

ログイン

2023年7月18日 13:27

<https://d-956703c220.awsapps.com/start#/>

IAMユーザー

608728620263

osuke_oyaizu

d2^KMU8P@s{E\$b

[TOROハンズオン環境\(学習用環境\)利用申請 – 一般ユーザー向け – Jira Service Management \(atlassian.net\)](#)

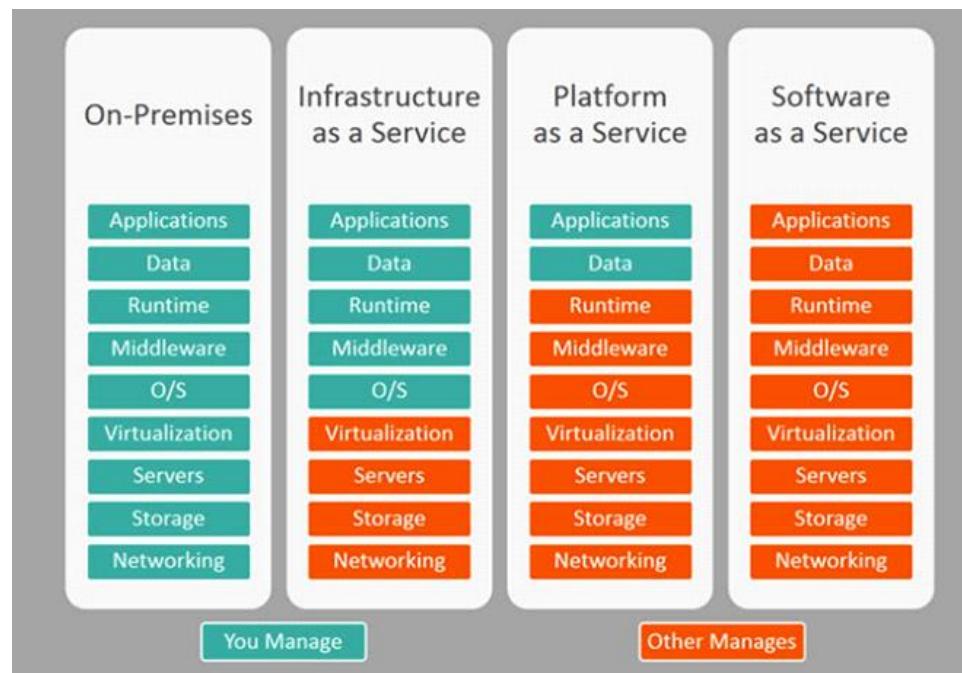
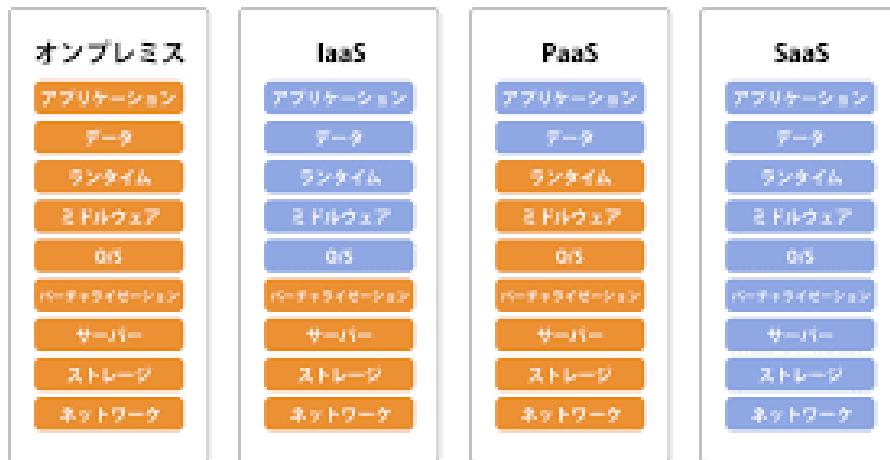
osuke_oyaizu

osuke_oyaizu@mail.toyota.co.jp

Gakuen844#000

iaas paas saas

2023年7月24日 16:04



課題 1

2023年7月31日 10:32



解決策 1：S3をVPC内に設置する

S3のエンドポイントを作成する

①エンドポイント作成

Gatewayだと無料(S3とDynamoDBのみ)

<https://blog.serverworks.co.jp/2022/07/08/122037>

②ルートテーブルの設定で新しく作成したプライベートサブネットを指定する

確認：ec2からs3にアクセス

```
# aws s3 ls s3://wordpress20230720lab2
```

解決策 2：インスタンスがインターネットにアクセスできるようにする

<https://ryonotes.com/difference-between-internet-gateway-and-nat-gateway/>

NAT gatewayを設置する・・・AZ間にまたいで配置する

①S3用のエンドポイントを削除する → 画像を保存アップロードできなくなる

②「VPC」→「NATゲートウェイ」

- Publicサブネットに配置する
- Elastic IP を割り当てる

https://docs.aws.amazon.com/ja_jp/vpc/latest/userguide/vpc-nat-gateway.html

<https://qiita.com/leomaro7/items/52147ee88c6da11048e2>

③ルートテーブル作成

- ルート編集

| ルート | サブネットの経由付け | Edge の経由付け | ルート伝播 | タグ |
|-------------|-----------------------|------------------|------------|----|
| ルート (2) | | | | |
| 送信先 | ターゲット | ステータス | 伝播済み | |
| 0.0.0.0/0 | nat-07c60fd6ac8d7bb5d | ○アクティブ ○アクティブ | いいえ いいえ | |
| 10.0.0.0/16 | local | | | |

- ec2インスタンスが配置してあるプライベートサブネットを割り当てる

プライベートサブネットからインターネットゲートウェイに直接通信することはない

なので一度プライベートサブネットからNAT Gatewayを通してからインターネットゲートウェイへ通信する

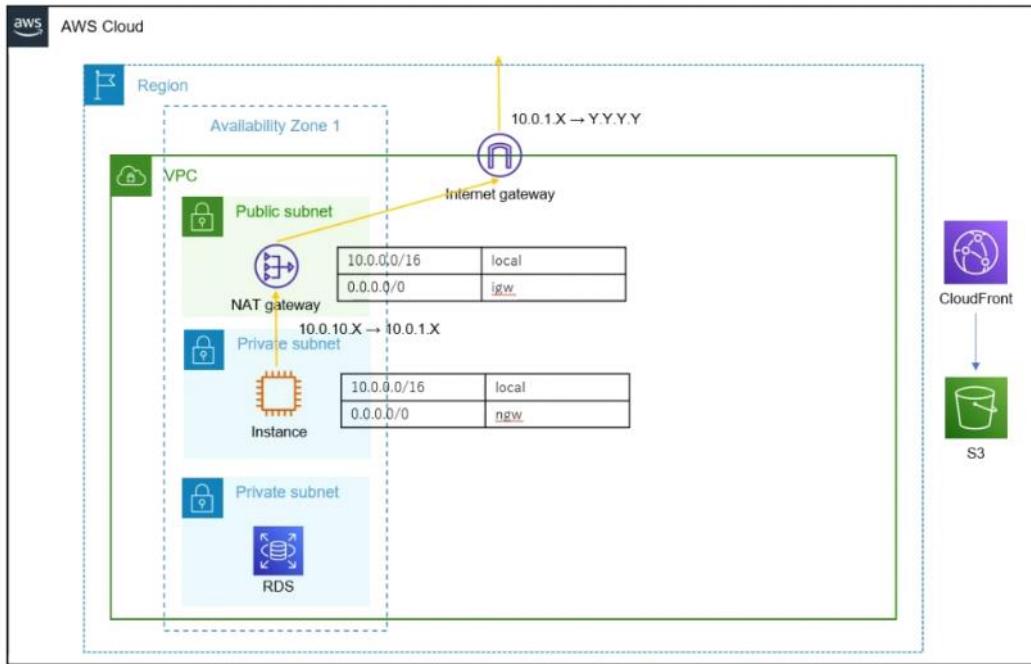
動き

プライベートサブネットからの通信 → NAT Gateway(パブリックサブネットへ) → Internet gateway(ElasticIPへ)
2回変換されている

インターネットゲートウェイとの違い

インターネットゲートウェイ：VPCにアタッチしそのVPC内のインスタンスが、インターネットなどVPC外のネットワークと通信するためのコンポーネントです。

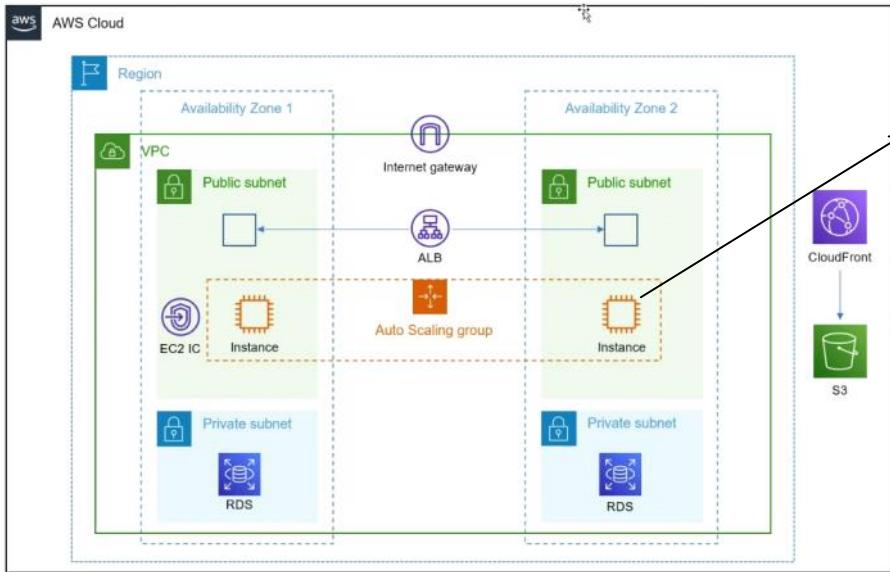
NATゲートウェイ：プライベートサブネット内のインスタンスがVPC外のネットワークと通信するためのコンポーネントです。



エンドポイントだと特定のサービスにのみアクセスできる
NAT Gatewayを使用すればインターネットへ全てアクセスできる

補足

2023年7月25日 16:03



<https://cloudfront.lab2...>

↑
何かわからないのでCloudFrontのドメインを
CNAMEで登録する

443アクセスには証明書が必須
証明書のFQDNと問い合わせのドメインが一致し
たらOK

<https://milestone-of-se.nesuke.com/sv-advanced/digicert/digital-certification-summary/>

クライアントは提示された証明書のサブジェ
クト代替名 (SANs) と、ブラウザに入力した
URL の FQDN (『https://』と『次の/』の間の
文字列) **が一致するかどうかを確認**します。
証明書がないと警告がでてしまう

アプリ一般公開

2023年8月1日 10:20

インスタンス作成

windows

t3.medium

キーペア : keypair01

ネットワーク

サブネットはプライベートサブネットを選択する



セキュリティ：デフォルト

インスタンスタイプもデフォルト 30 2gp

ロール割り当て : LambdaRole

ロールの内容→AmazonEC2RoleforSSM

アクション → セキュリティ → windowsパスワード取得

[keypair01.pem](#) (キー) の内容を書き込む

ユーザー名 : Administrator

パスワードを取得できる : J%(\$oDC1?L3Lak3keWeXj-bGReZbkquP

接続をする



→ユーザー名とパスワードを入力すると接続ができる

コマンドプロンプトからコマンド入力する

[AWSのCLIのインストール方法](#)

msiexec.exe /i <https://awscli.amazonaws.com/AWSSLIV2.msi>

コマンドプロンプトを開き直す

aws --version → バージョンが表示される

servermanager

IISをインストール

Visual Studioの設定

Server 発行 フォルダー

- C:\Users\Administrator\source\repos\Test\DeployTestApp\DeployTestApp\Server\bin\Release\net6.0\publish が生成される
- ZIPにしてs3にアップロードする

s3にアップロードしたServerフォルダをデスクトップにダウンロードする

```
aws s3 cp s3://ahaws-bucket-01/oyaizu.ZIP desktop
```

DotNetインストールファイルをダウンロードする

```
aws s3 cp s3://ahaws-bucket-01/dotnet-hosting-6.0.20-win.exe desktop
```

ブラウザがデフォルトでみにいく場所にコンテンツをコピーする

解凍したものをC:\inetpub\wwwrootにすべてコピーする

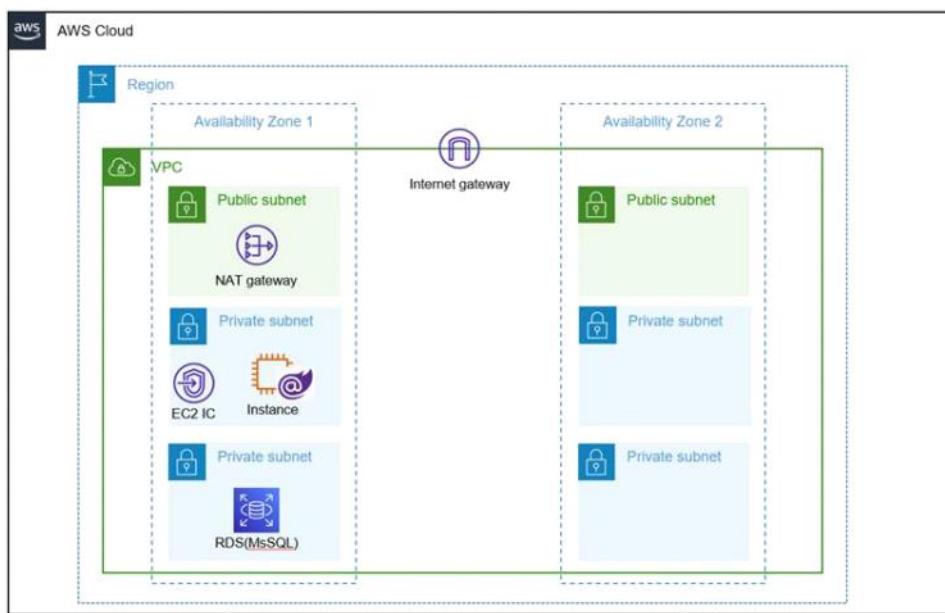
Edgeで自分のIPアドレスを検索するとアプリが実行される

SSMの場合はインスタンスが繋ぎにいくので (OUTPUT通信)

→SSHポートの穴あけが必要ない

データベース接続あり

2023年8月1日 14:24



RDS作成

データベース作成方法：標準作成

エンジンのオプション：Microsoft SQL Server

インスタンスタイプ：db.t3.small

vpc : vpc01

セキュリティ : default

自動スケーリング : OFF

Performance Insights : OFF

自動バックアップ : OFF

EC2インスタンス設定

仮想WindowsでSSMSインストール（データベース作成に便利なため） ※言語設定に注意

Server name:RDSのエンドポイント

ユーザー名 & パスワード・・・RDS作成時に設定したもの

データベース & テーブル作成

アプリ作成

パッケージなどインストール

scaffoldの設定をする

```
appsettings.json (user id,password : RDS作成時に設定したもの server : RDSのエンドポイント)
  "ConnectionStrings": {
    "DB": "user id=admin;password=password;server=oyaizu-kimbara-db.cnkcerumhjjp.ap-northeast-1.rds.amazonaws.com;initial catalog=oyaizu_db;TrustServerCertificate=True"
  }
```

仮想Windowsに適用

Serverを右クリック→発行

フォルダ→発行

Server¥bin¥Release¥net6.0¥publishの中のファイルを圧縮

ZIPファイルをS3バケットにいれる

コマンドプロンプトから自分のデスクトップにダウンロード

例)aws se cp s3://ahaws-bucket-01/oyaizu_db.ZIP desktop¥

展開したものをC:¥inetpub¥wwwroot配下に配置する

Edgeから自分のIPでアクセス→アプリが実行される

Linux(Ubuntu)

2023年8月1日 16:15

インスタンス
Ubuntu
プライベートサブネットに配置する

.NETをUbuntuにインストール ※バージョンに注意する

```
sudo apt-get update && sudo apt-get install -y dotnet-sdk-6.0
sudo apt-get update && sudo apt-get install -y aspnetcore-runtime-6.0
```

```
sudo apt install zip unzip
```

Linuxでawsコマンドを実行できるようにする

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86\_64.zip" -o "awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

```
sudo mkdir /work
```

アプリのフォルダをs3からダウンロード

```
sudo aws s3 cp s3://ahaws-bucket-01/oyaizu_db.ZIP /work
```

解凍コマンド

```
sudo unzip oyaizu_db.ZIP
```

ポートを開ける

```
sudo dotnet oyaizu.Server.dll ... 実行したままにしないといけない
Cntr + Cでシャットダウンできる
sudo ss -anpt → 管理者権限で実行すると5000番のみがあく、それ以外は5000番,5001番があく
```

APIにアクセスする

```
curl -k https://localhost:5001/api/users → データの中を取得できる -k : 証明書を確認を無視する
```

※dotnet oyaizu.Server.dllでエラーが起きるとき dotnetのpidをkillする

```
sudo ss -anpt
LISTEN      0      512          [::]:5000          [::]:*          users:(("dotnet",pid=5719,fd=179))
ubuntu@ip-192-168-102-40:/work$ sudo kill -KILL 5719
```

別のクライアントからIPアドレスで検索するとアプリが実行される

注意

sudo dotnet <ソリューション名>.Server.dllで5000番と5001番が開いてしまう場合は
[Server]

app.UseHttpsRedirection(); をコメントアウトする

公開する

nginxをで公開する方法



nginxはプロキシのような動き

```
sudo apt install nginx
```

```
cd /etc/nginx  
sudo vi sites-enabled/default の中に書き込む
```

<https://learn.microsoft.com/ja-jp/aspnet/core/host-and-deploy/linux-nginx?view=aspnetcore-7.0&tabs=linux-ubuntu>

```
proxy_pass http://127.0.0.1:5000; proxy_http_version 1.1; proxy_set_header Upgrade $http_upgrade;  
proxy_set_header Connection keep-alive; proxy_set_header Host $host; proxy_cache_bypass  
$http_upgrade; proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for; proxy_set_header X-  
Forwarded-Proto $scheme;
```

MySQLにする

2023年8月2日 9:29



Linux(Ubuntu)にmysqlをインストール

```
sudo apt install mysql-server-8.0
```

```
sudo mysql
```

データベース作成

```
create database <データベース名>
```

テーブル作成

```
use <データベース名>
```

```
例)create table <テーブル名> (ID int, Name varchar(50));
```

データ格納

```
例)insert into <テーブル名> (ID,Name) values (1,'oyaizu'); 文字列は"で囲う
```

ユーザー作成

```
例)create user '<ユーザー名>@'%' identified by '<パスワード>';
```

ユーザー権限割り当て

```
grant all on <データベース名>.* to '<ユーザー名>@'%';
```

アプリ側の設定

まずローカルのMySQLからデータを取るように設定する

[MySQL接続](#)の方法で

今回はLinuxにMySQLサーバーを設置したのでServer=localhost

appsettings.json

```
"DefaultConnection": "Server=localhost;Database=oyaizu_db_mysql;Uid=osuke_oyaizu;Password=password"
```

注意

sudo dotnet <ソリューション名>.Server.dllで5000番と5001番が開いてしまう場合は
[Server]

```
app.UseHttpsRedirection(); をコメントアウトする
```

デプロイ

発行→圧縮→s3保存→ダウンロード→解凍→sudo dotnet <ソリューション名>.Server.dll

環境変数

2023年8月2日 15:12

appsettings.jsonに書かずに環境変数から接続情報をとる

メリット：実際の環境ではデータベースのパスワードはローテーションで変わるので一か所変えればいい環境変数を使用する

デメリット：エラーが起きた時に内容がわからない

Linux環境変数設定方法 <https://rainbow-engine.com/linux-envvar-permanent/>

sudo vi ~/.bash_profile で以下を記入する

例) export DB_ConnectionString="Server=localhost;Database=oyaizu_db_mysql;Uid=osuke_oyaizu;Password=password"

確認

echo \$<環境変数名>

再起動か sudo /bin/bashを実行すると適用される

※OSに設定されている環境変数を読み込む為のコード

[Server]

Program.cs

```
string? connectionStrings = Environment.GetEnvironmentVariable("DB_ConnectionString");
builder.Services.AddDbContext<OyaizuDbMysqlContext>(a => a.UseMySql(connectionStrings,
ServerVersion.AutoDetect(connectionStrings)));
```

[Shared]

DbContext.cs

```
string? connectionStrings = Environment.GetEnvironmentVariable("DB_ConnectionString");
protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
=> optionsBuilder.UseMySql(connectionStrings, ServerVersion.Parse("8.0.33-mysql"));
```

dotnet <ソリューション名>.Server.dllで実行

※sudoで実行すると環境変数が実行されないのでエラーが起きる



```
① ▶ GET http://192.168.102.40/api/users  dotnet.6.0.18.0gj1vijzty9.js:1 ⓘ
500 (Internal Server Error)

② ▶ crit: Microsoft.AspNetCore.Components.WebAssembly.Rendering.WebAssemblyRenderer[100]
    Unhandled exception rendering component:
    net_http_message_not_success_statuscode, 500, Internal Server Error
    System.Net.Http.HttpRequestException:
    net_http_message_not_success_statuscode, 500, Internal Server Error
        at System.Net.Http.HttpResponseMessage.EnsureSuccessStatusCode()
        at System.Net.Http.Json.HttpClientJsonExtensions.
    <GetFromJsonAsyncCore>d__13`1[[System.Collections.Generic.List`1[[oyaizu_mysql.Shared.Models.User, oyaizu_mysql.Shared, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null]]], System.Private.CoreLib, Version=6.0.0.0, Culture=neutral, PublicKeyToken=7cec85d7bea7798e]].MoveNext()
        at oyaizu_mysql.Client.Pages.Index.OnInitializedAsync()
        at Microsoft.AspNetCore.Components.ComponentBase.RunInitAndSetParametersAsync()
        at Microsoft.AspNetCore.Components.RenderTree.Renderer.GetErrorHandledTask(Task , ComponentState )
```

アイコン

2023年8月1日 8:27



AWS-Archit
ecture-Ic...

大会

2023年8月9日 8:08

情報収集が目的・・・でたサービスについて

ZUracB シークレットキー

<https://jam.awsevents.com/>

osuke.oyaizu@gmail.com

Gakuen844#000

練習

2023年8月9日 11:04

RDS接続

2023年8月9日 11:04

RDS作成手順

サブネットグループを作成する

今回はプライベートサブネットA,Cを設定した

 申し訳ありませんが、DB サブネットグループ lab2-sub の作成リクエストは失敗しました。 

The DB subnet group doesn't meet Availability Zone (AZ) coverage requirement. Current AZ coverage:

ap-northeast-1a. Add subnets to cover at least 2 AZs.

