Guida per gli sviluppatori

AWS Cloud Map

AWS Cloud Map: Guida per gli sviluppatori

Table of Contents

Che cos'è AWS Cloud Map?

AWS Cloud Map è una soluzione completamente gestita che puoi utilizzare per mappare nomi logici ai servizi e alle risorse di backend da cui dipendono le tue applicazioni. Inoltre, aiuta le applicazioni a scoprire le risorse utilizzando una delle AWS SDKs chiamate RESTful API o le query DNS. AWS Cloud Map serve solo risorse sane, che possono essere tabelle Amazon DynamoDB (DynamoDB), code Amazon Simple Queue Service (Amazon SQS), qualsiasi servizio applicativo di livello superiore creato utilizzando EC2 istanze Amazon Elastic Compute Cloud (Amazon) o attività Amazon Elastic Container Service (Amazon ECS) e altro ancora.

Componenti di AWS Cloud Map

Spazio dei nomi

Per iniziare, devi prima creare uno spazio dei AWS Cloud Map nomi che funzioni come un modo per raggruppare i servizi per un'applicazione. Un namespace identifica il nome che desideri utilizzare per localizzare le risorse e specifica anche come desideri localizzare le risorse: utilizzando chiamate AWS Cloud Map DiscoverInstancesAPI, query DNS in un VPC o query DNS pubbliche. Nella maggior parte dei casi, uno spazio dei nomi contiene tutti i servizi di un'applicazione, ad esempio un'applicazione di fatturazione. Per ulteriori informazioni, consulta AWS Cloud Map namespace.

Servizio

Dopo aver creato uno spazio dei nomi, crei un AWS Cloud Map servizio per ogni tipo di risorsa che desideri utilizzare per localizzare gli endpoint. AWS Cloud Map Ad esempio, è possibile creare servizi per server Web e server di database.

Un servizio è un modello che viene AWS Cloud Map utilizzato quando l'applicazione aggiunge un'altra risorsa, ad esempio un altro server Web. Se hai deciso di individuare le risorse utilizzando DNS al momento della creazione dello spazio dei nomi, un servizio contiene le informazioni sui tipi di record da utilizzare per individuare il server Web. Un servizio indica anche se desideri verificare lo stato della risorsa e se desideri utilizzare i controlli dello stato di Amazon Route 53 o un dispositivo di controllo dello stato di terze parti. Per ulteriori informazioni, consulta AWS Cloud Map servizi.

Istanza del servizio

Quando l'applicazione aggiunge una risorsa, puoi richiamare l'azione AWS Cloud Map RegisterInstanceAPI nel codice, che crea un'istanza di AWS Cloud Map servizio in un servizio. L'istanza del servizio contiene informazioni su come l'applicazione può localizzare la risorsa, utilizzando DNS o utilizzando l'azione AWS Cloud Map DiscoverInstancesAPI.

Quando l'applicazione deve connettersi a una risorsa, chiama DiscoverInstanceso utilizza query DNS pubbliche o private specificando lo spazio dei nomi e il servizio associati alla risorsa. AWS Cloud Map restituisce informazioni su come individuare una o più risorse. Se hai specificato il controllo dello stato quando hai creato il servizio, AWS Cloud Map restituisce solo le istanze integre. Per ulteriori informazioni, consulta AWS Cloud Map istanze di servizio.

Accesso AWS Cloud Map

È possibile accedere AWS Cloud Map nei seguenti modi:

- AWS Management Console— Le procedure riportate in questa guida spiegano come utilizzarlo AWS Management Console per eseguire attività.
- AWS SDKs— Se utilizzi un linguaggio di programmazione che AWS fornisce un SDK per, puoi utilizzare un SDK per accedere. AWS Cloud Map SDKs semplifica l'autenticazione, si integra facilmente con il tuo ambiente di sviluppo e fornisci l'accesso ai AWS Cloud Map comandi. Per ulteriori informazioni, consulta Strumenti per Amazon Web Services.
- AWS Command Line Interface— Per ulteriori informazioni, consulta la Guida introduttiva AWS CLI alla Guida AWS Command Line Interface per l'utente.
- AWS Tools for Windows PowerShell— Per ulteriori informazioni, vedere la Guida introduttiva AWS Tools for Windows PowerShell nella Guida per l'AWS Strumenti per PowerShell utente.
- AWS Cloud Map API: se utilizzi un linguaggio di programmazione per il quale non è disponibile un SDK, consulta l'AWS Cloud Map API Reference per informazioni sulle azioni API e su come effettuare richieste API.

> **Note**
>
> IPv6 Supporto clienti: a partire dal 22 giugno 2023 in tutte le nuove regioni, tutti i comandi inviati AWS Cloud Map dai `IPv6` client vengono indirizzati a un nuovo endpoint dualstack (). `servicediscovery.<region>.api.aws` AWS Cloud Map `IPv6`-solo le reti sono raggiungibili sia per gli endpoint legacy

(**`servicediscovery.<region>.amazonaws.com`**) che dualstack nelle seguenti regioni rilasciate prima del 22 giugno 2023:

- Stati Uniti orientali (Ohio) - us-east-2
- Stati Uniti orientali (Virginia settentrionale) - us-east-1
- Stati Uniti occidentali (California settentrionale) - us-west-1
- Stati Uniti occidentali (Oregon) - us-west-2
- Africa (Città del Capo) – af-south-1
- Asia Pacifico (Hong Kong) - ap-east-1
- Asia Pacifico (Hyderabad) — ap-south-2
- Asia Pacifico (Giacarta) – ap-southeast-3
- Asia Pacifico (Melbourne) — ap-southeast-4
- Asia Pacifico (Mumbai) - ap-south-1
- Asia Pacifico (Osaka) - ap-northeast-3
- Asia Pacifico (Seoul) - ap-northeast-2
- Asia Pacifico (Singapore) - ap-southeast-1
- Asia Pacifico (Sydney) - ap-southeast-2
- Asia Pacifico (Tokyo) - ap-northeast-1
- Canada (Centrale) - ca-central-1
- UE (Francoforte) - eu-central-1
- Europa (Irlanda) - eu-west-1
- Europa (Londra) - eu-west-2
- Europa (Milano) – eu-south-1
- Europa (Parigi) - eu-ovest-3
- Europa (Spagna) — eu-south-2
- Europa (Stoccolma) - eu-nord-1
- Europa (Zurigo) — eu-central-2
- Medio Oriente (Bahrein) – me-south-1
- Medio Oriente (EAU) — me-central-1
- Sud America (San Paolo) - sa-east-1
- AWS GovCloud (Stati Uniti orientali) — -1 us-gov-east

- AWS GovCloud (Stati Uniti occidentali) — -1 us-gov-west

AWS Identity and Access Management

AWS Cloud Map si integra con AWS Identity and Access Management (IAM), un servizio che l'organizzazione può utilizzare per eseguire le seguenti azioni:

- Crea utenti e gruppi con l'account della AWS tua organizzazione
- Condividi le risorse del tuo AWS account tra gli utenti dell'account in modo efficiente
- Assegnare credenziali di sicurezza univoche a ciascun utente
- Controllare in modo granulare l'accesso dell'utente a servizi e risorse

Ad esempio, puoi utilizzare IAM con AWS Cloud Map per controllare quali utenti del tuo AWS account possono creare un nuovo namespace o registrare istanze.

Per informazioni generali su IAM, consulta le seguenti risorse:

- Identity and Access Management per AWS Cloud Map
- AWS Identity and Access Management
- Guida per l'utente di IAM

AWS Cloud Map Prezzi

AWS Cloud Map i prezzi si basano sulle risorse registrate nel registro dei servizi e sulle chiamate API effettuate per scoprirle. Non AWS Cloud Map sono previsti pagamenti anticipati e paghi solo per ciò che utilizzi.

Facoltativamente, è possibile impostare il rilevamento basato su DNS per le risorse con indirizzi IP. Puoi anche abilitare il controllo dello stato delle tue risorse utilizzando i controlli di integrità di Amazon Route 53, indipendentemente dal fatto che tu stia scoprendo istanze utilizzando chiamate API o query DNS. Saranno addebitati costi aggiuntivi relativi al DNS di Route 53 e all'utilizzo dei controlli sanitari.

Per ulteriori informazioni, consulta Prezzi di AWS Cloud Map.

AWS Cloud Map e Cloud Compliance AWS

Per informazioni sulla AWS Cloud Map conformità a varie normative di conformità alla sicurezza e standard di audit, consulta le pagine seguenti:

- AWS Conformità al cloud
- AWS Servizi compresi nell'ambito del programma di conformità

Guida introduttiva con AWS Cloud Map

Le seguenti guide mostrano come configurare l'uso AWS Cloud Map e l'esecuzione di attività comuni utilizzando AWS Cloud Map i namespace.

Panoramica della guida	Ulteriori informazioni
Iscrizione AWS e preparazione all'uso AWS Cloud Map	Configurazione per l'uso AWS Cloud Map
Utilizzo di query DNS e chiamate API per scoprire i servizi di backend.	Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con le query DNS e le chiamate API
Utilizzo di query DNS e chiamate API per scoprire i servizi di backend utilizzando. AWS CLI	Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con le query DNS e le chiamate API utilizzando AWS CLI
Creazione di un'applicazione di esempio e utilizzo di attributi personalizzati nel codice per scoprire risorse.	Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati
Creazione di un'applicazione di esempio e utilizzo di attributi personalizzati nel codice per scoprire risorse utilizzando AWS CLI.	Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati utilizzando AWS CLI

Configurazione per l'uso AWS Cloud Map

La panoramica e le procedure riportate nelle sezioni seguenti hanno lo scopo di aiutarti a iniziare a usarlo AWS e prepararti a iniziare a AWS Cloud Map utilizzarlo.

Argomenti

- Iscriviti per AWS
- Accedi all' AWS CLI API AWS Tools for Windows PowerShell o al AWS SDKs
- Configura o AWS Command Line InterfaceAWS Tools for Windows PowerShell
- Scarica un AWS SDK

Iscriviti per AWS

Iscriviti per un Account AWS

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la https://portal.aws.amazon.com/billing/registrazione.
2. Segui le istruzioni online.

   Nel corso della procedura di registrazione riceverai una telefonata o un messaggio di testo e ti verrà chiesto di inserire un codice di verifica attraverso la tastiera del telefono.

   Quando ti iscrivi a un Account AWS, Utente root dell'account AWSviene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire attività che richiedono l'accesso di un utente root.

AWS ti invia un'email di conferma dopo il completamento della procedura di registrazione. In qualsiasi momento, puoi visualizzare l'attività corrente del tuo account e gestirlo accedendo a https://aws.amazon.com/e scegliendo Il mio account.

Crea un utente con accesso amministrativo

Dopo esserti registrato Account AWS, proteggi Utente root dell'account AWS AWS IAM Identity Center, abilita e crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi AWS Management Consolecome proprietario dell'account scegliendo Utente root e inserendo il tuo indirizzo Account AWS email. Nella pagina successiva, inserisci la password.

   Per informazioni sull'accesso utilizzando un utente root, consulta la pagina Signing in as the root user della Guida per l'utente di Accedi ad AWS .
2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

   Per istruzioni, consulta Abilitare un dispositivo MFA virtuale per l'utente Account AWS root (console) nella Guida per l'utente IAM.

Crea un utente con accesso amministrativo

1. Abilita Centro identità IAM.

   Per istruzioni, consulta Abilitazione di AWS IAM Identity Center nella Guida per l'utente di AWS IAM Identity Center .

2. In IAM Identity Center, assegna l'accesso amministrativo a un utente.

   Per un tutorial sull'utilizzo di IAM Identity Center directory come fonte di identità, consulta Configurare l'accesso utente con l'impostazione predefinita IAM Identity Center directory nella Guida per l'AWS IAM Identity Center utente.

Accesso come utente amministratore

- Per accedere con l'utente IAM Identity Center, utilizza l'URL di accesso che è stato inviato al tuo indirizzo e-mail quando hai creato l'utente IAM Identity Center.

  Per informazioni sull'accesso utilizzando un utente IAM Identity Center, consulta AWS Accedere al portale di accesso nella Guida per l'Accedi ad AWS utente.

Assegna l'accesso a ulteriori utenti

1. In IAM Identity Center, crea un set di autorizzazioni conforme alla best practice dell'applicazione di autorizzazioni con il privilegio minimo.

   Segui le istruzioni riportate nella pagina Creazione di un set di autorizzazioni nella Guida per l'utente di AWS IAM Identity Center .

2. Assegna al gruppo prima gli utenti e poi l'accesso con autenticazione unica (Single Sign-On).

   Per istruzioni, consulta Aggiungere gruppi nella Guida per l'utente di AWS IAM Identity Center .

Accedi all' AWS CLI API AWS Tools for Windows PowerShell o al AWS SDKs

Per utilizzare l'API, il AWS CLI AWS Tools for Windows PowerShell, o il AWS SDKs, devi creare chiavi di accesso. Queste chiavi sono composte da un ID chiave di accesso e una chiave di accesso segreta, che vengono utilizzati per firmare le richieste a livello di programmazione che fai ad AWS.

Gli utenti necessitano dell'accesso programmatico se desiderano interagire con l' AWS AWS Management Console esterno di. Il modo per concedere l'accesso programmatico dipende dal tipo di utente che accede. AWS

Per fornire agli utenti l'accesso programmatico, scegli una delle seguenti opzioni.

Quale utente necessita dell'accesso programmatico?	Per	Come
Identità della forza lavoro (Utenti gestiti nel centro identità IAM)	Utilizza credenziali temporane e per firmare le richieste programmatiche a AWS CLI,, AWS SDKs o. AWS APIs	Segui le istruzioni per l'interfa ccia che desideri utilizzare. • Per la AWS CLI, vedere Configurazione dell'uso AWS IAM Identity Center nella AWS CLI Guida per l'utente.AWS Command Line Interface • Per AWS SDKs gli strumenti e AWS APIs, consulta l'autenticazione di IAM Identity Center nella Guida di riferimento AWS SDKs and Tools.
IAM	Utilizza credenziali temporane e per firmare le richieste programmatiche a AWS CLI, AWS SDKs, o. AWS APIs	Seguendo le istruzioni riportate in Utilizzo delle credenziali temporanee con le AWS risorse nella Guida per l'utente IAM.
IAM	(Non consigliato) Utilizza credenziali a lungo termine per firmare richieste programmatiche a AWS CLI,, AWS SDKs o. AWS APIs	Segui le istruzioni per l'interfa ccia che desideri utilizzare. • Per la AWS CLI, consulta Autenticazione tramite credenziali utente IAM nella

Quale utente necessita dell'accesso programmatico?	Per	Come
		Guida per l'utente.AWS Command Line Interface • Per gli strumenti AWS SDKs e gli strumenti, consulta Autenticazione tramite credenziali a lungo termine nella Guida di riferimento agli strumenti e agli AWS SDKs strumenti. • Per AWS APIs, consulta la sezione Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente IAM.

Configura o AWS Command Line InterfaceAWS Tools for Windows PowerShell

AWS Command Line Interface (AWS CLI) è uno strumento unificato per la gestione dei AWS servizi. Per informazioni su come installare e configurare AWS CLI, vedere Installazione o aggiornamento alla versione più recente di AWS CLI nella Guida per l'AWS Command Line Interface utente.

Se hai esperienza con Windows PowerShell, potresti preferire utilizzare AWS Tools for Windows PowerShell. Per ulteriori informazioni, consulta Configurazione della AWS Tools for Windows PowerShell nella Guida per l'utente di AWS Strumenti per PowerShell .

Scarica un AWS SDK

Se utilizzi un linguaggio di programmazione che AWS fornisce un SDK per, ti consigliamo di utilizzare un SDK anziché l'API. AWS Cloud Map L'utilizzo di un SDK offre diversi vantaggi. SDKs semplifica l'autenticazione, si integra facilmente con il tuo ambiente di sviluppo e fornisci l'accesso ai AWS Cloud Map comandi. Per ulteriori informazioni, consulta Strumenti per Amazon Web Services.

Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con le query DNS e le chiamate API

Il seguente tutorial simula un'architettura di microservizi con due servizi di backend. Il primo servizio sarà individuabile utilizzando una query DNS. Il secondo servizio sarà individuabile solo tramite l' AWS Cloud Map API.

> **Note**
>
> I dettagli delle risorse, come i nomi di dominio e gli indirizzi IP, sono solo a scopo di simulazione. Non possono essere risolti su Internet.

Per una end-to-end AWS CLI versione di questo tutorial, vediScopri come utilizzare il rilevamento dei AWS Cloud Map servizi con le query DNS e le chiamate API utilizzando AWS CLI.

Prerequisiti

I seguenti prerequisiti devono essere soddisfatti per completare correttamente il tutorial.

- Prima di iniziare, completa i passaggi descritti in Configurazione per l'uso AWS Cloud Map.
- Se non l'hai ancora installato AWS Command Line Interface, segui i passaggi indicati in Installazione o aggiornamento della versione più recente di AWS CLI per installarlo.

  Per eseguire i comandi nel tutorial, sono necessari un terminale a riga di comando o una shell (interprete di comandi). In Linux e macOS, utilizza la shell (interprete di comandi) e il gestore pacchetti preferiti.

  > **Note**
  >
  > Su Windows, alcuni comandi della CLI Bash utilizzati comunemente con Lambda (ad esempio, `zip`) non sono supportati dai terminali integrati del sistema operativo. Per ottenere una versione integrata su Windows di Ubuntu e Bash, installa il sottosistema Windows per Linux.

- Il tutorial richiede un ambiente locale con il comando `dig` DNS lookup utility.

Fase 1: Creare un namespace AWS Cloud Map

In questo passaggio, crei uno spazio dei nomi pubblico AWS Cloud Map . AWS Cloud Map crea una zona ospitata sulla Route 53 per tuo conto con lo stesso nome. Questo ti dà la possibilità di scoprire le istanze di servizio create in questo spazio dei nomi utilizzando record DNS pubblici o utilizzando chiamate API. AWS Cloud Map

1. Accedi a AWS Management Console e apri la console all'indirizzo. AWS Cloud Map https://console.aws.amazon.com/cloudmap/
2. Selezionare Create namespace (Crea spazio dei nomi).
3. Per il nome dello spazio dei nomi, specificare. `cloudmap-tutorial.com`

   > **Note**
   >
   > Se intendi utilizzarlo in produzione, assicurati di aver specificato il nome di un dominio di tua proprietà o a cui hai avuto accesso. Ma ai fini di questo tutorial, non è necessario che si tratti di un dominio effettivo che viene utilizzato.

4. (Facoltativo) Per la descrizione dello spazio dei nomi, specificate una descrizione per ciò per cui intendete utilizzare lo spazio dei nomi.
5. Per Instance Discovery, seleziona le chiamate API e le query DNS pubbliche.
6. Lascia il resto dei valori predefiniti e scegli Crea namespace.

Fase 2: Creare i servizi AWS Cloud Map

In questo passaggio, si creano due servizi. Il primo servizio sarà individuabile utilizzando chiamate DNS e API pubbliche. Il secondo servizio sarà individuabile solo tramite chiamate API.

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Nel riquadro di navigazione a sinistra, scegli Namespace per elencare i namespace che hai creato.
3. Dall'elenco dei namespace, seleziona lo spazio dei nomi e scegli Visualizza dettagli. **`cloudmap-tutorial.com`**
4. Nella sezione Servizi, scegli Crea servizio ed esegui le seguenti operazioni per creare il primo servizio.

a. Per Nome servizio, inserisci `public-service`. Il nome del servizio verrà applicato ai record DNS AWS Cloud Map creati. Il formato utilizzato è*<service-name>*.*<namespace-name>*.

b. Per Service Discovery Configuration, seleziona API e DNS.

c. Nella sezione Configurazione DNS, per Politica di routing, seleziona Routing di risposte multivalore.

> **Note**
>
> La console lo tradurrà in MULTIVALUE dopo averlo selezionato. Per ulteriori informazioni sulle opzioni di routing disponibili, vedere Choose a routing policy nella Route 53 Developer Guide.

d. Lascia il resto dei valori predefiniti e scegli Crea servizio che ti riporterà alla pagina dei dettagli del namespace.

5. Nella sezione Servizi, scegli Crea servizio ed esegui le seguenti operazioni per creare il secondo servizio.

a. Per Nome servizio, inserisci `backend-service`.

b. Per Service Discovery Configuration, seleziona Solo API.

c. Lascia il resto dei valori predefiniti e scegli Crea servizio.

Fase 3: Registrare le istanze del AWS Cloud Map servizio

In questo passaggio, crei due istanze di servizio, una per ogni servizio nel nostro namespace.

1. Accedi a AWS Management Console e apri la console all' AWS Cloud Map indirizzo. https://console.aws.amazon.com/cloudmap/
2. Dall'elenco dei namespace, seleziona lo spazio dei nomi creato nel passaggio 1 e scegli Visualizza dettagli.
3. Nella pagina dei dettagli del namespace, dall'elenco dei servizi, seleziona il servizio e scegli Visualizza dettagli. **`public-service`**
4. Nella sezione Istanze di servizio, scegli Registra istanza di servizio ed esegui le seguenti operazioni per creare la prima istanza di servizio.

   a. Per ID dell'istanza di servizio, specificare`first`.
   b. Per IPv4 l'indirizzo, specificare`192.168.2.1`.
   c. Lascia il resto dei valori predefiniti e scegli Register service instance.
5. Utilizzando il breadcrumb nella parte superiore della pagina, seleziona cloudmap-tutorial.com per tornare alla pagina di dettaglio del namespace.
6. Nella pagina dei dettagli del namespace, dall'elenco dei servizi, seleziona il servizio di backend e scegli Visualizza dettagli.
7. Nella sezione Istanze di servizio, scegli Registra istanza di servizio ed esegui le seguenti operazioni per creare la seconda istanza di servizio.

   a. Per ID dell'istanza di servizio, specifica `second` di indicare che si tratta della seconda istanza del servizio.
   b. Per Tipo di istanza, seleziona Informazioni di identificazione per un'altra risorsa.
   c. Per gli attributi personalizzati, aggiungi una coppia chiave-valore con `service-name` come chiave e `backend` come valore.
   d. Selezionare Register service instance (Registra istanza del servizio).

