso:

   ```
   aws servicediscovery get-operation \
       --operation-id 4yejorelbukcjzpnr6tlmrghsjwpngf4-k9xmplyzd \
       --region us-east-2
   ```

3. Registre a segunda instância com atributos personalizados para descoberta da API:

   ```
   aws servicediscovery register-instance \
       --service-id srv-ijkl5678xmplmnop \
       --instance-id second \
       --attributes service-name=backend \
   ```

```
    --region us-east-2
```

O comando retorna um ID de operação:

```
{
    "OperationId": "7zxcvbnmasdfghjklqwertyuiop1234-k9xmplyzd"
}
```

4. Verifique o status da operação para confirmar que a instância foi registrada com sucesso:

```
aws servicediscovery get-operation \
    --operation-id 7zxcvbnmasdfghjklqwertyuiop1234-k9xmplyzd \
    --region us-east-2
```

Descubra as instâncias AWS Cloud Map de serviço

Agora que você criou e registrou suas instâncias de serviço, pode verificar se tudo está funcionando descobrindo-as usando as consultas de DNS e a API. AWS Cloud Map

1. Primeiro, obtenha o ID da zona hospedada do Route 53:

```
aws route53 list-hosted-zones-by-name \
    --dns-name cloudmap-tutorial.com \
    --query "HostedZones[0].Id" \
    --output text
```

Isso retorna o ID da zona hospedada:

```
/hostedzone/Z1234ABCDXMPLEFGH
```

2. Obtenha os servidores de nomes para sua zona hospedada:

```
aws route53 get-hosted-zone \
    --id Z1234ABCDXMPLEFGH \
    --query "DelegationSet.NameServers[0]" \
    --output text
```

Isso retorna um dos servidores de nomes:

```
ns-1234.awsdns-12.org
```

3. Use o `dig` comando para consultar os registros DNS do seu serviço público:

```
dig @ns-1234.awsdns-12.org public-service.cloudmap-tutorial.com
```

A saída deve exibir o IPv4 endereço que você associou ao seu serviço:

```
;; ANSWER SECTION:
public-service.cloudmap-tutorial.com. 300 IN A 192.168.2.1
```

4. Use o AWS CLI para descobrir a instância do serviço de back-end:

```
aws servicediscovery discover-instances \
    --namespace-name cloudmap-tutorial.com \
    --service-name backend-service \
    --region us-east-2
```

A saída exibe os atributos que você associou ao serviço:

```
{
    "Instances": [
        {
            "InstanceId": "second",
            "NamespaceName": "cloudmap-tutorial.com",
            "ServiceName": "backend-service",
            "HealthStatus": "UNKNOWN",
            "Attributes": {
                "service-name": "backend"
            }
        }
    ],
    "InstancesRevision": 71462688285136850
}
```

Limpe os recursos

Depois de concluir o tutorial, limpe os recursos para evitar cobranças. AWS Cloud Map exige que você os limpe na ordem inversa: primeiro as instâncias de serviço, depois os serviços e, finalmente, o namespace.

1. Cancele o registro da primeira instância de serviço:

```
aws servicediscovery deregister-instance \
    --service-id srv-abcd1234xmplefgh \
    --instance-id first \
    --region us-east-2
```

2. Cancele o registro da segunda instância de serviço:

```
aws servicediscovery deregister-instance \
    --service-id srv-ijkl5678xmplmnop \
    --instance-id second \
    --region us-east-2
```

3. Exclua o serviço público:

```
aws servicediscovery delete-service \
    --id srv-abcd1234xmplefgh \
    --region us-east-2
```

4. Exclua o serviço de back-end:

```
aws servicediscovery delete-service \
    --id srv-ijkl5678xmplmnop \
    --region us-east-2
```

5. Exclua o namespace de :

```
aws servicediscovery delete-namespace \
    --id ns-abcd1234xmplefgh \
    --region us-east-2
```

6. Verifique se a zona hospedada do Route 53 foi excluída:

```
aws route53 list-hosted-zones-by-name \
```

```
    --dns-name cloudmap-tutorial.com
```

Saiba como usar a descoberta AWS Cloud Map de serviços com atributos personalizados

O tutorial a seguir demonstra como você pode usar a descoberta AWS Cloud Map de serviços com atributos personalizados que podem ser descobertos usando a AWS Cloud Map API. O tutorial orienta você na criação e execução de aplicativos cliente usando AWS CloudShell o. Os aplicativos usam duas funções Lambda para gravar dados em uma tabela do DynamoDB e depois ler a tabela. As funções Lambda e a tabela do DynamoDB são registradas como instâncias de serviço. AWS Cloud Map O código nos aplicativos cliente e nas funções do Lambda usa atributos AWS Cloud Map personalizados para descobrir os recursos necessários para realizar o trabalho.

Para obter uma versão AWS CLI baseada deste tutorial, consulteSaiba como usar a descoberta AWS Cloud Map de serviços com atributos personalizados usando o AWS CLI.

> **Important**
>
> Você criará AWS recursos durante o workshop, o que acarretará um custo em sua AWS conta. É recomendável limpar os recursos assim que terminar o workshop para minimizar o custo.

Pré-requisitos

Antes de começar, conclua as etapas em Configurado para usar AWS Cloud Map.

Etapa 1: criar um AWS Cloud Map namespace

Nesta etapa, você cria um AWS Cloud Map namespace. Um namespace é uma construção usada para agrupar serviços para um aplicativo. Ao criar o namespace, você especifica como os recursos serão descobertos. Os recursos criados no namespace criado nesta etapa poderão ser descobertos com chamadas de AWS Cloud Map API usando atributos personalizados.

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Escolha Create namespace (Criar namespace).

3. Para Nome do namespace, especifique. `cloudmap-tutorial`
4. (Opcional) Para a descrição do namespace, especifique uma descrição para o que você pretende usar o namespace.
5. Em Descoberta de instâncias, selecione Chamadas de API.
6. Deixe o resto dos valores padrão e escolha Criar namespace.

Etapa 2: criar uma tabela do DynamoDB

Nesta etapa, você cria uma tabela do DynamoDB. A tabela é usada para armazenar e recuperar dados para o aplicativo de amostra que você criará nas etapas a seguir.

Para obter informações sobre como criar um DynamoDB, consulte Etapa 1: Criar uma tabela no DynamoDB no DynamoDB Developer Guide e use a tabela a seguir para determinar quais opções especificar.

Opção	Valor
Nome da tabela	mapa da nuvem
Chave de partição	id

Mantenha os valores padrão para o restante das configurações e crie a tabela.

Etapa 3: criar um serviço de AWS Cloud Map dados e registrar a tabela do DynamoDB como uma instância

Nessa etapa, você cria um AWS Cloud Map serviço e depois registra a tabela do DynamoDB criada na última etapa como uma instância de serviço.

1. Abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/
2. Na lista de namespaces, selecione o **`cloudmap-tutorial`** namespace e escolha Exibir detalhes.
3. Na seção Serviços, escolha Criar serviço e faça o seguinte.
    a. Em Nome do serviço, digite `data-service`.

   b. Deixe o resto dos valores padrão e escolha Criar serviço.
4. Na seção Serviços, selecione o `data-service` serviço e escolha Exibir detalhes.
5. Na seção Instâncias de serviço, escolha Registrar instância de serviço.
6. Na página Registrar instância do serviço, faça o seguinte.

   a. Em Tipo de instância, selecione Informações de identificação para outro recurso.
   b. Para ID da instância de serviço, especifique`data-instance`.
   c. Na seção Atributos personalizados, especifique o seguinte par de valores-chave: chave =`tablename`, valor =. `cloudmap`

Etapa 4: criar uma função AWS Lambda de execução

Nesta etapa, você cria uma função do IAM que a AWS Lambda função na próxima etapa usa. Você pode nomear a função do IAM `cloudmap-tutorial-role` e omitir o limite de permissões porque a função é usada somente neste tutorial, e você pode excluí-la posteriormente.

Para criar a função de serviço para o Lambda (console do IAM)

1. Faça login no Console de gerenciamento da AWS e abra o console do IAM em https://console.aws.amazon.com/iam/.
2. No painel de navegação do console do IAM, escolha Perfis e, em seguida, Criar perfil.
3. Em Tipo de entidade confiável, escolha AWS service (Serviço da AWS).
4. Para Serviço ou caso de uso, escolha Lambda e, em seguida, escolha o caso de uso do Lambda.
5. Escolha Próximo.
6. Pesquise e selecione a caixa ao lado da `PowerUserAccess` política e escolha Avançar.
7. Escolha Próximo.
8. Em Nome da função, especifique`cloudmap-tutorial-role`.
9. Reveja a função e escolha Criar função.

Etapa 5: criar a função Lambda para gravar dados

Nesta etapa, você cria uma função Lambda criada do zero que grava dados na tabela do DynamoDB usando a API para consultar o AWS Cloud Map serviço que você criou. AWS Cloud Map

Para obter informações sobre como criar uma função Lambda, consulte Criar uma função Lambda com o console no Guia do AWS Lambda desenvolvedor e use a tabela a seguir para determinar quais opções especificar ou escolher.

Opção	Valor
Nome da função	função de gravação
Runtime	Python 3.12
Arquitetura	x86_64
Permissões	Use uma função existente
Função existente	cloudmap-tutorial-role

Depois de criar a função, atualize o código de exemplo para refletir o código Python a seguir e, em seguida, implante a função. Observe que você está especificando o atributo `datatable` personalizado associado à instância de AWS Cloud Map serviço criada para a tabela do DynamoDB. A função gera uma chave que é um número aleatório entre 1 e 100 e a associa a um valor que é passado para a função quando ela é chamada.

```
import json
import boto3
import random

def lambda_handler(event, context):

    serviceclient = boto3.client('servicediscovery')

    response = serviceclient.discover_instances(
        NamespaceName='cloudmap-tutorial',
        ServiceName='data-service')

    tablename = response["Instances"][0]["Attributes"]["tablename"]

    dynamodbclient = boto3.resource('dynamodb')

    table = dynamodbclient.Table(tablename)
```

```
    response = table.put_item(
        Item={ 'id': str(random.randint(1,100)), 'todo': event })

    return {
        'statusCode': 200,
        'body': json.dumps(response)
    }
```

Depois de implantar a função, para evitar erros de tempo limite, atualize o tempo limite da função para 5 segundos. Para obter mais informações, consulte Configurar tempo limite da função do Lambda no Guia do desenvolvedor do AWS Lambda .

Etapa 6: criar um serviço de AWS Cloud Map aplicativo e registrar a função de gravação do Lambda como uma instância

Nesta etapa, você cria um AWS Cloud Map serviço e depois registra a função de gravação do Lambda como uma instância de serviço.

1. Abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/
2. No painel de navegação à esquerda, escolha Namespaces.
3. Na lista de namespaces, selecione o **`cloudmap-tutorial`** namespace e escolha Exibir detalhes.
4. Na seção Serviços, escolha Criar serviço e faça o seguinte.
   a. Em Nome do serviço, digite `app-service`.
   b. Deixe o resto dos valores padrão e escolha Criar serviço.
5. Na seção Serviços, selecione o `app-service` serviço e escolha Exibir detalhes.
6. Na seção Instâncias de serviço, escolha Registrar instância de serviço.
7. Na página Registrar instância do serviço, faça o seguinte.
   a. Em Tipo de instância, selecione Informações de identificação para outro recurso.
   b. Para ID da instância de serviço, especifique`write-instance`.
   c. Na seção Atributos personalizados, especifique os seguintes pares de valores-chave.
      - chave =`action`, valor = `write`
      - chave =`functionname`, valor = `writefunction`

Etapa 7: criar a função Lambda para ler dados

Nesta etapa, você cria uma função Lambda criada do zero que grava dados na tabela do DynamoDB que você criou.

Para obter informações sobre como criar uma função Lambda, consulte Criar uma função Lambda com o console no Guia do AWS Lambda desenvolvedor e use a tabela a seguir para determinar quais opções especificar ou escolher.

Opção	Valor
Nome da função	função de leitura
Runtime	Python 3.12
Arquitetura	x86_64
Permissões	Use uma função existente
Função existente	cloudmap-tutorial-role

Depois de criar a função, atualize o código de exemplo para refletir o código Python a seguir e, em seguida, implante a função. A função escaneia a tabela e retorna todos os itens.

```
import json
import boto3

def lambda_handler(event, context):
    serviceclient = boto3.client('servicediscovery')

    response = serviceclient.discover_instances(NamespaceName='cloudmap-tutorial',
 ServiceName='data-service')

    tablename = response["Instances"][0]["Attributes"]["tablename"]

    dynamodbclient = boto3.resource('dynamodb')

    table = dynamodbclient.Table(tablename)

    response = table.scan(Select='ALL_ATTRIBUTES')
```

```
    return {
        'statusCode': 200,
        'body': json.dumps(response)
    }
```

Depois de implantar a função, para evitar erros de tempo limite, atualize o tempo limite da função para 5 segundos. Para obter mais informações, consulte Configurar tempo limite da função do Lambda no Guia do desenvolvedor do AWS Lambda .

Etapa 8: registrar a função de leitura do Lambda como uma AWS Cloud Map instância de serviço

Nesta etapa, você registra a função de leitura do Lambda como uma instância de serviço no `app-service` serviço que você criou anteriormente.

1. Abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/
2. No painel de navegação à esquerda, escolha Namespaces.
3. Na lista de namespaces, selecione o **`cloudmap-tutorial`** namespace e escolha Exibir detalhes.
4. Na seção Serviços, selecione o `app-service` serviço e escolha Exibir detalhes.
5. Na seção Instâncias de serviço, escolha Registrar instância de serviço.
6. Na página Registrar instância do serviço, faça o seguinte.

    a. Em Tipo de instância, selecione Informações de identificação para outro recurso.

    b. Para ID da instância de serviço, especifique`read-instance`.

    c. Na seção Atributos personalizados, especifique os seguintes pares de valores-chave.

    - chave =`action`, valor = `read`
    - chave =`functionname`, valor = `readfunction`

Etapa 9: criar e executar clientes de leitura e gravação no AWS CloudShell

Você pode criar e executar aplicativos cliente AWS CloudShell que usam código para descobrir os serviços nos quais você configurou AWS Cloud Map e fazer chamadas para esses serviços.

1. Abra o AWS CloudShell console em https://console.aws.amazon.com/cloudshell/
2. Use o comando a seguir para criar um arquivo chamado`writefunction.py`.

```
vim writeclient.py
```

3. No `writeclient.py` arquivo, entre no modo de inserção pressionando o `i` botão. Em seguida, copie e cole o código a seguir. Esse código descobre a função Lambda para gravar dados pesquisando o `name=writeservice` atributo personalizado no `app-service` serviço. O nome da função Lambda responsável por gravar dados na tabela do DynamoDB é retornado. Em seguida, a função Lambda é invocada, passando uma amostra de carga útil que é gravada na tabela como um valor.

```
import boto3

serviceclient = boto3.client('servicediscovery')

response = serviceclient.discover_instances(NamespaceName='cloudmap-tutorial',
 ServiceName='app-service', QueryParameters={ 'action': 'write' })

functionname = response["Instances"][0]["Attributes"]["functionname"]

lambdaclient = boto3.client('lambda')

resp = lambdaclient.invoke(FunctionName=functionname, Payload='"This is a test
 data"')

print(resp["Payload"].read())
```

4. Pressione a tecla escape `:wq`, digite e pressione a tecla enter para salvar o arquivo e sair.
5. Use o comando a seguir para executar o código Python.

```
python3 writeclient.py
```

A saída deve ser uma `200` resposta, semelhante à seguinte.

```
b'{"statusCode": 200, "body": "{\\"ResponseMetadata\\": {\\"RequestId\\": \
\"Q0M038IT0BPBVBJK8OCKK6I6M7VV4KQNSO5AEMVJF66Q9ASUAAJG\\", \\"HTTPStatusCode\
\": 200, \\"HTTPHeaders\\": {\\"server\\": \\"Server\\", \\"date\\": \\"Wed, 06
 Mar 2024 22:46:09 GMT\\", \\"content-type\\": \\"application/x-amz-json-1.0\\",
 \\"content-length\\": \\"2\\", \\"connection\\": \\"keep-alive\\", \\"x-amzn-
requestid\\": \\"Q0M038IT0BPBVBJK8OCKK6I6M7VV4KQNSO5AEMVJF66Q9ASUAAJG\\", \\"x-amz-
crc32\\": \\"2745614147\\"}, \\"RetryAttempts\\": 0}}"}'
```

6. Para verificar se a gravação foi bem-sucedida na etapa anterior, crie um cliente de leitura.

   a. Use o comando a seguir para criar um arquivo chamado`readfunction.py`.

   ```
   vim readclient.py
   ```

   b. No `readclient.py` arquivo, pressione o `i` botão para entrar no modo de inserção. Em seguida, copie e cole o código a seguir. Esse código escaneia a tabela e retornará o valor que você gravou na tabela na etapa anterior.

   ```
   import boto3

   serviceclient = boto3.client('servicediscovery')

   response = serviceclient.discover_instances(NamespaceName='cloudmap-tutorial',
    ServiceName='app-service', QueryParameters={ 'action': 'read' })

   functionname = response["Instances"][0]["Attributes"]["functionname"]

   lambdaclient = boto3.client('lambda')

   resp = lambdaclient.invoke(FunctionName=functionname,
    InvocationType='RequestResponse')

   print(resp["Payload"].read())
   ```

   c. Pressione a tecla escape`:wq`, digite e pressione a tecla enter para salvar o arquivo e sair.

   d. Use o comando a seguir para executar o código Python.

   ```
   python3 readclient.py
   ```

   A saída deve ser semelhante à seguinte, listando o valor gravado na tabela pela execução `writefunction.py` e a chave aleatória gerada na função de gravação do Lambda.

   ```
     b'{"statusCode": 200, "body": "{\\"Items\\": [{\\"id\\": \\"45\
   \", \\"todo\\": \\"This is a test data\\"}], \\"Count\\": 1, \
   \"ScannedCount\\": 1, \\"ResponseMetadata\\": {\\"RequestId\\": \
   \"9JF8J6SFQCKR6IDT5JG5NOM3CNVV4KQNSO5AEMVJF66Q9ASUAAJG\\", \\"HTTPStatusCode\
   \": 200, \\"HTTPHeaders\\": {\\"server\\": \\"Server\\", \\"date\\": \\"Thu, 25
    Jul 2024 20:43:33 GMT\\", \\"content-type\\": \\"application/x-amz-json-1.0\
   \", \\"content-length\\": \\"91\\", \\"connection\\": \\"keep-alive\\", \\"x-
   ```

```
amzn-requestid\\": \\"9JF8J6SFQCKR6IDT5JG5NOM3CNVV4KQNSO5AEMVJF66Q9ASUAAJG\\",
 \\"x-amz-crc32\\": \\"1163081893\\"}, \\"RetryAttempts\\": 0}}"}'
```

Etapa 10: limpar os recursos

Depois de concluir o tutorial, exclua os recursos para evitar cobranças adicionais. AWS Cloud Map exige que você as limpe na ordem inversa, primeiro as instâncias do serviço, depois os serviços e, finalmente, o namespace. As etapas a seguir orientam você na limpeza dos AWS Cloud Map recursos usados no tutorial.

Para excluir os AWS Cloud Map recursos

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Na lista de namespaces, selecione o **`cloudmap-tutorial`** namespace e escolha Exibir detalhes.
3. Na página de detalhes do namespace, na lista de serviços, selecione o `data-service` serviço e escolha Exibir detalhes.
4. Na seção Instâncias de serviço, selecione a `data-instance` instância e escolha Cancelar registro.
5. Usando o breadcrumb na parte superior da página, selecione cloudmap-tutorial.com para voltar à página de detalhes do namespace.
6. Na página de detalhes do namespace, na lista de serviços, selecione o serviço de serviços de dados e escolha Excluir.
7. Repita as etapas de 3 a 6 para o `app-service` serviço `write-instance` e as instâncias `read-instance` de serviço.
8. No painel de navegação à esquerda, escolha Namespaces.
9. Selecione o **`cloudmap-tutorial`** namespace e escolha Excluir.

A tabela a seguir lista os procedimentos que você pode seguir para excluir os outros recursos usados no tutorial.

Recurso	Etapas
Tabela DynamoDB	Etapa 6: (Opcional) Exclua sua tabela do DynamoDB para limpar os recursos no Amazon DynamoDB Developer Guide
Funções Lambda e função de execução do IAM associada	Limpe no Guia do AWS Lambda desenvolvedor

Saiba como usar a descoberta AWS Cloud Map de serviços com atributos personalizados usando o AWS CLI

Este tutorial demonstra como você pode usar a descoberta AWS Cloud Map de serviços com atributos personalizados. Você criará um aplicativo de microsserviços usado AWS Cloud Map para descobrir recursos dinamicamente usando atributos personalizados. O aplicativo consiste em duas funções Lambda que gravam e lêem dados em uma tabela do DynamoDB, com todos os recursos registrados em. AWS Cloud Map

Para obter uma Console de gerenciamento da AWS versão do tutorial, consulteSaiba como usar a descoberta AWS Cloud Map de serviços com atributos personalizados.

Pré-requisitos

Antes de começar este tutorial, conclua as etapas emConfigurado para usar AWS Cloud Map.

Crie um AWS Cloud Map namespace

Um namespace é uma construção usada para agrupar serviços para um aplicativo. Nesta etapa, você criará um namespace que permite que os recursos sejam descobertos por meio AWS Cloud Map de chamadas de API.

