このコンテンツは公開から3年以上経過しており内容が古い可能性があります
最新情報についてはサービス別資料もしくはサービスのドキュメントをご確認ください

[AWS Black Belt Online Seminar] AWS Fargate

サービスカットシリーズ

ソリューション アーキテクト
川﨑 一青

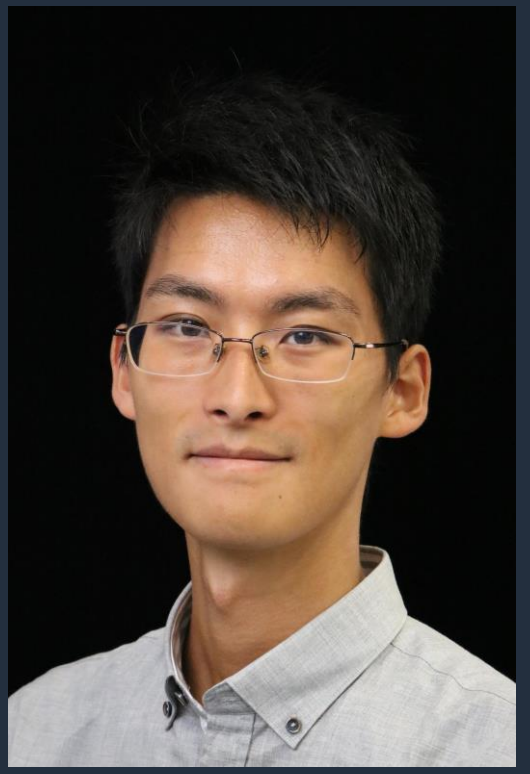
2019/9/25

AWS 公式 Webinar
https://amzn.to/JPWebinar

過去資料
https://amzn.to/JPArchive

自己紹介

川崎 一青 （かわさき いっせい）



所属：

アマゾン ウェブ サービス ジャパン
通信・公益ソリューション部
ソリューションアーキテクト

好きなAWSサービス

AWS Lambda, AWS Fargate

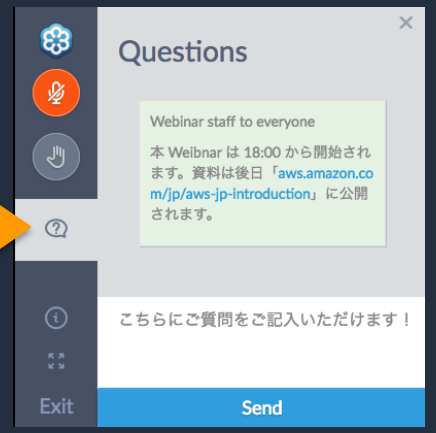
AWS Black Belt Online Seminar とは

「サービス別」「ソリューション別」「業種別」のそれぞれのテーマに分かれて、アマゾン ウェブ サービス ジャパン株式会社が主催するオンラインセミナーシリーズです。

質問を投げることができます！

- 書き込んだ質問は、主催者にしか見えません
- いただいたQ&Aをピックアップしてblogにご紹介させていただく場合がございます
- 今後のロードマップに関するご質問はお答えできませんのでご了承下さい

① 吹き出しをクリック
② 質問を入力
③ Sendをクリック



Twitter ハッシュタグは以下をご利用ください
#awsblackbelt

内容についての注意点

- 本資料では2019年9月25日時点のサービス内容および価格についてご説明しています。最新の情報はAWS公式ウェブサイト(http://aws.amazon.com)にてご確認ください。

- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS公式ウェブサイトの価格を優先とさせていただきます。

- 価格は税抜表記となっています。日本居住者のお客様が東京リージョンを使用する場合、別途消費税をご請求させていただきます。

- AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at http://aws.amazon.com/agreement/. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

アジェンダ

- AWS Fargate 概要
- AWS Fargate の基本
- AWS Fargate を利用したコンテナのデプロイ
- ベストプラクティス、Tips

アジェンダ

- AWS Fargate 概要
- AWS Fargate の基本
- AWS Fargate を利用したコンテナのデプロイ
- ベストプラクティス、Tips

AWS Fargate

サーバー管理なしのコンテナ実行コンピューティングエンジン

AWS マネージド

EC2 インスタンスのプロビジョン、スケール、管理不要

コンテナネイティブ

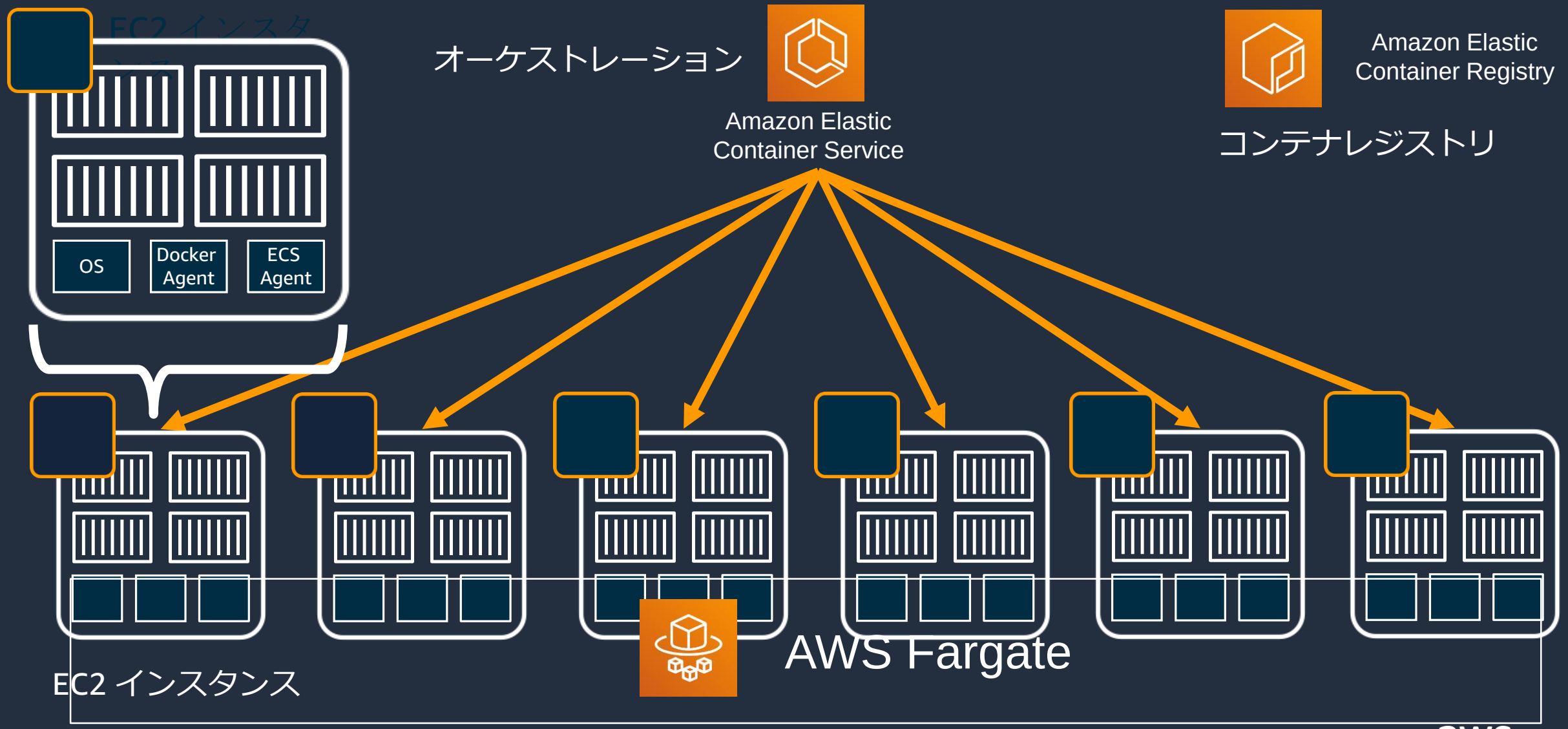
仮想マシンを意識しないシームレスなスケーリング
コンテナの起動時間・使用リソースに応じた料金設定

