aws

개발자 안내서

AWS Cloud Map

AWS Cloud Map: 개발자 안내서

Table of Contents

AWS Cloud Map이란 무엇입니까?

AWS Cloud Map은 완전 관리형 서비스로, 애플리케이션에서 사용하는 백엔드 서비스와 리소스의 맵을 생성하고 유지 관리하는 데 사용할 수 있습니다. AWS Cloud Map의 작동 방식은 다음과 같습니다.

1. 리소스를 찾는 데 사용하려는 이름을 식별하고 리소스를 찾는 방식(AWS Cloud Map DiscoverInstances API 호출, VPC에서 DNS 쿼리 또는 퍼블릭 DNS 쿼리 사용)을 지정하는 네임스페이스를 생성합니다. 대부분의 경우 네임스페이스는 요금 청구 애플리케이션 등과 같은 단일 애플리케이션에 대한 모든 서비스를 포함합니다.
2. AWS Cloud Map을 사용하여 엔드포인트를 찾으려는 각 리소스 유형에 대해 AWS Cloud Map 서비스를 생성합니다. 예를 들어, 웹 서버 및 데이터베이스 서버를 위한 서비스를 생성할 수 있습니다.

   서비스는 애플리케이션이 다른 리소스(예: 또 다른 웹 서버)를 추가하는 경우 AWS Cloud Map에서 사용하는 템플릿입니다. 네임스페이스를 생성할 때 DNS를 사용하여 리소스를 찾도록 선택한 경우, 서비스에는 웹 서버를 찾을 때 사용하려는 레코드 유형에 대한 정보가 포함됩니다. 또한 서비스는 리소스의 상태를 확인할지 여부를 나타내고, 확인하려는 경우 Amazon Route 53 상태 확인 또는 타사 상태 확인 프로그램을 사용할지 여부도 표시합니다.
3. 애플리케이션이 리소스를 추가하는 경우 애플리케이션은 서비스 인스턴스를 생성하는 AWS Cloud Map 인스턴스 등록 API 작업을 호출할 수 있습니다. 이 서비스 인스턴스에는 애플리케이션이 리소스를 찾을 수 있는 방법과 DNS를 사용할지, AWS Cloud Map DiscoverInstances API 작업을 사용할지에 대한 정보가 포함되어 있습니다.
4. 애플리케이션이 리소스에 연결해야 하는 경우 애플리케이션은 DiscoverInstances를 호출하고, 리소스와 연결된 네임스페이스 및 서비스를 지정합니다. AWS Cloud Map에서는 하나 이상의 리소스를 찾는 방법에 대한 정보를 반환합니다. 서비스를 생성할 때 상태 확인을 지정한 경우 AWS Cloud Map에서는 정상 상태인 인스턴스만 반환합니다.

AWS Cloud Map는 Amazon Elastic Container Service(Amazon ECS)와 긴밀하게 통합되어 있습니다. 새 컨테이너 작업은 실행 또는 종료될 때 AWS Cloud Map에 자동으로 등록합니다. Kubernetes ExternalDNS 커넥터를 사용하여 Amazon Elastic Kubernetes Service를 AWS Cloud Map와 통합할 수 있습니다. 또한 Amazon EC2 인스턴스, Amazon DynamoDB 테이블, Amazon S3 버킷, Amazon Simple Queue Service(Amazon SQS) 대기열 또는 Amazon API Gateway 위에 배포된 API 등과 같은 클라우드 리소스를 등록하고 찾는 데 AWS Cloud Map를 사용할 수 있습니다. 서비스 인스턴스에 대한 속성 값을 지정할 수 있고, 클라이언트는 이러한 속성을 사용하여 AWS Cloud Map에서 반환하는 리소스를 필터링할 수 있습니다. 예를 들어 애플리케이션은 특정 배포 단계의 리소스를 요청할 수 있습니다(예: BETA 또는 PROD).

주제

AWS Cloud Map에 액세스

AWS Cloud Map에 액세스하는 방법은 다음과 같습니다.

- AWS Management Console – 이 가이드의 절차는 AWS Management Console을 사용하여 작업을 수행하는 방법을 설명합니다.
- AWS SDKs – AWS에서 SDK를 제공하는 프로그래밍 언어를 사용하는 경우, SDK를 사용하여 AWS Cloud Map에 액세스할 수 있습니다. SDK는 인증을 단순화하고, 개발 환경에 쉽게 통합되며, AWS Cloud Map 명령에 액세스할 수 있도록 합니다. 자세한 내용은 Amazon Web Services용 도구를 참조하세요.
- AWS Command Line Interface— 자세한 정보는 AWS Command Line Interface 사용 설명서에서 AWS Command Line Interface 설정하기를 참조하세요.
- AWS Tools for Windows PowerShell— 자세한 정보는 AWS Tools for Windows PowerShell 사용 설명서에서 AWS Tools for Windows PowerShell 설정하기를 참조하세요.
- AWS Cloud Map API - SDK가 제공되지 않는 프로그래밍 언어를 사용하는 경우, AWS Cloud Map API 참조에서 API 작업에 대한 정보와 API 요청을 수행하는 방법을 참조하세요.

> **Note**
>
> IPv6 클라이언트 지원 – 2023년 6월 22일부터 모든 새 리전에서 IPv6 클라이언트에서 AWS Cloud Map로 전송된 모든 명령은 새로운 듀얼 스택 엔드포인트(servicediscovery.<region>.api.aws)로 라우팅됩니다. 2023년 6월 22일 이전에 출시된 다음 리전의 레거시(servicediscovery.<region>.amazonaws.com) 및 듀얼 스택 엔드포인트 모두 AWS Cloud Map IPv6 전용 네트워크에 연결할 수 있습니다.
>
> - 미국 동부(오하이오) - us-east-2
> - 미국 동부(버지니아 북부) - us-east-1
> - 미국 서부(캘리포니아 북부) - us-west-1
> - 미국 서부(오레곤) - us-west-2

- 아프리카(케이프타운) – af-south-1
- 아시아 태평양(홍콩) - ap-east-1
- 아시아 태평양(하이데라바드) – ap-south-2
- 아시아 태평양(자카르타) – ap-southeast-3
- 아시아 태평양(멜버른) – ap-southeast-4
- 아시아 태평양(뭄바이) - ap-south-1
- 아시아 태평양(오사카) – ap-northeast-3
- 아시아 태평양(서울) - ap-northeast-2
- 아시아 태평양(싱가포르) - ap-southeast-1
- 아시아 태평양(시드니) - ap-southeast-2
- 아시아 태평양(도쿄) - ap-northeast-1
- 캐나다(중부) - ca-central-1
- 유럽(프랑크푸르트) - eu-central-1
- 유럽(아일랜드) - eu-west-1
- 유럽(런던) - eu-west-2
- 유럽(밀라노) – eu-south-1
- 유럽(파리) - eu-west-3
- 유럽(스페인) – eu-south-2
- 유럽(스톡홀름) - eu-north-1
- 유럽(취리히) – eu-central-2
- 중동(바레인) – me-south-1
- 중동(UAE) – me-central-1
- 남아메리카(상파울루) - sa-east-1
- AWS GovCloud(미국 동부) – us-gov-east-1
- AWS GovCloud(미국 서부) – us-gov-west-1

AWS Identity and Access Management

AWS Cloud Map는 AWS Identity and Access Management(IAM)과 통합됩니다. 이 서비스를 이용하면 조직은 다음과 같은 작업을 수행할 수 있습니다.

- 조직의 AWS 계정에 사용자 및 그룹 생성
- 계정 사용자 간에 효율적으로 AWS 계정 리소스 공유
- 각 사용자에게 고유한 보안 자격 증명을 할당합니다.
- 서비스 및 리소스에 대한 사용자 액세스 상세 제어

예를 들어, AWS Cloud Map를 IAM과 함께 사용하여 새 네임스페이스를 생성하거나 인스턴스를 등록할 수 있는 AWS 계정의 사용자를 제어할 수 있습니다.

IAM에 대한 전반적인 정보는 다음 리소스를 참조하세요.

- AWS Identity and Access Management 에서 AWS Cloud Map
- AWS Identity and Access Management
- IAM 사용 설명서

AWS Cloud Map 요금

AWS Cloud Map 요금은 서비스 레지스트리에 등록한 리소스와 리소스를 검색하기 위해 수행한 API 호출 횟수를 기준으로 계산됩니다. AWS Cloud Map을 사용하면 선결제 금액이 없으며, 사용한 만큼만 비용을 지불합니다.

경우에 따라 IP 주소를 사용하여 리소스에 대한 DNS 기반 검색을 활성화할 수 있습니다. 또한 인스턴스 검색에 API 호출을 사용하는지 DNS 쿼리를 사용하는지와 관계없이 Amazon Route 53 상태 확인을 사용하여 리소스에 대한 상태 확인을 활성화할 수 있습니다. Route 53 DNS 및 상태 확인 사용과 관련해 추가 비용이 발생합니다.

자세한 내용은 AWS Cloud Map 요금을 참조하세요.

AWS Cloud Map 및 AWS 클라우드 규정 준수

AWS Cloud Map의 다양한 보안 준수 규정 및 감사 표준 준수에 대한 자세한 내용은 다음 페이지를 참조하세요.

- AWS 클라우드 규정 준수
- 규정 준수 프로그램 제공 AWS 범위 내 서비스

AWS Cloud Map설정

이 섹션의 개요와 절차는 AWS를 시작하는 데 도움이 됩니다.

주제

- 가입하기 AWS
- API, AWS CLIAWS Tools for Windows PowerShell, 또는 SDK에 AWS 액세스
- 또는 설정 AWS Command Line InterfaceAWS Tools for Windows PowerShell
- AWS SDK 다운로드

가입하기 AWS

등록해 보세요 AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 여세요.
2. 온라인 지시 사항을 따르세요.

   등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

   에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요. AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.
2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자 (콘솔) 에 대한 가상 MFA 디바이스 활성화를 참조하십시오.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

   를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

관리 액세스 권한이 있는 사용자로 로그인

- IAM IDentity Center 사용자로 로그인하려면 IAM IDentity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

  IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인을 참조하십시오.AWS 로그인

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

API, AWS CLIAWS Tools for Windows PowerShell, 또는 SDK에 AWS 액세스

API, AWS CLI AWS Tools for Windows PowerShell, 또는 AWS SDK를 사용하려면 액세스 키를 만들어야 합니다. 이들 키는 액세스 키 ID 및 보안 액세스 키로 이루어져 있는데, 이를 사용하여 AWS에 보내는 프로그래밍 방식의 요청에 서명할 수 있습니다.

사용자가 AWS 외부 사용자와 상호 작용하려는 경우 프로그래밍 방식의 액세스가 필요합니다. AWS Management Console프로그래밍 방식의 액세스 권한을 부여하는 방법은 액세스하는 사용자 유형에 따라 다릅니다. AWS

사용자에게 프로그래밍 방식 액세스 권한을 부여하려면 다음 옵션 중 하나를 선택합니다.

프로그래밍 방식 액세스가 필요한 사용자는 누구인가요?	To	액세스 권한을 부여하는 사용자
작업 인력 ID (IAM Identity Center가 관리하는 사용자)	임시 자격 증명을 사용하여 AWS CLI, AWS SDK 또는 API에 대한 프로그래밍 요청에 서명할 수 있습니다. AWS	사용하고자 하는 인터페이스에 대한 지침을 따릅니다. • AWS CLI 에 대한 내용은 사용 설명서의 AWS CLI사용을 AWS IAM Identity Center위한 구성을 참조하십시오.AWS Command Line Interface • AWS SDK, 도구 및 AWS API의 경우 AWS SDK 및 도구 참조 안내서의 IAM ID 센터 인증을 참조하십시오.
IAM	임시 자격 증명을 사용하여 AWS CLI, AWS SDK 또는 API에 대한 프로그래밍 방식 요청에 서명할 수 있습니다. AWS	IAM 사용 설명서의 AWS 리소스와 함께 임시 자격 증명 사용의 지침을 따르십시오.
IAM	(권장되지 않음) 장기 자격 증명을 사용하여 AWS CLI, AWS SDK 또는 API	사용하고자 하는 인터페이스에 대한 지침을 따릅니다.

프로그래밍 방식 액세스가 필요한 사용자는 누구인가요?	To	액세스 권한을 부여하는 사용자
	에 대한 프로그래밍 요청에 서명할 수 있습니다. AWS	• 에 대한 내용은 사용 설명서의 IAM 사용자 자격 증명을 사용한 인증을 참조하십시오. AWS CLIAWS Command Line Interface • AWS SDK 및 도구의 경우 SDK 및 도구 참조 안내서의 장기 자격 증명을 사용한 인증을 참조하십시오.AWS • AWS API의 경우 IAM 사용 설명서의 IAM 사용자의 액세스 키 관리를 참조하십시오.

또는 설정 AWS Command Line InterfaceAWS Tools for Windows PowerShell

AWS Command Line Interface (AWS CLI) 는 AWS 서비스를 관리하기 위한 통합 도구입니다. 설치 및 구성 방법에 대한 자세한 내용은 AWS Command Line Interface 사용 설명서의 AWS CLI설정을 참조하십시오. AWS Command Line Interface

Windows를 사용해 본 경험이 있다면 PowerShell Windows를 사용하는 것이 더 나을 수 AWS Tools for Windows PowerShell있습니다. 자세한 내용은 AWS Tools for Windows PowerShell 사용 설명서에서 AWS Tools for Windows PowerShell설정을 참조하세요.

AWS SDK 다운로드

SDK를 AWS 제공하는 프로그래밍 언어를 사용하는 경우 API 대신 SDK를 사용하는 것이 좋습니다. AWS Cloud Map SDK를 사용하면 여러 가지 장점이 있습니다. SDK는 인증을 단순화하고, 개발 환경에 쉽게 통합되며, AWS Cloud Map 명령에 액세스할 수 있도록 합니다. 자세한 내용은 Amazon Web Services용 도구를 참조하십시오.

AWS Cloud Map 사용

AWS Cloud Map는 애플리케이션의 리소스에 논리명을 매핑하는 데 사용할 수 있는 관리형 솔루션입니다. 또한 애플리케이션이 AWS SDK, RESTful API 호출 또는 DNS 쿼리 중 하나를 사용하여 리소스를 검색하는 데도 도움이 됩니다. AWS Cloud Map는 Amazon DynamoDB(DynamoDB) 테이블, Amazon Simple Queue Service(Amazon SQS) 대기열, Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스 또는 Amazon Elastic Container Service(Amazon ECS) 작업을 사용하여 구축된 모든 상위 수준 애플리케이션 서비스와 같은 정상적인 리소스만 제공합니다.

주제

AWS Cloud Map 사용 방법 개요

다음은 AWS Cloud Map의 사용 방법에 대한 개요입니다.

1. 서비스의 논리적 그룹인 네임스페이스를 생성합니다. 네임스페이스를 생성할 때 애플리케이션에서 인스턴스를 검색하는 데 사용할 이름을 지정합니다. 또한 AWS Cloud Map에 등록한 서비스 인스턴스를 검색하는 방법을 지정합니다(API 호출 사용 또는 DNS 쿼리 사용).

   자세한 정보는 다음 주제를 참조하세요.

   - AWS Cloud Map 네임스페이스 생성
   - AWS Cloud Map API 참조 내 CreatePublicDnsNamespace, CreatePrivateDnsNamespace, and CreateHttpNamespace

   퍼블릭 또는 프라이빗 DNS 네임스페이스를 생성하는 경우, AWS Cloud Map는 네임스페이스와 이름이 같은 Amazon Route 53 퍼블릭 또는 프라이빗 호스팅 영역을 자동으로 생성합니다. 퍼블릭 및 프라이빗 DNS 네임스페이스를 사용하더라도 AWS Cloud Map DiscoverInstances 요청을 사용하여 인스턴스를 검색할 수 있습니다.

   AWS Cloud Map API 요청을 제출할 수 있는 엔드포인트 목록은 Amazon Web Services 일반 참조의 “AWS 지역 및 엔드포인트” 장에서 AWS Cloud Map를 참조하세요.

2. 퍼블릭 DNS 네임스페이스를 생성한 경우, 다음 단계를 수행하여 도메인 등록을 위한 이름 서버를 네임스페이스를 만들었을 때 AWS Cloud Map에서 생성한 Route 53 호스팅 영역에 대한 이름 서버로 변경합니다.

   a. 퍼블릭 DNS 네임스페이스와 이름이 같은 도메인을 이미 등록한 경우 2b단계로 건너뜁니다.

      이 네임스페이스와 이름이 같은 도메인을 등록하지 않은 경우 도메인을 등록합니다. 도메인 등록에 Route 53을 사용하려면 Amazon Route 53 개발자 안내서의 새 도메인 등록을 참조하세요. 그런 다음 3단계로 건너뜁니다.

   b. 네임스페이스를 생성했을 때 반환된 `OperationId`를 사용하여 네임스페이스 ID를 얻습니다. 자세한 내용은 GetOperation을 참조하세요.

      > **Note**
      >
      > 프로그래밍 방식으로 이러한 단계를 수행하는 경우, 프로세스의 뒷부분에서 네임스페이스 ID를 사용하여 서비스를 생성합니다.

   c. 2b단계에서 얻은 네임스페이스 ID를 사용하여 AWS Cloud Map에서 생성한 Route 53 호스팅 영역의 ID를 얻습니다. 자세한 내용은 AWS Cloud Map용 API 참조의 GetNamespace를 참조하세요.

   d. 2c단계에서 얻은 호스팅 영역 ID를 사용하여 Route 53에서 호스팅 영역에 할당한 이름 서버의 이름을 얻습니다. 자세한 내용은 퍼블릭 호스팅 영역에 대한 이름 서버 가져오기를 참조하세요.

   e. 도메인에 할당된 이름 서버를 변경합니다. 도메인이 Route 53에 등록된 경우 자세한 내용은 도메인의 글루 레코드 및 이름 서버 추가 또는 변경을 참조하세요.

3. 애플리케이션의 리소스(예: 웹 서버, DynamoDB 테이블 또는 Amazon S3 버킷)에 접속하는 방법을 식별하는 서비스 인스턴스가 포함된 서비스를 생성합니다.

   1단계에서 퍼블릭 또는 프라이빗 DNS 네임스페이스를 생성한 경우 서비스에 대해 지정한 이름이 AWS Cloud Map에서 1단계에서 자동으로 생성한 Route 53 퍼블릭 또는 프라이빗 호스팅 영역의 레코드 이름 중 일부가 됩니다. 다음 단계에서 인스턴스를 등록하면 AWS Cloud Map이 호스팅 영역에서 레코드를 생성합니다. 레코드 이름은 서비스 이름(예: `backend`)과 네임스페이스 이름(예: `example.com`)의 조합입니다(예: `backend.example.com`).

   서비스를 생성할 때 서비스 인스턴스가 가리키는 리소스의 상태를 확인할지 여부를 선택할 수도 있습니다.

- 상태를 확인하지 않도록 선택한 경우 AWS Cloud Map 또는 Route 53에서는 해당 리소스의 상태와 관계없이 서비스 인스턴스를 반환합니다.
- Route 53 상태 확인을 선택한 경우(퍼블릭 DNS 네임스페이스의 경우에만 선택 가능), AWS Cloud Map에서는 Route 53 상태 확인을 자동으로 생성하여 해당 Route 53 레코드와 연결합니다. Route 53에서는 정상 리소스에 대한 레코드로만 DNS 쿼리에 응답합니다.
- 사용자 지정 상태 확인을 선택한 경우 타사 애플리케이션을 사용하여 리소스 상태를 확인할 수 있습니다. 타사 상태 확인 결과에 따라 UpdateInstanceCustomHealthStatus 요청을 AWS Cloud Map으로 보내 서비스 인스턴스의 상태를 업데이트합니다.

상태 확인을 구성하면 AWS Cloud Map 또는 Route 53에서는 DiscoverInstances 요청 또는 DNS 쿼리에 대한 응답으로 정상 리소스에 대한 서비스 인스턴스만 반환합니다.

자세한 정보는 다음 주제를 참조하세요.

- AWS Cloud Map 서비스 생성
- AWS Cloud Map API 참조의 CreateService

4. 하나 이상의 서비스 인스턴스를 등록합니다. 각 서비스 인스턴스에는 애플리케이션에서 애플리케이션의 리소스 하나에 접속하는 방법에 대한 정보가 포함되어 있습니다.

   자세한 정보는 다음 주제를 참조하세요.

   - AWS Cloud Map 서비스 인스턴스 등록
   - AWS Cloud Map API 참조의 RegisterInstance

5. AWS Cloud Map DiscoverInstances API 작업 또는 DNS 쿼리를 사용하여 인스턴스를 검색하도록 애플리케이션을 작성합니다.

   - 애플리케이션에서 DiscoverInstances를 사용하는 경우 AWS Cloud Map에서는 지정된 기준을 충족하는 사용 가능한 인스턴스에 대한 정보를 반환합니다.
   - 애플리케이션에서 DNS 쿼리를 사용하는 경우 Route 53에서는 레코드를 하나 이상 반환합니다.

   서비스 생성 시 상태 확인에 대한 설정을 지정한 경우 AWS Cloud Map 또는 Route 53에서는 정상 인스턴스에 대한 값만 반환합니다.

6. 리소스 사용을 중지하려는 경우 해당 서비스 인스턴스의 등록을 취소합니다. AWS Cloud Map에서는 연결된 Route 53 레코드와 상태 확인(있는 경우)을 자동으로 삭제합니다.

   자세한 정보는 다음 주제를 참조하세요.

- AWS Cloud Map 서비스 인스턴스 등록 취소
- AWS Cloud Map API 참조의 인스턴스 등록 취소

7. 서비스 및 네임스페이스가 더 이상 필요 없는 경우 삭제할 수 있습니다. 다음을 참조합니다.

   - 서비스를 삭제하기 전에 서비스를 사용해 등록한 모든 인스턴스를 등록 취소해야 합니다.
   - 네임스페이스를 삭제하려면 먼저 네임스페이스에서 생성한 서비스를 모두 삭제해야 합니다.

   자세한 정보는 다음 주제를 참조하세요.

   - AWS Cloud Map 서비스 삭제
   - AWS Cloud Map 네임스페이스 삭제
   - AWS Cloud Map API 참조의 DeleteService
   - AWS Cloud Map API 참조의 DeleteNamespace

구성 AWS Cloud Map

다음 섹션에서는 AWS Cloud Map 콘솔 사용 방법, 네임스페이스 및 AWS CLI 서비스를 생성, 확인 및 삭제하고 인스턴스를 등록 및 등록 취소하는 방법을 설명합니다.

프로덕션 환경에서는 아마도 대부분의 AWS Cloud Map 작업을 프로그래밍 방식으로 수행할 것입니다. 프로그래밍 방식 액세스에 AWS Cloud Map 대한 자세한 내용은 다음 문서 및 다운로드 페이지를 참조하십시오.