Fase 4: Scopri le istanze del servizio AWS Cloud Map

Ora che lo spazio dei AWS Cloud Map nomi, i servizi e le istanze del servizio sono stati creati, puoi verificare che tutto funzioni scoprendo le istanze. Utilizza il `dig` comando per verificare le impostazioni DNS pubbliche e l' AWS Cloud Map API per verificare il servizio di backend. Per ulteriori informazioni sul `dig` comando, vedere dig - DNS lookup utility.

1. Accedi AWS Management Console e apri la console Route 53 all'indirizzo. https://console.aws.amazon.com/route53/
2. Nel riquadro di navigazione a sinistra, scegliere Hosted zones (Zone ospitate).
3. Seleziona la zona ospitata da cloudmap-tutorial.com. Questo visualizza i dettagli della zona ospitata in un riquadro separato. Prendi nota dei name server associati alla tua zona ospitata poiché li useremo nel passaggio successivo.
4. Utilizzando il comando dig e uno dei name server Route 53 per la tua zona ospitata, interroga i record DNS per la tua istanza di servizio.

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

`ANSWER SECTION`Nell'output dovrebbe essere visualizzato l' IPv4 indirizzo associato al `public-service` servizio.

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

5. Utilizzando AWS CLI, interroga gli attributi per le seconde istanze del servizio.

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

L'output mostra gli attributi associati al servizio come coppie chiave-valore.

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

Fase 5: Pulisci le risorse

Una volta completato il tutorial, puoi eliminare le risorse. AWS Cloud Map richiede di ripulirle in ordine inverso, prima le istanze del servizio, poi i servizi e infine il namespace. AWS Cloud Map ripulirà le risorse della Route 53 per tuo conto durante questi passaggi.

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Dall'elenco dei namespace, seleziona lo spazio dei **`cloudmap-tutorial.com`** nomi e scegli Visualizza dettagli.

3. Nella pagina dei dettagli del namespace, dall'elenco dei servizi, seleziona il servizio e scegli Visualizza dettagli. **`public-service`**
4. Nella sezione Istanze di servizio, seleziona l'**`first`**istanza e scegli Annulla registrazione.
5. Utilizzando il breadcrumb nella parte superiore della pagina, seleziona cloudmap-tutorial.com per tornare alla pagina di dettaglio del namespace.
6. Nella pagina dei dettagli del namespace, dall'elenco dei servizi, seleziona il servizio di servizio pubblico e scegli Elimina.
7. Ripeti i passaggi 3-6 per. `backend-service`
8. Nella barra di navigazione a sinistra, scegli Namespace.
9. Seleziona lo **`cloudmap-tutorial.com`** spazio dei nomi e scegli Elimina.

> **Note**
>
> Sebbene AWS Cloud Map pulisca le risorse di Route 53 per tuo conto, puoi accedere alla console Route 53 per verificare che la zona `cloudmap-tutorial.com` ospitata venga eliminata.

Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con le query DNS e le chiamate API utilizzando AWS CLI

Questo tutorial dimostra come utilizzare il rilevamento dei AWS Cloud Map servizi utilizzando AWS Command Line Interface (CLI). Creerai un'architettura di microservizi con due servizi di backend: uno individuabile tramite query DNS e l'altro individuabile solo utilizzando l'API. AWS Cloud Map

Per un tutorial che include i passaggi della console, consulta. AWS Cloud Map Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con le query DNS e le chiamate API

Prerequisiti

I seguenti prerequisiti devono essere soddisfatti per completare correttamente il tutorial.

- Prima di iniziare, completa i passaggi descritti in Configurazione per l'uso AWS Cloud Map.
- Se non l'hai ancora installato AWS Command Line Interface, segui i passaggi indicati in Installazione o aggiornamento della versione più recente di AWS CLI per installarlo.

Per eseguire i comandi nel tutorial, sono necessari un terminale a riga di comando o una shell (interprete di comandi). In Linux e macOS, utilizza la shell (interprete di comandi) e il gestore pacchetti preferiti.

> **Note**
>
> Su Windows, alcuni comandi della CLI Bash utilizzati comunemente con Lambda (ad esempio, `zip`) non sono supportati dai terminali integrati del sistema operativo. Per ottenere una versione integrata su Windows di Ubuntu e Bash, installa il sottosistema Windows per Linux.

- Il tutorial richiede un ambiente locale con il comando `dig` DNS lookup utility.

Crea un namespace AWS Cloud Map

Per prima cosa, creerai uno spazio dei nomi pubblico AWS Cloud Map . AWS Cloud Map creerà una zona ospitata su Route 53 con lo stesso nome, abilitando l'individuazione dei servizi tramite record DNS e chiamate API.

1. Crea lo spazio dei nomi DNS pubblico:

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

   Il comando restituisce un ID operativo che è possibile utilizzare per verificare lo stato della creazione del namespace:

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

2. Controllate lo stato dell'operazione per confermare che il namespace è stato creato correttamente:

```
aws servicediscovery get-operation \
    --operation-id gv4g5meo7ndmeh4fqskygvk23d2fijwa-k9xmplyzd \
```

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

3. Una volta completata l'operazione, ottieni l'ID del namespace:

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

Questo comando restituisce l'ID dello spazio dei nomi, che ti servirà per i passaggi successivi:

```
ns-abcd1234xmplefgh
```

Crea i servizi AWS Cloud Map

Ora, crea due servizi all'interno del tuo namespace. Il primo servizio sarà individuabile utilizzando chiamate DNS e API, mentre il secondo sarà individuabile solo tramite chiamate API.

1. Crea il primo servizio con DNS discovery abilitato:

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

Il comando restituisce i dettagli sul servizio creato:

```
{
    "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. Crea il secondo servizio con il rilevamento solo tramite API:

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

Il comando restituisce i dettagli sul servizio creato:

```
{
    "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"
    }
}
```

Registra le istanze AWS Cloud Map del servizio

Successivamente, registra le istanze di servizio per ciascuno dei tuoi servizi. Queste istanze rappresentano le risorse effettive che verranno scoperte.

1. Registra la prima istanza con un IPv4 indirizzo per il rilevamento DNS:

```
aws servicediscovery register-instance \
```

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

Il comando restituisce un ID operativo:

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

2. Controlla lo stato dell'operazione per confermare che l'istanza è stata registrata correttamente:

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

3. Registra la seconda istanza con attributi personalizzati per l'individuazione delle API:

```
aws servicediscovery register-instance \
    --service-id srv-ijkl5678xmplmnop \
    --instance-id second \
    --attributes service-name=backend \
    --region us-east-2
```

Il comando restituisce un ID operativo:

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

4. Controlla lo stato dell'operazione per confermare che l'istanza è stata registrata correttamente:

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

Scopri le istanze AWS Cloud Map del servizio

Ora che hai creato e registrato le tue istanze di servizio, puoi verificare che tutto funzioni scoprendole utilizzando sia le query DNS che l'API. AWS Cloud Map

1. Per prima cosa, ottieni l'ID della zona ospitata di Route 53:

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

   Questo restituisce l'ID della zona ospitata:

   ```
   /hostedzone/Z1234ABCDXMPLEFGH
   ```

2. Ottieni i name server per la tua zona ospitata:

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

   Questo restituisce uno dei name server:

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

3. Usa il `dig` comando per interrogare i record DNS per il tuo servizio pubblico:

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

   L'output dovrebbe mostrare l' IPv4 indirizzo associato al servizio:

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

4. Usa il AWS CLI per scoprire l'istanza del servizio di backend:

   ```
   aws servicediscovery discover-instances \
       --namespace-name cloudmap-tutorial.com \
   ```

```
    --service-name backend-service \
    --region us-east-2
```

L'output mostra gli attributi associati al servizio:

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

Pulisci le risorse

Una volta completato il tutorial, ripulisci le risorse per evitare di incorrere in addebiti. AWS Cloud Map richiede di ripulirle in ordine inverso: prima le istanze del servizio, poi i servizi e infine lo spazio dei nomi.

1. Annulla la registrazione della prima istanza del servizio:

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

2. Annulla la registrazione della seconda istanza del servizio:

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

3. Eliminare il servizio pubblico:

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

4. Elimina il servizio di backend:

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

5. Elimina lo spazio dei nomi :

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

6. Verifica che la zona ospitata da Route 53 sia stata eliminata:

```
aws route53 list-hosted-zones-by-name \
    --dns-name cloudmap-tutorial.com
```

Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati

Il seguente tutorial mostra come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati individuabili tramite l' AWS Cloud Map API. Il tutorial illustra la creazione e l'esecuzione di applicazioni client utilizzando. AWS CloudShell Le applicazioni utilizzano due funzioni Lambda per scrivere dati in una tabella DynamoDB e quindi leggerli dalla tabella. Le funzioni Lambda e la tabella DynamoDB sono registrate come istanze di servizio. AWS Cloud Map Il codice nelle applicazioni client e nelle funzioni Lambda utilizza attributi AWS Cloud Map personalizzati per individuare le risorse necessarie per eseguire il lavoro.

Per una versione AWS CLI basata di questo tutorial, consultaScopri come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati utilizzando AWS CLI.

> **Important**
>
> Durante il workshop creerai AWS risorse che comporteranno un costo nel tuo AWS account. Si consiglia di ripulire le risorse non appena si finisce il workshop per ridurre al minimo i costi.

Prerequisiti

Prima di iniziare, completa i passaggi descritti in Configurazione per l'uso AWS Cloud Map.

Fase 1: Creare un AWS Cloud Map namespace

In questo passaggio, crei un AWS Cloud Map namespace. Un namespace è un costrutto utilizzato per raggruppare i servizi per un'applicazione. Quando si crea lo spazio dei nomi, si specifica in che modo le risorse saranno individuabili. Le risorse create nello spazio dei nomi creato in questo passaggio saranno rilevabili con AWS Cloud Map chiamate API che utilizzano attributi personalizzati.

1. Accedi a AWS Management Console e apri la console all' AWS Cloud Map indirizzo. https://console.aws.amazon.com/cloudmap/
2. Selezionare Create namespace (Crea spazio dei nomi).
3. Per il nome dello spazio dei nomi, specificare. `cloudmap-tutorial`
4. (Facoltativo) Per la descrizione dello spazio dei nomi, specificate una descrizione per il quale intendete utilizzare lo spazio dei nomi.
5. Per Instance Discovery, seleziona Chiamate API.
6. Lascia il resto dei valori predefiniti e scegli Crea namespace.

Fase 2: Creare una tabella DynamoDB

In questo passaggio, si crea una tabella DynamoDB. La tabella viene utilizzata per archiviare e recuperare i dati per l'applicazione di esempio che verrà creata nei passaggi seguenti.

Per informazioni su come creare un DynamoDB, vedere Passaggio 1: Creare una tabella in DynamoDB nella DynamoDB Developer Guide e utilizzare la tabella seguente per determinare quali opzioni specificare.

Opzione	Valore	
Nome tabella	mappa del cloud	
Chiave di partizione	id	

Mantieni i valori predefiniti per il resto delle impostazioni e crea la tabella.

Fase 3: Creare un servizio AWS Cloud Map dati e registrare la tabella DynamoDB come istanza

In questo passaggio, si crea un AWS Cloud Map servizio e quindi si registra la tabella DynamoDB creata nell'ultimo passaggio come istanza del servizio.

1. Apri la console all'indirizzo AWS Cloud Map https://console.aws.amazon.com/cloudmap/
2. Dall'elenco dei namespace, seleziona lo spazio dei **`cloudmap-tutorial`** nomi e scegli Visualizza dettagli.
3. Nella sezione Servizi, scegli Crea servizio ed esegui le seguenti operazioni.
   a. Per Nome servizio, inserisci `data-service`.
   b. Lascia il resto dei valori predefiniti e scegli Crea servizio.
4. Nella sezione Servizi, seleziona il `data-service` servizio e scegli Visualizza dettagli.
5. Nella sezione Istanze di servizio, scegli Registra istanza di servizio.
6. Nella pagina Registra istanza del servizio, procedi come segue.
   a. Per Tipo di istanza, seleziona Informazioni di identificazione per un'altra risorsa.
   b. Per ID dell'istanza del servizio, specificare`data-instance`.
   c. Nella sezione Attributi personalizzati, specifica la seguente coppia chiave-valore: key `=tablename`, value =. `cloudmap`

Fase 4: Creare un ruolo di esecuzione AWS Lambda

In questo passaggio, crei un ruolo IAM utilizzato dalla AWS Lambda funzione nella fase successiva. Puoi assegnare un nome al ruolo IAM `cloudmap-tutorial-role` e omettere il limite delle autorizzazioni perché il ruolo viene utilizzato solo per questo tutorial e puoi eliminarlo in seguito.

Per creare il ruolo di servizio per Lambda (console IAM)

1. Accedi AWS Management Console e apri la console IAM all'indirizzo https://console.aws.amazon.com/iam/.
2. Nel pannello di navigazione della console IAM, scegliere Ruoli e quindi Crea ruolo.
3. Per Trusted entity type (Tipo di entità attendibile), scegli Servizio AWS.
4. Per Servizio o caso d'uso, scegli Lambda, quindi scegli lo use case Lambda.
5. Scegli Next (Successivo).
6. Cerca e seleziona la casella accanto alla **`PowerUserAccess`** policy, quindi scegli Avanti.
7. Scegli Next (Successivo).
8. Per Nome del ruolo, specificare`cloudmap-tutorial-role`.
9. Verificare il ruolo e quindi scegliere Create role (Crea ruolo).

Fase 5: Creare la funzione Lambda per scrivere dati

In questo passaggio, crei una funzione Lambda creata da zero che scrive dati nella tabella DynamoDB utilizzando l' AWS Cloud Map API per interrogare il servizio creato. AWS Cloud Map

Per informazioni sulla creazione di una funzione Lambda, consulta Creare una funzione Lambda con la console nella Guida per gli AWS Lambda sviluppatori e usa la tabella seguente per determinare quali opzioni specificare o scegliere.

Opzione	Valore
Nome funzione	funzione di scrittura
Runtime	Python 3.12
Architettura	x86_64
Autorizzazioni	Usa un ruolo esistente
Ruolo esistente	cloudmap-tutorial-role

Dopo aver creato la funzione, aggiorna il codice di esempio in modo che rifletta il seguente codice Python, quindi distribuisci la funzione. Tieni presente che stai specificando l'attributo `datatable`

personalizzato che hai associato all'istanza del AWS Cloud Map servizio che hai creato per la tabella DynamoDB. La funzione genera una chiave che è un numero casuale compreso tra 1 e 100 e la associa a un valore che viene passato alla funzione quando viene chiamata.

```
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)
    }
```

Dopo aver distribuito la funzione, per evitare errori di timeout, aggiorna il timeout della funzione a 5 secondi. Per ulteriori informazioni, consulta Configura il timeout della funzione Lambda nella Guida per gli AWS Lambda sviluppatori.

Passaggio 6: creare un servizio AWS Cloud Map app e registrare la funzione di scrittura Lambda come istanza

In questo passaggio, crei un AWS Cloud Map servizio e quindi registri la funzione di scrittura Lambda come istanza del servizio.

1. Apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/

2. Nella barra di navigazione a sinistra, scegli Namespace.
3. Dall'elenco dei namespace, seleziona lo spazio dei nomi e scegli Visualizza dettagli. **`cloudmap-tutorial`**
4. Nella sezione Servizi, scegli Crea servizio ed esegui le seguenti operazioni.

   a. Per Nome servizio, inserisci `app-service`.

   b. Lascia il resto dei valori predefiniti e scegli Crea servizio.
5. Nella sezione Servizi, seleziona il `app-service` servizio e scegli Visualizza dettagli.
6. Nella sezione Istanze di servizio, scegli Registra istanza di servizio.
7. Nella pagina Registra istanza del servizio, procedi come segue.

   a. Per Tipo di istanza, seleziona Informazioni di identificazione per un'altra risorsa.

   b. Per ID dell'istanza del servizio, specificare`write-instance`.

   c. Nella sezione Attributi personalizzati, specificate le seguenti coppie chiave-valore.

      - chiave =**`action`**, valore = `write`
      - chiave =`functionname`, valore = `writefunction`

Fase 7: Creare la funzione Lambda per leggere i dati

In questo passaggio, crei una funzione Lambda creata da zero che scrive dati nella tabella DynamoDB che hai creato.

Per informazioni sulla creazione di una funzione Lambda, consulta Creare una funzione Lambda con la console nella Guida per gli AWS Lambda sviluppatori e usa la tabella seguente per determinare quali opzioni specificare o scegliere.

Opzione	Valore
Nome funzione	funzione di lettura
Runtime	Python 3.12
Architettura	x86_64
Autorizzazioni	Usa un ruolo esistente

Opzione	Valore	
Ruolo esistente	cloudmap-tutorial-role	

Dopo aver creato la funzione, aggiorna il codice di esempio in modo che rifletta il seguente codice Python, quindi distribuisci la funzione. La funzione analizza la tabella e restituisce tutti gli elementi.

```
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)
    }
```

Dopo aver distribuito la funzione, per evitare errori di timeout, aggiorna il timeout della funzione a 5 secondi. Per ulteriori informazioni, consulta Configura il timeout della funzione Lambda nella Guida per gli AWS Lambda sviluppatori.

Fase 8: Registrare la funzione di lettura Lambda come istanza di servizio AWS Cloud Map

In questo passaggio, si registra la funzione di lettura Lambda come istanza di servizio nel `app-service` servizio creato in precedenza.

1. Apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/

2. Nella barra di navigazione a sinistra, scegli Namespace.
3. Dall'elenco dei namespace, seleziona lo spazio dei nomi e scegli Visualizza dettagli. **`cloudmap-tutorial`**
4. Nella sezione Servizi, seleziona il **`app-service`** servizio e scegli Visualizza dettagli.
5. Nella sezione Istanze di servizio, scegli Registra istanza di servizio.
6. Nella pagina Registra istanza del servizio, procedi come segue.

   a. Per Tipo di istanza, seleziona Informazioni di identificazione per un'altra risorsa.

   b. Per ID dell'istanza del servizio, specificare`read-instance`.

   c. Nella sezione Attributi personalizzati, specificate le seguenti coppie chiave-valore.

      - chiave =**`action`**, valore = `read`
      - chiave =`functionname`, valore = `readfunction`

Fase 9: Creare ed eseguire client di lettura e scrittura su AWS CloudShell

È possibile creare ed eseguire applicazioni client AWS CloudShell che utilizzano il codice per scoprire i servizi in cui sono stati configurati AWS Cloud Map ed effettuare chiamate a tali servizi.

1. Apri la AWS CloudShell console all'indirizzo https://console.aws.amazon.com/cloudshell/
2. Utilizzate il seguente comando per creare un file chiamato`writefunction.py`.

   ```
   vim writeclient.py
   ```

3. Nel `writeclient.py` file, accedete alla modalità di inserimento premendo il `i` pulsante. Quindi, copia e incolla il seguente codice. Questo codice rileva la funzione Lambda per scrivere dati cercando l'`name=writeservice`attributo personalizzato nel `app-service` servizio. Viene restituito il nome della funzione Lambda responsabile della scrittura dei dati nella tabella DynamoDB. Quindi viene richiamata la funzione Lambda, passando un payload di esempio che viene scritto nella tabella come valore.

   ```
   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. Premi il tasto Esc`:wq`, digita e premi il tasto invio per salvare il file e uscire.
5. Usa il seguente comando per eseguire il codice Python.

```
python3 writeclient.py
```

L'output dovrebbe essere una `200` risposta, simile alla seguente.

```
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. Per verificare che la scrittura sia avvenuta correttamente nel passaggio precedente, crea un client di lettura.
   a. Utilizzate il seguente comando per creare un file chiamato`readfunction.py`.

   ```
   vim readclient.py
   ```

   b. Nel `readclient.py` file, premi il `i` pulsante per accedere alla modalità di inserimento. Quindi, copia e incolla il seguente codice. Questo codice analizza la tabella e restituirà il valore che hai scritto nella tabella nel passaggio precedente.

   ```
   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. Premi il tasto Esc`:wq`, digita e premi il tasto invio per salvare il file e uscire.

d. Usa il seguente comando per eseguire il codice Python.

```
python3 readclient.py
```

L'output dovrebbe essere simile al seguente, che elenca il valore scritto nella tabella eseguendo `writefunction.py` e la chiave casuale generata nella funzione di scrittura 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}}"}'
```

Fase 10: Pulire le risorse

Dopo aver completato il tutorial, elimina le risorse per evitare di incorrere in costi aggiuntivi. AWS Cloud Map richiede di ripulirle in ordine inverso, prima le istanze del servizio, poi i servizi e infine il namespace. I passaggi seguenti illustrano la pulizia AWS Cloud Map delle risorse utilizzate nel tutorial.

Per eliminare le AWS Cloud Map risorse

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Dall'elenco dei namespace, seleziona lo spazio dei **`cloudmap-tutorial`** nomi e scegli Visualizza dettagli.
3. Nella pagina dei dettagli del namespace, dall'elenco dei servizi, seleziona il servizio e scegli Visualizza dettagli. **`data-service`**
4. Nella sezione Istanze di servizio, seleziona l'**`data-instance`**istanza e scegli Annulla registrazione.
5. Utilizzando il breadcrumb nella parte superiore della pagina, seleziona cloudmap-tutorial.com per tornare alla pagina di dettaglio del namespace.
6. Nella pagina dei dettagli del namespace, dall'elenco dei servizi, seleziona il servizio data-service e scegli Elimina.
7. Ripeti i passaggi 3-6 per il `app-service` servizio e le istanze del servizio. `write-instance` `read-instance`
8. Nella barra di navigazione a sinistra, scegli Namespace.
9. Seleziona lo **`cloudmap-tutorial`** spazio dei nomi e scegli Elimina.

La tabella seguente elenca le procedure che è possibile seguire per eliminare le altre risorse utilizzate nel tutorial.

Risorsa	Fasi	
DynamoDB tabella	Fase 6: (Facoltativo) Eliminare la tabella DynamoDB per ripulire le risorse nell'Amazon DynamoDB Developer Guide	
Funzioni Lambda e ruolo di esecuzione IAM associato	Esegui la pulizia nella Guida per gli AWS Lambda sviluppatori	

Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati utilizzando AWS CLI

Questo tutorial dimostra come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati. Creerai un'applicazione di microservizi che consente di AWS Cloud Map scoprire le risorse in modo dinamico utilizzando attributi personalizzati. L'applicazione è composta da due funzioni Lambda che scrivono e leggono dati in una tabella DynamoDB, con tutte le risorse registrate. AWS Cloud Map

Per una AWS Management Console versione del tutorial, vedi. Scopri come utilizzare il rilevamento dei AWS Cloud Map servizi con attributi personalizzati

Prerequisiti

Prima di iniziare questo tutorial, completa i passaggi indicati inConfigurazione per l'uso AWS Cloud Map.

Crea un AWS Cloud Map namespace

Un namespace è un costrutto utilizzato per raggruppare i servizi per un'applicazione. In questo passaggio, creerai uno spazio dei nomi che consente di individuare le risorse tramite chiamate API. AWS Cloud Map

1. Esegui il comando seguente per creare uno spazio dei nomi HTTP:

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

   Il comando restituisce un ID di operazione. È possibile controllare lo stato dell'operazione con il seguente comando:

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

2. Una volta creato lo spazio dei nomi, è possibile recuperarne l'ID per utilizzarlo nei comandi successivi:

   ```
   aws servicediscovery list-namespaces \
   ```

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

3. Memorizza l'ID dello spazio dei nomi in una variabile per un uso successivo:

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

Creazione di una tabella DynamoDB

Quindi, crea una tabella DynamoDB che memorizzerà i dati per la tua applicazione:

1. Esegui il comando seguente per creare la tabella:

```
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. Attendi che la tabella diventi attiva prima di procedere:

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

   Questo comando attende che la tabella sia completamente creata e pronta per l'uso.