→少なくとも2つ以上のサブネットを指定する必要がある

データベースを作成する

・ストレージタイプはgp2にする ※重要！

・自動バックアップの有効化はチェック外す

・マイナーバージョンの自動アップデートは無効にする

エンジンのオプション：MySQL

テンプレート：開発/テスト

マスターユーザー名＆パスワード：mysqlコマンドでログインする際に使う

インスタンスの設定：サイズに注意する（推奨：バースト可能クラスを選択し、t2を選択）

接続：EC2コンピューティングリソースに接続しない

サブネットグループ：作成したものを使用

VPCセキュリティグループ：ソースはEC2に割り当てているセキュリティグループ

ストレージ割り当て：20GiBにする

インバウンドルール 情報

| セキュリティグループ ID | タイプ | 情報 | プロトコル | ポート範囲 | ソース | 情報 | 説明 - オプション | 情報 |
|-----------------------|--------------|----|-------|-------|--------|--|---|---|
| | | 情報 | 情報 | 情報 | | | | |
| sgr-08d7363f54454a1b0 | MySQL/Aurora | ▼ | TCP | 3306 | カスタム ▾ | <input type="text"/>  |  |  |

sg-
0876c8484773
1eb3f 

ec2インスタンス作成

プライベートサブネットに配置する

セッションマネージャーで接続するためにロールに[AmazonSSMManagedInstanceCore](#)

[セッションマネージャー接続設定](#)

セキュリティグループにアウトバウンド全許可をする

mysqlインストール

「Amazon linux 2023」と検索する

[Amazon Linux 2023にMySQL をインストールする](#)

\$ sudo dnf -y localinstall <https://dev.mysql.com/get/mysql80-community-release-el9-1.noarch.rpm>

\$ sudo dnf -y install mysql mysql-community-client

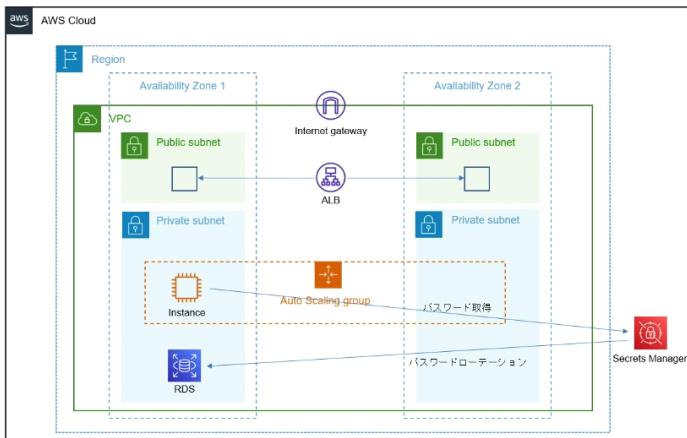
接続方法

\$ mysql -h database-1.cilprvhkuu3p.ap-northeast-1.rds.amazonaws.com -u <ユーザー名> -p

→データベースで設定したパスワードを入力する

接続できないとき

→セキュリティグループを確認する（インバウンドルールでMySQL接続の際にソースで接続元を指定する）



ロードバランサーのセキュリティグループはintraからのhttpsを許可

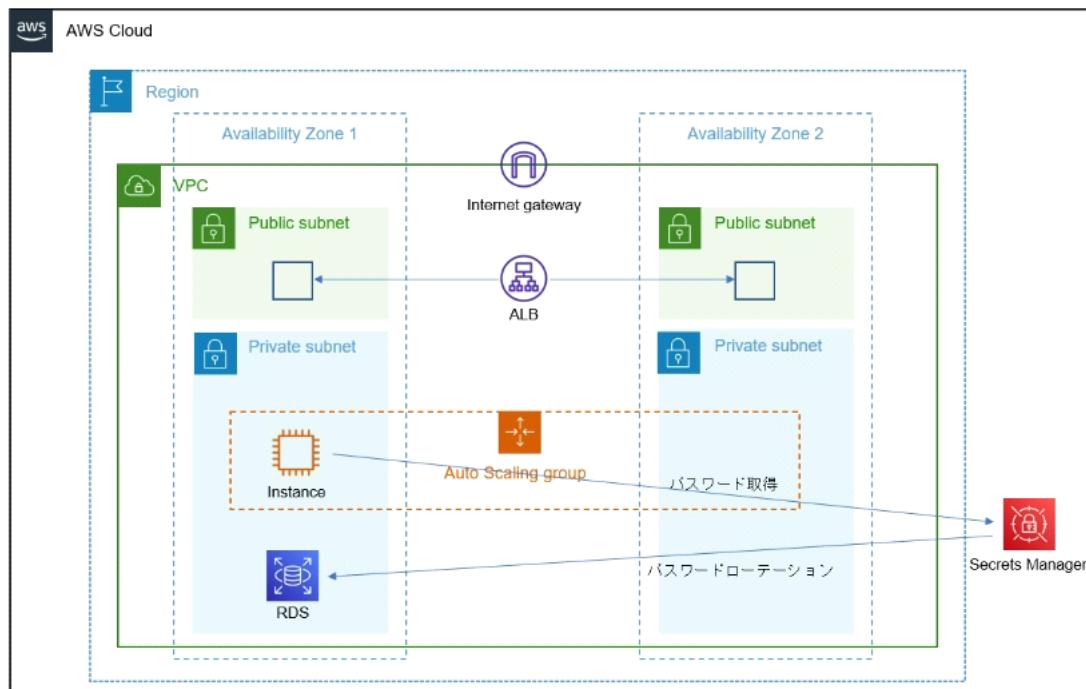
リスナーには443を設定する

インスタンスにソースにalbに割り当たっているもののhttpを許可

インスタンスでhttpd

Secret Manager

2023年8月10日 8:44



DevelopersIO のサイトが検索の際には見やすい

<https://dev.classmethod.jp/articles/about-secrets-manager/>

ログイン時に

認証情報を管理してくれる

→Secrets Managerに問合せがくる

ローテーション機能←データベースのパスワードを定期的に変更してくれる(実行は関数で行う)

「Secret Manager」 → 「新しいシークレットの作成」

ユーザー名：RDSのもの

パスワード：RDSのもの

シークレットの名前：一意の名前をつける

インスタンスの設定

sudo yum install python3-pip

sudo pip install boto3 botocore

.pyファイルの中にシークレットのサンプルコード(python3)をコピーする

以下のコードを一番下に追加する

```
print(secret)

if __name__ == '__main__':
    get_secret()
```

インスタンスにポリシーを割り当てる

オンラインポリシーを作成

```
python3 <pythonファイル名>
```

ででてきたエラーのポリシーを追加する(GetSecretValue)

ARN を指定

次のリソース:

- このアカウント 任意のアカウント その他のアカウント

リソースのリージョン

任意のリージョン

ap-northeast-1

Resource secret

任意の secret

dev/lab2-mysql-5jYi9y

ARN

任意のリソース

secret:dev/lab2-mysql-5jYi9y

"Version": "2012.10.17"

"Statement": [

1

"Sid": "VisualEditor0"

"Effect": "Allow"

"Action": "secretsmanager:GetSecretValue"

"Resource": "arn:aws:secretsmanager:ap-

自分のシークレットからみれる

northeast-1:157094121738:secret:dev/lab2-mysql-5jYi9v"

}

1

}

pypy3 <ファイル名>

```
→>{"username":"admin","password":"password","engine":"mysql","host": "database-1.cilprvhkuu3p.
```

```
ap-northeast-1.rds.amazonaws.com","port":3306,"dbInstancelIdentifier":"database-1"}
```

テーブルの中のデータを表示する (Secret Managerを使用して)

<https://qiita.com/mksamba/items/0080a342de180120073b>

Amazon Linux内でpythonでmysqlにアクセスする方法

```
$ sudo yum install mysql-connector-python3
```

コードの中身を置き換え

```
import mysql.connector  
connection = mysql.connector.connect(  
...  
)
```

下のコードを追加すると

```
python3 <ファイル名>  
→>{"username":"admin","password":"password","engine":"mysql","host":"database-1.cilprvhkuu3p.ap-northeast-1.rds.amazonaws.com","port":3306,"dbInstancelIdentifier":"database-1"}  
(1, 'oyaizu')  
(2, 'kimbara')
```

```
import mysql.connector
```

```
secret = ast.literal_eval(secret)
```

```
secret = ast.literal_eval(secret)
```

```
connection = mysql.connector.connect(  
    host=secret['host'],  
    user=secret['username'],  
    passwd=secret['password'],  
    db='lab2')
```

```
cursor = connection.cursor()
```

```
cursor.execute("SELECT * FROM lab2.user")
```

```
for row in cursor:
```

```
    print(row)
```

```
connection.commit()
```

```
connection.close()
```

```
if __name__ == '__main__':
```

```
    secret = get_secret()
```

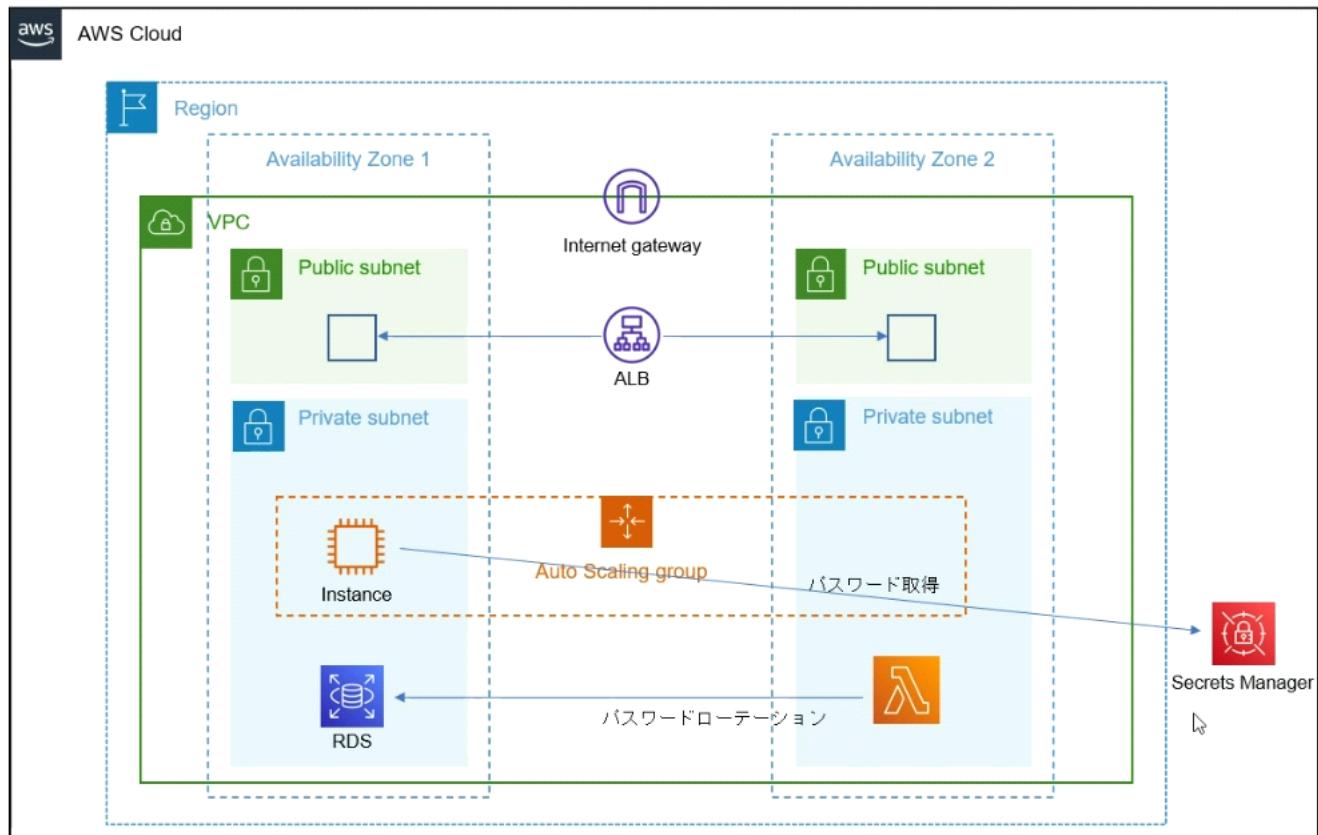
※上の関数を実行するコードを上に書く場合は**return (メソッド名)**が必要になる

パスワードのローテーションの設定

<https://dev.classmethod.jp/articles/secrets-manager-password-rotation-2022/>

DBのSGにLambda関数に割り当たっているSGのインバウンドルールに

MySQL/Aurora TCP 3306



ローテーション用のLambdaがプライベートサブネットに配置されるのでNATゲートウェイなどで外部と通信をできるようにする必要がある

KMS

2023年8月10日 14:05

鍵を管理してくれるサービス

<https://dev.classmethod.jp/articles/summary-aws-managed-key-and-customer-managed-key/>

鍵が2種類ある

AWSマネージドキー

カスタマーマネージドキー・・・アクセス権を変更できない

| キータイプ | 自動ロードーション | エイリアス名 | 使用範囲 | 管理*1 | 作成 | 削除 | 料金 |
|--------------|---------------|-------------|----------|------|-----------------------------------|----|--|
| AWSマネージドキー | 毎年 (必須) | 「aws/サービス名」 | 特定サービスのみ | AWS | AWSが作成 | 不可 | 従量課金 (リクエスト数) |
| カスタマーマネージドキー | 毎年 (無効化も可) | 何でも可 | ユーザー管理 | ユーザー | ①ユーザーが作成 ②外部から持込 ③カスタムキーストア | 可 | ①月額1ドル (日割り計算) ②従量課金 (リクエスト数) |

KMSからキーを作成しておく

S3のオブジェクトに暗号化設定をする 作成したkeyを指定する
デフォルトではs3の暗号化が使用されている

S3へのアクセスポリシーを追加しておく

```
aws s3 cp s3://lab23-docker/lab2-encrypt.txt /home/ec2-user/
```

```
→download failed: s3://lab23-docker/lab2-encrypt.txt to ./lab2-encrypt.txt An error occurred (AccessDenied) when calling the GetObject operation:User:arn:aws:sts::157094121738:assumed-role/lab2-WS_Lab_EC2/i-0b656860c7003dffbi is not authorized to perform: kms:Decrypt on resource: arn:aws:kms:ap-northeast-1:157094121738:key/70c5d9cf-196a-4fe2-96d9-bc9eefc816b6 because no identity-based policy allows the kms:Decrypt action
```

Decryptのポリシーを追加しておく



一番下にkeyのarnをコピーして貼り付ける

DeveloperTool

2023年8月22日 11:07

DeveloperTool・・・アプリ開発が便利になるサービス

Cloud9

2023年8月22日 11:28

Cloud9・・・コードの書き込み、実行、デバッグ用のクラウド IDE
IDE・・・コードの書き込み、実行、デバッグを行ってくれるのを含む(Visual Studioのようないいもの)

CodeCommit

2023年8月22日 11:27

CodeCommit・・・プライベートGitリポジトリ

Cloud9からファイルを追加する

プライベートサブネットにリポジトリを作成する
クローン用のURLをコピーする

Cloud9から追加する

git clone <パス> → ローカルにgitフォルダが作られる

git config --global user.name "oyaizu"

カレントディレクトリをgitフォルダに移動させる

git add <フォルダ内のファイルなど>

git commit -m "<コメント>"

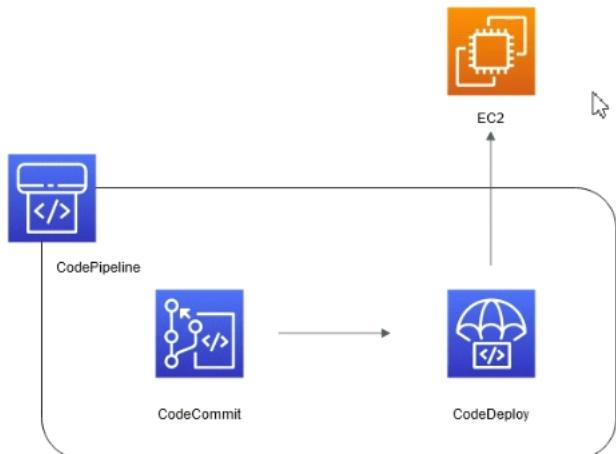
git push → される

CodeDeploy

2023年8月22日 13:07

構成図

簡単ですが、作成するコンポーネントの構成図はこのようになっています。



EC2手順

Cloud9コンソール上でコマンド実行 <https://blog.denet.co.jp/aws-codepipeline-tutorial/>

```
mkdir /tmp/code-demo && cd $_
wget
https://docs.aws.amazon.com/ja\_jp/codepipeline/latest/userguide/samples/SampleApp\_Linux.zip
unzip SampleApp_Linux.zip && rm SampleApp_Linux.zip
```

```
cp -r * ~/environment/<リポジトリ>
```

```
cd /environment/<リポジトリ>
```

```
ls
LabUserRole:~/environment/lab2-repo (master) $ ls
appspec.yml index.html LICENSE.txt scripts
```

```
git add .
git commit -m ""
git push
```

コードデプロイエージェントをインストール

https://docs.aws.amazon.com/ja_jp/codedeploy/latest/userguide/codedeploy-agent-operations-install-linux.html

```
sudo yum update
```

```
sudo yum install ruby
```

```
sudo yum install wget
```

```
cd /home/ec2-user
```

```
wget https://aws-codedeploy-ap-northeast-1.s3.ap-northeast-1.amazonaws.com/latest/install
```

```
chmod +x ./install
```

```
sudo ./install auto
```

確認コマンド

```
sudo systemctl status codedeploy-agent.service
```

デプロイ (CodeDeploy)

1. アプリケーション作成 (「CodeCommit」 → 「デプロイ」 → 「アプリケーション」)
2. デプロイグループ作成 (アプリケーションの中に入って設定する)

ロール

ポリシー名 ▾

⊕ AWSCodeDeployRole

⊕ AWSCodeDeployRoleForECS

環境設定 : EC2

タグ キー : Name 値 : <インスタンス名>

デプロイ設定 : 今回はインスタンスが一つのなのでAtOnceを選択する

ロードバランサー : 今回は使用していないので無効にする

3. デプロイ作成

デプロイ条件

インスタンスからコードデプロイに接続出来ていることが前提でデプロイ
インスタンスがインターネットに接続できる状態
インスタンスにコードデプロイのポリシーを割り当てる

- + LabS3Access

- + 📁 AmazonEC2RoleforAWSCodeDeploy

- + 📁 AmazonSSMManagedInstanceStateCore

appspec.ymlを読み込んで処理をする

(内容)

```
version:0.  
0  
2     os:linux  
3     files:  
4         - source:/index.html  
5             destination:/var/www/html/  
6     hooks:  
7         BeforeInstall:  
8             - location:scripts/install_dependencies      →scripts/install_dependenciesの内容  
9                 を実行  
10            timeout:300  
11            runas:root          →ユーザー権限  
12            - location:scripts/start_server  
13            timeout:300  
14            runas:root  
15            ApplicationStop:      →2回目以降の更新の時は再起動をする  
16            - location:scripts/stop_server  
17            timeout:300  
18            runas:root  
19
```

パイプライン

給水パイプラインが貯水池から蛇口まで水を移動するのと同じように、データパイプラインは収集ポイントからストレージにデータを移動します。データパイプラインは、ソースから

データを抽出し、変更を加えてから、特定の送信先に保存します。

メリット：開発者はpushをすれば本番環境にデプロイされる

<https://aws.amazon.com/jp/what-is/data-pipeline/#:~:text=AWS%20Data%20Pipeline%20%E3%81%AF%E3%80%81%E6%8C%87%E5%AE%9A,%E9%96%93%E3%81%A7%E7%A7%BB%E5%8B%95%E3%81%A7%E3%81%8D%E3%81%BE%E3%81%99%E3%80%82>

ソース

ソースプロバイダ：CodeCommit

ビルドステージ：今回はスキップする
デプロイステージ

ソースプロバイダ：CodeDeploy

確認方法

Cloud9を使ってindex.htmlを編集する

git pushをする

↓

パイプラインが実行される

↓

インスタンスから curl <http://localhost> や cat /var/www/html/index.html
などで変更されているかを確認する

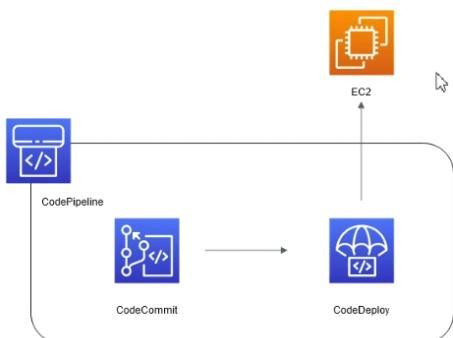
CodeBuild

2023年8月22日 14:59

<https://dev.classmethod.jp/articles/re-introduction-2022-aws-codebuild/>

構成図

簡単ですが、作成するコンポーネントの構成図はこのようになっています。



Dockerfile作成

```
FROM amazonlinux:2023
RUN yum install httpd -y
COPY index.html /var/www/html/
CMD ["/usr/sbin/httpd","-X"]
```

イメージ生成

docker build -t <名前> .

docker images → 確認する

docker run <名前>

```
LabUser@~/environment/lab2-repo (master) $ docker run lab2-httpd
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to
suppress this message
```

別ターミナルで

curl <http://172.17.0.2> → アクセスできる

①ECRにプッシュする

②クラスター作成

③タスクを定義する

ロールにecsTaskExecutionRole

⊕ CloudWatchLogsFullAccess

⊕ AmazonECSTaskExecutionRolePolicy

⊕ SecretsManagerReadWrite

⊕ GetParameterStore

④サービスを作成する

セキュリティグループでインスタンスからのアクセス許可をする

確認

curl <http://<タスクのプライベートIP>>

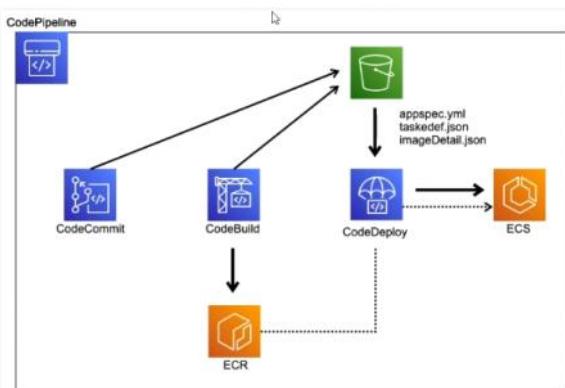
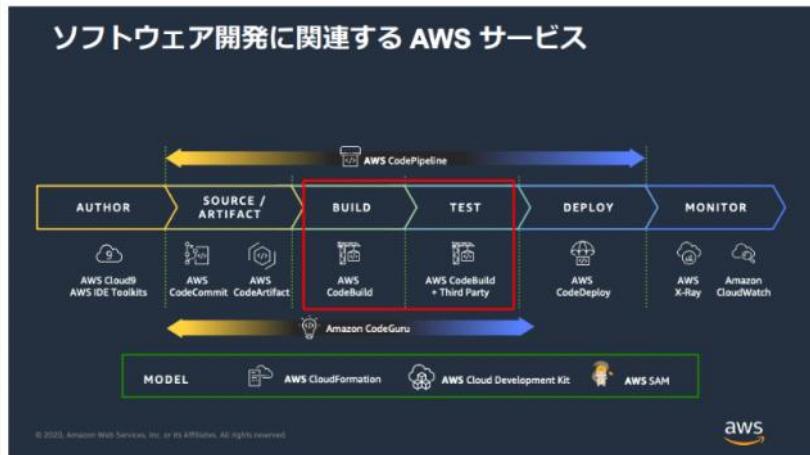
CodePipeline

2023年8月22日 18:13

git pushによってCodeCommitが更新される→CodeDeployが起動する→インスタンスにデプロイされる

ECS構築

2023年8月22日 16:11



今回はデプロイグループの作成は必要ない

DockerfileをCodeCommitにあげる

```
FROM amazonlinux:2023
RUN yum install httpd -y
COPY index.html /var/www/html/
CMD ["/usr/sbin/httpd","-X"]
```

scripts配下に

buildspec.ymlにコードを書き込む → CodeCommitにあげる

「codebuild docker」で検索

https://docs.aws.amazon.com/ja_jp/codebuild/latest/userguide/sample-docker.html

version: 0.2

```
phases:
  pre_build:
    commands:
      - echo Logging in to Amazon ECR...
      - aws ecr get-login-password --region $AWS_DEFAULT_REGION | docker login --username AWS --password-stdin
$AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com
  build:
    commands:
```

```

- echo Build started on `date`
- echo Building the Docker image...
- docker build -t $IMAGE_REPO_NAME:$IMAGE_TAG .
- docker tag $IMAGE_REPO_NAME:$IMAGE_TAG $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/
$IMAGE_REPO_NAME:$IMAGE_TAG
post_build:
commands:
- echo Build completed on `date`
- echo Pushing the Docker image...
- docker push $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/$IMAGE_REPO_NAME:$IMAGE_TAG