- AWS Cloud Map설정
- Amazon Web Services용 도구에는 SDK, 명령줄 도구 및 기타 개발자 리소스가 나와 있습니다.
- AWS Cloud Map API 참조에서는 SDK를 제공하지 AWS 않는 프로그래밍 언어를 사용할 때의 AWS Cloud Map API 사용에 대한 정보를 제공합니다.

주제

- AWS Cloud Map 네임스페이스 사용
- AWS Cloud Map 서비스 관련 작업
- AWS Cloud Map 서비스 인스턴스 사용
- AWS Cloud Map 콘솔에서 AWS Cloud Map 사용할 수 없는 기능

AWS Cloud Map 네임스페이스 사용

네임스페이스는 애플리케이션에 필요한 서비스를 그룹화하는 방식입니다. 네임스페이스를 생성할 때 등록하는 서비스 인스턴스를 검색하는 방법 (API 호출 사용 또는 DNS 쿼리 사용) 을 지정합니다 AWS Cloud Map. 또한 애플리케이션에서 인스턴스를 검색하는 데 사용할 이름도 지정합니다.

주제

AWS Cloud Map 네임스페이스 생성

네임스페이스를 생성하려면 다음 절차를 수행합니다.

AWS Management Console

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. Create namespace(네임스페이스 생성)를 선택합니다.
3. Create namespace(네임스페이스 생성) 페이지에서 해당 값을 입력합니다. 자세한 설명은 네임스페이스를 생성할 때 지정하는 값 섹션을 참조하세요.
4. Create namespace(네임스페이스 생성)를 선택합니다.

AWS CLI

- 원하는 인스턴스 검색 유형의 명령을 사용하여 네임스페이스를 생성(### 값을 사용자 고유 값으로 대체)합니다.

  - create-http-namespace를 사용하여 HTTP 네임스페이스를 생성합니다. HTTP 네임스페이스를 사용하여 등록하는 서비스 인스턴스는 DiscoverInstances 요청을 사용하여 검색할 수 있지만 DNS를 사용하여 검색할 수 없습니다.

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

  - create-private-dns-namespace를 사용하여 지정된 Amazon VPC 내에서만 볼 수 있는 DNS에 기반한 프라이빗 네임스페이스를 만듭니다. DiscoverInstances 요청을 사용

하거나 DNS를 사용하여 프라이빗 DNS 네임스페이스에 등록된 인스턴스를 검색할 수 있습니다.

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

- create-public-dns-namespace를 사용하여 인터넷에서 볼 수 있는 DNS 기반 퍼블릭 네임스페이스를 생성합니다. DiscoverInstances 요청을 사용하거나 DNS를 사용하여 퍼블릭 DNS 네임스페이스에 등록된 인스턴스를 검색할 수 있습니다.

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

> **Note**
>
> 네임스페이스 요구 사항:
>
> - 퍼블릭 DNS 쿼리용으로 구성된 네임스페이스는 최상위 도메인(예: .com)으로 끝나야 합니다.
> - 네임스페이스 이름은 최대 1,024자까지 가능하며 문자로 시작하고 끝나야 합니다.
> - 유효한 문자: A~Z, a~z, 0~9, -(하이픈), _(밑줄) 및 .(마침표).

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. 원하는 인스턴스 검색 유형의 명령을 사용하여 네임스페이스를 생성(### 값을 사용자 고유 값으로 대체)합니다.

   - create_http_namespace()를 사용하여 HTTP 네임스페이스를 생성합니다. HTTP 네임스페이스를 사용하여 등록하는 서비스 인스턴스는 discover_instances()를 사용하여 검색할 수 있지만 DNS를 사용하여 검색할 수 없습니다.

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

- `create_private_dns_namespace()`를 사용하여 지정된 Amazon VPC 내에서만 볼 수 있는 DNS에 기반한 프라이빗 네임스페이스를 만듭니다. `discover_instances()`를 사용하거나 DNS를 사용하여 퍼블릭 DNS 네임스페이스에 등록된 인스턴스를 검색할 수 있습니다.

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

- `create_public_dns_namespace()`를 사용하여 인터넷에서 볼 수 있는 DNS 기반 퍼블릭 네임스페이스를 생성합니다. `discover_instances()` 또는 DNS를 사용하여 퍼블릭 DNS 네임스페이스에 등록된 인스턴스를 검색할 수 있습니다.

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

- 예시 응답 출력

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

> **Note**
>
> 네임스페이스 요구 사항:
>
> - 퍼블릭 DNS 쿼리용으로 구성된 네임스페이스는 최상위 도메인(예: .com)으로 끝나야 합니다.
> - 네임스페이스 이름은 최대 1,024자까지 가능하며 문자로 시작하고 끝나야 합니다.
> - 유효한 문자: A~Z, a~z, 0~9, -(하이픈), _(밑줄) 및 .(마침표).

네임스페이스를 생성할 때 지정하는 값

AWS Cloud Map 네임스페이스를 만들 때 다음 값을 지정합니다.

> **Note**
>
> 네임스페이스를 생성하면, 태그를 변경할 수 있습니다. 그러나 다른 값은 변경할 수 없습니다.

값

- Namespace name
- Namespace description
- Instance discovery
- Tags
- VPC

네임스페이스 이름

네임스페이스에 지정하는 이름은 애플리케이션에서 인스턴스를 검색하는 방법에 따라 달라집니다. 인스턴스가 검색되는 방법은 인스턴스 검색에 선택한 옵션에 따라 결정됩니다. 옵션은 콘솔의 현재 페이지 뒷부분에 표시됩니다. 그 속성이란 다음과 같습니다.

API 호출

이 옵션을 선택하면 애플리케이션이 DiscoverInstances 요청에 네임스페이스 이름 및 서비스 이름을 지정해 서비스 인스턴스를 검색합니다. 자세한 내용을 알아보려면 AWS Cloud Map API 참조의 DiscoverInstances(을)를 참조하세요.

1~1,024자 길이의 이름을 지정할 수 있습니다. 이름은 대문자 및 소문자 모두, 숫자, 밑줄(_), 하이픈(-)을 포함할 수 있습니다.

VPC에서 API 호출 및 DNS 쿼리

VPC의 애플리케이션이 DNS 쿼리를 제출하여 인스턴스를 발견할 때 사용할 도메인 이름을 입력합니다. AWS Cloud Map 이 이름을 가진 Amazon Route 53 프라이빗 호스팅 영역을 자동으로 생성합니다. 서비스 인스턴스를 등록하면 AWS Cloud Map 은 호스팅 영역에 다음과 같은 형식의 이름을 가진 DNS 레코드를 생성합니다.

service-name.*namespace-name*

이 옵션을 선택하면 애플리케이션이 DiscoverInstances 요청에 네임스페이스 이름 및 서비스 이름을 지정해 인스턴스를 검색할 수 있습니다. 자세한 내용을 알아보려면 AWS Cloud Map API 참조의 DiscoverInstances(을)를 참조하세요.

이름을 먼저 퓨니코드로 변환하는 경우에는 다국어 도메인 이름(IDN)을 지정할 수 있습니다. 온라인 변환기에 대한 자세한 내용은 인터넷에서 "punycode converter"를 검색하세요.

또한 네임스페이스를 프로그래밍 방식으로 생성하는 경우 다국어 도메인 이름을 퓨니코드로 변환할 수 있습니다. 예를 들어, Java를 사용하는 경우 java.net.IDN 라이브러리의 `toASCII` 메서드를 사용하여 유니코드 값을 퓨니코드로 변환할 수 있습니다.

API 호출 및 퍼블릭 DNS 쿼리

퍼블릭 DNS 쿼리를 제출하여 인스턴스를 검색하는 경우 애플리케이션에서 사용하도록 할 도메인 이름을 입력합니다. 이 이름은 등록한 도메인 이름이어야 합니다. 네임스페이스를 생성하면 동일한 이름을 가진 Amazon Route 53 퍼블릭 호스팅 영역이 AWS Cloud Map 자동으로 생성됩니다. 서비스 인스턴스를 등록하면 AWS Cloud Map 은 호스팅 영역에 다음과 같은 형식의 이름을 가진 DNS 레코드를 생성합니다.

service-name.*namespace-name*

이 옵션을 선택하면 애플리케이션이 DiscoverInstances 요청에 네임스페이스 이름 및 서비스 이름을 지정해 인스턴스를 검색할 수 있습니다. 자세한 내용을 알아보려면 AWS Cloud Map API 참조의 DiscoverInstances(을)를 참조하세요.

이름을 먼저 퓨니코드로 변환하는 경우에는 다국어 도메인 이름(IDN)을 지정할 수 있습니다. 온라인 변환기에 대한 자세한 내용은 인터넷에서 "punycode converter"를 검색하세요.

또한 네임스페이스를 프로그래밍 방식으로 생성하는 경우 다국어 도메인 이름을 퓨니코드로 변환할 수 있습니다. 예를 들어, Java를 사용하는 경우 java.net.IDN 라이브러리의 `toASCII` 메서드를 사용하여 유니코드 값을 퓨니코드로 변환할 수 있습니다.

네임스페이스 설명

네임스페이스에 대한 설명을 입력합니다. 여기 입력한 값이 네임스페이스 페이지와 각 네임스페이스의 세부 정보 페이지에 표시됩니다.

인스턴스 검색

다음 중에서 애플리케이션이 등록된 인스턴스를 검색하는 방법을 선택합니다.

API 호출

애플리케이션이 API 호출만 사용해 등록된 인스턴스를 찾도록 하려는 경우 이 옵션을 선택합니다.

VPC에서 API 호출 및 DNS 쿼리

애플리케이션이 API 호출을 사용하거나 VPC에서 DNS 쿼리를 사용해 인스턴스를 검색할 수 있도록 하려는 경우 이 옵션을 선택합니다. 두 가지 방법을 모두 사용할 필요는 없습니다.

API 호출 및 퍼블릭 DNS 쿼리

애플리케이션이 API 호출을 사용하거나 퍼블릭 DNS 쿼리를 사용해 등록된 인스턴스를 검색할 수 있도록 하려는 경우 이 옵션을 선택합니다. 두 가지 방법을 모두 사용할 필요는 없습니다.

SOA TTL

VPC의 API 호출 및 DNS 쿼리 또는 API 호출 및 퍼블릭 DNS 쿼리의 경우 네임스페이스로 생성된 Route 53 호스팅 영역의 SOA(권한 시작) DNS 레코드에 대한 TTL(유지 시간) 값입니다. 값은 업데이트된 설정을 얻기 위해 DNS 해석기가 다른 DNS 쿼리를 Amazon Route 53에 전달하기 전에 이 레코드에 대한 정보를 캐싱하는 기간을 결정합니다. 또한 값이 작을수록 누락된 항목이 캐시되는 시간(음성 캐싱)이 줄어들지만, 해당 네임스페이스에 대한 추가 쿼리는 필요하지 않습니다.

태그

네임스페이스에 추가할 태그를 하나 이상 지정할 수 있습니다. 태그는 리소스에 할당할 수 있는 선택적 레이블입니다. AWS 각 태그는 키와 값으로 구성됩니다. 예를 들어 Key = Environment 및 Value = Production으로 태그를 정의할 수 있습니다. 태그를 사용하면 AWS 리소스를 분류하여 더 쉽게 관리할 수 있습니다.

태그를 만든 후에는 네임스페이스에서 태그를 업데이트하거나 제거할 수 있습니다. 자세한 설명은 AWS Cloud Map 리소스에 태그 지정 섹션을 참조하세요.

VPC

인스턴스 검색의 가치를 위해 VPC에서 API 호출 및 DNS 쿼리를 선택하면 동일한 이름을 가진 Amazon Route 53 프라이빗 호스팅 영역이 AWS Cloud Map 생성됩니다. AWS Cloud Map VPC 목록에서 선택한 VPC를 해당 프라이빗 호스팅 영역과 연결합니다.

Route 53 해석기는 프라이빗 호스팅 영역의 레코드를 사용하여 VPC에서 시작되는 DNS 쿼리를 해석합니다. 프라이빗 호스팅 영역에 DNS 쿼리의 도메인 이름과 일치하는 레코드가 없는 경우, Route 53에서는 NXDOMAIN(존재하지 않는 도메인)를 사용하여 쿼리에 응답합니다.

추가 VPC를 프라이빗 호스팅 영역과 연결할 수 있습니다. 자세한 내용은 Amazon Route 53 API WithHostedZone 참조의 AssociateVPC를 참조하십시오.

AWS Cloud Map 네임스페이스 보기

생성한 네임스페이스의 목록을 보려면 다음 절차를 수행하세요.

AWS Management Console

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.

AWS CLI

- `list-namespaces` 명령을 사용하여 네임스페이스를 나열합니다.

```
aws servicediscovery list-namespaces
```

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. `list_namespaces()`을 사용하여 네임스페이스를 나열합니다.

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

예시 응답 출력

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

AWS Cloud Map 네임스페이스 삭제

네임스페이스를 삭제하면 서비스 인스턴스를 등록 또는 검색하는 데 해당 네임스페이스를 더 이상 사용할 수 없습니다. 유념할 사항:

- 네임스페이스를 삭제하려면 먼저 네임스페이스에서 생성한 서비스를 모두 삭제해야 합니다. 자세한 설명은 AWS Cloud Map 서비스 삭제 섹션을 참조하세요.
- 서비스를 삭제하기 전에 해당 서비스를 사용해 등록한 모든 서비스 인스턴스를 등록 취소해야 합니다. 자세한 설명은 AWS Cloud Map 서비스 인스턴스 등록 취소 섹션을 참조하세요.
- 네임스페이스를 생성할 때 VPC에서 퍼블릭 DNS 쿼리 또는 DNS 쿼리를 사용하여 서비스 인스턴스를 검색하도록 지정하는 경우 Amazon Route 53 퍼블릭 또는 프라이빗 호스팅 영역을 생성합니다. AWS Cloud Map 네임스페이스를 삭제하면 해당 호스팅 영역이 AWS Cloud Map 삭제됩니다.

네임스페이스를 삭제하려면 다음 절차를 수행합니다.

AWS Management Console

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.

2. 탐색 창에서 네임스페이스를 선택합니다.
3. 삭제하려는 네임스페이스를 선택한 다음 삭제를 선택합니다.
4. [Delete] 를 다시 선택하여 서비스 삭제를 확인합니다.

AWS CLI

- delete-namespace 명령으로 네임스페이스를 삭제(### 값을 사용자 고유 값으로 대체)합니다. 네임스페이스에 여전히 하나 이상의 서비스가 포함되어 있으면 요청이 실패합니다.

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

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. delete_namespace()로 네임스페이스를 삭제(### 값을 사용자 고유 값으로 대체)합니다. 네임스페이스에 여전히 하나 이상의 서비스가 포함되어 있으면 요청이 실패합니다.

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

예시 응답 출력

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

AWS Cloud Map 서비스 관련 작업

서비스는 서비스 인스턴스를 등록하기 위한 템플릿으로, 네임스페이스를 구성한 방식에 따라 DNS 쿼리 또는 AWS Cloud Map DiscoverInstancesAPI 작업을 사용하여 애플리케이션의 리소스를 찾을 수 있습니다.

주제

- AWS Cloud Map 서비스 생성
- AWS Cloud Map 서비스 업데이트
- 네임스페이스에서 서비스 보기
- AWS Cloud Map 서비스 삭제

AWS Cloud Map 서비스 생성

서비스를 생성하려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 네임스페이스 페이지에서 서비스를 추가하려는 네임스페이스를 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 서비스 생성을 선택합니다.
5. 서비스 생성 페이지에서 해당 값을 입력합니다. 자세한 설명은 서비스를 생성할 때 지정하는 값 섹션을 참조하세요.
6. 서비스 생성을 선택합니다.

AWS CLI

- `create-service` 명령을 사용하여 서비스를 생성(### 값을 사용자 고유 값으로 대체)합니다.

```
aws servicediscovery create-service \
    --name service-name \
    --namespace-id  ns-xxxxxxxxxxx \
```

```
    --dns-config "NamespaceId=ns-
xxxxxxxxxxx,RoutingPolicy=MULTIVALUE,DnsRecords=[{Type=A,TTL=60}]"
```

출력:

```
{
        "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)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. create_service()로 서비스를 생성(### 값을 사용자 고유 값으로 대체)합니다.

```
response = client.create_service(
    DnsConfig={
        'DnsRecords': [
```

```
            {
                'TTL': 60,
                'Type': 'A',
            },
        ],
        'NamespaceId': 'ns-xxxxxxxxxxx',
        'RoutingPolicy': 'MULTIVALUE',
    },
    Name='service-name',
    NamespaceId='ns-xxxxxxxxxxx',
)
```

예시 응답 출력

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

> **Note**
>
> DNS 쿼리로 액세스할 수 있는 서비스의 경우, 철자는 같지만 대소문자만 다른 이름(예: EXAMPLE과 example)으로 여러 서비스를 생성할 수 없습니다. 그렇게 하지 않으면 이러한 서비스는 동일한 DNS 이름을 갖게 됩니다. API 호출로만 액세스할 수 있는 네임스페이스를 사용하는 경우, 철자는 같지만 대소문자는 다른 이름을 가진 서비스를 생성할 수 있습니다.

서비스를 생성할 때 지정하는 값

AWS Cloud Map 서비스를 생성할 때 다음 값을 지정합니다.

> **Note**
>
> 서비스를 생성한 후에는 서비스의 태그만 변경할 수 있습니다.

값

- Service name
- Service description
- Service discovery configuration
- Routing policy
- Record type
- TTL
- Health check options
- Failure threshold
- Health check protocol
- Health check path
- Tags

서비스 이름

이 서비스를 사용하여 등록할 인스턴스를 설명하는 이름을 입력합니다. 이 값은 API 호출 또는 DNS 쿼리에서 AWS Cloud Map 서비스 인스턴스를 검색하는 데 사용됩니다. 이는 네임스페이스를 생성할 때 선택한 인스턴스 검색 방법에 따라 달라집니다. 다음 방법 중 하나를 사용할 수 있습니다.

- API 호출 - 애플리케이션이 DiscoverInstances호출할 때 API 호출에는 네임스페이스와 서비스 이름이 포함됩니다.
- VPC에서 API 호출 및 DNS 쿼리 또는 API 호출 및 퍼블릭 DNS 쿼리 – 서비스 인스턴스를 등록하면 네임스페이스를 생성할 때 AWS Cloud Map 에서 생성한 Amazon Route 53 프라이빗 또는 퍼블릭 호스팅 영역을 생성합니다. 또한 해당 호스팅 영역에 DNS 레코드를 생성합니다. 이러한 레코드 이름의 형식은 다음과 같습니다.

  service-name.*namespace-name*

  애플리케이션이 DNS 쿼리를 제출하여 서비스 인스턴스를 검색하는 경우, 해당 쿼리는 레코드 이름에 서비스 이름을 포함하는 레코드에 대한 쿼리입니다.

> **Note**
>
> DNS 쿼리를 지원하는 네임스페이스에서 서비스를 생성할 때는 해당 서비스의 서비스 인스턴스가 DiscoverInstancesAPI 작업에 대한 호출로만 검색되고 DNS 쿼리는 검색되지 않도록 선택할 수 있습니다. Service discovery configuration을 참조하십시오.

특정 SRV 형식이 필요한 시스템 (예: HAProxy) 을 사용하는 경우 인스턴스를 등록할 때 SRV 레코드를 AWS Cloud Map 생성하려면 서비스 이름에 다음을 지정하십시오.

- 이름은 밑줄(_)로 시작합니다(예: _exampleservice).
- 이름을 *._protocol*로 끝냅니다(예: ._tcp).

인스턴스를 등록할 때 는 SRV 레코드를 AWS Cloud Map 생성하고 서비스 이름과 네임스페이스 이름을 연결하여 이름을 할당합니다. 예를 들면 다음과 같습니다.

_exampleservice._tcp.example.com

> **Note**
>
> DNS 쿼리로 액세스할 수 있는 서비스의 경우 철자는 같지만 대소문자는 다른 이름을 가진 여러 서비스를 생성할 수 없습니다(예: EXAMPLE과 example). 그렇게 하지 않으면 이러한 서비스의 DNS 이름이 같아져 구분할 수 없습니다.

서비스 설명

서비스에 대한 설명을 입력합니다. 여기 입력한 값이 서비스 페이지와 각 서비스의 세부 정보 페이지에 표시됩니다.

서비스 검색 구성

네임스페이스가 DNS 쿼리를 지원하는 경우 다음 서비스 검색 옵션을 지원합니다. AWS Cloud Map

API 및 DNS

AWS Cloud Map 서비스에 인스턴스를 등록할 때 SRV 레코드를 생성합니다. DiscoverInstancesAPI 작업을 사용하여 서비스 인스턴스를 검색할 수도 있습니다.

API 전용

AWS Cloud Map 예를 들어 서비스에 대한 SRV 레코드를 생성하지 않습니다. 서비스 인스턴스는 DiscoverInstancesAPI 작업을 통해서만 검색할 수 있습니다.

라우팅 정책(퍼블릭 및 프라이빗 DNS 네임스페이스만 해당)

퍼블릭 또는 프라이빗 DNS 네임스페이스를 사용하여 서비스를 생성하는 경우, 인스턴스 등록 시 AWS Cloud Map 에서 생성한 DNS 레코드에 대한 Amazon Route 53 라우팅 정책을 선택합니다. (퍼블릭 DNS 네임스페이스는 인스턴스 검색을 위한 API 호출 및 퍼블릭 DNS 쿼리 값을 가지며, 프라이빗 DNS 네임스페이스는 VPC의 API 호출 및 DNS 쿼리 값을 가집니다.)

> **Note**
>
> 인스턴스를 등록할 때 콘솔을 사용하여 Route 53 별칭 레코드를 AWS Cloud Map 생성하도록 구성할 수 없습니다. 프로그래밍 방식으로 인스턴스를 등록할 때 Elastic Load Balancing 로드 밸런서에 대한 별칭 레코드를 AWS Cloud Map 생성하려면 라우팅 정책에 대한 가중치 라우팅을 선택하십시오.

AWS Cloud Map 다음과 같은 Route 53 라우팅 정책을 지원합니다.

가중치 기반 라우팅

Route 53에서는 동일한 서비스를 사용하여 등록한 인스턴스 중 임의로 선택한 하나의 인스턴스에서 해당하는 값을 반환합니다. 모든 레코드가 동일한 가중치를 갖기 때문에 인스턴스로 라우팅되는 트래픽을 늘리거나 줄일 수 없습니다.

예를 들어, 서비스에 A 레코드 하나와 상태 확인에 대한 구성이 포함되어 있는데, 이 서비스를 사용하여 인스턴스 10개를 등록한다고 가정해 보겠습니다. Route 53은 정상 인스턴스 중 무작위로 선택한 하나의 인스턴스에 대한 IP 주소를 사용하여 DNS 쿼리에 응답합니다. 정상 인스턴스가 없는 경우 Route 53은 마치 모든 인스턴스가 정상인 것처럼 DNS 쿼리에 응답합니다.

이 서비스에 대해 상태 확인을 정의하지 않은 경우 Route 53에서는 모든 인스턴스가 정상이라고 가정하고 임의로 선택한 인스턴스 하나에 대해 해당 값을 반환합니다.

자세한 내용은 Amazon Route 53 개발자 안내서의 가중치 기반 라우팅을 참조하세요.

다중값 응답 라우팅

이 서비스에 대해 상태 확인을 정의했고 상태 확인 결과가 정상인 경우 Route 53에서는 최대 8개 인스턴스에 대해 해당 값을 반환합니다.

예를 들어 서비스에 하나의 A 레코드와 상태 확인에 대한 구성이 포함되어 있다고 가정합니다. 서비스를 사용하여 10개의 인스턴스를 등록합니다. Route 53에서는 최대 8개의 정상 인스턴스에 대해서만 IP 주소를 사용하여 DNS 쿼리에 응답합니다. 정상 인스턴스가 8개 미만인 경우 Route 53에서는 전체 정상 인스턴스의 IP 주소로 모든 DNS 쿼리에 응답합니다.

이 서비스에 대해 상태 확인을 정의하지 않은 경우 Route 53에서는 모든 인스턴스가 정상이라고 가정하고 최대 8개 인스턴스에 대한 값을 반환합니다.

자세한 내용은 Amazon Route 53 개발자 안내서의 다중 응답 라우팅을 참조하세요.

레코드 유형(퍼블릭 및 프라이빗 DNS 네임스페이스만 해당)

퍼블릭 또는 프라이빗 DNS 네임스페이스를 사용하여 서비스를 생성하는 경우, 인스턴스를 등록할 때 AWS Cloud Map 생성되는 레코드의 DNS 레코드 유형을 선택하십시오. Amazon Route 53는 등록된 인스턴스의 DNS 쿼리에 응답해 해당하는 값을 반환합니다.

다음 레코드 유형이 지원됩니다.

A

인스턴스를 등록할 때 리소스의 IP 주소를 IPv4 형식으로 지정합니다(예: 192.0.2.44).

AAAA

인스턴스를 등록할 때 리소스의 IP 주소를 IPv6 형식으로 지정합니다(예: 2001:0db8:85a3:0000:0000:abcd:0001:2345).

CNAME

인스턴스를 등록할 때 리소스의 도메인 이름을 지정합니다(예: www.example.com). 유념할 사항:

- CNAME을 선택하려면 라우팅 정책에 대해 가중치 기반 라우팅을 선택해야 합니다.
- CNAME을 선택하면 상태 확인 옵션에 대해 Route 53 상태 확인을 선택할 수 없습니다.

SRV

SRV 레코드의 값은 다음 값을 사용합니다.

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

다음은 이 값에 대한 유의 사항입니다.

- `priority` 및 `weight` 값은 둘 다 1로 설정되어 있고 변경할 수 없습니다.
- For`port`, AWS Cloud Map 는 인스턴스를 등록할 때 Port에 지정한 값을 사용합니다.
- `service-hostname`의 값은 다음 값의 연결입니다.
  - 인스턴스 등록 시 서비스 인스턴스 ID에 대해 지정한 값
  - 서비스의 이름
  - 네임스페이스의 이름

  예를 들어, 인스턴스를 등록할 때 서비스 인스턴스 ID에 대한 테스트를 지정한다고 가정해 보겠습니다. 서비스 이름은 백엔드이고 네임스페이스의 이름은 example.com입니다. AWS Cloud Map 에서는 SRV 레코드의 `service-hostname` 속성에 다음 값을 할당합니다.