Creare un servizio AWS Cloud Map dati e registrare la tabella DynamoDB

Ora, crea un servizio nel tuo namespace per rappresentare le risorse di archiviazione dei dati:

1. Esegui il comando seguente per creare un AWS Cloud Map servizio per le risorse di archiviazione dei dati:

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

2. Ottieni l'ID del servizio per il servizio dati:

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

3. Registra la tabella DynamoDB come istanza di servizio con un attributo personalizzato che specifica il nome della tabella:

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

L'attributo personalizzato `tablename=cloudmap` consente ad altri servizi di scoprire il nome della tabella DynamoDB in modo dinamico.

Creare un ruolo IAM per le funzioni Lambda

Crea un ruolo IAM che le funzioni Lambda utilizzeranno per accedere AWS alle risorse:

1. Crea il documento sulla politica di fiducia per il ruolo IAM:

```
cat > lambda-trust-policy.json << EOF
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "lambda.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
EOF
```

2. Esegui il comando seguente per creare il ruolo IAM utilizzando la policy di fiducia:

```
aws iam create-role \
  --role-name cloudmap-tutorial-role \
```

```
  --assume-role-policy-document file://lambda-trust-policy.json
```

3. Crea un file per una policy IAM personalizzata con i permessi minimi:

```
cat > cloudmap-policy.json << EOF
{
  "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"
    }
  ]
}
EOF
```

4. Crea e collega la policy al ruolo 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
```

Crea la funzione Lambda per scrivere dati

Per creare una funzione Lambda che scrive dati nella tabella DynamoDB, segui questi passaggi:

1. Crea il file Python per la funzione di scrittura:

```
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
```

Questa funzione utilizza AWS Cloud Map per scoprire il nome della tabella DynamoDB dall'attributo personalizzato, quindi scrive i dati nella tabella.

2. Package e distribuzione della funzione 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. Aggiorna il timeout della funzione per evitare errori di timeout:

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

Crea un servizio AWS Cloud Map app e registra la funzione di scrittura Lambda

Per creare un altro servizio nel tuo spazio dei nomi per rappresentare le funzioni dell'applicazione, procedi nel seguente modo:

1. Crea un servizio per le funzioni dell'applicazione:

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

2. Ottieni l'ID del servizio per il servizio dell'app:

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

3. Registra la funzione di scrittura Lambda come istanza di servizio con attributi personalizzati:

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

   Gli attributi personalizzati `functionname=writefunction` consentono `action=write` ai client di scoprire questa funzione in base al suo scopo.

Crea la funzione Lambda per leggere i dati

Per creare una funzione Lambda che legge i dati dalla tabella DynamoDB, segui questi passaggi:

1. Crea il file Python per la funzione read:

```
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
```

Questa funzione utilizza anche AWS Cloud Map per scoprire il nome della tabella DynamoDB, quindi legge i dati dalla tabella. Include la gestione degli errori e i commenti di impaginazione.

2. Package e distribuzione della funzione 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. Aggiorna il timeout della funzione:

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

Registra la funzione di lettura Lambda come istanza di servizio

Per registrare la funzione di lettura Lambda come un'altra istanza di servizio nel servizio app, procedi nel seguente passaggio:

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

Gli attributi personalizzati `functionname=readfunction` consentono `action=read` ai client di scoprire questa funzione in base al suo scopo.

Crea ed esegui applicazioni client

Per creare un'applicazione client Python da utilizzare AWS Cloud Map per scoprire e richiamare la funzione di scrittura, segui questi passaggi:

1. Crea un file Python per l'applicazione client di scrittura:

```
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
```

Questo client utilizza l'`QueryParameters`opzione per trovare istanze di servizio con l'`action=write`attributo.

2. Crea un file Python per l'applicazione client di lettura:

```
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. Esegui il client di scrittura per aggiungere dati alla tabella DynamoDB:

```
python3 writeclient.py
```

L'output dovrebbe mostrare una risposta corretta con il codice di stato HTTP 200.

4. Esegui il client di lettura per recuperare i dati dalla tabella DynamoDB:

```
python3 readclient.py
```

L'output dovrebbe mostrare i dati che sono stati scritti nella tabella, incluso l'ID generato casualmente e il valore «This is a test data».

Pulizia delle risorse

Al termine del tutorial, ripulisci le risorse per evitare di incorrere in costi aggiuntivi.

1. Innanzitutto, esegui il comando seguente per annullare la registrazione delle istanze del servizio:

```
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. Eseguite il comando seguente per eliminare i servizi:

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

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

3. Eseguite il comando seguente per eliminare lo spazio dei nomi:

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

4. Esegui il comando seguente per eliminare le funzioni Lambda:

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

5. Esegui il comando seguente per eliminare il ruolo e la policy 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. Esegui il comando seguente per eliminare la tabella DynamoDB:

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

7. Esegui il comando seguente per pulire i file temporanei:

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

AWS Cloud Map namespace

Un namespace è un'entità logica utilizzata per raggruppare AWS Cloud Map i servizi di un'applicazione con un nome e un livello di individuabilità comuni. Quando si crea uno spazio dei nomi, si specifica quanto segue:

- Un nome che desiderate venga utilizzato dall'applicazione per individuare le istanze.
- Il metodo con cui è possibile scoprire le istanze di servizio con cui ci AWS Cloud Map si registra. Puoi decidere se le tue risorse devono essere scoperte pubblicamente su Internet, privatamente in uno specifico cloud privato virtuale (VPC) o solo tramite chiamate API.

Di seguito sono riportati concetti generali sui namespace.

- I namespace sono specifici del tipo in cui vengono creati. Regione AWS Per utilizzarli AWS Cloud Map in più aree, è necessario creare namespace in ciascuna regione.
- Se crei uno spazio dei nomi per consentire, ad esempio, il rilevamento tramite query DNS in un VPC, crea AWS Cloud Map automaticamente una zona ospitata privata sulla Route 53. Questa zona ospitata può essere associata a più zone. VPCs Per ulteriori informazioni, consulta Associate VPCWith HostedZone nel riferimento alle API di Amazon Route 53.

Argomenti

Creazione di un AWS Cloud Map namespace per raggruppare i servizi applicativi

Puoi creare uno spazio dei nomi per raggruppare i servizi per la tua applicazione con un nome intuitivo che consenta l'individuazione delle risorse dell'applicazione tramite chiamate API o query DNS.

Opzioni di individuazione delle istanze

La tabella seguente riassume le diverse opzioni di individuazione delle istanze AWS Cloud Map e il tipo di namespace corrispondente che è possibile creare, a seconda dei servizi e della configurazione dell'applicazione.

Tipo di namespace	Metodo di individuazione delle istanze	Come funziona	Informazioni aggiuntive
HTTP	Chiamate API	Le risorse dell'applicazione possono scoprire altre risorse solo chiamando l'`DiscoverInstances` API.	• DiscoverInstances • CreateHttpNamespace
DNS privato	Chiamate API e query DNS in un VPC	Le risorse dell'applicazione possono scoprire altre risorse chiamando l'`DiscoverInstances` API e interrogando i nameserver nella zona ospitata privata di Route 53 che viene creata automaticamente. AWS Cloud Map La zona ospitata creata da AWS Cloud Map ha lo stesso nome dello spazio dei nomi e contiene record DNS con nomi nel formato.	• DiscoverInstances • CreatePrivateDnsNamespace

Tipo di namespace	Metodo di individuazione delle istanze	Come funziona	Informazioni aggiuntive
		service-name *namespace-name* . **Note** Route 53 Resolver risolve le query DNS che hanno origine nel VPC utilizzando i record nella zona ospitata privata. Se la zona ospitata privata non include un record che corrisponde al nome di dominio in una query DNS, Route 53 risponde alla query con (dominio inesistente). `NXDOMAIN`	

Tipo di namespace	Metodo di individuazione delle istanze	Come funziona	Informazioni aggiuntive
DNS pubblico	Chiamate API e query DNS pubbliche	Le risorse dell'applicazione possono scoprire altre risorse chiamando l'`DiscoverInstances` API e interrogando i nameserver nella zona ospitata pubblica di Route 53 che viene creata automaticamente. AWS Cloud Map La zona ospitata pubblica ha lo stesso nome dello spazio dei nomi e contiene record DNS con nomi nel formato. *`service-name namespace-name`* . **Note** Il nome del namespace in questo caso deve essere un nome di dominio che hai registrato.	• DiscoverInstances • CreatePublicDnsNamespace

Procedura

Puoi seguire questi passaggi per creare uno spazio dei nomi utilizzando AWS CLI AWS Management Console, o l'SDK per Python.

AWS Management Console

1. Accedi a AWS Management Console e apri la console all'indirizzo. AWS Cloud Map https://console.aws.amazon.com/cloudmap/
2. Selezionare Create namespace (Crea spazio dei nomi).
3. Per il nome dello spazio dei nomi, inserisci un nome che verrà utilizzato per scoprire le istanze.

   > **Note**
   >
   > - I namespace configurati per le query DNS pubbliche devono terminare con un dominio di primo livello. Ad esempio `.com`.
   > - È possibile specificare un nome di dominio internazionalizzato (IDN) convertendo prima il nome in Punycode. Per informazioni sui convertitori online, cerca su internet "convertitore punycode".
   >
   >   È possibile anche convertire un nome di dominio internazionalizzato in Punycode quando si creano spazi dei nomi in modo programmatico. Ad esempio, se stai utilizzando Java, puoi convertire un valore Unicode in Punycode utilizzando il metodo `toASCII` della libreria java.net.IDN.

4. (Facoltativo) Per la descrizione dello spazio dei nomi, immettete le informazioni sullo spazio dei nomi che saranno visibili nella pagina dei namespace e nella sezione Informazioni sullo spazio dei nomi. È possibile utilizzare queste informazioni per identificare facilmente un namespace.
5. Per Instance Discovery, puoi scegliere tra chiamate API, chiamate API e query DNS in VPCs, e chiamate API e query DNS pubbliche per creare rispettivamente uno spazio dei nomi HTTP, DNS privato o DNS pubblico. Per ulteriori informazioni, consulta Opzioni di individuazione delle istanze.

   In base alla selezione, segui questi passaggi.

- Se scegli chiamate API e query DNS in VPCs, per VPC, scegli un cloud privato virtuale (VPC) a cui desideri associare lo spazio dei nomi.
- Se scegli chiamate API e query DNS in VPCs o chiamate API e query DNS pubbliche, per TTL, specifica un valore numerico in secondi. Il valore time to live (TTL) determina per quanto tempo i resolver DNS memorizzano nella cache le informazioni per il record DNS di inizio dell'autorità (SOA) della zona ospitata da Route 53 creata con il tuo spazio dei nomi. Per ulteriori informazioni su TTL, consulta TTL (secondi) nella Amazon Route 53 Developer Guide.

6. (Facoltativo) In Tag, scegli Aggiungi tag, quindi specifica una chiave e un valore per etichettare il tuo namespace. Puoi specificare uno o più tag da aggiungere al tuo namespace. I tag consentono di classificare le AWS risorse in modo da gestirle più facilmente. Per ulteriori informazioni, consulta Taggare le tue risorse AWS Cloud Map.
7. Selezionare Create namespace (Crea spazio dei nomi). È possibile visualizzare lo stato dell'operazione utilizzando ListOperations. Per ulteriori informazioni, consulta la ListOperationssezione AWS Cloud Map API Reference

AWS CLI

- Crea uno spazio dei nomi con il comando per il tipo di individuazione delle istanze che preferisci (sostituisci i *red* valori con i tuoi).
  - Crea uno spazio dei nomi HTTP utilizzando. `create-http-namespace` Le istanze di servizio registrate utilizzando uno spazio dei nomi HTTP possono essere scoperte utilizzando una `DiscoverInstances` richiesta, ma non possono essere scoperte utilizzando DNS.

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

  - Crea uno spazio dei nomi privato basato su DNS e visibile solo all'interno di uno specifico Amazon VPC utilizzando. `create-private-dns-namespace` Puoi scoprire le istanze registrate con uno spazio dei nomi DNS privato utilizzando una richiesta o utilizzando DNS `DiscoverInstances`

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

- Crea uno spazio dei nomi pubblico basato su DNS visibile su Internet utilizzando. `create-public-dns-namespace` Puoi individuare le istanze registrate con uno spazio dei nomi DNS pubblico tramite una richiesta `DiscoverInstances` o utilizzando il DNS.

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

AWS SDK for Python (Boto3)

1. Se non l'hai ancora `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo qui. `Boto3`
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Crea uno spazio dei nomi con il comando per il tipo di individuazione delle istanze che preferisci (sostituisci i *red* valori con i tuoi):

   - Crea uno spazio dei nomi HTTP utilizzando. `create_http_namespace()` Le istanze di servizio registrate utilizzando uno spazio dei nomi HTTP possono essere scoperte utilizzando`discover_instances()`, ma non tramite DNS.

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

   - Crea uno spazio dei nomi privato basato su DNS e visibile solo all'interno di uno specifico Amazon VPC utilizzando. `create_private_dns_namespace()` Puoi scoprire le istanze registrate con uno spazio dei nomi DNS privato utilizzando uno dei due o utilizzando DNS `discover_instances()`

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

```
print(response)
```

- Crea uno spazio dei nomi pubblico basato su DNS visibile su Internet utilizzando. `create_public_dns_namespace()` Puoi scoprire le istanze registrate con uno spazio dei nomi DNS pubblico utilizzando uno dei due o utilizzando il DNS. `discover_instances()`

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

- Esempio di output di risposta

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

Passaggi successivi

Dopo aver creato uno spazio dei nomi, è possibile creare servizi nello spazio dei nomi per raggruppare le risorse dell'applicazione che collettivamente servono a uno scopo particolare nell'applicazione. Un servizio funge da modello per la registrazione delle risorse dell'applicazione come istanze. Per ulteriori informazioni sulla creazione di AWS Cloud Map servizi, vedere. Creazione di un AWS Cloud Map servizio per un componente dell'applicazione

Elencare i AWS Cloud Map namespace

Dopo aver creato i namespace, puoi visualizzare un elenco dei namespace che hai creato seguendo questi passaggi.

AWS Management Console

1. Accedi a e apri la console all'indirizzo. AWS Management Console AWS Cloud Map https://console.aws.amazon.com/cloudmap/

2. Nel riquadro di navigazione, scegli Namespace per visualizzare un elenco di namespace. Puoi ordinare i namespace per nome, descrizione, modalità di scoperta delle istanze o ID dello spazio dei nomi. Puoi anche inserire il nome o l'ID di un namespace nel campo di ricerca per individuare e visualizzare uno spazio dei nomi specifico.

AWS CLI

- Elenca i namespace con il comando. `list-namespaces`

```
aws servicediscovery list-namespaces
```

AWS SDK for Python (Boto3)

1. Se non l'hai ancora `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo qui. `Boto3`
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Elenca i namespace con. `list_namespaces()`

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

Esempio di output di risposta

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