1. Execute o comando a seguir para criar um namespace HTTP:

```
aws servicediscovery create-http-namespace \
  --name cloudmap-tutorial \
  --creator-request-id cloudmap-tutorial-request
```

O comando retorna um ID de operação. Você pode verificar o status da operação com o seguinte comando:

```
aws servicediscovery get-operation \
  --operation-id operation-id
```

2. Depois que o namespace for criado, você poderá recuperar seu ID para uso nos comandos subsequentes:

```
aws servicediscovery list-namespaces \
  --query "Namespaces[?Name=='cloudmap-tutorial'].Id" \
  --output text
```

3. Armazene o ID do namespace em uma variável para uso posterior:

```
NAMESPACE_ID=$(aws servicediscovery list-namespaces \
  --query "Namespaces[?Name=='cloudmap-tutorial'].Id" \
  --output text)
```

Criar uma tabela do DynamoDB

Em seguida, crie uma tabela do DynamoDB que armazenará dados para seu aplicativo:

1. Execute o comando a seguir para criar a tabela:

```
aws dynamodb create-table \
  --table-name cloudmap \
  --attribute-definitions AttributeName=id,AttributeType=S \
  --key-schema AttributeName=id,KeyType=HASH \
  --billing-mode PAY_PER_REQUEST
```

2. Aguarde até que a tabela fique ativa antes de continuar:

```
aws dynamodb wait table-exists --table-name cloudmap
```

Esse comando espera até que a tabela esteja totalmente criada e pronta para uso.

Crie um serviço de AWS Cloud Map dados e registre a tabela do DynamoDB

Agora, crie um serviço em seu namespace para representar os recursos de armazenamento de dados:

1. Execute o comando a seguir para criar um AWS Cloud Map serviço para recursos de armazenamento de dados:

```
aws servicediscovery create-service \
  --name data-service \
  --namespace-id $NAMESPACE_ID \
  --creator-request-id data-service-request
```

2. Obtenha o ID do serviço de dados:

```
DATA_SERVICE_ID=$(aws servicediscovery list-services \
  --query "Services[?Name=='data-service'].Id" \
  --output text)
```

3. Registre a tabela do DynamoDB como uma instância de serviço com um atributo personalizado que especifica o nome da tabela:

```
aws servicediscovery register-instance \
  --service-id $DATA_SERVICE_ID \
  --instance-id data-instance \
  --attributes tablename=cloudmap
```

   O atributo personalizado `tablename=cloudmap` permite que outros serviços descubram dinamicamente o nome da tabela do DynamoDB.

Crie uma função do IAM para funções Lambda

Crie uma função do IAM que as funções do Lambda usarão para acessar AWS recursos:

1. Crie o documento de política de confiança para a função do IAM usando o seguinte JSON:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Execute o comando a seguir para criar a função do IAM usando a política de confiança:

```
aws iam create-role \
  --role-name cloudmap-tutorial-role \
  --assume-role-policy-document file://lambda-trust-policy.json
```

3. Crie um arquivo para uma política personalizada do IAM com menos permissões de privilégio usando o seguinte JSON:

   JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents"
      ],
      "Resource": "arn:aws:logs:*:*:*"
    },
    {
      "Effect": "Allow",
```

```
      "Action": [
        "servicediscovery:DiscoverInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "dynamodb:PutItem",
        "dynamodb:Scan"
      ],
      "Resource": "arn:aws:dynamodb:*:*:table/cloudmap"
    }
  ]
}
```

4. Crie e anexe a política à função do IAM:

```
aws iam create-policy \
  --policy-name CloudMapTutorialPolicy \
  --policy-document file://cloudmap-policy.json

POLICY_ARN=$(aws iam list-policies \
  --query "Policies[?PolicyName=='CloudMapTutorialPolicy'].Arn" \
  --output text)

aws iam attach-role-policy \
  --role-name cloudmap-tutorial-role \
  --policy-arn $POLICY_ARN

aws iam attach-role-policy \
  --role-name cloudmap-tutorial-role \
  --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole
```

Crie a função Lambda para gravar dados

Para criar uma função Lambda que grava dados na tabela do DynamoDB, siga estas etapas:

1. Crie o arquivo Python para a função de gravação:

```
cat > writefunction.py << EOF
import json
```

```
import boto3
import random

def lambda_handler(event, context):
    try:
        serviceclient = boto3.client('servicediscovery')

        response = serviceclient.discover_instances(
            NamespaceName='cloudmap-tutorial',
            ServiceName='data-service')

        if not response.get("Instances"):
            return {
                'statusCode': 500,
                'body': json.dumps({"error": "No instances found"})
            }

        tablename = response["Instances"][0]["Attributes"].get("tablename")
        if not tablename:
            return {
                'statusCode': 500,
                'body': json.dumps({"error": "Table name attribute not found"})
            }

        dynamodbclient = boto3.resource('dynamodb')

        table = dynamodbclient.Table(tablename)

        # Validate input
        if not isinstance(event, str):
            return {
                'statusCode': 400,
                'body': json.dumps({"error": "Input must be a string"})
            }

        response = table.put_item(
            Item={ 'id': str(random.randint(1,100)), 'todo': event })

        return {
            'statusCode': 200,
            'body': json.dumps(response)
        }
    except Exception as e:
        return {
```

```
            'statusCode': 500,
            'body': json.dumps({"error": str(e)})
        }
EOF
```

Essa função é usada AWS Cloud Map para descobrir o nome da tabela do DynamoDB a partir do atributo personalizado e, em seguida, grava dados na tabela.

2. Package e implante a função Lambda:

```
zip writefunction.zip writefunction.py

ROLE_ARN=$(aws iam get-role --role-name cloudmap-tutorial-role \
  --query 'Role.Arn' --output text)

aws lambda create-function \
  --function-name writefunction \
  --runtime python3.12 \
  --role $ROLE_ARN \
  --handler writefunction.lambda_handler \
  --zip-file fileb://writefunction.zip \
  --architectures x86_64
```

3. Atualize o tempo limite da função para evitar erros de tempo limite:

```
aws lambda update-function-configuration \
  --function-name writefunction \
  --timeout 5
```

Crie um serviço de AWS Cloud Map aplicativo e registre a função de gravação do Lambda

Para criar outro serviço em seu namespace para representar as funções do aplicativo, siga estas etapas:

1. Crie um serviço para funções do aplicativo:

```
aws servicediscovery create-service \
  --name app-service \
  --namespace-id $NAMESPACE_ID \
```

```
  --creator-request-id app-service-request
```

2. Obtenha o ID do serviço do aplicativo:

```
APP_SERVICE_ID=$(aws servicediscovery list-services \
  --query "Services[?Name=='app-service'].Id" \
  --output text)
```

3. Registre a função de gravação do Lambda como uma instância de serviço com atributos personalizados:

```
aws servicediscovery register-instance \
  --service-id $APP_SERVICE_ID \
  --instance-id write-instance \
  --attributes action=write,functionname=writefunction
```

   Os atributos personalizados `functionname=writefunction` permitem que `action=write` os clientes descubram essa função com base em sua finalidade.

Crie a função Lambda para ler dados

Para criar uma função Lambda que leia dados da tabela do DynamoDB, siga estas etapas:

1. Crie o arquivo Python para a função de leitura:

```
cat > readfunction.py << EOF
import json
import boto3

def lambda_handler(event, context):
    try:
        serviceclient = boto3.client('servicediscovery')

        response = serviceclient.discover_instances(
            NamespaceName='cloudmap-tutorial',
            ServiceName='data-service')

        if not response.get("Instances"):
            return {
                'statusCode': 500,
                'body': json.dumps({"error": "No instances found"})
```

```
            }

        tablename = response["Instances"][0]["Attributes"].get("tablename")
        if not tablename:
            return {
                'statusCode': 500,
                'body': json.dumps({"error": "Table name attribute not found"})
            }

        dynamodbclient = boto3.resource('dynamodb')

        table = dynamodbclient.Table(tablename)

        # Use pagination for larger tables
        response = table.scan(
            Select='ALL_ATTRIBUTES',
            Limit=50  # Limit results for demonstration purposes
        )

        # For production, you would implement pagination like this:
        # items = []
        # while 'LastEvaluatedKey' in response:
        #     items.extend(response['Items'])
        #     response = table.scan(
        #         Select='ALL_ATTRIBUTES',
        #         ExclusiveStartKey=response['LastEvaluatedKey']
        #     )
        # items.extend(response['Items'])

        return {
            'statusCode': 200,
            'body': json.dumps(response)
        }
    except Exception as e:
        return {
            'statusCode': 500,
            'body': json.dumps({"error": str(e)})
        }
EOF
```

Essa função também é usada AWS Cloud Map para descobrir o nome da tabela do DynamoDB e depois ler os dados da tabela. Inclui tratamento de erros e comentários de paginação.

2. Package e implante a função Lambda:

```
zip readfunction.zip readfunction.py

aws lambda create-function \
  --function-name readfunction \
  --runtime python3.12 \
  --role $ROLE_ARN \
  --handler readfunction.lambda_handler \
  --zip-file fileb://readfunction.zip \
  --architectures x86_64
```

3. Atualize o tempo limite da função:

```
aws lambda update-function-configuration \
  --function-name readfunction \
  --timeout 5
```

Registre a função de leitura do Lambda como uma instância de serviço

Para registrar a função de leitura do Lambda como outra instância de serviço no serviço de aplicativo, siga esta etapa:

```
aws servicediscovery register-instance \
  --service-id $APP_SERVICE_ID \
  --instance-id read-instance \
  --attributes action=read,functionname=readfunction
```

Os atributos personalizados `functionname=readfunction` permitem que `action=read` os clientes descubram essa função com base em sua finalidade.

Crie e execute aplicativos cliente

Para criar um aplicativo cliente Python que use AWS Cloud Map para descobrir e invocar a função de gravação, siga estas etapas:

1. Crie um arquivo Python para o aplicativo cliente de gravação:

```
cat > writeclient.py << EOF
import boto3
import json
```

```
try:
    serviceclient = boto3.client('servicediscovery')

    print("Discovering write function...")
    response = serviceclient.discover_instances(
        NamespaceName='cloudmap-tutorial',
        ServiceName='app-service',
        QueryParameters={ 'action': 'write' }
    )

    if not response.get("Instances"):
        print("Error: No instances found")
        exit(1)

    functionname = response["Instances"][0]["Attributes"].get("functionname")
    if not functionname:
        print("Error: Function name attribute not found")
        exit(1)

    print(f"Found function: {functionname}")

    lambdaclient = boto3.client('lambda')

    print("Invoking Lambda function...")
    resp = lambdaclient.invoke(
        FunctionName=functionname,
        Payload='"This is a test data"'
    )

    payload = resp["Payload"].read()
    print(f"Response: {payload.decode('utf-8')}")

except Exception as e:
    print(f"Error: {str(e)}")
EOF
```

Esse cliente usa a `QueryParameters` opção de encontrar instâncias de serviço com o `action=write` atributo.

2. Crie um arquivo Python para o aplicativo cliente de leitura:

```
cat > readclient.py << EOF
```

```
import boto3
import json

try:
    serviceclient = boto3.client('servicediscovery')

    print("Discovering read function...")
    response = serviceclient.discover_instances(
        NamespaceName='cloudmap-tutorial',
        ServiceName='app-service',
        QueryParameters={ 'action': 'read' }
    )

    if not response.get("Instances"):
        print("Error: No instances found")
        exit(1)

    functionname = response["Instances"][0]["Attributes"].get("functionname")
    if not functionname:
        print("Error: Function name attribute not found")
        exit(1)

    print(f"Found function: {functionname}")

    lambdaclient = boto3.client('lambda')

    print("Invoking Lambda function...")
    resp = lambdaclient.invoke(
        FunctionName=functionname,
        InvocationType='RequestResponse'
    )

    payload = resp["Payload"].read()
    print(f"Response: {payload.decode('utf-8')}")

except Exception as e:
    print(f"Error: {str(e)}")
EOF
```

3. Execute o cliente de gravação para adicionar dados à tabela do DynamoDB:

```
python3 writeclient.py
```

A saída deve mostrar uma resposta bem-sucedida com o código de status HTTP 200.

4. Execute o cliente de leitura para recuperar dados da tabela do DynamoDB:

```
python3 readclient.py
```

A saída deve mostrar os dados que foram gravados na tabela, incluindo o ID gerado aleatoriamente e o valor “Isto é um dado de teste”.

Limpar os recursos

Ao concluir o tutorial, limpe os recursos para evitar cobranças adicionais.

1. Primeiro, execute o comando a seguir para cancelar o registro das instâncias do serviço:

```
aws servicediscovery deregister-instance \
  --service-id $APP_SERVICE_ID \
  --instance-id read-instance

aws servicediscovery deregister-instance \
  --service-id $APP_SERVICE_ID \
  --instance-id write-instance

aws servicediscovery deregister-instance \
  --service-id $DATA_SERVICE_ID \
  --instance-id data-instance
```

2. Execute o comando a seguir para excluir os serviços:

```
aws servicediscovery delete-service \
  --id $APP_SERVICE_ID

aws servicediscovery delete-service \
  --id $DATA_SERVICE_ID
```

3. Execute o comando a seguir para excluir o namespace:

```
aws servicediscovery delete-namespace \
  --id $NAMESPACE_ID
```

4. Execute o comando a seguir para excluir as funções do Lambda:

```
aws lambda delete-function --function-name writefunction
aws lambda delete-function --function-name readfunction
```

5. Execute o comando a seguir para excluir a função e a política do IAM:

```
aws iam detach-role-policy \
  --role-name cloudmap-tutorial-role \
  --policy-arn $POLICY_ARN

aws iam detach-role-policy \
  --role-name cloudmap-tutorial-role \
  --policy-arn arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole

aws iam delete-policy \
  --policy-arn $POLICY_ARN

aws iam delete-role --role-name cloudmap-tutorial-role
```

6. Execute o comando a seguir para excluir a tabela do DynamoDB:

```
aws dynamodb delete-table --table-name cloudmap
```

7. Execute o comando a seguir para limpar arquivos temporários:

```
rm -f lambda-trust-policy.json cloudmap-policy.json writefunction.py
 readfunction.py writefunction.zip readfunction.zip writeclient.py readclient.py
```

AWS Cloud Map namespaces

Um namespace é uma entidade lógica usada para agrupar AWS Cloud Map os serviços de um aplicativo sob um nome comum e um nível de descoberta. Ao criar um namespace, você especifica o seguinte:

- Um nome que você deseja que seu aplicativo use para descobrir instâncias.
- O método pelo qual as instâncias de serviço nas quais você se registra AWS Cloud Map podem ser descobertas. Você pode decidir se seus recursos precisam ser descobertos publicamente pela Internet, de forma privada em uma nuvem privada virtual (VPC) específica ou somente por chamadas de API.

A seguir estão os conceitos gerais sobre namespaces.

- Os namespaces são específicos do local em Região da AWS que foram criados. Para usar AWS Cloud Map em várias regiões, você precisará criar namespaces em cada região.
- Se você criar um namespace para permitir, por exemplo, a descoberta por consultas de DNS em uma VPC, cria AWS Cloud Map automaticamente uma zona hospedada privada do Route 53. Essa zona hospedada pode ser associada a várias VPCs. Para obter mais informações, consulte Associate VPCWith HostedZone in the Amazon Route 53 API Reference.

Tópicos

Criação de um AWS Cloud Map namespace para agrupar serviços de aplicativos

Você pode criar um namespace para agrupar serviços para seu aplicativo com um nome amigável que permita a descoberta de recursos do aplicativo por meio de chamadas de API ou consultas de DNS.

Opções de descoberta de instâncias

A tabela a seguir resume as diferentes opções de descoberta de instâncias AWS Cloud Map e o tipo de namespace correspondente que você pode criar, dependendo dos serviços e da configuração do seu aplicativo.

Tipo de namespace	Método de descoberta de instâncias	Como funciona	Mais informações
HTTP	Chamadas de API	Os recursos em seu aplicativo podem descobrir outros recursos chamando somente a `DiscoverInstances` API.	• DiscoverInstances • CreateHttpNamespace
DNS privado	Chamadas de API e consultas de DNS em uma VPC	Quando você cria um namespace DNS privado, AWS Cloud Map cria uma zona hospedada privada correspondente do Amazon Route 53. Os recursos em seu aplicativo podem descobrir outros recursos chamando a `DiscoverInstances` API e consultando os servidores de nomes na zona hospedada privada do Route 53 que é criada automaticamente. AWS Cloud Map	• DiscoverInstances • CreatePrivateDnsNamespace

Tipo de namespace	Método de descoberta de instâncias	Como funciona	Mais informações
		A zona hospedada criada por AWS Cloud Map tem o mesmo nome do namespace e contém registros DNS com nomes no formato. *service-name namespace-name*. **Note** O resolvedor do Route 53 resolve consultas ao DNS originadas na VPC usando registros na zona hospedada privada. Se a zona hospedada privada não incluir um registro que corresponda ao nome do domínio em uma consulta ao DNS, o Route 53	

Tipo de namespace	Método de descoberta de instâncias	Como funciona	Mais informações
		responderá à consulta com `NXDOMAIN` (domínio inexistente).	

Tipo de namespace	Método de descoberta de instâncias	Como funciona	Mais informações
DNS público	API calls and public DNS queries (Chamadas à API e consultas DNS públicas)	Quando você cria um namespace DNS público, AWS Cloud Map cria uma zona hospedada pública correspondente do Amazon Route 53. Os recursos em seu aplicativo podem descobrir outros recursos chamando a `DiscoverInstances` API e consultando os servidores de nomes na zona hospedada pública do Route 53 que é criada automaticamente. AWS Cloud Map A zona hospedada pública tem o mesmo nome do namespace e contém registros DNS com nomes no formato. *`service-name namespace-name`*. **Note** O nome do namespace,	• DiscoverInstances • CreatePublicDnsNamespace

Tipo de namespace	Método de descoberta de instâncias	Como funciona	Mais informações
		nesse caso, deve ser um nome de domínio que você registrou .	

Procedimento

Você pode seguir essas etapas para criar um namespace usando o AWS CLI, Console de gerenciamento da AWS, ou o SDK para Python.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Escolha Create namespace (Criar namespace).
3. Em Nome do namespace, insira um nome que será usado para descobrir instâncias.

> **Note**
>
> - Os namespaces configurados para consultas públicas de DNS devem terminar com um domínio de nível superior. Por exemplo, `..com`
> - É possível especificar um internationalized domain name (IDN - nome de domínio internacionalizado) se você converter o nome em Punycode primeiro. Para obter informações sobre conversores online, pesquise “conversor punycode” na Internet.
>
>   Você também pode converter um nome de domínio internacionalizado em Punycode ao criar namespaces de forma programática. Por exemplo, se você estiver usando Java, poderá converter um valor Unicode em Punycode usando o método `toASCII` da biblioteca java.net.IDN.

4. (Opcional) Em Descrição do namespace, insira as informações sobre o namespace que estarão visíveis na página Namespaces e em Namespace information. Você pode usar essas informações para identificar facilmente um namespace.
5. Para a descoberta de instâncias, você pode escolher entre chamadas de API, chamadas de API e consultas de DNS em VPCs, e chamadas de API e consultas de DNS público para criar um namespace HTTP, DNS privado ou DNS público, respectivamente. Para obter mais informações, consulte Opções de descoberta de instâncias.

   Com base na sua seleção, siga estas etapas.

   - Se você escolher chamadas de API e consultas de DNS em VPCs, para VPC, escolha uma nuvem privada virtual (VPC) à qual você deseja associar o namespace.
   - Se você escolher chamadas de API e consultas de DNS em VPCs ou chamadas de API e consultas públicas de DNS, para TTL, especifique um valor numérico em segundos. O valor de vida útil (TTL) determina por quanto tempo os resolvedores de DNS armazenam em cache as informações do registro DNS de início de autoridade (SOA) da zona hospedada do Route 53 criada com seu namespace. Para obter mais informações sobre TTL, consulte TTL (segundos) no Guia do desenvolvedor do Amazon Route 53.
6. (Opcional) Em Tags, escolha Adicionar tags e especifique uma chave e um valor para marcar seu namespace. É possível especificar uma ou mais tags para adicionar ao seu namespace . As tags permitem que você categorize seus AWS recursos para que você possa gerenciá-los com mais facilidade. Para obter mais informações, consulte Marcando seus recursos AWS Cloud Map.
7. Escolha Create namespace (Criar namespace). Você pode visualizar o status da operação usando ListOperations. Para obter mais informações, consulte ListOperationsa Referência AWS Cloud Map da API

AWS CLI

- Crie um namespace com o comando para o tipo de descoberta de instância que você preferir (substitua os *red* valores pelos seus).

  - Criar um namespace HTTP usando `create-http-namespace`. As instâncias de serviço que você registra usando um namespace HTTP podem ser descobertas usando uma solicitação `DiscoverInstances`, mas não podem ser detectadas usando DNS.

```
aws servicediscovery create-http-namespace --name name-of-namespace
```

- Cria um namespace privado com base no DNS, que será visível apenas dentro de uma Amazon VPC especificada usando `create-private-dns-namespace`. É possível descobrir instâncias que foram registradas com um namespace de DNS público utilizando uma solicitação `DiscoverInstances` ou usando o DNS

```
aws servicediscovery create-private-dns-namespace --name name-of-namespace --
vpc vpc-xxxxxxxxx
```

- Cria um namespace público baseado em DNS, que é visível na Internet usando `create-public-dns-namespace`. É possível descobrir instâncias que foram registradas com um namespace de DNS público utilizando uma solicitação `DiscoverInstances` ou usando o DNS.