  ```
  test.backend.example.com
  ```

SRV 레코드에 대한 설정을 지정하는 경우 다음 사항에 유의하세요.

- IPv4 주소, IPv6 주소 또는 둘 다에 대한 값을 지정하면 AWS Cloud Map 에서 SRV 레코드의 `service-hostname` 값과 이름이 동일한 A 및/또는 AAAA 레코드를 자동으로 생성합니다.
- 특정 SRV 형식(예: HAProxy)이 필요한 시스템을 사용하는 경우 올바른 이름 형식을 지정하는 방법에 대한 자세한 내용은 서비스 이름을 참조하세요.

레코드 유형은 다음 조합으로 지정할 수 있습니다.

- A
- AAAA
- A 및 AAAA
- CNAME
- SRV

A 및 AAAA 레코드 유형을 지정한 경우, 인스턴스를 등록할 때 IPv4 IP 주소, IPv6 IP 주소 또는 둘 다를 지정할 수 있습니다.

TTL(퍼블릭 및 프라이빗 DNS 네임스페이스만 해당)

퍼블릭 또는 프라이빗 DNS 네임스페이스를 사용하여 서비스를 생성하는 경우, TTL(Time To Live) 값을 입력합니다. TTL 값은 업데이트된 설정을 얻기 위해 DNS 해석기가 다른 DNS 쿼리를 Amazon Route 53에 전달하기 전에 이 레코드에 대한 정보를 캐싱하는 기간을 결정합니다.

상태 확인 옵션

상태 확인 없음

상태 확인을 구성하지 않을 경우 서비스 인스턴스가 정상인지 여부와 상관없이 트래픽이 서비스 인스턴스로 라우팅됩니다.

Route 53 상태 확인(프라이빗 DNS 네임스페이스에는 지원되지 않음)

Amazon Route 53 상태 확인에 대한 설정을 지정하면 인스턴스를 등록할 때마다 AWS Cloud Map 에서는 Route 53 상태 확인을 생성하고 해당 인스턴스를 등록 취소하면 상태 확인을 삭제합니다.

퍼블릭 DNS 네임스페이스의 경우, 상태 점검을 인스턴스를 등록할 때 AWS Cloud Map 생성되는 Route 53 레코드와 AWS Cloud Map 연결합니다.

API 호출을 사용하여 인스턴스를 검색하는 네임스페이스의 경우 Route 53 상태 점검을 AWS Cloud Map 생성합니다. 하지만 상태 확인을 AWS Cloud Map 연결할 DNS 레코드는 없습니다. 상태 확인이 정상인지 확인하려면 Route 53 콘솔 또는 Amazon을 사용하여 모니터링을 구성할 수 CloudWatch 있습니다. Route 53 콘솔 사용에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 상태 확인 실패 시 알림 메시지를 받음을 참조하세요. 사용에 CloudWatch 대한 자세한 내용은 Amazon CloudWatch API 참조를 참조하십시오 PutMetricAlarm.

Route 53 상태 확인 비용에 대한 자세한 내용은 Route 53 요금을 참조하세요.

사용자 지정 상태 확인

인스턴스를 AWS Cloud Map 등록할 때 사용자 지정 상태 확인을 사용하도록 구성한 경우 타사 상태 확인 프로그램을 사용하여 리소스 상태를 평가해야 합니다. 사용자 지정 상태 확인은 다음과 같은 경우에 유용합니다.

- 인터넷을 통해 리소스를 사용할 수 없어 Route 53 상태 확인을 사용할 수 없는 경우. 예를 들어, Amazon VPC에 있는 인스턴스가 있다고 가정해 보겠습니다. 이 인스턴스에 대해 사용자 지정 상태 확인을 사용할 수 있습니다. 하지만 상태 확인이 작동하려면 상태 확인 검사기가 인스턴스와 동일한 VPC에 있어야 합니다.

- 리소스 위치와 상관없이 타사 상태 확인 프로그램을 사용하려는 경우

실패 임계값(Route 53 상태 확인만 해당)

Amazon Route 53에서 리소스의 현재 상태를 정상에서 비정상 또는 그 반대로 변경하기 위해 리소스가 통과 또는 실패해야 하는 연속 Route 53 상태 확인 횟수입니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 Amazon Route 53이 상태 확인이 정상인지 여부를 판단하는 방법을 참조하세요.

상태 확인 프로토콜(Route 53 상태 확인만 해당)

리소스 상태 확인을 위해 Amazon Route 53에서 사용하도록 하려는 방법입니다.

HTTP

Route 53이 TCP 연결을 설정하려고 시도합니다. 성공할 경우 Route 53는 HTTP 요청을 제출하고 2xx 또는 3xx 형식의 HTTP 상태 코드를 기다립니다.

HTTPS

Route 53이 TCP 연결을 설정하려고 시도합니다. 성공할 경우 Route 53는 HTTPS 요청을 제출하고 2xx 또는 3xx 형식의 HTTP 상태 코드를 기다립니다.

> **Important**
>
> HTTPS를 선택할 경우, 리소스가 TLS v1.0 이상을 지원해야 합니다.

Health check protocol(상태 확인 프로토콜) 값으로 HTTPS를 선택할 경우 추가 요금이 적용됩니다. 자세한 내용은 Route 53 요금을 참조하세요.

TCP

Route 53이 TCP 연결을 설정하려고 시도합니다.

자세한 내용은 Amazon Route 53이 상태 확인이 정상인지 여부를 판단하는 방법을 참조하세요.

상태 확인 경로(Route 53 HTTP 및 HTTPS 상태 확인만 해당)

상태 확인을 수행할 때 Amazon Route 53이 요청할 경로입니다(있는 경우). 경로는 `/docs/route53-health-check.html` 파일과 같은 모든 값이 될 수 있습니다. 리소스가 정상일 때 반환되는 값은 2xx 또는 3xx 형식의 HTTP 상태 코드입니다. 쿼리 문자열 파라미터를 포함해도 됩니다(예: `/welcome.html?language=jp&login=y`). AWS Cloud Map 콘솔에서는 앞에 슬래시(/) 문자를 자동으로 덧붙입니다.

태그

서비스에 추가할 태그를 하나 이상 지정할 수 있습니다. 태그는 AWS 리소스에 할당할 수 있는 선택적 레이블입니다. 각 태그는 키와 값으로 구성됩니다. 예를 들어 Key = Environment 및 Value = Production으로 태그를 정의할 수 있습니다. 태그를 사용하여 AWS 리소스를 분류하면 해당 리소스를 더 쉽게 관리할 수 있습니다.

태그가 생성된 후에는 언제든지 네임스페이스에서 태그를 업데이트하거나 제거할 수 있습니다. 자세한 내용은 AWS Cloud Map 리소스에 태그 지정을(를) 참조하세요.

AWS Cloud Map 서비스 업데이트

서비스를 업데이트하려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 네임스페이스 페이지에서 서비스를 편집하려는 네임스페이스를 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 편집할 서비스를 선택하고 편집을 클릭합니다.
5. 서비스: ***service-name*** 페이지에서 편집을 클릭합니다.
6. 서비스 편집 페이지에서 해당 값을 입력합니다.
7. 서비스 업데이트를 클릭합니다.

AWS CLI

- `update-service` 명령을 사용하여 서비스를 업데이트(*###* 값을 사용자 고유 값으로 대체)합니다.

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

출력:

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

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. update_service()로 서비스를 업데이트(### 값을 사용자 고유 값으로 대체)합니다.

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

예시 응답 출력

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

네임스페이스에서 서비스 보기

네임스페이스에서 생성한 서비스 목록을 보려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 나열하려는 서비스가 포함된 네임스페이스의 이름을 선택합니다.

AWS CLI

- `list-services` 명령을 사용하여 서비스를 나열합니다.

```
aws servicediscovery list-services
```

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 `servicediscovery`를 사용하세요.

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

3. `list_services()`를 사용하여 서비스를 나열하세요.

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

예시 응답 출력

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

AWS Cloud Map 서비스 삭제

서비스를 삭제하기 전에 해당 서비스를 사용해 등록한 모든 서비스 인스턴스를 등록 취소해야 합니다. 자세한 설명은 AWS Cloud Map 서비스 인스턴스 등록 취소 섹션을 참조하세요.

서비스를 삭제하려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 삭제하려는 서비스가 포함된 네임스페이스에 대해 해당 옵션을 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 삭제하려는 서비스에 대해 해당 옵션을 선택합니다.
5. 삭제을 선택합니다.
6. 서비스 삭제를 확인합니다.

AWS CLI

- delete-service 명령을 사용하여 서비스를 삭제(### 값을 사용자 고유 값으로 대체)합니다.

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

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. delete_service()로 서비스를 삭제(### 값을 사용자 고유 값으로 대체)합니다.

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

예시 응답 출력

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

AWS Cloud Map 서비스 인스턴스 사용

서비스 인스턴스에는 애플리케이션의 리소스(예: 웹 서버)를 찾는 방법에 대한 정보가 포함되어 있습니다. 인스턴스를 등록한 후에는 DNS 쿼리 또는 AWS Cloud Map DiscoverInstancesAPI 작업을 사용하여 인스턴스를 찾습니다.

주제

AWS Cloud Map 서비스 인스턴스 등록

서비스 인스턴스를 등록하려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 네임스페이스 페이지에서 서비스 인스턴스 등록을 위한 템플릿으로 사용하려는 서비스가 포함된 네임스페이스를 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 사용하려는 서비스를 선택합니다.
5. 서비스: ***service-name*** 페이지에서 서비스 인스턴스 등록을 선택합니다.
6. 서비스 인스턴스 등록 페이지에서 해당 값을 입력합니다. 자세한 설명은 서비스 인스턴스를 등록하거나 업데이트할 때 지정하는 값 섹션을 참조하세요.
7. 서비스 인스턴스 등록을 선택합니다.

AWS CLI

- `RegisterInstance` 요청을 제출하는 경우:
  - `ServiceId`에 지정된 서비스에서 정의한 각 DNS 레코드에 대해 해당 네임스페이스와 연결된 호스팅 영역에서 레코드가 생성되거나 업데이트됩니다.
  - 서비스에 `HealthCheckConfig`가 포함된 경우, 상태 확인 구성의 설정을 기반으로 상태 확인이 생성됩니다.
  - 모든 상태 확인은 새 레코드 또는 업데이트된 각 레코드와 연결됩니다.

register-instance 명령을 사용하여 서비스 인스턴스를 등록(### 값을 사용자 고유 값으로 대체)합니다.

```
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. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. RegisterInstance 요청을 제출하는 경우:

   - ServiceId에 지정된 서비스에서 정의한 각 DNS 레코드에 대해 해당 네임스페이스와 연결된 호스팅 영역에서 레코드가 생성되거나 업데이트됩니다.
   - 서비스에 HealthCheckConfig가 포함된 경우, 상태 확인 구성의 설정을 기반으로 상태 확인이 생성됩니다.
   - 모든 상태 확인은 새 레코드 또는 업데이트된 각 레코드와 연결됩니다.

   register_instance()로 서비스 인스턴스를 등록(### 값을 사용자 고유 값으로 대체)합니다.

```
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)
```

예시 응답 출력

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

서비스 인스턴스를 등록하거나 업데이트할 때 지정하는 값

서비스 인스턴스를 등록할 때 다음 값을 지정합니다.

값

- Instance type
- Service instance ID
- IPv4 address
- IPv6 address
- Port
- EC2 instance ID
- Custom attributes

인스턴스 유형

다음 각 인스턴스 유형은 선택한 구성에만 사용할 수 있습니다.

IP 주소

서비스 인스턴스와 연결한 리소스에 IP 주소를 사용하여 액세스할 수 있는 경우 이 옵션을 선택합니다.

세 가지 모든 네임스페이스 유형(HTTP, 퍼블릭 DNS, 프라이빗 DNS)에 이 옵션을 선택할 수 있습니다.

EC2 인스턴스

서비스 인스턴스와 연결한 리소스에 EC2 인스턴스를 통해 액세스할 수 있는 경우 이 옵션을 선택합니다.

이 옵션은 HTTP에 대해 선택할 수 있습니다.

다른 리소스에 대한 정보 식별

서비스 인스턴스와 연결한 리소스에 IP 주소 또는 EC2 인스턴스 이외의 값을 사용하여 액세스할 수 있는 경우 이 옵션을 선택합니다. Custom attributes(사용자 지정 속성)에 다른 값을 지정합니다.

세 가지 모든 네임스페이스 유형(HTTP, 퍼블릭 DNS, 프라이빗 DNS)에 이 옵션을 선택할 수 있습니다.

서비스 인스턴스 ID

인스턴스와 연결시키기 원하는 식별자입니다. 유념할 사항:

- 새 인스턴스를 등록하려면 동일한 서비스를 사용해 등록한 인스턴스에 고유한 값을 지정해야 합니다.
- 서비스 인스턴스 ID로 지정된 서비스에 SRV 레코드에 대한 설정이 포함되어 있는 경우, 서비스 인스턴스 ID 값이 SRV 레코드 값의 일부로 자동으로 포함됩니다. 자세한 내용은 이전 섹션 서비스를 생성할 때 지정하는 값의 레코드 형식을 참조하세요.
- 기존 인스턴스를 프로그래밍 방식으로 업데이트할 수 있습니다. 를 호출하고 RegisterInstance, 서비스 인스턴스 ID 및 서비스 ID의 값을 지정하고, 서비스 인스턴스의 새 설정을 지정합니다. 인스턴스를 처음 등록할 때 상태 확인을 AWS Cloud Map 생성한 경우 이전 상태 확인을 AWS Cloud Map 삭제하고 새 상태 확인을 생성합니다.

> **Note**
>
> 상태 확인은 바로 삭제되지 않기 때문에 예를 들어 Amazon Route 53 `ListHealthChecks` 요청을 제출한 경우 상태 확인이 잠시 나타납니다.

IPv4 주소

애플리케이션이 이 서비스 인스턴스와 연결된 리소스에 액세스할 수 있는 IPv4 IP 주소입니다(있는 경우).

IPv6 주소

애플리케이션이 이 서비스 인스턴스와 연결된 리소스에 액세스할 수 있는 IPv6 IP 주소입니다(있는 경우).

포트

이 서비스 인스턴스와 연결된 리소스에 액세스하려면 애플리케이션에 포함되어 있어야 하는 포트입니다(있는 경우). 포트는 서비스에 SRV 레코드 또는 Amazon Route 53 상태 확인이 포함된 경우 필수입니다.

EC2 인스턴스

리소스에 대한 EC2 인스턴스 ID 형식의 인스턴스 ID입니다.

사용자 지정 속성

리소스와 연결하고자 하는 키-값 페어를 지정합니다(있는 경우).

사용자 지정 속성은 최대 30개까지 추가할 수 있습니다. 유념할 사항:

- 키와 값을 둘 다 지정해야 합니다.
- 키는 길이가 최대 255자로, a-z, A-Z, 0-9 및 기타 인쇄 가능한 ASCII 문자 33~126개(10진수)를 포함할 수 있습니다. 공백, 탭 및 기타 공백 문자는 허용되지 않습니다.
- 값은 길이가 최대 1,024자로, a-z, A-Z, 0-9, 기타 인쇄 가능한 ASCII 문자 33~126개(10진수), 공백 및 탭을 포함할 수 있습니다.

AWS Cloud Map 서비스 인스턴스 업데이트

업데이트하려는 값에 따라 다음 두 가지 방법으로 서비스 인스턴스를 업데이트할 수 있습니다.

- 값 업데이트: 사용자 지정 속성을 포함하여 서비스 인스턴스를 등록할 때 지정한 모든 값을 업데이트하려면 서비스 인스턴스를 재등록하고 모든 값을 다시 지정합니다. 서비스 인스턴스의 세부 정보 업데이트을 참조하십시오.
- 사용자 지정 속성만 업데이트: 서비스 인스턴스의 사용자 지정 속성만 업데이트하려는 경우 인스턴스를 재등록할 필요가 없습니다. 해당 값만 업데이트하면 됩니다. 서비스 인스턴스의 사용자 지정 속성 업데이트을 참조하십시오.

서비스 인스턴스의 세부 정보 업데이트

서비스 인스턴스를 업데이트하려면

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 네임스페이스 페이지에서 서비스 인스턴스를 등록하는 데 원래 사용한 서비스가 포함된 네임스페이스를 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 서비스 인스턴스를 등록하는 데 사용한 서비스를 선택합니다.
5. 서비스: ***service-name*** 페이지에서 업데이트하려는 서비스 인스턴스의 ID를 복사합니다.
6. 서비스 인스턴스 등록을 선택합니다.
7. 서비스 인스턴스 등록 페이지에서 5단계에서 복사한 ID를 서비스 인스턴스 ID에 붙여넣습니다.
8. 서비스 인스턴스에 적용하려는 기타 값을 모두 입력합니다. 서비스 인스턴스에 대한 이전 값은 유지되지 않습니다. 자세한 설명은 서비스 인스턴스를 등록하거나 업데이트할 때 지정하는 값 섹션을 참조하세요.
9. 서비스 인스턴스 등록을 선택합니다.

서비스 인스턴스의 사용자 지정 속성 업데이트

서비스 인스턴스에 대한 사용자 지정 속성만 업데이트하려면

1. AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 네임스페이스 페이지에서 서비스 인스턴스를 등록하는 데 원래 사용한 서비스가 포함된 네임스페이스를 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 서비스 인스턴스를 등록하는 데 사용한 서비스를 선택합니다.
5. 서비스: ***service-name*** 페이지에서 업데이트하려는 서비스 인스턴스의 이름을 선택합니다.
6. 사용자 지정 속성 섹션에서 편집을 선택합니다.
7. 서비스 인스턴스 편집: ***instance-name*** 페이지에서 사용자 지정 속성을 추가, 제거 또는 업데이트합니다. 기존 속성의 키와 값을 모두 업데이트할 수 있습니다.

8. 서비스 인스턴스 업데이트를 선택합니다.

AWS Cloud Map 서비스 인스턴스 보기

서비스를 사용하여 등록한 서비스 인스턴스 목록을 보려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 서비스 인스턴스를 나열하려는 서비스가 포함된 네임스페이스의 이름을 선택합니다.
4. 서비스 인스턴스를 생성하는 데 사용한 서비스의 이름을 선택합니다.

AWS CLI

- `list-instances` 명령을 사용하여 서비스 인스턴스를 나열합니다(*###* 값을 자신의 것으로 대체).

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

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우 Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 `servicediscovery`를 사용하세요.

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

3. `list_instances()`로 서비스 인스턴스를 나열합니다(*###* 값을 자체 값으로 대체).

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

예시 응답 출력

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

AWS Cloud Map 서비스 인스턴스 등록 취소

서비스를 삭제하기 전에 해당 서비스를 사용해 등록한 모든 서비스 인스턴스를 등록 취소해야 합니다.

서비스 인스턴스를 등록 취소하려면 다음 절차를 수행합니다.

AWS Management Console

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 탐색 창에서 네임스페이스를 선택합니다.
3. 등록 취소하려는 서비스 인스턴스가 포함된 네임스페이스에 대해 해당 옵션을 선택합니다.
4. 네임스페이스: ***namespace-name*** 페이지에서 서비스 인스턴스를 등록하는 데 사용한 서비스에 대해 해당 옵션을 선택합니다.
5. 서비스: ***Service-name*** 페이지에서 등록 취소하려는 서비스 인스턴스에 대해 해당 옵션을 선택합니다.
6. 등록 취소(Deregister)를 선택합니다.
7. 서비스 인스턴스 등록 취소를 확인합니다.