```
}
```

Eliminazione di un AWS Cloud Map namespace

Dopo aver finito di usare un namespace, puoi eliminarlo. Quando si elimina uno spazio dei nomi, non è più possibile utilizzarlo per registrare o individuare istanze dei servizi.

> **Note**
>
> Quando crei uno spazio dei nomi, se specifichi che desideri scoprire le istanze di servizio utilizzando query DNS pubbliche o query DNS in VPCs, AWS Cloud Map crea una zona ospitata pubblica o privata di Amazon Route 53. Quando elimini lo spazio dei nomi, elimina la zona ospitata corrispondente. AWS Cloud Map

Prima di eliminare uno spazio dei nomi, è necessario annullare la registrazione di tutte le istanze del servizio e quindi eliminare tutti i servizi che sono stati creati nello spazio dei nomi. Per ulteriori informazioni, consulta Annullamento della registrazione di un'istanza di servizio AWS Cloud Map e Eliminazione di un servizio AWS Cloud Map.

Dopo aver annullato la registrazione delle istanze ed eliminato i servizi creati in un namespace, segui questi passaggi per eliminare lo spazio dei nomi.

AWS Management Console

1. Accedi e apri la console all'indirizzo. AWS Management Console AWS Cloud Map https://console.aws.amazon.com/cloudmap/
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Seleziona lo spazio dei nomi che desideri eliminare, quindi scegli Elimina.
4. Conferma di voler eliminare il servizio selezionando nuovamente Elimina.

AWS CLI

- Elimina uno spazio dei nomi con il `delete-namespace` comando (sostituisci il *red* valore con il tuo). Se il namespace contiene ancora uno o più servizi, la richiesta ha esito negativo.

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

AWS SDK for Python (Boto3)

1. Se non l'hai ancora Boto3 installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo qui. Boto3
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Elimina uno spazio dei nomi con `delete_namespace()` (sostituisci il *red* valore con il tuo). Se il namespace contiene ancora uno o più servizi, la richiesta ha esito negativo.

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

Esempio di output di risposta

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

AWS Cloud Map servizi

Un AWS Cloud Map servizio è un modello per la registrazione delle istanze del servizio che comprende il nome del servizio e la configurazione DNS, se applicabile, del servizio. Puoi anche impostare un controllo dello stato di integrità per determinare lo stato di integrità delle istanze del servizio e filtrare le risorse non integre. Un servizio può rappresentare un componente dell'applicazione. Ad esempio, puoi creare un servizio per le risorse che gestiscono i pagamenti sull'applicazione e un altro per le risorse che gestiscono gli utenti.

Un servizio consente di localizzare le risorse per un'applicazione recuperando uno o più endpoint che possono essere utilizzati per connettersi alla risorsa. La localizzazione delle risorse viene effettuata utilizzando le query DNS o l'azione AWS Cloud Map `DiscoverInstances`API, a seconda di come è stato configurato lo spazio dei nomi. Puoi utilizzare la AWS Cloud Map console per definire l'ambito del rilevamento delle istanze a livello di servizio.

Puoi anche specificare metadati personalizzati come attributi a livello di servizio utilizzando l'API. `UpdateServiceAttributes` È possibile impostare gli attributi del servizio per evitare la duplicazione degli attributi tra le istanze e modificare questi attributi senza dover apportare modifiche agli attributi dell'istanza. Le informazioni che è possibile specificare come attributi a livello di servizio includono, a titolo esemplificativo ma non esaustivo, quanto segue:

- Pesi degli endpoint per lo spostamento del traffico durante le implementazioni progressive.
- Preferenze di servizio come i timeout delle API e le politiche suggerite per i nuovi tentativi.

Per ulteriori informazioni, consulta il riferimento UpdateServiceAttributesall'AWS Cloud Map API.

I seguenti argomenti descrivono il controllo dello stato di salute e le configurazioni DNS per i servizi e includono istruzioni per creare, elencare, aggiornare ed eliminare un servizio.

Argomenti

- AWS Cloud Map configurazione del controllo dello stato del servizio
- AWS Cloud Map configurazione DNS del servizio
- Creazione di un AWS Cloud Map servizio per un componente dell'applicazione
- Aggiornamento di un AWS Cloud Map servizio
- Elencare AWS Cloud Map i servizi in un namespace

- Eliminazione di un servizio AWS Cloud Map

AWS Cloud Map configurazione del controllo dello stato del servizio

I controlli dello stato aiutano a determinare se le istanze del servizio sono integre o meno. Se non configuri un controllo dello stato di integrità durante la creazione del servizio, il traffico verrà indirizzato alle istanze di servizio indipendentemente dallo stato di integrità delle istanze. Quando configuri un controllo dello stato, per impostazione predefinita AWS Cloud Map restituisce risorse integre. Puoi utilizzare il `HealthStatus` parametro dell'`DiscoverInstances`API per filtrare le risorse in base allo stato di integrità e ottenere un elenco di risorse non integre. Puoi anche utilizzare l'`GetInstancesHealthStatus`API per recuperare lo stato di salute di una particolare istanza del servizio.

Puoi configurare un controllo dello stato di Route 53 o un controllo dello stato personalizzato di terze parti quando crei un AWS Cloud Map servizio.

Controllo dell'integrità di Route 53

Se specifichi le impostazioni per un controllo dello stato di Amazon Route 53, AWS Cloud Map crea un controllo dello stato di Route 53 ogni volta che registri un'istanza ed elimina il controllo dello stato quando annulli la registrazione dell'istanza.

Per i namespace DNS pubblici, AWS Cloud Map associa il controllo dello stato al record Route 53 AWS Cloud Map creato quando si registra un'istanza. Se si specificano entrambi `A` i tipi di `AAAA` record nella configurazione DNS di un servizio, AWS Cloud Map crea un controllo dello stato che utilizza l' IPv4 indirizzo per verificare lo stato della risorsa. Se l'endpoint specificato dall' IPv4 indirizzo non è integro, Route 53 considera non integri sia i record che i record. `A` `AAAA` Se si specifica un tipo di `CNAME` record nella configurazione DNS di un servizio, non è possibile configurare un controllo dello stato di Route 53.

Per i namespace di cui utilizzi le chiamate API per scoprire le istanze, AWS Cloud Map crea un controllo dello stato di Route 53. Tuttavia, non esiste alcun record DNS AWS Cloud Map a cui associare il controllo dello stato. Per determinare se un controllo sanitario è corretto, puoi configurare il monitoraggio utilizzando la console Route 53 o Amazon CloudWatch. Per ulteriori informazioni sull'uso della console Route 53, consulta Get Notified When a Health Check Fails nella Amazon Route 53 Developer Guide. Per ulteriori informazioni sull'utilizzo CloudWatch, PutMetricAlarmconsulta Amazon CloudWatch API Reference.

> **Note**
>
> - Non puoi configurare un controllo dello stato di Amazon Route 53 per un servizio creato in uno spazio dei nomi DNS privato.
> - Un controllore dello stato di Route 53 in ogni controllo dello stato Regione AWS invia una richiesta di controllo dello stato a un endpoint ogni 30 secondi. In media, il tuo endpoint riceve una richiesta di controllo dello stato ogni due secondi. Tuttavia, i controlli dell'integrità non si coordinano tra loro. Pertanto, a volte è possibile che si verifichino diverse richieste in un secondo, seguite da alcuni secondi senza alcun controllo dell'integrità. Per un elenco delle aree in cui viene effettuato il controllo dello stato di salute, consulta Regioni.

Per informazioni sui costi per i controlli sanitari della Route 53, consulta i prezzi della Route 53.

Controlli dell'integrità personalizzati

Se AWS Cloud Map configuri l'utilizzo di un controllo sanitario personalizzato quando registri un'istanza, devi utilizzare un controllore sanitario di terze parti per valutare lo stato delle tue risorse. I controlli dello stato personalizzati sono utili nei seguenti casi:

- Non puoi utilizzare un controllo sanitario della Route 53 perché la risorsa non è disponibile su Internet. Ad esempio, supponiamo di avere un'istanza che si trova in un Amazon VPC. Puoi utilizzare un controllo sanitario personalizzato per questa istanza. Tuttavia, affinché il controllo dello stato funzioni, anche il tuo health checker deve trovarsi nello stesso VPC dell'istanza.
- Se si desidera utilizzare uno strumento di controllo dello stato di terza parte indipendente dalla posizione delle risorse.

Quando utilizzi un controllo sanitario personalizzato, AWS Cloud Map non verifica direttamente lo stato di una determinata risorsa. Invece, il correttore sanitario di terze parti verifica lo stato della risorsa e restituisce uno stato all'applicazione. La tua candidatura dovrà quindi inviare una `UpdateInstanceCustomHealthStatus` richiesta che trasmetta questo stato a. AWS Cloud Map Se lo stato iniziale inoltrato è `UNHEALTHY` e se non ce n'è un altro `UpdateInstanceCustomHealthStatus` entro 30 secondi che riporti lo stato di`HEALTHY`, si conferma che la risorsa non è integra. AWS Cloud Map interrompe l'indirizzamento del traffico verso quella risorsa.

AWS Cloud Map configurazione DNS del servizio

Quando crei un servizio in uno spazio dei nomi che supporta il rilevamento delle istanze tramite query DNS, AWS Cloud Map crea record DNS di Route 53. È necessario specificare una politica di routing di Route 53 e un tipo di record DNS da applicare a tutti i record DNS di Route 53 creati. AWS Cloud Map

Policy di routing

Una politica di routing determina in che modo Route 53 risponde alle query DNS utilizzate per il rilevamento delle istanze di servizio. Le politiche di routing supportate e il modo in cui si relazionano sono le seguenti. AWS Cloud Map

Routing ponderato

Route 53 restituisce il valore applicabile da un'istanza di AWS Cloud Map servizio selezionata casualmente tra le istanze registrate utilizzando lo stesso servizio. AWS Cloud Map Tutti i record hanno lo stesso peso, per cui non è possibile instradare più o meno traffico verso un'istanza.

Ad esempio, supponiamo che il servizio includa configurazioni per un record A e un controllo dello stato di salute e che tu utilizzi il servizio per registrare 10 istanze. Route 53 risponde alle query DNS con l'indirizzo IP per un'istanza selezionata casualmente tra tutte quelle integre. Se nessuna istanza è integra, Route 53 risponde alle query DNS come se tutte le istanze fossero integre.

Se non si definisce un controllo dello stato per il servizio, Route 53 presuppone che tutte le istanze siano integre e restituisce il valore applicabile per un'istanza selezionata in modo casuale.

Per ulteriori informazioni, consulta Weighted Routing nella Amazon Route 53 Developer Guide.

Routing di risposta multivalore

Se definisci un controllo dello stato del servizio e il risultato del controllo è corretto, Route 53 restituisce il valore applicabile per un massimo di otto istanze.

Ad esempio, supponiamo che il servizio includa configurazioni per un record A e un controllo sanitario. È possibile utilizzare il servizio per registrare 10 istanze. Route 53 risponde alle query DNS con indirizzi IP solo per un massimo di otto istanze integre. Se meno di otto istanze sono integre, Route 53 risponde a ogni query DNS con gli indirizzi IP di tutte le istanze integre.

Se non si definisce un controllo dello stato per il servizio, Route 53 presuppone che tutte le istanze siano integre e restituisce i valori per massimo otto istanze.

Per ulteriori informazioni, consulta Multivalue Answer Routing nella Amazon Route 53 Developer Guide.

Tipo di record

Un tipo di record DNS Route 53 determina il tipo di valore che Route 53 restituisce in risposta alle query DNS utilizzate per il rilevamento delle istanze di servizio. I diversi tipi di record DNS che è possibile specificare e i valori associati restituiti da Route 53 in risposta alle query sono i seguenti.

A

Se si specifica questo tipo, Route 53 restituisce l'indirizzo IP della risorsa in un IPv4 formato, ad esempio 192.0.2.44.

AAAA

Se si specifica questo tipo, Route 53 restituisce l'indirizzo IP della risorsa in un IPv6 formato, ad esempio 2001:0 db 8:85 a 3:0000:0000:abcd: 0001:2345.

CNAME

Se si specifica questo tipo, Route 53 restituisce il nome di dominio della risorsa (ad esempio www.example.com).

> **Note**
>
> - Per configurare un record DNS CNAME, è necessario specificare la politica di routing di routing ponderata.
> - Quando configuri un record DNS CNAME, non puoi configurare un controllo dello stato di Route 53.

SRV

Se si specifica questo tipo, Route 53 restituisce il valore di un SRV record. Il valore per un record SRV utilizza i seguenti valori:

```
priority weight port service-hostname
```

Considera i seguenti aspetti:

- I valori di `priority` e `weight` sono entrambi impostati su 1 e non possono essere modificati.
- `port`For, AWS Cloud Map utilizza il valore specificato per Port (AWS_INSTANCE_PORT) quando si registra un'istanza.
- Il valore di `service-hostname` è una concatenazione dei valori seguenti:
  - Il valore che specificate per Service instance ID (instanceID) quando registrate un'istanza
  - Il nome del servizio
  - Il nome dello spazio dei nomi

  Ad esempio, supponete di specificare test come ID di istanza quando registrate un'istanza. Il nome del servizio è backend e il nome dello spazio dei nomi è example.com. AWS Cloud Map assegna il seguente valore all'attributo nel record SRV: **`service-hostname`**

  `test.backend.example.com`

> **Note**
>
> Se si specifica un IPv4 indirizzo, un IPv6 indirizzo o entrambi quando si registra un'istanza, crea AWS Cloud Map automaticamente record A e/o AAAA con lo stesso nome del valore del record **`service-hostname`** SRV.

Puoi specificare i tipi di record nelle seguenti combinazioni:

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

Se specificate i tipi di record A e AAAA, potete specificare un indirizzo IPv4 IP, un indirizzo IPv6 IP o entrambi quando registrate un'istanza.

Creazione di un AWS Cloud Map servizio per un componente dell'applicazione

Dopo aver creato uno spazio dei nomi, puoi creare servizi per rappresentare diversi componenti dell'applicazione che servono a scopi particolari. Ad esempio, è possibile creare un servizio per le risorse dell'applicazione che elaborano i pagamenti.

> **Note**
>
> Non è possibile creare più servizi accessibili tramite query DNS con nomi che differiscono solo in base alle maiuscole e minuscole (come EXAMPLE ed example). In questo modo, questi servizi avranno lo stesso nome DNS. Se utilizzi uno spazio dei nomi accessibile solo tramite chiamate API, puoi creare servizi con nomi che differiscono solo per maiuscole e minuscole.

Segui questi passaggi per creare un servizio utilizzando AWS Management Console AWS CLI, e l'SDK per Python.

AWS Management Console

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo. https://console.aws.amazon.com/cloudmap/
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Nella pagina Namespaces (Spazio dei nomi), selezionare lo spazio dei nomi a cui aggiungere il servizio.
4. Nella *namespace-name* pagina Namespace:, scegli Crea servizio.
5. Per Nome del servizio, inserisci un nome che descriva le istanze registrate quando utilizzi questo servizio. Il valore viene utilizzato per scoprire le istanze del AWS Cloud Map servizio nelle chiamate API o nelle query DNS.

   > **Note**
   >
   > Se desideri AWS Cloud Map creare un record SRV quando registri un'istanza e utilizzi un sistema che richiede un formato SRV specifico (ad esempio HAProxy), specifica quanto segue per Nome del servizio:

- Iniziate il nome con un trattino basso (_), ad esempio _exampleservice.
- Termina il nome con, ad esempio. ***._protocol*** _tcp.

Quando si registra un'istanza, AWS Cloud Map crea un record SRV e assegna un nome concatenando il nome del servizio e il nome dello spazio dei nomi, ad esempio: _exampleservice._tcp.example.com

6. (Facoltativo) Per Descrizione del servizio, inserite una descrizione del servizio. La descrizione inserita qui viene visualizzata nella pagina Servizi e nella pagina di dettaglio di ciascun servizio.
7. Se il namespace supporta le query DNS, in Service Discovery Configuration è possibile configurare la reperibilità a livello di servizio. Scegli se consentire sia le chiamate API che le query DNS o solo le chiamate API per il rilevamento delle istanze in questo servizio.

> **Note**
>
> Se scegli le chiamate API, non AWS Cloud Map verranno creati record SRV quando registri un'istanza.

Se scegli API e DNS, segui questi passaggi per configurare i record DNS. Puoi aggiungere o rimuovere record DNS.

1. Per la politica di routing, seleziona la politica di routing di Amazon Route 53 per i record DNS AWS Cloud Map creati quando registri le istanze. Puoi scegliere tra Routing ponderato e Routing di risposte multivalore. Per ulteriori informazioni, consulta Policy di routing.

> **Note**
>
> Non è possibile utilizzare la console per configurare la creazione di un record AWS Cloud Map di alias Route 53 quando si registra un'istanza. Se desideri AWS Cloud Map creare record di alias per un sistema di bilanciamento del carico Elastic Load Balancing quando registri le istanze a livello di codice, scegli la politica Weighted routing for Routing.

2. Per Tipo di record, scegli il tipo di record DNS che determina il tipo di record restituito da Route 53 in risposta alle query DNS. AWS Cloud Map Per ulteriori informazioni, consulta Tipo di record.
3. Per TTL, specifica un valore numerico per definire il valore TTL (time to live), in secondi, a livello di servizio. Il valore di TTL determina per quanto tempo i resolver DNS memorizzano nella cache le informazioni per questo record prima che i resolver inoltrino un'altra query DNS ad Amazon Route 53 per ottenere impostazioni aggiornate.

8. In Configurazione Health check, per le opzioni Health check, scegli il tipo di controllo sanitario applicabile alle istanze del servizio. Puoi scegliere di non configurare alcun controllo dello stato oppure puoi scegliere tra un controllo dello stato della Route 53 o un controllo dello stato esterno per le tue istanze. Per ulteriori informazioni, consulta AWS Cloud Map configurazione del controllo dello stato del servizio.

> **Note**
>
> I controlli di integrità di Route 53 sono configurabili solo per i servizi nei namespace DNS pubblici.

Se scegli i controlli di integrità della Route 53, fornisci le seguenti informazioni.

1. Per la soglia di errore, fornisci un numero compreso tra 1 e 10 che definisca il numero di controlli di integrità consecutivi di Route 53 che un'istanza del servizio deve superare o non superare perché il suo stato di integrità cambi.
2. Per il protocollo Health check, selezionare il metodo che Route 53 utilizzerà per verificare lo stato delle istanze del servizio.
3. Se scegli il protocollo di controllo dello stato HTTP o HTTPS, per Health check path, fornisci un percorso che desideri che Amazon Route 53 richieda durante l'esecuzione dei controlli sanitari. Il percorso può essere qualsiasi valore, ad esempio il file`/docs/route53-health-check.html`. Quando la risorsa è integra, il valore restituito è un codice di stato HTTP in formato 2xx o 3xx. È inoltre possibile includere i parametri di stringa di query, ad esempio, `/welcome.html?language=jp&login=y`. La console AWS Cloud Map aggiunge automaticamente una barra (/) iniziale.

Per ulteriori informazioni sui controlli di integrità di Route 53, consulta How Amazon Route 53 Determina se un Health Check è integro nella Amazon Route 53 Developer Guide.

9. (Facoltativo) In Tag, scegli Aggiungi tag, quindi specifica una chiave e un valore per etichettare il tuo namespace. Puoi specificare uno o più tag da aggiungere al tuo namespace. I tag consentono di classificare le AWS risorse in modo da gestirle più facilmente. Per ulteriori informazioni, consulta Taggare le tue risorse AWS Cloud Map.
10. Selezionare Create service (Crea servizio).

AWS CLI

- Crea un servizio con il `create-service` comando. Sostituisci i *red* valori con i tuoi.

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

Output:

```
{
        "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)

Se non l'hai ancora `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo `Boto3` qui.

1. Importa `Boto3` e usa `servicediscovery` come servizio.

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

2. Crea un servizio con`create_service()`. Sostituisci i *red* valori con i tuoi. Per ulteriori informazioni, consulta create_service.

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

Esempio di output di risposta

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

```
            'NamespaceId': 'ns-xxxxxxxxxxx',
            'RoutingPolicy': 'MULTIVALUE',
        },
        'Id': 'srv-xxxxxxxxxxx',
        'Name': 'service-name',
        'NamespaceId': 'ns-xxxxxxxxxxx',
    },
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Passaggi successivi

Dopo aver creato un servizio, è possibile registrare le risorse dell'applicazione come istanze di servizio che contengono informazioni su come l'applicazione può localizzare la risorsa. Per ulteriori informazioni sulla registrazione delle istanze AWS Cloud Map di servizio, vedere. Registrazione di una risorsa come istanza di servizio AWS Cloud Map

Puoi anche specificare metadati personalizzati come i pesi degli endpoint, i timeout delle API e riprovare le politiche come attributi del servizio dopo aver creato un servizio. Per ulteriori informazioni, consulta le pagine ServiceAttributes e `UpdateServiceAttributes` nella Documentazione di riferimento dell'API AWS Cloud Map .

Aggiornamento di un AWS Cloud Map servizio

A seconda della configurazione del servizio, puoi aggiornarne i tag, la soglia di errore del controllo di integrità di Route 53 e il time to live (TTL) per i resolver DNS. Per aggiornare un servizio, eseguire la procedura seguente.

AWS Management Console

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Nella pagina Namespace, scegli lo spazio dei nomi in cui viene creato il servizio.
4. Nella ***`namespace-name`*** pagina Namespace:, seleziona il servizio che desideri modificare e scegli Visualizza dettagli.

5. Nella ***service-name*** pagina Servizio:, scegli Modifica.

> **Note**
>
> Non puoi utilizzare il flusso di lavoro del pulsante Modifica per modificare i valori per i servizi che consentono solo chiamate API, ad esempio il rilevamento. Tuttavia, puoi aggiungere o rimuovere tag nella *service-name* pagina Service:.

6. Nella pagina Modifica servizio, in Descrizione del servizio, è possibile aggiornare qualsiasi descrizione del servizio precedentemente impostata o aggiungere una nuova descrizione. Puoi anche aggiungere tag e aggiornare il TTL per i resolver DNS.
7. Nella configurazione DNS, per TTL, puoi specificare un periodo di tempo aggiornato, in secondi, che determina per quanto tempo i resolver DNS memorizzano nella cache le informazioni per questo record prima che i resolver inoltrino un'altra query DNS ad Amazon Route 53 per ottenere impostazioni aggiornate.
8. Se hai impostato i controlli di integrità di Route 53, per Soglia di errore, puoi specificare un nuovo numero compreso tra 1 e 10 che definisce il numero di controlli di integrità consecutivi di Route 53 che un'istanza del servizio deve superare o fallire perché il suo stato di integrità cambi.
9. Scegli il servizio di aggiornamento.

AWS CLI

- Aggiorna un servizio con il `update-service` comando (sostituisci il *red* valore con il tuo).

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

Output:

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

AWS SDK for Python (Boto3)

1. Se non l'hai ancora `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo `Boto3` qui.
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Aggiorna un servizio con `update_service()` (sostituisci il *red* valore con il tuo).

