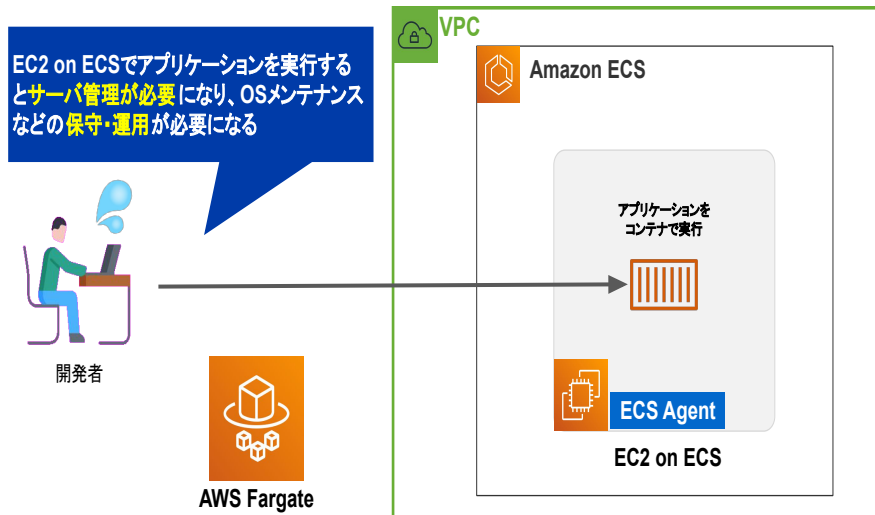




AWS Fargateについて解説します。

AWS Fargate



まず前回のECS講座の復習となります。

ECSでは**EC2**を利用して、アプリケーションをコンテナで実行する方法について解説しました。

EC2でコンテナを実行するのはいいのですが、**サーバ管理が必要になり、OSやメンテナンス、セキュリティ管理などの保守・運用が必要になります。**

コンテナを使うのにこのような保守・運用作業が発生するのは手間がかかりますよね。

このような場合、EC2ではなく、**Fargate**を利用することで、その管理を不要にすることができます。

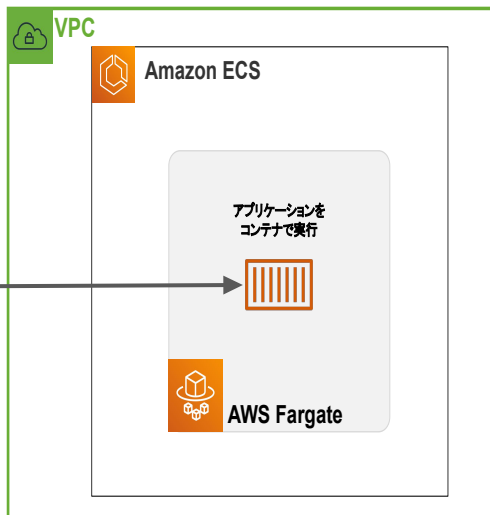
AWS Fargate

AWS上でコンテナをサーバーレスで実行することができる機能

AWS Fargateでアプリケーションを実行するとサーバ管理が不要になり、アプリケーションの開発作業に注力できる



開発者



CloudTech

AWS Fargateとは**AWS上でコンテナをサーバーレスで実行することができる機能**です。

先ほどまでEC2で実行されていたコンテナの実行環境が **AWS Fargate**に置き換わります。

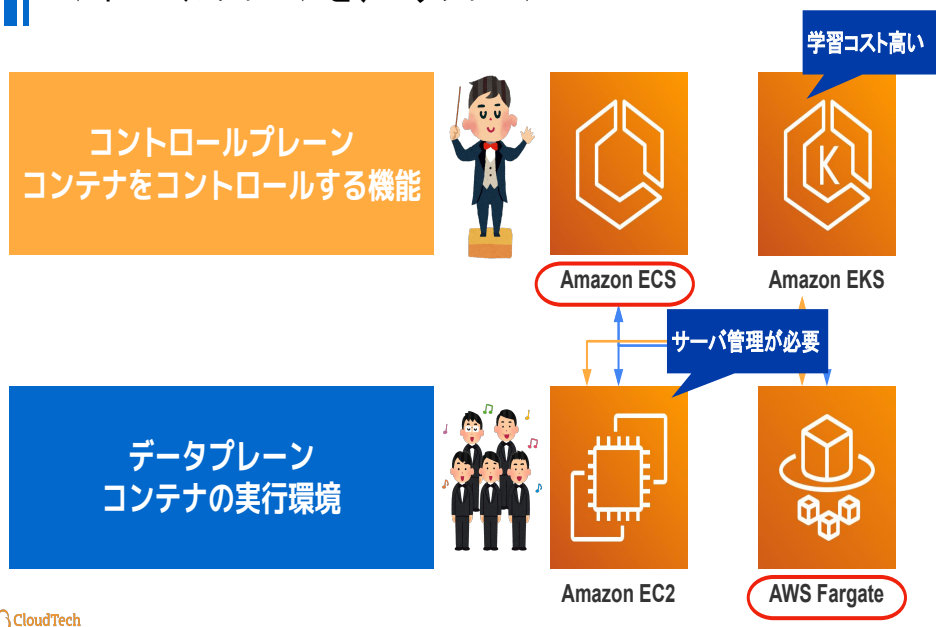
ちなみにサーバーレスとは実際にサーバがない訳ではなく、裏ではAWSが管理しています。我々AWSの利用者にとって、サーバ運用で必要となるOSのメンテナンス等の管理が不要になることを指します。

他にもLambdaやApp Runnerなどもサーバーレスのサービスとなっています。

このAWS Fargateを利用することで、**サーバ管理が不要になり、アプリケーションの開発作業に注力できる**ようになります。

金融系システムなどではサーバーのセキュリティ審査なども厳しく、ここをAWSのセキュリティ基準に丸投げしたり、セキュリティソフトの費用や管理もいなくなるメリットもあります。

コントロールプレーンとデータプレーン



