



Amazon Elastic Container Service (Amazon ECS) Anywhere

AWS Black Belt Online Seminar

落水 恭介

Solutions Architect

2021/12



このセッションで扱うこと

- Amazon ECS Anywhere とは
- Amazon ECS Anywhere のアーキテクチャ
- Amazon ECS Anywhere の開始方法
- Amazon ECS Anywhere の料金



Amazon Elastic Container Service (Amazon ECS)



パワフル & シンプル

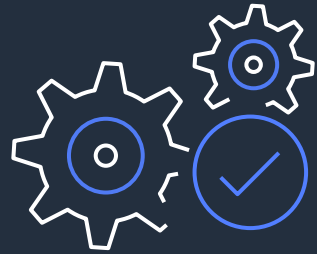
- クラウドでコンテナを本番環境利用するための
オーケストレーター
- 他の AWS サービスと高度に連携しコンテナを大規模に実行
- フルマネージドなコントロールプレーン
- 多様なワークロードをサポートする「タスク」
「サービス」というシンプルなリソース表現

[AWS Black Belt Online Seminar] CON201 ECS 入門

<https://aws.amazon.com/jp/blogs/news/aws-black-belt-online-seminar-con201-ecs-overview/>



Amazon ECS Anywhere に対するお客様のニーズ



一貫性のある オペレーション

オンプレミスとクラウドのそれぞれでアプリケーションを運用する際にオペレーションモデルを分けたくない



設備投資の回収

クラウドへ移行する前に償却する必要があるデータセンターに設備投資を行なっている



コンプライアンス 要件

市場や業界の規制により、インフラストラクチャの大部分を自身で所有する必要がある

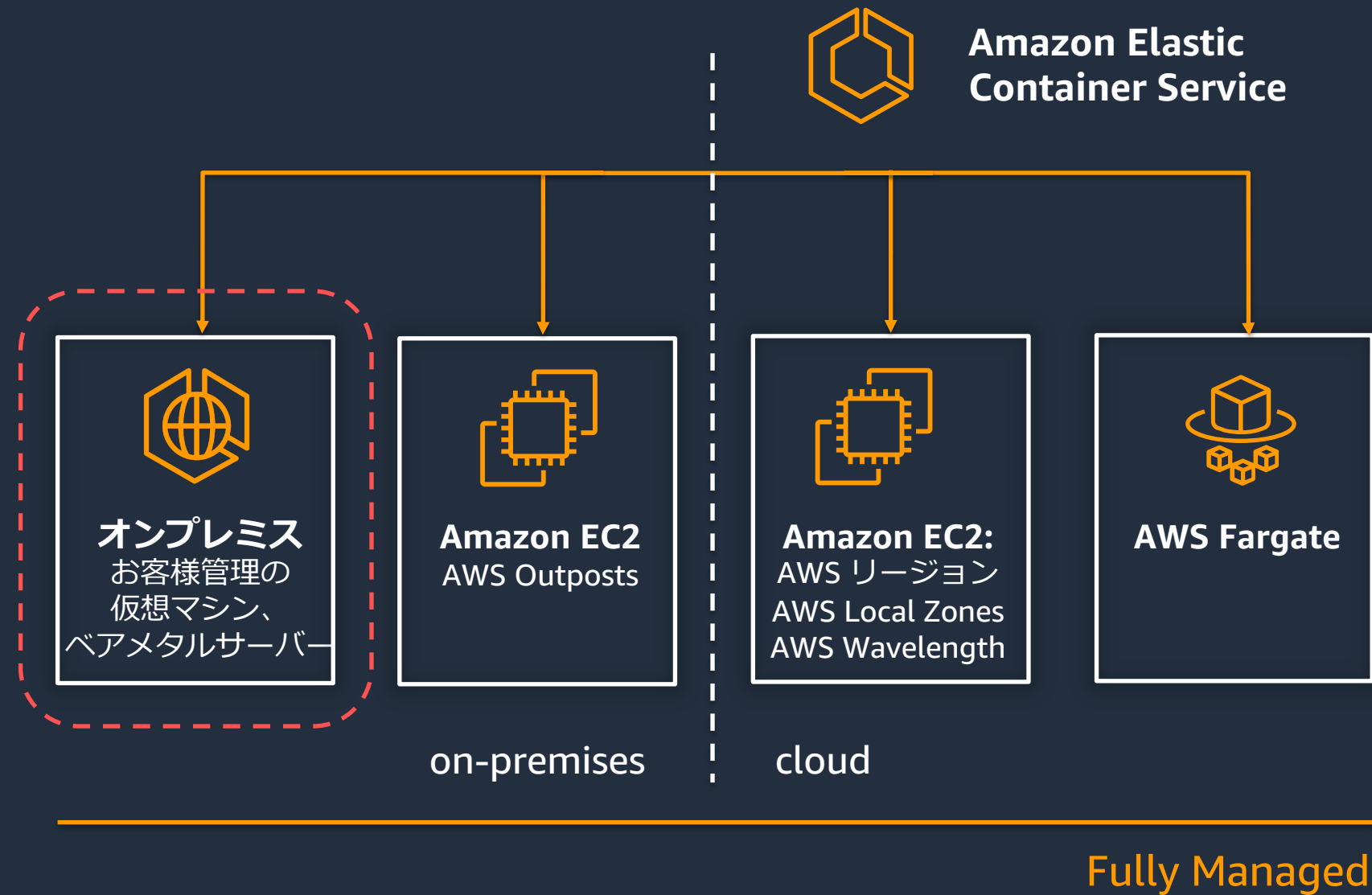


データに近接した ワークロード

大規模なデータセットを低レイテンシで処理するために、データの近くにアプリケーションをデプロイする必要がある

Amazon ECS Anywhere

お客様の管理するインフラストラクチャでコンテナを実行

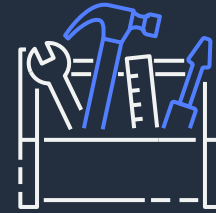


Amazon ECS Anywhere の主な特徴



クラウド上のフルマネージド なコントロールプレーン

オンプレミスでコンテナ
オーケストレーターの実行や
アップデート、メンテナンスが不要



ツールやガバナンスの 一貫性

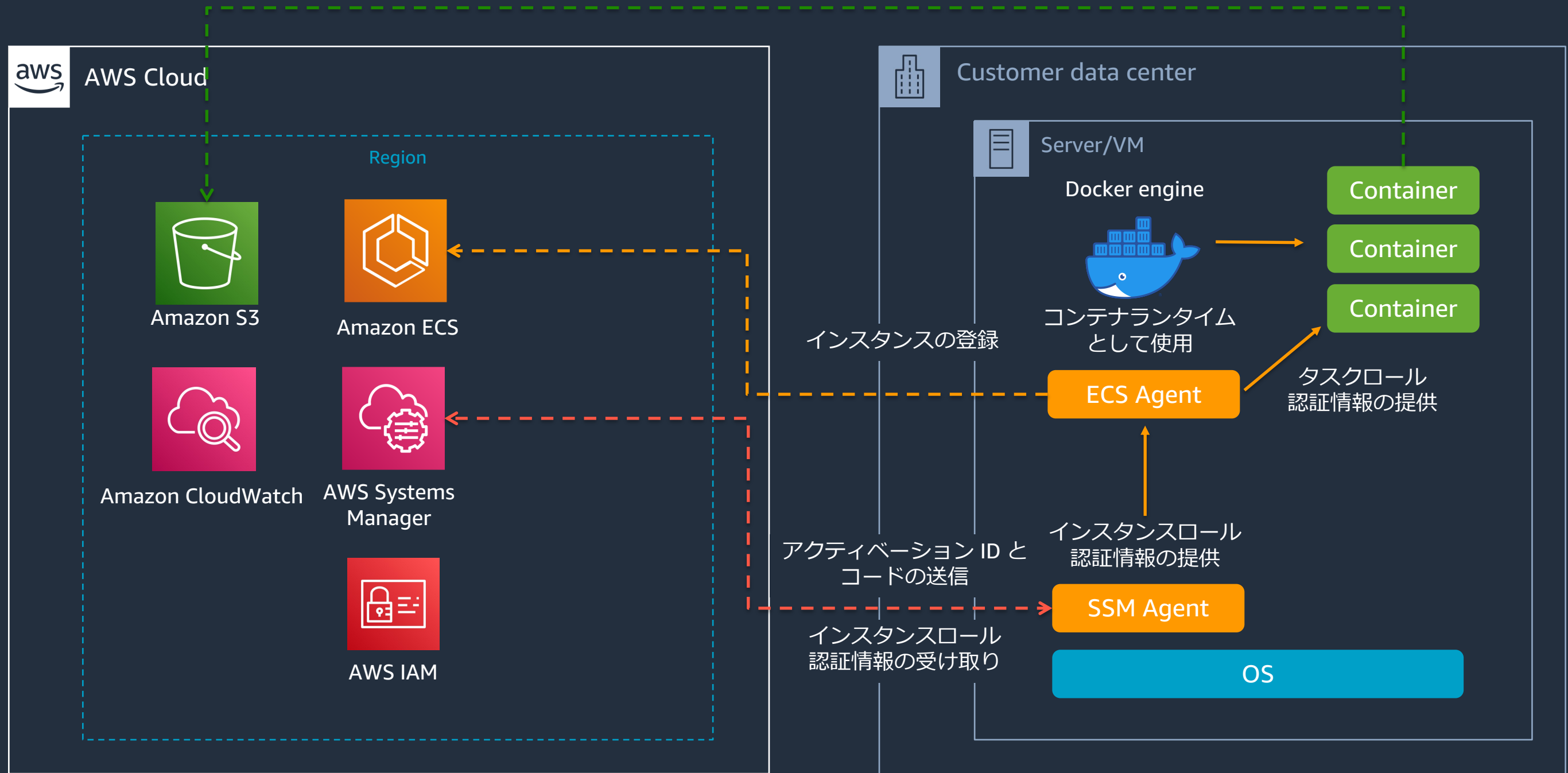
運用している環境によらず、
すべてのコンテナアプリケーションに
同じツールや API を使用