AWS CLI

- deregister-instance 명령을 사용하여 서비스 인스턴스를 등록 취소(### 값을 사용자 고유 값으로 대체)합니다. 이 명령은 Amazon Route 53 DNS 레코드 및 지정된 인스턴스에 대해 AWS Cloud Map 생성된 모든 상태 확인을 삭제합니다.

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

AWS SDK for Python (Boto3)

1. 아직 Boto3이 설치되지 않은 경우, Boto3을 사용하여 여기에서 설치, 구성, 사용에 대한 지침을 찾을 수 있습니다.
2. Boto3을 가져와서 서비스로 servicediscovery를 사용하세요.

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

3. deregister-instance()로 서비스 인스턴스를 등록 취소(### 값을 사용자 고유 값으로 대체)합니다. 이 명령은 Amazon Route 53 DNS 레코드 및 지정된 인스턴스에 대해 AWS Cloud Map 생성된 모든 상태 확인을 삭제합니다.

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

예시 응답 출력

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

AWS Cloud Map 콘솔에서 AWS Cloud Map 사용할 수 없는 기능

콘솔에서는 다음 AWS Cloud Map 기능을 사용할 수 없습니다. AWS Cloud Map 이러한 기능을 사용하려면 프로그래밍 방식을 사용하여 액세스해야 합니다. AWS Cloud Map

서비스 인스턴스 등록 시 Route 53 별칭 레코드 생성

콘솔을 사용하여 서비스 인스턴스를 등록하는 경우, 트래픽을 Elastic Load Balancing(ELB) 로드 밸런서로 라우팅하는 별칭 레코드를 생성할 수 없습니다. 유념할 사항:

- 서비스를 생성할 때 `RoutingPolicy`를 `WEIGHTED`로 지정해야 합니다. 콘솔에서 이와 같이 지정할 수 있습니다. 자세한 설명은 AWS Cloud Map 서비스 생성 섹션을 참조하세요.

  API를 사용하여 서비스를 생성하는 방법에 대한 자세한 내용은 AWS Cloud Map AWS Cloud Map API 참조를 참조하십시오 CreateService.
- 인스턴스를 등록할 때 `AWS_ALIAS_DNS_NAME` 속성을 포함시켜야 합니다. 자세한 내용을 알아보려면 AWS Cloud Map API 참조의 RegisterInstance(을)를 참조하세요.

사용자 지정 상태 확인에 대한 초기 상태 지정

사용자 지정 상태 확인이 포함된 서비스를 사용하여 인스턴스를 등록하는 경우 사용자 지정 상태 확인에 대한 초기 상태를 지정할 수 없습니다. 기본적으로 사용자 지정 상태 확인의 초기 상태는 정상입니다. 초기 상태를 이상 있음으로 설정하려면 인스턴스를 프로그래밍 방식으로 등록하고 `AWS_INIT_HEALTH_STATUS` 속성을 포함시킵니다. 자세한 내용을 알아보려면 AWS Cloud Map API 참조의 RegisterInstance(을)를 참조하세요.

완료되지 않은 작업의 상태 가져오기

네임스페이스를 생성한 후 네임스페이스 생성이 완료되지 않았는데 브라우저 창을 닫은 경우, 콘솔에서 현재 상태를 확인할 수 있는 방법이 없습니다. 이러한 경우에는 ListOperations를 사용하여 상태를 확인할 수 있습니다. 자세한 내용을 알아보려면 AWS Cloud Map API 참조의 ListOperations(을)를 참조하세요.

튜토리얼

다음 자습서에서는 AWS Cloud Map 네임스페이스를 사용하여 일반적인 작업을 수행하는 방법을 보여 줍니다.

주제

- 자습서: DNS 쿼리를 통한 AWS Cloud Map 서비스 검색 사용
- 자습서: 사용자 지정 속성과 함께 AWS Cloud Map 서비스 검색 사용

자습서: DNS 쿼리를 통한 AWS Cloud Map 서비스 검색 사용

이 자습서에서는 두 개의 백엔드 서비스가 있는 마이크로서비스 아키텍처를 시뮬레이션합니다. DNS 쿼리를 사용하여 첫 번째 서비스를 검색할 수 있습니다. 두 번째 서비스는 API로만 검색할 수 있습니다. AWS Cloud Map

> **Note**
>
> 이 자습서에서는 도메인 이름 및 IP 주소와 같은 리소스 세부 정보는 시뮬레이션 용도로만 사용됩니다. 인터넷으로는 해결할 수 없습니다.

필수 조건

이 자습서를 성공적으로 완료하려면 다음 사전 요구 사항을 충족해야 합니다.

가입하여 AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 여세요.
2. 온라인 지시 사항을 따르세요.

   등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

   에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한

을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요. AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.
2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자 (콘솔) 에 대한 가상 MFA 디바이스 활성화를 참조하십시오.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

   를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

관리 액세스 권한이 있는 사용자로 로그인

- IAM IDentity Center 사용자로 로그인하려면 IAM IDentity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인을 참조하십시오.AWS 로그인

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

설치 AWS Command Line Interface

를 아직 설치하지 않은 경우 최신 버전 설치 또는 업데이트의 단계에 AWS CLI따라 설치하십시오. AWS Command Line Interface

이 자습서에서는 명령을 실행할 셸 또는 명령줄 터미널이 필요합니다. Linux 및 macOS에서는 선호하는 셸과 패키지 관리자를 사용합니다.

> **Note**
>
> Windows에서는 Lambda와 함께 일반적으로 사용하는 일부 Bash CLI 명령(예:`zip`)은 운영 체제의 기본 제공 터미널에서 지원되지 않습니다. Ubuntu와 Bash의 Windows 통합 버전을 가져오려면 Linux용 Windows Subsystem을 설치합니다.

dig 유틸리티에 액세스할 수 있습니다.

이 자습서에는 `dig` DNS 조회 유틸리티 명령이 있는 로컬 환경이 필요합니다. `dig`명령에 대한 자세한 내용은 dig - DNS 조회 유틸리티를 참조하십시오.

1단계: 네임스페이스 생성 AWS Cloud Map

이 단계에서는 퍼블릭 AWS Cloud Map 네임스페이스를 생성합니다. AWS Cloud Map 사용자를 대신하여 이 동일한 이름으로 Route 53 호스팅 영역을 생성합니다. 이렇게 하면 퍼블릭 DNS 레코드를 사용하거나 AWS Cloud Map API 호출을 사용하여 이 네임스페이스에 생성된 서비스 인스턴스를 검색할 수 있습니다.

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. Create namespace(네임스페이스 생성)를 선택합니다.
3. 네임스페이스 이름에 대해 지정합니다. `cloudmap-tutorial.com`

> **Note**
>
> 프로덕션 환경에서 이 이름을 사용하려는 경우 소유했거나 액세스할 수 있었던 도메인의 이름을 지정했는지 확인하는 것이 좋습니다. 하지만 이 자습서에서는 사용 중인 실제 도메인일 필요는 없습니다.

4. (선택 사항) 네임스페이스 설명에는 네임스페이스의 용도에 대한 설명을 지정하십시오.
5. 인스턴스 검색의 경우 API 호출 및 퍼블릭 DNS 쿼리를 선택합니다.
6. 나머지 기본값은 그대로 두고 네임스페이스 생성을 선택합니다.

2단계: 서비스 생성 AWS Cloud Map

이 단계에서는 두 개의 서비스를 생성합니다. 퍼블릭 DNS 및 API 호출을 사용하여 첫 번째 서비스를 검색할 수 있습니다. 두 번째 서비스는 API 호출로만 검색할 수 있습니다.

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 네임스페이스를 선택하여 생성한 네임스페이스를 나열합니다.
3. 네임스페이스 목록에서 네임스페이스를 선택하고 세부 정보 보기를 선택합니다. **`cloudmap-tutorial.com`**
4. 서비스 섹션에서 서비스 생성을 선택하고 다음 작업을 수행하여 첫 번째 서비스를 생성합니다.

   a. 서비스 이름에 `public-service`를 입력합니다. AWS Cloud Map 생성한 DNS 레코드에 서비스 이름이 적용됩니다. 사용되는 형식은 입니다*`<service-name>`*.*`<namespace-name>`*.

   b. 서비스 검색 구성에서 API 및 DNS를 선택합니다.

   c. DNS 구성 섹션의 라우팅 정책에서 다중 값 응답 라우팅을 선택합니다.

> **Note**
>
> 선택한 후 콘솔은 이를 MULTIVALUE로 변환합니다. 사용 가능한 라우팅 옵션에 대한 자세한 내용은 Route 53 개발자 안내서의 라우팅 정책 선택을 참조하십시오.

d. 나머지 기본값은 그대로 두고 Create service (서비스 생성) 를 선택하면 네임스페이스 세부 정보 페이지로 돌아갑니다.

5. [서비스] 섹션에서 [Create service] 를 선택하고 다음 작업을 수행하여 두 번째 서비스를 생성합니다.

   a. 서비스 이름에 `backend-service`를 입력합니다.

   b. 서비스 검색 구성의 경우 API만을 선택합니다.

   c. 나머지 기본값은 그대로 두고 서비스 생성을 선택합니다.

3단계: AWS Cloud Map 서비스 인스턴스 생성

이 단계에서는 네임스페이스의 각 서비스에 대해 하나씩, 두 개의 서비스 인스턴스를 만듭니다.

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 네임스페이스 목록에서 1단계에서 생성한 네임스페이스를 선택하고 세부 정보 보기를 선택합니다.
3. 네임스페이스 세부 정보 페이지의 서비스 목록에서 서비스를 선택하고 세부 정보 보기를 선택합니다. **`public-service`**
4. 서비스 인스턴스 섹션에서 서비스 인스턴스 등록을 선택하고 다음 작업을 수행하여 첫 번째 서비스 인스턴스를 생성합니다.

   a. 서비스 인스턴스 ID에 대해 지정합니다`first`.

   b. IPv4 주소의 경우 지정합니다. `192.168.2.1`

   c. 나머지 기본값은 그대로 두고 서비스 인스턴스 등록을 선택합니다.

5. 페이지 상단의 이동 경로를 사용하여 cloudmap-tutorial.com을 선택하여 네임스페이스 세부 정보 페이지로 다시 이동합니다.
6. 네임스페이스 세부 정보 페이지의 서비스 목록에서 백엔드 서비스 서비스를 선택하고 세부 정보 보기를 선택합니다.

7. 서비스 인스턴스 섹션에서 서비스 인스턴스 등록을 선택하고 다음 작업을 수행하여 두 번째 서비스 인스턴스를 생성합니다.

   a. 서비스 인스턴스 ID의 경우 두 번째 서비스 인스턴스임을 `second` 나타내도록 지정합니다.
   b. 인스턴스 유형에서 다른 리소스의 식별 정보를 선택합니다.
   c. 사용자 지정 속성의 경우 키와 `backend` 값을 사용하여 `service-name` 키-값 쌍을 추가합니다.
   d. 서비스 인스턴스 등록을 선택합니다.

4단계: 서비스 인스턴스 검색 AWS Cloud Map

이제 AWS Cloud Map 네임스페이스, 서비스, 서비스 인스턴스가 생성되었으므로 인스턴스를 검색하여 모든 것이 제대로 작동하는지 확인할 수 있습니다. `dig`명령을 사용하여 퍼블릭 DNS 설정을 확인하고 AWS Cloud Map API를 사용하여 백엔드 서비스를 확인합니다. `dig`명령에 대한 자세한 내용은 dig - DNS 조회 유틸리티를 참조하십시오.

1. AWS Management Console 로그인하고 https://console.aws.amazon.com/route53/ 에서 Route 53 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 Hosted Zones(호스팅 영역)를 선택합니다.
3. cloudmap-tutorial.com 호스팅 영역을 선택합니다. 그러면 호스팅 영역 세부 정보가 별도의 창에 표시됩니다. 다음 단계에서 사용할 것이므로 호스팅 영역과 연결된 이름 서버를 기록해 두십시오.
4. dig 명령어와 호스팅 영역의 Route 53 이름 서버 중 하나를 사용하여 서비스 인스턴스의 DNS 레코드를 쿼리합니다.

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

   `ANSWER SECTION`출력에는 서비스에 연결한 IPv4 주소가 표시되어야 합니다`public-service`.

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

5. 를 사용하여 두 번째 서비스 인스턴스의 속성을 쿼리합니다. AWS CLI

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

   출력에는 서비스에 연결한 속성이 키-값 쌍으로 표시됩니다.

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

5단계: 리소스 정리

자습서를 완료한 후에는 리소스를 삭제할 수 있습니다. AWS Cloud Map 먼저 서비스 인스턴스를 정리하고, 서비스를, 마지막으로 네임스페이스를 정리하는 등 역순으로 정리해야 합니다. AWS Cloud Map 이 단계를 수행하면 사용자를 대신하여 Route 53 리소스를 정리합니다.

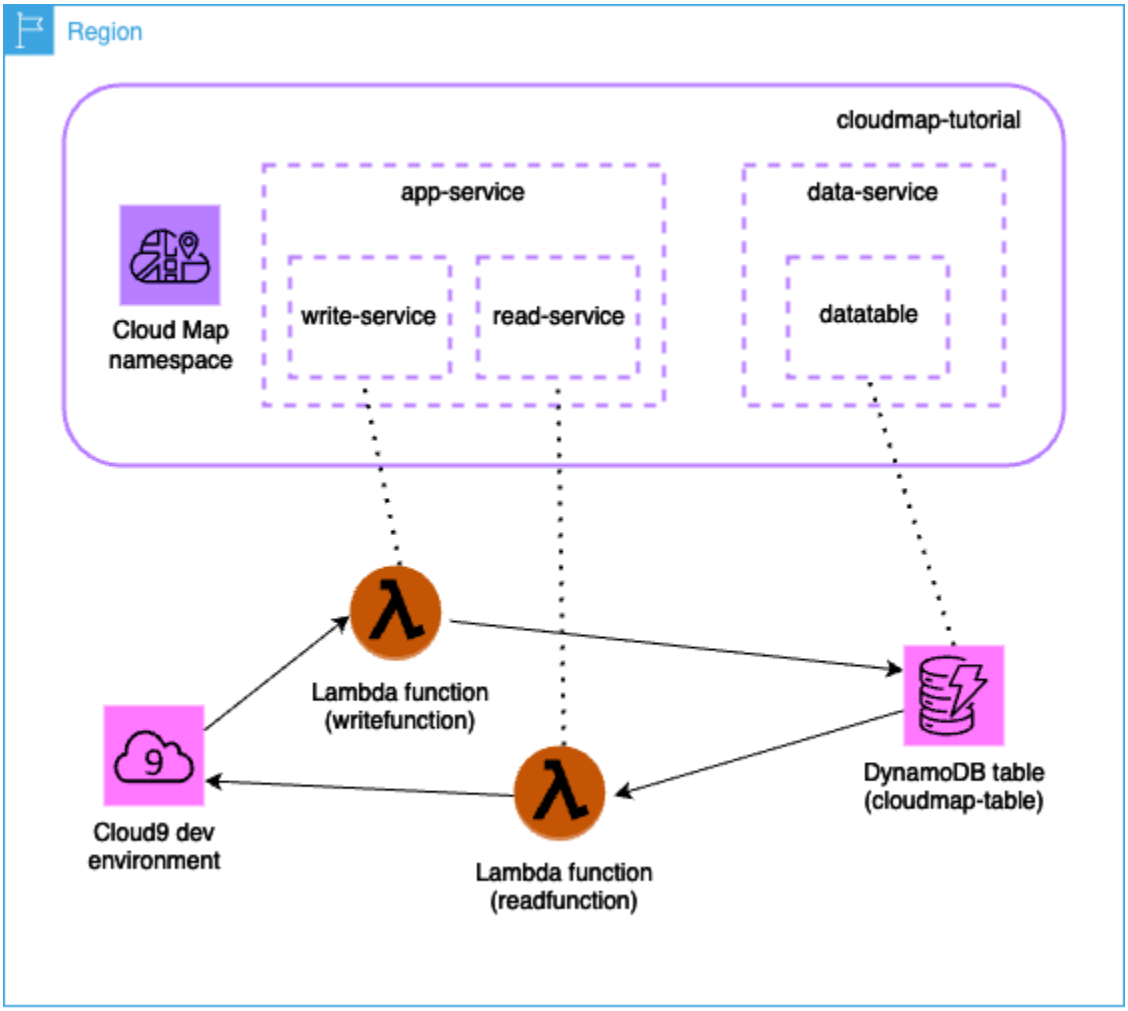
1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 네임스페이스 목록에서 네임스페이스를 선택하고 세부 정보 **cloudmap-tutorial.com** 보기를 선택합니다.
3. 네임스페이스 세부 정보 페이지의 서비스 목록에서 서비스를 선택하고 세부 정보 보기를 선택합니다. **public-service**
4. 서비스 인스턴스 섹션에서 인스턴스를 선택하고 등록 **first** 취소를 선택합니다.
5. 페이지 상단의 이동 경로를 사용하여 cloudmap-tutorial.com을 선택하여 네임스페이스 세부 정보 페이지로 다시 이동합니다.
6. 네임스페이스 세부 정보 페이지의 서비스 목록에서 공용 서비스 서비스를 선택하고 삭제를 선택합니다.
7. 에 대해 3-6단계를 반복합니다. `backend-service`
8. 왼쪽 탐색 메뉴에서 네임스페이스를 선택합니다.
9. `cloudmap-tutorial.com`네임스페이스를 선택하고 삭제를 선택합니다.

> **Note**
>
> 사용자를 대신하여 Route 53 리소스를 AWS Cloud Map 정리하지만 Route 53 콘솔로 이동하여 `cloudmap-tutorial.com` 호스팅 영역이 삭제되었는지 확인할 수 있습니다.

자습서: 사용자 지정 속성과 함께 AWS Cloud Map 서비스 검색 사용

이 가이드에서는 API를 사용하여 검색할 수 있는 사용자 지정 속성과 함께 AWS Cloud Map 서비스 검색을 사용하는 방법을 보여줍니다. AWS Cloud Map 이 자습서에서는 두 개의 Lambda 함수를 사용하여 DynamoDB 테이블에 데이터를 쓴 다음 테이블에서 읽는 AWS Cloud9 환경에서 클라이언트 애플리케이션을 생성하는 방법을 안내합니다. Lambda 함수 및 DynamoDB 테이블은 서비스 인스턴스로 등록됩니다. AWS Cloud Map 클라이언트 애플리케이션 및 Lambda 함수의 코드는 사용자 지정 속성을 AWS Cloud Map 사용하여 작업 수행에 필요한 리소스를 검색합니다.

다음 다이어그램은 이 자습서에서 사용하는 상위 수준 아키텍처를 보여줍니다.



> **Important**
>
> 워크숍 중에 AWS 리소스를 생성하게 되며, 이 경우 AWS 계정에 비용이 발생합니다. 비용을 최소화하려면 워크숍을 마치자마자 리소스를 정리하는 것이 좋습니다.

필수 조건

이 자습서를 성공적으로 완료하려면 다음 사전 요구 사항을 충족해야 합니다.

가입하여 AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 여세요.
2. 온라인 지시 사항을 따르세요.

   등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

   에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요. AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.
2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자 (콘솔) 에 대한 가상 MFA 디바이스 활성화를 참조하십시오.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

관리 액세스 권한이 있는 사용자로 로그인

- IAM IDentity Center 사용자로 로그인하려면 IAM IDentity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

  IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털 로그인을 참조하십시오.AWS 로그인

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

1단계: 네임스페이스 생성 AWS Cloud Map

이 단계에서는 네임스페이스를 생성합니다. AWS Cloud Map 네임스페이스는 애플리케이션의 서비스를 그룹화하는 데 사용되는 구조입니다. 네임스페이스를 생성할 때 리소스를 검색할 수 있는 방법을 지정합니다. 이 자습서에서는 사용자 지정 속성을 사용하는 API 호출을 통해 이 네임스페이스에 생성된 리소스를 검색할 수 있습니다. AWS Cloud Map 이에 대해서는 이후 단계에서 자세히 알아볼 것입니다.

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Management Console 로그인하고 AWS Cloud Map 콘솔을 엽니다.
2. Create namespace(네임스페이스 생성)를 선택합니다.
3. 네임스페이스 이름에 대해 지정합니다. `cloudmap-tutorial`
4. (선택 사항) 네임스페이스 설명에는 네임스페이스를 사용하려는 대상에 대한 설명을 지정합니다.
5. 인스턴스 검색의 경우 API 호출을 선택합니다.
6. 나머지 기본값은 그대로 두고 네임스페이스 생성을 선택합니다.

2단계: DynamoDB 테이블 생성

이 단계에서는 이 자습서의 뒷부분에서 만든 샘플 애플리케이션의 데이터를 저장하고 검색하는 데 사용되는 DynamoDB 테이블을 생성합니다.

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/dynamodb/ 에서 DynamoDB 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 테이블, 테이블 생성을 선택합니다.
3. 테이블 생성 페이지에서 다음 작업을 수행합니다.
   a. 테이블 이름에 대해 지정합니다`cloudmap-table`.
   b. 파티션 키에 대해 지정합니다`id`.
   c. 나머지 기본값은 그대로 두고 테이블 만들기를 선택합니다.

3단계: AWS Cloud Map 데이터 서비스 만들기

이 단계에서는 서비스를 생성한 다음 마지막 단계에서 생성한 DynamoDB 테이블을 AWS Cloud Map 서비스 인스턴스로 등록합니다.

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 네임스페이스 목록에서 네임스페이스를 선택하고 세부 정보 **`cloudmap-tutorial`** 보기를 선택합니다.
3. 서비스 섹션에서 서비스 생성을 선택하고 다음을 수행합니다.
   a. 서비스 이름에 `data-service`를 입력합니다.
   b. 나머지 기본값은 그대로 두고 서비스 생성을 선택합니다.
4. 서비스 섹션에서 서비스를 선택하고 세부 정보 보기를 선택합니다. `data-service`
5. 서비스 인스턴스 섹션에서 서비스 인스턴스 등록을 선택합니다.
6. 서비스 인스턴스 등록 페이지에서 다음을 수행합니다.
   a. 인스턴스 유형에서 다른 리소스의 식별 정보를 선택합니다.
   b. 서비스 인스턴스 ID에 대해 지정합니다`data-instance`.
   c. 사용자 지정 특성 섹션에서 다음 키-값 쌍을 지정합니다.
      - 키 **=`name`**, 값 = `datatable`

- 키 =`tablename`, 값 = `cloudmap`

d. 속성이 아래 이미지와 일치하는지 확인하고 서비스 인스턴스 등록을 선택합니다.

4단계: 실행 역할 생성 AWS Lambda

이 단계에서는 다음 단계에서 생성한 AWS Lambda 함수가 사용하는 IAM 역할을 생성합니다. 이 IAM 역할은 `cloudmap-role` 이 자습서에서만 사용되며 나중에 삭제할 수 있으므로 역할 이름을 지정하고 권한 경계를 생략할 수 있습니다.