```
response = client.update_service(
    Id='srv-xxxxxxxxxxx',
    Service={
        'DnsConfig': {
            'DnsRecords': [
                {
                    'TTL': 300,
                    'Type': 'A',
                },
            ],
        },
        'Description': "new description",
    }
)
```

Esempio di output di risposta

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

Elencare AWS Cloud Map i servizi in un namespace

Per visualizzare un elenco dei servizi creati in uno spazio dei nomi, eseguire la seguente procedura.

AWS Management Console

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo. https://console.aws.amazon.com/cloudmap/

2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Scegliere il nome dello spazio dei nomi contenente il servizio parte che fa parte dell'elenco. Puoi visualizzare un elenco di tutti i servizi in Servizi e inserire il nome o l'ID del servizio nel campo di ricerca per trovare un servizio specifico.

AWS CLI

- Elenca i servizi con il `list-services` comando. Il comando seguente elenca tutti i servizi in uno spazio dei nomi utilizzando l'ID dello spazio dei nomi come filtro. Sostituisci il valore *red* con uno in tuo possesso.

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

AWS SDK for Python (Boto3)

1. Se non l'hai ancora `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo qui. `Boto3`
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Elenca i servizi con`list_services()`.

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

Esempio di output di risposta

```
{
    '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-xxxxxxxxxxxxxxxx',
            'Name': 'myservice',
        },
    ],
    'ResponseMetadata': {
        '...': '...',
    },
}
```

Eliminazione di un servizio AWS Cloud Map

Prima di eliminare un servizio, è necessario annullare la registrazione di tutte le istanze del servizio registrate utilizzando il servizio. Per ulteriori informazioni, consulta Annullamento della registrazione di un'istanza di servizio AWS Cloud Map.

Dopo aver annullato la registrazione di tutte le istanze registrate utilizzando il servizio, esegui la seguente procedura per eliminare il servizio.

AWS Management Console

1. Accedi a AWS Management Console e apri la console all' AWS Cloud Map indirizzo. https://console.aws.amazon.com/cloudmap/
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Scegliere l'opzione per lo spazio dei nomi contenente il servizio che si desidera eliminare.
4. Nella *namespace-name* pagina Namespace:, scegli l'opzione per il servizio che desideri eliminare.
5. Scegli Elimina.
6. Conferma l'eliminazione del servizio.

AWS CLI

- Elimina un servizio con il `delete-service` comando (sostituisci il *red* valore con il tuo).

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

AWS SDK for Python (Boto3)

1. Se non l'hai ancora `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo `Boto3` qui.
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Elimina un servizio con `delete_service()` (sostituisci il *red* valore con il tuo).

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

Esempio di output di risposta

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

AWS Cloud Map istanze di servizio

Ogni istanza del servizio contiene informazioni su come individuare una risorsa, ad esempio un server Web, per un'applicazione. Dopo aver registrato le istanze, puoi individuarle utilizzando le query DNS o l'azione API. AWS Cloud Map DiscoverInstances Le risorse che puoi registrare includono, a titolo esemplificativo ma non esaustivo, le seguenti:

- EC2 Istanze Amazon
- Tabelle Amazon DynamoDB
- Bucket Amazon S3
- Code di Amazon Simple Queue Service (Amazon SQS)
- APIs distribuito su Amazon API Gateway

Puoi specificare i valori degli attributi per le istanze di servizi e i client possono utilizzare questi attributi per filtrare le risorse restituite. AWS Cloud Map Ad esempio, un'applicazione può richiedere le risorse in una fase particolare della distribuzione, come BETA o PROD. È inoltre possibile utilizzare gli attributi per il controllo delle versioni.

Le seguenti procedure descrivono come registrare le risorse dell'applicazione come istanze di servizio, visualizzare un elenco di istanze registrate in un servizio, modificare determinati parametri di istanza e annullare la registrazione di un'istanza.

Argomenti

- Registrazione di una risorsa come istanza di servizio AWS Cloud Map
- Elenco delle istanze AWS Cloud Map del servizio
- Aggiornamento di un'istanza AWS Cloud Map del servizio
- Annullamento della registrazione di un'istanza di servizio AWS Cloud Map

Registrazione di una risorsa come istanza di servizio AWS Cloud Map

Puoi registrare le risorse dell'applicazione come istanze in un servizio. AWS Cloud Map Ad esempio, supponiamo di aver creato un servizio chiamato `users` per tutte le risorse dell'applicazione che

gestiscono i dati degli utenti. È quindi possibile registrare una tabella DynamoDB utilizzata per archiviare i dati utente come istanza in questo servizio.

> **Note**
>
> Le seguenti funzionalità non sono disponibili sulla AWS Cloud Map console:
>
> - Quando registri un'istanza di servizio utilizzando la console, non puoi creare un record di alias che indirizza il traffico verso un sistema di bilanciamento del carico Elastic Load Balancing (ELB). Quando si registra un'istanza, è necessario includere l'attributo `AWS_ALIAS_DNS_NAME`. Per ulteriori informazioni, consulta RegisterInstance nella documentazione di riferimento dell'API AWS Cloud Map .
> - Se si registra un'istanza che utilizza un servizio che include un controllo dello stato personalizzato, non è possibile specificare lo stato iniziale del controllo di stato personalizzato. Per impostazione predefinita, lo stato iniziale per i controlli di stato personalizzati è Healthy (Integro). Se si desidera impostare lo stato iniziale su Unhealthy (Non integro), registrare l'istanza in modo programamtico e includere lattributo `AWS_INIT_HEALTH_STATUS`. Per ulteriori informazioni, consulta RegisterInstance nella documentazione di riferimento dell'API AWS Cloud Map .

Per registrare un'istanza in un servizio, segui questi passaggi.

AWS Management Console

1. Accedi AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Nella pagina Namespaces (Spazi dei nomi), scegliere lo spazio dei nomi che contiene il servizio che si desidera utilizzare come modello per registrare l'istanza di un servizio.
4. Nella *`namespace-name`* pagina Namespace:, scegli il servizio che desideri utilizzare.
5. Nella *`service-name`* pagina Servizio:, scegli Registra istanza del servizio.
6. Nella pagina Registra istanza del servizio, scegli un tipo di istanza. A seconda della configurazione del namespace Instance Discovery, puoi scegliere di specificare un indirizzo IP, un ID di EC2 istanza Amazon o altre informazioni identificative per una risorsa che non dispone di un indirizzo IP.

> **Note**
>
> Puoi scegliere l'EC2 istanza solo nei namespace HTTP.

7. Per Service Instance ID, fornisci un identificatore associato all'istanza del servizio.

> **Note**
>
> Se desideri aggiornare un'istanza esistente, fornisci l'identificatore associato all'istanza che desideri aggiornare. Quindi, utilizza i passaggi successivi per aggiornare i valori e registrare nuovamente l'istanza.

8. In base al tipo di istanza scelto, esegui i seguenti passaggi.

> **Important**
>
> Non è possibile utilizzare il `AWS_` prefisso (senza distinzione tra maiuscole e minuscole) in una chiave quando si specifica un attributo personalizzato.

Tipo di istanza	Fasi	
Indirizzo IP	a. In Attributi standard, per IPv4indirizzo, fornisci un IPv4 indirizzo, se disponibile, a cui l'applicazione può accedere alla risorsa associata a questa istanza di servizio. b. Per IPv6 l'indirizzo, fornisci un indirizzo IPv6 IP, se disponibile, a cui le applicazioni possono accedere alla risorsa	

Tipo di istanza	Fasi	
	associata a questa istanza di servizio. c. Per Porta, specifica qualsiasi porta che l'applicazione deve includere per accedere alla risorsa associata a questa istanza di servizio. La porta è necessaria quando il servizio include un record SRV o un controllo dello stato di Amazon Route 53. d. (Facoltativo) In Attributi personalizzati, specifica le coppie chiave-valore che desideri associare alla risorsa.	
EC2 istanza	a. Ad EC2 esempio ID, seleziona l'ID dell' EC2istanza Amazon che desideri registrare come istanza di AWS Cloud Map servizio. b. (Facoltativo) In Attributi personalizzati, specifica le coppie chiave-valore che desideri associare alla risorsa.	

Tipo di istanza	Fasi	
Informazioni di identific azione per un'altra risorsa	a. In Attributi standard, se la configurazione del servizio include un record DNS CNAME, vedrai un campo CNAME. Per CNAME, specifica il nome di dominio che desideri che Route 53 restituisca in risposta alle query DNS (ad esempio,) . `example.com` b. In Attributi personali zzati, specifica qualsiasi informazione identific ativa per una risorsa che non sia un indirizzo IP o un ID di EC2 istanza Amazon come coppia chiave-valore. Ad esempio, è possibile registrare una funzione Lambda specificando una chiave chiamata `function` e fornendo il nome della funzione Lambda come valore. È inoltre possibile specificare una chiave chiamata `name` e fornire un nome da utilizzare per il rilevamento programma tico delle istanze.	

9. Selezionare Register service instance (Registra istanza del servizio).

AWS CLI

- Quando invii una `RegisterInstance` richiesta:
  - Per ogni record DNS definito nel servizio specificato da`ServiceId`, viene creato o aggiornato un record nella zona ospitata associata allo spazio dei nomi corrispondente.
  - Se il servizio include`HealthCheckConfig`, viene creato un controllo dello stato di salute in base alle impostazioni nella configurazione del controllo dello stato.
  - Tutti i controlli sanitari sono associati a ciascuno dei record nuovi o aggiornati.

  Registra un'istanza di servizio con il `register-instance` comando (sostituisci i *red* valori con i tuoi).

```
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. Se non l'hai ancora `Boto3` installata, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo `Boto3` qui.
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Quando invii una `RegisterInstance` richiesta:
   - Per ogni record DNS definito nel servizio specificato da`ServiceId`, viene creato o aggiornato un record nella zona ospitata associata allo spazio dei nomi corrispondente.
   - Se il servizio include`HealthCheckConfig`, viene creato un controllo dello stato di salute in base alle impostazioni nella configurazione del controllo dello stato.
   - Tutti i controlli sanitari sono associati a ciascuno dei record nuovi o aggiornati.

   Registra un'istanza di servizio con `register_instance()` (sostituisci i *red* valori con i tuoi).

```
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)
```

Esempio di output di risposta

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

Elenco delle istanze AWS Cloud Map del servizio

Per visualizzare un elenco delle istanze del servizio registrate utilizzando un servizio, eseguire la procedura seguente.

AWS Management Console

1. Accedi AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Scegliere il nome dello spazio dei nomi contenente il servizio per cui si desidera elencare le istanze dei servizi.
4. Scegliere il nome del servizio utilizzato per creare le istanze del servizio. Vedrai un elenco di istanze in Istanze di servizio. Puoi inserire l'ID dell'istanza nel campo di ricerca per elencare un'istanza specifica.

AWS CLI

- Elenca le istanze del servizio con il `list-instances` comando (sostituisci il *red* valore con il tuo).

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

AWS SDK for Python (Boto3)

1. Se non l'hai già `Boto3` installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo `Boto3` qui.
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Elenca le istanze del servizio con `list_instances()` (sostituisci il *red* valore con il tuo).

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

Esempio di output di risposta

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

```
}
```

Aggiornamento di un'istanza AWS Cloud Map del servizio

È possibile aggiornare le istanze del servizio in due modi, a seconda dei valori che si desidera aggiornare:

- Aggiorna qualsiasi valore: se desideri aggiornare uno dei valori specificati per un'istanza di servizio al momento della registrazione, inclusi gli attributi personalizzati, devi registrare nuovamente l'istanza del servizio e specificare nuovamente tutti i valori. Segui i passaggi indicatiRegistrazione di una risorsa come istanza di servizio AWS Cloud Map, specificando l'ID dell'istanza di servizio esistente per Service Instance ID.

  In alternativa, puoi utilizzare l'`RegisterInstance`API. È possibile specificare l'ID dell'istanza e del servizio esistenti utilizzando i `ServiceId` parametri `InstanceId` and e specificare nuovamente altri valori.
- Aggiornare solo attributi personalizzati: se si desidera aggiornare solo gli attributi personalizzati per un'istanza del servizio, non è necessario registrare nuovamente l'istanza. È possibile aggiornare solo questi valori. Per informazioni, consulta Aggiornamento degli attributi personalizzati per un'istanza di servizio.

Aggiornamento degli attributi personalizzati per un'istanza di servizio

Per aggiornare solo gli attributi personalizzati per un'istanza del servizio

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo https://console.aws.amazon.com/cloudmap/.
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Nella pagina Namespaces (Spazi dei nomi), scegliere lo spazio dei nomi che contiene il servizio utilizzato in origine per registrare l'istanza di un servizio.
4. Nella *`namespace-name`* pagina Namespace:, scegli il servizio che hai usato per registrare l'istanza del servizio.
5. Nella *`service-name`* pagina Servizio:, scegli il nome dell'istanza del servizio che desideri aggiornare.
6. Nella sezione Attributi personalizzati scegliere Modifica.

7. Nella pagina Modifica istanza del servizio: ***instance-name*** pagina, aggiungi, rimuovi o aggiorna gli attributi personalizzati. È possibile aggiornare sia le chiavi che i valori per gli attributi esistenti.
8. Scegliere Aggiorna istanza del servizio.

Annullamento della registrazione di un'istanza di servizio AWS Cloud Map

Prima di eliminare un servizio, è necessario annullare la registrazione di tutte le istanze del servizio registrate utilizzando il servizio.

Per annullare la registrazione di un'istanza di un servizio, eseguire la procedura seguente.

AWS Management Console

1. Accedi a AWS Management Console e apri la AWS Cloud Map console all'indirizzo. https://console.aws.amazon.com/cloudmap/
2. Nel riquadro di navigazione seleziona Namespaces (Spazio dei nomi).
3. Scegliere l'opzione per lo spazio dei nomi contenente l'istanza del servizio per la quale si desidera annullare la registrazione.
4. Nella *namespace-name* pagina Namespace:, scegli il servizio che hai usato per registrare l'istanza del servizio.
5. Nella *service-name* pagina Servizio:, scegli l'istanza del servizio che desideri annullare la registrazione.
6. Scegli Annulla registrazione.
7. Confermare che si desidera annullare la registrazione dell'istanza del servizio.

AWS CLI

- Annulla la registrazione di un'istanza di servizio con il `deregister-instance` comando (sostituisci *red* i valori con i tuoi). Questo comando elimina i record DNS di Amazon Route 53 e tutti i controlli di integrità AWS Cloud Map creati per l'istanza specificata.

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

```
    --instance-id myservice-53
```

AWS SDK for Python (Boto3)

1. Se non l'hai ancora Boto3 installato, puoi trovare le istruzioni per l'installazione, la configurazione e l'utilizzo qui. Boto3
2. Importa `Boto3` e usa `servicediscovery` come servizio.

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

3. Annulla la registrazione di un'istanza del servizio con `deregister-instance()` (sostituisci i *red* valori con i tuoi). Questo comando elimina i record DNS di Amazon Route 53 e tutti i controlli di integrità AWS Cloud Map creati per l'istanza specificata.

```
response = client.deregister_instance(
    InstanceId='myservice-53',
    ServiceId='srv-xxxxxxxxx',
)
# If you want to see the response
print(response)
```

Esempio di output di risposta

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

Sicurezza in AWS Cloud Map

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di un data center e di un'architettura di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra AWS te e te. Il modello di responsabilità condivisa descrive questo come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi nel AWS cloud. AWS ti fornisce anche servizi che puoi utilizzare in modo sicuro. I revisori di terze parti testano e verificano regolarmente l'efficacia della sicurezza come parte dei programmi di conformitàAWS. Per ulteriori informazioni sui programmi di conformità applicabili AWS Cloud Map, consulta AWS Services in Scope by Compliance Program.
- Sicurezza nel cloud: la tua responsabilità è determinata dal AWS servizio che utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della tua azienda e le leggi e normative vigenti.

La seguente documentazione ti aiuta a capire come applicare il modello di responsabilità condivisa durante l'utilizzo AWS Cloud Map. I seguenti argomenti mostrano come eseguire la configurazione AWS Cloud Map per soddisfare gli obiettivi di sicurezza e conformità. Imparerai anche a utilizzare altri AWS servizi che ti aiutano a monitorare e proteggere AWS Cloud Map le tue risorse.

Argomenti

Identity and Access Management per AWS Cloud Map

AWS Identity and Access Management (IAM) è un software Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori

IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse. AWS Cloud Map IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

Argomenti

Destinatari

Il modo in cui usi AWS Identity and Access Management (IAM) varia a seconda del lavoro che AWS Cloud Map svolgi.

Utente del servizio: se utilizzi il AWS Cloud Map servizio per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni necessarie. Man mano che utilizzi più AWS Cloud Map funzionalità per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità di AWS Cloud Map, consulta Risoluzione dei problemi di AWS Cloud Map identità e accesso.

Amministratore del servizio: se sei responsabile delle AWS Cloud Map risorse della tua azienda, probabilmente hai pieno accesso a AWS Cloud Map. È tuo compito determinare a quali AWS Cloud Map funzionalità e risorse devono accedere gli utenti del servizio. Devi inviare le richieste all'amministratore IAM per cambiare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per saperne di più su come la tua azienda può utilizzare IAM con AWS Cloud Map, consultaCome AWS Cloud Map funziona con IAM.

Amministratore IAM: un amministratore IAM potrebbe essere interessato a ottenere dei dettagli su come scrivere policy per gestire l'accesso a AWS Cloud Map. Per visualizzare esempi di policy

AWS Cloud Map basate sull'identità che puoi utilizzare in IAM, consulta. Esempi di policy basate sull'identità per AWS Cloud Map

Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi essere autenticato (aver effettuato l' Utente root dell'account AWS accesso AWS) come utente IAM o assumendo un ruolo IAM.

Puoi accedere AWS come identità federata utilizzando le credenziali fornite tramite una fonte di identità. AWS IAM Identity Center Gli utenti (IAM Identity Center), l'autenticazione Single Sign-On della tua azienda e le tue credenziali di Google o Facebook sono esempi di identità federate. Se accedi come identità federata, l'amministratore ha configurato in precedenza la federazione delle identità utilizzando i ruoli IAM. Quando accedi AWS utilizzando la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente, puoi accedere al AWS Management Console o al portale di AWS accesso. Per ulteriori informazioni sull'accesso a AWS, vedi Come accedere al tuo Account AWS nella Guida per l'Accedi ad AWS utente.

Se accedi a AWS livello di codice, AWS fornisce un kit di sviluppo software (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le tue richieste utilizzando le tue credenziali. Se non utilizzi AWS strumenti, devi firmare tu stesso le richieste. Per ulteriori informazioni sul metodo consigliato per la firma delle richieste, consulta Signature Version 4 AWS per le richieste API nella Guida per l'utente IAM.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. Ad esempio, ti AWS consiglia di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza del tuo account. Per ulteriori informazioni, consulta Autenticazione a più fattori nella Guida per l'utente di AWS IAM Identity Center e Utilizzo dell'autenticazione a più fattori (MFA)AWS in IAM nella Guida per l'utente IAM.

Account AWS utente root

Quando si crea un account Account AWS, si inizia con un'identità di accesso che ha accesso completo a tutte Servizi AWS le risorse dell'account. Questa identità è denominata utente Account AWS root ed è accessibile effettuando l'accesso con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conserva le credenziali dell'utente root e utilizzale per eseguire le operazioni che solo l'utente root

può eseguire. Per un elenco completo delle attività che richiedono l'accesso come utente root, consulta la sezione Attività che richiedono le credenziali dell'utente root nella Guida per l'utente IAM.

Identità federata

Come procedura consigliata, richiedi agli utenti umani, compresi gli utenti che richiedono l'accesso come amministratore, di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente dell'elenco utenti aziendale, di un provider di identità Web AWS Directory Service, della directory Identity Center o di qualsiasi utente che accede utilizzando le Servizi AWS credenziali fornite tramite un'origine di identità. Quando le identità federate accedono Account AWS, assumono ruoli e i ruoli forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, consigliamo di utilizzare AWS IAM Identity Center. Puoi creare utenti e gruppi in IAM Identity Center oppure puoi connetterti e sincronizzarti con un set di utenti e gruppi nella tua fonte di identità per utilizzarli su tutte le tue applicazioni. Account AWS Per ulteriori informazioni su IAM Identity Center, consulta Cos'è IAM Identity Center? nella Guida per l'utente di AWS IAM Identity Center .

Utenti e gruppi IAM

Un utente IAM è un'identità interna Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ove possibile, consigliamo di fare affidamento a credenziali temporanee invece di creare utenti IAM con credenziali a lungo termine come le password e le chiavi di accesso. Tuttavia, se si hanno casi d'uso specifici che richiedono credenziali a lungo termine con utenti IAM, si consiglia di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta la pagina Rotazione periodica delle chiavi di accesso per casi d'uso che richiedono credenziali a lungo termine nella Guida per l'utente IAM.

Un gruppo IAM è un'identità che specifica un insieme di utenti IAM. Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, potresti avere un gruppo denominato IAMAdminse concedere a quel gruppo le autorizzazioni per amministrare le risorse IAM.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali

temporanee. Per ulteriori informazioni, consulta Casi d'uso per utenti IAM nella Guida per l'utente IAM.

Ruoli IAM

Un ruolo IAM è un'identità interna all'utente Account AWS che dispone di autorizzazioni specifiche. È simile a un utente IAM, ma non è associato a una persona specifica. Per assumere temporaneamente un ruolo IAM in AWS Management Console, puoi passare da un ruolo utente a un ruolo IAM (console). Puoi assumere un ruolo chiamando un'operazione AWS CLI o AWS API o utilizzando un URL personalizzato. Per ulteriori informazioni sui metodi per l'utilizzo dei ruoli, consulta Utilizzo di ruoli IAM nella Guida per l'utente IAM.

I ruoli IAM con credenziali temporanee sono utili nelle seguenti situazioni:

- Accesso utente federato: per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per ulteriori informazioni sulla federazione dei ruoli, consulta Create a role for a third-party identity provider (federation) nella Guida per l'utente IAM. Se utilizzi IAM Identity Center, configura un set di autorizzazioni. IAM Identity Center mette in correlazione il set di autorizzazioni con un ruolo in IAM per controllare a cosa possono accedere le identità dopo l'autenticazione. Per informazioni sui set di autorizzazioni, consulta Set di autorizzazioni nella Guida per l'utente di AWS IAM Identity Center .
- Autorizzazioni utente IAM temporanee: un utente IAM o un ruolo può assumere un ruolo IAM per ottenere temporaneamente autorizzazioni diverse per un'attività specifica.
- Accesso multi-account: è possibile utilizzare un ruolo IAM per permettere a un utente (un principale affidabile) con un account diverso di accedere alle risorse nell'account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, con alcuni Servizi AWS, è possibile allegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per informazioni sulle differenze tra ruoli e policy basate su risorse per l'accesso multi-account, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.
- Accesso a più servizi: alcuni Servizi AWS utilizzano le funzionalità di altri Servizi AWS. Ad esempio, quando effettui una chiamata in un servizio, è normale che quel servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.

- Sessioni di accesso inoltrato (FAS): quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama an Servizio AWS, combinate con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli delle policy relative alle richieste FAS, consulta Forward access sessions.
- Ruolo di servizio: un ruolo di servizio è un ruolo IAM che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione Create a role to delegate permissions to an Servizio AWS nella Guida per l'utente IAM.
- Ruolo collegato al servizio: un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un. Servizio AWS Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

- Applicazioni in esecuzione su Amazon EC2: puoi utilizzare un ruolo IAM per gestire le credenziali temporanee per le applicazioni in esecuzione su un' EC2 istanza e che AWS CLI effettuano richieste AWS API. Questa soluzione è preferibile alla memorizzazione delle chiavi di accesso all'interno dell' EC2 istanza. Per assegnare un AWS ruolo a un' EC2 istanza e renderlo disponibile per tutte le sue applicazioni, create un profilo di istanza collegato all'istanza. Un profilo di istanza contiene il ruolo e consente ai programmi in esecuzione sull' EC2 istanza di ottenere credenziali temporanee. Per ulteriori informazioni, consulta Utilizzare un ruolo IAM per concedere le autorizzazioni alle applicazioni in esecuzione su EC2 istanze Amazon nella IAM User Guide.