```

ビルドプロジェクト(CodeBuild)

環境イメージ

マネージド型イメージ
AWS CodeBuildによって管理されたイメージの使用

カスタムイメージ
Dockerイメージの指定

オペレーティングシステム

Amazon Linux 2

ランタイム

Standard

イメージ

aws/codebuild/amazonlinux2-x86_64-standard:5.0

イメージのバージョン

aws/codebuild/amazonlinux2-x86_64-standard:5.0-23.07.28

環境タイプ

Linux

特権付与

Dockerイメージを構築するか、ビルドで昇格されたアクセス権限を取得するには、このフラグを有効にします

サービスロール

新しいサービスロール
アカウントでサービスロールを作成

現存のサービスロール
アカウントから既存のサービスロールを選択

※特権付与のチェックをする

ロール

- CodeBuildBasePolicy-lab2-build-ap-northeast-1
- CodeBuildVpcPolicy-lab2-build-ap-northeast-1
- EC2InstanceProfileForImageBuilderECRContainerBuilds

VPCは選択しない

環境変数

VPC

AWS CodeBuildプロジェクトからアクセスするVPCを選択します。

コンピューティング

3 GBメモリ、2 vCPU

7 GBメモリ、4 vCPU

15 GBメモリ、8 vCPU

145 GBメモリ、72 vCPU

環境変数

| 名前 | 値 | タイプ | 削除 |
|--------------------|----------------|----------|----|
| AWS_DEFAULT_REGION | ap-northeast-1 | ブレーンテキスト | 削除 |
| AWS_ACCOUNT_ID | 608728620263 | ブレーンテキスト | 削除 |
| IMAGE_REPO_NAME | lab1-htpd | ブレーンテキスト | 削除 |
| IMAGE_TAG | latest | ブレーンテキスト | 削除 |

環境変数の追加

パラメータの作成

IMAGE_REPO_NAME・・・ECRのリポジトリ名

#3 ERROR: failed to copy: httpReadSeeker: failed open: unexpected status code

<https://registry-1.docker.io/v2/library/amazonlinux/manifests/sha256:aae09923a2c96cc7d6cc83873f3e1a31fd4c08f7f5d9df6567fdb6b8013e13bc>: 429 Too Many Requests - Server message: toomanyrequests: You have reached your pull rate limit. You may increase the limit by authenticating and upgrading: <https://www.docker.com/increase-rate-limit>

Too Many Requestは同一IPアドレスからのリクエスト上限に達した場合に出るエラー

codebuildでdockerイメージをpullする際、VPCを指定しないとAWSマネージドの環境で動作する。その際、AWS側のIPアドレスが振られるが、それは他のアカウントとも共有する。そのため、リクエスト総数が上限に達する場合がよくある。

対処法・・・Dockerfileを編集

```
FROM 608728620263.dkr.ecr.ap-northeast-1.amazonaws.com/amazonlinux:2023
RUN yum install httpd -y
COPY index.html /var/www/html/
CMD ["/usr/sbin/httpd", "-X"]
```

パイプライン作成(CodePipeline)

buildspec.ymlにコードを追加する

※インデントに注意する！！！

※無駄なスペースを入れない（下のコードをコピペしない）

「codepipeline ecs」で検索する

https://docs.aws.amazon.com/ja_jp/codepipeline/latest/userguide/ecs-cd-pipeline.html

```
- printf '[{"name":"lab2-httpd","imageUri":"%" }]' $AWS_ACCOUNT_ID.dkr.ecr.$AWS_DEFAULT_REGION.amazonaws.com/
```

```
$IMAGE_REPO_NAME:$IMAGE_TAG > imagedefinitions.json
```

↑

パイプラインが読み込むファイル

artifacts:

files: imagedefinitions.json

ECRのURL

コードを変更→push→パイプラインによってサービスを更新する

確認方法

新しく作成されたタスクのプライベートurlにアクセスする

IoT Core

2023年8月24日 11:04

中継的な役割

IoTを設定→インスタンスから情報を送る→処理させる

キーファイルは作成時にしかダウンロードできないので確実にダウンロードする
デバイス証明書・秘密鍵ファイルをダウンロードする

「管理」→「すべてのデバイス」→「モノ」

ポリシーを割り当てる

とりあえず全許可

ポリシードキュメント 情報

AWS IoT ポリシーには 1 つ以上のポリシーステートメントが含まれています。各ポリシーステートメントには、アクション、リソース、およびリソースによってアクションを許可または拒否する効果が含まれます。

ビルダー JSON

| | | |
|--------|-----------|----------|
| ポリシー効果 | ポリシーアクション | ポリシーリソース |
| 許可 | * | * |

新しいステートメントを追加

その他はデフォルト・名前を付けるだけ

証明書とキーをダウンロード

AWS に接続できるように、証明書とインストールするキーファイルをデバイスにダウンロードします。

デバイス証明書

証明書は今すぐアクティブ化することも、後でアクティブ化することもできます。デバイスが AWS IoT に接続するためには、証明書がアクティブである必要があります。

| | | |
|-------------------------------------|-------------|----------|
| デバイス証明書 8ad44da6c51...te.pem.crt | 証明書を非アクティブ化 | 💾 ダウンロード |
|-------------------------------------|-------------|----------|

キーファイル

キーファイルはこの証明書に固有であり、このページを離れるべくダウンロードできません。今すぐダウンロードして、安全な場所に保存してください。

⚠️ この証明書のキーファイルをダウンロードできるのは、この時点のみです。

| | | |
|--|----------|---------------|
| パブリックキーファイル 8ad44da6c51d0826f43b696...d831e2b-public.pem.key | 💾 ダウンロード | ✅ ダウンロードされたキー |
| プライベートキーファイル 8ad44da6c51d0826f43b696...831e2b-private.pem.key | 💾 ダウンロード | ✅ ダウンロードされたキー |

Cloud9にファイルをコピーする

<https://github.com/aws/aws-iot-device-sdk-python-v2> サンプルコード

git clone <サイトのURL>

\$ cd aws-iot-device-sdk-python-v2/samples/

pubsub.md で方法を確認できる

python3 pubsub.py --endpoint <endpoint> --cert <file> --key <file>

--endpoint ・・・ 「IoT Core」 → 「設定」 から確認できる

--cert ・・・ デバイス証明書のファイルパス

--key ・・・ privateがつくキーファイルパス

```
$ python3 pubsub.py --endpoint aw61t711eg2co-ats.iot.ap-northeast-1.amazonaws.com --cert ~/environment/8ad44da6c51d0826f43b696fe31bbdb281be4a3b690230b487b6c85bfd831e2b-certificate.pem.crt --key ~/environment/8ad44da6c51d0826f43b696fe31bbdb281be4a3b690230b487b6c85bfd831e2b-private.pem.key
```

エラーが出た場合

証明書と秘密鍵が合っていない →

MQTTを使用する

mqtt.publish

[MQTT テストクライアント](#)でtopicを確認できる

#・・・全部

```
$ python3 pubsub.py --endpoint aw61t711eg2co-ats.iot.ap-northeast-1.amazonaws.com --cert ~/environment/8ad44da6c51d0826f43b696fe31bbdb281be4a3b690230b487b6c85bfd831e2b-certificate.pem.crt --key ~/environment/8ad44da6c51d0826f43b696fe31bbdb281be4a3b690230b487b6c85bfd831e2b-private.pem.key --topic /test/lab2 --count 1
```

--topic ・・・ topicを指定できる

--count ・・・ 回数

ルールの作成

「メッセージのルーティング」 → 「ルール」 → 「ルールの作成」

SQLステートメントを設定

簡略化されたSQL構文を追加して、MQTTトピックで受信したメッセージをフィルタリングし、他の場所にデータをプッシュします。

SQLステートメント

SQLのバージョン
ルールを評価するときに使用するSQLルールエンジンのバージョン。
2016-03-23

SQLステートメント
以下のSELECT <Attribute> FROM <Topic Filter> WHERE <Condition>を使用してSQLステートメントを入力します。たとえばSELECT temperature FROM test/topic WHERE temperature > 50です。詳細については、「AWS IoT SQLリファレンス」を参照してください。

```
SELECT * FROM 'test/lab2'
```

今回はアクションにLambdaを指定する

ルールアクションをアタッチ

アクションは、特定のAWSのサービスにデータをルーティングします。

SQLステートメント

```
SELECT * FROM 'test/lab2'
```

ルールアクション

インバウンドメッセージが上のルールに一致したときに実行される1つ以上のアクションを選択します。アクションは、メッセージが到達した際に実行される追加のアクティビティ（データベースへの操作、クラウド関数の呼び出し、通知の送信など）を定義します。最大10個のアクションを追加できます。

アクション1

Lambda
Lambda関数にメッセージを送る

Lambda関数
lab_iot_message

Lambda関数のバージョン
\$LATEST

ルールアクションを追加

エラーを確認したい場合はルールでエラーアクションの設定をする

エラーアクション - オプション

必要に応じて、ルールの処理で問題が発生したときに実行されるアクションを設定できます。同じルールの2つのルールアクションが失敗した場合、エラーアクションは両方のエラーを含む1つのメッセージを受け取ります。

CloudWatch logs
CloudWatch Logsにメッセージデータを送信

削除

ログのグループ名
CloudWatch Logグループを選択

CloudWatch Logグループを作成

パッチモード
レコードのJSON配列を含むペイロードは、パッチコールを介してCloudWatchに送信されます。
 パッチモードを使用

IAMロール
エンドポイントへのアクセス権をAWS IoTに付与するロールを選択します。

IAMロールを選択

表示

新しいロールを作成

AWS IoTは、選択したIAMロールの下に「aws-iot-rule」というプレフィックスが付いたポリシーを作成します。

エラーアクションを追加

```
python3 pubsub.py --endpoint aw61t711eg2co-ats.iot.ap-northeast-1.amazonaws.com --cert ~/environment/8ad44da6c51d0826f43b696fe31bbdb281be4a3b690230b487b6c85bfd831e2b-certificate.pem.crt --key ~/environment/8ad44da6c51d0826f43b696fe31bbdb281be4a3b690230b487b6c85bfd831e2b-private.pem.key --topic test/lab2 --count 1 --message "lab2"
```

ポリシーの設定

| アクティブなバージョン: 3 倍幅 | | |
|-------------------|---------------|---|
| ポリシー効果 | ポリシーアクション | ポリシーリソース |
| Allow | iot:Connect | arn:aws:iot:ap-northeast-1:608728620263:client/lab1-* |
| Allow | iot:Publish | arn:aws:iot:ap-northeast-1:608728620263:topic/test/lab1 |
| Allow | iot:Receive | arn:aws:iot:ap-northeast-1:608728620263:topic/test/lab1 |
| Allow | iot:Subscribe | arn:aws:iot:ap-northeast-1:608728620263:topicfilter/test/lab1 |

--topic test/lab1 → できる

--topic test/lab2 → できない

--client lab1-001 → できる

--client lab2-001 → できない

課題で出るかも？

指定したtopicに来ないとルールで定義したものが動作しない

SQL ステートメント

SQL ステートメント

```
SELECT * FROM 'test/lab2'
```

SQL のバージョン

2016-03-23

Systems Manager

2023年8月24日 15:57

Session Manager

Fleet Manager

Run Command ・・・複数のインスタンスなどに同時にコマンドを実行したりできる
ポリシー追加

The screenshot shows the AWS Lambda Policies page. At the top, there are two dropdown menus: 'ポリシー名' (Policy Name) and 'タイプ' (Type). Below these, a list of policies is displayed, with one item selected: 'AmazonSSMFullAccess'. To the right of this item is a 'AWS 管理' (AWS Management) link.

アプリケーション管理

- パラメータストア（様々な値を渡すことができる）

データベースの扱いに関してはCecret Manager（ローテーション機能）の方がいい
パラメータを作成

The screenshot shows the 'Create Parameter' dialog for the AWS Parameter Store. The 'Name' field is set to '/test/lab2/param2'. The 'Type' is set to 'Text'. The 'Value' field contains a JSON object: { "value": "teststring1" }. The 'Tags' section at the bottom has a single tag: 'タグ' (Tag).

安全な文字列 → 暗号化されるので

```
aws ssm get-parameter --name /test/lab2/param2 --with-decryption
```

--with-decryption ・・・復号化して表示

- AppConfig

プロファイルの設定



自由形式の設定プロファイルを構築

固定プロファイルの詳細
このページはプロファイルを選択して、設定プロファイルの情報を入力してください。 詳細はごちら

名前
lab2 profile
最大文字数は 64 文字です。

説明
lab2 profile

設定ソース
設定ソースを複数選択していただけます。

AWS AppConfig プロファイル
AWS AppConfig のプロファイルを複数選択できます。

Amazon S3 プロジェクト
Amazon S3 バケットから AWS AppConfig のプロファイルを読み取ります。

AWS Systems Manager グループメント
AWS Lambda フィルターとして AWS Systems Manager のグループメントを読み取ります。

AWS Secrets Manager シークレット
AWS Lambda フィルターとして AWS Secrets Manager のシークレットを読み取ります。

AWS CloudFormation
AWS Lambda フィルターとして AWS CloudFormation を読み取ります。

内容
データを JSON 形式で提供するか、 YAML、 XML、またはフレーメット形式で提供します。

Text
Specify data content in text format.

```
stage: "dev"
region: "ap-northeast-1"
environment: "test"
```

JSON
Specify data content in JSON format.

XML
Specify data content in XML format.

・環境

環境を作成

環境の詳細
環境は、該当デプロイグループと一緒にモニターです。 詳細はごちら

名前
env
最大文字数は 64 文字です。

説明 - オプション
lab2

モニター

拡張機能を関連付ける

タグ

```
aws appconfig get-configuration --application <アプリケーション名> --environment <環境名> --configuration <profile名> --client-id test config
```

デプロイする

※デプロイ戦略でAtOnceを指定すると早い

AWS Systems Manager > AppConfig > iab2 > dev > デプロイを開始

デプロイを開始

デプロイの詳細
このページのオプションを選択して、新規または更新されたアプリケーション設定をデプロイします。 [詳細はこちら](#)

設定
デプロイする設定を選択してください。
unison

または [設定プロファイルを作成](#)

ホストされた設定バージョン
デプロイする新規または更新された設定バージョンを選択してください。
1

デプロイ範囲
デプロイする環境(アーティフicial)または開発にデプロイします。 [詳細はこちら](#)

または [デプロイ範囲を作成](#)

AppConfig AllAtOnce (Quick)

デプロイの説明 - オプション
このデプロイの説明を入力してください。

▶ 追加の暗号化オプション [削除](#)

▶ タグ

キャンセル [デプロイを開始](#)



- 設定変更のエラーを減らす ...
- 複数のターゲットにわたって迅速に変更をデプロイする ...
- 中断することなくアプリケーションを更新する ...
- アプリケーション全体で変更のデプロイを制御する
- 設定変更後にエラーが起きた際にはロールバックできる

徐々にデプロイしていくのですべての顧客にエラーの影響が出る前にロールバックできる

athena

2023年8月24日 18:01

ファイルに対してスケールを行えるサービス

ファイルからデータを選択したい→athena

s3などに保存されているファイルに対してSQLを実行できるサービス

The screenshot shows two main sections of the AWS Athena console.

Top Section (Settings):

- Header: エディタ | 最近のクエリ | 保存したクエリ | **設定** | ワークグループ primary | 管理
- Sub-section: クエリの結果と暗号化の設定
- Details:
 - クエリの結果の場所と暗号化:
クエリの結果の場所: s3://ahaws-bucket-01/athena/ | クエリ結果を暗号化: *
 - 予期されるバケット所有者: 608728620263
 - 権限: パケット所有者にクエリ結果に対する完全なコントロールを割り当てるオフになっています

Bottom Section (Query Editor):

- Header: エディタ | 最近のクエリ | 保存したクエリ | 設定
- Info Bar: SQL クエリの開発をスピードアップするために、Athena で先行入力のコード推奨はデフォルトで有効になっています。この設定はクエリエラ
- Query Editor:
 - クエリ 1: SELECT * FROM "d1"
 - データソース: AwsDataCatalog
 - データベース: default
 - テーブルとビュー:
 - 作成
 - データソースからテーブルを作成
 - S3 バケットデータ (selected)
 - AWS Glue クローラ
 - SQL を使用して作成
 - CREATE TABLE
 - CREATE TABLE AS SELECT
 - CREATE TABLE AS SELECT(ICEBERG)
 - CREATE VIEW

S3 バケットデータからテーブルを作成 情報

テーブルの詳細

テーブル名

テーブル名は 1~128 文字である必要があります また、一意である必要があります。 有効な文字は、a~z、A~Z、0~9、_ (アンダースコア) です。 テーブル名は、データが保存されるディレクトリに対応する傾向にあります。

説明 - オプション

テーブルの説明は 1~1024 文字である必要があります。 1020 文字残り。

データベース設定 情報

既存のデータベースを選択するか、新しいデータベースを作成

新しいテーブルを作成するために、既存のデータベースにアクセスするか、新しいデータベースを作成することを選択します。 Athena はデータスキーマを AWS Glue データカタログに保存します。

データベースを作成

既存のデータベースを選択

データ形式 情報

テーブルタイプ

ファイル形式

SerDe ライブラリ

SerDe プロパティ - オプション

名前

値

削除

削除

削除

削除

作成時にエラーが起きた場合は最初のクエリ結果の場所の設定ができていない可能性がある

ターゲットグループのヘルスチェックパス

2023年8月30日 8:33

/health にアクセスしたときに200番が返ってきた時のみhealthyになる

The screenshot shows the 'Health Check Settings' section of a target group configuration. It lists the following parameters:

| プロトコル | ポート | 正常のしきい値 |
|------------------------------|---------------|------------------|
| HTTP | トラフィックポート | 5 ヘルスチェックの連続的な成功 |
| 非正常のしきい値 2 ヘルスチェックの連続的な失敗 | 間隔 30 秒 | 成功コード 200 |
| バス /health | タイムアウト 5 秒 | |

コマンドから

```
curl http://192.168.1.1:5000 -I  
-I ・・・ ヘッダー情報を確認できる
```

The screenshot shows the 'Registered Targets (2)' section. Two targets are listed:

| インスタンス ID | 名前 | ポート | ゾーン | ヘルスステータス | ヘルスステータスの詳細 |
|---------------------|-------|------|-----------------|-----------|--|
| i-049e1ebf166543df8 | albok | 5000 | ap-northeast-1c | unhealthy | Health checks failed |
| i-0e4f6657ccb50a625 | albng | 5000 | ap-northeast-1a | unhealthy | Health checks failed with these codes: [500] |

ヘルスステータス

- Request Time Out ・・・そもそも繋がっていない
- Health checks failed
- Health checks failed with these codes:[500] ・・・ 500番エラーが返ってきてfailed

ポートが違う

アプリケーションが動作していない

ヘルスチェックパスが違う

Glue

2023年8月28日 16:12

クローラ

データの構造を読み込む

RDSでデータベース・テーブル・データを作成しておく

ETS

ロール

s3のGetObject・PutObject(AWSGlueServiceRole-01)を割り当てる

| | ポリシー名 | タイプ |
|--------------------------|-----------------------|---------|
| <input type="checkbox"/> | AWSGlueServiceRole-01 | カスタマー管理 |
| <input type="checkbox"/> | AWSGlueServiceRole | AWS 管理 |

Set crawler properties

Crawler details [Info](#)

Name: lab2
Name can be up to 255 characters long. Some character set including control characters are prohibited.

Description - optional
Enter a description
Descriptions can be up to 2048 characters long.

Tags - optional
Use tags to organize and identify your resources.

[Cancel](#) [Next](#)

Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

Not yet
Select one or more data sources to be crawled.

Yes
Select existing tables from your Glue Data Catalog.

Data sources (1) [Info](#)
The list of data sources to be scanned by the crawler.

| Type | Data source | Parameters |
|------|---------------------------|-------------|
| S3 | s3://ahaws-bucket-01/lab/ | Recrawl all |

Custom classifiers - optional
A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.

[Cancel](#) [Previous](#) [Next](#)

Configure security settings

IAM role [Info](#)

Existing IAM role
AWSGlueServiceRole-01 [View](#) [Edit](#)

Create new IAM role [Update chosen IAM role](#)

Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole." can be updated.

Lake Formation configuration - optional

Allow the crawler to use Lake Formation credentials for crawling the data source. [Learn more](#)

Use Lake Formation credentials for crawling S3 data source
Checking this box will allow the crawler to use Lake Formation credentials for crawling the data source. If the data source is registered in another account, you must provide the registered account ID. Otherwise, the crawler will crawl only those data sources associated to the account. Only applicable to S3, Glue Catalog, Iceberg, and Hudi data sources.

► Security configuration - optional

Enable at-rest encryption with a security configuration.

[Cancel](#) [Previous](#) [Next](#)

Set output and scheduling

Output configuration [Info](#)

Target database
default [View](#) [Edit](#)

[Clear selection](#) [Add database](#)

Table name prefix - optional
lab2

Maximum table threshold - optional
This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.
 Type a number greater than 0

► Advanced options

Crawler schedule

You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. [Learn more](#)

Frequency
On demand

[Cancel](#) [Previous](#) [Next](#)

「Run crawler」で実行できる
→ 「Tables」で作られる → athenaで使用できる

コネクション作成

[AWS Glue](#) > [Connectors](#) > Create connection

Create connection [Info](#)

Connection properties [Info](#)

Name
Enter a unique name for your connection.
lab2-mysql

Connection type
Amazon RDS

Require SSL connection
The connection will fail if it's unable to connect over SSL.

Database engine
MySQL

Description - optional
Descriptions can be up to 2048 characters long.

Connection access

Database instances
Provisioned Amazon Relational Database Service instances.

lab2-rds

Database name
unicorndb

Credential type
 Username and password
 AWS Secrets Manager

Username
admin

Password

VPC : RDSがあるVPC

Subnet : 今回はプライベートサブネット

セキュリティグループ : RDSが接続できるもの

▼ Network options

If your AWS Glue job needs to run on [Amazon Elastic Compute Cloud](#) (EC2) instances in a virtual private cloud (VPC) subnet, you must provide additional VPC-specific configuration information.

VPC [Info](#)

Choose the virtual private cloud that contains your data source.

vpc-09b7e4021b8de02e7

Subnet [Info](#)

Choose the subnet within your VPC.

subnet-0ce311d65e872df99

arn:aws:ec2:ap-northeast-1:608728620263:subnet/subnet-0ce311d65e872df99
zone: ap-northeast-1a

Security groups [Info](#)

Choose one or more security groups to allow access to the data store in your VPC subnet. Security groups are associated to the ENI attached to your subnet. You must choose at least one security group with a self-referencing inbound rule for all TCP ports.

Choose one or more security group

sg-02faa022e99790a02
default

クローラに追加する → Run crawler

| Crawler runs | Schedule | Data sources | Classifiers | Tags |
|--|---------------------------|------------------------------|-----------------------------|----------------------|
| Data sources (2) Info The list of data sources to be scanned by the crawler. | | | | |
| <input type="radio"/> S3 | Type | Data source | Parameters | |
| <input type="radio"/> JDBC | s3://ahaws-bucket-01/lab/ | unicorndb/% | Recrawl all | - |

Include path : <DB名>/%

Add data source

Data source
Choose the source of data to be crawled.