AWS サービスとの連携

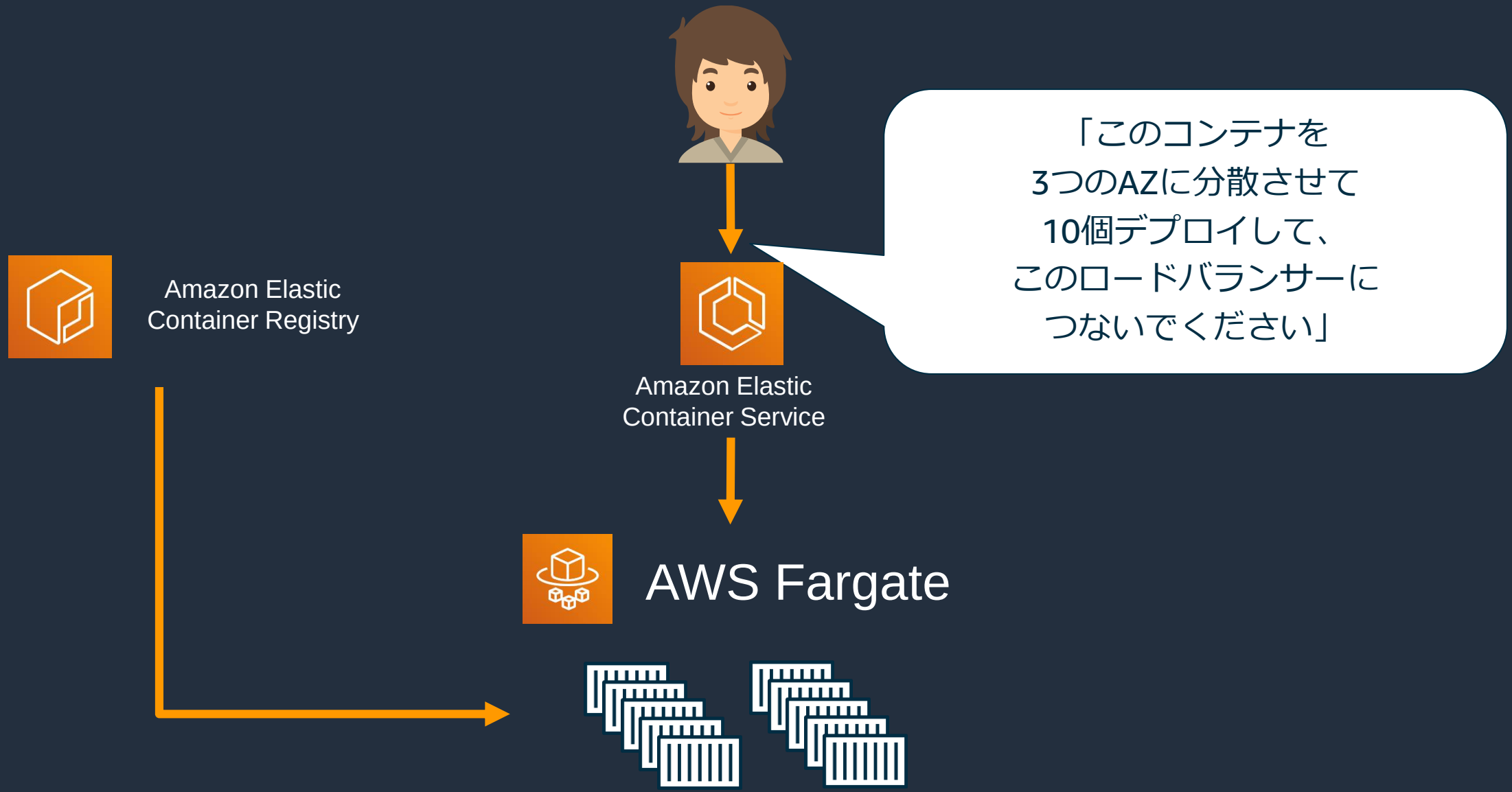
VPC ネットワーキング、Elastic Load Balancing、IAM、CloudWatch、etc.

* 現時点では Amazon ECS のみに対応、Amazon EKS へは将来的に対応の予定あり

Fargate を利用したコンテナ実行のイメージ



Fargate 環境のユーザー利用イメージ



Fargate のメリット

Fargate 利用により以下全てを AWS 側にアウトソース可能

- EC2 インスタンスのプロビジョニングや管理
  - 脆弱性対応のためのパッチ当てや OS アップグレード
  - EC2 インスタンス上で動くエージェント類のアップグレード
  - クラスタ内の EC2 インスタンス群それぞれの上で動くエージェント類やソフトウェアバージョンの整合性維持
  - などなど…
- 状態異常が発生した EC2 インスタンスの再起動や入れ替え
- ホストレベルのスケーリング管理

aws

AWS Fargate customers

“インフラの管理をせずにスケールでき、かつネットワークのきめ細やかな制御ができることが私たちにとって必要であり、それこそが Fargate に移行した理由です。”

Product Hunt

”クラスタの子守はしたくない。私たちにとってはクラスタ管理から生まれる価値は何もない。”

Shimon Tolts
CTO, DATREE

アジェンダ

- AWS Fargate 概要
- AWS Fargate の基本
    - コンピュート（CPU/メモリ）
    - ストレージ
    - ネットワーク
    - IAM連携
- AWS Fargate を利用したコンテナのデプロイ
- ベストプラクティス、Tips

タスクに割り当てるCPUとメモリの設定

柔軟な設定の選択肢
– **50** のCPU/メモリ設定から

1 vCPU = $0.05056/h
1 GB Mem = $0.00553/h

（2019/9/25現在, 東京リージョン）

CPU	Memory
256 (.25 vCPU)	512MB, 1GB, 2GB
512 (.5 vCPU)	1GB to 4GB (1GB 刻み)
1024 (1 vCPU)	2GB to 8GB (1GB 刻み)
2048 (2 vCPU)	4GB to 16GB (1GB 刻み)
4096 (4 vCPU)	8GB to 30GB (1GB 刻み)

＊ 2019年1月7日よりvCPU 20%, GB Memory 65%の値下げを実施

タスクへの CPU、メモリ割り当て（Fargate）

タスクレベル（必須）

- タスクに使用されるCPUとメモリの合計値
- CPU、メモリ共にハード制限
- Fargate では前ページの組み合わせから選択

コンテナレベル（オプション）

- cpu: コンテナ用に予約する cpu ユニット数
- memory: コンテナに適用されるメモリ量（MiB）のハード制限、これを超えようとすると強制終了
- memoryReservation: コンテナ用に予約するメモリのソフト制限（MiB）

Fargate のコストについて

Fargate の方が利用状況に対して最適化されることが多い

- EC2 のオンデマンド料金と比較して約1.17倍
  - EC2 (m5.large) : $0.124/h
  - Fargate (2vCPU, 8GB) : $0.14536/h　　※ 東京リージョンで試算

- EC2 利用時に、全リソースをアプリケーションに割り当てることは現実的に難しい
  - モニタリングツールやコンテナの配置への要件
- Fargate ではより細かい単位でスケール可能
  - 最小で 0.25vCPU, 0.5GB でタスクを実行可能