これまでECS・EC2・Fargateと出てきましたが、それぞれコンテナを実行する場合に役割があります。「コントロールプレーン」と「データプレーン」の用語とともに、少し整理しましょう。

プレーンとは、飛行機という意味もありますが、「階層」を意味する名詞もあります。

まずコントロールプレーンとは、**コンテナをコントロールする機能**のことを指します。

コントロールするサービスということで、**合唱団の指揮者**に例えるとわかりやすいと思います。

コントロールプレーンでは、前回の講座で学んだ AWS独自のオーケストレーションサービスである **Amazon ECS**とオープンソースのKubernetesをベースにした**Amazon EKS**を選択することができます。

対して、データプレーンは**コンテナの実行環境**のことを指します。
こちらは指揮者からのコントロールを受けるということで、**合唱団のメンバー**に例えるとわかりやすいと思います。
コントロールプレーンでは、前回の講座で学んだ仮想サーバのサービスである**Amazon EC2**と本講座で学ぶサーバレスのサービスである**AWS Fargate**を選択することができます。

コントロールする階層か、データの階層か、それぞれのプレーンに分かれているイメージですね。

これらのサービスを組み合わせることで、AWS上でのコンテナ実行を行うことができるようになります。
データプレーンとコントロールプレーンの組み合わせには、

- **Fargate+ECS**
- **Fargate+EKS**
- **EC2+ECS**
- **EC2+EKS**

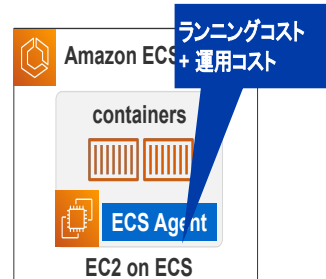
があります。

一般的にAWSではじめてコンテナの利用を始める場合には、「**Fargate+ECS**」の組み合わせを選択することが多くなります。
理由としては、EKSはバージョンアップへの追従が必須となるためECSに比べ学習コストが高くなるためです。
また、EC2は冒頭でお伝えのようにサーバ管理が必要になり、OSメンテナンスなどの保守・運用が必要になります。
そのため、「**Fargate+ECS**」の組み合わせがコンテナの利用を始める場合に選択することが多くなります。