JDBC

Connection
Select a connection to access the data sources below.

lab2-MySQL

Clear selection **Add new connection**

Include path
unicornedb/%

You can substitute the percent (%) character for a schema or table. For databases that support schemas, enter MyDatabase/MySchema/% to match all tables in MySchema within MyDatabase. Oracle Database and MySQL don't support schema in the path; instead, enter MyDatabase%. For Oracle database without SSL, MyDatabase can be either the system identifier (SID) or the service name (SERVICE_NAME). For Oracle database with SSL, MyDatabase must be the service name (SERVICE_NAME).

Additional metadata - optional

Select additional metadata properties for the crawler to crawl.

Exclude files matching pattern

Cancel **Add a JDBC data source**

s3に保存されているファイルからデータベースにデータ移行

「ETL jobs」

AWS Glue Studio [Info](#)

Create job [Info](#)

Visual with a source and target
Start with a source, ApplyMapping transform, and target.

Visual with a blank canvas
Author using an interactive visual interface.

Spark script editor
Write or upload your own Spark code.

Python Shell script editor
Write or upload your own Python shell script.

Jupyter Notebook
Write your own code in a Jupyter Notebook for interactive development.

Ray script editor [New](#)
Write your own code to run on Ray.

Source **Target**

Amazon S3 → MySQL
JSON, CSV, or Parquet files stored in S3. AWS Glue Data Catalog table with MySQL as the data target.

Your jobs (1) [Info](#)

| Job name | Type | Last modified | AWS Glue version |
|---------------|----------|--------------------|------------------|
| Lab1S3toMySQL | Glue ETL | 2023/8/28 15:59:19 | 4.0 |

Data source properties - S3

Name
S3 bucket

S3 source type [Info](#)
 S3 location
 Choose a file or folder in an S3 bucket.
 Data Catalog table

S3 URL

 Recursive
 Read files in all subdirectories.

Data format

JsonPath - optional
 Identify records with a JsonPath expression.

Multiline
 Indicates if JSON records can span multiple lines.

▶ Additional options

「Infer schema」をクリックする

Transform

Name
Change Schema

Node parents
 Choose which nodes will provide inputs for this one.

S3 bucket
 S3 - DataSource

Change Schema (Apply mapping)

| Source key | Target key | Data type | Drop |
|------------|------------|-----------|--------------------------|
| id | id | int | <input type="checkbox"/> |
| name | name | varchar | <input type="checkbox"/> |
| email | email | varchar | <input type="checkbox"/> |
| phone | phone | varchar | <input type="checkbox"/> |

Data target properties - MySQL

Name
MySQL table

Node parents
 Choose which nodes will provide inputs for this one.

Change Schema
 ApplyMapping - Transform

Database
 default

▶ Use runtime parameters

Table
 lab1unicorndb_unicorns

▶ Use runtime parameters

Data target properties - MySQL Output schema Data preview

Name
MySQL table

Node parents
Choose which nodes will provide inputs for this one.
Choose one or more parent node

Change Schema [ApplyMapping - Transform](#)

Database
default

▶ Use runtime parameters

Table
lab2unicorndb_unicorns

▶ Use runtime parameters

左上で名前を変更しておく

Job detailsタブの修正必要箇所を修正（ロールの割り当て）して「Save」する
「Run job」をするとデータが移行される

レイヤー追加

2023年8月30日 8:41

Lambdaでコードに

mkdir python ※他の名前は使用できない！

sudo yum install python3-pip -y

pip install pandas -t python/

zip python.zip -r python/ ※ディレクトリ毎zipしなければだめ

※依存関係があるものもインストール

バージョンが古いものだとLambdaでTestしてもエラーが起きる

→ pip install pandas numpy==1.25.2 -t python/

→ やり直す

Lambdaのランタイムと同じ

s3バケットに送る

レイヤー作成でs3バケットから取得する

ランタイムを関数と一致させる

awsがpythonディレクトリを見に行くのでpythonディレクトリに展開されない

機械学習

2023年8月30日 9:14

- Translate
- 自動言語翻訳サービス
- Textract
- PDFなどのファイルからテキストを抽出する
- Transcribe
- 音声を入力し、書き起こしする
- Lex
- チャットボット等の会話インターフェースを作成する
- Rekognition
- 機械学習を利用した画像や動画の分析を簡単に実施できる
- Polly
- テキストを音声に変換する
- Kendra
- 様々なデータソースを横断的に検索できるサービス
- Forecast
- 統計アルゴリズムと機械学習アルゴリズムを使用して、時系列予測を実現する
- Comprehend 感情
- テキストから洞察を見つける自然言語処理サービス
- Lookout for Vision
- 工業製品の視覚的欠陥を正確かつ大規模に見つけることができる
- Fraud Detector
- 機械学習でオンライン不正をより早く検出する
- Lookout for Metrics
- 時系列データから異常を検出し、その発生原因の抽出までを一貫して運用管理できる
- HealthLake
- HIPAA（医療保険の相互運用性と説明責任に関する法律）に準拠したサービス
- 医療関連データの作成、読み取り、更新、削除、クエリが可能になる
- Monitron
- 産業機械の異常な動作を検出するサービス
- 専用デバイスを取り付ける？日本では未対応
- Lookout for Equipment
- 温度やモーターの回転数、湿度、水の流量、圧力など、様々なセンサーデータを機械学習により、リアルタイムに分析する
- DevOps Guru
- アプリケーションが平常時と逸脱した動きをしたことを検知する

[【簡単導入】機械学習で運用を効率化！Amazon DevOps Guruを有効化してみた | DevelopersIO \(classmethod.jp\)](#)

- Personalize
- レコメンドシステム

[Amazon Personalizeを使ってみた | DevelopersIO \(classmethod.jp\)](#)

Sage Maker

2023年8月30日 9:23

<https://www.intellilink.co.jp/column/ai/2018/091300.aspx>

一から学ばせるときに使用する

Amazon SageMaker > ノートブックインスタンス > ノートブックインスタンスの作成

ノートブックインスタンスの作成

Amazon SageMaker は、Jupyter ノートブックを実行する構築済みでフルマネージド型のノートブックインスタンスを提供します。ノートブックインスタンスには、一般的なモデルトレーニングおよびホスティングの演習のためのコード例が用意されています。 [詳細はこちら](#)

ノートブックインスタンス設定

ノートブックインスタンス名
lab2

最大 63 文字の英数字を使用できます。ハイフン (-) は含めることができます。スペースは含めないでください。一つのAWS リージョンのアカウント内で一意である必要があります。

ノートブックインスタンスのタイプ
ml.t3.medium

Elastic Inference [詳細はこちら](#)

なし

プラットフォーム識別子 [詳細はこちら](#)

Amazon Linux 2, Jupyter Lab 3

▶ 追加設定

アクセス許可と暗号化

IAM ロール
ノートブックインスタンスでは、SageMaker と S3 を含む他のサービスを呼び出すアクセス許可が必要です。ロールを選択するか、[AmazonSageMakerFullAccess](#) AWS に IAM ポリシーがアタッチされたロールを作成させます。

AmazonSageMakerServiceCatalogProductsUserRole

ロール作成ウィザードを使用してロールを作成

ルートアクセス - オプション

有効化 - ノートブックへのルートアクセス権をユーザーに付与する

無効化 - ノートブックへのルートアクセス権をユーザーに付与しない

ライフサイクル設定には常にルートアクセス権が付与されます

暗号化キー - オプション
ノートブックデータを暗号化します。既存の KMS キーを選択するか、キーの ARN を入力します。

カスタム暗号化なし

VPC内のデータにアクセスしたいときはVPCを選択する

▼ ネットワーク - オプション

VPC - オプション
VPC 設定が指定されていないため、ノートブックインスタンスには、SageMaker で提供されたインターネットアクセスが提供されます。

非 VPC

▶ Git リポジトリ - オプション

▶ タグ - オプション

キャンセル [ノートブックインスタンスの作成](#)

「Jupyterを開く」からコードを書き込んで実行させることができる
pythonなどを使用できる

コードでForbidden

本番で出た場合はどのように出題されるのかわからぬので
解けなくても問題の内容を情報収集する

Textract

2023年8月30日 9:23

事前に準備されているモデルを使用する

東京リージョンにはまだない

バージニア北部が最新

日本語はまだ読み込めない

画像から文字を読み取るコード

```
import boto3

client = boto3.client('textract', region_name='us-east-1')

with open('vaccination_card.jpg', 'rb') as file:
    img_test = file.read()
    bytes_test = bytearray(img_test)

response = client.analyze_document(Document={'Bytes':bytes_test}, FeatureTypes=['FORMS'])

print(response)
```

Redshift

2023年8月30日 10:46

AWS上にデータウェアハウスを構築することができるマネージド型サービス

データウェアハウス(DWH)・・・企業内の複数システムから大量のデータを時系列で蓄積するシステム
生成されたデータをそのまま格納する。使用する側が形を変更して使用する。

データベースとの違い <https://www.dal.co.jp/column/b-dwh/>

データ分析に特化したデータベース

高速でスケーラブルな費用対効果の高いマネージド型のDWH

- ・データ分析用のリレーションナルデータベースであり、OLTPには利用不可

ノードタイプ

利用するデータサイズやケースに応じてノードタイプを選択

RA3：データ量の増大が予想される場合に推奨（もっとも高性能）

DS2：低コストで利用可能

DC：500GB超える際に推奨

クラスターというグループ単位で、複数ノードによってデータ処理を実行する構成

マルチAZ構成可能

Redshift Serverless

クラスターをセットアップ、調整、管理することなく、データにアクセスして分析が可能な構成

インスタンス パブリックサブネットA

Redshift プライベートサブネットA・B・C

データレイクのデータ解析用としても使用できる

クロスAZクラスターリカバリー・・・スナップショットを利用して他のAZに構築し直す

運用の自動化

CloudWatchとの連動

バックアップ

自動メンテナンス

機械学習によるクエリ効率化

テーブルメンテナンスの自動化

自動ワークロードの管理

ショートアクセラレーション

設定のレコメンデーション

トラフィック制御

拡張VPCルーティングによってVPCにトラフィックを強制しつつ、モニタリングが可能

KMCやACMで暗号化を実施

保存データの暗号化

通信の暗号化

ワークロード管理 (WLM)

スケーリング

ノードのタイプ変更・追加とクラスターの追加によってスケーリング可能

Redshift Spectrum

ユーザーが管理するS3バケットに対して直接データ解析を実行可能

データ連携(To Redshift)

S3

Kinesis

RDS

DynamoDB

Amazon EMR

Amazon

データ連携(From Redshift)

Amazon QuickSight

S3

Amazon Machine Learning

RDS

リザーブドノードの利用

クラスターというグループ単位で、複数ノードによってデータ処理を実行する構成

構築

2023年8月30日 11:02

クラスターを作成 [\[編集\]](#)

Looking for free trial? Try Redshift Serverless. First-time Redshift Serverless customers receive a \$300 credit to use in their account. [Launch Redshift Serverless](#) [×](#)

クラスター設定

クラスター名 この名前は、クラスターを識別する唯一のキーです。

クラスターのサイズを選択する
 を選択します。
 選択のヘルプ

ノードの種類 [\[確認\]](#)
CPU, RAM, ストレージ容量、およびドライブタイプの组合せを決めるノードの種類を選択します。
 ▾

ノード数
このクラスター内のノードの数を入力します。
 説明 (1~32)

設定の概要 [\[確認\]](#)
dc2.large | 2 個のノード

料金
\$458.44/月
オンデマンドコンピューティング料金の見積もり
リザーブドモードを購入することで、コストの 60% 稼働を節約できます。 [詳しくはごちら](#) [?] [\[確認\]](#)

320 GB
圧縮ストレージの合計
選択したノード数をデプロイする場合のクラスターの合計ストレージ容量です。

サンプルデータ [\[情報\]](#)

サンプルデータをロード
Redshift クラスターにサンプルデータをロードし、クエリエディタを使用してデータのクエリを開始します。

データベース設定

管理者ユーザー名
DB インスタンスの管理者ユーザーのログイン ID を入力します。
 名前は 1~128 文字の英数字にする必要があります。 [予約語](#) [?] にすることはできません。

パスワードを自動生成
Amazon Redshift はパスワードを生成することも、独自のパスワードを指定することもできます。

管理者ユーザー/パスワード

8~64 文字である必要があります。少なくとも 1 つの大文字、1 つの大文字、および 1 つの数字を含める必要があります。「/」、「*」、または「@」を除く任意の印刷可能な ASCII 文字を使用できます。

パスワードを表示

▼ ネットワークとセキュリティ 情報

Virtual Private Cloud (VPC)
このVPCは、このクラスターの仮想ネットワーキング環境を定義します。

vpc01
vpc-06246538b63f6ddde

① クラスターの作成後に、このクラスターに開通付けられているVPCを変更することはできません。詳細 はこちら

VPCセキュリティグループ
このVPCセキュリティグループは、クラスターがVPCで使用できるサブネットとIP範囲を定義します。

1つ以上のセキュリティグループを選択する

sg-0eff21fe889480ff8 X

クラスターサブネットグループ 情報
クラスターを起動するAmazon Redshiftサブネットグループを選択します。

cluster-subnet-group-1

アベイラビリティーゾーン
クラスターを作成するアベイラビリティーゾーンを指定します。そうしない場合、Amazon Redshiftによってアベイラビリティーゾーンが選択されます。

No preference

拡張されたVPCのルーティング
このオプションを有効にすると、インターネットではなくVPC経由で、クラスターとデータリポジトリ間のネットワークトラフィックが強制されます。Learn more about getting started cluster in vpc

オフにする
 オンにする

パブリックにアクセス可能
 [パブリックにアクセス可能] をオンにする
Allow public connections to Amazon Redshift.

① It can take about ten minutes for the setting to change and connections to succeed.

その他はデフォルト

VPCのルーティング：VPC経由でRedshiftにアクセス トライフィックを制御できる
パブリックアクセスなどは本来しないほうがいい

「メンテナンス」タブからスナップショットや使用制限ができる

「Redshift】→「設定」

クラスター識別子はRA3でないとできない

Redshift マネージド VPC エンドポイントを作成

クラスターにアクセスするためのエンドポイントを作成します。クラスターが別のアカウントにある場合、所有者はクラスターへのアクセス権を付与する必要があります。

エンドポイントの設定

エンドポイント名

作成したエンドポイントのプレフィックスを入力します。

udemyc

名前は 1~30 文字にする必要があります。有効な文字は、a~z、0~9、およびハイフン (-) です。最初の文字は文字である必要があります。名前には、連続する 2 つのハイフンを含めることはできず、ハイフンで終わることもできません。

AWS アカウント ID

608728620263 (マイアカウント) ▾

クラスター識別子

この識別子は、このエンドポイントで接続するクラスターを指定します。

クラスター識別子を選択 ▾

Virtual Private Cloud (VPC)

エンドポイントが作成される VPC。

vpc01 ▾

VPC を使用するには、関連付けられたサブネットグループが必要です。

サブネットグループ

エンドポイントに関連付けるサブネット。

cluster-subnet-group-1 ▾

VPC セキュリティグループ - オプション

エンドポイントに関連付けるセキュリティグループ。

セキュリティグループを選択 ▾

default



sg-0eff21fe889480ff8

セキュリティグループの設定

インバウンドルール

| セキュリティグループルール ID | タイプ | プロトコル | ポート範囲 | ソース | 説明 - オプション | 削除 |
|-----------------------|----------|-------|-------|------|----------------|----|
| sgr-0a76336a1cd779375 | Redshift | TCP | 5439 | カスタム | Q_ 0.0.0.0/0 X | |

ルールを追加

Kinesis

2023年8月30日 15:05

ストリームデータ処理用の分析システムやアプリケーションを構築するサービス

IoTデータ → **Kinesis Streams** → Spark Streaming → アプリケーション

ストリームデータを収集・処理するためのフルマネージド型サービスで4つのサービスで構成される

- Amazon Kinesis Data Streams : ストリームデータを処理する
- Amazon Kinesis Data Firehose : ストリームデータをS3やRedshiftなどへ簡単に配信
- Amazon Kinesis Data Analytics : ストリームデータを標準的なSQLクエリでリアルタイムに可視化・分析

連携するサービス

| | |
|-----------------------|---|
| Kinesis Data Streams | Lambda EC2 EMR Kinesis Data Firehose/Analytics |
| Kinesis Data Firehose | S3 Redshift OpenSearch |

アプリケーションの構築

次の関連機能を活用してストリーミング処理アプリケーションを構築する

Kinesis Agent : Kinesisサービスにデータを簡単に収集して取り込むOSSのスタンダロンJavaアプリケーション

Kinesis Producer Library : Kinesis Streamsにデータを送信するOSSの補助ライブラリ

Fluent plugin for Amazon Kinesis : StreamsとFirehoseにイベントを送信するOSSのFluentd出力プラグイン

Kinesis Data Generator(KDG) : StreamsまたはFirehoseにテストデータを簡単に送信できる

Kinesis Client Library(KCL) : KCLを利用してKinesisアプリケーションを作成する。OSSのクライアントライブラリで、EC2インスタンスなどにデプロイして利用する。ワーカーがシャード数に応じて、レコードプロセッサのライフサイクル管理を実施

スケーリング

Kinesisのスケーリングでは、リシャーディングによりシャード数を増加させる。

| | |
|------------|--|
| リシャーディング | 分割：シャード数を増加することでパフォーマンスを向上させる 結合：シャード数を減少させることでコストを削減する 1 シャードに対して1インスタンスまで対応できる |
| 拡張ファンアウト機能 | スループット専用のコンシューマーの開発機能 コンシューマーは、シャードあたり1秒間に最大2MBのデータのスループットで、ストリームからレコードを受け取ることができる。 |

Kinesis Data Streams

2023年8月30日 15:13

ストリームデータ処理用の分析システムやアプリケーションを構築するサービス

- ・ログとデータフィードの取り込みと処理の高速化
- ・リアルタイムのデータ分析とレポート作成
- ・ウェブサイトのクリックストリームをリアルタイムで分析するなどのリアルタイムデータ分析
- ・データストリームの有向非循環グラフ (DAG) 作成などの複雑なストリーム処理
- ・データのリアルタイム集計

ストリーミング処理をシャードに分けて分散させて実行するため高速処理が可能

以下の要素で成り立っている。シャード単位でパフォーマンスを向上させる

シャード：データストリームの基本的なスループットの単位

レコード：データストリームに保存されるデータの単位レコードはシーケンス番号、パーティションキー、データのBLOBで構成

データBLOB：データプロデューサーがデータストリームに追加する、処理対象のデータ

パーティションキー：レコードを分離してデータストリームの異なるシャードにルーティングするために使用

シーケンス番号：各レコードの一意の識別子

特徴

- ・ミリ秒単位でリアルタイム処理
- ・フルマネージド型のサーバレス機能
- ・一度取り込まれたデータは削除不可
- ・データの保有期間はデフォルト24時間、最大8760時間
- ・シャード単位でデータ転送料金が発生

キャパシティモードの設定

オンデマンドモード：キャパシティが予想できないときに使用。自動でスケーリング

プロビジョニングモード：事前予想可能なときに使用。予測した処理量に基づいてデータストリームのシャード数を指定

※24時間ごとに2回切り替え可能

Amazon Kinesis > データストリーム > データストリームの作成

データストリームを作成 情報

データストリームの設定

データストリーム名

lab2-streams

使用できる文字は英字(大文字と小文字)、数字、アンダースコア、ハイフン、ピリオドです。

データストリームの容量 情報

容量モード

オンデマンド

このモードは、データストリームのスループット要件が予測不能で可変である場合に使用します。オンデマンドモードでは、データストリームの容量が自動的にスケールします。

プロビジョンド

データストリームのスループット要件を確実に推定できる場合は、プロビジョンドモードを使用します。プロビジョンドモードでは、データストリームの容量が固定されます。

データストリームの合計容量

デフォルトでは、オンデマンドモードのデータストリームがスループットを自動的にスケールして、書き込み容量として最大で 200 MiB/秒および 200,000 レコード/秒のトラフィックに対応します。トラフィックが容量を超えると、データストリームがスロットルされます。書き込みを 1 GB/秒、読み取りを 2 GB/秒まで容量の増加をリクエストするには、[サポートチケットを送信](#) をクリックします。

書き込み容量

最大

200 MiB/秒、200,000 レコード/秒

読み込み容量

最大 (コンシューマーあたり)

400 MiB/秒

最大 2 つのデフォルトコンシューマー。より多くのコンシューマーには、拡張ファンアウト (EFO) を使用します。EFO は、それ専用のスループットを持つ最大 20 のコンシューマーの追加をサポートします。

① オンデマンドモードには、スループットあたりの料金モデルがあります。次を参照してください [オンデマンドモードの Kinesis の料金](#)

データストリーム設定

データストリームを作成後と有効なステータスになったら、設定を編集できます。

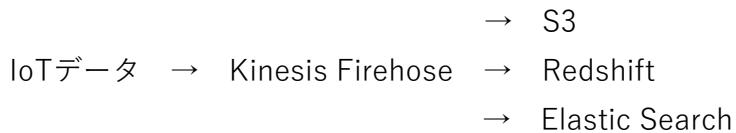
| 設定 | 値 | 作成後に編集可能 |
|-----------------|--------|--|
| 容量モード | オンデマンド | <input checked="" type="checkbox"/> はい |
| データ保持期間 | 1 日 | <input checked="" type="checkbox"/> はい |
| サーバー側の暗号化 | 無効 | <input checked="" type="checkbox"/> はい |
| 拡張メトリクスのモニタリング中 | 無効 | <input checked="" type="checkbox"/> はい |
| タグ | - | <input checked="" type="checkbox"/> はい |

Kinesis Data Firehose

2023年8月30日 15:31

ストリームデータを各種DBに配信するためのサービス

Lambdaと連携してETLとしても機能する



特徴

- ・最小60秒間隔で受信するストリーミングデータを特定のサイズにバッファし送信先に配信
- ・フルマネージド型のサーバレス機能
- ・Lambda関数を利用したカスタムのデータ変換が可能
- ・失敗したデータやバックアップをS3バケットに保存

StreamsとFirehoseの違い

| | Kinesis Data Streams | Kinesis Data Firehose |
|-------|------------------------------------|-----------------------|
| 実行内容 | 高速なデータ処理 データ分析・集計 複雑なストリーム処理 | データ形式の変換 データの配信 |
| 処理速度 | ミリ秒単位のリアルタイム | 60秒間隔のバッチ |
| データ保存 | 24時間～365日 | データ保存不可（ストリーム配信自体には） |

リアルタイムか60秒間隔か

SQSにはコマンド(命令)を投入し、Kinesisにはデータを投入するという使い分けをする。

2023年8月30日 16:09

Amazon Kinesis > Data Firehose > 配信ストリームを作成

配信ストリームを作成 情報

▶ Amazon Kinesis Data Firehose: 仕組み

ソースと送信先を選択

配信ストリームのソースと送信先を指定します。配信ストリームの作成後に、配信ストリームのソースと送信先を変更することはできません。

ソース 情報

Amazon Kinesis Data Streams

送信先 情報

Amazon S3

ソースの設定

Kinesis データストリーム

arn:aws:kinesis:ap-northeast-1:608728620263:stream/lab2-streams

参照

作成 

形式: arn:aws:kinesis:[Region]:[AccountId]:stream/[StreamName]

配信ストリーム名

配信ストリーム名

lab2-firehose

使用できる文字は、大文字と小文字、数字、アンダースコア、ハイフン、ピリオドです。

レコードを変換および転換 - オプション

レコードデータを変換および転換するように Kinesis Data Firehose を設定します。

AWS Lambda でソースレコードを変換 | 情報

Kinesis Data Firehose は AWS Lambda 関数を呼び出して、ソースデータレコードを転換、フィルタリング、圧縮解除、および処理できます。指定された AWS Lambda 関数を使用して、指定された送信先に配信する前に、受信ソースデータの動的パーティショニングキーを提供することもできます。