```
aws servicediscovery create-public-dns-namespace --name name-of-namespace
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Crie um namespace com o comando para o tipo de descoberta de instância que você preferir (substitua os *red* valores pelos seus):

- Criar um namespace HTTP usando `create_http_namespace()`. As instâncias de serviço que você registra usando um namespace HTTP podem ser descobertas usando uma solicitação `discover_instances()`, mas não podem ser detectadas usando DNS.

```
response = client.create_http_namespace(
    Name='name-of-namespace',
)
# If you want to see the response
```

```
print(response)
```

- Cria um namespace privado com base no DNS, que será visível apenas dentro de uma Amazon VPC especificada usando `create_private_dns_namespace()`. É possível descobrir instâncias que foram registradas com um namespace de DNS público utilizando uma solicitação `discover_instances()` ou usando o DNS

```
response = client.create_private_dns_namespace(
    Name='name-of-namespace',
    Vpc='vpc-1c56417b',
)
# If you want to see the response
print(response)
```

- Cria um namespace público baseado em DNS, que é visível na Internet usando `create_public_dns_namespace()`. É possível descobrir instâncias que foram registradas com um namespace de DNS público utilizando uma solicitação `discover_instances()` ou usando o DNS.

```
response = client.create_public_dns_namespace(
    Name='name-of-namespace',
)
# If you want to see the response
print(response)
```

- Exemplo de objeto de resposta

```
{
    'OperationId': 'gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9302yzd',
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Próximas etapas

Depois de criar um namespace, você pode criar serviços no namespace para agrupar recursos do aplicativo que, coletivamente, servem a uma finalidade específica em seu aplicativo. Um serviço atua como um modelo para registrar recursos do aplicativo como instâncias. Para obter mais informações

sobre a criação AWS Cloud Map de serviços, consulteCriação de um AWS Cloud Map serviço para um componente do aplicativo.

Listando AWS Cloud Map namespaces

Depois de criar namespaces, você pode ver uma lista dos namespaces que você criou seguindo estas etapas.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces para ver uma lista de namespaces. Você pode ordenar namespaces por nome, descrição, modo de descoberta da instância, proprietário ou ID do namespace. Você também pode inserir um nome ou ID de namespace no campo de pesquisa para localizar e visualizar um namespace específico.

AWS CLI

- Liste os namespaces com o comando `list-namespaces`.

```
aws servicediscovery list-namespaces
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Liste os namespaces com `list_namespaces()`.

```
response = client.list_namespaces()
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'Namespaces': [
        {
            'Arn': 'arn:aws::servicediscovery:us-west-2:123456789012:namespace/
ns-xxxxxxxxxxxxxx',
            'CreateDate': 1585354387.357,
            'Id': 'ns-xxxxxxxxxxxxxx',
            'Name': 'myFirstNamespace',
            'Properties': {
                'DnsProperties': {
                    'HostedZoneId': 'Z06752353VBUDTC32S84S',
                },
                'HttpProperties': {
                    'HttpName': 'myFirstNamespace',
                },
            },
            'Type': 'DNS_PRIVATE',
        },
        {
            'Arn': 'arn:aws::servicediscovery:us-west-2:123456789012:namespace/
ns-xxxxxxxxxxxxxx',
            'CreateDate': 1586468974.698,
            'Description': 'My second namespace',
            'Id': 'ns-xxxxxxxxxxxxxx',
            'Name': 'mySecondNamespace.com',
            'Properties': {
                'DnsProperties': {
                },
                'HttpProperties': {
                    'HttpName': 'mySecondNamespace.com',
                },
            },
            'Type': 'HTTP',
        },
        {
            'Arn': 'arn:aws::servicediscovery:us-west-2:123456789012:namespace/
ns-xxxxxxxxxxxxxx',
            'CreateDate': 1587055896.798,
            'Id': 'ns-xxxxxxxxxxxxxx',
            'Name': 'myThirdNamespace.com',
            'Properties': {
```

```
                'DnsProperties': {
                    'HostedZoneId': 'Z09983722P0QME1B3KC8I',
                },
                'HttpProperties': {
                    'HttpName': 'myThirdNamespace.com',
                },
            },
            'Type': 'DNS_PRIVATE',
        },
    ],
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Excluindo um namespace AWS Cloud Map

Depois de terminar de usar um namespace, você pode excluí-lo. Ao excluir um namespace, você não poderá mais usá-lo para registrar ou descobrir instâncias de serviço.

> **Note**
>
> Quando você exclui um namespace DNS, AWS Cloud Map exclui a zona hospedada correspondente do Amazon Route 53 criada durante a criação do namespace.

Antes de excluir um namespace, você deve cancelar o registro de todas as instâncias de serviço e, em seguida, excluir todos os serviços que foram criados no namespace. Para obter mais informações, consulte Cancelando o registro de uma instância de serviço AWS Cloud Map e Excluindo um serviço AWS Cloud Map.

Depois de cancelar o registro de instâncias e excluir serviços que foram criados em um namespace, siga estas etapas para excluir o namespace.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.

3. Selecione o namespace que você deseja excluir e escolha Excluir.
4. Confirme que você deseja excluir o serviço escolhendo Excluir novamente.

AWS CLI

- Exclua um namespace com o `delete-namespace` comando (substitua o *red* valor pelo seu). Se o namespace ainda contiver um ou mais serviços, a solicitação falhará.

```
aws servicediscovery delete-namespace --id ns-xxxxxxxxxxx
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Exclua um namespace com `delete_namespace()` (substitua o *red* valor pelo seu). Se o namespace ainda contiver um ou mais serviços, a solicitação falhará.

```
response = client.delete_namespace(
    Id='ns-xxxxxxxxxxx',
)
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'OperationId': 'gv4g5meo7ndmeh4fqskygvk23d2fijwa-k98y6drk',
    'ResponseMetadata': {
        '...': '...',
    },
}
```

AWS Cloud Map Namespaces compartilhados

AWS Cloud Map permite que os proprietários de namespaces compartilhem seus namespaces com outras pessoas Contas da AWS ou dentro de uma organização AWS Organizations para simplificar a descoberta de serviços entre contas e o registro de serviços. Isso facilita o uso de namespaces gerenciados por outras pessoas Contas da AWS ou por equipes dentro de uma AWS organização.

AWS Cloud Map integra-se com AWS Resource Access Manager (AWS RAM) para permitir o compartilhamento de recursos. AWS RAM é um serviço que permite que você compartilhe alguns AWS Cloud Map recursos com outras pessoas Contas da AWS ou por meio de AWS Organizations. Com AWS RAM, você compartilha recursos de sua propriedade criando um compartilhamento de recursos. Um compartilhamento de atributos especifica os atributos a serem compartilhados, e os consumidores com os quais compartilhá-los. Os consumidores podem incluir:

- Específico Contas da AWS dentro de sua organização em AWS Organizations
- Uma unidade organizacional dentro de sua organização em AWS Organizations
- Toda a sua organização em AWS Organizations

Para obter mais informações sobre AWS RAM, consulte o Guia AWS RAM do usuário.

Este tópico explica como compartilhar recursos que você possui e como usar os recursos que são compartilhados com você.

Conteúdo

Considerações sobre o compartilhamento de namespaces

- Para compartilhar um namespace, você deve possuí-lo em seu. Conta da AWS Isso significa que o recurso deve ser alocado ou provisionado em sua conta. Você não pode compartilhar um namespace que tenha sido compartilhado com você.
- Para compartilhar um namespace com sua organização ou unidade organizacional em AWS Organizations, você deve habilitar o compartilhamento com. AWS Organizations Para obter mais informações, consulte Habilitar o compartilhamento com o AWS Organizations no Guia do usuário do AWS RAM .
- Para a descoberta de serviços usando consultas de DNS em um namespace DNS privado compartilhado, o proprietário do namespace precisará ligar `create-vpc-association-authorization` com o ID da zona hospedada privada associada ao namespace e à VPC do consumidor.

  ```
  aws route53 create-vpc-association-authorization --hosted-zone-id Z1234567890ABC --
  vpc VPCRegion=us-east-1,VPCId=vpc-12345678
  ```

  O consumidor do namespace precisará chamar `associate-vpc-with-hosted-zone` com o ID da zona hospedada privada.

  ```
  aws route53 associate-vpc-with-hosted-zone --hosted-zone-id Z1234567890ABC --vpc
   VPCRegion=us-east-1,VPCId=vpc-12345678
  ```

  Para obter mais informações, consulte Associando uma Amazon VPC e uma zona hospedada privada que você criou com Contas da AWS diferentes no Guia do desenvolvedor do Amazon Route 53.
- Depois de descobrir os locais de up-to-date rede dos serviços associados a um namespace DNS compartilhado, talvez seja necessário configurar a conectividade entre VPCs para se comunicar com os serviços, caso estejam em locais diferentes. VPCs Isso pode ser feito usando uma conexão de emparelhamento da VPC. Para obter mais informações, consulte Criar ou excluir uma conexão de emparelhamento da VPC no Guia de emparelhamento da Amazon Virtual Private Cloud (VPC).
- Você não pode usar `ListOperations` para listar operações em namespaces compartilhados que são executadas por outras contas.
- A marcação não é compatível com namespaces compartilhados.

Compartilhando um AWS Cloud Map namespace

Ao compartilhar um AWS Cloud Map namespace que você possui com outros Contas da AWS (consumidores), você permite que essas contas descubram os locais de up-to-date rede dos serviços no namespace sem a necessidade de credenciais temporárias.

Para compartilhar um namespace, você deve adicioná-lo a um compartilhamento de recursos. Um compartilhamento de recursos é um recurso do AWS RAM que permite que você compartilhe seus recursos entre Contas da AWS. Um compartilhamento de recursos especifica os recursos a serem compartilhados, e os clientes com os quais compartilhá-los. Para adicionar o namespace a um novo compartilhamento de recursos, primeiro você deve criar o compartilhamento de recursos usando o AWS RAM console.

Se você faz parte de uma organização AWS Organizations e o compartilhamento dentro de sua organização está ativado, os consumidores em sua organização recebem automaticamente acesso ao namespace compartilhado. Caso contrário, os consumidores receberão um convite para participar do compartilhamento de recursos e terão acesso ao namespace compartilhado após aceitarem o convite.

Você pode compartilhar um namespace de sua propriedade usando o AWS RAM console ou o. AWS CLI

AWS RAM console

Para compartilhar um namespace que você possui usando o console AWS RAM

Consulte Criar um compartilhamento de recursos no AWS RAM no Guia do usuário do AWS RAM .

AWS CLI

Para compartilhar um namespace que você possui usando o AWS CLI

Use o comando AWS RAM create-resource-share.

Pare de compartilhar um AWS Cloud Map namespace

Quando um namespace não é mais compartilhado, o namespace e quaisquer serviços e instâncias associados a ele não podem mais ser acessados pelo consumidor. Contas da AWS Isso inclui recursos criados no namespace pelos consumidores quando eles tiveram acesso ao namespace.

Para parar de compartilhar um namespace de sua propriedade, você deve removê-lo do compartilhamento de recursos. Você pode fazer isso usando o AWS RAM console ou AWS CLI o.

AWS RAM console

Para parar de compartilhar um namespace que você possui usando o console AWS RAM

Consulte Atualização de um compartilhamento de atributos no Guia do usuário do AWS RAM .

AWS CLI

Para parar de compartilhar um namespace que você possui usando o AWS CLI

Use o comando disassociate-resource-share.

Identificação de um AWS Cloud Map namespace compartilhado

Proprietários e consumidores podem identificar namespaces compartilhados usando o AWS Cloud Map console e. AWS CLI O proprietário do namespace pode ser identificado usando a `ResourceOwner` propriedade. O Conta da AWS que cria um serviço ou registra uma instância no namespace compartilhado pode ser identificado usando a propriedade. `CreatedByAccount`

AWS Cloud Map console

Para identificar um namespace compartilhado usando o console AWS Cloud Map

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. Na página Namespaces, em Resource Owner, você pode encontrar o ID do proprietário do Conta da AWS namespace.
3. Escolha o nome de domínio do namespace que você deseja identificar.
4. Na *`namespace-name`* página Namespace:, na seção Informações do namespace, em Proprietário do recurso, você pode encontrar o ID do proprietário do Conta da AWS namespace.

AWS CLI

Para identificar um namespace compartilhado usando o AWS CLI, use o comando list-namespaces. O comando retorna os namespaces que você possui e os namespaces que são

compartilhados com você. O `ResourceOwner` campo mostra o ID da AWS conta do proprietário do namespace.

A `list-namespaces` chamada a seguir é feita por conta`111122223333`.

```
aws servicediscovery list-namespaces
```

Saída:

```
{
    "Namespaces": [
        {
            "Arn": "arn:aws:servicediscovery:us-west-2:111122223333:namespace/ns-
abcdef01234567890",
            "CreateDate": 1585354387.357,
            "Id": "ns-abcdef01234567890",
            "Name": "local",
            "Properties": {
                "DnsProperties": {
                    "HostedZoneId": "Z06752353VBUDTC32S84S"
                },
                "HttpProperties": {
                    "HttpName": "local"
                 }
            },
            "Type": "DNS_PRIVATE",
            "ServiceCount": 2,
           "ResourceOwner": "111122223333"
        },
        {
            "Arn": "arn:aws:servicediscovery:us-west-2:444455556666:namespace/
ns-021345abcdef6789",
            "CreateDate": 1586468974.698,
            "Description": "Shared second namespace",
            "Id": "ns-021345abcdef6789",
            "Name": "My-second-namespace",
            "Properties": {
                "DnsProperties": {},
                "HttpProperties": {
                    "HttpName": "Shared-second-namespace"
                }
            },
            "Type": "HTTP",
```

```
            "ServiceCount": 0,
            "ResourceOwner": "444455556666"
        }
    ]
}
```

Nesse cenário, o namespace `ns-abcdef01234567890` é criado e de propriedade de e o namespace `ns-021345abcdef6789` é criado `111122223333` e de propriedade de. `444455556666` O namespace `ns-021345abcdef6789` é compartilhado com conta `111122223333` por conta. `444455556666`

Conceder permissões para compartilhar um namespace

É necessário um conjunto mínimo de permissões para que um diretor do IAM compartilhe um namespace. Recomendamos usar as políticas `AWSCloudMapFullAccess` `AWSResourceAccessManagerFullAccess` gerenciadas para garantir que seus diretores do IAM tenham as permissões necessárias para compartilhar e usar namespaces compartilhados.

Se você usa uma política personalizada do IAM, as `servicediscovery:DeleteResourcePolicy` ações `servicediscovery:PutResourcePolicyservicediscovery:GetResourcePolicy`, e são necessárias para compartilhar namespaces. Essas são ações do IAM realizadas somente com permissão. Se um diretor do IAM não tiver essas permissões concedidas, ocorrerá um erro ao tentar compartilhar o namespace usando. AWS RAM

Para obter mais informações sobre como AWS RAM usa o IAM, consulte Como AWS RAM usa o IAM no Guia AWS RAM do usuário.

Responsabilidades e permissões para namespaces compartilhados

O proprietário e o consumidor do namespace podem realizar ações diferentes em um namespace compartilhado.

Permissões para proprietários

O proprietário de um namespace pode realizar as seguintes ações em um namespace compartilhado:

- Acesse serviços associados ao namespace, incluindo serviços criados por contas de consumidores e instâncias registradas nesses serviços.

- Revogue o acesso ao namespace, incluindo o acesso a serviços criados por contas de consumidores e instâncias registradas nesses serviços.
- Configure permissões para que outras contas registrem e cancelem o registro de instâncias em serviços criados no namespace compartilhado pelos consumidores ou pelo proprietário do namespace.
- Exclua serviços e cancele o registro de instâncias, incluindo serviços criados e instâncias registradas por contas de consumidores.
- Atualize ou exclua um namespace compartilhado.

Permissões para clientes

Um consumidor de namespace pode realizar as seguintes ações em um namespace compartilhado:

- Crie e exclua serviços no namespace.
- Registre e cancele o registro de instâncias em serviços criados no namespace.
- Descubra instâncias registradas em serviços criados no namespace.

Um consumidor não pode atualizar ou excluir um namespace compartilhado. Depois de perder o acesso ao namespace compartilhado, as contas do consumidor também perderão o acesso aos serviços que criaram no namespace.

Faturamento e medição

Os proprietários são cobrados por todas as instâncias registradas no namespace compartilhado e por todas as verificações de saúde do Route 53 criadas quando eles registram essas instâncias. Os consumidores são cobrados por todas as instâncias registradas no namespace e por todas as verificações de saúde do Route 53 criadas ao registrar essas instâncias. Se o namespace compartilhado for um namespace DNS, o proprietário do namespace será cobrado pelos registros DNS do Route 53 que são criados quando os serviços são criados no namespace. Os proprietários são cobrados por todas `DiscoverInstances` as `DiscoverInstancesRevision` chamadas que fizerem. Os consumidores são cobrados por todas `DiscoverInstances` as `DiscoverInstancesRevision` chamadas que fizerem.

Cotas

Os namespaces compartilhados contam somente para a cota de namespaces do proprietário do namespace por região. As instâncias registradas por um consumidor no namespace compartilhado

contam para a cota de instâncias por namespace do proprietário. Se um consumidor criar um serviço em um namespace compartilhado, todas as instâncias registradas no serviço contam para as instâncias do consumidor por cota de serviço. Se um proprietário criar um serviço em um namespace compartilhado, todas as instâncias registradas no serviço contam para as instâncias do proprietário por cota de serviço.

AWS Cloud Map serviços

Um AWS Cloud Map serviço é um modelo para registrar instâncias de serviço que consiste no nome do serviço e na configuração de DNS, se aplicável, para o serviço. Você também pode configurar uma verificação de saúde para determinar o status de integridade das instâncias no serviço e filtrar recursos não íntegros. Um serviço pode representar um componente do seu aplicativo. Por exemplo, você pode criar um serviço para recursos que gerenciam pagamentos em seu aplicativo e outro para recursos que gerenciam usuários.

Um serviço permite que você localize os recursos de um aplicativo recuperando um ou mais endpoints que podem ser usados para se conectar ao recurso. A localização dos recursos é feita usando consultas de DNS ou a ação da AWS Cloud Map `DiscoverInstances`API, dependendo de como você configurou o namespace. Você pode usar o AWS Cloud Map console para definir o escopo da descoberta de instâncias no nível do serviço.

Você também pode especificar metadados personalizados como atributos no nível do serviço usando a API. `UpdateServiceAttributes` Você pode definir atributos de serviço para evitar a duplicação de atributos entre instâncias e modificar esses atributos sem precisar fazer alterações nos atributos da instância. As informações que você pode especificar como atributos no nível de serviço incluem, mas não estão limitadas ao seguinte:

- Pesos de endpoint para deslocar o tráfego durante implantações progressivas.
- Preferências de serviço, como tempos limite de API e políticas de repetição sugeridas.

Para obter mais informações, consulte UpdateServiceAttributesa referência AWS Cloud Map da API.

Os tópicos a seguir descrevem a verificação de integridade e as configurações de DNS para serviços e incluem instruções para criar, listar, atualizar e excluir um serviço.

Tópicos

AWS Cloud Map configuração de verificação de integridade do serviço

As verificações de saúde ajudam a determinar se as instâncias do serviço estão íntegras ou não. Se você não configurar uma verificação de saúde durante a criação do serviço, o tráfego será roteado para as instâncias do serviço, independentemente do status de integridade das instâncias. Quando você configura uma verificação de saúde, AWS Cloud Map retorna recursos íntegros por padrão. Você pode usar o `HealthStatus` parâmetro da `DiscoverInstances` API para filtrar recursos por status de saúde e obter uma lista de recursos não íntegros. Você também pode usar a `GetInstancesHealthStatus` API para recuperar o status de integridade de uma instância de serviço específica.

Você pode configurar uma verificação de saúde do Route 53 ou uma verificação de saúde personalizada de terceiros ao criar um AWS Cloud Map serviço.

Verificações de integridade do Route 53

Se você especificar configurações para uma verificação de saúde do Amazon Route 53, AWS Cloud Map cria uma verificação de saúde do Route 53 sempre que você registra uma instância e exclui a verificação de saúde ao cancelar o registro da instância.

Para namespaces DNS públicos, AWS Cloud Map associa a verificação de saúde ao registro do Route 53 AWS Cloud Map criado quando você registra uma instância. Se você especificar ambos `A` e os tipos de `AAAA` registro na configuração de DNS de um serviço, AWS Cloud Map cria uma verificação de saúde que usa o IPv4 endereço para verificar a integridade do recurso. Se o endpoint especificado pelo IPv4 endereço não estiver íntegro, o Route 53 considerará que os `AAAA` registros `A` e não estão íntegros. Se você especificar um tipo de `CNAME` registro na configuração de DNS de um serviço, não poderá configurar uma verificação de integridade do Route 53.

Para namespaces que você usa chamadas à API para descobrir instâncias, o AWS Cloud Map cria uma verificação de integridade do Route 53. No entanto, não há registro DNS ao qual AWS Cloud Map associar a verificação de saúde. Para determinar se uma verificação de saúde está íntegra, você pode configurar o monitoramento usando o console do Route 53 ou usando a Amazon CloudWatch. Para obter mais informações sobre como usar o console do Route 53, consulte Receber notificação quando uma verificação de integridade apresentar falha no Guia do desenvolvedor Amazon Route 53. Para obter mais informações sobre o uso CloudWatch, consulte PutMetricAlarmna Amazon CloudWatch API Reference.

> **Note**
>
> - Você não pode configurar uma verificação de saúde do Amazon Route 53 para um serviço criado em um namespace DNS privado.
> - Um verificador de saúde do Route 53 em cada verificação de saúde Região da AWS envia uma solicitação de verificação de saúde para um endpoint a cada 30 segundos. Em média, seu endpoint recebe uma solicitação de verificação de integridade a cada dois segundos. Porém, os verificadores de integridade não se coordenam uns com os outros. Portanto, pode haver um período de várias solicitações em um segundo, seguido de outro período de alguns segundos sem qualquer verificação de integridade. Para obter uma lista das regiões de verificação de saúde, consulte Regiões.

Para obter informações sobre as cobranças de verificações de integridade, consulte do Route 53, consulte Preço do Route 53.