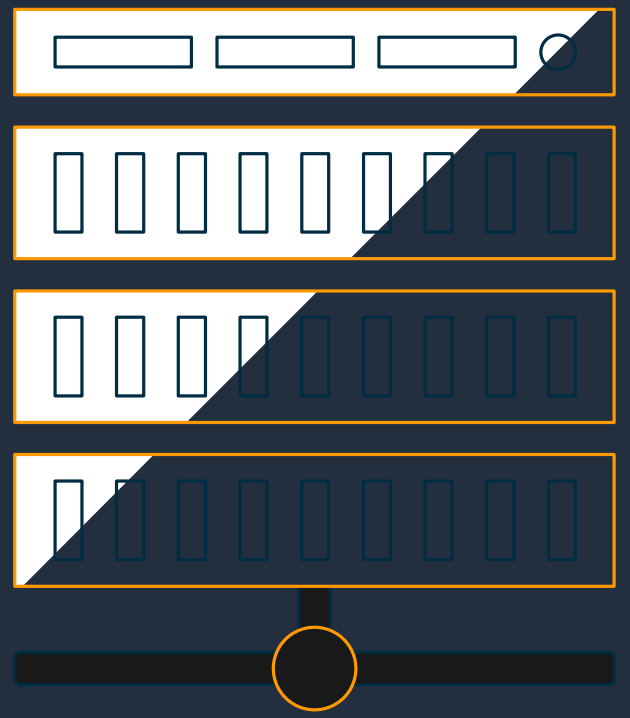
aws

ストレージ

下記 2 種の非永続化ストレージを提供

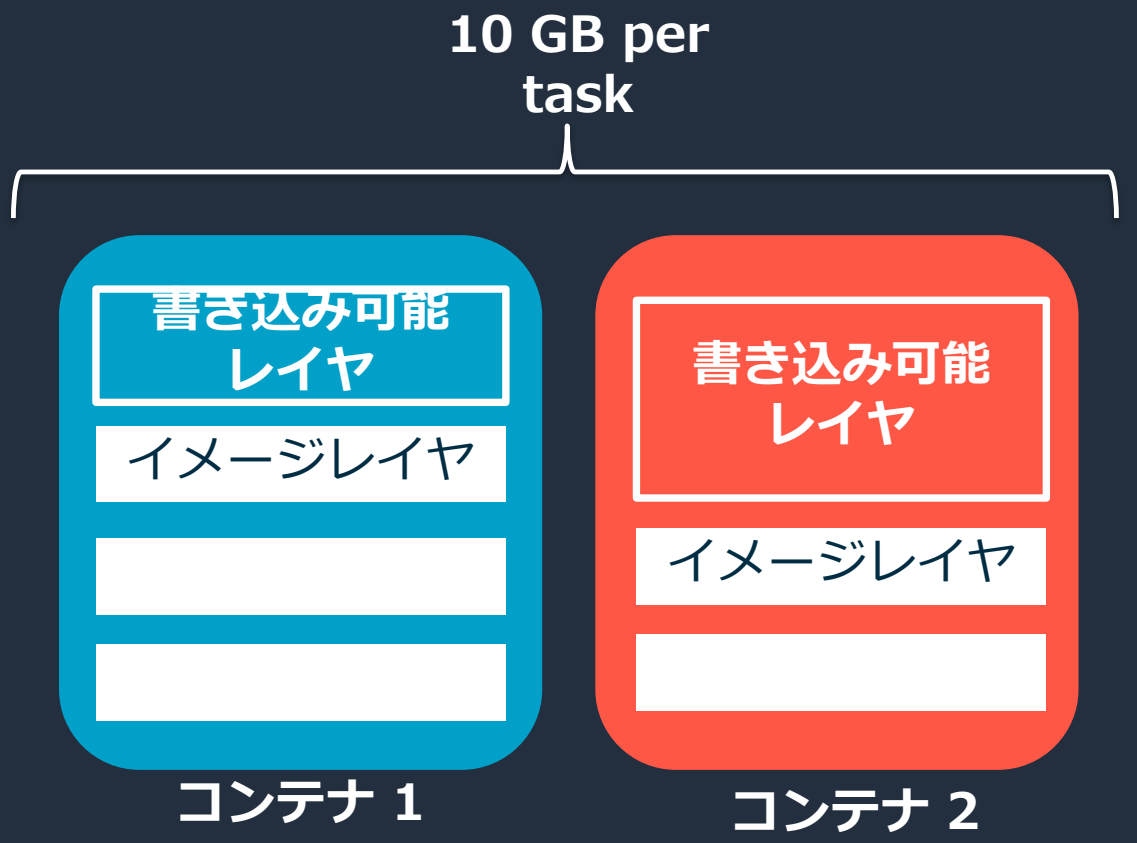
- 書き込み可能レイヤストレージ – 10 GB

- ボリュームストレージ – 4 GB



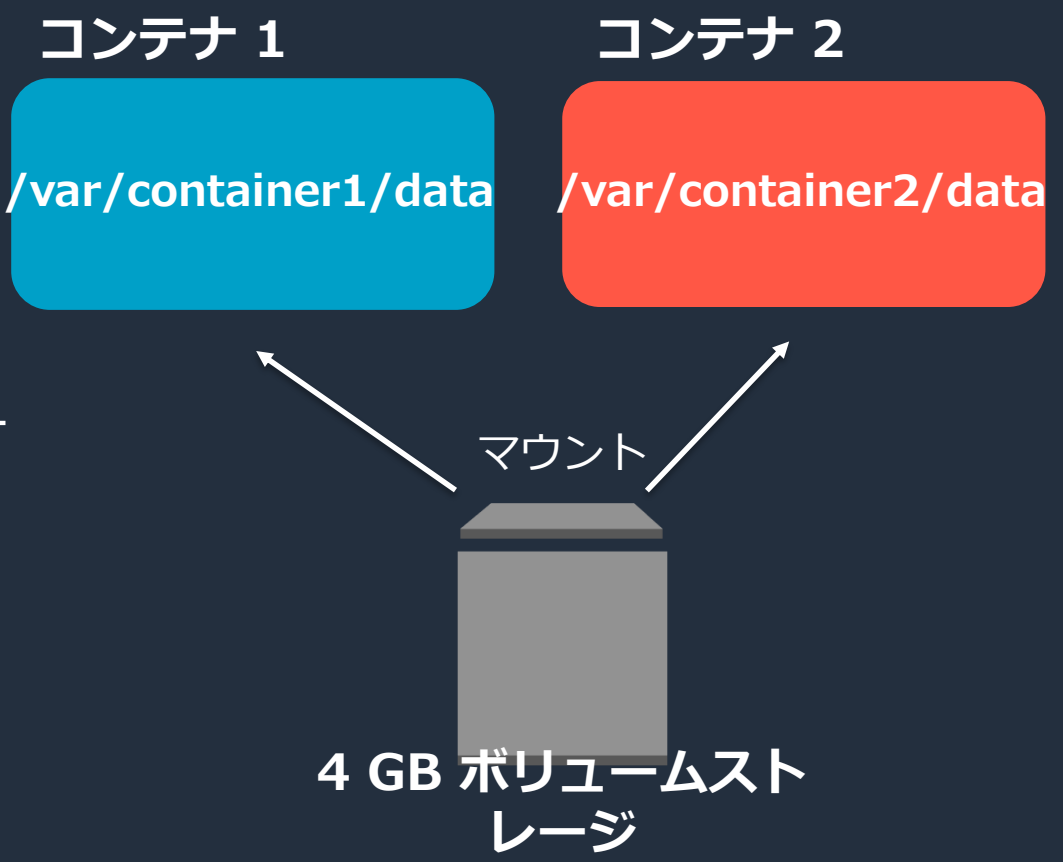
レイヤストレージ

- Docker イメージはレイヤで構成されており、最上位のレイヤは書き込み可能で、実行中のコンテナのファイル変更をキャプチャ
- タスクごとに10 GB のレイヤストレージを利用可能（容量にはイメージレイヤを含み、タスク内の複数のコンテナの合計値）
- 書き込み内容はコンテナ間で互いに独立
- 揮発性を持ち、タスク停止後には利用不可



ボリュームストレージ

- タスクごとに4GBのボリュームを利用可能
- タスク定義内でボリュームマウントとして構成
  - 複数のコンテナ間での共有
  - 異なる containerPath でのマウントも可
- 揮発性があり、タスク停止後は利用不可



ネットワークモード

タスクのコンテナで使用する Docker ネットワーキングモード

- **none:** タスクのコンテナの外部接続なし
- **bridge:** タスクは Docker の組み込み仮想ネットワークを使用
- **host:** EC2 インスタンスのネットワークインターフェイスにコンテナポートを直接マッピング
- **awsvpc:** タスクごとに Elastic Network Interface が割り当てられる