ハイブリッド フットプリントの管理

オンプレミス環境で
アプリケーションを実行し、
移行の際はクラウドへの
拡張が容易に

ECS Anywhere のアーキテクチャ



Amazon ECS Anywhere の開始方法

(前提) オンプレミスサーバーや仮想マシン (VM) の用意

1. 外部インスタンスで利用する ECS Anywhere IAM ロールの作成
2. ECS クラスターへの外部インスタンスの登録
3. 外部インスタンスと互換性のあるタスク定義の作成
4. スタンドアロンタスクの実行、または ECS サービスの作成

オンプレミスで管理されるインスタンスの前提条件

AWS API サービスエンドポイントへのアウトバウンド接続

- インターネット
- Direct Connect または VPN を経由した VPC (PrivateLink)

エンドポイントとコンテナランタイムスタック

- Systems Manager (SSM) エージェント
- ECS エージェント
- Docker engine

インストールスクリプトを提供

サポートされている Linux ディストリビューション/システムアーキテクチャ ※1

- Ubuntu 18.04, 20.04
- RHEL/CentOS 7, 8
- Fedora 32, 33 / Debian 9, 10 / openSUSE Tumbleweed / SUSE Enterprise Server 15
- x86_64 および ARM64

※1: 2021 年 12 月 13 日時点

外部インスタンスの登録プロセス

ECS コンソールにて取得できる登録コマンドを外部マシンで実行する

Externalインスタンスの登録

ステップ 2: Externalインスタンスの登録

コマンド

```
curl --proto "https" -o "/tmp/ecs-anywhere-install.sh"
"https://amazon-ecs-agent.s3.amazonaws.com/ecs-anywhere-install-latest.sh" && bash /tmp/ecs-anywhere-install.sh --region "ap-northeast-1" --cluster "ecs-anywhere-cluster" --activation-id
```

このコマンドをExternalインスタンスの root として実行して、登録プロセスを完了します。コマンドが root として実行されていない場合、エラーが返されます。

ステータス: **ALL** ACTIVE DRAINING

Filter by attributes (click or press down arrow to view filter options)

<input type="checkbox"/>	Externalインスタンス	接続されたエージェント	ステータス	実行中のタスクの数	使用可能な CPU	使用可能なメモリ (MIB)
<input type="checkbox"/>	true	true	ACTIVE	0	2048	3683
<input type="checkbox"/>	false	true	ACTIVE	0	2048	7733
<input type="checkbox"/>	false	true	ACTIVE	0	2048	7733