Gestione dell'accesso con policy

Puoi controllare l'accesso AWS creando policy e collegandole a AWS identità o risorse. Una policy è un oggetto AWS che, se associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste politiche quando un principale (utente, utente root o sessione di ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per ulteriori informazioni sulla struttura e sui contenuti dei documenti delle policy JSON, consulta Panoramica delle policy JSON nella Guida per l'utente IAM.

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire operazioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti l'autorizzazione a eseguire operazioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Le policy IAM definiscono le autorizzazioni relative a un'operazione, a prescindere dal metodo utilizzato per eseguirla. Ad esempio, supponiamo di disporre di una policy che consente l'operazione `iam:GetRole`. Un utente con tale policy può ottenere informazioni sul ruolo dall' AWS Management Console AWS CLI, dall'o dall' AWS API.

Policy basate sull'identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le operazioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente IAM.

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono integrate direttamente in un singolo utente, gruppo o ruolo. Le politiche gestite sono politiche autonome che puoi allegare a più utenti, gruppi e ruoli nel tuo Account AWS. Le politiche gestite includono politiche AWS gestite e politiche gestite dai clienti. Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta Scelta fra policy gestite e policy inline nella Guida per l'utente IAM.

Policy basate sulle risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Esempi di policy basate sulle risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le operazioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario specificare un principale in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non puoi utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

Liste di controllo degli accessi () ACLs

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLs sono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3 e Amazon VPC sono esempi di servizi che supportano. AWS WAF ACLs Per ulteriori informazioni ACLs, consulta la panoramica della lista di controllo degli accessi (ACL) nella Amazon Simple Storage Service Developer Guide.

Altri tipi di policy

AWS supporta tipi di policy aggiuntivi e meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- Limiti delle autorizzazioni: un limite delle autorizzazioni è una funzionalità avanzata nella quale si imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM (utente o ruolo IAM). È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal`sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta Limiti delle autorizzazioni per le entità IAM nella Guida per l'utente IAM.
- Politiche di controllo del servizio (SCPs): SCPs sono politiche JSON che specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa (OU) in. AWS Organizations AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata di più di proprietà dell' Account AWS azienda. Se abiliti tutte le funzionalità di un'organizzazione, puoi applicare le politiche di controllo del servizio (SCPs) a uno o tutti i tuoi account. L'SCP limita le autorizzazioni per le entità presenti negli account dei membri, inclusa ciascuna di esse. Utente root dell'account AWS Per ulteriori informazioni su Organizations and SCPs, consulta le politiche di controllo dei servizi nella Guida AWS Organizations per l'utente.
- Politiche di controllo delle risorse (RCPs): RCPs sono politiche JSON che puoi utilizzare per impostare le autorizzazioni massime disponibili per le risorse nei tuoi account senza aggiornare le politiche IAM allegate a ciascuna risorsa di tua proprietà. L'RCP limita le autorizzazioni per le risorse negli account dei membri e può influire sulle autorizzazioni effettive per le identità, incluse le Utente root dell'account AWS, indipendentemente dal fatto che appartengano o meno all'organizzazione. Per ulteriori informazioni su Organizations e RCPs, incluso un elenco di

Servizi AWS tale supporto RCPs, vedere Resource control policies (RCPs) nella Guida per l'AWS Organizations utente.

- Policy di sessione: le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta Policy di sessione nella Guida per l'utente IAM.

Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire o meno una richiesta quando sono coinvolti più tipi di policy, consulta la logica di valutazione delle policy nella IAM User Guide.

Come AWS Cloud Map funziona con IAM

Prima di utilizzare IAM per gestire l'accesso a AWS Cloud Map, scopri con quali funzionalità IAM è disponibile l'uso AWS Cloud Map.

Funzionalità IAM	AWS Cloud Map supporto
Policy basate su identità	Sì
Policy basate su risorse	No
Azioni di policy	Sì
Risorse relative alle policy	Sì
Chiavi di condizione della policy (specifica del servizio)	Sì
ACLs	No
ABAC (tag nelle policy)	Sì
Credenziali temporanee	Sì

Funzionalità IAM	AWS Cloud Map supporto
Inoltro delle sessioni di accesso (FAS)	Sì
Ruoli di servizio	No
Ruoli collegati al servizio	No

Per avere una panoramica di alto livello su come AWS Cloud Map e altri AWS servizi funzionano con la maggior parte delle funzionalità IAM, consulta AWS i servizi che funzionano con IAM nella IAM User Guide.

Politiche basate sull'identità per AWS Cloud Map

Supporta le policy basate su identità: sì

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le operazioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente nella Guida per l'utente IAM.

Con le policy basate su identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Non è possibile specificare l'entità principale in una policy basata sull'identità perché si applica all'utente o al ruolo a cui è associato. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta Guida di riferimento agli elementi delle policy JSON IAM nella Guida per l'utente di IAM.

Esempi di politiche basate sull'identità per AWS Cloud Map

Per visualizzare esempi di politiche basate sull' AWS Cloud Map identità, vedere. Esempi di policy basate sull'identità per AWS Cloud Map

Politiche basate sulle risorse all'interno AWS Cloud Map

Supporta le policy basate su risorse: no

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Esempi di policy basate sulle risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket

Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le operazioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario specificare un principale in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, puoi specificare un intero account o entità IAM in un altro account come principale in una policy basata sulle risorse. L'aggiunta di un principale multi-account a una policy basata sulle risorse rappresenta solo una parte della relazione di trust. Quando il principale e la risorsa sono diversi Account AWS, un amministratore IAM dell'account affidabile deve inoltre concedere all'entità principale (utente o ruolo) l'autorizzazione ad accedere alla risorsa. L'autorizzazione viene concessa collegando all'entità una policy basata sull'identità. Tuttavia, se una policy basata su risorse concede l'accesso a un principale nello stesso account, non sono richieste ulteriori policy basate su identità. Per ulteriori informazioni, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Azioni politiche per AWS Cloud Map

Supporta le operazioni di policy: si

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento `Action`di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso a un criterio. Le azioni politiche in genere hanno lo stesso nome dell'operazione AWS API associata. Ci sono alcune eccezioni, ad esempio le operazioni di sola autorizzazione che non hanno un'operazione API corrispondente. Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includi le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco di AWS Cloud Map azioni, vedere Azioni definite da AWS Cloud Map nel Service Authorization Reference.

Le azioni politiche in AWS Cloud Map uso utilizzano il seguente prefisso prima dell'azione:

```
servicediscovery
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

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

Per visualizzare esempi di politiche AWS Cloud Map basate sull'identità, vedere. Esempi di policy basate sull'identità per AWS Cloud Map

Risorse politiche per AWS Cloud Map

Supporta le risorse di policy: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento JSON `Resource`della policy specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Le istruzioni devono includere un elemento `Resource`o un elemento `NotResource`. Come best practice, specifica una risorsa utilizzando il suo nome della risorsa Amazon (ARN). È possibile eseguire questa operazione per operazioni che supportano un tipo di risorsa specifico, note come autorizzazioni a livello di risorsa.

Per le azioni che non supportano le autorizzazioni a livello di risorsa, ad esempio le operazioni di elenco, utilizza un carattere jolly (*) per indicare che l'istruzione si applica a tutte le risorse.

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

Per visualizzare un elenco dei tipi di AWS Cloud Map risorse e relativi ARNs, vedere Resources defined by AWS Cloud Map nel Service Authorization Reference. Per informazioni sulle operazioni con cui è possibile specificare l'ARN di ogni risorsa, consulta la sezione Operazioni definite da AWS Cloud Map.

Per visualizzare esempi di politiche AWS Cloud Map basate sull'identità, vedere. Esempi di policy basate sull'identità per AWS Cloud Map

Chiavi relative alle condizioni delle politiche per AWS Cloud Map

Supporta le chiavi di condizione delle policy specifiche del servizio: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento `Condition`(o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition`è facoltativo. È possibile compilare espressioni condizionali che utilizzano operatori di condizione, ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specifichi più elementi `Condition`in un'istruzione o più chiavi in un singolo elemento `Condition`, questi vengono valutati da AWS utilizzando un'operazione `AND`logica. Se si specificano più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'operazione logica. `OR` Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

È possibile anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, è possibile autorizzare un utente IAM ad accedere a una risorsa solo se è stata taggata con il relativo nome utente IAM. Per ulteriori informazioni, consulta Elementi delle policy IAM: variabili e tag nella Guida per l'utente di IAM.

AWS supporta chiavi di condizione globali e chiavi di condizione specifiche del servizio. Per visualizzare tutte le chiavi di condizione AWS globali, consulta le chiavi di contesto delle condizioni AWS globali nella Guida per l'utente IAM.

Per visualizzare un elenco di chiavi di AWS Cloud Map condizione, consulta Condition keys for AWS Cloud Map nel Service Authorization Reference. Per sapere con quali azioni e risorse puoi utilizzare una chiave di condizione, vedi Azioni definite da AWS Cloud Map.

AWS Cloud Map supporta le seguenti chiavi di condizione specifiche del servizio che puoi utilizzare per fornire filtri granulari per le tue politiche IAM.

`servicediscovery:NamespaceArn`

Filtro che consente di ottenere oggetti specificando l'Amazon Resource Name (ARN) per lo spazio dei nomi correlato.

`servicediscovery:NamespaceName`

Filtro che consente di ottenere oggetti specificando il nome dello spazio dei nomi correlato.

`servicediscovery:ServiceArn`

Filtro che consente di ottenere oggetti specificando l'Amazon Resource Name (ARN) per il servizio correlato.

`servicediscovery:ServiceName`

Filtro che consente di ottenere oggetti specificando il nome del servizio correlato.

Per visualizzare esempi di politiche basate sull'identità, consulta. AWS Cloud Map Esempi di policy basate sull'identità per AWS Cloud Map

ACLs in AWS Cloud Map

Supporti ACLs: no

Le liste di controllo degli accessi (ACLs) controllano quali principali (membri dell'account, utenti o ruoli) dispongono delle autorizzazioni per accedere a una risorsa. ACLs sono simili alle politiche basate sulle risorse, sebbene non utilizzino il formato del documento di policy JSON.

ABAC con AWS Cloud Map

Supporta ABAC (tag nelle policy): sì

Il controllo dell'accesso basato su attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base agli attributi. In AWS, questi attributi sono chiamati tag. Puoi allegare tag a entità IAM (utenti o ruoli) e a molte AWS risorse. L'assegnazione di tag alle entità e alle risorse è il primo passaggio di ABAC. In seguito, vengono progettate policy ABAC per consentire operazioni quando il tag dell'entità principale corrisponde al tag sulla risorsa a cui si sta provando ad accedere.

La strategia ABAC è utile in ambienti soggetti a una rapida crescita e aiuta in situazioni in cui la gestione delle policy diventa impegnativa.

Per controllare l'accesso basato su tag, fornisci informazioni sui tag nell'elemento condizione di una policy utilizzando le chiavi di condizione `aws:ResourceTag/`*`key-name`*, `aws:RequestTag/`*`key-name`*o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Yes (Sì). Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per ulteriori informazioni su ABAC, consulta Definizione delle autorizzazioni con autorizzazione ABAC nella Guida per l'utente IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consulta Utilizzo del controllo degli accessi basato su attributi (ABAC) nella Guida per l'utente di IAM.

Utilizzo di credenziali temporanee con AWS Cloud Map

Supporta le credenziali temporanee: sì

Alcuni Servizi AWS non funzionano quando si accede utilizzando credenziali temporanee. Per ulteriori informazioni, incluse quelle che Servizi AWS funzionano con credenziali temporanee, consulta la sezione relativa alla Servizi AWS compatibilità con IAM nella IAM User Guide.

Stai utilizzando credenziali temporanee se accedi AWS Management Console utilizzando qualsiasi metodo tranne nome utente e password. Ad esempio, quando accedi AWS utilizzando il link Single Sign-On (SSO) della tua azienda, tale processo crea automaticamente credenziali temporanee. Le credenziali temporanee vengono create in automatico anche quando accedi alla console come utente e poi cambi ruolo. Per ulteriori informazioni sullo scambio dei ruoli, consulta Passaggio da un ruolo utente a un ruolo IAM (console) nella Guida per l'utente IAM.

È possibile creare manualmente credenziali temporanee utilizzando l'API or. AWS CLI AWS È quindi possibile utilizzare tali credenziali temporanee per accedere. AWS AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta Credenziali di sicurezza provvisorie in IAM.

Sessioni di accesso diretto per AWS Cloud Map

Supporta l'inoltro delle sessioni di accesso (FAS): sì

Quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama an Servizio AWS, in combinazione con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli delle policy relative alle richieste FAS, consulta Forward access sessions.

Ruoli di servizio per AWS Cloud Map

Supporta i ruoli di servizio: no

Un ruolo di servizio è un ruolo IAM che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione Create a role to delegate permissions to an Servizio AWS nella Guida per l'utente IAM.

> **Warning**
>
> La modifica delle autorizzazioni per un ruolo di servizio potrebbe compromettere la funzionalità di AWS Cloud Map . Modifica i ruoli di servizio solo quando AWS Cloud Map fornisce indicazioni in tal senso.

Ruoli collegati ai servizi per AWS Cloud Map

Supporta i ruoli collegati ai servizi: no

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un. Servizio AWS Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati ai servizi, consulta Servizi AWS supportati da IAM. Trova un servizio nella tabella che include un `Yes` nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

Esempi di policy basate sull'identità per AWS Cloud Map

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare risorse AWS Cloud Map . Inoltre, non possono eseguire attività utilizzando AWS Management Console, AWS Command Line Interface (AWS CLI) o l'API. AWS Per concedere agli utenti l'autorizzazione a eseguire operazioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta Creazione di policy IAM (console) nella Guida per l'utente IAM.

Per informazioni dettagliate sulle azioni e sui tipi di risorse definiti da AWS Cloud Map, incluso il formato di ARNs per ogni tipo di risorsa, consulta Azioni, risorse e chiavi di condizione AWS Cloud Map nel Service Authorization Reference.

Argomenti

- Best practice per le policy

- Utilizzo della console di AWS Cloud Map
- AWS Cloud Map esempio di accesso alla console
- Consenti AWS Cloud Map agli utenti di visualizzare le proprie autorizzazioni
- Consenti l'accesso in lettura a tutte le risorse AWS Cloud Map
- AWS Cloud Map esempio di istanza di servizio
- Crea un esempio di servizio AWS Cloud Map
- Esempio di creazione di AWS Cloud Map namespace

Best practice per le policy

Le politiche basate sull'identità determinano se qualcuno può creare, accedere o eliminare AWS Cloud Map risorse nel tuo account. Queste azioni possono comportare costi aggiuntivi per l' Account AWS. Quando crei o modifichi policy basate su identità, segui queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche gestite che concedono le autorizzazioni per molti casi d'uso comuni.AWS Sono disponibili nel tuo. Account AWS Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti specifiche per i tuoi casi d'uso. Per ulteriori informazioni, consulta Policy gestite da AWSo Policy gestite da AWS per le funzioni dei processi nella Guida per l'utente IAM.
- Applica le autorizzazioni con privilegio minimo: quando imposti le autorizzazioni con le policy IAM, concedi solo le autorizzazioni richieste per eseguire un'attività. È possibile farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegi minimi. Per ulteriori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta Policy e autorizzazioni in IAM nella Guida per l'utente IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso: per limitare l'accesso a operazioni e risorse è possibile aggiungere una condizione alle tue policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio AWS CloudFormation. Per ulteriori informazioni, consulta la sezione Elementi delle policy JSON di IAM: condizione nella Guida per l'utente IAM.
- Utilizzo di IAM Access Analyzer per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali: IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano alla

sintassi della policy IAM (JSON) e alle best practice di IAM. IAM Access Analyzer offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per ulteriori informazioni, consulta Convalida delle policy per il Sistema di analisi degli accessi IAM nella Guida per l'utente IAM.

- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungi le condizioni MFA alle policy. Per ulteriori informazioni, consulta Protezione dell'accesso API con MFA nella Guida per l'utente IAM.

Per maggiori informazioni sulle best practice in IAM, consulta Best practice di sicurezza in IAM nella Guida per l'utente di IAM.

Utilizzo della console di AWS Cloud Map

Per accedere alla AWS Cloud Map console, devi disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle AWS Cloud Map risorse del tuo. Account AWS Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario consentire autorizzazioni minime per la console agli utenti che effettuano chiamate solo verso AWS CLI o l' AWS API. Al contrario, concedi l'accesso solo alle operazioni che corrispondono all'operazione API che stanno cercando di eseguire.

Per garantire che utenti e ruoli possano ancora utilizzare la AWS Cloud Map console, allega anche la policy AWS Cloud Map *ConsoleAccess* o la policy *ReadOnly* AWS gestita alle entità. Per ulteriori informazioni, consulta Aggiunta di autorizzazioni a un utente nella Guida per l'utente IAM.

AWS Cloud Map esempio di accesso alla console

Per concedere l'accesso completo alla AWS Cloud Map console, concedi le autorizzazioni nella seguente politica di autorizzazione:

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":"*"
      }
   ]
}
```

Di seguito viene descritto perché le autorizzazioni sono necessarie:

`servicediscovery:*`

Consente di eseguire tutte le AWS Cloud Map azioni.

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

Consente di AWS Cloud Map gestire le zone ospitate quando si creano ed eliminano namespace DNS pubblici e privati.

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

AWS Cloud Map Gestiamo i controlli di integrità quando includi i controlli di integrità di Amazon Route 53 quando crei un servizio.

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

Permettiamo di AWS Cloud Map gestire le zone private ospitate.

Consenti AWS Cloud Map agli utenti di visualizzare le proprie autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono cpllegate alla relativa identità utente. Questa politica include le autorizzazioni per completare questa azione sulla console o utilizzando l'API o a livello di codice. AWS CLI 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": "*"
        }
    ]
}
```

Consenti l'accesso in lettura a tutte le risorse AWS Cloud Map

La policy di autorizzazioni seguente concede all'utente l'accesso in sola lettura a tutte le risorse AWS Cloud Map :

JSON

```
{
   "Version": "2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "servicediscovery:Get*",
            "servicediscovery:List*",
            "servicediscovery:DiscoverInstances"
         ],
         "Resource":"*"
      }
   ]
}
```

AWS Cloud Map esempio di istanza di servizio

L'esempio seguente mostra una politica di autorizzazioni che concede all'utente l'autorizzazione a registrare, annullare la registrazione e scoprire le istanze del servizio. Il `Sid`, o ID dichiarazione, è facoltativo:

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": "*"
      }
   ]
}
```

La policy concede le autorizzazioni per tutte le azioni necessarie per registrare e gestire le istanze dei servizi. L'autorizzazione Route 53 è necessaria se utilizzi namespace DNS pubblici o privati perché AWS Cloud Map crea, aggiorna ed elimina i record Route 53 e i controlli di integrità quando registri e annulli la registrazione delle istanze. Il carattere jolly (*) in consente l'accesso a tutte le AWS Cloud Map istanze, ai record e ai controlli di `Resource` integrità di Route 53 di proprietà dell'account corrente. AWS

Crea un esempio di servizio AWS Cloud Map

Quando aggiungi una policy di autorizzazioni per consentire a un'identità IAM di creare un AWS Cloud Map servizio, devi specificare l'Amazon Resource Name (ARN) sia del namespace che del servizio AWS Cloud Map nel campo delle risorse. L'ARN include la regione, l'ID dell'account e l'ID dello spazio dei nomi. Poiché non saprai ancora qual è l'ID di servizio del servizio, ti consigliamo di utilizzare un jolly. Di seguito è riportato un esempio di frammento di policy.

JSON

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

```
         ],
         "Resource":[
            "arn:aws:servicediscovery:region:111122223333:namespace/ns-
p32123EXAMPLE",
            "arn:aws:servicediscovery:region:111122223333:service/*"
         ]
      }
   ]
}
```

Esempio di creazione di AWS Cloud Map namespace

La seguente politica di autorizzazione consente agli utenti di creare tutti i tipi di namespace: AWS Cloud Map

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 politiche gestite per AWS Cloud Map

Una politica AWS gestita è una politica autonoma creata e amministrata da AWS. AWS le politiche gestite sono progettate per fornire autorizzazioni per molti casi d'uso comuni, in modo da poter iniziare ad assegnare autorizzazioni a utenti, gruppi e ruoli.

Tieni presente che le policy AWS gestite potrebbero non concedere le autorizzazioni con il privilegio minimo per i tuoi casi d'uso specifici, poiché sono disponibili per tutti i clienti. AWS Ti consigliamo pertanto di ridurre ulteriormente le autorizzazioni definendo policy gestite dal cliente specifiche per i tuoi casi d'uso.

Non è possibile modificare le autorizzazioni definite nelle politiche gestite. AWS Se AWS aggiorna le autorizzazioni definite in una politica AWS gestita, l'aggiornamento ha effetto su tutte le identità principali (utenti, gruppi e ruoli) a cui è associata la politica. AWS è più probabile che aggiorni una policy AWS gestita quando ne Servizio AWS viene lanciata una nuova o quando diventano disponibili nuove operazioni API per i servizi esistenti.