Verificações de integridade personalizadas

Se você configurar AWS Cloud Map para usar uma verificação de saúde personalizada ao registrar uma instância, deverá usar um verificador de saúde terceirizado para avaliar a integridade dos seus recursos. As verificações de integridade personalizadas são úteis nas seguintes circunstâncias:

- Você não pode usar uma verificação de integridade do Route 53 porque o recurso não está disponível pela Internet. Por exemplo, suponha que você tenha uma instância localizada em uma Amazon VPC. Você poderá usar uma verificação de integridade personalizada para essa instância. No entanto, para que a verificação de integridade funcione, seu verificador de integridade também deverá estar na mesma VPC da sua instância.
- Você deseja usar um verificador de integridade de terceiros, independentemente de onde os recursos estão.

Quando você usa uma verificação de saúde personalizada, AWS Cloud Map não verifica diretamente a integridade de um determinado recurso. Em vez disso, o verificador de saúde terceirizado verifica a integridade do recurso e retorna um status ao seu aplicativo. Em seguida, sua inscrição precisará enviar uma `UpdateInstanceCustomHealthStatus` solicitação que retransmita esse status para AWS Cloud Map. Se o status inicial retransmitido for`UNHEALTHY`, e se não houver outro `UpdateInstanceCustomHealthStatus` em 30 segundos que retransmita um status de`HEALTHY`,

o recurso será confirmado como não íntegro. AWS Cloud Map interrompe o roteamento do tráfego para esse recurso.

AWS Cloud Map configuração de DNS do serviço

Quando você cria um serviço em um namespace que oferece suporte à descoberta de instâncias por consultas de DNS, AWS Cloud Map cria registros DNS do Route 53. Você deve especificar uma política de roteamento do Route 53 e um tipo de registro DNS que se aplicarão a todos os registros DNS do Route 53 criados. AWS Cloud Map

Política de roteamento

Uma política de roteamento determina como o Route 53 responde às consultas de DNS que são usadas para a descoberta de instâncias de serviço. As políticas de roteamento suportadas e como elas se relacionam AWS Cloud Map são as seguintes.

Roteamento ponderado

O Route 53 retorna o valor aplicável de uma instância de AWS Cloud Map serviço selecionada aleatoriamente dentre as instâncias que você registrou usando o mesmo AWS Cloud Map serviço. Todos os registros têm o mesmo peso. Portanto, você não pode rotear mais ou menos tráfego para nenhuma instância.

Por exemplo, suponha que o serviço inclua configurações para um registro A e uma verificação de integridade e você use o serviço para registrar dez instâncias. O Route 53 responde às consultas de DNS com o endereço IP de uma instância selecionada aleatoriamente entre as instâncias íntegras. Se nenhuma instância estiver íntegra, o Route 53 responderá às consultas ao DNS como se todas as instâncias estivessem íntegras.

Se você não definir uma verificação de integridade para o serviço, o Route 53 pressuporá que todas as instâncias estão íntegras e retornará o valor aplicável para uma instância selecionada aleatoriamente.

Para mais informações, consulte Roteamento ponderado no Guia do desenvolvedor do Amazon Route 53.

Roteamento de resposta com vários valores

Se você definir uma verificação de integridade para o serviço e a verificação de integridade for íntegra, o Route 53 retornará o valor aplicável para até oito instâncias.

Por exemplo, suponha que o serviço inclua configurações para um registro A e uma verificação de integridade. Você usa o serviço para registrar dez instâncias. O Route 53 responderá às consultas ao DNS com endereços IP de até oito instâncias íntegras. Se menos que oito instâncias estiverem íntegras, o Route 53 responderá a cada consulta ao DNS com os endereços IP de todas as instâncias íntegras.

Se você não definir uma verificação de integridade para o serviço, o Route 53 pressuporá que todas as instâncias estão íntegras e retornará os valores aplicáveis para até oito instâncias.

Para obter mais informações, consulte Roteamento por resposta com vários valores no Guia do desenvolvedor do Amazon Route 53.

Tipo de registro

Um tipo de registro DNS do Route 53 determina o tipo de valor que o Route 53 retorna em resposta às consultas de DNS que são usadas para a descoberta da instância de serviço. Os diferentes tipos de registro DNS que você pode especificar e os valores associados retornados pelo Route 53 em resposta às consultas são os seguintes.

A

Se você especificar esse tipo, o Route 53 retornará o endereço IP do recurso em IPv4 formato, como 192.0.2.44.

AAAA

Se você especificar esse tipo, o Route 53 retornará o endereço IP do recurso em IPv6 formato, como 2001:0 db 8:85 a 3:0000:0000:abcd: 0001:2345.

CNAME

Se você especificar esse tipo, o Route 53 retornará o nome de domínio do recurso (como www.exemplo.com).

> **Note**
>
> - Para configurar um registro DNS CNAME, você deve especificar a política de roteamento de roteamento ponderado.
> - Ao configurar um registro DNS CNAME, você não pode configurar uma verificação de saúde do Route 53.

SRV

Se você especificar esse tipo, o Route 53 retornará o valor de um SRV registro. O valor de um registro de SRV usa os seguintes valores:

`priority weight port service-hostname`

Considere o seguinte:

- Os valores de `priority` e `weight` são definidos como 1 e não podem ser alterados.
- Para`port`, AWS Cloud Map usa o valor que você especifica para Port (AWS_INSTANCE_PORT) ao registrar uma instância.
- O valor de `service-hostname` é uma concatenação dos seguintes valores:
  - O valor que você especifica para o ID da instância de serviço (instanceID) ao registrar uma instância
  - O nome do serviço
  - O nome do namespace

  Por exemplo, suponha que você especifique test como um ID de instância ao registrar uma instância. O nome do serviço é backend e o nome do namespace é example.com. O AWS Cloud Map atribui o seguinte valor ao atributo `service-hostname` no registro de SRV:

  `test.backend.example.com`

> **Note**
>
> Se você especificar valores como um IPv4 endereço, um IPv6 endereço ou ambos ao registrar uma instância, cria AWS Cloud Map automaticamente registros A and/or AAAA que têm o mesmo nome do valor do **`service-hostname`** registro SRV.

Você pode especificar tipos de registro nas seguintes combinações:

- A
- AAAA
- A e AAAA
- CNAME
- SRV

Se você especificar os tipos de registro A e AAAA, poderá especificar um endereço IPv4 IPv6 IP, um endereço IP ou ambos ao registrar uma instância.

Criação de um AWS Cloud Map serviço para um componente do aplicativo

Depois de criar um namespace, você pode criar serviços para representar diferentes componentes do seu aplicativo que atendem a propósitos específicos. Por exemplo, você pode criar um serviço para recursos em seu aplicativo que processam pagamentos.

> **Note**
>
> Você não pode criar vários serviços acessíveis por consultas de DNS com nomes que diferem apenas por maiúsculas e minúsculas (como EXEMPLO e EXEMPLO). Tentar fazer isso resultará em que esses serviços tenham o mesmo nome de DNS. Se o namespace só puder ser acessado por chamadas à API, é possível criar serviços com nomes que diferem apenas por maiúsculas e minúsculas.

Siga estas etapas para criar um serviço usando o Console de gerenciamento da AWS, AWS CLI, e SDK para Python.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Na página Namespaces, escolha o namespace ao qual você deseja adicionar o serviço.
4. Na *`namespace-name`* página Namespace:, escolha Criar serviço.
5. Em Nome do serviço, insira um nome que descreva as instâncias que você registra ao usar esse serviço. O valor é usado para descobrir instâncias AWS Cloud Map de serviço em chamadas de API ou em consultas de DNS.

> **Note**
>
> Se você quiser AWS Cloud Map criar um registro SRV ao registrar uma instância e estiver usando um sistema que exige um formato SRV específico (como HAProxy), especifique o seguinte para Nome do serviço:
>
> - Comece o nome com um sublinhado (_), por exemplo, _exampleservice.
> - Termine o nome com *._protocol*, por exemplo. _tcp.
>
> Quando você registra uma instância, AWS Cloud Map cria um registro SRV e atribui um nome concatenando o nome do serviço e o nome do namespace, por exemplo: _exampleservice._tcp.example.com

6. (Opcional) Em Descrição do serviço, insira uma descrição para o serviço. A descrição que você insere aqui aparece na página Serviços e na página de detalhes de cada serviço.
7. Se o namespace oferecer suporte a consultas de DNS, em Configuração de descoberta de serviços, você poderá configurar a capacidade de descoberta no nível do serviço. Escolha entre permitir chamadas de API e consultas de DNS ou somente chamadas de API para a descoberta de instâncias nesse serviço.

> **Note**
>
> Se você escolher chamadas de API, não AWS Cloud Map criará registros SRV ao registrar uma instância.

Se você escolher API e DNS, siga estas etapas para configurar os registros DNS. Você pode adicionar ou remover registros DNS.

1. Em Política de roteamento, selecione a política de roteamento do Amazon Route 53 para os registros DNS AWS Cloud Map criados quando você registra instâncias. Você pode selecionar entre roteamento ponderado e roteamento de respostas de vários valores. Para obter mais informações, consulte Política de roteamento.

> **Note**
>
> Você não pode usar o console para configurar AWS Cloud Map a criação de um registro de alias do Route 53 ao registrar uma instância. Se você quiser AWS Cloud Map criar registros de alias para um balanceador de carga do Elastic Load Balancing ao registrar instâncias programaticamente, escolha Roteamento ponderado para a política de roteamento.

2. Em Tipo de registro, escolha o tipo de registro DNS que determina o que o Route 53 retorna em resposta às consultas de DNS. AWS Cloud Map Para obter mais informações, consulte Tipo de registro.
3. Para TTL, especifique um valor numérico para definir o valor do tempo de vida (TTL), em segundos, no nível do serviço. O valor de TTL determina por quanto tempo os resolvedores de DNS armazenam informações desse registro em cache antes que os resolvedores encaminhem outra consulta ao DNS para o Amazon Route 53 para obter as configurações atualizadas.

8. Em Configuração da verificação de integridade, em Opções de verificação de integridade, escolha o tipo de verificação de saúde aplicável às instâncias de serviço. Você pode optar por não configurar nenhuma verificação de saúde ou escolher entre uma verificação de saúde do Route 53 ou uma verificação de saúde externa para suas instâncias. Para obter mais informações, consulte AWS Cloud Map configuração de verificação de integridade do serviço.

> **Note**
>
> As verificações de integridade do Route 53 são configuráveis somente para serviços em namespaces DNS públicos.

Se você escolher as verificações de saúde do Route 53, forneça as seguintes informações.

1. Para Limite de falha, forneça um número entre 1 e 10 que defina o número de verificações de saúde consecutivas do Route 53 que uma instância de serviço deve passar ou falhar para que seu status de saúde mude.
2. Em Health check protocol, selecione o método que o Route 53 usará para verificar a integridade das instâncias do serviço.

3. Se você escolher o protocolo de verificação de saúde HTTP ou HTTPS, em Health check path, forneça um caminho que você deseja que o Amazon Route 53 solicite ao realizar verificações de saúde. O caminho pode ser qualquer valor, como o arquivo `/docs/route53-health-check.html`. Quando o recurso está íntegro, o valor retornado é um código de status HTTP em formato 2xx ou 3xx. Você também pode incluir parâmetros de strings de consulta, por exemplo, `/welcome.html?language=jp&login=y`. O console do AWS Cloud Map adiciona automaticamente um caractere de barra (/) à esquerda.

Para obter mais informações sobre as verificações de saúde do Route 53, consulte Como o Amazon Route 53 determina se uma verificação de saúde está íntegra no Guia do desenvolvedor do Amazon Route 53.

9. (Opcional) Em Tags, escolha Adicionar tags e especifique uma chave e um valor para marcar seu namespace. É possível especificar uma ou mais tags para adicionar ao seu namespace . As tags permitem que você categorize seus AWS recursos para que você possa gerenciá-los com mais facilidade. Para obter mais informações, consulte Marcando seus recursos AWS Cloud Map.
10. Escolha Create service.

AWS CLI

- Crie um serviço com o `create-service` comando. Substitua *red* os valores pelos seus.

```
aws servicediscovery create-service \
    --name service-name \
    --namespace-id  ns-xxxxxxxxxxx \
    --dns-config "NamespaceId=ns-
xxxxxxxxxxx,RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=60}]"
```

Saída:

```
{
        "Service": {
        "Id": "srv-xxxxxxxxxxx",
        "Arn": "arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
xxxxxxxxxxx",
        "Name": "service-name",
        "NamespaceId": "ns-xxxxxxxxxxx",
        "DnsConfig": {
```

```
            "NamespaceId": "ns-xxxxxxxxxxx",
            "RoutingPolicy": "MULTIVALUE",
            "DnsRecords": [
                {
                    "Type": "A",
                    "TTL": 60
                }
            ]
        },
        "CreateDate": 1587081768.334,
        "CreatorRequestId": "567c1193-6b00-4308-bd57-ad38a8822d25"
    }
}
```

AWS SDK for Python (Boto3)

Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.

1. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

2. Crie um serviço com`create_service()`. Substitua *red* os valores pelos seus. Para obter mais informações, consulte create_service.

```
response = client.create_service(
    DnsConfig={
        'DnsRecords': [
            {
                'TTL': 60,
                'Type': 'A',
            },
        ],
        'NamespaceId': 'ns-xxxxxxxxxxx',
        'RoutingPolicy': 'MULTIVALUE',
    },
    Name='service-name',
    NamespaceId='ns-xxxxxxxxxxx',
)
```

Exemplo de objeto de resposta

```
{
    'Service': {
        'Arn': 'arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
xxxxxxxxxx',
        'CreateDate': 1587081768.334,
        'DnsConfig': {
            'DnsRecords': [
                {
                    'TTL': 60,
                    'Type': 'A',
                },
            ],
            'NamespaceId': 'ns-xxxxxxxxxx',
            'RoutingPolicy': 'MULTIVALUE',
        },
        'Id': 'srv-xxxxxxxxxx',
        'Name': 'service-name',
        'NamespaceId': 'ns-xxxxxxxxxx',
    },
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Próximas etapas

Depois de criar um serviço, você pode registrar os recursos do aplicativo como instâncias de serviço que contêm informações sobre como o aplicativo pode localizar o recurso. Para obter mais informações sobre o registro de instâncias AWS Cloud Map de serviço, consulteRegistrando um recurso como instância de AWS Cloud Map serviço.

Você também pode especificar metadados personalizados, como pesos de endpoints, tempos limite de API e políticas de repetição, como atributos de serviço após criar um serviço. Para obter mais informações, consulte ServiceAttributes e `UpdateServiceAttributes` na Referência da API do AWS Cloud Map .

Atualizando um AWS Cloud Map serviço

Dependendo da configuração de um serviço, você pode atualizar suas tags, o limite de falha na verificação de integridade do Route 53 e o tempo de vida (TTL) para resolvedores de DNS. Para atualizar uma instância de serviço, execute o procedimento a seguir.

> **Note**
>
> Você não pode atualizar as configurações dos serviços associados aos namespaces HTTP.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Na página Namespaces, escolha o namespace no qual o serviço é criado.
4. Na *namespace-name* página Namespace:, selecione o serviço que você deseja editar e escolha Exibir detalhes.
5. Na *service-name* página Serviço:, escolha Editar.

   > **Note**
   >
   > Você não pode usar o fluxo de trabalho do botão Editar para editar valores de serviços que permitem somente chamadas de API para descoberta de instâncias. No entanto, você pode adicionar ou remover tags na *service-name* página Serviço:.

6. Na página Editar serviço, em Descrição do serviço, você pode atualizar qualquer descrição definida anteriormente para o serviço ou adicionar uma nova descrição. Você também pode adicionar tags e atualizar o TTL para resolvedores de DNS.
7. Em Configuração de DNS, para TTL, você pode especificar um período de tempo atualizado, em segundos, que determina por quanto tempo os resolvedores de DNS armazenam as informações desse registro antes que os resolvedores encaminhem outra consulta de DNS para o Amazon Route 53 para obter as configurações atualizadas.
8. Se você configurou as verificações de saúde do Route 53, para Limite de falha, você pode especificar um novo número entre 1 e 10 que define o número de verificações de saúde

consecutivas do Route 53 que uma instância de serviço deve passar ou falhar para que seu status de saúde mude.

9. Escolha Serviço de atualização.

AWS CLI

- Atualize um serviço com o `update-service` comando (substitua o *red* valor pelo seu próprio).

```
aws servicediscovery update-service \
    --id  srv-xxxxxxxxxxx \
    --service "Description=new
 description,DnsConfig={DnsRecords=[{Type=A,TTL=60}]}"
```

Saída:

```
{
    "OperationId": "l3pfx7f4ynndrbj3cfq5fm2qy2z37bms-5m6iaoty"
}
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Atualize um serviço com `update_service()` (substitua o *red* valor pelo seu).

```
response = client.update_service(
    Id='srv-xxxxxxxxxxx',
    Service={
        'DnsConfig': {
            'DnsRecords': [
                {
                    'TTL': 300,
                    'Type': 'A',
```

```
                },
            ],
        },
        'Description': "new description",
    }
)
```

Exemplo de objeto de resposta

```
{
    "OperationId": "l3pfx7f4ynndrbj3cfq5fm2qy2z37bms-5m6iaoty"
}
```

Listando AWS Cloud Map serviços em um namespace

Para visualizar uma lista dos serviços que você criou em um namespace, execute o procedimento a seguir.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Escolha o nome de domínio do namespace que contém os serviços que você deseja listar. Você pode ver uma lista de todos os serviços em Serviços e inserir o nome ou ID do serviço no campo de pesquisa para encontrar um serviço específico. Você pode identificar quem criou Conta da AWS o serviço usando o campo Criado por e a conta proprietária do serviço usando o campo Proprietário do recurso.

> **Note**
>
> Se o namespace for um namespace compartilhado, o Conta da AWS ID em Proprietário do recurso é a conta que criou e compartilhou o namespace. O ID da conta em Criado por pode ser diferente do ID em Proprietário do recurso se um consumidor de namespace tiver criado o serviço. A conta IDs pode não ser igual ao ID da sua conta. Para obter mais informações sobre namespaces compartilhados, consulte. AWS Cloud Map Namespaces compartilhados

AWS CLI

- Liste os serviços com o comando `list-services`. O comando a seguir lista todos os serviços em um namespace usando o ID do namespace como filtro. Substitua o valor *red* pelos seus próprios valores.

```
aws servicediscovery list-services --filters
 Name=NAMESPACE_ID,Values=ns-1234567890abcdef,Condition=EQ
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Liste os serviços com `list_services()`.

```
response = client.list_services()
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'Services': [
        {
            'Arn': 'arn:aws:servicediscovery:us-west-2:123456789012:service/srv-
xxxxxxxxxxxxxxx',
            'CreateDate': 1587081768.334,
            'DnsConfig': {
                'DnsRecords': [
                    {
                        'TTL': 60,
                        'Type': 'A',
                    },
                ],
                'RoutingPolicy': 'MULTIVALUE',
```

```
            },
            'Id': 'srv-xxxxxxxxxxxxxxx',
            'Name': 'myservice',
        },
    ],
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Excluindo um serviço AWS Cloud Map

Para poder excluir um serviço, você deve cancelar o registro de todas as instâncias de serviço que foram registradas usando o serviço. Para obter mais informações, consulte Cancelando o registro de uma instância de serviço AWS Cloud Map.

Depois de cancelar o registro de todas as instâncias registradas usando o serviço, execute o procedimento a seguir para excluir o serviço.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Escolha a opção do namespace que contém o serviço que você deseja excluir.
4. Na *namespace-name* página Namespace:, escolha a opção para o serviço que você deseja excluir.
5. Escolha Excluir.
6. Confirme se você deseja excluir o serviço.

AWS CLI

- Exclua um serviço com o `delete-service` comando (substitua o *red* valor pelo seu).

```
aws servicediscovery delete-service --id srv-xxxxxx
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Exclua um serviço com `delete_service()` (substitua o *red* valor pelo seu).