データ変換を有効にする

レコード形式を転換 | 情報

Apache Parquet または Apache ORC 形式のデータは、通常、JSON よりも効率的にクエリを実行できます。Kinesis Data Firehose は、[AWS Glue](#) で定義されているテーブルのスキーマを使用して、JSON 形式のソースレコードを転換できます。JSON 形式ではないレコードについては、上記の「AWS Lambda でソースレコードを変換」セクションで、JSON に転換する Lambda 関数を作成します。

レコード形式の変換を有効にする

送信先の設定 | 情報

配信ストリームの送信先の設定を指定します。

S3 バケット

s3://lab2-s3

参照

作成

形式: s3://bucket

動的パーティショニング | 情報

動的パーティショニングでは、パーティショニングキーに基づいてストリーミング S3 データをパーティショニングすることにより、ターゲットデータセットを作成できます。インライン解析や指定された AWS Lambda 関数を使用して、ソースデータをパーティショニングできます。動的パーティショニングは、新しい配信ストリームを作成する場合にのみ有効にできます。既存の配信ストリーム用に動的パーティショニングを有効にすることはできません。動的パーティショニングを有効にすると、パーティショニングされたデータの GiB あたりの追加コストが発生します。詳細については、「[Kinesis Data Firehose の料金](#)」を参照してください。

有効ではありません

有効

S3 バケットプレフィックス - オプション

デフォルトでは、Kinesis Data Firehose は Amazon S3 に配信するデータにプレフィックス「YYYY/MM/dd/HH」(UTC) を追加します。このデフォルトを上書きするには、実行時に評価される式を含むカスタムプレフィックスを指定します。

プレフィックスを入力

S3 バケットプレフィックスで同じキーを繰り返すことができます。S3 バケットプレフィックスの最大文字数: 1024。

S3 バケットエラー出力プレフィックス - オプション

エラーが発生している状況で使用される S3 バケットエラー出力プレフィックスを指定できます。このプレフィックスには、Kinesis Data Firehose が実行時に評価する式を含めることができます。

プレフィックスを入力

Kinesis Data Analytics

2023年8月30日 15:42

(ストリーミングリソース)

Kinesis Firehose →

Kinesis Analytics

Kinesis Streams →

(ストリーミングデスティネーション)

→Kinesis Firehose

→Kinesis Streams

Amazon Kinesis > ストリーミングアプリケーション > ストリーミングアプリケーションの作成

ストリーミングアプリケーションの作成 情報

Kinesis Data Analytics は、接続されたストリーミングソースからのデータをリアルタイムで継続的に読み取り、分析します。Kinesis Data Analytics のリソースは AWS 無料利用枠の対象外であり、使用量ベースの料金が適用されます。詳細については、[Kinesis Data Analytics の料金](#) を参照してください。

Apache Flink の設定

ランタイム

アプリケーションを作成した後、ランタイム環境のタイプまたはバージョンを変更することはできません。

Apache Flink - ストリーミングアプリケーション

Apache Flink は、無制限および制限のある両方のデータストリームでステートフルな計算を行う、オープンソースフレームワークの分散処理エンジンです。このオプションを使用して、Java、Scala、および Python で Apache Flink を使用してストリーミングアプリケーションを構築します。また、Apache Beam を使用すると Java ベースのストリーミングアプリケーションを構築することもできます。Apache Beam はオープンソースの統合モデルであり、各言語専用の一連の SDK によりデータ処理ワークフローを定義して実行できます。

Apache Flink バージョン

Apache Flink 1.15 (推奨)



ⓘ Python Kinesis Data Analytics アプリケーションを実行するには、アプリケーションの作成後にアプリケーションのプロパティを設定してコードファイルを指定します。 [Learn more](#)

アプリケーション設定

アプリケーション名

lab2-analytics

使用できる文字は、大文字と小文字、数字、アンダースコア、ハイフン、ピリオドです。

説明 - オプション

lab2-analytics

アプリケーションリソースへのアクセス

必要なアクセス権限を持つ IAM ロールを作成または選択します。 [Learn more](#)

- 必要なポリシーを使用して IAM ロール **kinesis-analytics-lab2-analytics-ap-northeast-1** を作成/更新します
- Kinesis Data Analytics が設定できる IAM ロールから選択

タグ - オプション

タグは、AWS リソースに割り当てるラベルです。各タグはキーとオプションの値で構成されています。タグを使用して、リソースの検索およびフィルタリング、または AWS コストの追跡を行うことができます。 [Learn more](#)

このアプリケーションに関連付けられたタグはありません。

[タグを追加](#)

50 個まで tags を追加できます。

アプリケーション設定のテンプレート

ユースケースに合わせてサンプルテンプレートを選択します。すべての設定は、アプリケーション作成後に編集できます。

テンプレート

開発

コストを最低限に抑えるには、これらの設定を使用します。

本番稼働用

高い可用性の高速で一貫したパフォーマンスを実現するには、これらの設定を使用します。

開発テンプレート

| 設定 | 値 |
|---|-------------|
| スナップショット | オフ |
| Amazon CloudWatch Logs によるログ記録中 | INFO |
| Amazon CloudWatch を使用したメトリクスレベルのモニタリング中 | Application |
| 並列度 | 1 |
| KPUあたりの並列度 | 1 |
| 自動スケーリング | オン |

キャンセル

ストリーミングアプリケーションの作成

実践

2023年8月30日 16:49

kinesis.py(dataの情報を送る)

```
import boto3
import datetime as dt
import json
import uuid

STREAM = 'lab1-stream'

kinesis = boto3.client('kinesis', region_name='ap-northeast-1')

data = {'message':'testmsg1','timestamp':str(dt.datetime.now())}

response = kinesis.put_record(
    Data=json.dumps(data),
    PartitionKey=str(uuid.uuid1()),
    StreamName=STREAM
)

print(response)
```

実行コマンド

```
$ sudo python kinesis.py
→モジュール(boto3)が足りない

# sudo su - ← ルートユーザーで操作

pip install boto3

#exit

$ sudo python kinesis.py
```

トラブルシューティング

Firehoseに割り当たっているロールにs3・kinesis streamに対してのポリシーが必要
S3用のポリシー

```
{  
    "Sid": "",  
    "Effect": "Allow",  
    "Action": [  
        "s3:AbortMultipartUpload",  
        "s3:GetBucketLocation",  
        "s3:GetObject",  
        "s3>ListBucket",  
        "s3>ListBucketMultipartUploads",  
        "s3:PutObject"  
    ],  
    "Resource": [  
        "arn:aws:s3:::lab2-s3",  
        "arn:aws:s3:::lab2-s3/*"  
    ]  
},
```

```
{  
    "Sid": "",  
    "Effect": "Allow",  
    "Action": [  
        "kinesis:DescribeStream",  
        "kinesis:GetShardIterator",  
        "kinesis:GetRecords",  
        "kinesis>ListShards"  
    ],  
    "Resource": "arn:aws:kinesis:ap-northeast-1:608728620263:stream/lab2-  
streams"  
},
```

トラブルシューティング

2023年8月30日 17:15

CloudTrail

2023年8月30日 17:01

どのユーザーがどの操作をしたのかを確認できる

検索方法

時系列で検索

ルックアップ属性を指定して絞り込みができる

athenaで検索

AWS Config

2023年8月30日 17:05

リソースに対してどのような処理がされたかを確認できる

The screenshot shows the AWS Config Resource Inventory interface. At the top, there are three dropdown filters: 'リソースカテゴリ' (AWS リソース), 'リソースタイプ' (Multiple selected), and 'コンプライアンス' (準拠). Below these are search fields for 'リソース識別子 - オプション' and a checkbox for '削除されたリソースを含める'. The main area displays a table of resources:

| リソース識別子 | タイプ | コンプライアンス |
|---------------------|--------------|----------|
| i-015bf3f80c4f4ba53 | EC2 Instance | 準拠 |
| i-04db5bb3efba6ffc | EC2 Instance | 準拠 |

ルールを追加

ルールに違反しているリソースを検索できる

自動修復を設定できる

GuardDuty

2023年9月5日 17:56

アクセスログ

2023年8月31日 17:34

ALB

ロードバランサーの属性の設定から

モニタリング

● アクセスログ
アクセスログは、Elastic Load Balancer に対して行われたすべてのリクエストの詳細なログを提供します。既存の S3 の場所を選択してください。プレフィックスを指定しない場合、アクセスログはバケットのルートに保存されます。これには追加料金が適用されます。 詳細はこちら

S3 URI

s3://lab2-alblog

X 表示 S3 を参照

s3バケットにアクセスポリシーを追加する

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::582318560864:root"
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::lab2-alblog/*"
    }
  ]
}
```

| リージョン | リージョン名 | Elastic Load Balancing アカウント ID |
|----------------|------------------|---------------------------------|
| us-east-1 | 米国東部 (バージニア北部) | 127311923021 |
| eu-central-1 | 欧州 (フランクフルト) | 054676820928 |
| ap-northeast-1 | アジアパシフィック (東京) | 582318560864 |
| ap-southeast-2 | アジアパシフィック (シドニー) | 783225319266 |
| ap-south-1 | アジアパシフィック (ムンバイ) | 718504428378 |
| cn-north-1* | 中国 (北京) | 638102146993 |

athenaを利用してアクセスログを確認する

https://docs.aws.amazon.com/ja_jp/AmazonS3/latest/userguide/using-s3-access-logs-to-identify-requests.html

| テーブルのプロパティ | | |
|------------------------|---|--|
| テーブル名 lab1_alb_logs | データベース名 default | データソース名 AwsDataCatalog |
| テーブルの説明 - | 作成時刻 2023-08-31T17:31:50.000+09:00 | データソースタイプ AWS Glue データカタログ |
| | 最終アクセス時刻 1970-01-01T09:00:00.000+09:00 | 更新時刻 2023-08-31T17:31:50.000+09:00 |
| | 場所 s3://ahaws-lab1-web/alblog/AWSLogs/608728620263/elasticloadbalancing/ap-northeast-1/ | シリアル化ライブラリ org.apache.hadoop.hive.serde2.RegexSerDe |
| | 入力形式 org.apache.hadoop.mapred.TextInputFormat | 暗号化されたデータあり false |
| | 出力形式 org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat | |

API Gateway のログ

ロールを作成する

ユースケース

EC2、Lambda、その他の AWS のサービスがこのアカウントでアクションを実行することを許可します。

一般的なユースケース

- EC2
Allows EC2 instances to call AWS services on your behalf.
- Lambda
Allows Lambda functions to call AWS services on your behalf.

他の AWS のサービスのユースケース:

- API Gateway

▼
- API Gateway
Allows API Gateway to push logs to CloudWatch Logs.

ロールのarnを貼り付ける

設定

アカウント内の CloudWatch ログに対して書き込み権限を持つ Identity and Access Management (IAM) ロールの ARN を追加します。

CloudWatch ログのロール ARN*

アカウントレベルのスロットリング 現在のアカウントレベルのスロットリングレートは、1秒あたりのリクエスト数が **10000** で、バーストのリクエスト数は **5000** です。 !

* 必須

ロググループをつくる

ステージのロギングおよびトレース設定を指定します。

CloudWatch 設定 [詳細はこちら](#)

CloudWatch ログ ロギングが無効

詳細 CloudWatch メトリクスを有効化 ?

カスタムアクセスのログ記録

アクセスログの有効化

Access Log Destination ARN ?

ログの形式 ?

入力の例:

KMS

2023年8月30日 17:16

鍵を管理するサービス

「KMS」 → 「カスタマー管理型のキー」
キーを作成する

CloudFront経由でs3にあるindex.htmlにアクセスする

CloudFront作成

オリジン

オリジンドメイン
AWS オリジンを選択するか、お使いのオリジンのドメイン名を入力します。

オリジンパス - オプション [情報](#)

オリジンリクエストのオリジンドメイン名に追加する URL パスを入力します。

名前

このオリジンの名前を入力します。

「コントロール設定を作成」で作成

オリジンアクセス [情報](#)

Public Bucket must allow public access.

Origin access control settings (recommended) Bucket can restrict access to only CloudFront.

Legacy access identities Use a CloudFront origin access identity (OAI) to access the S3 bucket.

Origin access control

Select an existing origin access control (recommended) or create a new configuration.

バケットポリシー

Policy must allow access to CloudFront IAM service principal role.

ポリシーを手動で更新する

⚠️ S3 バケットポリシーを更新する必要があります

CloudFront は、ディストリビューションの作成後にポリシーステートメントを提供します。

カスタムヘッダーを追加 - オプション

CloudFront は、オリジンに送信するすべてのリクエストにこのヘッダーを含めます。

デフォルトのキャッシュビヘイビア

パスパターン [情報](#)

デフォルト (*)

オブジェクトを自動的に圧縮 [情報](#)

No

Yes

ビューワー

ビューワープロトコルポリシー

HTTP and HTTPS

Redirect HTTP to HTTPS

HTTPS only

許可された HTTP メソッド

GET, HEAD

GET, HEAD, OPTIONS

GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE

ビューワーのアクセスを制限する

ビューワーのアクセスを制限する場合、ビューワーがコンテンツにアクセスするには CloudFront 著名付き URL または著名付き cookie を使用する必要があります。

No

Yes

デフォルトルートオブジェクト・オプション
ピューワーが特定のオブジェクトの代わりにルート URL ([I](#)) を要求したときに返されるオブジェクト（ファイル名）。

標準ログ記録

Amazon S3 パケットに配信されたピューワークエストのログを取得します。

- オフ
 オン

- IPv6
 オフ
 オン

説明 - オプション

キャンセル

ディストリビューションを作成

オリジンを編集からポリシーをコピーする

① このポリシーステートメントを使用して CloudFront へのアクセスを許可する必要があります。S3 バケットにアクセスするアクセス許可を CloudFront に付与する方法 [\[?\]](#) の詳細をご覧ください。

[\[?\] ポリシーをコピー](#)

[\[?\] S3 バケットアクセス許可に移動 \[\\[?\\]\]\(#\)](#)

S3バケットにポリシーを追加する

「S3」 → 「アクセス許可」 → 「バケットポリシー」

S3オブジェクトにキーを登録する

サーバー側の暗号化

サーバー側の暗号化は、保管時のデータを保護します。

暗号化設定

- デフォルトの暗号化のバケット設定を使用する
 デフォルトの暗号化のバケット設定を上書きする

暗号化タイプ [\[情報\]](#)

- Amazon S3 メッセージキーを使用したサーバー側の暗号化 (SSE-S3)
 AWS Key Management Service キーを使用したサーバー側の暗号化 (SSE-KMS)
 AWS Key Management Service キーを使用したデュアルレイヤーサーバー側の暗号化 (DSS-E-KMS)
2つの異なる暗号化レイヤーでオブジェクトを保護します。料金の詳細については、[Amazon S3 の料金](#) [\[?\]](#) ページの [ストレージ] タブの DSS-E-KMS の料金をご覧ください。

AWS KMS キー [\[情報\]](#)

- AWS KMS キーから選択する
 AWS KMS キー ARN を入力する

AWS KMS キー ARN

[X](#)

[\[?\] KMS キーを作成する \[\\[?\\]\]\(#\)](#)

形式 (キー ID を使用): arn:aws:kms:<region><account-ID>/key/<key-id>

または (アリスを使用): arn:aws:kms:<region><account-ID>/alias/<alias-name>

バケットキー

SSE-KMS に S3 バケットキーを使用すると、AWS KMS への呼び出しを減らすことで暗号化コストを削減できます。DSS-E-KMS では S3 バケットキーはサポートされていません。[\[詳細\] \[\\[?\\]\]\(#\)](#)

- 無効にする

- 有効にする

指定されたオブジェクト

< 1 >

| 名前 | ▲ | タイプ | ▼ | 最終更新日時 | ▼ | サイズ | ▼ | ストレージクラス | ▼ |
|----------------------------|---|------|---|----------------------------|---|-------|---|----------|---|
| index.html | | html | | 2023/08/30 05:30:41 PM JST | | 3.0 B | | スタンダード | |

アクセスする

→クラウドフロントがキーを使用できるように設定する

KMSからキーを追加

https://docs.aws.amazon.com/ja_jp/AmazonCloudFront/latest/DeveloperGuide/private-content-restricting-access-to-s3.html

「例 CloudFront OAC による SSE-KMS の KMS キーへのアクセスを許可する KMS キーポリシーステートメント」

```
{  
    "Version": "2012-10-17",  
    "Id": "key-consolepolicy-3",  
    "Statement": [  
        {  
            "Sid": "Enable IAM User Permissions",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "arn:aws:iam::608728620263:root"  
            },  
            "Action": "kms:*",  
            "Resource": "*"  
        },  
        {  
            "Sid": "AllowCloudFrontServicePrincipalsSSE-KMS",  
            "Effect": "Allow",  
            "Principal": {  
                "AWS": "arn:aws:iam::<アカウントID>:root",  
                "Service": "cloudfront.amazonaws.com"  
            },  
            "Action": [  
                "kms:Decrypt",  
                "kms:Encrypt",  
                "kms:GenerateDataKey"  
            ],  
            "Resource": "*",  
            "Condition": {  
                "StringEquals": {  
                    "AWS:SourceArn": "arn:aws:cloudfront::<アカウントID>:distribution/<ディストリビューション名>"  
                }  
            }  
        }  
    ]  
}
```


EventBridge

2023年8月30日 18:12

ルールの詳細を定義 情報

ルールの詳細

名前

lab2

数字、小文字/大文字、. (ピリオド)、- (ハイフン)、_ (アンダーバー) を含め、最大 64 文字まで使用できます。

説明 - オプション

説明を入力

イベントバス 情報

このルールを適用するイベントバス (デフォルトのイベントバス、カスタムイベントバス、パートナーアイベントバスのいずれか) を選択します。

default

選択したイベントバスでルールを有効にする

ルールタイプ 情報

イベントパターンを持つルール

定義したイベントパターンとイベントが一致したときに実行されるルール。EventBridge は指定されたターゲットにイベントを送信します。

スケジュール

スケジュールに従って実行されるルール

キャンセル

次へ

イベントパターンを構築

情報

イベントソース

イベントソース

イベントの送信元となるイベントソースを選択します。

- AWS イベントまたは EventBridge パートナーイベント
AWS のサービスまたは EventBridge パートナーから送信されたイベント。

- その他
複数のソースから送信されたカスタムイベントまたはイベント (AWS のサービスやパートナーからのイベントなど)。

- すべてのイベント
アカウントに送信したすべてのイベント。

サンプルイベント - オプション

サンプルイベントの選択または入力は必要ありませんが、イベントパターンまたはフィルター基準を作成およびテストするときに参照できるように、サンプルイベントを選択または入力することをお勧めします。

イベントパターンを書き込むときにサンプルイベントを参照することも、サンプルイベントを使用してイベントパターンと一致するかどうかをテストすることもできます。サンプルイベントを検索するか、独自のイベントを入力するか、以下のサンプルイベントを編集してください。必須フィールドの詳細については、[サンプルイベントをご覧ください](#)。 

サンプルイベントタイプ

- AWS イベント

- EventBridge パートナーイベント

- ご自身名前を入力

サンプルイベント

イベントソースとタイプ、またはキーワードでフィルタリングします。

選択



イベントパターン 情報

イベントソース
ソースとして AWS のサービスまたは EventBridge パートナー

AWS のサービス

AWS のサービス
イベントソースとしての AWS のサービスの名前

EC2

イベントタイプ
一致パターンのソースとしてのイベントのタイプ

EC2 Instance State-change Notification

Event Type Specification 1

- 任意の状態
 特定の状態

特定の状態

running

イベントパターン
イベントパターン、またはイベントと照合するフィルター