データプレーンの違い(EC2・Fargate)

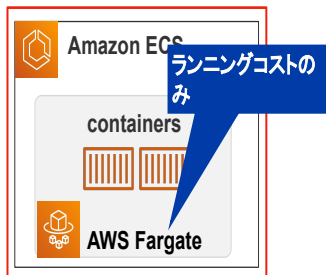
EC2起動タイプ

- EC2上にてDockerコンテナを実行
- EC2はユーザーにて管理
 - 詳細なサーバー設定が可能
 - スケーリングを調整



Fargate起動タイプ

- サーバーレス実行エンジン
- EC2の管理が不要(AWS側にて管理)
- コンテナをすぐに実行可能
- EC2に比べ、ややコストが高い



続いて、EC2・Fargateそれぞれのデータプレーンを利用する場合の特徴について解説します。

まず起動タイプがEC2の場合です。

EC2の場合、EC2上にてDockerコンテナを実行することになります。この際、EC2はユーザーにて管理が必要となります。仮想サーバ上で動作するEC2の詳細なサーバー設定が可能ですが、スケーリングを調整したりといった管理をユーザーが担当します。

大量のデータを高速で読み書きするなど、OSレベルのチューニングが重要となるような特殊な状況を除き、EC2を管理しなければならないのは少し辛いです。

次に起動タイプがFargateの場合です。

Fargateの場合、サーバーレスでコンテナを実行することになります。つまり、EC2起動タイプで必要であったホストOSの設定やコンテナエージェントのインストール、スケーリングの調整などの

EC2の管理が不要で、AWS側が管理してくれます。

よってユーザーはこれらの設定なしで、**コンテナをすぐに実行すること**が可能になります。

一方、AWSがサーバの管理をやってくれているため、**EC2に比べ、ややコストが高い**といった特徴もあります。

「EC2とFargateどちらを利用するのが良いの？」

と疑問に思う方がいると思います。

結論、自分たちでサーバーを管理しないといけないといった特別な要件がない限りは**Fargateを使うようにしましょう！**

理由は、先ほどのコントロールプレーンとデータプレーンでお伝えしように、Fargateを選択した方がサーバ管理が不要になるといった恩恵を受けれるためです。

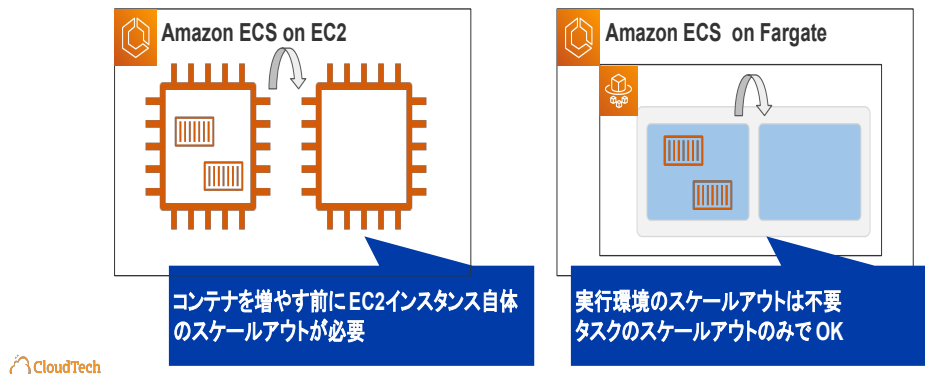
Fargateの方が若干ランニングコストが高くなりますが、**インスタンス管理の運用コストが発生しません。**

一方、EC2だとランニングコストは安くなるが、**インスタンス管理の運用コストが発生します。**

トータルのコストを考えると、EC2の方が安くなるわけではないので、運用コストがかからないFargateを利用した方が良いという結論になります。

AWS Fargateのメリット

- **ホストマシンのOSの管理**が必要ない
- **インスタンスタイプ**や**クラスター管理**が不要になる
- **スケーリング**が容易にできる



それでは、これまで解説したことを踏まえて AWS Fargateを利用するメリットをおさらいしましょう。

1つ目は「**ホストマシンのOSの管理が必要ない**」という点です。AWS Fargateを利用することで、コンテナ実行環境のホストマシンなどの管理が必要なくなるというメリットがあります。