```
response = client.delete_service(
    Id='srv-xxxxxx',
)
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'ResponseMetadata': {
        '...': '...',
    },
}
```

AWS Cloud Map instâncias de serviço

Uma instância de serviço contém informações sobre como localizar um recurso, como um servidor web, para um aplicativo. Depois de registrar as instâncias, você as localiza usando consultas de DNS ou a ação da AWS Cloud Map DiscoverInstancesAPI. Os recursos que você pode registrar incluem, mas não estão limitados aos seguintes:

- Instâncias do Amazon EC2
- Tabelas do Amazon DynamoDB
- Buckets do Amazon S3
- Filas do Amazon Simple Queue Service (Amazon SQS)
- APIs implantado em cima do Amazon API Gateway

Você pode especificar valores de atributos para instâncias de serviços, e os clientes podem usar esses atributos para filtrar os recursos que AWS Cloud Map retornam. Por exemplo, um aplicativo pode solicitar recursos em um determinado estágio de implantação, como BETA ou PROD. Você também pode usar atributos para controle de versão.

Os procedimentos a seguir descrevem como você pode registrar recursos em seu aplicativo como instâncias de serviço, visualizar uma lista de instâncias registradas em um serviço, editar determinados parâmetros de instância e cancelar o registro de uma instância.

Tópicos

Registrando um recurso como instância de AWS Cloud Map serviço

Você pode registrar os recursos do seu aplicativo como instâncias em um AWS Cloud Map serviço. Por exemplo, suponha que você tenha criado um serviço chamado `users` para todos os recursos

do aplicativo que gerenciam dados do usuário. Em seguida, você pode registrar uma tabela do DynamoDB usada para armazenar dados do usuário como uma instância nesse serviço.

> **Note**
>
> Os seguintes recursos não estão disponíveis no AWS Cloud Map console:
>
> - Ao registrar uma instância de serviço usando o console, você não pode criar um registro de alias que roteia o tráfego para um balanceador de carga Elastic Load Balancing (ELB). Ao registrar uma instância, você deve incluir o atributo `AWS_ALIAS_DNS_NAME`. Para obter mais informações, consulte RegisterInstance na Referência de APIs do AWS Cloud Map .
> - Se você registrar uma instância usando um serviço que inclua uma verificação de integridade personalizada, não será possível especificar o status inicial da verificação de integridade personalizada. Por padrão, o status inicial de verificações de integridade personalizadas é Healthy (Íntegra). Para que o status de integridade inicial seja Unhealthy (Não íntegra), registre a instância de forma programática e inclua o atributo `AWS_INIT_HEALTH_STATUS`. Para obter mais informações, consulte RegisterInstance na Referência de APIs do AWS Cloud Map .

Para registrar uma instância em um serviço, siga estas etapas.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Na página Namespaces, escolha o namespace que contém o serviço que você deseja usar como modelo para registrar uma instância do serviço.
4. Na *`namespace-name`* página Namespace:, escolha o serviço que você deseja usar.
5. Na *`service-name`* página Serviço:, escolha Registrar instância do serviço.
6. Na página Registrar instância do serviço, escolha um tipo de instância. Dependendo da configuração de descoberta da instância de namespace, você pode optar por especificar um endereço IP, um ID de instância do Amazon EC2 ou outras informações de identificação para um recurso que não tem um endereço IP.

> **Note**
>
> Você pode escolher a instância EC2 somente em namespaces HTTP.

7. Para ID da instância de serviço, forneça um identificador associado à instância de serviço.

> **Note**
>
> Se você quiser atualizar uma instância existente, forneça o identificador associado à instância que você deseja atualizar. Em seguida, use as próximas etapas para atualizar os valores e registrar novamente a instância.

8. Com base no tipo de instância de sua escolha, execute as etapas a seguir.

> **Important**
>
> Você não pode usar o `AWS_` prefixo (sem distinção entre maiúsculas e minúsculas) em uma chave ao especificar um atributo personalizado.

Tipo de instância	Etapas	
IP address (endereço de IP)	a. Em Atributos padrão, para IPv4endereço, forneça um IPv4 endereço, se houver, em que seu aplicativ o possa acessar o recurso associado a essa instância de serviço. b. Para IPv6 endereço, forneça um endereço IPv6 IP, se houver, em que seus aplicativ os possam acessar o	

Tipo de instância	Etapas	
	recurso associado a essa instância de serviço. c. Para Port, especifiq ue qualquer porta que seu aplicativo deve incluir para acessar o recurso associado a essa instância de serviço. A porta é necessária quando o serviço inclui um registro SRV ou uma verificação de saúde do Amazon Route 53. d. (Opcional) Em Atributos personalizados, especifiq ue os pares de valores-chave que você deseja associar ao recurso.	
Instância do EC2	a. Para o ID da instância do EC2, selecione o ID da instância do Amazon EC2 que você deseja registrar como AWS Cloud Map uma instância de serviço. b. (Opcional) Em Atributos personalizados, especifiq ue os pares de valores-chave que você deseja associar ao recurso.	

Tipo de instância	Etapas	
Identifying information for another resource (Identificar informações de outro recurso)	a. Em Atributos padrão, se a configuração do serviço incluir um registro DNS CNAME, você verá um campo CNAME. Para CNAME, especifique o nome de domínio que você deseja que o Route 53 retorne em resposta às consultas de DNS (por exemplo,). `example.com` b. Em Atributos personalizados, especifique qualquer informação de identificação de um recurso que não seja um endereço IP ou uma ID de instância do Amazon EC2 como um par de valores-chave. Por exemplo, você pode registrar uma função Lambda especificando uma chave chamada `function` e fornecendo o nome da função Lambda como um valor. Você também pode especificar uma chave chamada `name` e fornecer um nome que você pode usar para a	

Tipo de instância	Etapas	
	descoberta programática de instâncias.	

9. Escolha Registrar instância de serviço.

AWS CLI

- Quando você envia uma solicitação de `RegisterInstance`:
  - Para cada registro de DNS definido no serviço especificado por `ServiceId`, um registro é criado ou atualizado na zona hospedada associada ao namespace correspondente.
  - Caso o serviço inclua `HealthCheckConfig`, uma verificação de integridade será criada com base nas configurações da verificação de integridade.
  - Todas as verificações de integridade estão associadas a um dos registros novos ou atualizados.

  Registre uma instância de serviço com o `register-instance` comando (substitua *red* os valores pelos seus).

```
aws servicediscovery register-instance \
    --service-id srv-xxxxxxxxx \
    --instance-id myservice-xx \
    --attributes=AWS_INSTANCE_IPV4=172.2.1.3,AWS_INSTANCE_PORT=808
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Quando você envia uma solicitação de `RegisterInstance`:

- Para cada registro de DNS definido no serviço especificado por `ServiceId`, um registro é criado ou atualizado na zona hospedada associada ao namespace correspondente.
- Caso o serviço inclua `HealthCheckConfig`, uma verificação de integridade será criada com base nas configurações da verificação de integridade.
- Todas as verificações de integridade estão associadas a um dos registros novos ou atualizados.

Registre uma instância de serviço com `register_instance()` (substitua *red* os valores pelos seus).

```
response = client.register_instance(
    Attributes={
        'AWS_INSTANCE_IPV4': '172.2.1.3',
        'AWS_INSTANCE_PORT': '808',
    },
    InstanceId='myservice-xx',
    ServiceId='srv-xxxxxxxxx',
)
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'OperationId': '4yejorelbukcjzpnr6tlmrghsjwpngf4-k95yg2u7',
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Listando instâncias AWS Cloud Map de serviço

Para visualizar uma lista de instâncias de serviço que você registrou usando um serviço, execute o procedimento a seguir.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Escolha o nome do namespace que contém o serviço do qual você deseja listar instâncias de serviço.
4. Escolha o nome do serviço usado para criar as instâncias de serviço. Você verá uma lista de instâncias em Instâncias de serviço. Você pode inserir o ID da instância no campo de pesquisa para listar uma instância específica. O campo Criado por mostra o ID do Conta da AWS que registrou a instância.

> **Note**
>
> Se o namespace no qual a instância está registrada for um namespace compartilhado, o Conta da AWS ID em Criado por pode não ser o mesmo que o ID da sua conta. Para obter mais informações sobre namespaces compartilhados, consulte. AWS Cloud Map Namespaces compartilhados

AWS CLI

- Liste as instâncias do serviço com o `list-instances` comando (substitua o *red* valor pelo seu próprio).

```
aws servicediscovery list-instances --service-id srv-xxxxxxxxx
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Liste as instâncias de serviço com `list_instances()` (substitua o *red* valor pelo seu).

```
response = client.list_instances(
    ServiceId='srv-xxxxxxxxx',
)
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'Instances': [
        {
            'Attributes': {
                'AWS_INSTANCE_IPV4': '172.2.1.3',
                'AWS_INSTANCE_PORT': '808',
            },
            'Id': 'i-xxxxxxxxxxxxxxxx',
        },
    ],
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Atualização de uma instância AWS Cloud Map de serviço

É possível atualizar instâncias de serviço de duas maneiras, dependendo dos valores que deseja atualizar:

- Atualizar qualquer valor: se você quiser atualizar qualquer um dos valores que você especificou para uma instância de serviço ao registrá-la, incluindo atributos personalizados, você precisa registrar novamente a instância de serviço e especificar novamente todos os valores. Siga as etapas abaixoRegistrando um recurso como instância de AWS Cloud Map serviço, especificando o ID da instância de serviço existente para o ID da instância de serviço.

  Como alternativa, você pode usar a `RegisterInstance`API. Você pode especificar a ID da instância e do serviço existentes usando os `ServiceId` parâmetros `InstanceId` e e reespecificar outros valores.

- Atualizar somente atributos personalizados: se você quiser atualizar somente os atributos personalizados de uma instância de serviço, não será necessário registrar novamente a instância. É possível atualizar somente esses valores. Consulte Atualização dos atributos personalizados de uma instância de serviço.

Atualização dos atributos personalizados de uma instância de serviço

Como atualizar somente atributos personalizados de uma instância de serviço

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.
2. No painel de navegação, escolha Namespaces.
3. Na página Namespaces, escolha o namespace que contém o serviço que você usou originalmente para registrar a instância de serviço.
4. Na *namespace-name* página Namespace:, escolha o serviço que você usou para registrar a instância do serviço.
5. Na *service-name* página Serviço:, escolha o nome da instância de serviço que você deseja atualizar.
6. Na seção Atributos personalizados escolha Editar.
7. Na *instance-name* página Editar instância do serviço:, adicione, remova ou atualize atributos personalizados. É possível atualizar chaves e valores de atributos existentes.
8. Escolha Atualizar instância do serviço.

Cancelando o registro de uma instância de serviço AWS Cloud Map

Para poder excluir um serviço, você deve cancelar o registro de todas as instâncias de serviço que foram registradas usando o serviço.

Para cancelar o registro de uma instância de serviço, execute o procedimento a seguir.

Console de gerenciamento da AWS

1. Faça login no Console de gerenciamento da AWS e abra o AWS Cloud Map console em https://console.aws.amazon.com/cloudmap/.

2. No painel de navegação, escolha Namespaces.
3. Escolha a opção do namespace que contém a instância de serviço da qual você deseja cancelar o registro.
4. Na *namespace-name* página Namespace:, escolha o serviço que você usou para registrar a instância do serviço.
5. Na *service-name* página Serviço:, escolha a instância de serviço que você deseja cancelar o registro.
6. Escolha Cancelar registro.
7. Confirme se você deseja cancelar o registro da instância de serviço.

AWS CLI

- Cancele o registro de uma instância de serviço com o `deregister-instance` comando (substitua os *red* valores pelos seus). Esse comando exclui os registros DNS do Amazon Route 53 e todas as verificações de saúde AWS Cloud Map criadas para a instância especificada.

```
aws servicediscovery deregister-instance \
    --service-id srv-xxxxxxxxx \
    --instance-id myservice-53
```

AWS SDK for Python (Boto3)

1. Caso ainda não tenha instalado `Boto3`, é possível encontrar instruções para instalação, configuração e uso do `Boto3` aqui.
2. Importe `Boto3` e use `servicediscovery` como seu serviço.

```
import boto3
client = boto3.client('servicediscovery')
```

3. Cancele o registro de uma instância de serviço com `deregister-instance()` (substitua *red* os valores pelos seus). Esse comando exclui os registros DNS do Amazon Route 53 e todas as verificações de saúde AWS Cloud Map criadas para a instância especificada.

```
response = client.deregister_instance(
    InstanceId='myservice-53',
```

```
    ServiceId='srv-xxxxxxxxx',
)
# If you want to see the response
print(response)
```

Exemplo de objeto de resposta

```
{
    'OperationId': '4yejorelbukcjzpnr6tlmrghsjwpngf4-k98rnaiq',
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Segurança em AWS Cloud Map

A segurança na nuvem AWS é a maior prioridade. Como AWS cliente, você se beneficia de uma arquitetura de data center e rede criada para atender aos requisitos das organizações mais sensíveis à segurança.

A segurança é uma responsabilidade compartilhada entre você AWS e você. O modelo de responsabilidade compartilhada descreve isto como segurança da nuvem e segurança na nuvem.

- Segurança da nuvem — AWS é responsável por proteger a infraestrutura que executa AWS os serviços na AWS nuvem. AWS também fornece serviços que você pode usar com segurança. Auditores de terceiros testam e verificam regularmente a eficácia da nossa segurança como parte dos programas de conformidade da AWS. Para saber mais sobre os programas de conformidade aplicáveis AWS Cloud Map, consulte AWS Serviços no escopo por programa de conformidade.
- Segurança na nuvem — Sua responsabilidade é determinada pelo AWS serviço que você usa. Você também é responsável por outros fatores, incluindo a confidencialidade de seus dados, os requisitos da empresa e as leis e regulamentos aplicáveis.

A documentação a seguir ajuda você a entender como aplicar o modelo de responsabilidade compartilhada ao usar AWS Cloud Map. Os tópicos a seguir mostram como configurar para atender AWS Cloud Map aos seus objetivos de segurança e conformidade. Você também aprenderá a usar outros AWS serviços que ajudam a monitorar e proteger seus AWS Cloud Map recursos.

Tópicos

Identity and Access Management para AWS Cloud Map

AWS Identity and Access Management (IAM) é uma ferramenta AWS service (Serviço da AWS) que ajuda o administrador a controlar com segurança o acesso aos AWS recursos. Os administradores do IAM controlam quem pode ser autenticado (conectado) e autorizado (tem permissões) a usar

AWS Cloud Map os recursos. O IAM é um AWS service (Serviço da AWS) que você pode usar sem custo adicional.

Tópicos

Público

A forma como você usa AWS Identity and Access Management (IAM) difere com base na sua função:

- Usuário do serviço: solicite permissões ao seu administrador se você não conseguir acessar os atributos (consulte Solução de problemas AWS Cloud Map de identidade e acesso).
- Administrador do serviço: determine o acesso do usuário e envie solicitações de permissão (consulte Como AWS Cloud Map funciona com o IAM)
- Administrador do IAM: escreva políticas para gerenciar o acesso (consulte Exemplos de políticas baseadas em identidade para AWS Cloud Map)

Autenticação com identidades

A autenticação é a forma como você faz login AWS usando suas credenciais de identidade. Você deve estar autenticado como usuário do IAM ou assumindo uma função do IAM. Usuário raiz da conta da AWS

Você pode fazer login como uma identidade federada usando credenciais de uma fonte de identidade como Centro de Identidade do AWS IAM (IAM Identity Center), autenticação de login único ou credenciais. Google/Facebook Para ter mais informações sobre como fazer login, consulte Como fazer login em sua Conta da AWS no Guia do usuário do Início de Sessão da AWS .

Para acesso programático, AWS fornece um SDK e uma CLI para assinar solicitações criptograficamente. Para ter mais informações, consulte AWS Signature Version 4 para solicitações de API no Guia do usuário do IAM.

Conta da AWS usuário root

Ao criar um Conta da AWS, você começa com uma identidade de login chamada usuário Conta da AWS raiz que tem acesso completo a todos Serviços da AWS os recursos. É altamente recomendável não usar o usuário-raiz em tarefas diárias. Consulte as tarefas que exigem credenciais de usuário-raiz em Tarefas que exigem credenciais de usuário-raiz no Guia do usuário do IAM.

Identidade federada

Como prática recomendada, exija que os usuários humanos usem a federação com um provedor de identidade para acessar Serviços da AWS usando credenciais temporárias.

Uma identidade federada é um usuário do seu diretório corporativo, provedor de identidade da web ou Directory Service que acessa Serviços da AWS usando credenciais de uma fonte de identidade. As identidades federadas assumem funções que oferecem credenciais temporárias.

Para o gerenciamento de acesso centralizado, recomendamos Centro de Identidade do AWS IAM. Para saber mais, consulte O que é o IAM Identity Center? no Guia do usuário do Centro de Identidade do AWS IAM .

Usuários e grupos do IAM

Um usuário do IAM é uma identidade com permissões específicas para uma única pessoa ou aplicação. É recomendável usar credenciais temporárias, em vez de usuários do IAM com credenciais de longo prazo. Para obter mais informações, consulte Exigir que usuários humanos usem a federação com um provedor de identidade para acessar AWS usando credenciais temporárias no Guia do usuário do IAM.

Um grupo do IAM especifica um conjunto de usuários do IAM e facilita o gerenciamento de permissões para grandes conjuntos de usuários. Para ter mais informações, consulte Casos de uso de usuários do IAM no Guia do usuário do IAM.

Perfis do IAM

Uma perfil do IAM é uma identidade com permissões específicas que oferece credenciais temporárias. Você pode assumir uma função mudando de um usuário para uma função do IAM

(console) ou chamando uma operação de AWS API AWS CLI ou. Para saber mais, consulte Métodos para assumir um perfil no Manual do usuário do IAM.

Os perfis do IAM são úteis para acesso de usuário federado, permissões de usuário do IAM temporárias, acesso entre contas, acesso entre serviços e aplicações em execução no Amazon EC2. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Gerenciar o acesso usando políticas

Você controla o acesso AWS criando políticas e anexando-as a AWS identidades ou recursos. Uma política define permissões quando associada a uma identidade ou recurso. AWS avalia essas políticas quando um diretor faz uma solicitação. A maioria das políticas é armazenada AWS como documentos JSON. Para ter mais informações sobre documentos de política JSON, consulte Visão geral das políticas JSON no Guia do usuário do IAM.

Por meio de políticas, os administradores especificam quem tem acesso a que, definindo qual entidade principal pode realizar ações em quais recursos e sob quais condições.

Por padrão, usuários e perfis não têm permissões. Um administrador do IAM cria políticas do IAM e as adiciona aos perfis, os quais os usuários podem então assumir. As políticas do IAM definem permissões, independentemente do método usado para realizar a operação.

Políticas baseadas em identidade

As políticas baseadas em identidade são documentos de políticas de permissão JSON que você anexa a uma identidade (usuário, grupo ou perfil). Essas políticas controlam quais ações as identidades podem realizar, em quais recursos e sob quais condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

As políticas baseadas em identidade podem ser políticas em linha (incorporadas diretamente em uma única identidade) ou políticas gerenciadas (políticas autônomas anexadas a várias identidades). Para saber como escolher entre uma política gerenciada e políticas em linha, consulte Escolher entre políticas gerenciadas e políticas em linha no Guia do usuário do IAM.

Políticas baseadas em recursos

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. Entre os exemplos estão políticas de confiança de perfil do IAM e políticas de bucket do Amazon S3.

Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. É necessário especificar uma entidade principal em uma política baseada em recursos.

Políticas baseadas em recursos são políticas em linha localizadas nesse serviço. Você não pode usar políticas AWS gerenciadas do IAM em uma política baseada em recursos.

Outros tipos de política

AWS oferece suporte a tipos de políticas adicionais que podem definir o máximo de permissões concedidas por tipos de políticas mais comuns:

- Limites de permissões: definem o número máximo de permissões que uma política baseada em identidade pode conceder a uma entidade do IAM. Para saber mais sobre limites de permissões, consulte Limites de permissões para identidades do IAM no Guia do usuário do IAM.
- Políticas de controle de serviço (SCPs) — Especifique as permissões máximas para uma organização ou unidade organizacional em AWS Organizations. Para saber mais, consulte Políticas de controle de serviço no Guia do usuário do AWS Organizations .
- Políticas de controle de recursos (RCPs) — Defina o máximo de permissões disponíveis para recursos em suas contas. Para obter mais informações, consulte Políticas de controle de recursos (RCPs) no Guia AWS Organizations do usuário.
- Políticas de sessão: políticas avançadas transmitidas como um parâmetro durante a criação de uma sessão temporária para um perfil ou um usuário federado. Para saber mais, consulte Políticas de sessão no Guia do usuário do IAM.

Vários tipos de política

Quando vários tipos de política são aplicáveis a uma solicitação, é mais complicado compreender as permissões resultantes. Para saber como AWS determinar se uma solicitação deve ser permitida quando vários tipos de políticas estão envolvidos, consulte Lógica de avaliação de políticas no Guia do usuário do IAM.

Como AWS Cloud Map funciona com o IAM

Antes de usar o IAM para gerenciar o acesso AWS Cloud Map, saiba com quais recursos do IAM estão disponíveis para uso AWS Cloud Map.

Recurso do IAM	AWS Cloud Map apoio
Políticas baseadas em identidade	Sim
Políticas baseadas em recurso	Não
Ações de políticas	Sim
Recursos de políticas	Sim
Chaves de condição de política (específicas do serviço)	Sim
ACLs	Não
ABAC (tags em políticas)	Sim
Credenciais temporárias	Sim
Sessões de acesso direto (FAS)	Sim
Perfis de serviço	Não
Perfis vinculados ao serviço	Sim

Para ter uma visão de alto nível de como AWS Cloud Map e outros AWS serviços funcionam com a maioria dos recursos do IAM, consulte AWS os serviços que funcionam com o IAM no Guia do usuário do IAM.

Políticas baseadas em identidade para AWS Cloud Map

Compatível com políticas baseadas em identidade: sim

As políticas baseadas em identidade são documentos de políticas de permissões JSON que podem ser anexados a uma identidade, como usuário do IAM, grupo de usuários ou perfil. Essas políticas controlam quais ações os usuários e perfis podem realizar, em quais recursos e em que condições. Para saber como criar uma política baseada em identidade, consulte Definir permissões personalizadas do IAM com as políticas gerenciadas pelo cliente no Guia do Usuário do IAM.

Com as políticas baseadas em identidade do IAM, é possível especificar ações e recursos permitidos ou negados, assim como as condições sob as quais as ações são permitidas ou negadas. Para saber mais sobre todos os elementos que podem ser usados em uma política JSON, consulte Referência de elemento de política JSON do IAM no Guia do usuário do IAM.

Exemplos de políticas baseadas em identidade para AWS Cloud Map

Para ver exemplos de políticas AWS Cloud Map baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para AWS Cloud Map

Políticas baseadas em recursos dentro AWS Cloud Map

Compatibilidade com políticas baseadas em recursos: não

Políticas baseadas em recursos são documentos de políticas JSON que você anexa a um recurso. São exemplos de políticas baseadas em recursos as políticas de confiança de perfil do IAM e as políticas de bucket do Amazon S3. Em serviços compatíveis com políticas baseadas em recursos, os administradores de serviço podem usá-las para controlar o acesso a um recurso específico. Para o atributo ao qual a política está anexada, a política define quais ações uma entidade principal especificado pode executar nesse atributo e em que condições. É necessário especificar uma entidade principal em uma política baseada em recursos. Os diretores podem incluir contas, usuários, funções, usuários federados ou. Serviços da AWS

Para permitir o acesso entre contas, é possível especificar uma conta inteira ou as entidades do IAM em outra conta como a entidade principal em uma política baseada em recursos. Consulte mais informações em Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

> **Note**
>
> Você pode usar AWS Resource Access Manager (AWS RAM) para compartilhar com segurança um AWS Cloud Map namespace. Uma política baseada em recursos é aplicada ao seu namespace pelo serviço. AWS RAM Para obter mais informações, consulte AWS Cloud Map Namespaces compartilhados.