EC2 起動タイプでの選択肢

Fargate 起動タイプを使用する場合は
awsvpc ネットワークモードを利用

awsvpc ネットワークモード

Task 毎に ENI を自動割り当て

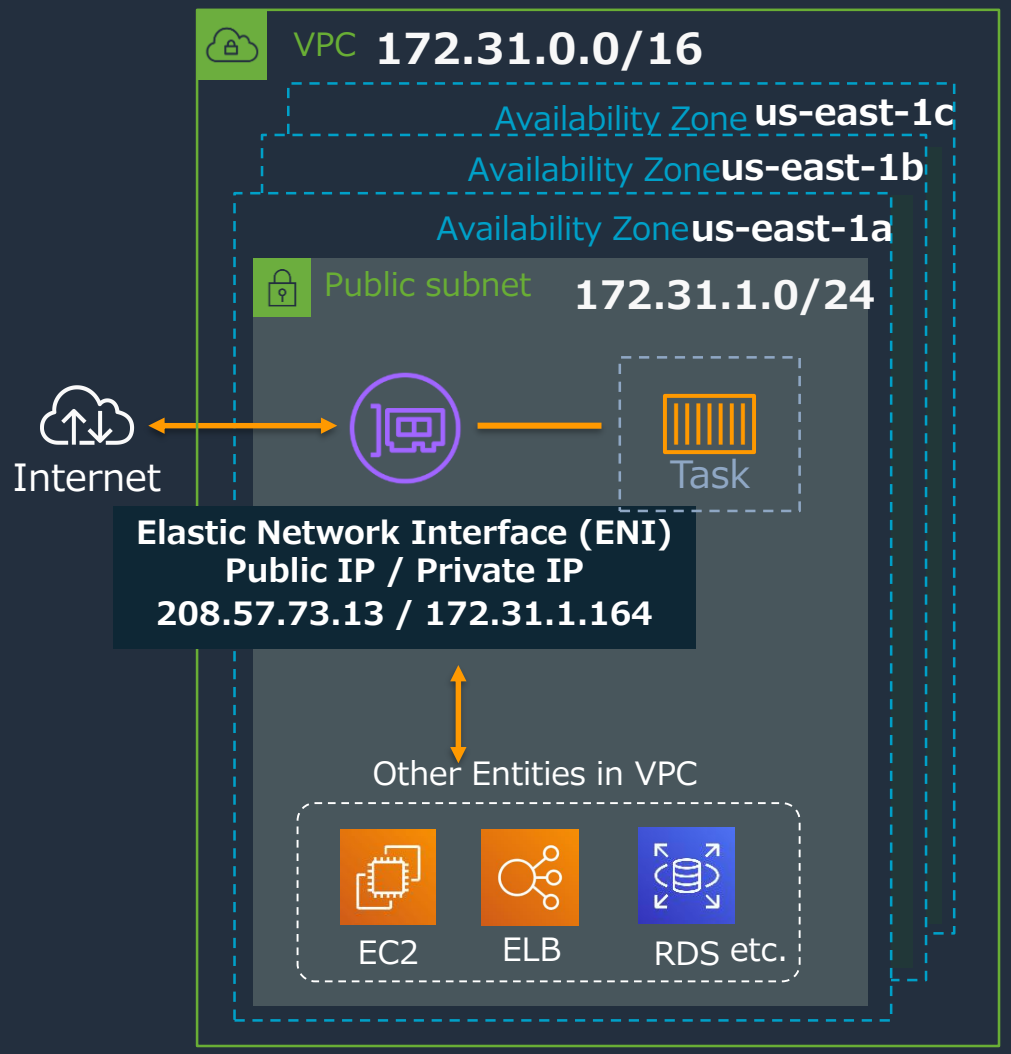
- Security Group を Task 毎に設定可
- VPC Flow Logs でのモニタリング