これにより、インフラ専任の技術者がいなくても、手軽に開発を行えます。

このメリットは、SAA試験でも頻出の観点なのでよく覚えておきましょう。

2つ目は「**インスタンスタイプやクラスター管理が不要になる**」という点です。

Fargateを利用してコンテナを実行するときはインスタンスタイプの指定は不要で、意識するのは、実行時に必要な CPUやメモリの組み合わせのみです。

使用量に応じて課金されるため、過剰なプロビジョニングによる

無駄な費用が発生しません。

3つ目は「**スケーリングが容易にできる**」という点です。

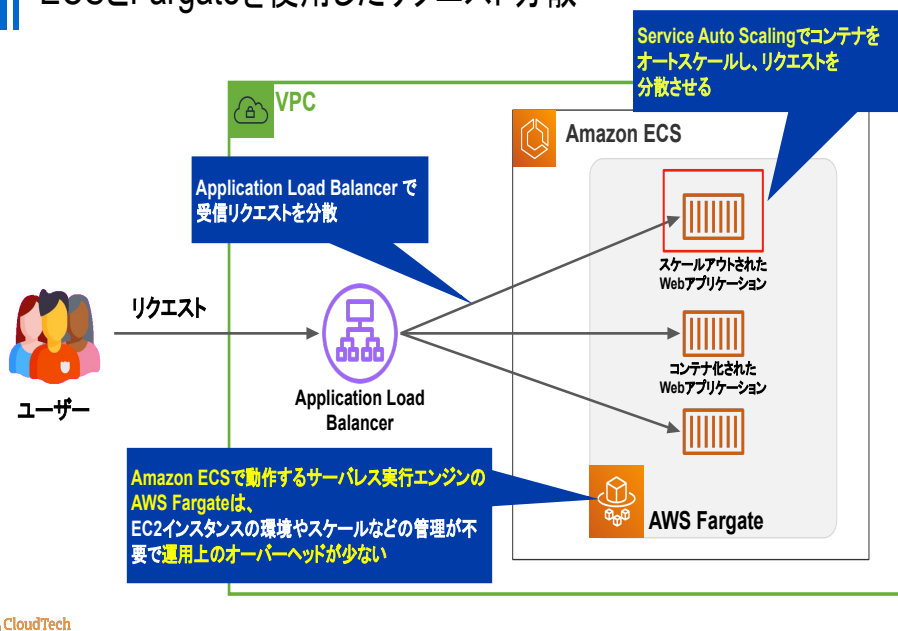
EC2・Fargate両方ともAutoScalingの機能により、稼働するアプリケーションの実行環境の数を増やしたり減らしたりして、安定的にサービスが稼働できるようにします。

しかしEC2の場合は、コンテナを増やす前に **EC2インスタンス自体つまり実行環境のスケールアウトが必要となります**。

一方、Fargateの場合は**実行環境のスケールアウトは不要で、タスクのAutoScaling設定のみ**でOKです。

そのため、EC2に比べて、スケーリングが容易にできるといったメリットがあります。

ECSとFargateを使用したリクエスト分散



先ほど説明したAWS Fargateのメリットも踏まえて、Amazon ECSとAWS Fargateを使用してWebアプリケーションをデプロイする時の**簡単な構成図**をみてみましょう。

1. **ユーザーからのリクエスト**が、VPC内のApplication Load Balancer (ALB)に到達します。
2. **ALBが受信したリクエスト**を、Amazon ECS上で実行中のコンテナ化されたWebアプリケーションに振り分けます。
3. Amazon ECSは、必要に応じて**コンテナのスケールを自動的に**行います。このスケール処理には、AWS Fargateが活用されます。
4. AWS Fargateは、Amazon ECSコンテナを実行するための**サーバーレス環境を提供します**。従来の仮想マシンインスタンスを管理する必要がなく、自動で起動・停止されるので、コストとリソース管理が最適化されます。