Ações políticas para AWS Cloud Map

Compatível com ações de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Action` de uma política JSON descreve as ações que podem ser usadas para permitir ou negar acesso em uma política. Incluem ações em uma política para conceder permissões para executar a operação associada.

Para ver uma lista de AWS Cloud Map ações, consulte Ações definidas por AWS Cloud Map na Referência de Autorização de Serviço.

As ações de política AWS Cloud Map usam o seguinte prefixo antes da ação:

```
servicediscovery
```

Para especificar várias ações em uma única declaração, separe-as com vírgulas.

```
"Action": [
      "servicediscovery:action1",
      "servicediscovery:action2"
         ]
```

Para ver exemplos de políticas AWS Cloud Map baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para AWS Cloud Map

Recursos políticos para AWS Cloud Map

Compatível com recursos de políticas: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento de política JSON `Resource` especifica o objeto ou os objetos aos quais a ação se aplica. Como prática recomendada, especifique um recurso usando seu nome do recurso da Amazon (ARN). Para ações que não oferecem compatibilidade com permissões em nível de recurso, use um curinga (*) para indicar que a instrução se aplica a todos os recursos.

```
"Resource": "*"
```

Para ver uma lista dos tipos de AWS Cloud Map recursos e seus ARNs, consulte Recursos definidos por AWS Cloud Map na Referência de Autorização de Serviço. Para saber com quais ações é possível especificar o ARN de cada atributo, consulte Ações definidas pelo AWS Cloud Map.

Para ver exemplos de políticas AWS Cloud Map baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para AWS Cloud Map

Chaves de condição de política para AWS Cloud Map

Compatível com chaves de condição de política específicas de serviço: sim

Os administradores podem usar políticas AWS JSON para especificar quem tem acesso ao quê. Ou seja, qual entidade principal pode executar ações em quais recursos e em que condições.

O elemento `Condition` especifica quando as instruções são executadas com base em critérios definidos. É possível criar expressões condicionais que usem agentes de condição, como “igual a” ou “menor que”, para fazer a condição da política corresponder aos valores na solicitação. Para ver todas as chaves de condição AWS globais, consulte as chaves de contexto de condição AWS global no Guia do usuário do IAM.

Para ver uma lista de chaves de AWS Cloud Map condição, consulte Chaves de condição AWS Cloud Map na Referência de autorização de serviço. Para saber com quais ações e recursos você pode usar uma chave de condição, consulte Ações definidas por AWS Cloud Map.

AWS Cloud Map oferece suporte às seguintes chaves de condição específicas do serviço que você pode usar para fornecer uma filtragem refinada para suas políticas do IAM.

`servicediscovery:NamespaceArn`

Um filtro que permite obter objetos especificando o nome de recurso da Amazon (ARN) do namespace relacionado.

`servicediscovery:NamespaceName`

Um filtro que permite obter objetos especificando o nome do namespace relacionado.

`servicediscovery:ServiceArn`

Um filtro que permite obter objetos especificando o nome de recurso da Amazon (ARN) do serviço relacionado.

`servicediscovery:ServiceName`

Um filtro que permite obter objetos especificando o nome do serviço relacionado.

`servicediscovery:ServiceCreatedByAccount`

Um filtro que permite obter objetos especificando a ID da pessoa Conta da AWS que criou o serviço.

Para ver exemplos de políticas AWS Cloud Map baseadas em identidade, consulte. Exemplos de políticas baseadas em identidade para AWS Cloud Map

ACLs in AWS Cloud Map

Suportes ACLs: Não

As listas de controle de acesso (ACLs) controlam quais diretores (membros da conta, usuários ou funções) têm permissões para acessar um recurso. ACLs são semelhantes às políticas baseadas em recursos, embora não usem o formato de documento de política JSON.

ABAC com AWS Cloud Map

Compatível com ABAC (tags em políticas): sim

O controle de acesso por atributo (ABAC) é uma estratégia de autorização que define permissões com base em atributos chamados de tags. Você pode anexar tags a entidades e AWS recursos do IAM e, em seguida, criar políticas ABAC para permitir operações quando a tag do diretor corresponder à tag no recurso.

Para controlar o acesso baseado em tags, forneça informações sobre as tags no elemento de condição de uma política usando as `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`* ou chaves de condição `aws:TagKeys`.

Se um serviço for compatível com as três chaves de condição para cada tipo de recurso, o valor será Sim para o serviço. Se um serviço for compatível com as três chaves de condição somente para alguns tipos de recursos, o valor será Parcial

Para saber mais sobre o ABAC, consulte Definir permissões com autorização do ABAC no Guia do usuário do IAM. Para visualizar um tutorial com etapas para configurar o ABAC, consulte Usar controle de acesso por atributo (ABAC) no Guia do usuário do IAM.

Usando credenciais temporárias com AWS Cloud Map

Compatível com credenciais temporárias: sim

As credenciais temporárias fornecem acesso de curto prazo aos AWS recursos e são criadas automaticamente quando você usa a federação ou troca de funções. AWS recomenda que você gere credenciais temporárias dinamicamente em vez de usar chaves de acesso de longo prazo. Para ter mais informações, consulte Credenciais de segurança temporárias no IAM e Serviços da Serviços da AWS que funcionam com o IAM no Guia do usuário do IAM.

Sessões de acesso direto para AWS Cloud Map

Compatibilidade com o recurso de encaminhamento de sessões de acesso (FAS): sim

As sessões de acesso direto (FAS) usam as permissões do principal chamando um AWS service (Serviço da AWS), combinadas com a solicitação AWS service (Serviço da AWS) de fazer solicitações aos serviços posteriores. Para obter detalhes da política ao fazer solicitações de FAS, consulte Sessões de acesso direto.

Funções de serviço para AWS Cloud Map

Compatível com perfis de serviço: não

O perfil de serviço é um perfil do IAM que um serviço assume para executar ações em seu nome. Um administrador do IAM pode criar, modificar e excluir um perfil de serviço do IAM. Para saber mais, consulte Criar um perfil para delegar permissões a um AWS service (Serviço da AWS) no Guia do Usuário do IAM.

> **Warning**
>
> Alterar as permissões de um perfil de serviço pode prejudicar a funcionalidade do AWS Cloud Map . Edite as funções de serviço somente quando AWS Cloud Map fornecer orientação para fazer isso.

Funções vinculadas a serviços para AWS Cloud Map

Compatibilidade com perfis vinculados a serviços: sim

Uma função vinculada ao serviço é um tipo de função de serviço vinculada a um. AWS service (Serviço da AWS) O serviço pode assumir o perfil de executar uma ação em seu nome. As funções

vinculadas ao serviço aparecem em você Conta da AWS e são de propriedade do serviço. Um administrador do IAM pode visualizar, mas não editar as permissões para perfis vinculados ao serviço.

Para obter detalhes sobre como criar ou gerenciar perfis vinculados a serviços, consulte Serviços da AWS que funcionam com o IAM. Encontre um serviço na tabela que inclua um `Yes` na coluna Perfil vinculado ao serviço. Escolha o link Sim para visualizar a documentação do perfil vinculado a serviço desse serviço.

Exemplos de políticas baseadas em identidade para AWS Cloud Map

Por padrão, usuários e perfis não têm permissão para criar ou modificar recursos do AWS Cloud Map . Para conceder permissão aos usuários para executar ações nos recursos que eles precisam, um administrador do IAM pode criar políticas do IAM.

Para aprender a criar uma política baseada em identidade do IAM ao usar esses documentos de política em JSON de exemplo, consulte Criar políticas do IAM (console) no Guia do usuário do IAM.

Para obter detalhes sobre ações e tipos de recursos definidos por AWS Cloud Map, incluindo o formato do ARNs para cada um dos tipos de recursos, consulte Ações, recursos e chaves de condição AWS Cloud Map na Referência de Autorização de Serviço.

Tópicos

- Práticas recomendadas de política
- Usando o AWS Cloud Map console
- AWS Cloud Map exemplo de acesso ao console
- Permita que AWS Cloud Map os usuários visualizem suas próprias permissões
- Permitir acesso de leitura a todos os AWS Cloud Map recursos
- AWS Cloud Map exemplo de instância de serviço
- Crie um exemplo AWS Cloud Map de serviço
- Exemplo de criação de AWS Cloud Map namespaces

Práticas recomendadas de política

As políticas baseadas em identidade determinam se alguém pode criar, acessar ou excluir AWS Cloud Map recursos em sua conta. Essas ações podem incorrer em custos para sua Conta da AWS. Ao criar ou editar políticas baseadas em identidade, siga estas diretrizes e recomendações:

- Comece com as políticas AWS gerenciadas e avance para as permissões de privilégios mínimos — Para começar a conceder permissões aos seus usuários e cargas de trabalho, use as políticas AWS gerenciadas que concedem permissões para muitos casos de uso comuns. Eles estão disponíveis no seu Conta da AWS. Recomendamos que você reduza ainda mais as permissões definindo políticas gerenciadas pelo AWS cliente que sejam específicas para seus casos de uso. Para saber mais, consulte Políticas gerenciadas pela AWS ou Políticas gerenciadas pela AWS para funções de trabalho no Guia do usuário do IAM.
- Aplique permissões de privilégio mínimo: ao definir permissões com as políticas do IAM, conceda apenas as permissões necessárias para executar uma tarefa. Você faz isso definindo as ações que podem ser executadas em recursos específicos sob condições específicas, também conhecidas como permissões de privilégio mínimo. Para saber mais sobre como usar o IAM para aplicar permissões, consulte Políticas e permissões no IAM no Guia do usuário do IAM.
- Use condições nas políticas do IAM para restringir ainda mais o acesso: é possível adicionar uma condição às políticas para limitar o acesso a ações e recursos. Por exemplo, é possível escrever uma condição de política para especificar que todas as solicitações devem ser enviadas usando SSL. Você também pode usar condições para conceder acesso às ações de serviço se elas forem usadas por meio de uma ação específica AWS service (Serviço da AWS), como CloudFormation. Para saber mais, consulte Elementos da política JSON do IAM: condição no Guia do usuário do IAM.
- Use o IAM Access Analyzer para validar suas políticas do IAM a fim de garantir permissões seguras e funcionais: o IAM Access Analyzer valida as políticas novas e existentes para que elas sigam a linguagem de política do IAM (JSON) e as práticas recomendadas do IAM. O IAM Access Analyzer oferece mais de cem verificações de política e recomendações práticas para ajudar a criar políticas seguras e funcionais. Para saber mais, consulte Validação de políticas do IAM Access Analyzer no Guia do Usuário do IAM.
- Exigir autenticação multifator (MFA) — Se você tiver um cenário que exija usuários do IAM ou um usuário root, ative Conta da AWS a MFA para obter segurança adicional. Para exigir MFA quando as operações de API forem chamadas, adicione condições de MFA às suas políticas. Para saber mais, consulte Configuração de acesso à API protegido por MFA no Guia do Usuário do IAM.

Para saber mais sobre as práticas recomendadas do IAM, consulte Práticas recomendadas de segurança no IAM no Guia do usuário do IAM.

Usando o AWS Cloud Map console

Para acessar o AWS Cloud Map console, você deve ter um conjunto mínimo de permissões. Essas permissões devem permitir que você liste e visualize detalhes sobre os AWS Cloud Map recursos em seu Conta da AWS. Caso crie uma política baseada em identidade mais restritiva que as permissões mínimas necessárias, o console não funcionará como pretendido para entidades (usuários ou perfis) com essa política.

Você não precisa permitir permissões mínimas do console para usuários que estão fazendo chamadas somente para a API AWS CLI ou para a AWS API. Em vez disso, permita o acesso somente a ações que correspondam à operação de API que estiverem tentando executar.

Para garantir que usuários e funções ainda possam usar o AWS Cloud Map console, anexe também a política AWS Cloud Map *ConsoleAccess* ou a política *ReadOnly* AWS gerenciada às entidades. Para saber mais, consulte Adicionar permissões a um usuário no Guia do usuário do IAM.

AWS Cloud Map exemplo de acesso ao console

Para conceder acesso total ao AWS Cloud Map console, você concede as permissões na seguinte política de permissões:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "servicediscovery:*",
            "route53:GetHostedZone",
            "route53:ListHostedZonesByName",
            "route53:CreateHostedZone",
            "route53:DeleteHostedZone",
            "route53:ChangeResourceRecordSets",
            "route53:CreateHealthCheck",
            "route53:GetHealthCheck",
            "route53:DeleteHealthCheck",
            "route53:UpdateHealthCheck",
            "ec2:DescribeInstances",
            "ec2:DescribeVpcs",
```

```
            "ec2:DescribeRegions"
         ],
         "Resource":"*"
      }
   ]
}
```

Veja por que as permissões são necessárias:

`servicediscovery:*`

Permite que você execute todas AWS Cloud Map as ações.

`route53:CreateHostedZone`, **`route53:GetHostedZone`**, **`route53:ListHostedZonesByName`**, **`route53:DeleteHostedZone`**

Permite AWS Cloud Map gerenciar zonas hospedadas quando você cria e exclui namespaces DNS públicos e privados.

`route53:CreateHealthCheck`, **`route53:GetHealthCheck`**, **`route53:DeleteHealthCheck`**, **`route53:UpdateHealthCheck`**

Permite AWS Cloud Map gerenciar verificações de saúde quando você inclui verificações de saúde do Amazon Route 53 ao criar um serviço.

`ec2:DescribeVpcs` e **`ec2:DescribeRegions`**

Vamos AWS Cloud Map gerenciar zonas hospedadas privadas.

Permita que AWS Cloud Map os usuários visualizem suas próprias permissões

Este exemplo mostra como criar uma política que permita que os usuários do IAM visualizem as políticas gerenciadas e em linha anexadas a sua identidade de usuário. Essa política inclui permissões para concluir essa ação no console ou programaticamente usando a API AWS CLI ou AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
```

```
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Permitir acesso de leitura a todos os AWS Cloud Map recursos

A seguinte política de permissões concede ao usuário acesso somente leitura a todos os recursos do AWS Cloud Map :

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "servicediscovery:Get*",
```

```
            "servicediscovery:List*",
            "servicediscovery:DiscoverInstances"
         ],
         "Resource":"*"
      }
   ]
}
```

AWS Cloud Map exemplo de instância de serviço

O exemplo a seguir mostra uma política de permissões que concede ao usuário permissão para registrar, cancelar o registro e descobrir instâncias de serviço. O `Sid`, ou o ID de instrução, é opcional:

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Sid" : "AllowInstancePermissions",
         "Effect": "Allow",
         "Action": [
            "servicediscovery:RegisterInstance",
            "servicediscovery:DeregisterInstance",
            "servicediscovery:DiscoverInstances",
            "servicediscovery:Get*",
            "servicediscovery:List*",
            "route53:GetHostedZone",
            "route53:ListHostedZonesByName",
            "route53:ChangeResourceRecordSets",
            "route53:CreateHealthCheck",
            "route53:GetHealthCheck",
            "route53:DeleteHealthCheck",
            "route53:UpdateHealthCheck",
            "ec2:DescribeInstances"
         ],
         "Resource": "*"
      }
   ]
}
```

A política concede permissões para as ações que são necessárias para registrar e gerenciar instâncias de serviço. A permissão do Route 53 é necessária se você estiver usando namespaces DNS públicos ou privados porque AWS Cloud Map cria, atualiza e exclui registros e verificações de saúde do Route 53 quando você registra e cancela o registro de instâncias. O caractere curinga (*) em `Resource` concede acesso a todas as AWS Cloud Map instâncias e aos registros e verificações de saúde do Route 53 que pertencem à AWS conta atual.

Crie um exemplo AWS Cloud Map de serviço

Ao adicionar uma política de permissões para permitir que uma identidade do IAM crie um AWS Cloud Map serviço, você deve especificar o Amazon Resource Name (ARN) do AWS Cloud Map namespace e do serviço no campo do recurso. O ARN inclui a região, o ID da conta e o ID do namespace. Como você ainda não sabe qual é a ID do serviço, recomendamos usar um caractere curinga. Veja a seguir um exemplo de trecho de política.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "servicediscovery:CreateService"
            ],
            "Resource": [
                "arn:aws:servicediscovery:us-east-1:111122223333:namespace/ns-
p32123EXAMPLE",
                "arn:aws:servicediscovery:us-east-1:111122223333:service/*"
            ]
        }
    ]
}
```

Exemplo de criação de AWS Cloud Map namespaces

A política de permissões a seguir permite que os usuários criem todos os tipos de AWS Cloud Map namespaces:

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "servicediscovery:CreateHttpNamespace",
            "servicediscovery:CreatePrivateDnsNamespace",
            "servicediscovery:CreatePublicDnsNamespace",
            "route53:CreateHostedZone",
            "route53:GetHostedZone",
            "route53:ListHostedZonesByName",
            "ec2:DescribeVpcs",
            "ec2:DescribeRegions"
         ],
         "Resource":"*"
      }
   ]
}
```

AWS políticas gerenciadas para AWS Cloud Map

Uma política AWS gerenciada é uma política autônoma criada e administrada por AWS. AWS as políticas gerenciadas são projetadas para fornecer permissões para muitos casos de uso comuns, para que você possa começar a atribuir permissões a usuários, grupos e funções.

Lembre-se de que as políticas AWS gerenciadas podem não conceder permissões de privilégio mínimo para seus casos de uso específicos porque elas estão disponíveis para uso de todos os AWS clientes. Recomendamos que você reduza ainda mais as permissões definindo as políticas gerenciadas pelo cliente que são específicas para seus casos de uso.

Você não pode alterar as permissões definidas nas políticas AWS gerenciadas. Se AWS atualizar as permissões definidas em uma política AWS gerenciada, a atualização afetará todas as identidades principais (usuários, grupos e funções) às quais a política está anexada. AWS é mais provável que atualize uma política AWS gerenciada quando uma nova AWS service (Serviço da AWS) é lançada ou novas operações de API são disponibilizadas para serviços existentes.

Para saber mais, consulte AWS Políticas gerenciadas pela no Guia do usuário do IAM.

AWS política gerenciada: AWSCloud MapDiscoverInstanceAccess

É possível anexar `AWSCloudMapDiscoverInstanceAccess` às entidades do IAM. Fornece acesso à API AWS Cloud Map Discovery.

Para visualizar as permissões para esta política, consulte AWSCloudMapDiscoverInstanceAccess na Referência de políticas gerenciadas pela AWS .

AWS política gerenciada: AWSCloud MapReadOnlyAccess

É possível anexar `AWSCloudMapReadOnlyAccess` às entidades do IAM. Concede acesso somente para leitura a todas AWS Cloud Map as ações.

Para visualizar as permissões para esta política, consulte AWSCloudMapReadOnlyAccess na Referência de políticas gerenciadas pela AWS .

AWS política gerenciada: AWSCloud MapRegisterInstanceAccess

É possível anexar `AWSCloudMapRegisterInstanceAccess` às entidades do IAM. Concede acesso somente leitura aos namespaces e serviços, além de permissão para registrar e cancelar o registro de instâncias de serviço.

Para visualizar as permissões para esta política, consulte AWSCloudMapRegisterInstanceAccess na Referência de políticas gerenciadas pela AWS .

AWS política gerenciada: AWSCloud MapFullAccess

É possível anexar `AWSCloudMapFullAccess` às entidades do IAM. Fornece acesso total a todas as AWS Cloud Map ações

Para visualizar as permissões para esta política, consulte AWSCloudMapFullAccess na Referência de políticas gerenciadas pela AWS .

AWS Cloud Map atualizações nas políticas AWS gerenciadas

Veja detalhes sobre as atualizações das políticas AWS gerenciadas AWS Cloud Map desde que esse serviço começou a rastrear essas alterações. Para alertas automáticos sobre alterações, assine o feed RSS na página de histórico do AWS Cloud Map documento.

Alteração	Descrição	Data
AWSCloudMapDiscoverInstanceAccess, AWSCloudMapRegisterInstanceAccess, AWSCloudMapReadOnlyAccess— Atualizações das políticas existentes.	AWS Cloud Map atualizou essas políticas para fornecer acesso às novas operações AWS Cloud Map `DiscoverInstanceRevision` da API.	15 de agosto de 2023

AWS Cloud Map Referência de permissões da API

Ao configurar o controle de acesso e escrever uma política de permissões que você pode anexar a uma identidade do IAM (políticas baseadas em identidade), você pode usar a lista a seguir como referência. A lista inclui cada ação AWS Cloud Map da API e as ações às quais você deve conceder permissões de acesso. Você especifica as ações no `Action` campo para a política. Para obter detalhes sobre o valor do recurso que você deve especificar no `Resource` campo ou na política do IAM, consulte Ações, recursos e chaves de condição AWS Cloud Map na Referência de autorização de serviço.

Você pode usar chaves de condição AWS Cloud Map específicas em suas políticas do IAM para algumas operações. Para obter mais informações, consulte Chaves de condição do AWS Cloud Map na Referência de autorização do serviço.

Para especificar uma ação, use o prefixo `servicediscovery` seguido do nome da ação da API, por exemplo, `servicediscovery:CreatePublicDnsNamespace` e `route53:CreateHostedZone`.

Permissões necessárias para AWS Cloud Map ações

CreateHttpNamespace

Permissões necessárias (ação da API):

- `servicediscovery:CreateHttpNamespace`

CreatePrivateDnsNamespace

Permissões necessárias (ação da API):

- `servicediscovery:CreatePrivateDnsNamespace`

- `route53:CreateHostedZone`
- `route53:GetHostedZone`
- `route53:ListHostedZonesByName`
- `ec2:DescribeVpcs`
- `ec2:DescribeRegions`

CreatePublicDnsNamespace

Permissões necessárias (ação da API):

- `servicediscovery:CreatePublicDnsNamespace`
- `route53:CreateHostedZone`
- `route53:GetHostedZone`
- `route53:ListHostedZonesByName`

CreateService

Permissões obrigatórias (ação de API): `servicediscovery:CreateService`

DeleteNamespace

Permissões necessárias (ação da API):

- `servicediscovery:DeleteNamespace`

DeleteService

Permissões obrigatórias (ação de API): `servicediscovery:DeleteService`

DeleteServiceAttributes

Permissões obrigatórias (ação de API): `servicediscovery:DeleteServiceAttributes`

DeregisterInstance

Permissões necessárias (ação da API):

- `servicediscovery:DeregisterInstance`
- `route53:GetHealthCheck`
- `route53:DeleteHealthCheck`
- `route53:UpdateHealthCheck`

DiscoverInstances

Permissões obrigatórias (ação de API): `servicediscovery:DiscoverInstances`

GetInstance

Permissões obrigatórias (ação de API): `servicediscovery:GetInstance`

GetInstancesHealthStatus

Permissões obrigatórias (ação de API): `servicediscovery:GetInstancesHealthStatus`

GetNamespace

Permissões obrigatórias (ação de API): `servicediscovery:GetNamespace`

GetOperation

Permissões obrigatórias (ação de API): `servicediscovery:GetOperation`

GetService

Permissões obrigatórias (ação de API): `servicediscovery:GetService`

GetServiceAttributes

Permissões obrigatórias (ação de API): `servicediscovery:GetServiceAttributes`

ListInstances

Permissões obrigatórias (ação de API): `servicediscovery:ListInstances`

ListNamespaces

Permissões obrigatórias (ação de API): `servicediscovery:ListNamespaces`

ListOperations

Permissões obrigatórias (ação de API): `servicediscovery:ListOperations`

ListServices

Permissões obrigatórias (ação de API): `servicediscovery:ListServices`

ListTagsForResource

Permissões obrigatórias (ação de API): `servicediscovery:ListTagsForResource`

RegisterInstance

Permissões necessárias (ação da API):

- `servicediscovery:RegisterInstance`
- `route53:GetHealthCheck`
- `route53:CreateHealthCheck`
- `route53:UpdateHealthCheck`
- `ec2:DescribeInstances`

TagResource

Permissões obrigatórias (ação de API): `servicediscovery:TagResource`

UntagResource

Permissões obrigatórias (ação de API): `servicediscovery:UntagResource`

UpdateHttpNamespace

Permissões obrigatórias (ação de API): `servicediscovery:UpdateHttpNamespace`

UpdateInstanceCustomHealthStatus

Permissões obrigatórias (ação de API): `servicediscovery:UpdateInstanceCustomHealthStatus`

UpdatePrivateDnsNamespace

Permissões necessárias (ação da API):

- `servicediscovery:UpdatePrivateDnsNamespace`
- `route53:ChangeResourceRecordSets`

UpdatePublicDnsNamespace

Permissões necessárias (ação da API):

- `servicediscovery:UpdatePublicDnsNamespace`
- `route53:ChangeResourceRecordSets`

UpdateService

Permissões necessárias (ação da API):

- `servicediscovery:UpdateService`

- `route53:GetHealthCheck`
- `route53:CreateHealthCheck`
- `route53:DeleteHealthCheck`
- `route53:UpdateHealthCheck`

UpdateServiceAttributes

Permissões obrigatórias (ação de API): `servicediscovery:UpdateServiceAttributes`

Solução de problemas AWS Cloud Map de identidade e acesso

Use as informações a seguir para ajudá-lo a diagnosticar e corrigir problemas comuns que você pode encontrar ao trabalhar com AWS Cloud Map um IAM.

Tópicos

- Não estou autorizado a realizar uma ação em AWS Cloud Map
- Não estou autorizado a realizar iam: PassRole
- Quero permitir que pessoas fora da minha Conta da AWS acessem meus AWS Cloud Map recursos

Não estou autorizado a realizar uma ação em AWS Cloud Map

Se você receber uma mensagem de erro informando que não tem autorização para executar uma ação, suas políticas deverão ser atualizadas para permitir que você realize a ação.

O erro do exemplo a seguir ocorre quando o usuário do IAM `mateojackson` tenta usar o console para visualizar detalhes sobre um atributo *`my-example-widget`* fictício, mas não tem as permissões `servicediscovery:`*`GetWidget`* fictícias.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 servicediscovery:GetWidget on resource: my-example-widget
```