Lambda용 서비스 역할을 생성하려면 (IAM 콘솔)

1. https://console.aws.amazon.com/iam/ 에서 AWS Management Console 로그인하고 IAM 콘솔을 엽니다.
2. IAM 콘솔의 탐색 창에서 역할을 선택하고 역할 생성을 선택합니다.
3. 신뢰할 수 있는 엔터티 유형에 AWS 서비스를 선택합니다.
4. 서비스 또는 사용 사례의 경우 Lambda를 선택한 다음 Lambda 사용 사례를 선택합니다.
5. 다음을 선택합니다.
6. `PowerUserAccess`정책을 검색하고 옆의 상자를 선택한 후 [Next] 를 선택합니다.
7. 다음을 선택합니다.
8. 역할 이름에 대해 지정합니다`cloudmap-tutorial-role`.
9. 역할을 검토한 다음 역할 생성을 선택합니다.

5단계: 데이터를 쓰는 Lambda 함수 생성

이 단계에서는 API를 사용하여 AWS Cloud Map 생성한 서비스를 쿼리하여 DynamoDB 테이블에 데이터를 쓰는 Lambda 함수를 생성합니다. AWS Cloud Map

1. AWS Management Console 로그인하고 https://console.aws.amazon.com/lambda/ 에서 콘솔을 엽니다. AWS Lambda
2. 왼쪽 탐색창에서 함수, 함수 생성을 선택합니다.
3. 함수 생성 페이지에서 다음을 수행하십시오.

   a. 새로 작성을 선택합니다.

   b. 함수 이름에 대해 지정합니다`writefunction`.

   c. 런타임에서 선택합니다`Python 3.12`.

   d. 아키텍처에서 을 선택합니다`x86_64`.

   e. 권한 섹션에서 다음을 수행합니다.

      i. 기본 실행 역할 변경 옵션을 확장하고 기존 역할 사용을 선택합니다.

      ii. 기존 역할의 경우 드롭다운 메뉴를 사용하여 에서 생성한 IAM 역할을 선택합니다. 4단계: 실행 역할 생성 AWS Lambda

      iii. 나머지 기본값은 그대로 두고 함수 생성을 선택합니다.

   f. 코드 탭의 코드 소스 섹션에서 다음 Python 코드를 반영하도록 예제 코드를 업데이트합니다. DynamoDB 테이블용으로 생성한 AWS Cloud Map 서비스 인스턴스에 연결한 `datatable` 사용자 지정 속성을 지정하고 있다는 점에 유의하십시오.

```
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',
        QueryParameters={ 'name': 'datatable' })

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

    dynamodbclient = boto3.resource('dynamodb')

    table = dynamodbclient.Table('cloudmap-table')

    response = table.put_item(
```

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

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

g. Deploy를 선택하여 함수를 업데이트하십시오.

6단계: AWS Cloud Map 앱 서비스 생성

이 단계에서는 서비스를 생성한 다음 Lambda write 함수를 AWS Cloud Map 서비스 인스턴스로 등록합니다.

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 왼쪽 탐색창에서 네임스페이스를 선택합니다.
3. 네임스페이스 목록에서 네임스페이스를 선택하고 세부 정보 보기를 **cloudmap-tutorial** 선택합니다.
4. 서비스 섹션에서 서비스 생성을 선택하고 다음을 수행합니다.
   a. 서비스 이름에 app-service를 입력합니다.
   b. 나머지 기본값은 그대로 두고 서비스 생성을 선택합니다.
5. 서비스 섹션에서 서비스를 선택하고 세부 정보 보기를 선택합니다. app-service
6. 서비스 인스턴스 섹션에서 서비스 인스턴스 등록을 선택합니다.
7. 서비스 인스턴스 등록 페이지에서 다음을 수행합니다.
   a. 인스턴스 유형에서 다른 리소스의 식별 정보를 선택합니다.
   b. 서비스 인스턴스 ID에 대해 지정합니다write-instance.
   c. 사용자 지정 특성 섹션에서 다음 키-값 쌍을 지정합니다.
      - 키 =**name**, 값 = writeservice
      - 키 =function, 값 = writefunction
   d. 속성이 아래 이미지와 일치하는지 확인하고 서비스 인스턴스 등록을 선택합니다.

7단계: 데이터를 읽는 Lambda 함수 생성

이 단계에서는 생성한 DynamoDB 테이블에 데이터를 쓰는 Lambda 함수를 생성합니다.

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/lambda/ 에서 콘솔을 엽니다. AWS Lambda
2. 왼쪽 탐색창에서 함수, 함수 생성을 선택합니다.
3. 함수 생성 페이지에서 다음을 수행하십시오.
   a. 새로 작성을 선택합니다.
   b. 함수 이름에 대해 지정합니다`readfunction`.
   c. 런타임에서 선택합니다`Python 3.12`.
   d. 아키텍처에서 을 선택합니다`x86_64`.
   e. 권한 섹션에서 다음을 수행합니다.
      i. 기본 실행 역할 변경 옵션을 확장하고 기존 역할 사용을 선택합니다.
      ii. 기존 역할의 경우 드롭다운 메뉴를 사용하여 에서 생성한 IAM 역할을 선택합니다. 4단계: 실행 역할 생성 AWS Lambda
      iii. 나머지 기본값은 그대로 두고 함수 생성을 선택합니다.
   f. 코드 탭의 코드 소스 섹션에서 다음 Python 코드를 반영하도록 예제 코드를 업데이트합니다.

```
import json
import boto3

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

```
    response = serviceclient.discover_instances(NamespaceName='cloudmap-
tutorial', ServiceName='data-service', QueryParameters={ 'name': 'datatable' })

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

    dynamodbclient = boto3.resource('dynamodb')

    table = dynamodbclient.Table('cloudmap-table')

    response = table.get_item(Key={'id': event})

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

g. Deploy를 선택하여 함수를 업데이트합니다.

8단계: AWS Cloud Map 서비스 인스턴스 생성

이 단계에서는 Lambda 읽기 함수를 이전에 생성한 서비스의 서비스 인스턴스로 `app-service` 등록합니다.

1. https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 왼쪽 탐색창에서 네임스페이스를 선택합니다.
3. 네임스페이스 목록에서 네임스페이스를 선택하고 세부 정보 보기를 **`cloudmap-tutorial`** 선택합니다.
4. 서비스 섹션에서 서비스를 선택하고 세부 정보 보기를 **`app-service`** 선택합니다.
5. 서비스 인스턴스 섹션에서 서비스 인스턴스 등록을 선택합니다.
6. 서비스 인스턴스 등록 페이지에서 다음을 수행합니다.

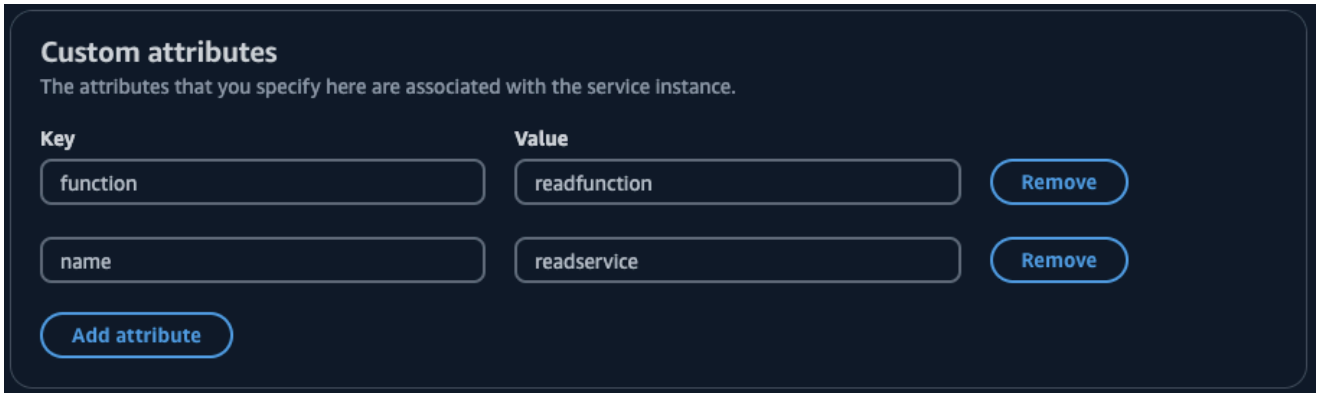
   a. 인스턴스 유형에서 다른 리소스의 식별 정보를 선택합니다.

   b. 서비스 인스턴스 ID에 대해 지정합니다`read-instance`.

   c. 사용자 지정 특성 섹션에서 다음 키-값 쌍을 지정합니다.

      - 키 =**`name`**, 값 = `readservice`
      - 키 =`function`, 값 = `readfunction`

   d. 속성이 아래 이미지와 일치하는지 확인하고 서비스 인스턴스 등록을 선택합니다.



9단계: 개발 환경 만들기

AWS Cloud9 에서 관리하는 통합 개발 환경 (IDE) 입니다. AWS AWS Cloud9 IDE는 동적 프로그래밍에 필요한 소프트웨어와 도구를 제공합니다. 이 단계에서는 AWS Cloud9 환경을 만들고 API로 프로그래밍할 수 있도록 구성합니다. AWS SDK for Python (Boto3) AWS

1. AWS Management Console 로그인하고 https://console.aws.amazon.com/cloud9/ 에서 AWS Cloud9 콘솔을 엽니다.
2. 왼쪽 탐색 메뉴에서 내 환경을 선택한 다음 환경 만들기를 선택합니다.
3. 환경 만들기 페이지에서 다음을 수행하여 개발 환경을 만드십시오.

   a. 이름에는 를 사용하십시오`cloudmap-tutorial`.
   b. 환경 유형에서 새 EC2 인스턴스를 선택합니다.
   c. 인스턴스 유형으로는 t2.micro를 선택합니다.
   d. 플랫폼의 경우 드롭다운 메뉴를 사용하여 우분투 서버 22.04 LTS를 선택합니다.
   e. 나머지 기본 선택은 그대로 두고 [Create] 를 선택합니다.
4. AWS Cloud9 환경이 생성되면 환경을 선택하고 Cloud9에서 열기를 선택합니다. `cloudmap-tutorial` 그러면 새 탭에서 개발 환경이 열리고 작업할 수 있는 bash 셸이 제공됩니다.

   > **Important**
   >
   > AWS Cloud9 환경을 여는 데 문제가 있는 경우 AWS Cloud9 사용 설명서의AWS Cloud9 문제 해결: 환경을 열 수 없음을 참조하십시오.

5. bash 셸을 사용하여 다음 명령을 실행하여 환경을 구성합니다.

   a. 환경 업데이트.

```
sudo apt-get -y update
```

b. 설치되어 python3 있는지 확인하세요.

```
python3 --version
```

c. 환경에 Boto3 패키지를 설치합니다.

```
sudo apt install -y python3-boto3
```

10단계: 프론트엔드 클라이언트 만들기

이전 단계에서 만든 AWS Cloud9 개발 환경을 사용하여 구성한 서비스를 검색하고 이러한 서비스를 호출하는 코드를 사용하는 프런트 엔드 클라이언트를 만듭니다. AWS Cloud Map

1. AWS Management Console 로그인하고 https://console.aws.amazon.com/cloud9/ 에서 AWS Cloud9 콘솔을 엽니다.
2. 왼쪽 탐색 메뉴에서 내 환경을 선택한 다음 환경을 선택하고 Cloud9에서 열기를 선택합니다. `cloudmap-tutorial`
3. AWS Cloud9 환경의 파일 메뉴에서 새 파일을 선택하여 이름이 지정된 파일을 생성합니다. `Untitled1`
4. `Untitled1`파일에서 다음 코드를 복사하여 붙여넣습니다. 이 코드는 서비스의 사용자 지정 `name=writeservice` 속성을 검색하여 데이터를 쓰는 Lambda 함수를 검색합니다. `app-service` DynamoDB 테이블에 데이터를 쓰는 역할을 하는 Lambda 함수의 이름이 반환됩니다. 그런 다음 Lambda 함수가 호출되어 샘플 페이로드가 전달됩니다.

```
import boto3

serviceclient = boto3.client('servicediscovery')

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

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

lambdaclient = boto3.client('lambda')
```

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

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

5. 파일 메뉴에서 다른 이름으로 저장... 을 선택합니다. 파일을 다른 이름으로 저장합니다`writeclient.py`.
6. 사용자 AWS Cloud9 환경의 bash 셸에서 다음 명령을 사용하여 Python 코드를 실행합니다.

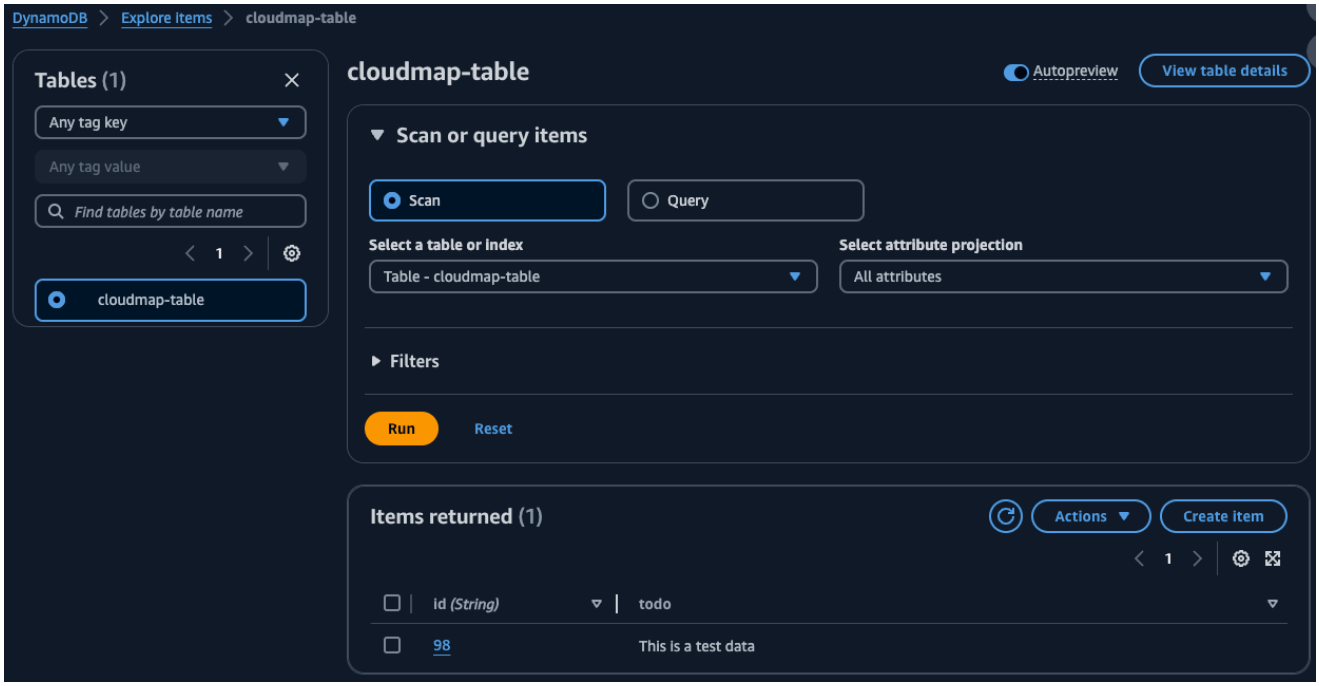
```
python3 writeclient.py
```

출력은 다음과 비슷한 `200` 응답이어야 합니다.

```
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}}"}'
```

7. 이전 단계에서 쓰기가 성공했는지 확인하려면 읽기 클라이언트를 만드십시오.
   a. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/dynamodb/ 에서 DynamoDB 콘솔을 엽니다.
   b. 왼쪽 탐색 창에서 테이블을 선택합니다.
   c. 테이블 목록에서 클라우드맵 테이블을 선택하고 작업 메뉴를 사용하여 항목 탐색을 선택합니다.
   d. 반품된 항목 섹션에서 id (문자열) 열의 숫자 값을 기록해 둡니다.

      다음은 id (문자열) 값이 인 예제입니다. 98



e. AWS Cloud9 환경의 파일 메뉴에서 새 파일을 선택하여 이름이 지정된 파일을 생성합니다. `Untitled1`

f. `Untitled1`파일에서 다음 코드를 복사하여 붙여넣습니다. `Payload`값을 이전 단계의 DynamoDB 테이블의 `id (String)` 값으로 대체합니다. 이 코드는 테이블에서 읽고 이전 단계에서 테이블에 쓴 값을 반환합니다.

```
import boto3

serviceclient = boto3.client('servicediscovery')

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

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

lambdaclient = boto3.client('lambda')

resp = lambdaclient.invoke(FunctionName=functionname,
 InvocationType='RequestResponse', Payload='"98"')

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

g. 파일 메뉴에서 다른 이름으로 저장... 을 선택합니다. 파일을 다른 이름으로 저장합니다`readclient.py`.

h. 사용자 AWS Cloud9 환경의 bash 셸에서 다음 명령을 사용하여 Python 코드를 실행합니다.

```
python3 readclient.py
```

출력은 다음과 유사합니다.