Task 内のコンテナは localhost インターフェースを共有

- Link 不要で互いにアクセス可能

VPC 内の他のリソースへ Private IP で通信が可能

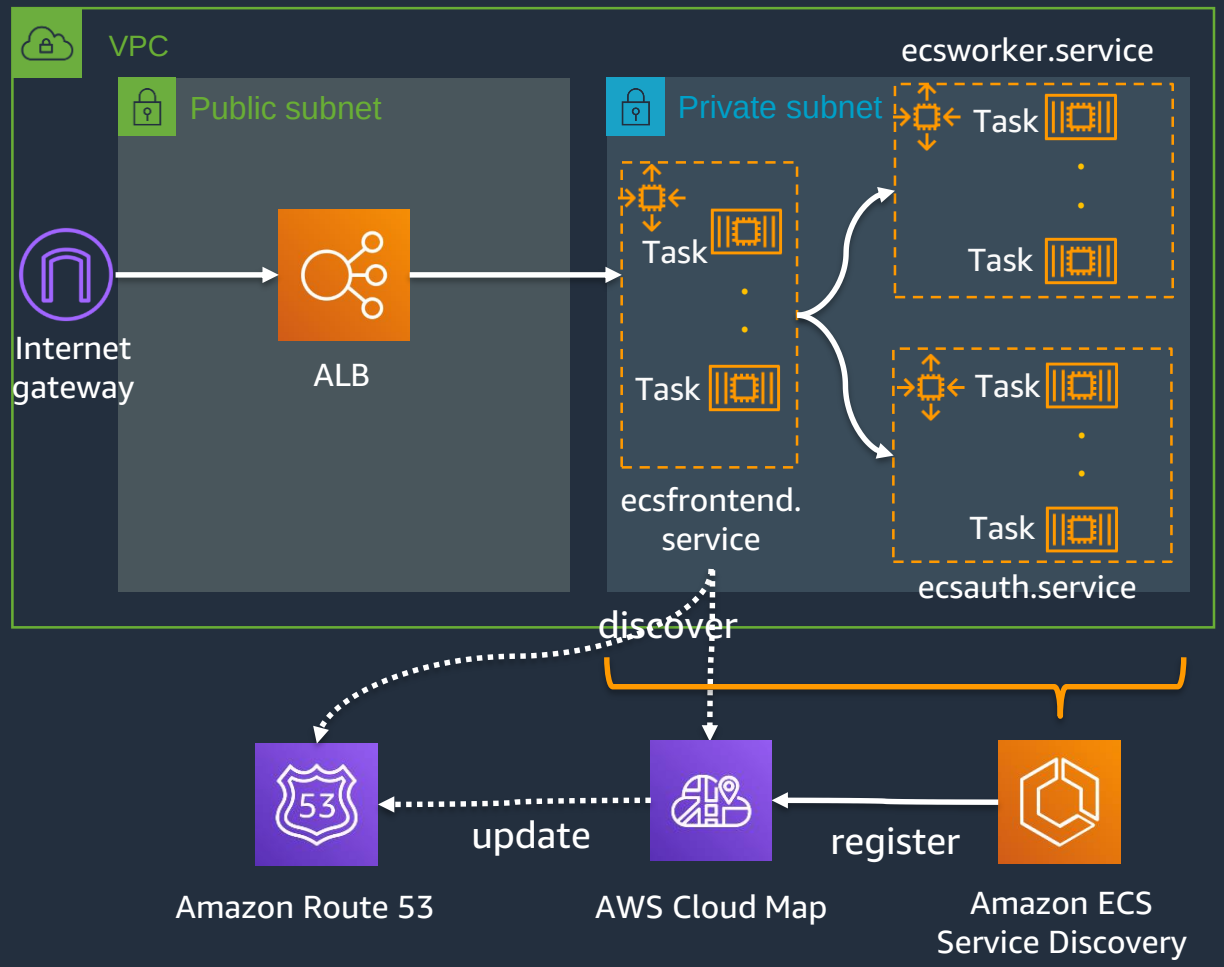
- Fargate では Public IPの割当も可能



インバウンドアクセス

ディスカバリ方式に応じた 2 種の
インバウンドアクセス

- LB ベースのディスカバリ
  → LB経由のアクセス
- DNS ベースのディスカバリ
  → タスク間のアクセス



ロードバランサー

awsvpc ネットワークモードでサポートされる Elastic Load Balancing の種別

Application Load Balancer 

Network Load Balancer 

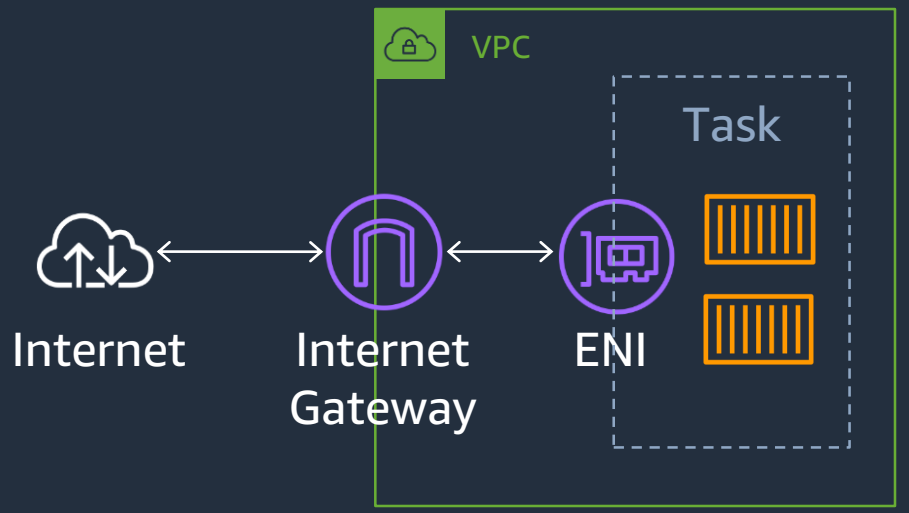
Task 定義でネットワークモードが awsvpc の場合、
ALB/NLB のターゲットグループの target type は ip とする

aws

VPC 外へのアクセス

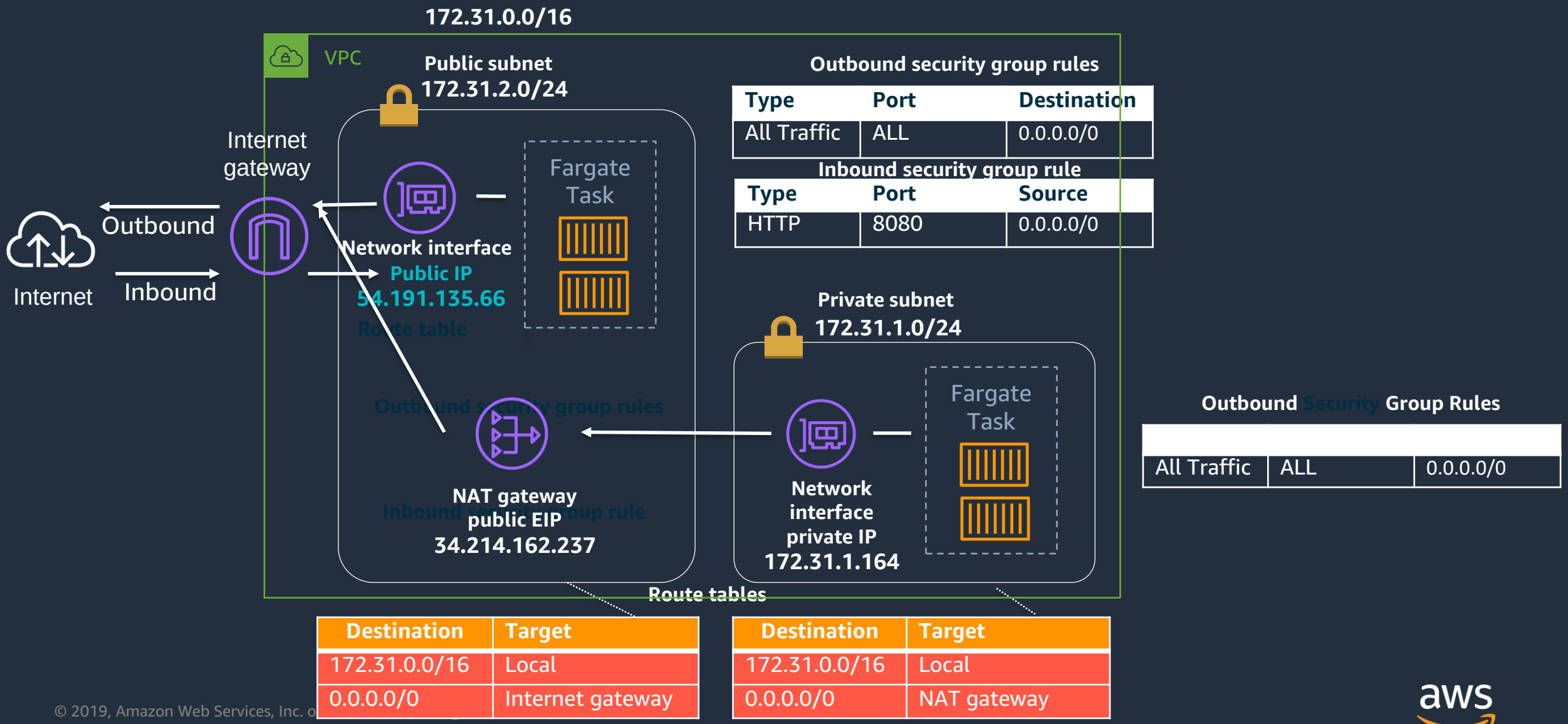
下記のエンドポイントへは到達可能であること
- イメージのプル
- ログ送信

→ 必要に応じ、インターネットゲートウェイ
　または VPC エンドポイント経由でアクセス



インターネット宛の2つのアクセスパターン
- パブリックタスク：
  - インターネットと双方向に通信
- プライベートタスク：
  - インバウンドのインターネットトラフィックなし
  - アウトバウンドのインターネットアクセスはあり