このように、FargateはAWS上でコンテナ化された Webアプリを

効率的にデプロイし、負荷に応じてスケーリングすることができます。

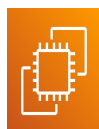
AWS Fargateのデメリット

🚀 > **ホスト管理**に直接介入できない

🚀 > **コンテナの起動**が遅い

🚀 > EC2に比べ、**ややコストが高い**

約\$190 ↑



Amazon EC2



\$1086.24



AWS Fargate



\$1273.3536

■ EC2 m5.large, Linux, Tokyo region
vCPU: 2
Mem : 8 GiB

料金 : 0.124 USD / hour(2024/3/13時点)

■ Fargate, Tokyo region,
EC2と同様に 2vCPU, 8GB メモリで計算
vCPU: $2 \times 0.05056 \text{ USD / hour} = 0.10112 \text{ USD / hour}$
Mem : $8 \times 0.00553 \text{ USD / hour} = 0.04424 \text{ USD / hour}$

料金: $0.10112 \text{ USD / hour} + 0.04424 \text{ USD / hour}$
= $0.14536 \text{ USD / hour}$ (2024/3/13時点)

CloudTech

反対に、AWS Fargateを利用するデメリットもおさえておきましょう。

1つ目は「**ホスト管理に直接介入できない**」という点です。
AWS Fargateを利用することで、コンテナ実行環境のホストマシンなどの管理が必要なくなるというメリットがありましたね。
一方、運用によってはそれがデメリットになることもあります。
OSの設定やチューニングを手動で行うことができないため、細かい設定を行いたい人にとってはデメリットになります。

例えば、先ほど言ったように、高い I/O性能が必要なケースです。
従来のEC2インスタンスでは、EBSボリュームのタイプを選択したり、ネットワークの設定を最適化することで I/O性能を向上させることが可能です。しかし、Fargateではこれらのホストレベルの細かいチューニングができません。

2つ目は「**コンテナの起動が遅い**」という点です。

Fargateで起動するコンテナは awsvpc というネットワークモードのため、起動のたびに ENI のアタッチが必要で時間がかかります。
また EC2 と異なりイメージキャッシュ(コンテナイメージは容量が大きくなりがちのため、イメージを更新する場合はキャッシュを使うことがありますが)これができないため、コンテナイメージの取得に時間がかかります。

2022年4月に改善されたものの、EC2より起動が遅いというデメリットは残り続けています。

3つ目は「**EC2に比べ、ややコストが高い**」という点です。

ここで EC2 と Fargate の費用を比較してみましょう。

まずは EC2 です。 m5.large というタイプのオンデマンドインスタンスを東京リージョンで使った時の費用は 2024年3月時点で 0.124 USD / hour となります。

次に Fargate です。 同じ vCPU・Mem を利用した時の費用 はこのように計算され、同時点で 0.14536 USD / hour となります。

そのため、24時間365日フルで稼働させるとすると、1年間で**約\$190**ほどコスト差が生じます。

これを高いと感じるか安いと感じるかは個人によりますが、先ほどお伝えしたように Fargate は運用コストを削減できることがかなり大きなメリットです。

トータルのコストを考えると、そこまでデメリットに感じない人も多いかと思います。

AWS Fargateのまとめ

-  > ECSで利用可能な**サーバレス**コンテナ実行環境
-  > ECSの基盤となる**インスタンスの管理をオフロード**できる
-  > 管理者は**インスタンスの運用**から解放される



では、最後にAWS Fargateについてももう一度おさらいしましょう。
AWS Fargateは

- **ECSで利用可能なサーバレスコンテナ実行環境**
- **ECSの基盤となるインスタンスの管理をオフロード(手放す)**
- **管理者はインスタンスの運用から解放される**

AWS Fargateを利用することで、サーバ管理から解放され、アプリケーションの開発・構築に集中することが出来ます。
ECSを利用する上で、欠かせないサービスとなるので、この後のハンズオンもぜひ実施してみましょう！

【公式ドキュメント】

https://docs.aws.amazon.com/ja_jp/AmazonECS/latest/developerguide/AWS_Fargate.html