```
b'{"statusCode": 200, "body": "{\\"Item\\": {\\"id\\": \\"98\\", \\"todo\
\": \\"This is a test data\\"}, \\"ResponseMetadata\\": {\\"RequestId\\": \
\"JSO5DLRGF0JUPQN4NCH369ABMBVV4KQNSO5AEMVJF66Q9ASUAAJG\\", \\"HTTPStatusCode\
\": 200, \\"HTTPHeaders\\": {\\"server\\": \\"Server\\", \\"date\\": \\"Wed, 06
 Mar 2024 23:03:38 GMT\\", \\"content-type\\": \\"application/x-amz-json-1.0\
\", \\"content-length\\": \\"61\\", \\"connection\\": \\"keep-alive\\", \\"x-
amzn-requestid\\": \\"JSO5DLRGF0JUPQN4NCH369ABMBVV4KQNSO5AEMVJF66Q9ASUAAJG\\",
 \\"x-amz-crc32\\": \\"3104232745\\"}, \\"RetryAttempts\\": 0}}"}'
```

11단계: 리소스 정리

자습서를 완료한 후 추가 요금이 발생하지 않도록 리소스를 삭제할 수 있습니다. AWS Cloud Map 먼저 서비스 인스턴스를 정리하고, 서비스를, 마지막으로 네임스페이스를 정리하는 등 역순으로 정리해야 합니다. 다음 단계는 이 자습서에 사용된, Lambda AWS Cloud Map, AWS Cloud9 DynamoDB 및 리소스를 정리하는 과정을 안내합니다.

리소스를 삭제하려면 AWS Cloud9

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloud9/ 에서 AWS Cloud9 콘솔을 엽니다.
2. 왼쪽 탐색 메뉴에서 내 환경을 선택합니다.
3. `cloudmap-tutorial`환경을 선택하고 삭제를 선택합니다.
4. 를 입력하여 삭제를 `Delete` 확인한 다음 삭제를 선택합니다.

Lambda 함수를 삭제하려면

1. https://console.aws.amazon.com/lambda/ 에서 AWS Management Console 로그인하고 AWS Lambda 콘솔을 엽니다.
2. 왼쪽 탐색창에서 함수를 선택합니다.
3. `writefunction`및 `readfunction` 함수를 모두 선택합니다.

4. 작업 메뉴에서 삭제를 선택합니다.
5. 를 입력하여 삭제를 `delete` 확인한 다음 삭제를 선택합니다.

DynamoDB 테이블을 삭제하려면

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/dynamodb/ 에서 DynamoDB 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 테이블을 선택합니다.
3. `cloudmap-table`테이블을 선택하고 삭제를 선택합니다.
4. 를 입력하여 삭제를 `confirm` 확인한 다음 삭제를 선택합니다.

AWS Cloud Map 리소스를 삭제하려면

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com/cloudmap/ 에서 AWS Cloud Map 콘솔을 엽니다.
2. 네임스페이스 목록에서 네임스페이스를 선택하고 세부 정보 **`cloudmap-tutorial`** 보기를 선택합니다.
3. 네임스페이스 세부 정보 페이지의 서비스 목록에서 서비스를 선택하고 세부 정보 보기를 선택합니다. **`data-service`**
4. 서비스 인스턴스 섹션에서 인스턴스를 선택하고 등록 **`data-instance`** 취소를 선택합니다.
5. 페이지 상단의 이동 경로를 사용하여 cloudmap-tutorial.com을 선택하여 네임스페이스 세부 정보 페이지로 다시 이동합니다.
6. 네임스페이스 세부 정보 페이지의 서비스 목록에서 데이터 서비스 서비스를 선택하고 삭제를 선택합니다.
7. `app-service`서비스와 및 서비스 인스턴스에 대해 3-6단계를 반복합니다. `write-instance` `read-instance`
8. 왼쪽 탐색창에서 네임스페이스를 선택합니다.
9. `cloudmap-tutorial`네임스페이스를 선택하고 삭제를 선택합니다.

보안 내부 AWS Cloud Map

클라우드 AWS 보안이 최우선 과제입니다. AWS 고객은 가장 보안에 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 혜택을 누릴 수 있습니다.

보안은 기업과 기업 간의 AWS 공동 책임입니다. 공동 책임 모델은 이 사항을 클라우드 내 보안 및 클라우드의 보안으로 설명합니다.

- 클라우드 보안 - AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호하는 역할을 합니다. AWS AWS 또한 안전하게 사용할 수 있는 서비스를 제공합니다. 서드 파티 감사원은 정기적으로 AWS 규정 준수 프로그램의 일환으로 보안 효과를 테스트하고 검증합니다. 적용되는 규정 준수 프로그램에 대해 알아보려면 규정 준수 프로그램별 범위 내AWS 서비스를 참조하십시오. AWS Cloud Map
- 클라우드에서의 보안 - 귀하의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 귀하는 귀사의 데이터의 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 공동 책임 모델을 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 AWS Cloud Map됩니다. 다음 항목에서는 보안 및 규정 준수 목표를 AWS Cloud Map 충족하도록 구성하는 방법을 보여줍니다. 또한 AWS Cloud Map 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법도 알아봅니다.

주제

- AWS Identity and Access Management 에서 AWS Cloud Map
- 로그인 및 모니터링 AWS Cloud Map
- 규정 준수 검증: AWS Cloud Map
- 레질리언스: AWS Cloud Map
- AWS Cloud Map의 인프라 보안
- 를 사용하여 AWS Cloud Map API 호출 로깅 AWS CloudTrail

AWS Identity and Access Management 에서 AWS Cloud Map

도메인 등록 또는 레코드 업데이트와 같은 AWS Cloud Map 리소스에 대한 작업을 수행하려면 AWS Identity and Access Management (IAM) 승인된 사용자임을 인증해야 합니다. AWS AWS Cloud Map 콘솔을 사용하는 경우 AWS 사용자 이름과 암호를 제공하여 ID를 인증합니다. AWS Cloud Map 프로그

래밍 방식으로 액세스하는 경우 애플리케이션은 액세스 키를 사용하거나 요청에 서명하여 ID를 인증합니다.

자격 증명을 인증한 후 IAM은 사용자에게 작업을 수행하고 AWS 리소스에 액세스할 권한이 있는지 확인하여 액세스를 제어합니다. 계정 관리자인 경우 IAM을 사용하여 계정과 관련된 리소스에 대한 다른 사용자의 액세스를 제어할 수 있습니다.

이 장에서는 IAM을 사용하고 리소스를 보호하는 데 도움이 되는 AWS Cloud Map 방법을 설명합니다.

주제

인증

다음 중 AWS 하나로 액세스할 수 있습니다.

- AWS 계정 루트 사용자 – AWS 계정을 처음 생성할 때는 해당 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 Single Sign-In 자격 증명으로 시작합니다. 이 자격 증명은 AWS 계정 루트 사용자라고 하며, 계정을 생성할 때 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 계정을 AWS 계정만들면 먼저 계정의 모든 AWS 서비스 리소스에 완전히 액세스할 수 있는 하나의 로그인 ID로 시작합니다. 이 ID를 AWS 계정 루트 사용자라고 하며, 계정을 만들 때 사용한 이메일 주소와 비밀번호로 로그인하여 액세스할 수 있습니다. 일상적인 작업에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는 작업을수행하는 데 사용합니다. 루트 사용자로 로그인해야 하는 작업의 전체 목록은 IAM 사용자 안내서의 루트 사용자 보안 인증이 필요한 작업을 참조하세요.
- IAM 사용자 - IAM 사용자는 특정 사용자 지정 권한 (예: HTTP 네임스페이스를 생성할 수 있는 권한)이 있는 AWS 계정 내의 자격 증명입니다. AWS Cloud Map IAM 로그인 보안 인증 정보를 사용하여 보안 AWS 웹 페이지(예: AWS Management Console, AWS 토론 포럼, AWS Support 센터)에 로그인할 수 있습니다.

  로그인 보안 인증 정보 외에도 각 사용자마다 액세스 키를 생성할 수도 있습니다. 여러 SDK 중 하나를 통해 또는 를 사용하여 프로그래밍 방식으로 AWS 서비스에 액세스할 때 이러한 키를 사용할 수 있습니다. AWS Command Line Interface SDK 및 CLI 도구는 액세스 키를 사용하여 암호화 방식으로 요청에 서명합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. AWS Cloud Map 인바운드 API 요청을 인증하기 위한 프로토콜인 서명 버전 4를 지원합니다. 요청 인증에 대한 자세한 내용은 Amazon Web Services 일반 참조의 서명 버전 4 서명 프로세스를 참조하세요.

- IAM 역할 – IAM 역할은 계정에 만들 수 있는, 특정 권한을 지닌 IAM 자격 증명입니다. IAM 역할은 자격 증명이 수행할 수 있는 작업과 수행할 수 없는 작업을 결정하는 권한 정책을 가진 AWS 자격 증명이라는 점에서 IAM 사용자와 유사합니다. AWS그러나 역할은 한 사람하고만 연관되지 않고 해당 역할이 필요한 사람이라면 누구든지 맡을 수 있어야 합니다. 또한 역할에는 그와 연관된 암호 또는 액세스 키와 같은 표준 장기 자격 증명이 없습니다. 대신에 역할을 맡은 사람에게는 해당 역할 세션을 위한 임시 보안 자격 증명이 제공됩니다. 임시 보안 인증 정보가 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

  - 연동 사용자 액세스 — IAM 사용자를 생성하는 대신 엔터프라이즈 사용자 디렉토리 또는 웹 ID 공급자의 기존 사용자 ID를 사용할 수 있습니다. AWS Directory Service이러한 사용자를 페더레이션 사용자라고 합니다. AWS ID 공급자를 통해 액세스를 요청할 때 연동 사용자에게 역할을 할당합니다. 페더레이션 사용자에 대한 자세한 정보는 IAM 사용 설명서의 페더레이션 사용자 및 역할을 참조하세요.
  - AWS 서비스 액세스 — 계정의 IAM 역할을 사용하여 AWS 서비스에 계정 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 예를 들어, Amazon Redshift에서 사용자 대신 Amazon S3 버킷에 액세스하도록 허용하는 역할을 생성한 다음 해당 버킷의 데이터를 Amazon Redshift 클러스터로 로드할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 서비스에 권한을 위임하기 위한 역할 생성을 참조하십시오.
  - Amazon EC2에서 실행되는 애플리케이션 — IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행 중이고 API 요청을 하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. AWS 이는 Amazon EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. Amazon EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있게 하려면 인스턴스에 연결된 인스턴스 프로필을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 Amazon EC2 인스턴스에서 실행되는 프로그램이 임시 보안 인증 정보를 얻을 수 있습니다. 자세한 정보는 IAM 사용 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하세요.

액세스 제어

AWS Cloud Map 리소스를 생성, 업데이트, 삭제 또는 나열하려면 작업을 수행할 수 있는 권한과 해당 리소스에 액세스할 수 있는 권한이 필요합니다. 또한 프로그래밍 방식으로 작업을 수행하려면 유효한 액세스 키가 필요합니다.

다음 섹션에서는 에 대한 권한을 관리하는 방법을 설명합니다 AWS Cloud Map. 먼저 개요를 읽어 보면 도움이 됩니다.

- AWS Cloud Map 리소스에 대한 액세스 권한 관리 개요

- ID 기반 정책 (IAM 정책) 사용 대상 AWS Cloud Map
- AWS Cloud Map API 권한: 작업, 리소스, 조건 참조

AWS Cloud Map 리소스에 대한 액세스 권한 관리 개요

모든 AWS 리소스는 AWS 계정이 소유하며 리소스를 만들거나 액세스할 수 있는 권한은 권한 정책에 의해 관리됩니다.

> **Note**
>
> 계정 관리자 또는 관리자 사용자는 관리자 권한이 있는 사용자입니다. 관리자에 대한 자세한 내용은 IAM 사용 설명서의 IAM 모범 사례를 참조하세요.

권한을 부여할 때는 권한을 받는 사용자, 해당 권한의 대상 리소스, 그리고 해당 권한으로 수행할 수 있는 작업을 결정합니다.

주제

- 리소스용 ARN AWS Cloud Map
- 리소스 소유권 이해
- 리소스에 대한 액세스 관리
- 정책 요소 지정: 리소스, 작업, 효과, 보안 주체
- IAM 정책에서 조건 지정

리소스용 ARN AWS Cloud Map

선택한 작업의 네임스페이스 및 서비스에 대해 리소스 수준 권한을 부여하거나 거부할 수 있습니다. 자세한 정보는 AWS Cloud Map API 권한: 작업, 리소스, 조건 참조을 참조하세요.

리소스 소유권 이해

누가 리소스를 만들었든 관계없이 계정은 계정에서 생성된 리소스를 소유합니다. AWS 구체적으로, 리소스 소유자는 리소스 생성 요청을 인증하는 보안 주체 (즉, 루트 사용자 계정, IAM 사용자 또는 IAM 역할) 의 계정입니다. AWS

다음 예에서는 이러한 작동 방식을 설명합니다.

- 계정의 AWS 루트 사용자 계정 자격 증명을 사용하여 HTTP 네임스페이스를 생성하는 경우 해당 AWS 계정이 리소스의 소유자가 됩니다.
- AWS 계정에서 IAM 사용자를 생성하고 해당 사용자에게 HTTP 네임스페이스를 생성할 권한을 부여하면 사용자가 HTTP 네임스페이스를 생성할 수 있습니다. 하지만 해당 사용자가 속한 AWS 계정이 HTTP 네임스페이스 리소스를 소유합니다.
- AWS 계정에 HTTP 네임스페이스를 생성할 권한이 있는 IAM 역할을 생성하는 경우 해당 역할을 수임할 수 있는 사람은 누구나 HTTP 네임스페이스를 생성할 수 있습니다. 이 경우 HTTP 네임스페이스 리소스는 역할이 속한 AWS 계정이 소유합니다.

리소스에 대한 액세스 관리

권한 정책은 누가 무엇에 액세스 할 수 있는지를 지정합니다. 이 단원에서는 AWS Cloud Map에 대한 권한 정책을 생성하기 위한 옵션을 설명합니다. IAM 정책 구문과 설명에 대한 일반적인 내용은 IAM 사용 설명서의 IAM 정책 참조를 참조하세요.

IAM 자격 증명에 연결된 정책을 자격 증명 기반 정책(IAM 정책)이라고 하고, 리소스에 연결된 정책을 리소스 기반 정책이라고 합니다. AWS Cloud Map 의 경우에는 자격 증명 기반 정책(IAM 정책)만 지원합니다.

주제

- ID 기반 정책(IAM 정책)
- 리소스 기반 정책

ID 기반 정책(IAM 정책)

정책을 IAM ID에 연계할 수 있습니다. 예를 들면, 다음을 수행할 수 있습니다:

- 계정 내 사용자 또는 그룹에 권한 정책 연결 계정 관리자는 특정 사용자에 연결된 권한 정책을 사용하여 해당 사용자에게 AWS Cloud Map 리소스 생성 권한을 부여할 수 있습니다.
- 역할에 권한 정책 연결 (계정 간 권한 부여) - 다른 계정에서 생성한 AWS Cloud Map 작업을 수행할 권한을 사용자에게 부여할 수 있습니다. AWS 이렇게 하려면 권한 정책을 IAM 역할에 연결한 다음 다른 계정의 사용자가 역할을 담당할 수 있도록 허용합니다. 다음 예제에서는 계정 A와 계정 B라는 두 개의 AWS 계정에 대해 이 작업을 적용하는 방법을 설명합니다.
  1. 계정 A 관리자는 IAM 역할을 생성하고 계정 A가 소유한 리소스를 생성하거나 액세스할 권한을 부여하는 권한 정책을 역할에 연결합니다.

2. 계정 A 관리자는 신뢰 정책을 역할에 연결합니다. 신뢰 정책은 역할을 담당할 수 있는 보안 주체로 계정 B를 식별합니다.
3. 그런 다음 계정 B 관리자는 역할을 담당할 권한을 계정 B의 사용자 또는 그룹에게 위임할 수 있습니다. 이렇게 하면 계정 B의 사용자가 계정 A에서 리소스를 생성하거나 액세스할 수 있습니다.

다른 AWS 계정의 사용자에게 권한을 위임하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 액세스 관리를 참조하세요.

다음 예제 정책은 사용자가 작업을 수행하여 모든 계정에 대한 퍼블릭 DNS 네임스페이스를 만들 수 있도록 허용합니다. CreatePublicDnsNamespace AWS Amazon Route 53 권한이 필요합니다. 퍼블릭 DNS 네임스페이스를 생성할 때 Route 53 호스팅 AWS Cloud Map 영역도 생성되기 때문입니다.

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

정책을 프라이빗 DNS 네임스페이스에 대신 적용하려면 작업을 사용할 권한을 부여해야 합니다. AWS Cloud Map CreatePrivateDnsNamespace 또한 Route 53 프라이빗 호스팅 영역이 AWS Cloud Map 생성되므로 이전 예와 동일한 Route 53 작업을 사용할 권한을 부여합니다. 또한 `DescribeVpcs` 및 `DescribeRegions`의 Amazon EC2 작업 두 개를 사용할 권한도 부여합니다.

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

```
            "route53:CreateHostedZone",
            "route53:GetHostedZone",
            "route53:ListHostedZonesByName"
         ],
         "Resource":"*"
      },
      {
         "Effect": "Allow",
         "Action": [
            "ec2:DescribeVpcs",
            "ec2:DescribeRegions"
         ],
         "Resource":"*"
      }
   ]
}
```

의 정책을 ID에 연결하는 방법에 대한 자세한 내용은 을 참조하십시오. AWS Cloud MapID 기반 정책 (IAM 정책) 사용 대상 AWS Cloud Map 사용자, 그룹, 역할 및 권한에 대한 자세한 내용은 IAM 사용 설명서의 자격 증명(사용자, 그룹 및 역할)을 참조하세요.

리소스 기반 정책

Amazon S3 등 다른 서비스에서도 권한 정책을 리소스에 연결할 수 있습니다. 예를 들어 정책을 S3 버킷에 연결하여 해당 버킷에 대한 액세스 권한을 관리할 수 있습니다. AWS Cloud Map 리소스에 정책을 연결하는 것을 지원하지 않습니다.

정책 요소 지정: 리소스, 작업, 효과, 보안 주체

AWS Cloud Map 각 AWS Cloud Map 리소스에서 사용할 수 있는 AWS Cloud Map API 작업 (API 참조) 을 포함합니다 (참조리소스용 ARN AWS Cloud Map). 사용자 또는 연동 사용자에게 이러한 작업 중 하나 또는 전부를 수행할 권한을 부여할 수 있습니다. 퍼블릭 DNS 네임스페이스 생성과 같은 일부 API 작업을 수행하려면 둘 이상의 작업을 수행할 권한이 필요합니다.

다음은 기본 정책 요소입니다.

- 리소스 – Amazon 리소스 이름(ARN)을 사용하여 정책을 적용할 리소스를 식별합니다. 자세한 정보는 리소스용 ARN AWS Cloud Map을 참조하세요.
- 작업 – 작업 키워드를 사용하여 허용 또는 거부할 리소스 작업을 식별합니다. 예를 들어`Effect`, 지정된 내용에 따라 `servicediscovery:CreateHttpNamespace` 권한은 사용자에게 AWS Cloud Map CreateHttpNamespace작업 수행 권한을 허용하거나 거부합니다.

- 효과 – 사용자가 지정된 리소스에서 작업을 수행하려고 할 때 효과를 허용 또는 거부로 지정합니다. 작업에 대한 액세스 권한을 명시적으로 부여하지 않으면 액세스는 묵시적으로 거부됩니다. 한 리소스에 대한 액세스를 명시적으로 거부할 수도 있으며, 다른 정책에 따라 액세스 권한이 부여되더라도 사용자가 이 리소스에 액세스하지 못하도록 조치를 취할 수 있습니다.
- 보안 주체 – 정체 기반 정책(IAM 정책)에서 정책이 연계된 사용자는 암묵적인 보안 주체입니다. 리소스 기반 정책의 경우 사용자, 계정, 서비스 또는 권한의 수신자인 기타 개체를 지정합니다(리소스 기반 정책에만 해당). AWS Cloud Map 의 경우 리소스 기반 정책을 지원하지 않습니다.

IAM 정책 구문과 설명에 대한 자세한 내용은 IAM 사용 설명서의 IAM 정책 참조를 참조하세요.

AWS Cloud Map API 작업 및 해당 작업이 적용되는 리소스 목록은 을 참조하십시오AWS Cloud Map API 권한: 작업, 리소스, 조건 참조.

IAM 정책에서 조건 지정

권한을 부여할 때 IAM 정책 언어를 사용하여 정책이 언제 적용되는지를 지정할 수 있습니다. 예를 들어, 지정한 날짜 이후에만 정책을 적용하거나 지정한 네임스페이스에만 정책을 적용하도록 할 수 있습니다.

조건을 표현하려면 사전 정의된 조건 키를 사용합니다. AWS Cloud Map 자체 조건 키 세트를 정의하며 일부 글로벌 조건 키 사용도 지원합니다. 자세한 정보는 다음 주제를 참조하십시오:

- AWS Cloud Map 조건 키에 대한 자세한 내용은 을 참조하십시오AWS Cloud Map API 권한: 작업, 리소스, 조건 참조.
- AWS 글로벌 조건 키에 대한 자세한 내용은 IAM 사용 설명서의AWS 글로벌 조건 컨텍스트 키를 참조하십시오.
- 정책 언어의 조건 지정에 대한 자세한 내용은 IAM 사용 안내서의 IAM JSON 정책 요소: 조건을 참조하세요.

ID 기반 정책 (IAM 정책) 사용 대상 AWS Cloud Map

이 주제에서는 계정 관리자가 IAM 자격 증명 (사용자, 그룹, 역할) 에 권한 정책을 연결하여 리소스에서 작업을 수행할 권한을 부여하는 방법을 보여주는 ID 기반 정책의 예를 제공합니다. AWS Cloud Map

> **Important**
>
> 먼저 리소스에 대한 액세스를 관리하기 위한 기본 개념과 옵션을 설명하는 소개 주제를 검토하는 것이 좋습니다. AWS Cloud Map 자세한 정보는 AWS Cloud Map 리소스에 대한 액세스 권한 관리 개요을 참조하세요.

주제

- AWS Cloud Map 콘솔 사용에 필요한 권한

다음 예제에서는 사용자에게 서비스 인스턴스를 등록 및 등록 취소할 수 있는 권한을 부여하는 권한 정책을 보여줍니다. Sid(문 ID)는 선택 사항입니다.

```
{
   "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": "*"
      }
   ]
}
```

이 정책은 서비스 인스턴스를 등록 및 관리하는 데 필요한 작업에 대한 권한을 부여합니다. 인스턴스를 등록하고 등록 취소할 때 Route 53 레코드와 상태 점검을 AWS Cloud Map 생성, 업데이트 및 삭제하므로 퍼블릭 또는 프라이빗 DNS 네임스페이스를 사용하는 경우 Route 53 권한이 필요합니다. 와일드카드 문자 (*) 는 현재 계정이 소유한 모든 AWS Cloud Map 인스턴스와 Route 53 레코드 및 상태 확인에 대한 액세스 권한을 `Resource` 부여합니다. AWS

작업과 각 작업을 사용할 권한을 부여하거나 거부하기 위해 지정하는 ARN의 목록은 AWS Cloud Map API 권한: 작업, 리소스, 조건 참조 단원을 참조하세요.

AWS Cloud Map 콘솔 사용에 필요한 권한

AWS Cloud Map 콘솔에 대한 전체 액세스 권한을 부여하려면 다음 권한 정책에서 권한을 부여해야 합니다.

```
{
   "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":"*"
      }
   ]
}
```

다음은 권한이 필요한 이유입니다.

servicediscovery:*

모든 AWS Cloud Map 작업을 수행할 수 있습니다.

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

퍼블릭 및 프라이빗 DNS 네임스페이스를 생성하고 삭제할 때 호스팅 영역을 AWS Cloud Map 관리할 수 있습니다.

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

서비스를 생성할 때 Amazon Route 53 상태 확인을 포함하면 상태 확인을 AWS Cloud Map 관리할 수 있습니다.

ec2:DescribeVpcs 및 **ec2:DescribeRegions**

프라이빗 호스팅 영역을 AWS Cloud Map 관리해 보세요.

AWS Cloud Map의 AWS 관리형 정책

AWS 관리형 정책은 AWS에 의해 생성되고 관리되는 독립 실행형 정책입니다. AWS 관리형 정책은 사용자, 그룹 및 역할에 권한 할당을 시작할 수 있도록 많은 일반 사용 사례에 대한 권한을 제공하도록 설계되었습니다.

AWS 관리형 정책은 모든 AWS 고객이 사용할 수 있기 때문에 특정 사용 사례에 대해 최소 권한을 부여하지 않을 수 있습니다. 사용 사례에 고유한 고객 관리형 정책을 정의하여 권한을 줄이는 것이 좋습니다.

AWS 관리형 정책에서 정의한 권한은 변경할 수 없습니다. AWS에서 AWS 관리형 정책에 정의된 권한을 업데이트할 경우 정책이 연결되어 있는 모든 보안 주체 엔터티(사용자, 그룹 및 역할)에도 업데이트가 적용됩니다. 새로운 AWS 서비스를 시작하거나 새로운 API 작업을 기존 서비스에 이용하는 경우 AWS가 AWS 관리형 정책을 업데이트할 가능성이 높습니다.

자세한 내용은 IAM 사용 설명서의 AWS 관리형 정책을 참조하세요.

AWS 관리형 정책: AWSCloudMapDiscoverInstanceAccess

`AWSCloudMapDiscoverInstanceAccess`를 IAM 엔터티에 연결할 수 있습니다. AWS Cloud Map Discovery API에 대한 액세스 권한을 제공합니다.

이 정책의 권한을 보려면 AWS 관리형 정책 참조의 AWSCloudMapDiscoverInstanceAccess를 확인하세요.

AWS 관리형 정책: AWSCloudMapReadOnlyAccess

`AWSCloudMapReadOnlyAccess`를 IAM 엔터티에 연결할 수 있습니다. 모든 AWS Cloud Map 작업에 대한 읽기 전용 액세스 권한을 부여합니다.

이 정책의 권한을 보려면 AWS 관리형 정책 참조의 AWSCloudMapReadOnlyAccess를 확인하세요.

AWS관리형 정책: AWSCloudMapRegisterInstanceAccess

`AWSCloudMapRegisterInstanceAccess`를 IAM 엔터티에 연결할 수 있습니다. 네임스페이스 및 서비스에 대한 읽기 전용 액세스 권한을 부여하고, 서비스 인스턴스를 등록 및 등록 취소하는 권한을 부여합니다.

이 정책의 권한을 보려면 AWS 관리형 정책 참조의 AWSCloudMapRegisterInstanceAccess를 확인하세요.

AWS 관리형 정책: AWSCloudMapFullAccess

`AWSCloudMapFullAccess`를 IAM 엔터티에 연결할 수 있습니다. 모든 AWS Cloud Map 작업에 대한 모든 액세스 권한 제공

이 정책의 권한을 보려면 AWS 관리형 정책 참조의 AWSCloudMapFullAccess를 확인하세요.

AWS 관리형 정책으로 AWS Cloud Map 업데이트

이 서비스가 이러한 변경 내용을 추적하기 시작한 이후부터 AWS Cloud Map의 AWS 관리형 정책 업데이트에 대한 세부 정보를 봅니다. 이 페이지의 변경 사항에 대한 자동 알림을 받으려면 AWS Cloud Map 문서 기록 페이지에서 RSS 피드를 구독하세요.

변경 사항	설명	날짜
AWSCloudMapDiscoverInstanceAccess, AWSCloudMapRegisterInstanceAccess, AWSCloudMapReadOnlyAccess – 기존 정책 업데이트.	AWS Cloud Map에서 새로운 AWS Cloud Map `DiscoverInstanceRevision` API 작업에 대한 액세스 권한을 제공하도록 이러한 정책을 업데이트했습니다.	2023년 8월 15일

고객 관리형 정책 예

AWS Cloud Map 작업에 대한 권한을 허용하는 고유의 사용자 지정 IAM 정책을 생성할 수 있습니다. 지정된 권한이 필요한 IAM 사용자 또는 그룹에 이러한 사용자 지정 정책을 연결할 수 있습니다. 이러한 정책은 AWS Cloud Map API, AWS SDK 또는 AWS CLI를 사용하는 경우에 적용됩니다. 다음 예제에서는 몇 가지 일반적인 사용 사례의 권한을 보여 줍니다. 사용자에게 AWS Cloud Map에 대한 전체 액세스 권한을 부여하는 정책에 대해서는 AWS Cloud Map 콘솔 사용에 필요한 권한 단원을 참조하세요.

예시

- 예제 1: 모든 AWS Cloud Map 리소스에 대한 읽기 액세스 허용
- 예제 2: 모든 유형의 네임스페이스 생성 허용

예제 1: 모든 AWS Cloud Map 리소스에 대한 읽기 액세스 허용

다음 권한 정책은 사용자에게 모든 AWS Cloud Map 리소스에 대한 읽기 전용 액세스 권한을 부여합니다.

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

예제 2: 모든 유형의 네임스페이스 생성 허용

다음 권한 정책은 사용자가 모든 유형의 네임스페이스를 생성할 수 있도록 허용합니다.