インターネットへのアクセス



Outbound security group rules

Type	Port	Destination
All Traffic	ALL	0.0.0.0/0

Inbound security group rule

Type	Port	Source
HTTP	8080	0.0.0.0/0

Destination	Target
172.31.0.0/16	Local
0.0.0.0/0	Internet gateway

Destination	Target
172.31.0.0/16	Local
0.0.0.0/0	NAT gateway

Outbound Security Group Rules

All Traffic	ALL	0.0.0.0/0

VPC エンドポイントによるプライベートなアクセス

Fargate は インターフェース VPC エンドポイント（PrivateLink）をサポート

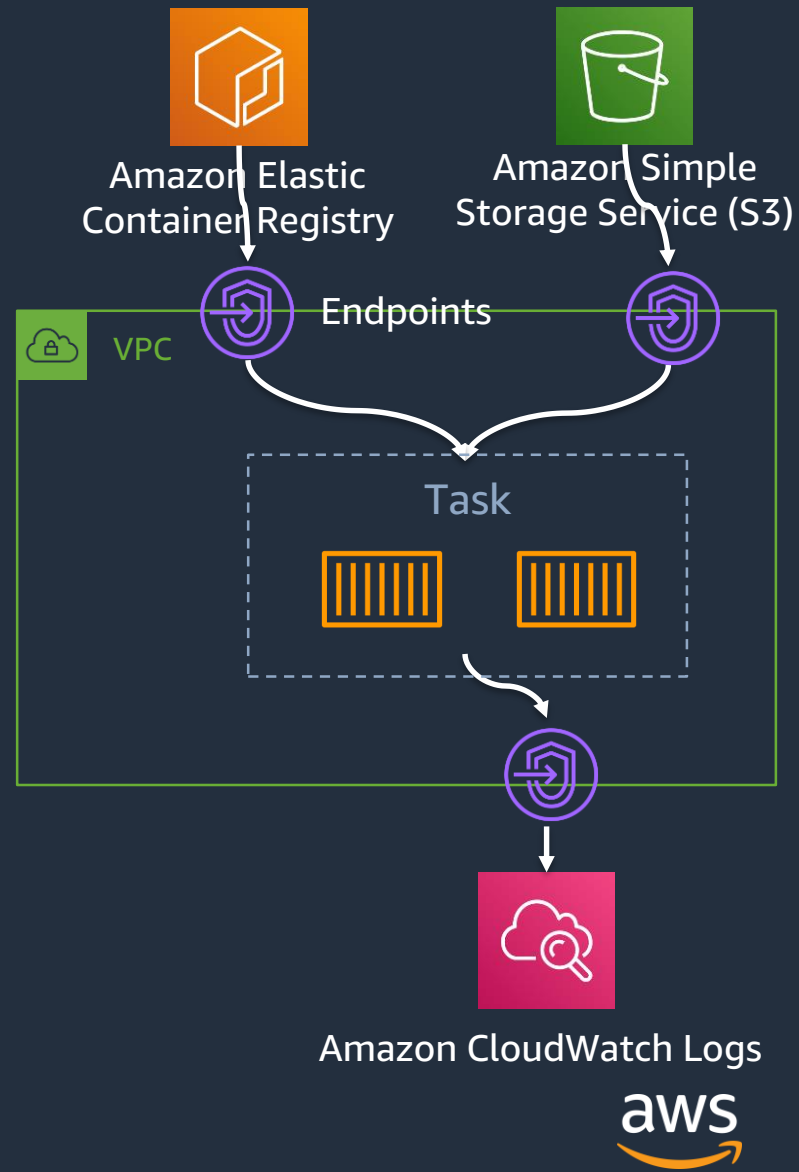
- Fargate タスクでは ECS 用のエンドポイントは不要

ECR からプライベートイメージをプル

- ECR インターフェイス VPC エンドポイント
- S3 ゲートウェイエンドポイント

awslogs ログドライバーを使用してタスクからログ情報を CloudWatch Logs に送信

- CloudWatch Logs インターフェイス VPC エンドポイント



レジストリサポート

以下いずれのオプションも利用可能

Amazon Elastic Container Registry
(Amazon ECR)

パブリックレジストリ

サードパーティー プライベートレジストリ*

* プライベートレジストリの利用にはプラットフォームバージョン 1.2.0 以上が必要
認証には AWS Secrets Manager を使用

IAM 連携

クラスタパーミッション

誰がクラスタ内でタスクを起動/参照できるのかを制御

アプリケーション: タスクロール

アプリケーションのコンテナがAWSリソースに安全にアクセスすることを許可する

ハウスキーピング: タスク周りの下働きの実行を許可

タスク実行ロール

- プライベートレジストリのイメージを取得
- CloudWatch Logs への書き込み

ECS のサービスにリンクされたロール

- Elastic network interfaceの作成
- ELBへのターゲットの登録/解除

タスク実行ロール

- Amazon ECS コンテナエージェントがユーザーに代わって Amazon ECS API アクションを呼び出す
- Fargate 起動タイプでは以下の目的に必要なポリシーをアタッチ
  - コンテナイメージをプルするための Amazon ECR の呼び出し
  - コンテナアプリケーションログを保存するための CloudWatch の呼び出し

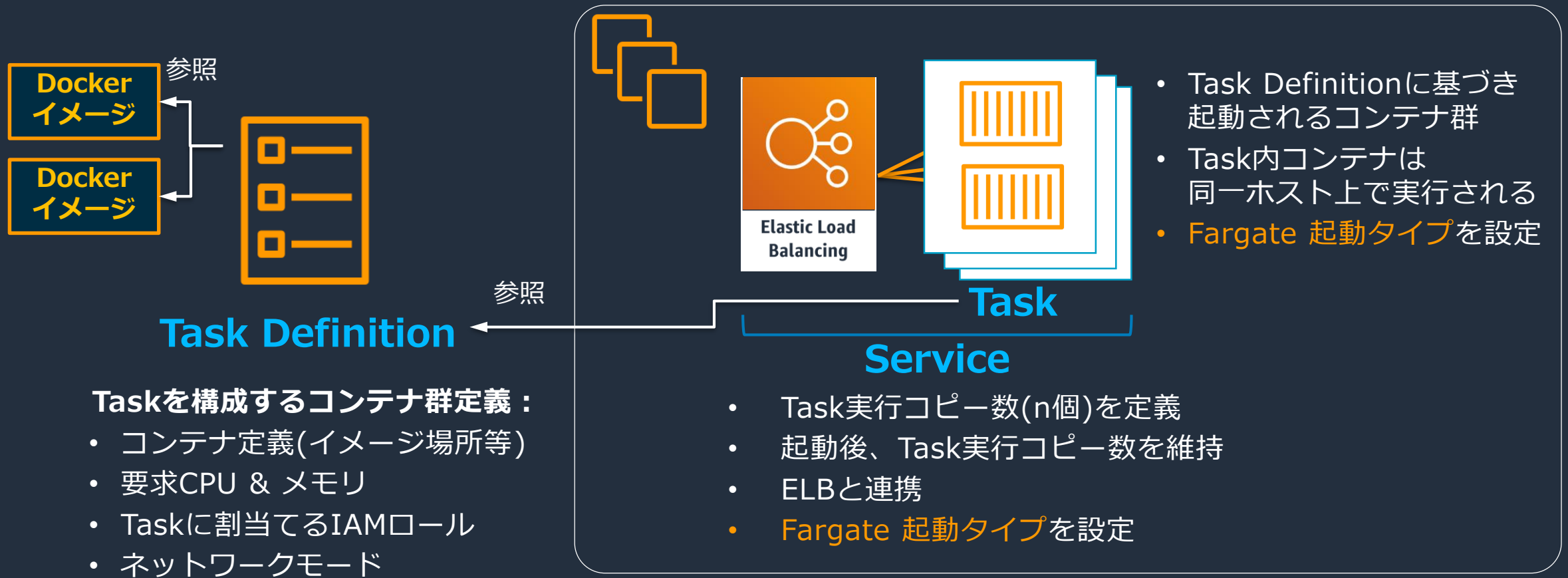
  * 管理ポリシー AmazonECSTaskExecutionRolePolicy を利用可能

- プライベートレジストリ認証機能を使用する場合は、インラインポリシーとして認証情報を保存した AWS Secrets Manager へのアクセス許可を追加

アジェンダ

- AWS Fargate 概要
- AWS Fargate の基本
- AWS Fargate を利用したコンテナのデプロイ
- ベストプラクティス、Tips

Fargate を利用したコンテナのデプロイ



- Task Definitionに基づき起動されるコンテナ群
- Task内コンテナは同一ホスト上で実行される
- Fargate 起動タイプを設定

Taskを構成するコンテナ群定義：

- コンテナ定義(イメージ場所等)
- 要求CPU & メモリ
- Taskに割当てるIAMロール
- ネットワークモード
- etc...
- **Fargate 起動タイプと互換性を持たせる**

- Task実行コピー数(n個)を定義
- 起動後、Task実行コピー数を維持
- ELBと連携
- Fargate 起動タイプを設定

- 実行環境の境界
- IAM権限の境界(クラスターに対する操作)
- スケジュールされたタスクの実行を設定可能

タスク定義を作成 - Fargate と互換性を持たせる

Fargate 起動タイプと互換性を持たせる

```
{
    "containerDefinitions": [
        <略>
    ],
    "cpu": "256",
    "executionRoleArn": "arn:aws:iam::012345678910:role/ecsTaskExecutionRole",
    "family": "fargate-task-definition",
    "memory": "512",
    "networkMode": "awsvpc",
    "requiresCompatibilities": [
        "FARGATE"
    ]
}
```

必須のパラメータではありませんが、
タスク定義で使用されているすべてのパラメータが、
起動タイプの要件を満たしていることを確認できます。

クラスタを作成 - EC2インスタンスは不要

EC2 インスタンス（Auto Scaling グループ）の起動は不要

サービスを作成 - Fargate 起動タイプを選択

作成されたサービス/タスクの確認

AWS CLI を使用した Fargate タスクを実行するサービス作成

```
aws ecs create-service --cluster fargate-cluster --service-name fargate-service --task-definition sample-fargate:1 --desired-count 2 --launch-type "FARGATE" --network-configuration "awsvpcConfiguration={subnets=[subnet-abcd1234],securityGroups=[sg-abcd1234]}"
```

```
{
    "service": {
        "status": "ACTIVE",
        "taskDefinition": "arn:aws:ecs:region:aws_account_id:task-definition/sample-fargate:1",
        "pendingCount": 0,
        "launchType": "FARGATE",
            :
        <略>
            :
    }
}
```