```
1 {
2   "source": ["aws.ec2"],
3   "detail-type": ["EC2 Instance State-change Notification"]
4   "detail": {
5     "state": ["running"],
6     "instance-id": ["i-0e02dca1ed7c22a4d"]
7   }
8 }
```

コピーテストパターン

パターンを編集

Event Type Specification 2

- 任意のインスタンス
 個別のインスタンス ID

個別のインスタンス ID

i-0e02dca1ed7c22a4d

削除

追加

SNS作成

トピックの作成

詳細

タイプ 情報

トピックの作成後にトピックタイプを変更することはできません

- FIFO (先入れ先出し、先出し)

- 堆積に保存されたメッセージの順序付け
- 1回のみメッセージ配信
- 高スループット、最大 300 件のパブリッシュ/サブスクリプションプロトコル-SQS

- スタンダード

- ベストエフォート型メッセージの順序付け
- 少なくとも 1 回のメッセージ配信
- 1 秒あたりのパブリッシュ/スレーブット
- サブスクリプションプロトコル-SQS, Lambda, HTTP, SMS, メール、モバイルアプリケーションエンドポイント

名前

lab2

最大文字数は 256 文字です。英数字、ハイフン (-)、およびアンダースコア (_) を含めることができます。

表示名 - オプション 情報

SMS のサブスクリプションでこのトピックを使用するには、表示名を入力します。SMS メッセージには最初の 10 文字のみが表示されます。

lab2

最大 100 文字。

詳細

トピック ARN
am:aws:sns:ap-northeast-1:608728620263:lab2

プロトコル
サブスクリプションのエンドポイントのタイプ
Eメール

エンドポイント
Amazon SNS から通知を受信できる E メールアドレス。
osuke_oyaizu@gmail.com

① サブスクリプションを作成した後、それを確認する必要があります。 [情報](#)

▶ サブスクリプションフィルターポリシー - オプション [情報](#)
このポリシーで受信者が受け取るメッセージがフィルターされます。

▶ Redrive ポリシー (デッドレターキュー) - オプション [情報](#)
配信不能メッセージをデッドレターキューに送信します。

ターゲット 1

ターゲットタイプ

EventBridge イベントバス、EventBridge API の宛先 (SaaS パートナー)、または別の AWS のサービスをターゲットとして選択します。

- EventBridge イベントバス
- EventBridge API の宛先
- AWS のサービス

ターゲットを選択 [情報](#)

イベントがイベントパターンと一致したとき、またはスケジュールがトリガーされたときに呼び出すターゲットを選択します (1 レールごとに 5 個のターゲットに制限されます)。

SNS トピック

トピック
lab2

▶ 追加設定

スケジュール方式

ルールの詳細

名前

rule-name

数字、小文字/大文字、.(ピリオド)、-(ハイフン)、_(アンダーバー)を含め、最大 64 文字まで使用できます。

説明 - オプション

説明を入力

イベントバス [情報](#)

このルールを適用するイベントバス(デフォルトのイベントバス、カスタムイベントバス、パートナーアイベントバスのいずれか)を選択します。

default ▾

選択したイベントバスでルールを有効にする

ルールタイプ [情報](#)

イベントパターンを持つルール

定義したイベントパターンとイベントが一致したときに実行されるルール。EventBridge は指定されたターゲットにイベントを送信します。

スケジュール

スケジュールに従って実行されるルール

EventBridge Scheduler - AWS の新しいスケジューリング機能! [新規](#)

イベントバスやルールにかかわらず、1回限りの定期的なスケジューリング機能を提供する、新しい EventBridge スケジューリング機能。Lambda 関数などのターゲットを呼び出すスケジュールを作成できます。

[詳細はこちら](#) 

スケジュール名と説明

スケジュール名

lab2

文字、数字、ダッシュ、ドット、またはアンダースコアのみを使用してください。最大 64 文字。

説明 - オプション

lab2

最大文字数は 512 文字です。

スケジュールグループ

各スケジュールはスケジュールグループに配置する必要があります。デフォルトでは、スケジュールは「デフォルト」グループに配置されます。[独自のスケジュールグループを作成](#)することもできます。タグは、スケジュールグループにのみ追加でき、スケジュールには追加できません。

default



スケジュールのパターン

頻度 | [情報](#)

1 回限りのスケジュールまたは定期的なスケジュールを定義できます。

1 回限りのスケジュール

定期的なスケジュール

日付と時刻

ターゲットを起動する日時。

2023/08/30



12:30

(UTC+09:00) Asia/Tokyo



YYYY/MM/DD

24 時間形式のタイムスタンプ (hh:mm) を使
用

タイムゾーン

フレックストライムウィンドウ

フレックストライムウィンドウを選択した場合、スケジューラは指定した時間枠内でスケジュールを呼び出します。例えば、15 分を選択した場合、スケジュールはスケジュールの開始時刻から 15 分以内に実行されます。

選択



平日の 1 時に実行したい場合

スケジュールのパターン

頻度 | 情報

1回限りのスケジュールまたは定期的なスケジュールを定義できます。

1回限りのスケジュール

定期的なスケジュール

スケジュールの種類

ニーズに最適なスケジュールの種類を選択します。

cron ベースのスケジュール

特定の時刻 (毎月第1月曜日の午前8時 (PST) など) に実行される cron 式を使用したスケジュールのセット。

rate ベースのスケジュール

規則的なレートで実行されるスケジュール (10分ごとなど)。

cron 式 | 情報

スケジュールの cron 式を定義

cron (

)

分

時間

日付

月

曜日

 コピー  クリア

次の 10 個のトリガー日

日時は、現在のタイムゾーンで UTC 形式で表示されます。例:
「Wed, Nov 9, 2022 09:00 (UTC - 08:00)」(太平洋時間の場合)

Thu, 31 Aug 2023 01:00:00 (UTC+09:00)

Fri, 01 Sep 2023 01:00:00 (UTC+09:00)

Mon, 04 Sep 2023 01:00:00 (UTC+09:00)

Tue, 05 Sep 2023 01:00:00 (UTC+09:00)

Wed, 06 Sep 2023 01:00:00 (UTC+09:00)

Thu, 07 Sep 2023 01:00:00 (UTC+09:00)

Fri, 08 Sep 2023 01:00:00 (UTC+09:00)

Mon, 11 Sep 2023 01:00:00 (UTC+09:00)

Tue, 12 Sep 2023 01:00:00 (UTC+09:00)

Wed, 13 Sep 2023 01:00:00 (UTC+09:00)

フレックストライムウィンドウ

フレックストライムウィンドウを選択した場合、スケジューラは指定した時間枠内でスケジュールを呼び出します。例えば、15分を選択した場合、スケジュールはスケジュールの開始時刻から15分以内に実行されます。

選択



スケジュール名と説明

スケジュール名

lab2

文字、数字、ダッシュ、ドット、またはアンダースコアのみを使用してください。最大 64 文字。

説明 - オプション

lab2

最大文字数は 512 文字です。

スケジュールグループ

各スケジュールはスケジュールグループに配置する必要があります。デフォルトでは、スケジュールは「デフォルト」グループに配置されます。[独自のスケジュールグループを作成](#)することもできます。タグは、スケジュールグループにのみ追加でき、スケジュールには追加できません。

default



スケジュールのパターン

頻度 | [情報](#)

1回限りのスケジュールまたは定期的なスケジュールを定義できます。

1回限りのスケジュール

定期的なスケジュール

日付と時刻

ターゲットを起動する日時。

2023/08/30



12:30

(UTC+09:00) Asia/Tokyo



YYYY/MM/DD

24 時間形式のタイムスタンプ (hh:mm) を使用
用

タイムゾーン

フレックストライムウインドウ

フレックストライムウインドウを選択した場合、スケジューラは指定した時間枠内でスケジュールを呼び出します。例えば、15 分を選択した場合、スケジュールはスケジュールの開始時刻から 15 分以内に実行されます。

オフ



時間になったらメールを送るようにする

ターゲットの詳細

ターゲット API | [情報](#)

スケジュールのターゲットとして呼び出す API を選択してください。

テンプレート化されたターゲット

すべての API



CodeBuild
StartBuild



CodePipeline
StartPipelineExecut...



Amazon ECS
RunTask



Amazon EventBridge
PutEvents



Kinesis Data Firehose
PutRecord



Amazon Inspector V1
StartAssessmentRun



Kinesis Data Streams
PutRecord



AWS Lambda
Invoke



SageMaker
StartPipelineExecut...



AWS Step Functions
StartExecution



Amazon SNS
Publish

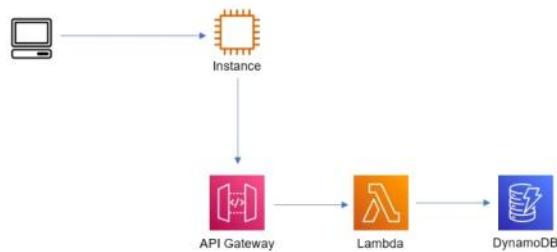


Amazon SQS
SendMessage

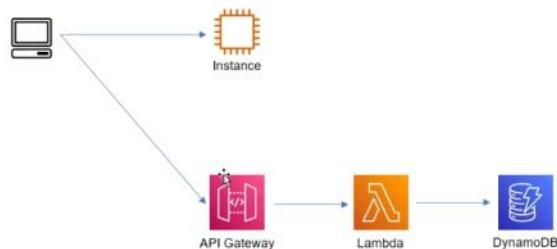
CORS

2023年8月31日 15:46

サーバサイド



クライアントサイド



クライアントサイド

https://toyotajp.sharepoint.com/sites/msteams_c3fcad/SitePages/Web_CORS.aspx

クロスオリジン

サーバからベースのHTMLをもらい、クライアント自身がJavaScriptによって処理して表示する

CORSの有効化設定

[設定前]

test page

| | |
|------|--------|
| text | Submit |
|------|--------|

DevTools is now available in Japanese!

Always match Chrome's language Switch DevTools to Japanese Don't show again

Elements Console Sources Network Performance

Preserve log Disable cache No throttling Invert Hide data URLs

All Fetch/XHR JS CSS Img Media Font Doc WS Wasm Manifest Other Has blocked cookies

Blocked Requests 3rd-party requests

Timing: 50 ms | 100 ms | 150 ms | 200 ms | 250 ms | 300 ms

| Name | Headers | Preview | Response | Initiator | Timing |
|-------------------|------------------|---------|----------|-----------|--------|
| bootstrap.min.css | General | | | | |
| script.js | Response Headers | | | | |
| lab2 | Raw | | | | |

2 / 5 requests | 26.5 kB / 29.3 kB t

Console What's New Issues

Default levels ▾ 1 Issue: 1

Access to fetch at 'https://dmlho1go66.execute-api.ap-northeast-1.amazonaws.com/lab2' from origin 'http://1ab2-kadai-alb-1096227687.ap-northeast-1.elb.amazonaws.com' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource. If an opaque response serves your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled.

GET https://dmlho1go66.execute-api.ap-northeast-1.amazonaws.com/lab2 (index):22 net::ERR_FAILED 200 (OK)

Uncaught (in promise) TypeError: Failed to fetch at getdata ((index):22:2) at (index):51:5

APIGateway・Lambdaで設定する必要がある

APIGatewayの設定

The screenshot shows the AWS API Gateway Resource settings for a specific endpoint. The left sidebar lists methods: GET, OPTIONS, and POST. The 'OPTIONS' method is selected. The main panel shows CORS configuration for the 'OPTIONS' method. It specifies 'CORS の有効化' (Enable CORS) and 'リソースのアクション' (Resource Actions). The 'OPTIONS' method is highlighted in green, indicating it is being edited. The configuration shows '認可なし' (No Authorization) and 'API キー不要' (No API Key Required).

Allow-Originにオリジンであるオリジン(ALBのURL)を貼り付ける

<http://lab2-kadai-alb-1096227687.ap-northeast-1.elb.amazonaws.com/lab2>

※オリジンはパスは関数ない

<http://lab2-kadai-alb-1096227687.ap-northeast-1.elb.amazonaws.com/> ←最後の/を消す

CORS の有効化

lab2-pub-api API のゲートウェイレスポンス DEFAULT 4XX DEFAULT 5XX [?](#)

メソッド GET OPTIONS POST [?](#)

Access-Control-Allow-Methods GET, OPTIONS, POST [?](#)

Access-Control-Allow-Headers 'Content-Type,X-Amz-Date,Authorization' [?](#)

Access-Control-Allow-Origin* p-northeast-1.elb.amazonaws.com [?](#)

▶ アドバンスト

Lambda関数設定

Pythonコードのreturnにhedaer情報を追加する

```
import json  
import os  
import boto3  
import uuid
```

```

REGION = os.environ['REGION']
TABLE = os.environ['TABLE']

dynamo = boto3.resource('dynamodb', region_name=REGION)
table = dynamo.Table(TABLE)

def lambda_handler(event, context):

    if event['httpMethod'] == 'GET':
        data = table.scan()['Items']
        print(data)
    elif event['httpMethod'] == 'POST':
        postData = json.loads(event['body'])
        postData['id'] = str(uuid.uuid1())
        table.put_item(Item = postData)
        data = {'message': 'successfull'}

    return {
        'statusCode': 200,
        'headers': [
            'Access-Control-Allow-Headers': 'Content-Type',
            'Access-Control-Allow-Origin': "http://lab2-kadai-alb-1096227687.ap-northeast-1.elb.amazonaws.com",
            'Access-Control-Allow-Methods': 'GET',
        ],
        'body': json.dumps(data)
    }

```

S3の場合

https://docs.aws.amazon.com/ja_jp/AmazonS3/latest/userguide/ManageCorsUsing.html

Python

2023年8月31日 17:05

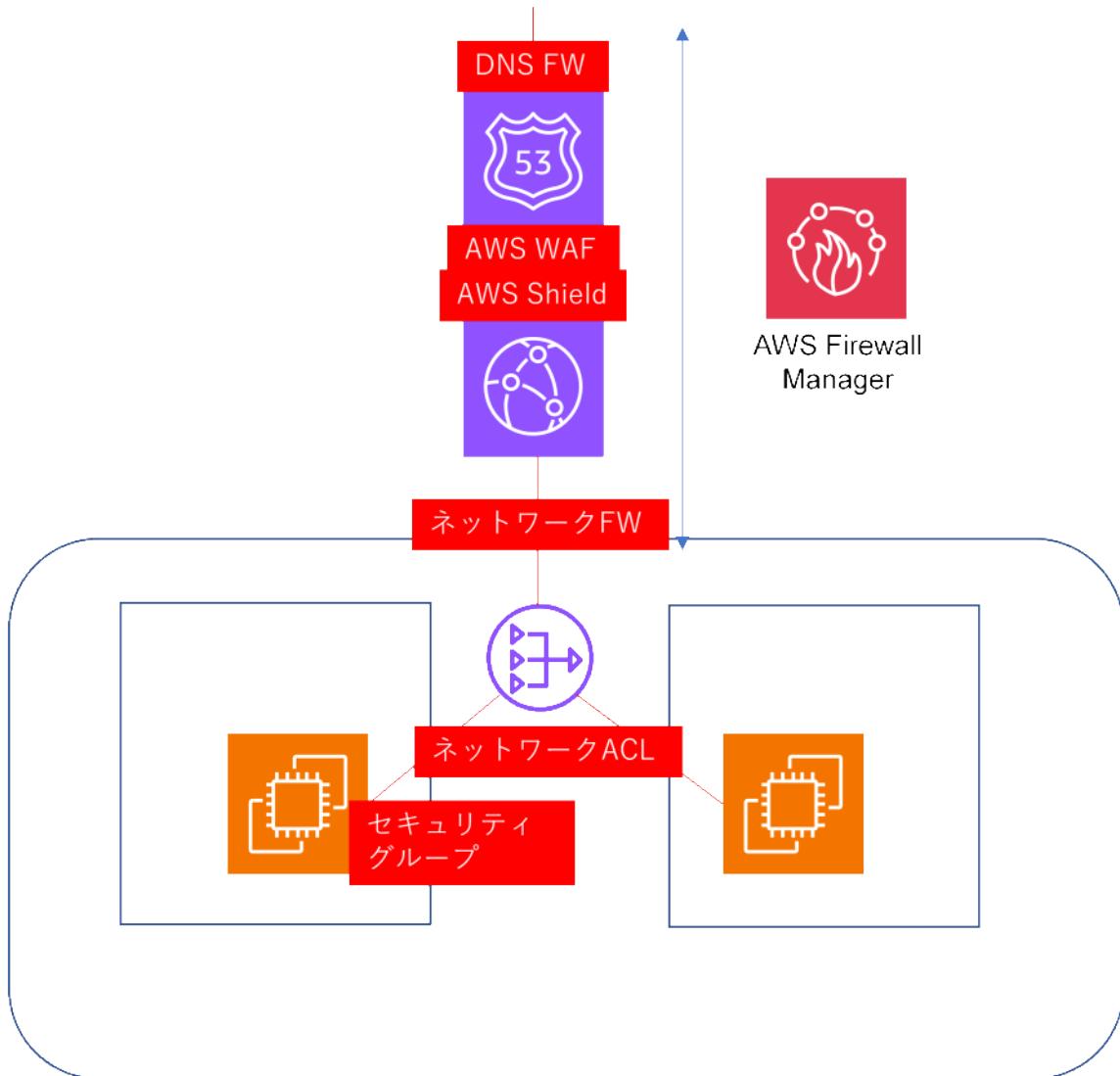
モジュールをインストールしても使えない

ルートユーザーでもインストールしておくと使えるようになる場合がある

[requirements.txt] ← 依存関係を保存するファイル