Nesse caso, a política do usuário `mateojackson` deve ser atualizada para permitir o acesso ao recurso *`my-example-widget`* usando a ação `servicediscovery:`*`GetWidget`*.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Não estou autorizado a realizar iam: PassRole

Se você receber uma mensagem de erro informando que não está autorizado a executar a ação `iam:PassRole`, as suas políticas devem ser atualizadas para permitir que você passe uma função para o AWS Cloud Map.

Alguns Serviços da AWS permitem que você passe uma função existente para esse serviço em vez de criar uma nova função de serviço ou uma função vinculada ao serviço. Para fazê-lo, você deve ter permissões para passar o perfil para o serviço.

O exemplo de erro a seguir ocorre quando uma usuária do IAM chamada `marymajor` tenta utilizar o console para executar uma ação no AWS Cloud Map. No entanto, a ação exige que o serviço tenha permissões concedidas por um perfil de serviço. Mary não tem permissões para passar o perfil para o serviço.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Nesse caso, as políticas de Mary devem ser atualizadas para permitir que ela realize a ação `iam:PassRole`.

Se precisar de ajuda, entre em contato com seu AWS administrador. Seu administrador é a pessoa que forneceu suas credenciais de login.

Quero permitir que pessoas fora da minha Conta da AWS acessem meus AWS Cloud Map recursos

É possível criar um perfil que os usuários de outras contas ou pessoas fora da organização podem usar para acessar seus recursos. É possível especificar quem é confiável para assumir o perfil. Para serviços que oferecem suporte a políticas baseadas em recursos ou listas de controle de acesso (ACLs), você pode usar essas políticas para conceder às pessoas acesso aos seus recursos.

Para saber mais, consulte:

- Para saber se é AWS Cloud Map compatível com esses recursos, consulteComo AWS Cloud Map funciona com o IAM.
- Para saber como fornecer acesso aos seus recursos em todos os Contas da AWS que você possui, consulte Como fornecer acesso a um usuário do IAM em outro Conta da AWS que você possui no Guia do usuário do IAM.

- Para saber como fornecer acesso aos seus recursos a terceiros Contas da AWS, consulte Como fornecer acesso Contas da AWS a terceiros no Guia do usuário do IAM.
- Para saber como conceder acesso por meio da federação de identidades, consulte Conceder acesso a usuários autenticados externamente (federação de identidades) no Guia do usuário do IAM.
- Para conhecer a diferença entre perfis e políticas baseadas em recurso para acesso entre contas, consulte Acesso a recursos entre contas no IAM no Guia do usuário do IAM.

Validação de conformidade para AWS Cloud Map

Para saber se um AWS service (Serviço da AWS) está dentro do escopo de programas de conformidade específicos, consulte Serviços da AWS Escopo por Programa de Conformidade Serviços da AWS e escolha o programa de conformidade em que você está interessado. Para obter informações gerais, consulte Programas de AWS conformidade Programas AWS de .

Você pode baixar relatórios de auditoria de terceiros usando AWS Artifact. Para obter mais informações, consulte Baixar relatórios em AWS Artifact .

Sua responsabilidade de conformidade ao usar Serviços da AWS é determinada pela confidencialidade de seus dados, pelos objetivos de conformidade de sua empresa e pelas leis e regulamentações aplicáveis. Para obter mais informações sobre sua responsabilidade de conformidade ao usar Serviços da AWS, consulte a Documentação AWS de segurança.

Resiliência em AWS Cloud Map

A infraestrutura AWS global é construída em torno de AWS regiões e zonas de disponibilidade. AWS As regiões fornecem várias zonas de disponibilidade fisicamente separadas e isoladas, conectadas a redes de baixa latência, alta taxa de transferência e alta redundância. Com as zonas de disponibilidade, é possível projetar e operar aplicações e bancos de dados que executam o failover automaticamente entre as zonas de disponibilidade sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

AWS Cloud Map é principalmente um serviço global. No entanto, você pode usar AWS Cloud Map para criar verificações de saúde do Route 53 que verificam a integridade dos recursos em regiões específicas, como instâncias do Amazon EC2 e balanceadores de carga do Elastic Load Balancing.

Para obter mais informações sobre AWS regiões e zonas de disponibilidade, consulte Infraestrutura AWS global.

Segurança da infraestrutura em AWS Cloud Map

Como serviço gerenciado, AWS Cloud Map é protegido pela segurança de rede AWS global. Para obter informações sobre serviços AWS de segurança e como AWS proteger a infraestrutura, consulte AWS Cloud Security. Para projetar seu AWS ambiente usando as melhores práticas de segurança de infraestrutura, consulte Proteção de infraestrutura no Security Pillar AWS Well-Architected Framework.

Você usa chamadas de API AWS publicadas para acessar AWS Cloud Map pela rede. Os clientes devem oferecer compatibilidade com:

- Transport Layer Security (TLS). Exigimos TLS 1.2 e recomendamos TLS 1.3.
- Conjuntos de criptografia com perfect forward secrecy (PFS) como DHE (Ephemeral Diffie-Hellman) ou ECDHE (Ephemeral Elliptic Curve Diffie-Hellman). A maioria dos sistemas modernos, como Java 7 e versões posteriores, comporta esses modos.

Você pode melhorar a postura de segurança da sua VPC AWS Cloud Map configurando para usar uma interface VPC endpoint. Para obter mais informações, consulte Acesso AWS Cloud Map usando um endpoint de interface ()AWS PrivateLink.

Acesso AWS Cloud Map usando um endpoint de interface ()AWS PrivateLink

Você pode usar AWS PrivateLink para criar uma conexão privada entre sua VPC e. AWS Cloud Map Você pode acessar AWS Cloud Map como se estivesse em sua VPC, sem o uso de um gateway de internet, dispositivo NAT, conexão VPN ou conexão. Direct Connect As instâncias na sua VPC não precisam de endereços IP públicos para acessar o AWS Cloud Map.

Estabeleça essa conectividade privada criando um endpoint de interface, habilitado pelo AWS PrivateLink. Criaremos um endpoint de interface de rede em cada sub-rede que você habilitar para o endpoint de interface. Estas são interfaces de rede gerenciadas pelo solicitante que servem como ponto de entrada para o tráfego destinado ao AWS Cloud Map.

Para saber mais, consulte Acessar os Serviços da AWS pelo AWS PrivateLink no Guia do AWS PrivateLink .

Considerações para AWS Cloud Map

Antes de configurar um endpoint de interface para AWS Cloud Map, consulte Considerações no AWS PrivateLink Guia.

Se sua Amazon VPC não tiver um gateway de internet e suas tarefas usarem o driver de `awslogs` log para enviar informações de log para CloudWatch Logs, você deverá criar uma interface VPC endpoint para Logs. CloudWatch Para obter mais informações, consulte Como usar CloudWatch registros com endpoints VPC de interface no Guia do usuário do Amazon CloudWatch Logs.

Os VPC endpoints não oferecem suporte AWS a solicitações entre regiões. Garanta a criação do seu endpoint na mesma Região onde planeja emitir as chamadas de API para o AWS Cloud Map.

Os endpoints da VPC oferecem suporte somente a DNS fornecidos pela Amazon por meio do Amazon Route 53. Se quiser usar seu próprio DNS, você pode usar o encaminhamento de DNS condicional. Para obter mais informações, consulte Conjuntos de opções de DHCP no Guia do usuário da Amazon VPC.

O grupo de segurança anexado ao endpoint da VPC deve permitir conexões de entrada na porta 443 na sub-rede privada da Amazon VPC.

Crie um endpoint de interface para AWS Cloud Map

Você pode criar um endpoint de interface para AWS Cloud Map usar o console Amazon VPC ou AWS Command Line Interface o AWS CLI(). Para obter mais informações, consulte Criar um endpoint de interface no Guia do usuário do AWS PrivateLink .

Crie um endpoint de interface para AWS Cloud Map usar os seguintes nomes de serviço:

> **Note**
>
> A API `DiscoverInstances` não estará disponível nesses dois endpoints.

```
com.amazonaws.region.servicediscovery
```

```
com.amazonaws.region.servicediscovery-fips
```

Crie um endpoint de interface para o plano de AWS Cloud Map dados acessar a `DiscoverInstances` API usando os seguintes nomes de serviço:

```
com.amazonaws.region.data-servicediscovery
```

```
com.amazonaws.region.data-servicediscovery-fips
```

> **Note**
>
> Você precisará desativar a injeção de prefixo do host ao fazer a chamada do `DiscoverInstances` com os nomes de VPCE para DNS de região ou zona, para os endpoints do plano de dados. O AWS CLI e AWS SDKs precede o endpoint de serviço com vários prefixos de host quando você chama cada operação de API, o que produz URLS inválidos quando você especifica um VPC endpoint.

Se você habilitar o DNS privado para o endpoint da interface, poderá fazer solicitações de API a AWS Cloud Map usando seu nome DNS regional padrão. Por exemplo, `.servicediscovery.us-east-1.amazonaws.com`

A AWS PrivateLink conexão VPCE é suportada em qualquer região em que AWS Cloud Map há suporte; no entanto, o cliente precisa verificar quais zonas de disponibilidade oferecem suporte ao VPCE antes de definir um endpoint. Para descobrir quais zonas de disponibilidade são compatíveis com endpoints de VPC de interface em uma região, use o describe-vpc-endpoint-services comando ou use o. Console de gerenciamento da AWS Por exemplo, os comandos a seguir retornam as zonas de disponibilidade em que você pode implantar endpoints da VPC para interface AWS Cloud Map na região Leste dos EUA (Ohio):

```
aws --region us-east-2 ec2 describe-vpc-endpoint-services --query 'ServiceDetails[?
ServiceName==`com.amazonaws.us-east-2.servicediscovery`].AvailabilityZones[]'
```

Monitoramento AWS Cloud Map

O monitoramento é uma parte importante da manutenção da confiabilidade, disponibilidade e performance das suas soluções da AWS . Você deve coletar dados de monitoramento de todas as partes da sua AWS solução para poder depurar com mais facilidade uma falha multiponto, caso ocorra. No entanto, antes de iniciar o monitoramento, é necessário criar um plano que inclua respostas às seguintes perguntas:

- Quais são seus objetivos de monitoramento?
- Quais recursos você vai monitorar?
- Com que frequência você vai monitorar esses recursos?
- Quais ferramentas de monitoramento você usará?
- Quem realizará o monitoramento das tarefas?
- Quem deve ser notificado quando algo der errado?

Tópicos

Registre chamadas de AWS Cloud Map API usando AWS CloudTrail

AWS Cloud Map é integrado com AWS CloudTrail, um serviço que fornece um registro das ações realizadas por um usuário, função ou um AWS service (Serviço da AWS). CloudTrail captura todas as chamadas de API AWS Cloud Map como eventos. As chamadas capturadas incluem chamadas do AWS Cloud Map console e chamadas de código para as operações AWS Cloud Map da API. Usando as informações coletadas por CloudTrail, você pode determinar a solicitação que foi feita AWS Cloud Map, o endereço IP do qual a solicitação foi feita, quando foi feita e detalhes adicionais.

Cada entrada de log ou evento contém informações sobre quem gerou a solicitação. As informações de identidade ajudam a determinar o seguinte:

- Se a solicitação foi feita com credenciais de usuário raiz ou credenciais de usuário.
- Se a solicitação foi feita em nome de um usuário do Centro de Identidade do IAM.

- Se a solicitação foi feita com credenciais de segurança temporárias de um perfil ou de um usuário federado.
- Se a solicitação foi feita por outro AWS service (Serviço da AWS).

CloudTrail está ativo Conta da AWS quando você cria a conta e você tem acesso automático ao histórico de CloudTrail eventos. O histórico de CloudTrail eventos fornece um registro visível, pesquisável, baixável e imutável dos últimos 90 dias de eventos de gerenciamento registrados em um. Região da AWS Para obter mais informações, consulte Trabalhando com o histórico de CloudTrail eventos no Guia AWS CloudTrail do usuário. Não há CloudTrail cobrança pela visualização do histórico de eventos.

Para um registro contínuo dos eventos dos Conta da AWS últimos 90 dias, crie uma trilha ou um armazenamento de dados de eventos do CloudTrailLake.

CloudTrail trilhas

Uma trilha permite CloudTrail entregar arquivos de log para um bucket do Amazon S3. Todas as trilhas criadas usando o Console de gerenciamento da AWS são multirregionais. Só é possível criar uma trilha de região única ou de várias regiões usando a AWS CLI. É recomendável criar uma trilha multirregional porque você captura todas as atividades Regiões da AWS em sua conta. Ao criar uma trilha de região única, é possível visualizar somente os eventos registrados na Região da AWS da trilha. Para obter mais informações sobre trilhas, consulte Criar uma trilha para a Conta da AWS e Criar uma trilha para uma organização no Guia do usuário do AWS CloudTrail .

Você pode entregar uma cópia dos seus eventos de gerenciamento contínuos para o bucket do Amazon S3 sem nenhum custo CloudTrail criando uma trilha. No entanto, há cobranças de armazenamento do Amazon S3. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços. Para receber informações sobre a definição de preços do Amazon S3, consulte Definição de preços do Amazon S3.

CloudTrail Armazenamentos de dados de eventos em Lake

CloudTrail O Lake permite que você execute consultas baseadas em SQL em seus eventos. CloudTrail O Lake converte eventos existentes no formato JSON baseado em linhas para o formato Apache ORC. O ORC é um formato colunar de armazenamento otimizado para recuperação rápida de dados. Os eventos são agregados em armazenamentos de dados de eventos, que são coleções imutáveis de eventos baseados nos critérios selecionados com a aplicação de seletores de eventos avançados. Os seletores que aplicados a um armazenamento

de dados de eventos controlam quais eventos persistem e estão disponíveis para consulta. Para obter mais informações sobre o CloudTrail Lake, consulte Trabalhando com o AWS CloudTrail Lake no Guia AWS CloudTrail do Usuário.

CloudTrail Os armazenamentos e consultas de dados de eventos em Lake incorrem em custos. Ao criar um armazenamento de dados de eventos, você escolhe a opção de preço que deseja usar para ele. A opção de preço determina o custo para a ingestão e para o armazenamento de eventos, e o período de retenção padrão e máximo para o armazenamento de dados de eventos. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços.

AWS Cloud Map eventos de dados em CloudTrail

Os eventos de dados fornecem informações sobre as operações de recursos realizadas em ou em um recurso (por exemplo, descobrir uma instância registrada em um namespace). Também são conhecidas como operações de plano de dados. Eventos de dados geralmente são atividades de alto volume. Por padrão, CloudTrail não registra eventos de dados. O histórico de CloudTrail eventos não registra eventos de dados.

Há cobranças adicionais para eventos de dados. Para obter mais informações sobre CloudTrail preços, consulte AWS CloudTrail Preços.

Você pode registrar eventos de dados para os tipos de AWS Cloud Map recursos usando o CloudTrail console ou AWS CLI as operações CloudTrail da API. Para obter mais informações sobre como registrar eventos de dados em log, consulte Registrar eventos de dados com o Console de gerenciamento da AWS e Registrar eventos de dados com a AWS Command Line Interface no Guia do usuário do AWS CloudTrail .

A tabela a seguir lista os tipos de AWS Cloud Map recursos para os quais você pode registrar eventos de dados. A coluna Tipo de evento de dados (console) mostra o valor a ser escolhido na lista Tipo de evento de dados no CloudTrail console. A coluna de valor resources.type mostra o `resources.type` valor, que você especificaria ao configurar seletores de eventos avançados usando o ou. AWS CLI CloudTrail APIs A CloudTrail coluna Dados APIs registrados em mostra as chamadas de API registradas CloudTrail para o tipo de recurso.

Tipo de evento de dados (console)	valor resources.type	Dados APIs registrados em CloudTrail
AwsApiCall	`AWS::ServiceDiscovery::Namespace`	• DiscoverInstances • DiscoverInstancesRevision
AwsApiCall	`AWS::ServiceDiscovery::Service`	• DiscoverInstances • DiscoverInstancesRevision • GetServiceAttributes

É possível configurar seletores de eventos avançados para filtrar os campos `eventName`, `readOnly` e `resources.ARN` para registrar em log somente os eventos que são importantes para você. Para obter mais informações sobre esses campos, consulte AdvancedFieldSelector na Referência de API do AWS CloudTrail .

O exemplo a seguir mostra como configurar seletores de eventos avançados para registrar todos os eventos AWS Cloud Map de dados.

```
"AdvancedEventSelectors":
[
    {
        "Name": "Log all AWS Cloud Map data events",
        "FieldSelectors": [
            { "Field": "eventCategory", "Equals": ["Data"] },
            { "Field": "resources.type", "Equals":
 ["AWS::ServiceDiscovery::Namespace"] }
        ]
    }
]
```

AWS Cloud Map eventos de gerenciamento em CloudTrail

Os eventos de gerenciamento fornecem informações sobre as operações de gerenciamento que são realizadas nos recursos do seu Conta da AWS. Também são conhecidas como operações de ambiente de gerenciamento. Por padrão, CloudTrail registra eventos de gerenciamento.

AWS Cloud Map registra todas as operações do plano de AWS Cloud Map controle como eventos de gerenciamento. Para ver uma lista das operações do plano de AWS Cloud Map controle AWS Cloud Map registradas CloudTrail, consulte a Referência da AWS Cloud Map API.