タスク定義の作成

requiresCompatibilities パラメーターに EXTERNAL を指定したタスク定義を登録

起動タイプの互換性の選択

タスクを起動する場所に基づいて、タスク定義との互換性を持たせる起動タイプを選択します。

FARGATE



タスクサイズに基づく価格

ネットワークモード awsipc が必要

AWS が管理するインフラストラクチャ、管理する Amazon EC2 インスタンスはありません

EC2



リソース使用量に基づく価格

複数のネットワークモードを使用可能

Amazon EC2 インスタンスを使用した自己管理インフラストラクチャ

EXTERNAL



使用した他の AWS サービスのインスタンス時間と追加料金に基づく料金

ECS Anywhere による自己管理型のオンプレミスインフラストラクチャ

```
{
  "requiresCompatibilities": [
    "EXTERNAL"
  ],
  "containerDefinitions": [{
    "name": "nginx",
    "image": "public.ecr.aws/nginx/nginx:latest",
    "memory": 256,
    "cpu": 256,
    "essential": true,
    "portMappings": [{
      "containerPort": 80,
      "hostPort": 8080,
      "protocol": "tcp"
    }]
  }],
  "networkMode": "bridge",
  "family": "nginx"
}
```

ネットワーク切断時の挙動

切断中

- タスクは引き続き実行される（終了しない限り）
- 新しいタスクを開始することはできない
- タスクをスケールインすることはできない



再接続時

- コントロールプレーンがインスタンスの状態を再評価
- タスクロールの認証情報をリフレッシュ
- PENDING 状態のタスクを配置・開始



オンプレミスにデプロイする際の基本的な制限

※ 2021 年 12 月時点

- bridge、host、none ネットワークモードを使用
 - awsvpc ネットワークモードは未サポート
- ECS Service Discovery は未サポート
- サービスのロードバランシング機能は未サポート
- ECS Capacity Providers は未サポート
- ECS Exec は未サポート
- ..., etc

https://docs.aws.amazon.com/ja_jp/AmazonECS/latest/developerguide/ecs-anywhere.html#ecs-anywhere-considerations



GPU ベースのワークロード管理

1. ECS コンソールで取得した登録コマンドに `--enable-gpu` フラグを追加して実行
2. GPU デバイスを利用する ECS タスク定義を作成
 - `containerDefinitions.resourceRequirements` で GPU を指定
3. 2 で登録した ECS タスク定義を用いて ECS タスクを実行

```
curl --proto "https" \  
  -o "/tmp/ecs-anywhere-install.sh" "https://amazon-ecs-  
agent.s3.amazonaws.com/ecs-anywhere-install-latest.sh" \  
&& bash /tmp/ecs-anywhere-install.sh \  
  --region ""<YOUR_ECS_CLUSTER_REGION>" \  
  --cluster "<YOUR_ECS_CLUSTER_NAME>" \  
  --activation-id "xxxxx" \  
  --activation-code "xxxxx" \  
  --enable-gpu
```

```
{  
  "containerDefinitions": [  
    {  
      (中略)  
      "resourceRequirements": [{  
        "type": "GPU",  
        "value": "1"  
      }]  
    },  
    "family": "example-ecs-anywhere-gpu"  
  ]  
}
```

<https://aws.amazon.com/blogs/containers/running-gpu-based-container-applications-with-amazon-ecs-anywhere/>

料金体系

- ECS Anywhere によって管理される実行中のインスタスの料金のみが発生
- マネージド ECS Anywhere オンプレミスインスタスごとに、
1 時間あたり **0.01025 USD**
 - ECS Anywhere でインスタスを登録してからインスタスが登録解除されるまでの、ECS Anywhere がオンプレミスインスタスを管理している時間数に基づいて計算
- **無料利用枠**には、すべてのリージョンのアカウントごとに 6 か月間、毎月 2200 インスタス時間が含まれる

<https://aws.amazon.com/jp/ecs/anywhere/pricing/>

本セッションの担当: 落水 恭介

落水 恭介 (Ochimizu Kyosuke)

ソリューションアーキテクト

- Sler
- 教育業界ベンチャー
- Cloud Integrator
- サポートチーム / アマゾン ウェブ サービス ジャパン
- 現在のロール



好きな AWS サービス:



Amazon Elastic Container Service (Amazon ECS)