```
blinker==1.6.2
certifi==2023.7.22
charset-normalizer==3.2.0
click==8.1.7
Flask==2.3.3
idna==3.4
importlib-metadata==6.8.0
itsdangerous==2.1.2
Jinja2==3.1.2
MarkupSafe==2.1.3
requests==2.31.0
urllib3==2.0.4
Werkzeug==2.3.7
zipp==3.16.2
```

pip install -r requirements.txt ←必要なモジュールをファイルからインストールする



セキュリティグループ：インスタンスレベルのトラフィック制御

ネットワークACL：サブネットレベルのトラフィック制御 拒否制御ができる

ネットワークFW：VPC全体を管理するFW

AWS WAF：パブリッククラウド型のFW

ELBやCloudFrontの前後に設置して不正アクセスを保護する

AWS Shield：DDoS攻撃に対してAWSで実行しているアプリケーションを保護する

DNS FW：DNSレベルの保護にはRoute53 DNSファイアウォールを利用して、悪質なDNSアクセスを防ぐことが可能

AWS Firewall Manager：複数アカウントの様々なファイアウォール機能を一元管理する

セキュリティグループは許可のみ

特定のアドレスからを拒否することはできない

セキュリティまとめ

■セキュリティグループ

- 用途) インスタンス等、個々の通信制限
- 注意) 特定の送信元からの通信を遮断できない
→ルールは許可する通信しか指定できない

■ネットワークACL

- 用途) サブネットレベル（複数のインスタンス）への通信制限
- 注意) ステートレスのため、戻りパケットの許可が必要

■Network Firewall

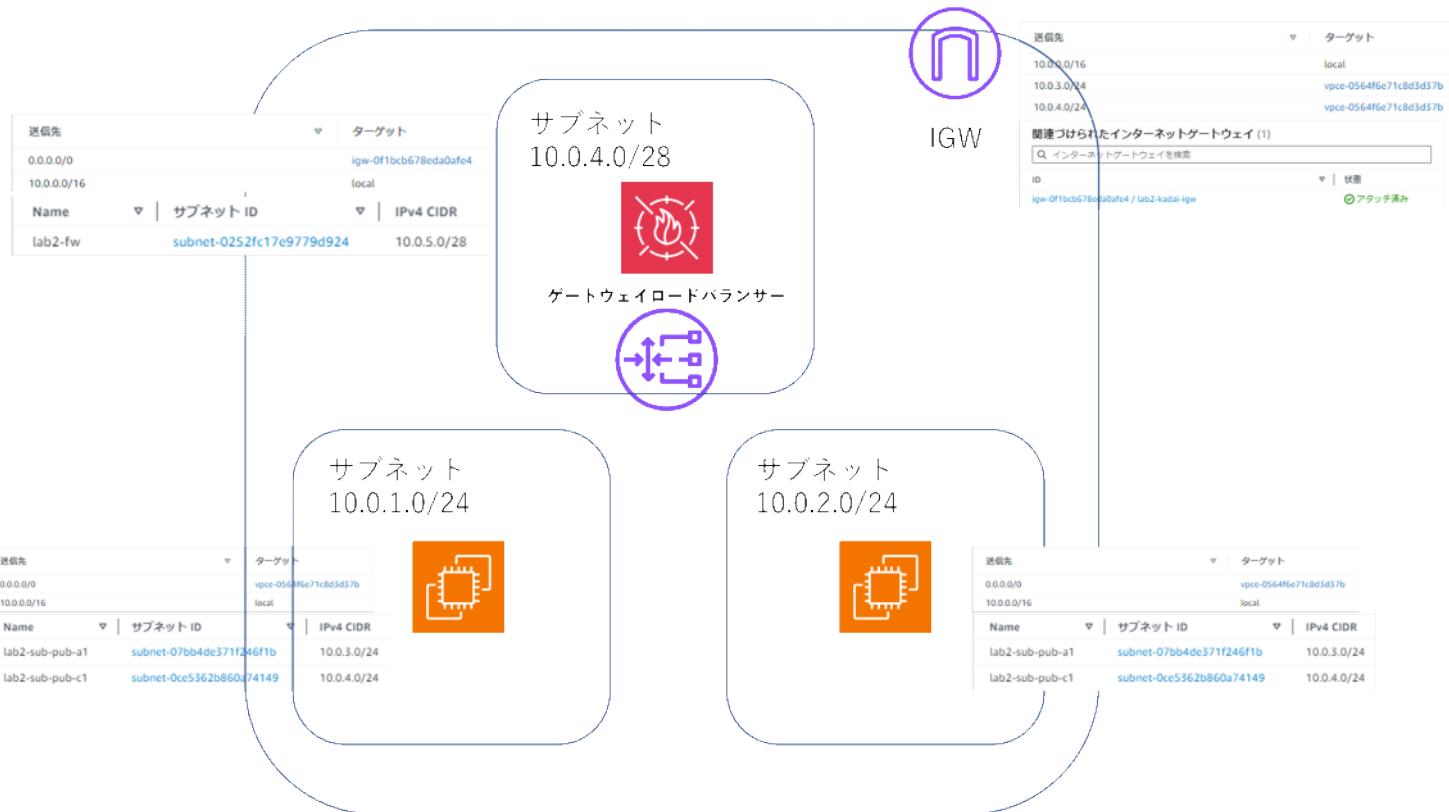
- 用途) 複数のサブネットに対して、一元的に通信を制限
特定のプロトコル（ポートに頼らない）やドメインに対して制限が可能
- 注意) 通信経路（往復分）にFirewallを経由させる様にルート登録が必要
ステートレス→ステートフルでルールを評価

■WAF

- 用途) Webアプリケーションを保護
SQLインジェクションやDOS攻撃に対して
- 注意) 適用できるのは、ALB・CloudFront・API Gatewayのみ（学習した範囲のリソースでは）

Network Firewall

2023年9月1日 15:40



- ①ファイアウォール用のサブネット作成
- ②ファイアウォール作成・・・ゲートウェイロードバランサーも同時に作成される
- ③ステートレスルール・ステートフルルールを作成
- ④ルートの編集

ファイアウォール用のサブネットを作成する

10.0.5.0/28 とする

ファイアウォール作成

サブネットは①のサブネット

VPC
保護が必要なアベイラビリティーゾーンごとに、ファイアウォールエンドポイント専用のパブリックサブネットをネットワークファイアウォールに提供します。ここで指定したファイアウォールサブネットのみファイアウォールに使用できます。他の目的には使用しないでください。

VPC
このファイアウォールを作成する VPC を選択します。

lab2-vpc

ファイアウォールサブネット
各サブネットには、使用可能な IP アドレスが 1 つ必要です。作成した後にサブネットの IP アドレスタイプを変更することはできません。

| | | |
|-----------------|---------------------|------------|
| アベイラビリティーゾーン | サブネット | IP アドレスタイプ |
| ap-northeast-1a | subnet-0252fc17e... | IPv4 |

新しいサブネットを追加

関連付けられたファイアウォールポリシー

ファイアウォールポリシーには、ファイアウォールがウェブトラフィックを検査および管理する方法を定義するルールグループのリストが含まれています。ファイアウォールを作成した後で、関連するファイアウォールポリシーを設定できます。

ファイアウォールポリシー

新しいファイアウォールポリシーには、ファイアウォールがウェブトラフィックを検査および管理する方法を定義するルールグループのリストが含まれています。ファイアウォールを作成した後で、関連するファイアウォールポリシーを設定できます。

- 空のファイアウォールポリシーを作成して関連付ける
- 既存のファイアウォールポリシーを関連付ける

新しいファイアウォールポリシー名

ファイアウォールポリシーの一意の名前を入力します。

lab2-firewall-policy

名前は 1~128 文字にする必要があります。有効な文字は a~z, A~Z, 0~9, - (ハイフン) です。名前の先頭と末尾にハイフンを使用することはできません。また、ハイフンを 2 つ連続して含めることはできません。

説明 - オプション

説明には 0~256 文字を使用できます。

ファイアウォールポリシーの説明を入力してください

ステートレス

ステートレスルールグループ (1)

| 名前 | 状態 | カバーティー |
|-----------|----|--------|
| stateless | 正常 | 100 |

ルールの追加

ルールグループに必要なステートレスルールを追加します。追加したルールは、下のルールの一覧の箇にリストされています。

| | |
|---|--|
| ルール | 1 |
| ルール詳細 | |
| ルールグループに必要なルールが最初に表示されます。ルールグループ内のルールには、一箇の規則が表示されます。 | |
| 1 | |
| プロトコル | 複数するトランスポートプロトコル。 |
| オプションを選択 | <input type="button" value="All protocols"/> |
| 送信元 | 複数する送信元アドレスとアドレス範囲です。單一のアドレスと CIDR ブロックを指定できます。 |
| 任意の IPv4 アドレス | 0.0.0.0/0 |
| 1 行につき 1 つの値を入力し、IPv4 または IPv6 のアドレスか範囲を指定します。複数を一緒に選択することはできません。 | |
| 送信元ポート範囲 | 複数する送信元ポートとポート範囲です。これは TCP および UDP プロトコルにのみ適用されます。 |
| 任意のポート | 0-65535 |
| 1 行につき 1 つの値を入力し、範囲を一緒に選択することはできません。 | |
| 受信元 | 複数する受信元アドレスとアドレス範囲です。單一のアドレスと CIDR ブロックを指定できます。 |
| 任意の IPv4 アドレス | 0.0.0.0/0 |
| 1 行につき 1 つの値を入力し、IPv4 または IPv6 のアドレスか範囲を指定します。複数を一緒に選択することはできません。 | |
| 受信元ポート範囲 | 複数する受信元ポートとポート範囲です。これは TCP および UDP プロトコルにのみ適用されます。 |
| 任意のポート | 0-65535 |
| 1 行につき 1 つの値を入力し、範囲を一緒に選択することはできません。 | |

パス：許可

ドロップ：拒否

TCP フラグ - オプション

複数する TCP フラグマスク。複数しない場合、これはありゆるフラグと一致します。この段階で TCP ポートにのみ使用されます。

マスク - オプション

複数する TCP フラグ。複数の場合はすべてのフラグを複数するには、オプションを選択してください。

オプションを選択

フラグ - オプション

[マスク] フラグと組み合わせて使用して、パケットが一致するために必要なフラグと設定しておならぬフラグを定義します。この段階では、[マスク] 設定でも記述されている除外を指定できます。

オプションを選択

アクション

アクション

ルールが 1 回ずつパケットをファイアウォールで処理する方法を選択します。

- [パス]
- ドロップ
- ステートフルルールグループに転送

カスタムアクション - オプション

標準のルールアクションに加えて、CloudWatch メトリクスを実行するカスタムアクションを追加します。ここで定義したカスタムアクションは、このルールグループの他のルールで使用できます。

ルールの追加

ステートフルルールグループ

ステートフルルールグループ (0)

| 名前 | 状態 | カバーティー | マターホースト ID |
|----------|-----|--------|----------------|
| stateful | 未登録 | 100 | マターホースト ID を選択 |

| | |
|--------------------------|---------|
| ルールを編集 | アクション ▾ |
| ステートフルルールグループ | |
| マターホーストとステートフルルールグループの追加 | |
| マターホーストとステートフルルールグループを選択 | |
| ルールグループの選択 | |

| ドメインおよびIP ルールグループ (4) | | | |
|-------------------------------------|--|---|---|
| < 1 > ⌂ | | | |
| | 名前 | キャバシティー | アラートモードで実行しますか? |
| <input checked="" type="checkbox"/> | AbusedLegitBotNetCommandAndControlDomainsStrictOrder | Contains rules that allow you to block requests to a class of domains which are generally legitimate but are compromised and may host botnets. This can help reduce the risk of resources accessing botnets originating from these sources with poor reputation. | 200 <input checked="" type="radio"/> 無効 |
| <input checked="" type="checkbox"/> | BotNetCommandAndControlDomainsStrictOrder | Contains rules that allow you to block requests to domains that are known for hosting botnets. This can help reduce the risk of resources accessing botnets originating from these known sources. | 200 <input checked="" type="radio"/> 無効 |
| <input checked="" type="checkbox"/> | MalwareDomainsStrictOrder | Contains rules that allow you to block requests to domains that are known for hosting malware. This can help reduce the risk of receiving malware or viruses originating from these known sources. | 200 <input checked="" type="radio"/> 無効 |
| <input checked="" type="checkbox"/> | AbusedLegitMalwareDomainsStrictOrder | Contains rules that allow you to block requests to a class of domains which are generally legitimate but are compromised and may host malware. This can help reduce the risk of receiving malware or viruses originating from these sources with poor reputation. | 200 <input checked="" type="radio"/> 無効 |

| ステートフルルールグループ (4) | | | |
|--------------------------|--|---------|-------|
| 優先度を編集 ▲ アクション ▲ | | | |
| | 名前 | キャバシティー | |
| <input type="checkbox"/> | 1 AbusedLegitBotNetCommandAndControlDomainsStrictOrder | 200 | |
| <input type="checkbox"/> | 2 BotNetCommandAndControlDomainsStrictOrder | 200 | はい 無効 |
| <input type="checkbox"/> | 3 MalwareDomainsStrictOrder | 200 | はい 無効 |
| <input type="checkbox"/> | 4 AbusedLegitMalwareDomainsStrictOrder | 200 | はい 無効 |

ルールの追加 情報
ルールグループに必要なステートフルなルールを追加します。追加した各ルールは、下のルールの一覧表示されています。

プロトコル
検査するトランスポートプロトコル。

| | | |
|----|---|--------------------------------------|
| IP | 送信元 検査する送信元 IP アドレスとアドレス範囲です。単一のアドレスと CIDR ブロックを指定できます。 ANY | 送信元ポート 検査する送信元ポートまたはポート範囲。 ANY |
| | ANY | ANY |

サポートされているポートは 0~65535 です。

送信先
検査する送信先 IP アドレスとアドレス範囲です。単一のアドレスと CIDR ブロックを指定できます。
ANY

| | |
|-----|--------------------------------------|
| ANY | 送信先ポート 検査する送信先ポートまたはポート範囲。 ANY |
| ANY | ANY |

サポートされているポートは 0~65535 です。

ルートの設定

- ファイアウォール用のルート作成
パブリックサブネットをターゲット : ゲートウェイロードバランサーのエンドポイント

ト

パブリックサブネットにアクセスが来た場合はファイアウォールに飛ばす
Edgeの関連付けでigwからきたアクセスをルートに適用する

| 送信先 | ターゲット |
|-------------|------------------------|
| 10.0.0.0/16 | local |
| 10.0.3.0/24 | vpce-0564f6e71c8d3d37b |
| 10.0.4.0/24 | vpce-0564f6e71c8d3d37b |

ルートを追加

「Edgeの関連付け」からインターネットゲートウェイを選択する

| ルート | サブネットの関連付け | Edge の関連付け | ルート伝播 | タグ |
|---|--------------|------------------------------|---------------------|---------------|
| 関連づけられたインターネットゲートウェイ (1) | | | | |
| ID igw-0f1bc678eda0afe4 / lab2-kadai-igw | 状態 アタッチ済み | VPC vpc-0b1a904e87d31f1ca | 所有者 608728620263 | Edge の関連付けを編集 |

Edge の関連付けを編集 (1/1)

ルートテーブルの基本的な詳細

| | | |
|-------------------------------------|---------------------------|---|
| ルートテーブル ID rtb-0241c9bca33706fea | ルートテーブル名 lab2-from-igw | ルートテーブル VPC ID vpc-0b1a904e87d31f1ca |
|-------------------------------------|---------------------------|---|

インターネットゲートウェイ

ゲートウェイ ID
igw-0f1bc678eda0afe4 / lab2-kadai-igw [\[編集\]](#)

状態
Attached

Owner or ASN (Amazon side)
608728620263

・パブリック用のルートテーブル

firewallを経由してインターネットゲートウェイに行くように設定する

| 送信先 | ターゲット |
|---|--|
| 10.0.0.0/16 | <input type="text" value="local"/> [編集] |
| <input type="text" value="0.0.0.0/0"/> [編集] | <input type="text" value="vpce-0564f6e71c8d3d37b"/> [編集] |

[\[ルートを追加\]](#)

・ファイアウォール用(インターネットゲートウェイにアクセスしにいく)のルートテーブル

| 送信先 | ターゲット |
|---|--|
| 10.0.0.0/16 | <input type="text" value="local"/> [編集] |
| <input type="text" value="0.0.0.0/0"/> [編集] | <input type="text" value="igw-0f1bc678eda0afe4"/> [編集] |

[\[ルートを追加\]](#)

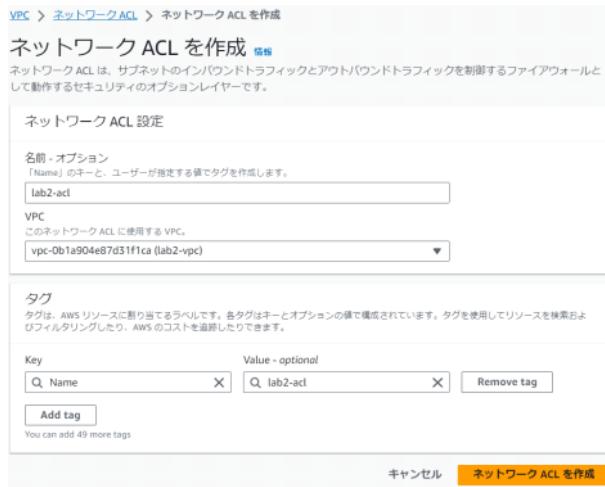
ファイアウォール用のサブネットを関連付ける

| ルート | サブネットの関連付け | Edge の関連付け | ルート伝播 | タグ |
|--------------------|--------------------------------------|--------------------------|-------|----|
| 明示的なサブネットの関連付け (1) | | | | |
| Name lab2-fw | サブネット ID subnet-0252fc17e9779d924 | IPv4 CIDR 10.0.5.0/28 | | |

ネットワークACL

2023年9月4日 8:32

①ネットワークACLの作成



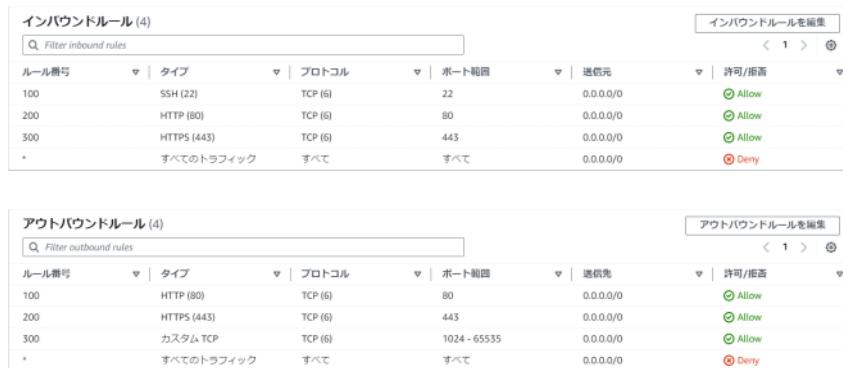
The screenshot shows the 'Create Network ACL' wizard step 1 of 3. It's titled 'Network ACL Settings'. The 'Name - Options' section has 'Name' set to 'lab2-acl'. The 'VPC' section shows 'VPC' selected and 'vpc-0b1a904e87d31f1ca (lab2-vpc)' chosen. The 'Tags' section shows a single tag 'Name: lab2-acl'. At the bottom right is a yellow 'Create Network ACL' button.

ネットワークACL・・・サブネット全体に対してのACL

→ サブネット内にあるインスタンスに対して一括でセキュリティを適用できる

②インバウンドルール・アウトバウンドルールを設定する

SSH・HTTP・HTTPSを許可する



The screenshot shows two tables of security rules. The top table is 'Inbound Rules (4)' and the bottom is 'Outbound Rules (4)'. Both tables have columns for Rule ID, Type, Protocol, Port Range, Source, and Action (Allow or Deny). The inbound rules allow SSH (22), HTTP (80), HTTPS (443), and all traffic from 0.0.0.0/0. The outbound rules allow HTTP (80), HTTPS (443), custom TCP (1024-65535), and all traffic to 0.0.0.0/0.

| ルール番号 | タイプ | プロトコル | ポート範囲 | 送信元 | 許可/拒否 |
|-------|-------------|---------|-------|-----------|-------|
| 100 | SSH (22) | TCP (6) | 22 | 0.0.0.0/0 | Allow |
| 200 | HTTP (80) | TCP (6) | 80 | 0.0.0.0/0 | Allow |
| 300 | HTTPS (443) | TCP (6) | 443 | 0.0.0.0/0 | Allow |
| * | すべてのトラフィック | すべて | すべて | 0.0.0.0/0 | Deny |

| ルール番号 | タイプ | プロトコル | ポート範囲 | 送信先 | 許可/拒否 |
|-------|-------------|---------|--------------|-----------|-------|
| 100 | HTTP (80) | TCP (6) | 80 | 0.0.0.0/0 | Allow |
| 200 | HTTPS (443) | TCP (6) | 443 | 0.0.0.0/0 | Allow |
| 300 | カスタム TCP | TCP (6) | 1024 - 65535 | 0.0.0.0/0 | Allow |
| * | すべてのトラフィック | すべて | すべて | 0.0.0.0/0 | Deny |

SSHはインバウンドとアウトバウンドでポートが変わる

アウトバウンドにはエフェメラル(Ephemeral)ポートを指定

セキュリティグループとの違い

<https://tenshoku-careerchange.jp/column/1123/>

<https://dev.classmethod.jp/articles/amazon-vpc-acl/>

・ネットワークACL

ネットワークACLはステートレスです。この場合のステートレスとは、出の通信と入りの通信（戻ってくる通信）は別々に評価されます。つまり、出の通信は許可するが、戻ってくる入りの通信は拒否するということが可能になります。

・セキュリティグループ

セキュリティグループはステートフルであるため、対応する反対の通信は自動で許可される。よって反対の向きの設定は不要。

ステートフルとは状態（ステート）を記憶しているということ。出て行ったパケットの情報を記憶しているため、対する戻りは明示的に許可されていなくても受け入れる。

| | ネットワークACL | セキュリティグループ |
|---------------|-------------|------------|
| 設定対象 | サブネット単位 | インスタンス単位 |
| 設定ルール | 許可ルールと拒否ルール | 許可ルールのみ |
| 設定方向 | 入りと出 | 入りと出 |
| ステートフル/ステートレス | ステートレス | ステートフル |
| 評価順 | 順番に評価される | 全て評価される |

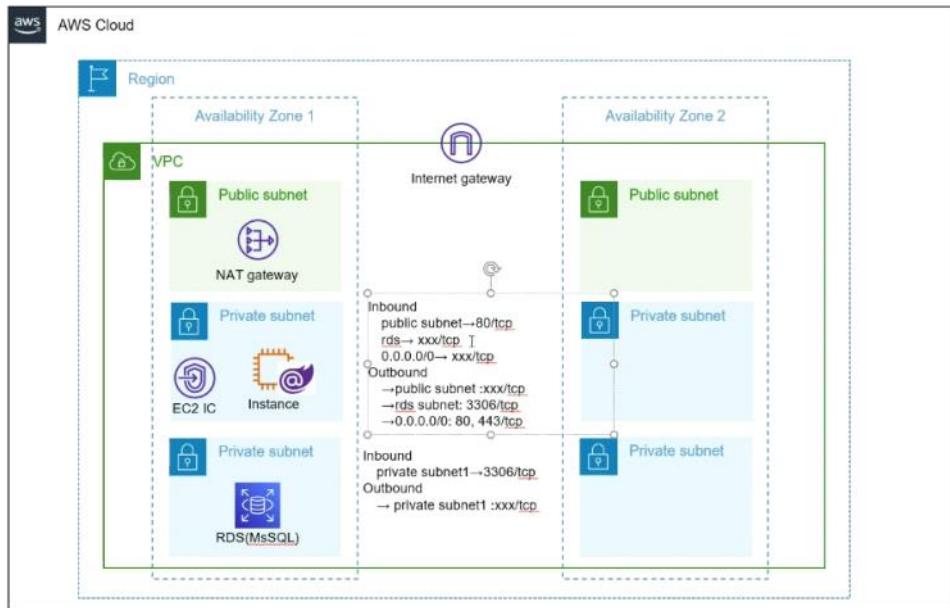
③サブネットを関連付ける

今回はパブリックを関連付ける

サブネットの関連付け (2)

サブネットの関連付け

| サブネットの関連付け (2) | | | | |
|---|--------------------------|----------------------------------|-----------------|-------------|
| <input type="text"/> Filter subnet associations | | | | |
| 名前 | サブネット ID | 次と関連付け: | アベイラビリティゾーン | IPv4 CIDR |
| lab2-sub-pub-a1 | subnet-07bb4de371f246... | acl-071ce55b224967be8 / lab2-acl | ap-northeast-1a | 10.0.3.0/24 |
| lab2-sub-pub-c1 | subnet-0ce5362b860a74... | acl-071ce55b224967be8 / lab2-acl | ap-northeast-1c | 10.0.4.0/24 |



WAF

2023年9月5日 16:52

WAF(web application firewall) . . . webサイトを含むウェブアプリケーションを守る

課題6の構成から

GETメソッドの認可を外す

インターネットにアクセスできるインスタンスからcurl <APIのURL> → 確認できる

WAFを作成

リージョンを選択する

Web ACL details

Name
lab2-webacl
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Description - optional
The description can have 1-256 characters.

CloudWatch metric name
lab2-webacl
The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Resource type
Choose the type of resource to associate with this web ACL.
 Amazon CloudFront distributions
 Regional resources (Application Load Balancers, Amazon API Gateway REST APIs, Amazon App Runner services, AWS AppSync GraphQL APIs and Amazon Cognito user pools)

Region
Choose the AWS region to create this web ACL in.
Asia Pacific (Tokyo)

SQLインジェクション攻撃対策

Rules (0)
If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.

Edit Delete Add rules ▲
Add managed rule groups
Name Action
Add my own rules and rule groups

No rules.
You don't have any rules added.

▼ AWS managed rule groups

| SQL database | 200 | Add to web ACL |
|---|-----|--|
| Contains rules that allow you to block request patterns associated with exploitation of SQL databases, like SQL injection attacks. This can help prevent remote injection of unauthorized queries. Learn More  | |  |

Rules (1)

If a request matches a rule, take the corresponding action. The rules are prioritized in order they appear.

[Edit](#)[Delete](#)[Add rules ▾](#)

| <input type="checkbox"/> | Name | Capacity | Action |
|--------------------------|--------------------------------|----------|------------------|
| <input type="checkbox"/> | AWS-AWSManagedRulesSQLiRuleSet | 200 | Use rule actions |

Web ACL capacity units (WCUs) used by your rules

The total WCUs for a web ACL can't exceed 5000. Using over 1500 WCUs affects your costs. [AWS WAF Pricing](#)

[200/5000 WCUs](#)

Default web ACL action for requests that don't match any rules

Default action

- Allow
- Block
- ▶ Custom request - optional

インジェクション攻撃のURL

<https://xo3b30zvvi.execute-api.ap-northeast-1.amazonaws.com/lab2?id=1> OR 1=1

APIゲートウェイに設定する

設定 ログ/トレース ステージ変数 SDK の生成 エクスポート デプロイ履歴 ドキュメント履歴 Canary

キャッシュ設定

API キャッシュを有効化

デフォルトのメソッドスロットリング

このステージのメソッド用のデフォルトのスロットリングレベルを選択します。このステージの各メソッドでは、これらのレートおよびバーストの設定を優先します。現在のアカウントレベルのスロットリングレートは、1秒あたりのリクエスト数が 10000 で、バーストのリクエスト数が 5000 です。 [API Gateway のスロットリングの詳細](#)

スロットリングの有効化

レート: 10000 リクエスト数/秒

バースト: 5000 リクエスト数

ウェブアプリケーションファイアウォール (WAF) [詳細はごちら。](#)

このステージに適用されるウェブ ACL を選択します。

ウェブ ACL: lab2-webacl (wafv2) [ウェブ ACLを作成する](#)

クライアント証明書

このステージの統合エンドポイントを呼び出すには、API ゲートウェイが使用するクライアント証明書を選択してください。

証明書: なし

[変更を保存](#)

特定のIPアドレスからのアクセスを拒否する

IPsetsを作成する

NATゲートウェイを使用している場合はNATゲートウェイに割り当てているElastic IPを指定する

[AWS WAF > IP sets](#)

| IP sets Info | | Asia Pacific (Tokyo) ▾ | Copy ARN | Delete | Create IP set |
|-----------------------------------|-------------|--|--------------------------|------------------------|-------------------------------|
| <input type="text"/> Find IP sets | | ◀ 1 ▶ ⚙️ | | | |
| Name | Description | ▼ | ID | | |
| | | | | | |

IP set details

IP set name

lab2

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

Description - *optional*

The description can have 1-256 characters.

Region

Choose the AWS region to create this IP set in.

Asia Pacific (Tokyo)



IP version

IPv4

IPv6

IP addresses

52.198.16.61/32

ルールを追加する

Rules (2)

Edit

Delete

Add rules ▲

Find rules

Add managed rule groups

Add my own rules and rule groups

| <input type="checkbox"/> | Name | Action | Priority | Custom response |
|--------------------------|--------------------------------|------------------|----------|-----------------|
| <input type="checkbox"/> | AWS-AWSManagedRulesSQLiRuleSet | Use rule actions | 0 | - |
| <input type="checkbox"/> | lab2-iprule | Block | 1 | - |

Rule type

Rule type

IP set

Use IP sets to identify a specific list of IP addresses.

Rule builder

Use a custom rule to inspect for patterns including query strings, headers, countries, and rate limit violations.

Rule group

Use a rule group to combine rules into a single logical set.

Rule

Name

lab2-ec2-deny

The name must have 1-128 characters. Valid characters: A-Z, a-z, 0-9, - (hyphen), and _ (underscore).

IP set

IP set

lab2-ec2 ▾

IP address to use as the originating address
When a request comes through a CDN or other proxy network, the source IP address identifies the proxy and the original IP address is sent in a header. Use caution with the option, IP address in header, because headers can be handled inconsistently by proxies and they can be modified to bypass inspection.

Source IP address
 IP address in header

Action
Choose an action to take when a request originates from one of the IP addresses in this IP set.

Allow
 Block
 Count
 CAPTCHA
 Challenge

▶ Custom response - optional

Cancel

Add rule

確認

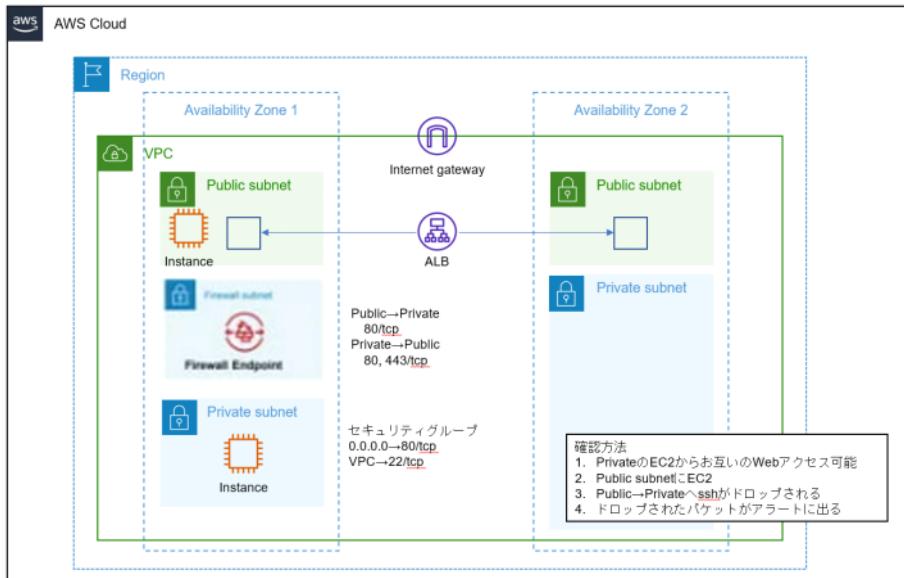
インスタンスからアクセスする

curl <APIのURL> → {"message":"Forbidden"}

CloudFrontに設定する場合はセキュリティ保護を有効化する

確認方法

2023年9月4日 11:36



ファイアウォールのポリシー設定

ステートレスデフォルトアクション

X

フラグメント化されたパケット

- すべてのパケットに同じアクションを使用する
- 完全なパケットとフラグメント化されたパケットに異なるアクションを使用する

完全なパケットに対するデフォルトアクション

ルールアクション

- パス
- ドロップ
- ステートフルルールグループに転送

メトリクスを発行 - オプション

カスタム Amazon CloudWatch メトリクスを発行して、ステートレスルールグループの使用状況をモニタリングします。

- 有効化

フラグメント化されたパケットのデフォルトのアクション

ルールアクション

- パス
- ドロップ
- ステートフルルールグループに転送

メトリクスを発行 - オプション

カスタム Amazon CloudWatch メトリクスを発行して、ステートレスルールグループの使用状況をモニタリングします。

- 有効化

キャンセル

保存

すべてをドロップ・・・HTTP80を許可するというルールだとドロップされる
HTTP80はある程度やり取りされてからHTTP80だとわかるので最初の時点でドロップ
TCP80許可なら最初からTCP80なのでパスする
確立された接続のパケットをドロップ・・・HTTP80でもパスされる 評価されたものを確認する
ドメインリストの場合は使用する

ステートフルなデフォルトアクション

デフォルトのアクション
[ドロップ] アクションは最大で 1 つしか選択できず、[アラート] アクションはいずれかまたは両方を選択できます。

すべてをドロップ
すべてのパケットをドロップします。

確立された接続のパケットをドロップ
確立された接続にあるパケットのみをドロップします。

すべてアラート
すべてのパケットで ALERT_ALL メッセージをログ記録します。

確立された接続のパケットをアラート
確立された接続にあるパケットに ALERT_ESTABLISHED メッセージをログ記録します。

キャンセル 保存

ステートフルルールを作成する

ステートフルルールグループ

名前

ステートフルルールグループ内で一意のルールグループの名前を入力します。

lab2-private-public

名前は 1~128 文字にする必要があります。有効な文字は a~z、A~Z、0~9、- (ハイフン) です。名前の先頭と末尾にハイフンを使用することはできません。また、ハイフンを 2 つ連続して含めることはできません。

説明 - オプション

説明には 0~256 文字を使用できます。

キャバシティ 情報

存続期間中にこのルールグループに含まれることが想定されるルールの数。ルールグループの作成後にキャバシティを変更することはできないため、ルール数が増大する場合に備えて余裕を持たせてください。

100

キャバシティは 1 以上 30,000 未満である必要があります。

お客様のデータはデフォルトで、AWS が所有し管理するキーで暗号化されます。別のキーを選択するには、暗号化設定をカスタマイズしてください。

暗号化設定をカスタマイズする (高度)

ステートフルルールグループのオプション

Standard stateful rule

送信元と宛先 IP アドレスとポート、プロトコル、およびその他のルールオプションを指定します。

Domain list

ドメイン名のリストと、ドメインの 1 つにアクセスしようとするトラフィックに対して実行するアクションを指定します。

Suricata compatible rule

string

Suricata ルール構文を使用して高度なファイアウォールルールを指定します。Suricata は標準のルールベースの言語を含むオープンソースの脅威検出エンジンです。

ルールを編集

X

プロトコル

検査するトランSPORTプロトコル。

TCP



送信元

検査する送信元 IP アドレスとアドレス範囲。

カスタム



10.0.1.0/24

送信元ポート

検査する送信元ポートまたはポート範囲。

任意のポート



ANY



サポートされているポートは 0~65535 です。

送信先

検査する送信先 IP アドレスとアドレス範囲。

すべて



ANY



送信先ポート

検査する送信先ポートまたはポート範囲。

カスタム



80
443



サポートされているポートは 0~65535 です。

キャンセル

保存

ルートテーブルの編集

①パブリックサブネット用のルートテーブル

宛先がプライベートサブネットはゲートウェイロードバランサー
パブリックサブネットを関連付ける

| 送信先 | ▼ | ターゲット | ▼ |
|-------------|---|------------------------|---|
| 0.0.0.0/0 | | igw-0f1bcb678eda0afe4 | |
| 10.0.0.0/16 | | local | |
| 10.0.1.0/24 | | vpce-0431202ac30f23e82 | |
| 10.0.2.0/24 | | vpce-0431202ac30f23e82 | |

②プライベートサブネット用のルートテーブル

宛先がパブリックサブネットはゲートウェイロードバランサー
プライベートサブネットを関連付ける

| 送信先 | ▼ | ターゲット | ▼ |
|-------------|---|------------------------|---|
| 0.0.0.0/0 | | vpce-0431202ac30f23e82 | |
| 10.0.0.0/16 | | local | |
| 10.0.3.0/24 | | vpce-0431202ac30f23e82 | |
| 10.0.4.0/24 | | vpce-0431202ac30f23e82 | |

③ファイアウォール用のルートテーブル

natgatewayに送信する

ファイアウォールを関連付ける

| 送信先 | ▼ | ターゲット | ▼ |
|-------------|---|-----------------------|---|
| 0.0.0.0/0 | | nat-06c4cf1fe18dc6d1d | |
| 10.0.0.0/16 | | local | |

ユーザー権限を使用することができる

The screenshot shows the AWS IAM User Details page for a user named 'user01'. The left sidebar has a search bar and navigation links for Dashboard, Access Management (Groups, Users, Roles, Policies), and IAM Home. The main content area shows the user's ARN (arn:aws:iam::608728620263:user/user01) and creation date (September 05, 2023, 17:57 (UTC+09:00)). It also displays session information: 'Console access via MFA not enabled' (with a warning icon), 'Last console sign-in today' (with a checkmark icon), and two active access keys (AKIAY3OYNTTT7LF44AOS and AKIAY3OYNTTT7LF44AOS). A 'Delete' button is visible in the top right corner.