AWS Cloud Map exemplos de eventos

Um evento representa uma única solicitação de qualquer fonte e inclui informações sobre a operação de API solicitada, a data e a hora da operação, os parâmetros da solicitação e assim por diante. CloudTrail os arquivos de log não são um rastreamento de pilha ordenado das chamadas públicas de API, portanto, os eventos não aparecem em nenhuma ordem específica.

O exemplo a seguir mostra um evento CloudTrail de gerenciamento que demonstra a `CreateHTTPNamespace` operação.


```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE:alejandro_rosalez",
        "arn": "arn:aws:sts::111122223333:assumed-role/users/alejandro_rosalez",
        "accountId": "111122223333",
        "accessKeyId": "AIDACKCEVSQ6C2EXAMPLE",
        "sessionContext": {
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AROA123456789EXAMPLE",
                "arn": "arn:aws:iam::111122223333:role/readonly-role",
                "accountId": "111122223333",
                "userName": "alejandro_rosalez"
            },
            "attributes": {
                "creationDate": "2024-03-19T16:15:37Z",
                "mfaAuthenticated": "false"
            }
        }
    },
    "eventTime": "2024-03-19T19:23:13Z",
    "eventSource": "servicediscovery.amazonaws.com",
    "eventName": "CreateHttpNamespace",
    "awsRegion": "eu-west-3",
    "sourceIPAddress": "192.0.2.0",
```

```
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/122.0.0.0 Safari/537.36",
    "requestParameters": {
        "name": "example-namespace",
        "creatorRequestId": "eda8b524-ca14-4f68-a176-dc4dfd165c26",
        "tags": []
    },
    "responseElements": {
        "operationId": "7xm4i7ghhkaalma666nrg6itf2eylcbp-gwipo38o"
    },
    "requestID": "641274d0-dbbe-4e64-9b53-685769a086c7",
    "eventID": "4a1ab076-ef1b-4bcf-aa95-cec5fb64f2bd",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "servicediscovery.eu-west-3.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```

O exemplo a seguir mostra um evento de CloudTrail dados que demonstra a `DiscoverInstances` operação.

```
{
            "eventVersion": "1.09",
            "userIdentity": {
                "type": "AssumedRole",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE:alejandro_rosalez",
                "arn": "arn:aws:sts::111122223333:assumed-role/role/Admin",
                "accountId": "111122223333",
                "accessKeyId": "AIDACKCEVSQ6C2EXAMPLE",
                "sessionContext": {
                    "sessionIssuer": {
                        "type": "Role",
                        "principalId": "AROA123456789EXAMPLE",
                        "arn": "arn:aws:iam::"111122223333":role/Admin",
                        "accountId": "111122223333",
                        "userName": "Admin"
```

```
                    },
                    "attributes": {
                        "creationDate": "2024-03-19T16:15:37Z",
                        "mfaAuthenticated": "false"
                    }
                }
            },
            "eventTime": "2024-03-19T21:19:12Z",
            "eventSource": "servicediscovery.amazonaws.com",
            "eventName": "DiscoverInstances",
            "awsRegion": "eu-west-3",
            "sourceIPAddress": "13.38.34.79",
            "userAgent": "Boto3/1.20.34 md/Botocore#1.34.60 ua/2.0 os/linux#6.5.0-1014-
aws md/arch#x86_64 lang/python#3.10.12 md/pyimpl#CPython cfg/retry-mode#legacy
 Botocore/1.34.60",
            "requestParameters": {
                "namespaceName": "example-namespace",
                "serviceName": "example-service",
                "queryParameters": {"example-key": "example-value"}
            },
            "responseElements": null,
            "requestID": "e5ee36f1-edb0-4814-a4ba-2e8c97621c79",
            "eventID": "503cedb6-9906-4ee5-83e0-a64dde27bab0",
            "readOnly": true,
            "resources": [
                {
                    "accountId": "111122223333",
                    "type": "AWS::ServiceDiscovery::Namespace",
                    "ARN": "arn:aws:servicediscovery:eu-west-3:111122223333:namespace/
ns-vh4nbmhEXAMPLE"
                },
                {
                    "accountId": "111122223333",
                    "type": "AWS::ServiceDiscovery::Service",
                    "ARN": "arn:aws:servicediscovery:eu-west-3:111122223333:service/
srv-h46op6ylEXAMPLE"
                }
            ],
            "eventType": "AwsApiCall",
            "managementEvent": false,
            "recipientAccountId": "111122223333",
            "eventCategory": "Data",
            "tlsDetails": {
                "tlsVersion": "TLSv1.3",
```

```
                "cipherSuite": "TLS_AES_128_GCM_SHA256",
                "clientProvidedHostHeader": "data-servicediscovery.eu-
west-3.amazonaws.com"
            },
            "sessionCredentialFromConsole": "true"
        }
```

Para obter informações sobre o conteúdo do CloudTrail registro, consulte o conteúdo do CloudTrail registro no Guia AWS CloudTrail do usuário.

Marcando seus recursos AWS Cloud Map

Uma tag é um rótulo que você atribui a um AWS recurso. Cada tag consiste em uma chave e um valor opcional, ambos definidos por você.

As tags permitem que você categorize seus AWS recursos por, por exemplo, finalidade, proprietário ou ambiente. Caso possua muitos recursos do mesmo tipo, você pode identificar rapidamente um recurso específico com base nas tags atribuídas a ele. Por exemplo, você pode definir um conjunto de tags para seus AWS Cloud Map serviços para ajudá-lo a rastrear o proprietário e o nível de pilha de cada serviço. Recomendamos planejar um conjunto consistente de chaves de tags para cada tipo de recurso.

Tags não são automaticamente atribuídas aos recursos. Após adicionar uma tag, você pode editar as chaves e os valores das tags ou removê-las de um recurso a qualquer momento. Caso exclua um recurso, todas as respectivas tags também serão excluídas.

As tags não têm nenhum significado semântico AWS Cloud Map e são interpretadas estritamente como uma sequência de caracteres. É possível definir o valor de uma tag em uma string vazia, mas não configurar o valor de um tag como nula. Caso adicione uma tag com a mesma chave de outra existente no recurso, o novo valor substituirá o antigo.

Você pode trabalhar com tags usando a Console de gerenciamento da AWS AWS CLI, a e a AWS Cloud Map API.

Se você estiver usando AWS Identity and Access Management (IAM), você pode controlar quais usuários em sua AWS conta têm permissão para criar, editar ou excluir tags.

Como os recursos são marcados

Você pode marcar AWS Cloud Map namespaces e serviços novos ou existentes.

Se você estiver usando o AWS Cloud Map console, poderá aplicar tags aos novos recursos quando eles forem criados ou aos recursos existentes a qualquer momento usando a guia Tags na página do recurso relevante.

Se você estiver usando a AWS Cloud Map API AWS CLI, o ou um AWS SDK, poderá aplicar tags a novos recursos usando o `tags` parâmetro na ação relevante da API ou aos recursos existentes usando a ação da TagResourceAPI. Para obter mais informações, consulte TagResource.

Algumas ações de criação de recursos permitem especificar tags para um recurso quando o mesmo for criado. Caso as tags não possam ser aplicadas durante a criação dos recursos, haverá falha no processo de criação de recursos. Isso garante que recursos que você pretenda marcar na criação sejam criados com as tags especificadas ou não. Caso marque recursos no momento da criação, não precisará executar scripts de marcação personalizados após a criação do recurso.

A tabela a seguir descreve os AWS Cloud Map recursos que podem ser marcados e os recursos que podem ser marcados na criação.

Suporte de marcação para recursos AWS Cloud Map

Recurso	Compatível com tags	Compatível com a propagação de tags	Suporta marcação na criação (AWS Cloud Map API AWS CLI, AWS SDK)
AWS Cloud Map namespaces	Sim	Não. As tags de namespace não são propagadas para nenhum outro recurso associado ao namespace.	Sim
AWS Cloud Map serviços	Sim	Não. As tags de serviço não são propagadas para nenhum outro recurso associado ao serviço.	Sim

Restrições

As restrições básicas a seguir se aplicam a tags:

- O número máximo de tags para cada recurso – 50
- Em todos os recursos, cada chave de tag deve ser exclusiva e possuir apenas um valor.
- Comprimento máximo da chave –- 128 caracteres Unicode em UTF-8
- Comprimento máximo do valor: 256 caracteres Unicode em UTF-8

- Se seu esquema de marcação for usado em vários AWS serviços e recursos, lembre-se de que outros serviços podem ter restrições quanto aos caracteres permitidos. Em geral, caracteres permitidos incluem letras, números, espaços representáveis em UTF-8 e os caracteres + - = . _ : / @.
- Chaves e valores de tags diferenciam maiúsculas de minúsculas.
- Não use `aws:AWS:`, ou qualquer combinação de maiúsculas ou minúsculas, como prefixo, para chaves ou valores, pois está reservado para uso. AWS Você não pode editar nem excluir chaves nem valores de tags com esse prefixo. Tags com esse prefixo não contam para seu tags-per-resource limite.

Atualização de tags para AWS Cloud Map recursos

Use os seguintes AWS CLI comandos ou operações de AWS Cloud Map API para adicionar, atualizar, listar e excluir as tags dos seus recursos.

Suporte de marcação para recursos AWS Cloud Map

Tarefa	Ação API	AWS CLI	AWS Tools for Windows PowerShell
Adicione ou sobrescreva uma ou mais tags.	TagResource	tag-resource	Adicionar - SDResource Etiqueta
Exclua uma ou mais tags.	UntagResource	untag-resource	Remover- SDResource Tag
Lista de tags para um recurso	ListTagsForResource	list-tags-for-resource	Obter uma SDResource etiqueta

Os exemplos a seguir mostram como marcar ou desmarcar recursos usando AWS CLI.

Exemplo 1: marcar um recurso existente

O comando a seguir marca um recurso existente.

```
aws servicediscovery tag-resource --resource-arn resource_ARN --tags team=devs
```

Exemplo 2: desmarcar um recurso existente

O comando a seguir exclui uma tag de um recurso existente.

```
aws servicediscovery untag-resource --resource-arn resource_ARN --tag-keys tag_key
```

Exemplo 3: listar etiquetas para um recurso

O comando a seguir lista as tags associadas a um recurso existente.

```
aws servicediscovery list-tags-for-resource --resource-arn resource_ARN
```

Algumas ações de criação de recursos permitem especificar tags ao criar o recurso. As ações a seguir são compatíveis com o uso de tags na criação.

Tarefa	Ação API	AWS CLI	AWS Tools for Windows PowerShell
Criar um namespace HTTP	CreateHttpNamespace	create-http-namespace	Novo - SDHttp Namespace
Criar um namespace privado com base no DNS	CreatePrivateDnsNamespace	create-private-dns-namespace	Novo- SDPrivate DnsNamespace
Criar um namespace público com base no DNS	CreatePublicDnsNamespace	create-public-dns-namespace	Novo- SDPublic DnsNamespace
Criar um serviço	CreateService	create-service	Novo- SDService

AWS Cloud Map cotas de serviço

AWS Cloud Map os recursos estão sujeitos às seguintes cotas de serviço em nível de conta. Cada cota listada se aplica a cada AWS região em que você cria AWS Cloud Map recursos.

Name	Padrão	Ajustá	Description
Atributos personalizados por instância	Cada região compatível: 30	Não	O número máximo de atributos personalizados que você pode especific ar ao registrar uma instância.
DiscoverInstances taxa de explosão de operação por conta	Cada região com suporte: 2.000	Sim	A taxa máxima de intermitência para a DiscoverInstances operação de chamadas a partir de uma única conta.
DiscoverInstances taxa estável de operação por conta	Cada região com suporte: 1.000	Sim	A taxa fixa máxima para a DiscoverInstances operação de chamadas a partir de uma única conta.
DiscoverInstancesRevision taxa de operação por conta	Cada região compatível: 3.000	Sim	A taxa máxima para a DiscoverInstancesR evision operação de chamadas a partir de uma única conta.
Instâncias por namespace	Cada região compatível: 2.000	Sim	O número máximo de instâncias de serviço que você pode registrar usando o mesmo namespace.
Instâncias por serviço	Cada região com suporte: 1.000	Não	O número máximo de instâncias de serviço que

Name	Padrão	Ajustá	Description
			você pode registrar em uma Região usando o mesmo serviço.
Namespaces por região	Cada região compatível: 50	Sim	O número máximo de repositórios que você pode criar por Região.

* Quando você cria um namespace, nós criamos automaticamente uma zona hospedada do Amazon Route 53. Essa zona hospedada é contabilizada na cota do número de zonas hospedadas que você pode criar com uma AWS conta. Para obter mais informações, consulte Cotas em zonas hospedadas no Guia do desenvolvedor do Amazon Route 53.

** Para aumentar as instâncias de namespaces DNS para AWS Cloud Map é necessário um aumento no limite de registros por zona hospedada do Route 53, gerando cobranças adicionais.

Gerenciando suas cotas AWS Cloud Map de serviço

AWS Cloud Map foi integrado ao Service Quotas, um AWS serviço que permite visualizar e gerenciar suas cotas a partir de um local central. Para obter mais informações, consulte O que são cotas de serviço? no Guia do usuário do Service Quotas.

As Cotas de Serviço facilitam a pesquisa do valor de suas cotas de AWS Cloud Map serviço.

Console de gerenciamento da AWS

Para visualizar as cotas de AWS Cloud Map serviço usando o Console de gerenciamento da AWS

1. Abra o console do Service Quotas em https://console.aws.amazon.com/servicequotas/.
2. No painel de navegação, escolha Serviços da AWS .
3. Na lista de serviços da AWS , procure e selecione AWS Cloud Map.
4. Na lista de cotas de serviço para AWS Cloud Map, você pode ver o nome da cota de serviço, o valor aplicado (se estiver disponível), a cota AWS padrão e se o valor da cota é ajustável.

Para ver informações adicionais sobre uma cota de serviço, como a descrição, escolha o nome da cota para exibir os detalhes da cota.

5. (Opcional) Para solicitar um aumento de cota, selecione a cota que você deseja aumentar e escolha Solicitar aumento no nível da conta.

Para trabalhar mais com cotas de serviço usando o, Console de gerenciamento da AWS consulte o Guia do usuário de cotas de serviço.

AWS CLI

Para visualizar as cotas de AWS Cloud Map serviço usando o AWS CLI

Execute o comando a seguir para ver as AWS Cloud Map cotas padrão.

```
aws service-quotas list-aws-default-service-quotas \
    --query 'Quotas[*].
{Adjustable:Adjustable,Name:QuotaName,Value:Value,Code:QuotaCode}' \
    --service-code AWSCloudMap \
    --output table
```

Execute o comando a seguir para ver suas AWS Cloud Map cotas aplicadas.

```
aws service-quotas list-service-quotas \
    --service-code AWSCloudMap
```

Para obter mais informações sobre como trabalhar com cotas de serviço usando o AWS CLI, consulte a Referência de Comandos de AWS CLI Cotas de Serviço. Para solicitar um aumento de quotas, consulte o comando `request-service-quota-increase` na Referência de comandos do AWS CLI.

Lidar com a limitação de solicitações de AWS Cloud Map DiscoverInstances API

AWS Cloud Map limita as solicitações de DiscoverInstancesAPI para cada AWS conta por região. A limitação ajuda a melhorar o desempenho do serviço e ajuda a fornecer um uso justo para todos os AWS Cloud Map clientes. A limitação garante que as chamadas para a AWS Cloud Map DiscoverInstancesAPI não excedam as cotas máximas permitidas de solicitações de

DiscoverInstancesAPI. DiscoverInstances As chamadas de API provenientes de qualquer uma das seguintes fontes estão sujeitas às cotas de solicitação:

- Aplicativos de terceiros
- Uma ferramentas da linha de comando
- O AWS Cloud Map console

Se você exceder a cota de controle de utilização de API, receberá o código de erro `RequestLimitExceeded`. Para obter mais informações, consulte the section called “Limitação de intervalo de solicitações”.

Como o controle de utilização é aplicado

AWS Cloud Map usa o algoritmo de token bucket para implementar a limitação de API. Com esse algoritmo, sua conta tem um bucket que contém um número específico de tokens. O número de tokens no bucket representa sua cota de controle de utilização a qualquer segundo. Há um bucket para cada região e ele se aplica a todos os endpoints na região.

Limitação de intervalo de solicitações

A limitação limita o número de solicitações de DiscoverInstancesAPI que você pode fazer. Cada solicitação feita remove um token do bucket. Por exemplo, o tamanho do bucket para a operação da DiscoverInstancesAPI é de 2.000 tokens, então você pode fazer até 2.000 DiscoverInstancessolicitações em um segundo. Se você exceder as 2.000 solicitações em um segundo, você será limitado pelo controle de utilização e as solicitações excedentes nesse segundo falharão.

Os buckets são recarregados automaticamente a uma taxa definida. Se o bucket não atingir a capacidade máxima, um determinado número de tokens será adicionado novamente a cada segundo até que o bucket atinja a capacidade máxima. Se o bucket atingir a capacidade máxima quando os tokens de recarga forem adicionados, esses tokens serão descartados. O tamanho do bucket para a operação da DiscoverInstancesAPI é de 2.000 tokens e a taxa de recarga é de 1.000 tokens a cada segundo. Se você fizer 2.000 solicitações de DiscoverInstancesAPI em um segundo, o bucket será imediatamente reduzido para zero (0) tokens. O bucket é, então, reabastecido com até 1.000 tokens a cada segundo até atingir sua capacidade máxima de 2.000 tokens.

Você pode usar tokens à medida que eles são adicionados ao bucket. Para fazer solicitações de API, não é necessário esperar que o bucket atinja sua capacidade máxima. Se você esgotar o

bucket fazendo 2.000 solicitações de DiscoverInstancesAPI em um segundo, ainda poderá fazer até 1.000 solicitações de DiscoverInstancesAPI a cada segundo depois disso, pelo tempo que precisar. Isso significa que você pode usar imediatamente os tokens de recarga à medida que eles são adicionados ao seu bucket. O bucket só começa a ser recarregado até a capacidade máxima quando você faz menos solicitações de API a cada segundo do que a taxa de recarga.

Repetições ou processamento em lote

Caso uma solicitação de API falhe, seu aplicativo pode precisar repetir a solicitação. Para reduzir a taxa de solicitações de API, use um intervalo de latência apropriado entre as solicitações sucessivas. Para obter os melhores resultados, use um intervalo de latência crescente ou variável.

Calcular o intervalo de repouso

Quando você precisar fazer a sondagem ou repetir uma solicitação de API, é recomendável usar um algoritmo de recuo exponencial para calcular o intervalo de latência entre as chamadas de API. Ao usar tempos de espera progressivamente maiores entre as novas tentativas de respostas de erro consecutivas, é possível reduzir o número de solicitações com falha. Para obter mais informações e exemplos de implementação desse algoritmo, consulte Retry Behavior no Guia de referência de ferramentas AWS SDKs e ferramentas.

Ajustar as cotas de controle de utilização da API

Você pode solicitar um aumento nas cotas de limitação de API para sua conta. AWS Para solicitar um ajuste de cota, entre em contato com a Central do AWS Support.

Histórico do documento para AWS Cloud Map

A tabela a seguir descreve as principais atualizações e novos atributos para o Guia do usuário do AWS Cloud Map . Também atualizamos a documentação com frequência para abordar os comentários enviados por você.

Alteração	Descrição	Data
AWS Cloud Map compartilhamento de namespace entre contas	Agora você pode compartilhar namespaces com outras pessoas Contas da AWS ou dentro de uma organização AWS Organizations usando AWS Resource Access Manager (AWS RAM) para simplificar a descoberta e o registro de serviços entre contas.	14 de agosto de 2025
AWS Cloud Map atributos de serviço	Agora você pode especificar atributos no nível do serviço para evitar a duplicação de atributos nas instâncias registradas em um serviço. Você pode usar esses atributos para roteamento de tráfego complexo, definir valores de tempo limite e de repetição e para coordenação entre serviços e integrações externas.	13 de dezembro de 2024
Tutoriais adicionados	Dois tutoriais mostrando casos de uso comuns de uso AWS Cloud Map foram adicionados.	27 de março de 2024

CloudTrail documentação de integração atualizada	A documentação que descreve a AWS Cloud Map integração com CloudTrail o log da atividade da API foi atualizada.	20 de março de 2024
Atualização da política gerenciada	As políticas de `AWSCloudMapDiscoverInstanceAccess` ,`AWSCloudMapRegisterInstanceAccess` e `AWSCloudMapReadOnlyAccess` foram atualizadas.	20 de setembro de 2023
Cloud Map e AWS PrivateLink	Agora você pode usar um AWS PrivateLink para criar uma conexão privada entre sua VPC e. AWS Cloud Map	15 de setembro de 2023
Atualização da política gerenciada	A política de `AWSCloudMapDiscoverInstanceAccess` foi atualizada.	15 de agosto de 2023
AWS SDK para Python	Foram adicionados exemplos de linha de comando do Python.	13 de setembro de 2022
IPv6 apoio	Os endpoints da API estão disponíveis somente em redes `IPv6`.	28 de janeiro de 2022

Descoberta de instâncias de serviço	AWS Cloud Map adicionou suporte para a criação de serviços em um namespace que oferece suporte a consultas de DNS que podem ser descobertas somente usando a operação de DiscoverInstancesAPI e não usando consultas de DNS.	24 de março de 2021
Marcação de recursos	AWS Cloud Map adicionou suporte para adicionar tags de metadados aos seus namespaces e serviços usando o. Console de gerenciamento da AWS	8 de fevereiro de 2021
Marcação de recursos	AWS Cloud Map adicionou suporte para adicionar tags de metadados aos seus namespaces e serviços usando o e. AWS CLI APIs	22 de junho de 2020
Versão inicial	Esta é a primeira versão do Guia do desenvolvedor do AWS Cloud Map .	28 de novembro de 2018

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.