```
{
   "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":"*"
         }
      ]
}
```

액세스 권한을 제공하려면 사용자, 그룹 또는 역할에 권한을 추가합니다.

- AWS IAM Identity Center의 사용자 및 그룹:

  권한 세트를 생성합니다. AWS IAM Identity Center 사용 설명서의 권한 세트 생성의 지침을 따르세요.

- 자격 증명 공급자를 통해 IAM에서 관리되는 사용자:

  아이덴티티 페더레이션을 위한 역할을 생성합니다. IAM 사용 설명서의 서드 파티 자격 증명 공급자의 역할 만들기(연합)의 지침을 따르세요.

- IAM 사용자:
  - 사용자가 맡을 수 있는 역할을 생성합니다. IAM 사용 설명서에서 IAM 사용자의 역할 생성의 지침을 따르세요.
  - (권장되지 않음) 정책을 사용자에게 직접 연결하거나 사용자를 사용자 그룹에 추가합니다. IAM 사용 설명서에서 사용자(콘솔)에 권한 추가의 지침을 따르세요.

AWS Cloud Map API 권한: 작업, 리소스, 조건 참조

IAM 자격 증명에 연결할 수 있는 액세스 제어 및 쓰기 권한 정책(자격 증명 기반 정책)을 설정할 때 아래 목록을 참조로 사용할 수 있습니다. 목록에는 각 AWS Cloud Map API 작업, 액세스 권한을 부여해야 하는 작업, 액세스 권한을 부여해야 하는 AWS 리소스가 포함됩니다. 정책의 `Action` 필드에서 작업을 지정하고, 정책의 `Resource` 필드에서 리소스 값을 지정합니다.

일부 작업의 경우 IAM 정책에서 AWS Cloud Map—특정 조건 키를 사용할 수 있습니다. 자세한 정보는 AWS Cloud Map 조건 키 참조을 참조하세요. AWS 폭넓은 조건 키를 사용할 수도 있습니다. AWS 와이드 키의 전체 목록은 IAM 사용 설명서의 사용 가능한 키를 참조하십시오.

작업을 지정하려면 `servicediscovery` 접두사 다음에 API 작업 이름을 사용합니다(예: `servicediscovery:CreatePublicDnsNamespace` 및 `route53:CreateHostedZone`).

주제

- 작업에 AWS Cloud Map 필요한 권한
- AWS Cloud Map 조건 키 참조

작업에 AWS Cloud Map 필요한 권한

CreateHttpNamespace

필요한 권한(API 작업):

- `servicediscovery:CreateHttpNamespace`

리소스: *

CreatePrivateDnsNamespace

필요한 권한(API 작업):

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

리소스: *

CreatePublicDnsNamespace

필요한 권한(API 작업):

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

- route53:ListHostedZonesByName

리소스: *

CreateService

필요한 권한(API 작업): servicediscovery:CreateService

리소스: *

DeleteNamespace

필요한 권한(API 작업):

- servicediscovery:DeleteNamespace

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:namespace/*namespace-id*

DeleteService

필요한 권한(API 작업): servicediscovery:DeleteService

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:service/*service-id*

DeregisterInstance

필요한 권한(API 작업):

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

리소스: *

DiscoverInstances

필요한 권한(API 작업): servicediscovery:DiscoverInstances

리소스: *

GetInstance

필요한 권한(API 작업): servicediscovery:GetInstance

리소스: *

GetInstancesHealthStatus

필요한 권한(API 작업): servicediscovery:GetInstancesHealthStatus

리소스: *

GetNamespace

필요한 권한(API 작업): servicediscovery:GetNamespace

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:namespace/*namespace-id*

GetOperation

필요한 권한(API 작업): servicediscovery:GetOperation

리소스: *

GetService

필요한 권한(API 작업): servicediscovery:GetService

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:service/*service-id*

ListInstances

필요한 권한(API 작업): servicediscovery:ListInstances

리소스: *

ListNamespaces

필요한 권한(API 작업): servicediscovery:ListNamespaces

리소스: *

ListOperations

필요한 권한(API 작업): servicediscovery:ListOperations

리소스: *

ListServices

필요한 권한(API 작업): servicediscovery:ListServices

리소스: *

ListTagsForResource

필요한 권한(API 작업): `servicediscovery:ListTagsForResource`

리소스: *

RegisterInstance

필요한 권한(API 작업):

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

리소스: *

TagResource

필요한 권한(API 작업): `servicediscovery:TagResource`

리소스: *

UntagResource

필요한 권한(API 작업): `servicediscovery:UntagResource`

리소스: *

UpdateHttpNamespace

필요한 권한(API 작업): `servicediscovery:UpdateHttpNamespace`

리소스: *, `arn:aws:servicediscovery:`*`region`*`:`*`account-id`*`:namespace/`*`namespace-id`*

UpdateInstanceCustomHealthStatus

필요한 권한(API 작업): `servicediscovery:UpdateInstanceCustomHealthStatus`

리소스: *

UpdatePrivateDnsNamespace

필요한 권한(API 작업):

- servicediscovery:UpdatePrivateDnsNamespace
- route53:ChangeResourceRecordSets

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:namespace/*namespace-id*

UpdatePublicDnsNamespace

필요한 권한(API 작업):

- servicediscovery:UpdatePublicDnsNamespace
- route53:ChangeResourceRecordSets

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:namespace/*namespace-id*

UpdateService

필요한 권한(API 작업):

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

리소스: *, arn:aws:servicediscovery:*region*:*account-id*:service/*service-id*

AWS Cloud Map 조건 키 참조

AWS Cloud Map IAM 정책의 Condition 요소에서 특정 AWS Cloud Map 작업에 사용할 수 있는 다음과 같은 조건 키를 정의합니다. 이러한 키를 사용하여 정책 설명이 적용되는 조건을 보다 상세하게 설정할 수 있습니다. 이러한 조건 키를 허용하는 AWS Cloud Map 작업에 대한 자세한 내용은 정의된 작업을 참조하십시오. AWS Cloud Map일반적인 조건 키에 대한 자세한 내용은 을 참조하십시오IAM 정책에서 조건 지정.

servicediscovery:NamespaceArn

관련 네임스페이스에 대해 Amazon 리소스 이름(ARN)을 지정하여 객체를 가져올 수 있는 필터입니다.

servicediscovery:NamespaceName

관련 네임스페이스의 이름을 지정하여 객체를 가져올 수 있는 필터입니다.

servicediscovery:ServiceArn

관련 서비스에 대해 Amazon 리소스 이름(ARN)을 지정하여 객체를 가져올 수 있는 필터입니다.

servicediscovery:ServiceName

관련 서비스의 이름을 지정하여 객체를 가져올 수 있는 필터입니다.

로그인 및 모니터링 AWS Cloud Map

모니터링은 AWS 솔루션의 신뢰성, 가용성 및 성능을 유지하는 데 있어 중요한 부분입니다. 다중 지점 장애가 발생할 경우 이를 보다 쉽게 디버깅할 수 있도록 AWS 솔루션의 모든 부분에서 모니터링 데이터를 수집해야 합니다. 하지만 모니터링을 시작하기 전에 다음 질문에 대한 답변을 포함하는 모니터링 계획을 작성해야 합니다.

- 모니터링의 목표
- 모니터링할 리소스
- 이러한 리소스를 모니터링하는 빈도
- 사용할 모니터링 도구
- 모니터링 작업을 수행할 사람
- 문제 발생 시 알려야 할 대상

규정 준수 검증: AWS Cloud Map

의 AWS Cloud Map 보안 및 규정 준수는 건강 보험 이전 및 책임에 관한 법률 (HIPAA), 결제 카드 산업 데이터 보안 표준 (PCI DSS), ISO 및 FIPS를 비롯한 여러 AWS 규정 준수 프로그램의 일환으로 타사 감사자가 평가합니다.

특정 규정 준수 프로그램 범위 내 AWS 서비스 목록은 규정 준수 프로그램별 범위 내 서비스를 참조하십시오.AWS 일반 정보는 AWS 규정 준수 프로그램을 참조하세요.

를 사용하여 타사 감사 보고서를 다운로드할 수 AWS Artifact있습니다. 자세한 내용은 AWS Artifact의 보고서 다운로드를 참조하십시오.

AWS 서비스 이용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 결정됩니다. AWS 규정 준수에 도움이 되는 리소스를 제공합니다.

- 보안 및 규정 준수 퀵 스타트 가이드 - 이 배포 가이드에서는 아키텍처 고려 사항을 설명하고 보안 및 규정 준수에 중점을 둔 기본 환경을 배포하기 위한 단계를 제공합니다. AWS
- HIPAA 보안 및 규정 준수를 위한 설계 백서 — 이 백서에서는 기업이 HIPAA 준수 애플리케이션을 만드는 AWS 데 사용할 수 있는 방법을 설명합니다.
- AWS 규정 준수 리소스 — 이 워크북 및 가이드 모음은 해당 산업 및 지역에 적용될 수 있습니다.
- AWS Config— 이 AWS 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub— 이 AWS 서비스는 보안 업계 표준 및 모범 사례를 준수하는지 확인하는 데 도움이 AWS 되는 보안 상태를 종합적으로 보여줍니다.

레질리언스: AWS Cloud Map

AWS 글로벌 인프라는 AWS 지역 및 가용 영역을 중심으로 구축됩니다. AWS 지역은 물리적으로 분리되고 격리된 여러 가용 영역을 제공하며, 이러한 가용 영역은 지연 시간이 짧고 처리량이 높으며 중복성이 높은 네트워킹으로 연결됩니다. 가용 영역을 사용하면 중단 없이 가용 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 복수 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS Cloud Map 주로 글로벌 서비스입니다. 하지만 Amazon EC2 인스턴스 및 Elastic Load Balancing 로드 밸런서와 같은 특정 지역의 리소스 상태를 확인하는 Route 53 상태 확인을 생성하는 데 사용할 AWS Cloud Map 수 있습니다.

AWS 지역 및 가용 영역에 대한 자세한 내용은 AWS 글로벌 인프라를 참조하십시오.

AWS Cloud Map의 인프라 보안

관리형 서비스인 AWS Cloud Map는 AWS 글로벌 네트워크 보안으로 보호됩니다. AWS 보안 서비스와 AWS의 인프라 보호 방법에 대한 자세한 내용은 AWS 클라우드 보안을 참조하세요. 인프라 보안에 대한 모범 사례를 사용하여 AWS 환경을 설계하려면 보안 원칙 AWS Well-Architected Framework의 인프라 보호를 참조하세요.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 AWS Cloud Map에 액세스합니다. 고객은 다음을 지원해야 합니다.

- 전송 계층 보안(TLS). TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 보안 액세스 키를 사용하여 서명해야 합니다. 또는 AWS Security Token Service(AWS STS)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

인터페이스 VPC 엔드포인트를 사용하도록 AWS Cloud Map을 구성하여 VPC의 보안 태세를 향상시킬 수 있습니다. 자세한 내용은 인터페이스 엔드포인트(AWS PrivateLink)를 사용하여 AWS Cloud Map에 액세스합니다. 섹션을 참조하세요.

인터페이스 엔드포인트(AWS PrivateLink)를 사용하여 AWS Cloud Map에 액세스합니다.

AWS PrivateLink를 사용하여 VPC와 AWS Cloud Map 사이에 프라이빗 연결을 생성할 수 있습니다. 인터넷 게이트웨이, NAT 디바이스, VPN 연결 또는 AWS Direct Connect 연결을 사용하지 않고 VPC에 있는 것처럼 AWS Cloud Map에 액세스할 수 있습니다. VPC의 인스턴스에서 AWS Cloud Map에 액세스하는 데 퍼블릭 IP 주소가 필요하지 않습니다.

AWS PrivateLink에서 제공되는 인터페이스 엔드포인트를 생성하여 이 프라이빗 연결을 설정합니다. 인터페이스 엔드포인트에 대해 사용 설정하는 각 서브넷에서 엔드포인트 네트워크 인터페이스를 생성합니다. 이는 AWS Cloud Map로 향하는 트래픽의 진입점 역할을 하는 요청자 관리형 네트워크 인터페이스입니다.

자세한 내용은 AWS PrivateLink 가이드의 AWS PrivateLink를 통해 AWS 서비스에 액세스를 참조하세요.

AWS Cloud Map에 대한 고려 사항

AWS Cloud Map에 대한 인터페이스 엔드포인트를 설정하려면 먼저 AWS PrivateLink 가이드의 고려 사항를 검토합니다.

Amazon VPC에 인터넷 게이트웨이가 없고 작업에서 `awslogs` 로그 드라이버를 사용하여 로그 정보를 CloudWatch Logs로 전송하는 경우에는 CloudWatch Logs용 인터페이스 VPC 엔드포인트를 생성해야 합니다. 자세한 정보는 Amazon CloudWatch Logs 사용 설명서의 인터페이스 VPC 엔드포인트에서 CloudWatch Logs 사용을 참조하세요.

VPC 엔드포인트는 AWS 교차 리전 요청을 지원하지 않습니다. API 호출을 AWS Cloud Map로 발행할 계획인 동일 리전에서 엔드포인트를 생성해야 합니다.

VPC 엔드포인트는 Amazon Route 53을 통해 Amazon이 제공하는 DNS만 지원합니다. 자신의 DNS를 사용하는 경우에는 조건적인 DNS 전송을 사용할 수 있습니다. 자세한 정보는 Amazon VPC 사용 설명서의 DHCP 옵션 세트를 참조하세요.

VPC 엔드포인트에 연결된 보안 그룹은 Amazon VPC의 프라이빗 서브넷에서 443 포트로 들어오는 연결을 허용해야 합니다.

AWS Cloud Map용 인터페이스 엔드포인트 생성

Amazon VPC 콘솔 또는 AWS Command Line Interface(AWS CLI)를 사용하여 AWS Cloud Map에 대한 인터페이스 엔드포인트를 생성할 수 있습니다. 자세한 내용은 AWS PrivateLink 가이드의 인터페이스 엔드포인트 생성을 참조하세요.

다음 서비스 이름을 사용하여 AWS Cloud Map에 대한 엔드포인트를 생성합니다.

> **Note**
>
> 이 두 엔드포인트에서는 `DiscoverInstances` API를 사용할 수 없습니다.

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

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

다음과 같은 서비스 이름을 사용하여 `DiscoverInstances` API에 액세스할 AWS Cloud Map 데이터 영역의 인터페이스 엔드포인트를 생성합니다.

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

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

> **Note**
>
> 데이터 영역 엔드포인트의 리전 또는 영역 VPCE DNS 이름을 사용하여 `DiscoverInstances`를 호출할 때는 호스트 접두사 삽입을 비활성화해야 합니다. AWS CLI

> 및 AWS SDK는 각 API 작업을 호출할 때 서비스 엔드포인트 앞에 다양한 호스트 접두사를 추가하며, 이로 인해 VPC 엔드포인트를 지정할 때 잘못된 URL이 생성됩니다.

인터페이스 엔드포인트에 프라이빗 DNS를 사용하도록 설정하는 경우, 리전에 대한 기본 DNS 이름을 사용하여 AWS Cloud Map에 API 요청을 할 수 있습니다. 예: `servicediscovery.us-east-1.amazonaws.com`.

VPCE AWS PrivateLink 연결은 AWS Cloud Map에서 지원되는 모든 리전에서 지원되지만, 고객은 엔드포인트를 정의하기 전에 VPCE를 지원하는 가용 영역을 확인해야 합니다. 리전의 인터페이스 VPC 엔드포인트에서 지원되는 가용 영역을 확인하려면 describe-vpc-endpoint-services 명령을 사용하거나 AWS Management Console을 사용하세요. 예를 들어 다음 명령은 미국 동부(오하이오) 리전 내에 AWS Cloud Map 인터페이스 VPC 엔드포인트를 배포할 수 있는 가용 영역을 반환합니다.

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

를 사용하여 AWS Cloud Map API 호출 로깅 AWS CloudTrail

AWS Cloud Map 사용자 AWS CloudTrail, 역할 또는 담당자가 수행한 작업의 기록을 제공하는 서비스와 통합되어 AWS 서비스있습니다. CloudTrail 모든 API 호출을 AWS Cloud Map 이벤트로 캡처합니다. 캡처된 호출에는 AWS Cloud Map 콘솔에서의 호출 및 AWS Cloud Map API 작업에 대한 코드 호출이 포함됩니다. 에서 수집한 CloudTrail 정보를 사용하여 요청을 받은 사람 AWS Cloud Map, 요청한 IP 주소, 요청 시기 및 추가 세부 정보를 확인할 수 있습니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 보안 인증 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트 사용자로 했는지 사용자 보안 인증으로 했는지 여부.
- IAM Identity Center 사용자를 대신하여 요청이 이루어졌는지 여부입니다.
- 역할 또는 페더레이션 사용자에 대한 임시 보안 인증 정보를 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

CloudTrail 계정을 만들 AWS 계정 때 활성화되며 자동으로 CloudTrail 이벤트 기록에 액세스할 수 있습니다. CloudTrail 이벤트 기록은 지난 90일간의 기록된 관리 이벤트를 보고, 검색하고, 다운로드할 수 있고, 변경할 수 없는 기록을 제공합니다. AWS 리전자세한 내용은 사용 설명서의 CloudTrail 이벤트

기록 사용을 참조하십시오.AWS CloudTrail 이벤트 기록 조회에는 CloudTrail 요금이 부과되지 않습니다.

AWS 계정 지난 90일 동안 진행 중인 이벤트 기록을 보려면 트레일 또는 CloudTrail호수 이벤트 데이터 저장소를 생성하세요.

CloudTrail 트레일

트레일을 사용하면 CloudTrail Amazon S3 버킷에 로그 파일을 전송할 수 있습니다. 를 사용하여 생성된 모든 트레일은 멀티 AWS Management Console 리전입니다. AWS CLI를 사용하여 단일 리전 또는 다중 리전 추적을 생성할 수 있습니다. 계정의 모든 활동을 기록할 수 있으므로 멀티 리전 트레일을 생성하는 것이 좋습니다 AWS 리전 . 단일 리전 추적을 생성하는 경우 추적의 AWS 리전에 로깅된 이벤트만 볼 수 있습니다. 추적에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 Creating a trail for your AWS 계정 및 Creating a trail for an organization을 참조하세요.

트레일을 CloudTrail 생성하여 진행 중인 관리 이벤트의 사본 하나를 Amazon S3 버킷으로 무료로 전송할 수 있지만 Amazon S3 스토리지 요금이 부과됩니다. CloudTrail 요금에 대한 자세한 내용은 AWS CloudTrail 요금을 참조하십시오. Amazon S3 요금에 대한 자세한 내용은 Amazon S3 요금을 참조하세요.

CloudTrail Lake 이벤트 데이터 스토어

CloudTrail Lake를 사용하면 이벤트에 대한 SQL 기반 쿼리를 실행할 수 있습니다. CloudTrail Lake는 행 기반 JSON 형식의 기존 이벤트를 Apache ORC 형식으로 변환합니다. ORC는 빠른 데이터 검색에 최적화된 열 기반 스토리지 형식입니다. 이벤트는 이벤트 데이터 스토어로 집계되며, 이벤트 데이터 스토어는 고급 이벤트 선택기를 적용하여 선택한 기준을 기반으로 하는 변경 불가능한 이벤트 컬렉션입니다. 이벤트 데이터 스토어에 적용하는 선택기는 어떤 이벤트가 지속되고 쿼리할 수 있는지 제어합니다. CloudTrail Lake에 대한 자세한 내용은 사용 설명서의 Lake 사용을 참조하십시오. AWS CloudTrail AWS CloudTrail

CloudTrail Lake 이벤트 데이터 저장 및 쿼리로 인해 비용이 발생합니다. 이벤트 데이터 스토어를 생성할 때 이벤트 데이터 스토어에 사용할 요금 옵션을 선택합니다. 요금 옵션에 따라 이벤트 모으기 및 저장 비용과 이벤트 데이터 스토어의 기본 및 최대 보존 기간이 결정됩니다. CloudTrail 요금에 대한 자세한 내용은 AWS CloudTrail 요금을 참조하십시오.

AWS Cloud Map 의 데이터 이벤트 CloudTrail

데이터 이벤트는 리소스에서 또는 리소스에서 수행되는 리소스 작업에 대한 정보를 제공합니다 (예: 네임스페이스에서 등록된 인스턴스 검색). 이를 데이터 영역 작업이라고도 합니다. 데이터 이벤트가 대량

활동인 경우도 있습니다. 기본적으로 는 데이터 이벤트를 기록하지 CloudTrail 않습니다. CloudTrail 이벤트 기록에는 데이터 이벤트가 기록되지 않습니다.

데이터 이벤트에는 추가 요금이 적용됩니다. CloudTrail 요금에 대한 자세한 내용은 AWS CloudTrail 요금을 참조하십시오.

CloudTrail 콘솔 또는 CloudTrail API 작업을 사용하여 AWS Cloud Map 리소스 유형에 대한 데이터 이벤트를 기록할 수 있습니다. AWS CLI데이터 이벤트를 로깅하는 방법에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 Logging data events with the AWS Management Console 및 Logging data events with the AWS Command Line Interface를 참조하세요.

다음 표에는 데이터 이벤트를 기록할 수 있는 AWS Cloud Map 리소스 유형이 나열되어 있습니다. 데이터 이벤트 유형 (콘솔) 열에는 CloudTrail 콘솔의 데이터 이벤트 유형 목록에서 선택할 수 있는 값이 표시됩니다. resources.type 값 열에는 또는 `resources.type` API를 사용하여 고급 이벤트 선택기를 구성할 때 지정하는 값이 표시됩니다. AWS CLI CloudTrail 데이터 API 로깅 대상 CloudTrail 열에는 해당 리소스 유형에 대해 로깅된 API 호출이 표시됩니다. CloudTrail

데이터 이벤트 유형(콘솔)	resources.type 값	로깅된 데이터 API CloudTrail
AwsApiCall	`AWS::ServiceDiscovery::Namespace`	• DiscoverInstances • DiscoverInstancesRevision
AwsApiCall	`AWS::ServiceDiscovery::Service`	• DiscoverInstances • DiscoverInstancesRevision

`eventName`, `readOnly` 및 `resources.ARN` 필드를 필터링하여 중요한 이벤트만 로깅하도록 고급 이벤트 선택기를 구성할 수 있습니다. 이러한 필드에 대한 자세한 내용은 AWS CloudTrail API 참조의 AdvancedFieldSelector 섹션을 참조하세요.

다음 예제는 모든 AWS Cloud Map 데이터 이벤트를 기록하도록 고급 이벤트 선택기를 구성하는 방법을 보여줍니다.

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

```
            { "Field": "resources.type", "Equals":
 ["AWS::ServiceDiscovery::Namespace"] }
        ]
    }
]
```

AWS Cloud Map 의 관리 이벤트 CloudTrail

관리 이벤트는 내 리소스에 대해 수행되는 관리 작업에 대한 정보를 제공합니다 AWS 계정. 이를 제어 영역 작업이라고도 합니다. 기본적으로 관리 이벤트를 CloudTrail 기록합니다.

AWS Cloud Map 모든 AWS Cloud Map 컨트롤 플레인 작업을 관리 이벤트로 기록합니다. AWS Cloud Map 로그되는 AWS Cloud Map 컨트롤 플레인 작업 목록은 AWS Cloud Map API 참조를 참조하십시오. CloudTrail

AWS Cloud Map 이벤트 예제

이벤트는 모든 소스의 단일 요청을 나타내며 요청된 API 작업, 작업 날짜 및 시간, 요청 파라미터 등에 대한 정보를 포함합니다. CloudTrail 로그 파일은 공개 API 호출의 정렬된 스택 트레이스가 아니므로 이벤트가 특정 순서로 표시되지 않습니다.

다음 예제는 `CreateHTTPNamespace` 작업을 보여주는 CloudTrail 관리 이벤트를 보여줍니다.