アジェンダ

- AWS Fargate 概要
- AWS Fargate の基本
- AWS Fargate を利用したコンテナのデプロイ
- ベストプラクティス、Tips

EC2 起動タイプとの違い - タスク定義パラメータ

次のタスク定義パラメータは Fargate 起動タイプではサポートされません。

- disableNetworking
- dnsSearchDomains
- dnsServers
- dockerSecurityOptions
- extraHosts
- gpu
- ipcMode
- links
- pidMode
- placementConstraints
- privileged
- systemControls

EC2 起動タイプとの違い - 用途の比較

下記のような用途では、EC2起動タイプが適する

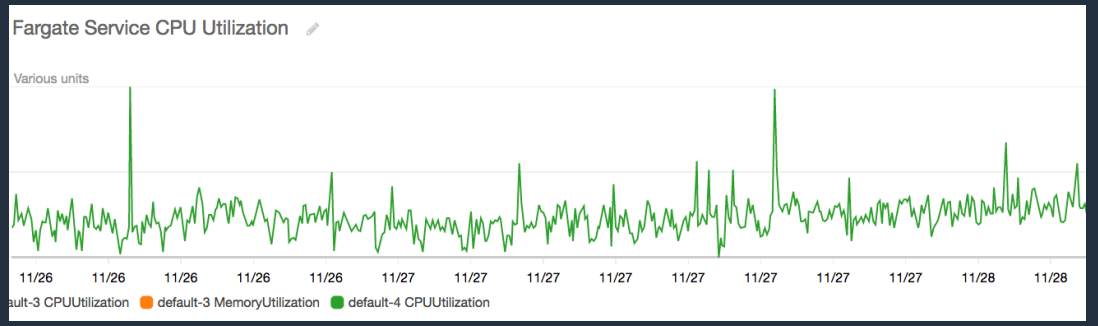
- docker exec のようなインタラクティブなデバッグ
- GPUサポート
- Windows コンテナ
- Spot や RI ベースの価格モデルの適用

上記の要件がない場合、または再考可能な場合は、
Fargate 利用のメリット享受を検討

可視化とモニタリング

CloudWatch Logs
CloudWatch Events サポート

サービスレベルメトリックス



ログ記録

Fargate タスク定義は awslogs および splunk ログドライバーをサポート

- [Preview] Fluentd または Fluent Bit をログルーターとする構成をタスク定義から設定（FireLens） *NEW!!*

awslogs ログドライバー

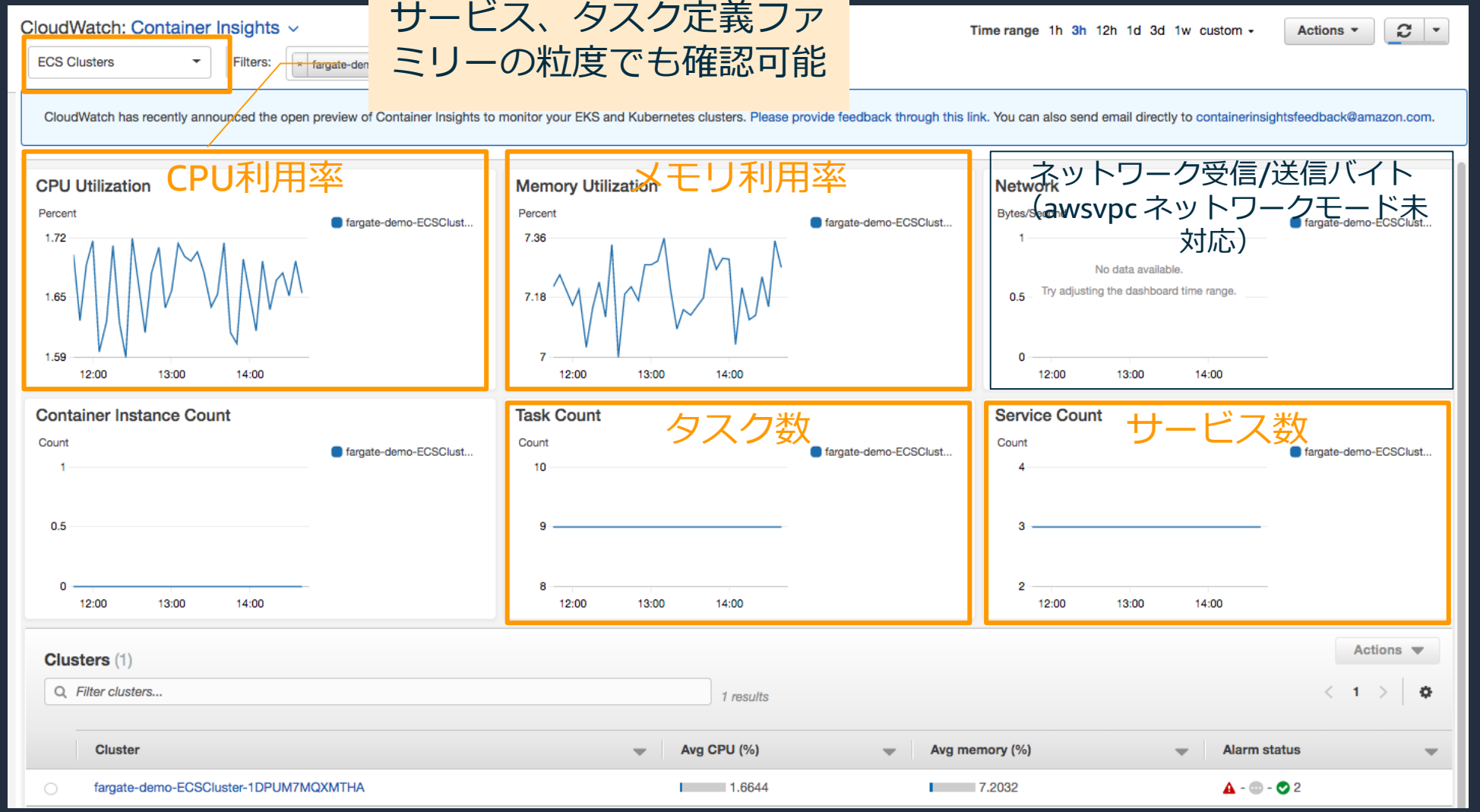
タスクのコンテナを設定して Dockerから CloudWatch Logs にログ情報を送信

- デフォルトでは、キャプチャされるログは、コンテナをローカルに実行した場合にインタラクティブターミナルに表示されるコマンド出力 (STDOUT および STDERR I/O ストリーム)