Per ulteriori informazioni, consultare Policy gestite da AWSnella Guida per l'utente di IAM.

AWS politica gestita: AWSCloud MapDiscoverInstanceAccess

È possibile collegare `AWSCloudMapDiscoverInstanceAccess` alle entità IAM. Fornisce l'accesso all'API AWS Cloud Map Discovery.

Per vedere le autorizzazioni per questa policy, consulta AWSCloudMapDiscoverInstanceAccess nella Guida di riferimento sulle policy gestite da AWS .

AWS politica gestita: AWSCloud MapReadOnlyAccess

È possibile collegare `AWSCloudMapReadOnlyAccess` alle entità IAM. Garantisce l'accesso in sola lettura a tutte le azioni. AWS Cloud Map

Per vedere le autorizzazioni per questa policy, consulta AWSCloudMapReadOnlyAccess nella Guida di riferimento sulle policy gestite da AWS .

AWS politica gestita: AWSCloud MapRegisterInstanceAccess

È possibile collegare `AWSCloudMapRegisterInstanceAccess` alle entità IAM. Concede l'accesso in sola lettura ai namespace e ai servizi e concede l'autorizzazione a registrare e annullare la registrazione delle istanze del servizio.

Per vedere le autorizzazioni per questa policy, consulta AWSCloudMapRegisterInstanceAccess nella Guida di riferimento sulle policy gestite da AWS .

AWS politica gestita: AWSCloud MapFullAccess

È possibile collegare `AWSCloudMapFullAccess` alle entità IAM. Fornisce accesso completo a tutte le AWS Cloud Map azioni

Per vedere le autorizzazioni per questa policy, consulta AWSCloudMapFullAccess nella Guida di riferimento sulle policy gestite da AWS .

AWS Cloud Map aggiornamenti alle politiche AWS gestite

Visualizza i dettagli sugli aggiornamenti delle politiche AWS gestite AWS Cloud Map da quando questo servizio ha iniziato a tenere traccia di queste modifiche. Per ricevere avvisi automatici sulle modifiche, iscriviti al feed RSS nella pagina della cronologia dei AWS Cloud Map documenti.

Modifica	Descrizione	Data
AWSCloudMapDiscoverInstanceAccess, AWSCloudMapRegisterInstanceAccess, AWSCloudMapReadOnlyAccess— Aggiornamenti alle politiche esistenti.	AWS Cloud Map ha aggiornato queste politiche per fornire l'accesso alle nuove operazioni AWS Cloud Map `DiscoverInstanceRevision` API.	15 agosto 2023

AWS Cloud Map Riferimento alle autorizzazioni API

Quando configuri il controllo degli accessi e scrivi una politica di autorizzazioni da allegare a un'identità IAM (politiche basate sull'identità), puoi utilizzare il seguente elenco come riferimento. L'elenco include ogni azione AWS Cloud Map API e le azioni a cui devi concedere le autorizzazioni di accesso. È possibile specificare le azioni nel `Action` campo relativo alla politica. Per i dettagli sul valore della risorsa da specificare nel `Resource` campo o nella policy IAM, consulta Azioni, risorse e chiavi di condizione AWS Cloud Map nel Service Authorization Reference.

Puoi utilizzare le chiavi di condizione AWS Cloud Map specifiche nelle tue politiche IAM per alcune operazioni. Per ulteriori informazioni, consulta Condition keys for AWS Cloud Map nel Service Authorization Reference.

Per specificare un'operazione, utilizzare il prefisso `servicediscovery` seguito dal nome dell'operazione API, ad esempio `servicediscovery:CreatePublicDnsNamespace` e `route53:CreateHostedZone`.

Autorizzazioni necessarie per le operazioni AWS Cloud Map

CreateHttpNamespace

Autorizzazioni richieste (azione API):

- `servicediscovery:CreateHttpNamespace`

CreatePrivateDnsNamespace

Autorizzazioni richieste (azione API):

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

CreatePublicDnsNamespace

Autorizzazioni richieste (azione API):

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

CreateService

Autorizzazioni richieste (azione API): `servicediscovery:CreateService`

DeleteNamespace

Autorizzazioni richieste (azione API):

- `servicediscovery:DeleteNamespace`

DeleteService

Autorizzazioni richieste (azione API): `servicediscovery:DeleteService`

DeleteServiceAttributes

Autorizzazioni richieste (azione API): `servicediscovery:DeleteServiceAttributes`

DeregisterInstance

Autorizzazioni richieste (azione API):

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

DiscoverInstances

Autorizzazioni richieste (azione API): `servicediscovery:DiscoverInstances`

GetInstance

Autorizzazioni richieste (azione API): `servicediscovery:GetInstance`

GetInstancesHealthStatus

Autorizzazioni richieste (azione API): `servicediscovery:GetInstancesHealthStatus`

GetNamespace

Autorizzazioni richieste (azione API): `servicediscovery:GetNamespace`

GetOperation

Autorizzazioni richieste (azione API): `servicediscovery:GetOperation`

GetService

Autorizzazioni richieste (azione API): `servicediscovery:GetService`

GetServiceAttributes

Autorizzazioni richieste (azione API): `servicediscovery:GetServiceAttributes`

ListInstances

Autorizzazioni richieste (azione API): `servicediscovery:ListInstances`

ListNamespaces

Autorizzazioni richieste (azione API): `servicediscovery:ListNamespaces`

ListOperations

Autorizzazioni richieste (azione API): `servicediscovery:ListOperations`

ListServices

Autorizzazioni richieste (azione API): `servicediscovery:ListServices`

ListTagsForResource

Autorizzazioni richieste (azione API): `servicediscovery:ListTagsForResource`

RegisterInstance

Autorizzazioni richieste (azione API):

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

TagResource

Autorizzazioni richieste (azione API): `servicediscovery:TagResource`

UntagResource

Autorizzazioni richieste (azione API): `servicediscovery:UntagResource`

UpdateHttpNamespace

Autorizzazioni richieste (azione API): `servicediscovery:UpdateHttpNamespace`

UpdateInstanceCustomHealthStatus

Autorizzazioni richieste (azione API): `servicediscovery:UpdateInstanceCustomHealthStatus`

UpdatePrivateDnsNamespace

Autorizzazioni richieste (azione API):

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

UpdatePublicDnsNamespace

Autorizzazioni richieste (azione API):

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

UpdateService

Autorizzazioni richieste (azione API):

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

UpdateServiceAttributes

Autorizzazioni richieste (azione API): `servicediscovery:UpdateServiceAttributes`

Risoluzione dei problemi di AWS Cloud Map identità e accesso

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con AWS Cloud Map IAM.

Argomenti

- Non sono autorizzato a eseguire alcuna azione in AWS Cloud Map
- Non sono autorizzato a eseguire iam: PassRole
- Voglio consentire a persone esterne a me di accedere Account AWS alle mie AWS Cloud Map risorse

Non sono autorizzato a eseguire alcuna azione in AWS Cloud Map

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa *`my-example-widget`* fittizia ma non dispone di autorizzazioni `servicediscovery:`*`GetWidget`* fittizie.

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

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa *`my-example-widget`* utilizzando l'azione `servicediscovery:`*`GetWidget`*.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Non sono autorizzato a eseguire iam: PassRole

Se ricevi un errore che indica che non sei autorizzato a eseguire l'operazione `iam:PassRole`, le tue policy devono essere aggiornate per poter passare un ruolo a AWS Cloud Map.

Alcuni Servizi AWS consentono di passare un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

L'errore di esempio seguente si verifica quando un utente IAM denominato `marymajor` cerca di utilizzare la console per eseguire un'operazione in AWS Cloud Map. Tuttavia, l'operazione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

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

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

Voglio consentire a persone esterne a me di accedere Account AWS alle mie AWS Cloud Map risorse

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo.

Per i servizi che supportano politiche basate sulle risorse o liste di controllo degli accessi (ACLs), puoi utilizzare tali politiche per concedere alle persone l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se AWS Cloud Map supporta queste funzionalità, consulta. Come AWS Cloud Map funziona con IAM
- Per scoprire come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta Fornire l'accesso a soggetti Account AWS di proprietà di terze parti nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta Fornire l'accesso a utenti autenticati esternamente (Federazione delle identità) nella Guida per l'utente IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consulta Accesso a risorse multi-account in IAM nella Guida per l'utente IAM.

Convalida della conformità per AWS Cloud Map

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione Scope by Compliance Program Servizi AWS e scegli il programma di conformità che ti interessa. Per informazioni generali, consulta Programmi di AWS conformità Programmi di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta Scaricamento dei report in AWS Artifact .

La vostra responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- Governance e conformità per la sicurezza: queste guide all'implementazione di soluzioni illustrano considerazioni relative all'architettura e i passaggi per implementare le funzionalità di sicurezza e conformità.
- Riferimenti sui servizi conformi ai requisiti HIPAA: elenca i servizi HIPAA idonei. Non tutti Servizi AWS sono idonei alla normativa HIPAA.

- AWS Risorse per la per la conformità: questa raccolta di cartelle di lavoro e guide potrebbe essere valida per il tuo settore e la tua località.
- AWS Guide alla conformità dei clienti: comprendi il modello di responsabilità condivisa attraverso la lente della conformità. Le guide riassumono le migliori pratiche per la protezione Servizi AWS e mappano le linee guida per i controlli di sicurezza su più framework (tra cui il National Institute of Standards and Technology (NIST), il Payment Card Industry Security Standards Council (PCI) e l'International Organization for Standardization (ISO)).
- Valutazione delle risorse con regole nella Guida per gli AWS Config sviluppatori: il AWS Config servizio valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida e alle normative del settore.
- AWS Security Hub— Ciò Servizio AWS fornisce una visione completa dello stato di sicurezza interno. AWS La Centrale di sicurezza utilizza i controlli di sicurezza per valutare le risorse AWS e verificare la conformità agli standard e alle best practice del settore della sicurezza. Per un elenco dei servizi e dei controlli supportati, consulta la pagina Documentazione di riferimento sui controlli della Centrale di sicurezza.
- Amazon GuardDuty: Servizio AWS rileva potenziali minacce ai tuoi carichi di lavoro Account AWS, ai contenitori e ai dati monitorando l'ambiente alla ricerca di attività sospette e dannose. GuardDuty può aiutarti a soddisfare vari requisiti di conformità, come lo standard PCI DSS, soddisfacendo i requisiti di rilevamento delle intrusioni imposti da determinati framework di conformità.
- AWS Audit Manager— Ciò Servizio AWS consente di verificare continuamente l' AWS utilizzo per semplificare la gestione del rischio e la conformità alle normative e agli standard di settore.

Resilienza in AWS Cloud Map

L'infrastruttura AWS globale è costruita attorno a AWS regioni e zone di disponibilità. AWS Le regioni forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità, è possibile progettare e gestire applicazioni e database che eseguono il failover automatico tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture tradizionali a data center singolo o multiplo.

AWS Cloud Map è principalmente un servizio globale. Tuttavia, puoi utilizzarli AWS Cloud Map per creare controlli di integrità di Route 53 che controllano lo stato delle risorse in regioni specifiche, come le EC2 istanze Amazon e i sistemi di bilanciamento del carico Elastic Load Balancing.

Per ulteriori informazioni su AWS regioni e zone di disponibilità, consulta Global Infrastructure.AWS

Sicurezza dell'infrastruttura in AWS Cloud Map

In quanto servizio gestito, AWS Cloud Map è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi AWS di sicurezza e su come AWS protegge l'infrastruttura, consulta AWS Cloud Security. Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Utilizzate chiamate API AWS pubblicate per accedere AWS Cloud Map attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. O puoi utilizzare AWS Security Token Service (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

Puoi migliorare il livello di sicurezza del tuo VPC AWS Cloud Map configurando l'uso di un endpoint VPC di interfaccia. Per ulteriori informazioni, consulta Accesso AWS Cloud Map tramite un endpoint di interfaccia ()AWS PrivateLink.

Accesso AWS Cloud Map tramite un endpoint di interfaccia ()AWS PrivateLink

Puoi usarlo AWS PrivateLink per creare una connessione privata tra il tuo VPC e. AWS Cloud Map Puoi accedere AWS Cloud Map come se fosse nel tuo VPC, senza l'uso di un gateway Internet, un dispositivo NAT, una connessione VPN o una connessione. AWS Direct Connect Le istanze del tuo VPC non necessitano di indirizzi IP pubblici per accedervi. AWS Cloud Map

Stabilisci questa connessione privata creando un endpoint di interfaccia attivato da AWS PrivateLink. In ciascuna sottorete viene creata un'interfaccia di rete endpoint da abilitare per l'endpoint di interfaccia. Queste sono interfacce di rete gestite dal richiedente che fungono da punto di ingresso per il traffico destinato a AWS Cloud Map.

Per ulteriori informazioni, consulta la sezione Accesso a Servizi AWS tramite AWS PrivateLink nella Guida di AWS PrivateLink .

Considerazioni per AWS Cloud Map

Prima di configurare un endpoint di interfaccia per AWS Cloud Map, consulta le considerazioni nella Guida.AWS PrivateLink

Se il tuo Amazon VPC non dispone di un gateway Internet e le tue attività utilizzano il driver di `awslogs` registro per inviare informazioni di log a CloudWatch Logs, devi creare un endpoint VPC di interfaccia per Logs. CloudWatch Per ulteriori informazioni, consulta Using CloudWatch Logs with Interface VPC Endpoints nella CloudWatch Amazon Logs User Guide.

Gli endpoint VPC non supportano AWS le richieste interregionali. Assicurati di creare l'endpoint nella stessa regione in cui prevedi di inviare le chiamate API a AWS Cloud Map.

Gli endpoint VPC supportano solo il DNS fornito da Amazon tramite Amazon Route 53. Se si desidera utilizzare il proprio DNS, è possibile usare l'inoltro condizionale sul DNS. Per ulteriori informazioni, consulta DHCP Options Sets nella Amazon VPC User Guide.

Il gruppo di sicurezza collegato all'endpoint VPC deve consentire le connessioni in entrata sulla porta 443 dalla sottorete privata di Amazon VPC.

Crea un endpoint di interfaccia per AWS Cloud Map

Puoi creare un endpoint di interfaccia per AWS Cloud Map utilizzare la console Amazon VPC o AWS Command Line Interface ().AWS CLI Per ulteriori informazioni, consulta la sezione Creazione di un endpoint di interfaccia nella Guida per l'utente di AWS PrivateLink .

Crea un endpoint di interfaccia per AWS Cloud Map utilizzare i seguenti nomi di servizio:

> **Note**
>
> `DiscoverInstances`L'API non sarà disponibile su questi due endpoint.

```
com.amazonaws.region.servicediscovery
```

```
com.amazonaws.region.servicediscovery-fips
```

Crea un endpoint di interfaccia per il piano AWS Cloud Map dati per accedere all'`DiscoverInstances`API utilizzando i seguenti nomi di servizio:

```
com.amazonaws.region.data-servicediscovery
```

```
com.amazonaws.region.data-servicediscovery-fips
```

> **Note**
>
> È necessario disabilitare l'inserimento del prefisso dell'host quando si effettuano chiamate `DiscoverInstances` con i nomi DNS VPCE regionali o zonali per gli endpoint del piano dati. Aggiungi AWS CLI e AWS SDKs anteponi all'endpoint del servizio vari prefissi host quando chiami ogni operazione API, il che produce URL non validi quando specifichi un endpoint VPC.

Se abiliti il DNS privato per l'endpoint di interfaccia, puoi effettuare richieste API utilizzando il nome DNS regionale predefinito. AWS Cloud Map Ad esempio `servicediscovery.us-east-1.amazonaws.com`.

La AWS PrivateLink connessione VPCE è supportata in qualsiasi regione in cui AWS Cloud Map è supportata; tuttavia, un cliente deve verificare quali zone di disponibilità supportano VPCE prima di definire un endpoint. Per scoprire quali zone di disponibilità sono supportate con gli endpoint VPC di interfaccia in una regione, usa il describe-vpc-endpoint-services comando o usa il. AWS Management Console Ad esempio, i seguenti comandi restituiscono le zone di disponibilità in cui è possibile implementare un endpoint VPC di AWS Cloud Map interfaccia all'interno della regione Stati Uniti orientali (Ohio):

```
aws --region us-east-2 ec2 describe-vpc-endpoint-services --query 'ServiceDetails[?
ServiceName==`com.amazonaws.us-east-2.servicediscovery`].AvailabilityZones[]'
```

Monitoraggio AWS Cloud Map

Il monitoraggio è importante per garantire l'affidabilità, la disponibilità e le prestazioni delle soluzioni AWS . È necessario raccogliere i dati di monitoraggio da tutte le parti della AWS soluzione in modo da poter eseguire più facilmente il debug di un errore multipunto, se si verifica. Tuttavia, prima di iniziare il monitoraggio è opportuno creare un piano di monitoraggio che includa le risposte alle seguenti domande:

- Quali sono gli obiettivi del monitoraggio?
- Di quali risorse si intende eseguire il monitoraggio?
- Con quale frequenza sarà eseguito il monitoraggio di queste risorse?
- Quali strumenti di monitoraggio verranno utilizzati?
- Chi eseguirà i processi di monitoraggio?
- Chi deve ricevere una notifica quando si verifica un problema?

Argomenti

Registra le chiamate AWS Cloud Map API utilizzando AWS CloudTrail

AWS Cloud Map è integrato con AWS CloudTrail, un servizio che fornisce una registrazione delle azioni intraprese da un utente, ruolo o un. Servizio AWS CloudTrail acquisisce tutte le chiamate API AWS Cloud Map come eventi. Le chiamate acquisite includono chiamate dalla AWS Cloud Map console e chiamate di codice alle operazioni AWS Cloud Map API. Utilizzando le informazioni raccolte da CloudTrail, è possibile determinare a quale richiesta è stata effettuata AWS Cloud Map, l'indirizzo IP da cui è stata effettuata la richiesta, quando è stata effettuata e ulteriori dettagli.

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni di identità consentono di determinare quanto segue:

- Se la richiesta è stata effettuata con le credenziali utente root o utente.
- Se la richiesta è stata effettuata per conto di un utente del Centro identità IAM.

- Se la richiesta è stata effettuata con le credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Se la richiesta è stata effettuata da un altro Servizio AWS.

CloudTrail è attivo nel tuo account Account AWS quando crei l'account e hai automaticamente accesso alla cronologia degli CloudTrail eventi. La cronologia CloudTrail degli eventi fornisce un record visualizzabile, ricercabile, scaricabile e immutabile degli ultimi 90 giorni di eventi di gestione registrati in un. Regione AWSPer ulteriori informazioni, consulta Lavorare con la cronologia degli CloudTrail eventi nella Guida per l'utente.AWS CloudTrail Non sono CloudTrail previsti costi per la visualizzazione della cronologia degli eventi.

Per una registrazione continua degli eventi degli Account AWS ultimi 90 giorni, crea un trail o un data store di eventi CloudTrailLake.

CloudTrail sentieri

Un trail consente di CloudTrail inviare file di log a un bucket Amazon S3. Tutti i percorsi creati utilizzando il AWS Management Console sono multiregionali. È possibile creare un trail per una singola Regione o per più Regioni tramite AWS CLI. La creazione di un percorso multiregionale è consigliata in quanto consente di registrare l'intera attività del proprio Regioni AWS account. Se si crea un trail per una singola Regione, è possibile visualizzare solo gli eventi registrati nella Regione AWS del trail. Per ulteriori informazioni sui trail, consulta Creating a trail for your Account AWS e Creating a trail for an organization nella Guida per l'utente di AWS CloudTrail .

Puoi inviare gratuitamente una copia dei tuoi eventi di gestione in corso al tuo bucket Amazon S3 CloudTrail creando un percorso, tuttavia ci sono costi di storage di Amazon S3. Per ulteriori informazioni sui CloudTrail prezzi, consulta la pagina Prezzi.AWS CloudTrail Per informazioni sui prezzi di Amazon S3, consulta Prezzi di Amazon S3.

CloudTrail Archivi di dati sugli eventi di Lake

CloudTrail Lake ti consente di eseguire query basate su SQL sui tuoi eventi. CloudTrail Lake converte gli eventi esistenti in formato JSON basato su righe in formato Apache ORC. ORC è un formato di archiviazione a colonne ottimizzato per il recupero rapido dei dati. Gli eventi vengono aggregati in archivi di dati degli eventi, che sono raccolte di eventi immutabili basate sui criteri selezionati applicando i selettori di eventi avanzati. I selettori applicati a un archivio di dati degli eventi controllano quali eventi persistono e sono disponibili per l'esecuzione della query. Per ulteriori informazioni su CloudTrail Lake, consulta Working with AWS CloudTrail Lake nella Guida per l'utente.AWS CloudTrail

CloudTrail Gli archivi e le richieste di dati sugli eventi di Lake comportano dei costi. Quando crei un datastore di eventi, scegli l'opzione di prezzo da utilizzare per tale datastore. L'opzione di prezzo determina il costo per l'importazione e l'archiviazione degli eventi, nonché il periodo di conservazione predefinito e quello massimo per il datastore di eventi. Per ulteriori informazioni sui CloudTrail prezzi, consulta la sezione Prezzi.AWS CloudTrail

AWS Cloud Map eventi relativi ai dati in CloudTrail

Gli eventi relativi ai dati forniscono informazioni sulle operazioni eseguite sulle risorse su o all'interno di una risorsa (ad esempio, la scoperta di un'istanza registrata in un namespace). Queste operazioni sono definite anche operazioni del piano dei dati. Gli eventi di dati sono spesso attività che interessano volumi elevati di dati. Per impostazione predefinita, CloudTrail non registra gli eventi relativi ai dati. La cronologia CloudTrail degli eventi non registra gli eventi relativi ai dati.