```
{
    "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"
}
```

다음 예제는 작업을 보여주는 CloudTrail 데이터 이벤트를 보여줍니다. `DiscoverInstances`

```
{
            "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"
        }
```

CloudTrail 레코드 내용에 대한 자세한 내용은 AWS CloudTrail 사용 설명서의 CloudTrail 레코드 내용을 참조하십시오.

AWS Cloud Map 리소스에 태그 지정

AWS Cloud Map 리소스 관리를 돕기 위해 태그 형식으로 각 리소스에 고유한 메타데이터를 할당할 수 있습니다. 이 주제에서는 태그를 설명하고 태그를 생성하는 방법을 보여 줍니다.

목차

태그 기본 사항

태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 사용자가 정의하는 키와 선택적 값으로 구성됩니다.

태그를 사용하면 AWS 리소스를 용도, 소유자, 환경과 같은 다양한 기준으로 분류할 수 있습니다. 동일한 유형의 리소스가 많은 경우 할당한 태그에 따라 특정 리소스를 빠르게 식별할 수 있습니다. 예를 들어 AWS Cloud Map 서비스에 태그 집합을 정의하면 각 서비스의 소유자 및 스택 수준을 추적하는 데 도움이 됩니다. 각 리소스 유형에 대해 일관된 태그 키 집합을 고안하는 것이 좋습니다.

태그가 리소스에 자동으로 할당되는 것은 아닙니다. 태그를 추가한 후에는 언제든지 태그 키와 값을 편집하거나 리소스에서 태그를 제거할 수 있습니다. 리소스를 삭제하면 리소스 태그도 삭제됩니다.

태그는 AWS Cloud Map에는 의미가 없으며 엄격하게 문자열로 해석됩니다. 태그의 값을 빈 문자열로 설정할 수 있지만 태그의 값을 Null로 설정할 수는 없습니다. 해당 리소스에 대해 키가 기존 태그와 동일한 태그를 추가하는 경우 새 값이 이전 값을 덮어씁니다.

AWS Management Console, AWS CLI, AWS Cloud Map API를 사용하여 태그 관련 작업을 수행할 수 있습니다.

AWS Identity and Access Management(IAM)를 사용하는 경우 AWS 계정에서 태그를 생성, 편집 또는 삭제할 수 있는 권한이 있는 사용자를 제어할 수 있습니다.

리소스에 태그 지정

새 네임스페이스 및 서비스 또는 기존 AWS Cloud Map 네임스페이스 및 서비스에 태그를 지정할 수 있습니다.

AWS Cloud Map 콘솔을 사용 중인 경우 관련 리소스 페이지의 태그탭을 사용하면 새로 생성된 리소스 또는 기존 리소스에 태그를 언제든지 적용할 수 있습니다.

AWS Cloud Map API, AWS CLI 또는 AWS SDK를 사용 중인 경우 관련 API 작업의 `tags` 파라미터를 사용하여 새 리소스에 태그를 적용하거나 TagResource API 작업을 사용하여 기존 리소스에 태그를 적용할 수 있습니다. 자세한 내용은 TagResource를 참조하세요.

일부 리소스 생성 작업에서는 리소스 생성 시 리소스에 태그를 지정할 수 있습니다. 리소스 생성 중에 태그를 적용할 수 없는 경우 리소스 생성 프로세스는 실패합니다. 이로써 생성 중에 태그를 지정하려는 리소스는 지정된 태그와 함께 생성되거나 전혀 생성되지 않습니다. 생성 시 리소스에 태그를 지정하면 리소스 생성 후 사용자 지정 태그 지정 스크립트를 실행할 필요가 없습니다.

다음 표는 태그를 지정할 수 있는 AWS Cloud Map 리소스와 생성 시 태그를 지정할 수 있는 리소스를 설명합니다.

AWS Cloud Map 리소스 태그 지정 지원

리소스	태그 지원	태그 전달 지원	생성 시 태그 지정 지원(AWS Cloud Map API, AWS CLI, AWS SDK)
AWS Cloud Map 네임스페이스	예	아니요. 네임스페이스 태그는 네임스페이스에 연결된 다른 리소스로 전파되지 않습니다.	예
AWS Cloud Map 서비스	예	아니요. 서비스 태그는 서비스에 연결된 다른 리소스로 전파되지 않습니다.	예

태그 제한

태그에 적용되는 기본 제한은 다음과 같습니다.

- 각 리소스의 최대 태그 수는 50입니다.
- 각 리소스에 대해 각 태그 키는 고유하며 하나의 값만 가질 수 있습니다.
- 최대 키 길이 - UTF-8 형식의 유니코드 문자 128자
- 최대 값 길이 - UTF-8 형식의 유니코드 문자 256자
- 태깅 스키마를 여러 AWS 서비스와 리소스에서 사용하는 경우, 다른 서비스에서는 허용되는 문자에 제한이 있을 수 있다는 점에 주의하세요. 일반적으로 허용되는 문자는 UTF-8로 표시할 수 있는 문자, 숫자 및 공백과 특수 문자 + - = . _ : / @입니다.
- 태그 키와 값은 대/소문자를 구분합니다.
- AWS 용도로 예약된 키 또는 값에는 `aws:`, `AWS:` 또는 이러한 접두사의 대문자 또는 소문자 조합을 사용하지 마십시오. 이 접두사가 지정된 태그 키나 값은 편집하거나 삭제할 수 없습니다. 이 접두사가 지정된 태그는 리소스당 태그 수 제한에 포함되지 않습니다.

CLI 또는 API를 사용한 태그 작업

다음 AWS CLI 명령 또는 AWS Cloud Map API 작업을 사용하여 리소스에 대한 태그를 추가, 업데이트, 나열 및 삭제합니다.

AWS Cloud Map 리소스 태그 지정 지원

작업	API 작업	AWS CLI	AWS Tools for Windows PowerShell
하나 이상의 태그를 추가하거나 덮어씁니다.	TagResource	tag-resource	Add-SDResourceTag
하나 이상의 태그를 삭제합니다.	UntagResource	untag-resource	Remove-SDResourceTag
리소스에 대한 태그를 나열합니다.	ListTagsForResource	list-tags-for-resource	Get-SDResourceTag

다음 예제는 AWS CLI를 사용하여 리소스에 태그를 지정하거나 태그를 제거하는 방법을 보여줍니다.

예제 1: 기존 리소스에 태그 지정

다음 명령은 기존 리소스에 태그를 지정합니다.

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

예제 2: 기존 리소스에서 태그 제거

다음 명령은 기존 리소스에서 태그를 삭제합니다.

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

예제 3: 리소스의 태그 목록 조회

다음 명령은 기존 리소스와 연결된 태그를 나열합니다.

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

일부 리소스 생성 작업에서는 리소스를 생성할 때 태그를 지정할 수 있습니다. 다음 작업은 생성 시 태그 지정을 지원합니다.

작업	API 작업	AWS CLI	AWS Tools for Windows PowerShell
HTTP 네임스페이스 생성	CreateHttpNamespace	create-http-namespace	New-SDHttpNamespace
DNS를 기반으로 프라이빗 네임스페이스 생성	CreatePrivateDnsNamespace	create-private-dns-namespace	New-SDPrivateDnsNamespace
DNS를 기반으로 공용 네임스페이스 생성	CreatePublicDnsNamespace	create-public-dns-namespace	New-SDPublicDnsNamespace
서비스 생성	CreateService	create-service	New-SDService

AWS Cloud Map 서비스 할당량

AWS Cloud Map 리소스에는 다음과 같은 계정 수준 서비스 할당량이 적용됩니다. 나열된 각 할당량은 리소스를 생성하는 각 AWS 지역에 적용됩니다. AWS Cloud Map

명칭	기본값	조정 가능	설명
인스턴스당 사용자 지정 속성	지원되는 각 리전: 30개	아니요	인스턴스 등록 시 지정한 사용자 지정 속성의 최대 개수입니다.
DiscoverInstances 계정당 작업, 버스트 속도	지원되는 각 리전: 2,000개	예	단일 계정의 통화 DiscoverInstances 작업에 대한 최대 버스트 속도입니다.
DiscoverInstances 계정별 오퍼레이션 안정적 속도	지원되는 각 리전: 1,000개	예	단일 계정에서 통화 DiscoverInstances 작업을 할 수 있는 최대 고정 속도입니다.
DiscoverInstancesRevision 계정별 오퍼레이션 요율	지원되는 각 리전: 3,000	예	단일 계정에서 DiscoverInstancesRevision 작업을 호출할 수 있는 최대 요율입니다.
네임스페이스당 인스턴스	지원되는 각 리전: 2,000	예	동일한 네임스페이스를 사용하여 등록할 수 있는 최대 서비스 인스턴스 수입니다.
서비스당 인스턴스	지원되는 각 리전: 1,000개	아니요	동일한 서비스를 사용하여 리전에서 등록할 수 있는 최대 인스턴스 수.

명칭	기본값	조정 가능	설명
리전당 네임스페이스	지원되는 각 지역: 50	예	리전당 생성할 수 있는 최대 네임스페이스 수.

* 사용자가 네임스페이스를 생성하면 Amazon Route 53 호스팅 영역이 자동으로 생성됩니다. 이 호스팅 영역은 AWS 계정으로 생성할 수 있는 호스팅 영역 수의 할당량에 포함됩니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 호스팅 영역에 대한 할당량을 참조하세요.

** AWS Cloud Map 의 DNS 네임스페이스 인스턴스를 늘리려면 호스팅 영역 Route 53 한도당 레코드 수를 늘려야 하며, 이 경우 추가 요금이 발생합니다.

AWS Cloud Map 서비스 할당량 관리

AWS Cloud Map 중앙 위치에서 할당량을 보고 관리할 수 있는 AWS 서비스인 Service Quotas와 통합되었습니다. 자세한 내용은 Service Quotas 사용 설명서의 Service Quotas는 무엇입니까?를 참조하세요.

Service Quotas를 사용하면 서비스 할당량의 AWS Cloud Map 가치를 쉽게 조회할 수 있습니다.

AWS Management Console

를 사용하여 서비스 할당량을 보려면 AWS Cloud MapAWS Management Console

1. https://console.aws.amazon.com/servicequotas/에서 Service Quotas 콘솔을 엽니다.
2. 탐색 창에서 AWS 서비스를 선택합니다.
3. AWS 서비스 목록에서 AWS Cloud Map를 검색하여 선택합니다.
4. 의 서비스 할당량 목록에서 서비스 할당량 이름 AWS Cloud Map, 적용된 값 (사용 가능한 경우), AWS 기본 할당량, 할당량 값 조정 가능 여부를 확인할 수 있습니다.

   설명과 같은 서비스 할당량에 대한 추가 정보를 보려면 할당량 이름을 선택하여 할당량 세부 정보를 불러오세요.
5. (선택 사항) 할당량 증가를 요청하려면 늘리려는 할당량을 선택하고 계정 수준에서 증가 요청을 선택합니다.

Service Quotas 사용 설명서를 사용하여 서비스 할당량을 더 자세히 다루려면 Service Quotas 사용 설명서를 AWS Management Console 참조하십시오.

AWS CLI

를 사용하여 서비스 할당량을 보려면 AWS Cloud MapAWS CLI

다음 명령을 실행하여 기본 AWS Cloud Map 할당량을 확인합니다.

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

다음 명령을 실행하여 적용된 AWS Cloud Map 할당량을 확인합니다.

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

를 사용하여 서비스 할당량을 처리하는 방법에 대한 자세한 내용은 Service Quotas 명령 참조를 참조하십시오. AWS CLI AWS CLI 할당량 증가를 요청하려면 AWS CLI 명령 참조에서 `request-service-quota-increase` 명령을 참조하세요.

AWS Cloud Map DiscoverInstances API 요청 제한

AWS Cloud Map 지역별로 각 AWS 계정에 대한 DiscoverInstancesAPI 요청을 제한합니다. 스로틀링은 서비스 성능을 개선하고 모든 고객에게 공정한 사용을 제공하는 데 도움이 됩니다. AWS Cloud Map 스로틀을 사용하면 API에 대한 호출이 최대 허용 AWS Cloud Map DiscoverInstancesDiscoverInstancesAPI 요청 할당량을 초과하지 않도록 할 수 있습니다. DiscoverInstances다음 소스 중 하나에서 발생하는 API 호출에는 요청 할당량이 적용됩니다.

- 타사 애플리케이션
- 명령줄 도구
- 콘솔 AWS Cloud Map

API 제한 할당량을 초과하면 `RequestLimitExceeded` 오류 코드가 표시됩니다. 자세한 설명은 the section called “요청 속도 제한” 섹션을 참조하세요.

제한 적용 방법

AWS Cloud Map 토큰 버킷 알고리즘을 사용하여 API 스로틀링을 구현합니다. 이 알고리즘을 사용하면 계정에 특정 수의 토큰을 보관하는 버킷이 있습니다. 버킷의 토큰 수는 지정된 초당 제한 할당량을 나타냅니다. 단일 리전에는 버킷이 하나 있으며 이는 해당 리전의 모든 엔드포인트에 적용됩니다.

요청 속도 제한

스로틀링은 만들 수 있는 DiscoverInstancesAPI 요청 수를 제한합니다. 각 요청은 버킷에서 하나의 토큰을 제거합니다. 예를 들어 DiscoverInstancesAPI 작업의 버킷 크기는 2,000토큰이므로 1초에 최대 2,000개의 DiscoverInstances요청을 할 수 있습니다. 매초 요청이 2,000개를 초과하면 병목 현상이 발생하고 해당 초 내에 나머지 요청은 실패합니다.

버킷은 설정된 속도로 자동으로 다시 채워집니다. 버킷 용량이 부족하면 버킷 용량에 도달할 때까지 매초마다 정해진 수의 토큰이 다시 추가됩니다. 다시 채우기 토큰이 도착했을 때 버킷 용량이 다 차면 해당 토큰은 폐기됩니다. DiscoverInstancesAPI 작업의 버킷 크기는 2,000토큰이고 리필 비율은 초당 1,000토큰입니다. 1초에 2,000개의 DiscoverInstancesAPI 요청을 하면 버킷은 즉시 0개의 토큰으로 줄어듭니다. 그러면 최대 용량 2,000개에 도달할 때까지 매초마다 최대 1,000개의 토큰이 버킷에 다시 채워집니다.

버킷에 추가된 토큰은 그대로 사용할 수 있습니다. API를 요청하기 전에 버킷이 최대 용량이 될 때까지 기다릴 필요가 없습니다. 1초에 2,000개의 DiscoverInstancesAPI 요청을 실행하여 버킷을 고갈시킨 경우에도 필요한 기간 동안 그 이후로는 1초마다 최대 1,000개의 DiscoverInstancesAPI 요청을 할 수 있습니다. 즉, 다시 채우기 토큰이 버킷에 추가되면 즉시 사용할 수 있습니다. 버킷은 초당 API 요청 횟수가 다시 채우기 속도보다 적은 경우에만 최대 용량까지 다시 채워지기 시작합니다.

재시도 또는 일괄 처리

API 요청이 실패하는 경우 애플리케이션에서 요청을 재시도해야 할 수 있습니다. API 요청 수를 줄이려면 연속적인 요청 사이에 적절한 절전 간격을 사용하세요. 최상의 결과를 얻으려면 절전 시간 간격을 늘리거나 가변적으로 사용합니다.

휴면 간격 계산

API 요청을 폴링하거나 재시도해야 하는 경우 지수 백오프 알고리즘을 사용하여 API 호출 간 절전 시간 간격을 계산하는 것이 좋습니다. 연속적인 오류 응답에 대한 재시도 사이의 대기 시간을 점진적으로 늘리면 실패한 요청 수를 줄일 수 있습니다. 이 알고리즘에 대한 자세한 내용 및 구현 예는 에서 오류 재시도 및 지수 백오프를 참조하세요. AWS

API 제한 할당량 조정

계정에 대한 API 제한 할당량 증가를 요청할 수 있습니다. AWS 할당량 조정을 요청하려면 AWS Support Center에 문의하세요.

관련 정보

아래에 AWS Cloud Map를 이용할 때 참조할 수 있는 관련 리소스가 나와 있습니다.

주제

- AWS 리소스
- 타사 도구 및 라이브러리

AWS 리소스

다음 표에는 이 서비스를 이용할 때 참조할 수 있는 관련 리소스가 나와 있습니다.

- 교육 및 워크숍 – 역할 기반의 과정 및 전문 과정은 물론 자습형 실습에 대한 링크를 통해 AWS 기술을 연마하고 실무에 도움이 되는 경험을 쌓을 수 있습니다.
- AWS 개발자 센터 – 자습서를 살펴보고, 도구를 다운로드하고, AWS 개발자 이벤트에 대해 알아보세요.
- AWS 개발자 도구 – AWS 애플리케이션을 개발 및 관리하기 위한 개발자 도구, SDK, IDE 도구 키트 및 명령줄 도구 링크입니다.
- 시작하기 리소스 센터 - AWS 계정을(를) 설정하고 AWS 커뮤니티에 가입하고 첫 번째 애플리케이션을 시작하는 방법을 알아보세요.
- 실습 자습서 – 단계별 자습서에 따라 AWS에서 첫 번째 애플리케이션을 시작하세요.
- AWS 백서 – AWS 솔루션 아키텍트 또는 기타 기술 전문가가 아키텍처, 보안 및 경제 등의 주제에 대해 작성한 포괄적 AWS 기술 백서 목록의 링크입니다.
- AWS Support 센터 – AWS Support 사례를 생성하고 관리할 수 있는 허브입니다. 또한 포럼, 기술 FAQ, 서비스 상태 및 AWS Trusted Advisor 등의 기타 유용한 자료에 대한 링크가 있습니다.
- AWS Support – 클라우드에서 1대 1로 애플리케이션을 구축 및 실행하도록 지원하는 빠른 응답 지원 채널인 AWS Support에 대한 정보가 포함된 기본 웹 페이지입니다.
- 문의처 - AWS 결제, 계정, 이벤트, 침해 및 기타 문제에 대해 문의할 수 있는 중앙 연락 창구입니다.
- AWS 사이트 약관 – 저작권 및 상표, 사용자 계정, 라이선스 및 사이트 액세스와 기타 주제에 대한 세부 정보입니다.

타사 도구 및 라이브러리

AWS 리소스 외에 다음의 타사 도구 및 라이브러리도 AWS Cloud Map 작업 시 유용할 수 있습니다.

- 클라우드 애플리케이션 프레임워크(AWS Cloud Map) – AWS Cloud Map를 이용하여 대기 중 메시지, 이벤트 게시, 클라우드 함수 호출 등의 일반적인 클라우드 플랫폼 작업을 처리하는 라이브러리입니다.
- Kubernetes용 ExternalDNS – Kubernetes 수신 및 서비스에 대한 Amazon Route 53 및 AWS Cloud Map를 포함하여 외부 DNS 서비스를 구성하기 위한 도구입니다.

에 대한 문서 기록 AWS Cloud Map

다음 표에서 AWS Cloud Map 개발자 설명서의 중요한 업데이트 및 새 기능이 나와 있습니다. 사용자로부터 받은 의견을 수렴하기 위해 설명서가 자주 업데이트됩니다.

변경 사항	설명	날짜
튜토리얼 추가	일반적인 사용 사례를 보여주는 두 개의 자습서가 추가되었습니다. AWS Cloud Map	2024년 3월 27일
CloudTrail 통합 설명서가 업데이트되었습니다.	CloudTrail to log API AWS Cloud Map 활동과의 통합을 설명하는 설명서가 업데이트되었습니다.	2024년 3월 20일
관리형 정책 업데이트	`AWSCloudMapDiscoverInstanceAccess` , `AWSCloudMapRegisterInstanceAccess` , `AWSCloudMapReadOnlyAccess` 정책이 업데이트되었습니다.	2023년 9월 20일
Cloud Map 및 AWS PrivateLink	이제 를 AWS PrivateLink 사용하여 VPC와 사이에 프라이빗 연결을 생성할 수 있습니다. AWS Cloud Map	2023년 9월 15일
관리형 정책 업데이트	`AWSCloudMapDiscoverInstanceAccess` 정책이 업데이트되었습니다.	2023년 8월 15일
AWS 파이썬용 SDK	Python 명령줄 예제가 추가되었습니다.	2022년 9월 13일

IPv6 지원	API 엔드포인트는 이제 IPv6 네트워크에서만 사용할 수 있습니다.	2022년 1월 28일
서비스 인스턴스 검색	AWS Cloud Map DNS 쿼리를 사용하지 않고 DiscoverInstancesAPI 작업으로만 검색할 수 있는 DNS 쿼리를 지원하는 네임스페이스에서 서비스를 생성하는 지원이 추가되었습니다.	2021년 3월 24일
리소스에 태깅	AWS Cloud Map 를 사용하여 네임스페이스와 서비스에 메타데이터 태그를 추가할 수 있는 지원이 추가되었습니다. AWS Management Console	2021년 2월 8일
리소스에 태깅	AWS Cloud Map 및 API를 사용하여 네임스페이스와 서비스에 메타데이터 태그를 추가할 수 있는 지원이 추가되었습니다. AWS CLI	2020년 6월 22일
최초 릴리스	이 문서는 첫 번째 AWS Cloud Map 개발자 안내서 릴리스입니다.	2018년 11월 28일

AWS 용어집

최신 AWS 용어는 AWS 용어집 참조의 AWS 용어집을 참조하세요.

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.