Splunk ログドライバー

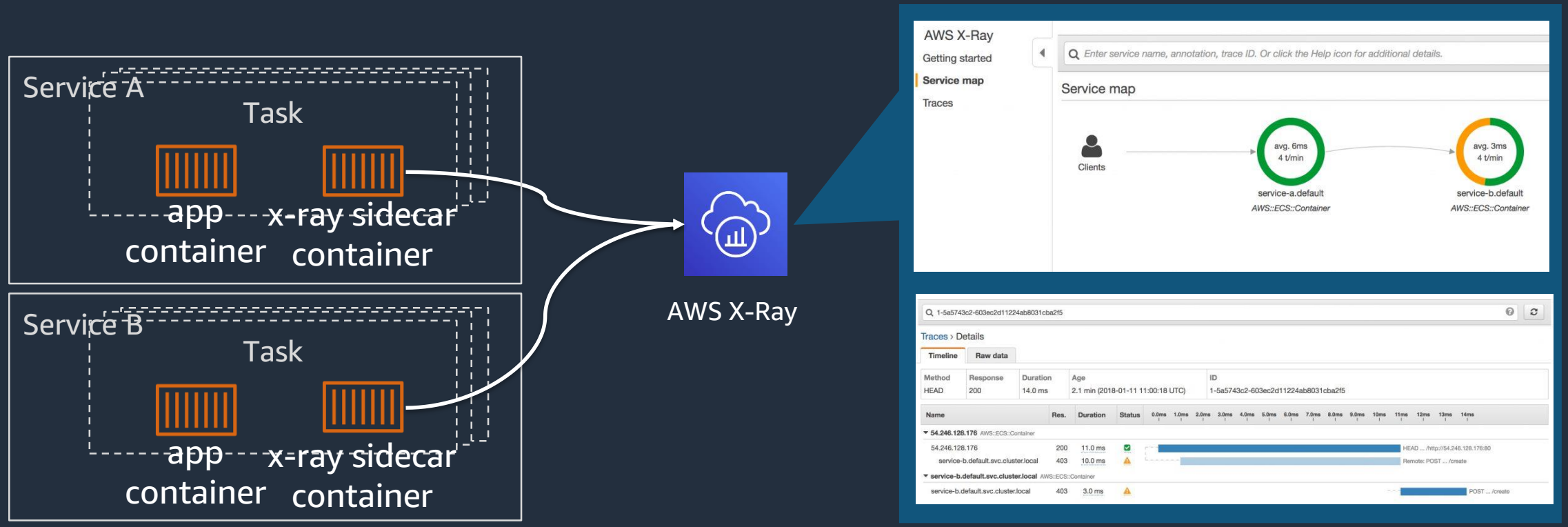
Splunk URL、Splunk トークンパラメータを指定してログ情報を送信

Amazon CloudWatch Container Insights



サイドカーコンテナを利用したアーキテクチャ

データ収集、送信を行うサイドカーコンテナをタスク内で実行
例）X-Ray によるトレーシング

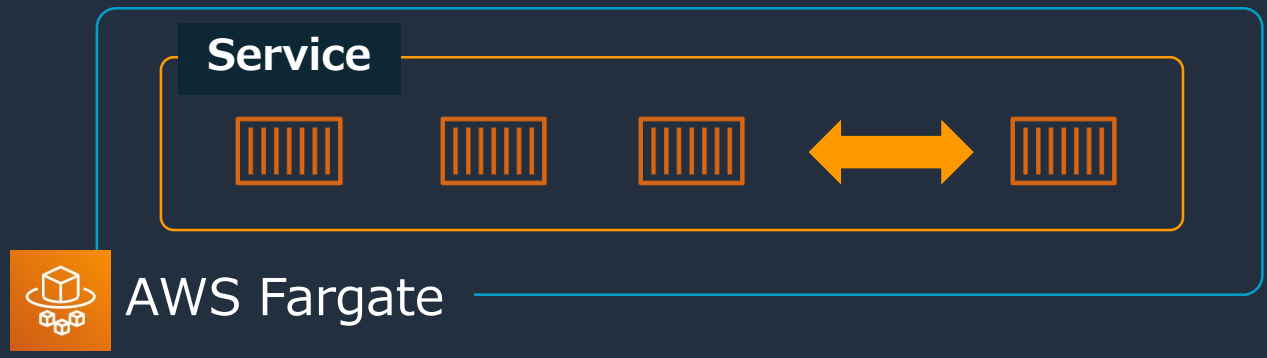


https://github.com/aws-samples/aws-xray-fargate

Fargateを利用したコンテナAuto Scalingの優位性

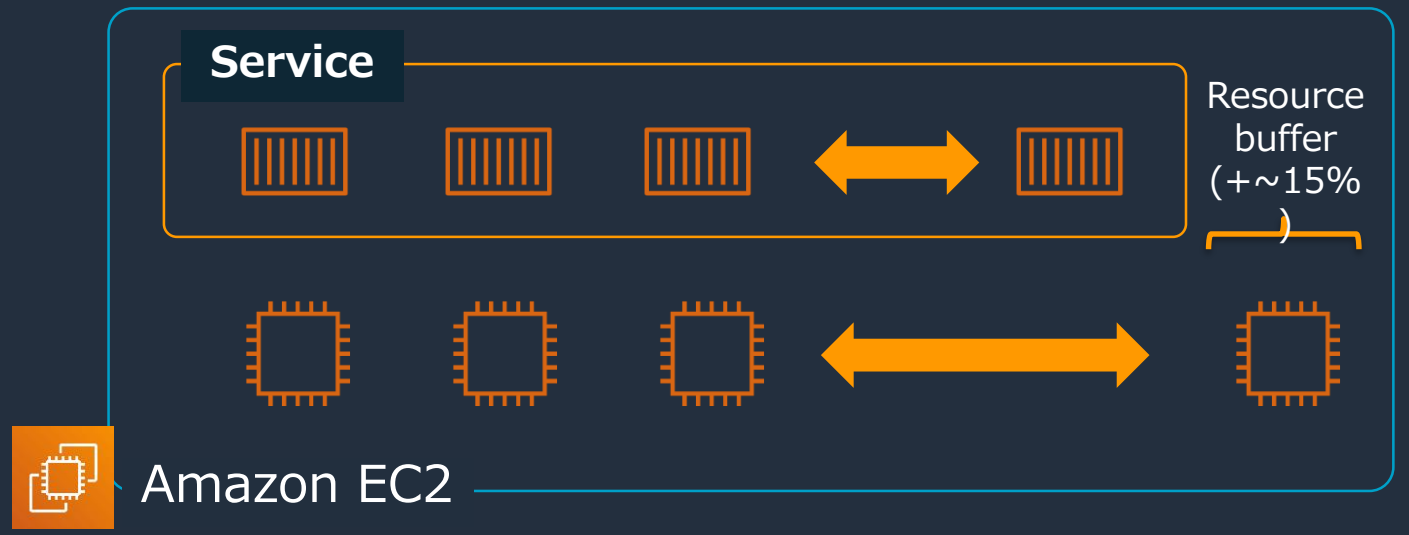
Fargateの場合

- Serviceのスケールに応じて自然にコンテナが起動・終了する
- コンテナの起動時間に対してのみ課金



EC2の場合

- インスタンスのリソースも上手くスケールさせる必要があり煩雑
- 余分に持っているバッファ分もインスタンスの課金が必要



サービスレベルアグリーメント + コンプライアンス

Amazon ECS　AWS Fargate

99.99

＋

HIPAA 資格要件を満たすとともに、ISO、PCI、SOC 1、SOC 2、および SOC 3 コンプライアンスの基準を満たしています

Fargate プラットフォームのバージョン

- ランタイム環境が進化するにつれて新しいプラットフォームバージョンをリリース
    - カーネルやオペレーティングシステムの更新、新機能、バグ修正、セキュリティの更新があったとき等
    - プラットフォームのバージョンに影響を与えるセキュリティ上の問題が見つかった場合、AWS はそのプラットフォームバージョンにパッチを適用
- 基本は最新のプラットフォームバージョンのご利用を推奨
    - より多くの修正が適用されている
    - 多くのコンテナはプラットフォームに依存しないよう作られる
- 特定のバージョン番号を指定して使用することも可能

本セッションの内容振り返り

- AWS Fargate 概要
- AWS Fargate の基本
    - コンピュート（CPU/メモリ）
    - ストレージ
    - ネットワーク
    - IAM連携
- AWS Fargate を利用したコンテナのデプロイ
    - Fargate 起動タイプ
- ベストプラクティス、Tips
    - EC2起動タイプとの違い
    - 監視、ログ記録
    - Auto Scaling、SLA、コンプライアンス

Q&A

ご質問については,
AWS Japan Blog 「https://aws.amazon.com/jp/blogs/news/」
にて資料公開と併せて、後日掲載します

AWS の日本語資料の場所「AWS 資料」で検索

https://amzn.to/JPArchive

AWS Well-Architected 個別技術相談会

毎週”W-A個別技術相談会”を実施中

- AWSのソリューションアーキテクト(SA)に対策などを相談することも可能
- 申込みはイベント告知サイトから

(https://aws.amazon.com/jp/about-aws/events/)

AWS イベント で[検索]

ご視聴ありがとうございました

AWS 公式 Webinar
https://amzn.to/JPWebinar

過去資料
https://amzn.to/JPArchive