Per gli eventi di dati sono previsti costi aggiuntivi. Per ulteriori informazioni sui CloudTrail prezzi, consulta la sezione AWS CloudTrail Prezzi.

Puoi registrare gli eventi relativi ai dati per i tipi di AWS Cloud Map risorse utilizzando la CloudTrail console o AWS CLI le operazioni CloudTrail dell'API. Per ulteriori informazioni su come registrare gli eventi di dati, consulta Registrazione di eventi di dati con AWS Management Console e Registrazione di eventi di dati con AWS Command Line Interface nella Guida all'utente AWS CloudTrail .

La tabella seguente elenca i tipi di AWS Cloud Map risorse per i quali è possibile registrare gli eventi relativi ai dati. La colonna Data event type (console) mostra il valore da scegliere dall'elenco Data event type (console) sulla CloudTrail console. La colonna del valore resources.type mostra il `resources.type` valore da specificare durante la configurazione dei selettori di eventi avanzati utilizzando o. AWS CLI CloudTrail APIs La CloudTrail colonna Dati APIs registrati mostra le chiamate API registrate per il tipo di risorsa. CloudTrail

Tipo di evento di dati (console)	valore resources.type	Dati registrati APIs su CloudTrail
AwsApiCall	`AWS::ServiceDiscovery::Namespace`	• DiscoverInstances • DiscoverInstancesRevision
AwsApiCall	`AWS::ServiceDiscovery::Service`	• DiscoverInstances

Tipo di evento di dati (console)	valore resources.type	Dati registrati APIs su CloudTrail
		• DiscoverInstancesRevision

È possibile configurare selettori di eventi avanzati per filtrare i campi `eventName`, `readOnly` e `resources.`ARN per registrare solo gli eventi importanti per l'utente. Per ulteriori informazioni su questi campi, vedere AdvancedFieldSelector nel documento di riferimento delle API AWS CloudTrail

L'esempio seguente mostra come configurare i selettori di eventi avanzati per registrare tutti gli eventi AWS Cloud Map relativi ai dati.

```
"AdvancedEventSelectors":
[
    {
        "Name": "Log all AWS Cloud Map data events",
        "FieldSelectors": [
            { "Field": "eventCategory", "Equals": ["Data"] },
            { "Field": "resources.type", "Equals":
 ["AWS::ServiceDiscovery::Namespace"] }
        ]
    }
]
```

AWS Cloud Map eventi di gestione in CloudTrail

Gli eventi di gestione forniscono informazioni sulle operazioni di gestione eseguite sulle risorse dell'azienda Account AWS. Queste operazioni sono definite anche operazioni del piano di controllo (control-plane). Per impostazione predefinita, CloudTrail registra gli eventi di gestione.

AWS Cloud Map registra tutte le operazioni AWS Cloud Map del piano di controllo come eventi di gestione. Per un elenco delle operazioni del piano di AWS Cloud Map controllo a cui si AWS Cloud Map effettua l'accesso CloudTrail, consulta l'AWS Cloud Map API Reference.

AWS Cloud Map esempi di eventi

Un evento rappresenta una singola richiesta proveniente da qualsiasi fonte e include informazioni sull'operazione API richiesta, la data e l'ora dell'operazione, i parametri della richiesta e così via. CloudTrail i file di registro non sono una traccia ordinata dello stack delle chiamate API pubbliche, quindi gli eventi non vengono visualizzati in un ordine specifico.

L'esempio seguente mostra un evento CloudTrail di gestione che dimostra l'`CreateHTTPNamespace`operazione.


```
{
    "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"
}
```

L'esempio seguente mostra un evento di CloudTrail dati che dimostra l'`DiscoverInstances`operazione.

```
{
            "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"
        }
```


Per informazioni sul contenuto dei CloudTrail record, consultate il contenuto dei CloudTrail record nella Guida per l'AWS CloudTrail utente.

Taggare le tue risorse AWS Cloud Map

Un tag è un'etichetta che si assegna a una risorsa. AWS Ogni tag è composto da una chiave e da un valore opzionale, entrambi personalizzabili.

I tag consentono di classificare le AWS risorse in base, ad esempio, allo scopo, al proprietario o all'ambiente. Se disponi di un numero elevato di risorse, puoi individuare rapidamente una risorsa specifica in base ai tag assegnati. Ad esempio, puoi definire un set di tag per AWS Cloud Map i tuoi servizi per aiutarti a tenere traccia del proprietario e del livello di stack di ciascun servizio. Consigliamo di definire un set coerente di chiavi di tag per ciascun tipo di risorsa.

I tag non vengono assegnati in automatico alle risorse. Dopo aver aggiunto un tag, puoi modificarne le chiavi e i valori oppure rimuovere i tag da una risorsa in qualsiasi momento. Se elimini una risorsa, verranno eliminati anche tutti i tag a essa associati.

I tag non hanno alcun significato semantico AWS Cloud Map e vengono interpretati rigorosamente come una stringa di caratteri. Puoi impostare il valore di un tag su una stringa vuota, ma non su null. Se aggiungi un tag con la stessa chiave di un tag esistente a una risorsa specifica, il nuovo valore sovrascrive quello precedente.

Puoi lavorare con i tag utilizzando l' AWS Management Console AWS CLI, la e l' AWS Cloud Map API.

Se utilizzi AWS Identity and Access Management (IAM), puoi controllare quali utenti del tuo AWS account sono autorizzati a creare, modificare o eliminare i tag.

Assegnazione di tag alle risorse

Puoi taggare AWS Cloud Map namespace e servizi nuovi o esistenti.

Se utilizzi la AWS Cloud Map console, puoi applicare i tag alle nuove risorse al momento della creazione o alle risorse esistenti in qualsiasi momento utilizzando la scheda Tag nella pagina delle risorse pertinente.

Se utilizzi l' AWS Cloud Map API, l' AWS CLI o un AWS SDK, puoi applicare i tag a nuove risorse utilizzando il `tags` parametro sull'azione API pertinente o alle risorse esistenti utilizzando l'azione TagResourceAPI. Per ulteriori informazioni, consulta TagResource.

Alcune operazioni per la creazione di risorse ti consentono di specificare tag per una risorsa durante la sua creazione. Se i tag non possono essere applicati durante la creazione della risorsa, il processo di creazione della risorsa avrà esito negativo. In questo modo, le risorse a cui desideri applicare tag al momento della creazione vengono create con tag specifici o non vengono create affatto. Se aggiungi tag alle risorse al momento della creazione, non devi eseguire script di tagging personalizzati dopo la creazione delle risorse.

La tabella seguente descrive le AWS Cloud Map risorse che possono essere taggate e le risorse che possono essere taggate al momento della creazione.

Supporto per AWS Cloud Map l'etichettatura delle risorse

Risorsa	Supporta tag	Supporta la propagazione di tag	Supporta l'etichettatura alla creazione (AWS Cloud Map API AWS CLI, AWS SDK)
AWS Cloud Map namespace	Sì	No. I tag dello spazio dei nomi non si propagano a nessun'altra risorsa associata allo spazio dei nomi.	Sì
AWS Cloud Map servizi	Sì	No. I tag di servizio non si propagano ad altre risorse associate al servizio.	Sì

Restrizioni

Ai tag si applicano le seguenti limitazioni di base:

- Numero massimo di tag per ogni risorsa: 50
- Per ciascuna risorsa, ogni chiave del tag deve essere univoca e ogni chiave del tag può avere un solo valore.
- Lunghezza massima della chiave: 128 caratteri Unicode in formato UTF-8

- Lunghezza massima del valore: 256 caratteri Unicode in formato UTF-8
- Se il tuo schema di etichettatura viene utilizzato su più AWS servizi e risorse, ricorda che altri servizi potrebbero avere restrizioni sui caratteri consentiti. I caratteri generalmente consentiti sono: lettere, numeri, spazi rappresentabili in formato UTF-8 e i seguenti caratteri speciali: + - = . _ : / @.
- I valori e le chiavi dei tag rispettano la distinzione tra maiuscole e minuscole.
- Non utilizzare alcuna combinazione maiuscola o minuscola `aws:AWS:`, ad esempio un prefisso per chiavi o valori, poiché è riservata all'uso. AWS Non è possibile modificare né eliminare le chiavi o i valori di tag con tale prefisso. I tag con questo prefisso non rientrano nel tuo limite. tags-per-resource

Aggiornamento dei tag per le risorse AWS Cloud Map

Utilizza i seguenti AWS CLI comandi o operazioni AWS Cloud Map API per aggiungere, aggiornare, elencare ed eliminare i tag delle tue risorse.

Supporto per AWS Cloud Map l'etichettatura delle risorse

Attività	Azione API	AWS CLI	AWS Tools for Windows PowerShell
Aggiungere sovrascrivere uno o più tag.	TagResource	tag-resource	Aggiungi tag SDResource
Eliminare uno o più tag.	UntagResource	untag-resource	Rimuovi- SDResource Tag
Elencazione dei tag associati a una risorsa	ListTagsForResource	list-tags-for-resource	Ottieni- SDResource Tag

I seguenti esempi mostrano come aggiungere o rimuovere tag alle o dalle risorse utilizzando la AWS CLI.

Esempio 1: tag a una risorsa esistente

Il comando seguente applica un tag a una risorsa esistente.

```
aws servicediscovery tag-resource --resource-arn resource_ARN --tags team=devs
```

Esempio 2: rimozione di un tag da una risorsa esistente

Il comando seguente elimina un tag da una risorsa esistente.

```
aws servicediscovery untag-resource --resource-arn resource_ARN --tag-keys tag_key
```

Esempio 3: elencazione dei tag di una risorsa

Il comando seguente elenca i tag associati a una risorsa esistente.

```
aws servicediscovery list-tags-for-resource --resource-arn resource_ARN
```

Alcune operazioni per la creazione di risorse ti consentono di specificare tag quando crei le risorse. Le seguenti operazioni supportano il tagging in fase di creazione.

Attività	Azione API	AWS CLI	AWS Tools for Windows PowerShell
Creare uno spazio dei nomi HTTP	CreateHttpNamespace	create-http-namespace	Nuovo SDHttp spazio dei nomi
Creare uno spazio dei nomi privato basato su DNS	CreatePrivateDnsNamespace	create-private-dns-namespace	Nuovo- SDPrivate DnsNamespace
Creare uno spazio dei nomi pubblico basato su DNS	CreatePublicDnsNamespace	create-public-dns-namespace	Nuovo- SDPublic DnsNamespace
Creazione di un servizio	CreateService	create-service	Nuovo- SDService

AWS Cloud Map quote di servizio

AWS Cloud Map le risorse sono soggette alle seguenti quote di servizio a livello di account. Ogni quota elencata si applica a ogni AWS regione in cui si creano risorse. AWS Cloud Map

Nome	Predefinita	Adattabile	Descrizione
Attributi personalizzati per istanza	Ogni regione supportata: 30	No	Il numero massimo di attributi personalizzati che puoi specificare al momento della registrazione di un'istanza.
DiscoverInstances frequenza di interruzione delle operazioni per account	Ogni regione supportata: 2.000	Sì	La frequenza di burst massima per l' DiscoverInstances operazione di chiamata da un singolo account.
DiscoverInstances operatività per account (tasso costante)	Ogni regione supportata: 1.000	Sì	La tariffa fissa massima per le DiscoverInstances operazioni di chiamata da un singolo account.
DiscoverInstancesRevision tariffa operativa per conto	Ogni regione supportata: 3.000	Sì	La velocità massima per le DiscoverInstancesRevision operazioni di chiamata da un singolo account.
Istanze per namespace	Ogni regione supportata: 2.000	Sì	Il numero massimo di istanze di servizio che puoi registrare utilizzando lo stesso spazio dei nomi.
Istanze per servizio	Ogni regione supportata: 1.000	No	Il numero massimo di istanze che puoi registrar

Nome	Predefinita	Adatta e	Descrizione
			e utilizzando in una regione utilizzando lo stesso servizio.
Spazio dei nomi per regione	Ogni Regione supportata: 50	Sì	Il numero massimo di spazi dei nomi che puoi creare per regione.

* Quando crei un namespace, creiamo automaticamente una zona ospitata su Amazon Route 53. Questa zona ospitata viene conteggiata sulla quota del numero di zone ospitate che puoi creare con un AWS account. Per ulteriori informazioni, consulta Quotas on hosted zones nella Amazon Route 53 Developer Guide.

** L'aumento delle istanze per i namespace DNS AWS Cloud Map richiede un aumento del limite di record per zona ospitata Route 53, che comporta costi aggiuntivi.

Gestione delle quote di servizio AWS Cloud Map

AWS Cloud Map si è integrato con Service Quotas, un AWS servizio che consente di visualizzare e gestire le quote da una posizione centrale. Per ulteriori informazioni, consulta Cos'è Service Quotas? nella Guida per l'utente di Service Quotas.

Service Quotas semplifica la ricerca del valore delle quote di AWS Cloud Map servizio.

AWS Management Console

Per visualizzare le quote AWS Cloud Map di servizio utilizzando il AWS Management Console

1. Apri la console Service Quotas all'indirizzo https://console.aws.amazon.com/servicequotas/.
2. Nel pannello di navigazione, scegli Servizi AWS .
3. Dall'elenco di servizi AWS , cerca e seleziona AWS Cloud Map.
4. Nell'elenco delle quote di servizio per AWS Cloud Map, è possibile visualizzare il nome della quota di servizio, il valore applicato (se disponibile), la quota AWS predefinita e se il valore della quota è regolabile.

Per visualizzare informazioni aggiuntive su una quota di servizio, ad esempio la descrizione, scegli il nome della quota per visualizzare i dettagli della quota.

5. (Facoltativo) Per richiedere un aumento della quota, seleziona la quota che desideri aumentare e scegli Richiedi aumento a livello di account.

Per lavorare meglio con le quote di servizio, AWS Management Console consulta la Service Quotas User Guide.

AWS CLI

Per visualizzare le quote AWS Cloud Map di servizio utilizzando il AWS CLI

Eseguire il comando seguente per visualizzare le AWS Cloud Map quote predefinite.

```
aws service-quotas list-aws-default-service-quotas \
    --query 'Quotas[*].
{Adjustable:Adjustable,Name:QuotaName,Value:Value,Code:QuotaCode}' \
    --service-code AWSCloudMap \
    --output table
```

Esegui il comando seguente per visualizzare le AWS Cloud Map quote applicate.

```
aws service-quotas list-service-quotas \
    --service-code AWSCloudMap
```

Per ulteriori informazioni sull'utilizzo delle quote di servizio utilizzando il AWS CLI, vedere Service Quotas Command AWS CLI Reference. Per richiedere un aumento delle quote, consultare il comando `request-service-quota-increase` nella Documentazione di riferimento sui comandi AWS CLI.

Gestisci la AWS Cloud Map DiscoverInstances limitazione delle richieste API

AWS Cloud Map limita le richieste DiscoverInstancesAPI per ogni AWS account in base alla regione. Il throttling aiuta a migliorare le prestazioni del servizio e a garantire un utilizzo equo per tutti i clienti. AWS Cloud Map La limitazione garantisce che le chiamate all' AWS Cloud Map DiscoverInstancesAPI non superino le quote massime consentite DiscoverInstancesper le richieste

API. DiscoverInstances Le chiamate API provenienti da una delle seguenti fonti sono soggette alle quote di richiesta:

- Un'applicazione di terze parti
- Uno strumento da riga di comando
- La AWS Cloud Map console

Se superi una quota di limitazione dell'API, viene visualizzato il codice di `RequestLimitExceeded` errore. Per ulteriori informazioni, consulta the section called “Limitazione del tasso di richiesta”.

Come viene applicata la limitazione

AWS Cloud Map utilizza l'algoritmo token bucket per implementare il throttling delle API. Con questo algoritmo, il tuo account dispone di un bucket che contiene un numero specifico di token. Il numero di token nel bucket rappresenta la tua quota di throttling in un dato secondo. Esiste un bucket per una singola regione e si applica a tutti gli endpoint della regione.

Limitazione del tasso di richiesta

La limitazione limita il numero di richieste DiscoverInstancesAPI che è possibile effettuare. Ogni richiesta rimuove un token dal bucket. Ad esempio, la dimensione del bucket per l'operazione DiscoverInstancesAPI è di 2.000 token, quindi puoi effettuare fino a 2.000 DiscoverInstancesrichieste in un secondo. Se superi le 2.000 richieste in un secondo, vieni limitato e le richieste rimanenti entro quel secondo hanno esito negativo.

I secchi si ricaricano automaticamente a una velocità prestabilita. Se il bucket non è al massimo, viene aggiunto un determinato numero di token ogni secondo finché il bucket non raggiunge la capacità. Se il bucket è al massimo della capacità quando arrivano i token di ricarica, questi token vengono scartati. La dimensione del bucket per il funzionamento dell'DiscoverInstancesAPI è di 2.000 token e la frequenza di ricarica è di 1.000 token al secondo. Se effettui 2.000 richieste DiscoverInstancesAPI in un secondo, il bucket viene immediatamente ridotto a zero (0) token. Il bucket viene quindi ricaricato fino a 1.000 token al secondo fino a raggiungere la capacità massima di 2.000 token.

Puoi utilizzare i token man mano che vengono aggiunti al bucket. Non è necessario attendere che il bucket raggiunga la capacità massima prima di effettuare richieste API. Se esaurisci il bucket effettuando 2.000 richieste DiscoverInstancesAPI in un secondo, puoi comunque effettuare fino a 1.000 richieste DiscoverInstancesAPI ogni secondo per tutto il tempo necessario. Ciò significa che

puoi utilizzare immediatamente i token di ricarica non appena vengono aggiunti al tuo bucket. Il bucket inizia a ricaricarsi fino alla capacità massima solo quando si effettuano meno richieste API ogni secondo rispetto alla frequenza di ricarica.

Tentativi ripetuti o elaborazione batch

Se una richiesta API fallisce, l'applicazione potrebbe dover riprovare la richiesta. Per ridurre il numero di richieste API, utilizzate un intervallo di sospensione appropriato tra le richieste successive. Per ottimizzare i risultati, utilizzare un intervallo di attesa incrementale o variabile.

Calcolo dell'intervallo di attesa

Quando è necessario eseguire il polling o rieseguire una richiesta API, è consigliato l'uso di un algoritmo di backoff esponenziale per calcolare l'intervallo di tempo di attesa tra le chiamate API. Utilizzando tempi di attesa progressivamente più lunghi tra un tentativo e l'altro per le risposte di errore consecutive, è possibile ridurre il numero di richieste non riuscite. Per ulteriori informazioni ed esempi di implementazione di questo algoritmo, consulta Retry Behavior nella AWS SDKs and Tools Reference Guide.

Regolazione delle quote di limitazione delle API

Puoi richiedere un aumento delle quote di limitazione delle API per il tuo account. AWS Per richiedere un adeguamento delle quote, contatta il centro Supporto AWS.

Cronologia dei documenti per AWS Cloud Map

La tabella seguente descrive i principali aggiornamenti e le nuove funzionalità della AWS Cloud Map Developer Guide. Inoltre, aggiorniamo frequentemente la documentazione tenendo conto dei feedback ricevuti.

Modifica	Descrizione	Data
AWS Cloud Map attributi del servizio	È ora possibile specificare gli attributi a livello di servizio per evitare la duplicazione degli attributi tra le istanze registrate su un servizio. È possibile utilizzare questi attributi per il routing del traffico complesso, l'impostazione di valori di timeout e di nuovo tentativo e per il coordinamento tra servizi e integrazioni esterne.	13 dicembre 2024
Tutorial aggiunti	Due tutorial che mostrano i casi d'uso più comuni per l'utilizzo di add. AWS Cloud Map	27 marzo 2024
CloudTrail documentazione di integrazione aggiornata	La documentazione che descrive l' AWS Cloud Map integrazione con CloudTrail to log API activity è stata aggiornata.	20 marzo 2024
Aggiornamenti delle politiche gestiti	`AWSCloudMapDiscoverInstanceAccess` e `AWSCloudMapRegisterInstanceAccess` le `AWSCloudMapReadOnl`	20 settembre 2023

	`yAccess` politiche sono state aggiornate.	
Cloud Map e AWS PrivateLink	Ora puoi usare an AWS PrivateLink per creare una connessione privata tra il tuo VPC e. AWS Cloud Map	15 settembre 2023
Aggiornamento della policy gestita	`AWSCloudMapDiscoverInstanceAccess` la politica è stata aggiornata.	15 agosto 2023
AWS SDK per Python	Aggiunti esempi di riga di comando in Python.	13 settembre 2022
IPv6 supporto	Gli endpoint API sono ora disponibili `IPv6` solo nelle reti.	28 gennaio 2022
Identificazione delle istanze di servizio	AWS Cloud Map ha aggiunto il supporto per la creazione di servizi in uno spazio dei nomi che supporta le query DNS individuabili solo utilizzando l'operazione DiscoverInstancesAPI e non utilizzando le query DNS.	24 marzo 2021
Aggiunta di tag alle risorse	AWS Cloud Map ha aggiunto il supporto per l'aggiunta di tag di metadati ai namespace e ai servizi utilizzando. AWS Management Console	8 febbraio 2021
Aggiunta di tag alle risorse	AWS Cloud Map ha aggiunto il supporto per l'aggiunta di tag di metadati ai namespace e ai servizi utilizzando and. AWS CLI APIs	22 giugno 2020

Versione iniziale	Questa è la prima versione della AWS Cloud Map Developer Guide.	28 novembre 2018

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.