```
import boto3
dynamodb = boto3.resource('dynamodb', region_name='ap-northeast-1', aws_access_key_id='AKIAY3OYNTTT7LF44AOS', aws_secret_access_key='0FbdD6TlaAxb/pHdzAhpO/6rGDlSOefiAjJkK5E')
table = dynamodb.Table('lab1-kadai5')
data = table.scan()['Items']
print(data)
```

※基本的にはロールで行うのが正しいがエラーなどを確認してトラブルシューティングをする

RDS・KMS

2023年9月6日 8:28

<https://blog.denet.co.jp/rdsrdscode/>

S3静的ウェブホスティング

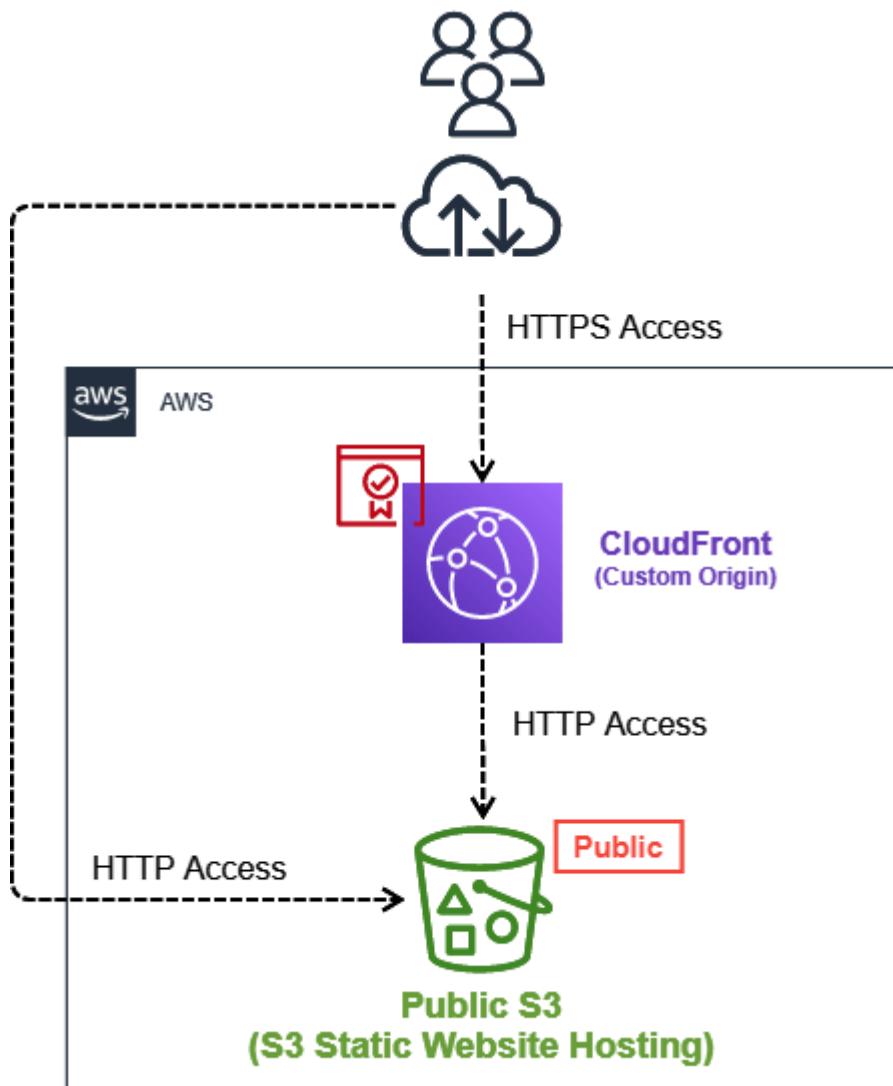
2023年9月6日 8:37

1. S3静的ウェブサイトを使うパターン

はじめの構成は「S3静的ウェブサイト」を使ったパターンです。

S3にはバケットの内容を静的ウェブサイトとしてホストできる[静的ウェブサイトホスティング](#)の機能があります。

この機能ではHTTPのみ利用可能なためHTTPSを使う場合はCloudFrontと組み合わせる必要があります。



ACM手順

2023年9月6日 8:53

発行するにはリージョンをバージニア北部じゃなとだめ(?)

AWS Certificate Manager (ACM) の機能

- ▶ (例) CloudFront ディストリビューションに証明書を設定



証明書をリクエスト

証明書タイプ 情報

ACM 証明書は、インターネットまたは内部ネットワーク内で安全な通信アクセスを確立するために使用できます。acm が提供する証明書のタイプを選択します。

パブリック証明書をリクエスト
Amazon からのパブリック SSL/TLS 証明書をリクエストします。デフォルトでは、パブリック証明書はブラウザとオペレーティングシステムによって信頼されます。

プライベート証明書をリクエスト
発行できるプライベート CA がありません。

プライベート証明書をリクエストするには、プライベート認証機関 (CA) を作成する必要があります。プライベート CA を作成するには、次にアクセスしてください: [AWS Private Certificate Authority](#)

キャンセル 次へ

Route53のドメイン名を追加

ドメイン名

証明書の 1 つ以上のドメイン名を指定します。

完全修飾ドメイン名 情報

lab2.ahaws.toyota-bibliotheca.com

この証明書に別の名前を追加

この証明書にはさらに名前を追加できます。例えば、「www.example.com」の証明書をリクエストする場合、顧客がいざれかの名前でサイトにアクセスできるように、「example.com」という名前を追加できます。

検証方法 情報

ドメインの所有権を検証する方法を選択

DNS 検証 - 推奨
証明書リクエストでドメインの DNS 設定を変更できる場合は、このオプションを選択します。

Eメール検証
証明書リクエストでドメインの DNS 設定を変更する許可がない場合、または当該許可を取得できない場合は、このオプションを選択します

「Route53でレコードを作成」からCNAMEレコードが作成される

発行済みになること・ステータスが成功になっていることを確認する

Route53のレコード作成からエイリアスを設定

レコードをクリック作成

ウィザードに切り替える

▼ レコード 1

削除

レコード名 | [情報](#)

subdomain

ahaws.toyota-bibliotheca.com

ルートドメインのレコードを作成するには、空白のままにします。

レコードタイプ | [情報](#)

A – IPv4 アドレスと一部の AWS リソースにトラフィックをルーティン...

▼

エイリアス

トラフィックのルーティング先 | [情報](#)

CloudFront ディストリビューションへのエイリアス

▼

米国東部 (バージニア北部)

▼

CloudFront ディストリビューションのエイリアスと同じホストゾーン内の別のレコードはグローバルで、米国東部 (バージニア北部) でのみ使用できます。

https://d1vfulg12mrvje.cloudfront.net

X

ルーティングポリシー | [情報](#)

シンプルルーティング

ターゲットのヘルスを評価

いいえ

[別のレコードを追加](#)

ドメイン名でアクセスすると表示される

Aurora

2023年9月6日 9:51

耐障害性 处理速度に優れている

作成方法

エンジンのタイプをAuroraにする その他は普段通りに作成する

エンジンのオプション

エンジンのタイプ [情報](#)

| | | |
|---|---|---|
| <input checked="" type="radio"/> Aurora (MySQL Compatible)  | <input type="radio"/> Aurora (PostgreSQL Compatible)  | <input type="radio"/> MySQL  |
| <input type="radio"/> MariaDB  | <input type="radio"/> PostgreSQL  | <input type="radio"/> Oracle  |
| <input type="radio"/> Microsoft SQL Server  | | |

Writer用とReader用のエンドポイントが存在するので書き込みをする際はWriter用を使用してログインする

[接続とセキュリティ](#) | [モニタリング](#) | [ログとイベント](#) | [設定](#) | [ゼロのETL統合](#) | [メンテナンスとバックアップ](#) | [タグ](#)

Endpoints (2)

Find resources

| エンドポイント名 | ステータス | タイプ | ポート |
|---|-------|--------|------|
| lab2-aurora.cluster-cnkerumhjjp.ap-northeast-1.rds.amazonaws.com | 利用可能 | Writer | 3306 |
| lab2-aurora.cluster-ro-cnkerumhjjp.ap-northeast-1.rds.amazonaws.com | 利用可能 | Reader | 3306 |

ログイン用コマンド

```
$ mysql -h lab2-aurora.cluster-cnkerumhjjp.ap-northeast-1.rds.amazonaws.com -u admin -p
```

バックアップ[°]

2023年9月6日 13:51

AWS Backup

2023年9月6日 14:35

efsの場合 自動でポールドが作成される

[Amazon EFS](#) > [ファイルシステム](#) > [fs-071ff754a0d05da62](#) > ファイルシステムを編集

編集

全般

自動バックアップ

推奨設定を使用して、AWS Backup でファイルシステムデータを自動的にバックアップします。追加料金が適用されます。 [詳細はこちら](#)

自動バックアップを有効化

ライフサイクル管理

標準 - 低頻度アクセス (IA) ストレージクラスにファイルを移動することで、アクセスパターンの変化に応じて自動的に費用を節約できます。 [詳細はこちら](#)

IA へ移行

標準アクセスから標準 - 低頻度アクセスにファイルを移行します。

なし

IA から移行

標準 - 低頻度アクセスから標準アクセスにファイルを移行します。

なし

ライフサイクル管理を有効にしてファイルを IA ストレージに移行している場合にのみ有効です。

ポールドの作成

[AWS Backup](#) > [バックアップポールト](#) > バックアップ保管庫の作成

バックアップ保管庫の作成 情報

全般

バックアップポールト名

lab2-vault

バックアップポールト名では大文字と小文字が区別されます。2~50 文字の英数字または「_」を含める必要があります。

暗号化キー 情報

KMS キーを選択してください

バックアップポールトタグ - オプション

ここで指定されたタグは、バックアップポールトを整理して追跡するのに役立ちます

この保管庫にはタグが関連付けられていません。

[新しいタグを追加](#)

最大 50 個のタグをさらに追加できます。

キャンセル

[バックアップ保管庫の作成](#)

バックアッププランの作成

毎日12:00にバックアップを取得する

バックアッププランを作成 情報

起動オプション

バックアッププランのオプション 情報 テンプレートで開始する

AWS Backup から提供されたテンプレートに基づいてバックアッププランを作成します。

 新しいプランを立てる

新しいバックアッププランを最初から設定します。

 JSON を使用してプランを定義

既存のバックアッププランの JSON 式を変更、または新しい式を作成します。

バックアッププラン名

lab2

バックアッププラン名では大文字と小文字が区別されます。1~50 文字の英数字または「-_」を含める必要があります。

▶ バックアッププランに追加されたタグ - オプション

バックアップルールの設定 情報

バックアップルール名

lab2

バックアップルール名では大文字と小文字が区別されます。1~50 文字の英数字または「-_」を含める必要があります。

バックアップポールト 情報

lab2-vault



新しいバックアップポールトを作成

バックアップ頻度 情報

毎日

継続的バックアップ 情報

継続的なバックアップでは、選択した特定の時刻 (1 秒単位で指定でき、最大 35 日前まで) に巻き戻すことによって、AWS Backup でサポートされているリソースを復元できます。RDS, S3, および SAP HANA on Amazon EC2 リソースで使用できます。

 ポイントインタイムリカバリ (PITR) のために継続的なバックアップを有効化バックアップ期間 情報

開始時間

バックアップを開始する時刻を指定します。該当する場合、時刻は夏時間に合わせて調整され、1 年を通して同じ現地時間が維持されます。

12 : 00

Asia/Tokyo (UTC+09:00)

内から始める 情報

指定した時間にバックアッププランが開始されない場合は、バックアッププランを開始する期間を指定します。

8 時間

次の時間以内に完了 情報

7 日



リソースの割り当てする(作成したプランから)

リソースの割り当て 情報

全般

リソース割り当て名

lab2-ec2

リソース割り当て名では大文字と小文字が区別されます。1~50 文字の英数字または「_」を含める必要があります。

IAM ロール 情報

AWS Backup は、ユーザーに代わって復旧ポイントを作成および管理するときにこの IAM ロールを引き受けます。

デフォルトのロール

AWS Backup のデフォルトのロールが存在しない場合は、正しい許可を持つロールが作成されます。

IAM ロールを選択してください

リソースの選択 情報

タグとリソース ID を使用して、このバックアッププランにリソースを割り当てます。

1. リソース選択を定義 情報

すべてのリソースを保護するか、タイプまたは ID でリソースを指定します。

すべてのリソースタイプを含める

アカウントで有効になっているすべてのリソースタイプを保護します。

特定のリソースタイプを含める

タイプ別にリソースを選択するか、ID で個別のリソースを指定します。

2. 特定のリソースタイプを選択 情報

このバックアップ計画で保護する特定のリソースタイプを選択します。特定のリソース ID を選択から除外することもできます。

リソースタイプを選択 ▼

リソースタイプ

EC2

インスタンス ID

リソースを選択 ▼

削除

i-0cdf301160629218c X

3. 選択したリソースタイプから特定のリソース ID を除外する - オプション 情報

この割り当てから除外する特定のリソース ID を選択します。

リソースタイプを選択 ▼

4. タグを使用して選択を絞り込む - オプション 情報

タグでリソースをフィルタリングします。タグが複数ある場合、リソースはすべてのタグ条件を満たす場合にのみバックアッププランに割り当てられます。

リソース選択を絞り込むタグが選択されていません。

タグを追加

最大 30 個のタグを追加できます。

キャンセル

リソースの割り当て

復元方法

| 復旧ポイント (1/1) 情報 | | | | | | | <input type="button" value="C"/> | すべて解除 | アクション ▲ |
|-------------------------------------|-----------------------------|-------|----------|------------------------------|---------|--------|-----------------------------------|-----------------------------------|---------|
| | | | | | | | | | |
| <input checked="" type="checkbox"/> | 復旧ポイント ID | ステータス | リソース名 | リソース ID | リソースタイプ | バックアップ | | | |
| <input checked="" type="checkbox"/> | image/ami-08aa76f3cee2d3b7f | 完了 | lab2-ec2 | instance/i-0cdf301160629218c | EC2 | イメージ | <input type="button" value="復元"/> | <input type="button" value="削除"/> | 23% |

※ロールが必要

AWS Backup > ショグ > 790BA0DF-0AB5-0A3B-AE3D-A10BAC08906E

復元 - 790BA0DF-0AB5-0A3B-AE3D-A10BAC08906E

復元ジョブの詳細ページで、最新の復元ジョブのレコードにアクセスできます。

詳細

| | | | |
|--|--|--------------|----------------|
| 復旧ポイント ARN arn:aws:ec2:ap-northeast-1:image/ami-023c52d26ac8da47a | ステータス 失敗 | リソース ID - | リソースタイプ EC2 |
|--|--|--------------|----------------|

| | | |
|-------------------------------------|-------------------------------------|-------------------|
| 作成日 2023年9月6日, 14:40 (UTC+09:00) | 完了日 2023年9月6日, 14:40 (UTC+09:00) | バックアップサイズ 8 GB |
|-------------------------------------|-------------------------------------|-------------------|

IAM ロール

デフォルトのロール

✖ You are not authorized to perform this operation. Please consult the permissions associated with your AWS Backup role(s), and refer to the AWS Backup documentation for more details. Encoded authorization failure message: taJzAv9SQQGzqPySMBlvBbOZ0XjEFkLnZnrlQLf6i4yZrEGUYI5Wgrzu1Rxtj7z_NwsXf5bwI-jvV4ta9j1e15mPW2FAOrmwoN2PH0QBls3UObumi1rnKWKv12VdouP62iAwWMk3h1aufTqaMVuUM-pUDh_4JkswzzmB2zh4TkKoKwSYDjwvC0LwsbHTHgDdN6Hy41TWn7p21UxCSG6EdpEkzBtpcz8s0anqNT1qpFn_n0VEUgHUtdN2c_iiRrR0pBK8mZC8P0_i2luJzqDVLD6vd1bdALR20rmcmZtyVv87UwvDtp0_f52l0tgbqjYShZqaSCsxeSkW41KT9pVU0609UrD5Q_Q_wubQjmQCT90yplKo6dDf7qNsqJ56y0gM2PKzenp35SobQ2lEz7-1Y7A-Wcs-HsvmcGxolyG2u0fRqM6nBFik9W1OY6iBHNrKsC-MuGLFHTal1TX5Aqm8TP_CbWBUsdDMxGvPzKeXtcQPO9eCbMw7K-BcZvvuHgnjVaU6yunfmBRE1QuwlDDlyaqVpr7os

| <input type="checkbox"/> | ポリシー名 | タイプ |
|--------------------------|---------------------------------------|--------|
| <input type="checkbox"/> | AmazonEC2FullAccess | AWS 管理 |
| <input type="checkbox"/> | IAMFullAccess | AWS 管理 |
| <input type="checkbox"/> | AWSBackupServiceRolePolicyForBackup | AWS 管理 |
| <input type="checkbox"/> | AWSBackupServiceRolePolicyForRestores | AWS 管理 |

EC2バックアップ

2023年9月6日 10:17

①スナップショットを作成する

バックアップを取りたいインスタンスを選択する

スナップショットを作成 情報

EBS ボリュームのポイントインタイムスナップショットを作成し、新しいボリュームまたはデータバックアップのベースラインとして使用します。個々のボリュームからスナップショットを作成するか、インスタンスにアタッチされたすべてのボリュームからマルチボリュームスナップショットを作成することができます。

スナップショットの設定

リソースタイプ 情報

ボリューム
特定のボリュームからスナップショットを作成します。

インスタンス
インスタンスからマルチボリュームスナップショットを作成します。

インスタンス ID
マルチボリュームスナップショットの作成元となるインスタンス。
i-0c69f794a7a07b744

説明
スナップショットの説明を追加します。
最大 255 文字

②スナップショットからボリュームを作成する

スナップショット (1/1) 情報

自己所有 検索

lab2 フィルターをクリア

| Name | スナップショット ID | ボリューム... | 説明 | ストレー... | ス... |
|------------------------|-------------|----------|--------|-------------------------------------|------------------|
| snap-01601cabca078ab46 | 8 GiB | lab2-ec2 | スタンダード | <input checked="" type="checkbox"/> | スナップショットの高速復元を管理 |

アクション ▲ **スナップショットの作成**

- スナップショットからボリュームを作成
- スナップショットからイメージを作成
- スナップショットをコピー
- アクセス権限を変更
- スナップショットの高速復元を管理

ボリューム設定

スナップショット ID
 snap-01601cabca078ab46

ボリュームタイプ 情報

汎用 SSD (gp2)

サイズ (GiB)

8

最小: 1 GiB、最大: 16384 GiB。値は整数である必要があります。

IOPS

100 / 3000

1 GiBあたり 3 IOPS のベースライン、最小 100 IOPS、3000 IOPS にバースト可能。

スループット (MiB/秒) 情報

該当しません

アベイラビリティーゾーン 情報

ap-northeast-1a

高速スナップショット復元 情報

選択されたスナップショットでは有効になっていません

暗号化 情報

EC2 インスタンスに関連付けられた EBS リソースの暗号化ソリューションとして、Amazon EBS 暗号化を使用します。

このボリュームを暗号化する

③ボリュームをアタッチする

基本的な詳細

ボリューム ID
vol-0b7c25bb4bd7049cf

アベイラビリティーゾーン
ap-northeast-1a

インスタンス 情報
i-0c69f794a7a07b744

選択したボリュームと同じアベイラビリティーゾーンにあるインスタンスのみが表示されます。

デバイス名 情報
/dev/sdf

Linux 用の推奨デバイス名: ルートボリュームの場合は /dev/sda1。データボリュームの場合は /dev/sd[f-p]。

① ここで入力された (および詳細情報に表示される) デバイス名が /dev/sdf から /dev/sdp であっても、新しい Linux カーネルによっては内部でデバイスの名前が /dev/xvdf から /dev/xvdp に変更されることがあります。

キャンセル ボリュームのアタッチ

インスタンスでコマンド実行する

```
sudo mount -o nouuid /dev/sdf1 /mnt
```

コマンド以外の方法

詳細 | セキュリティ | ネットワーキング | **ストレージ** | ステータスチェック | モニタリング | タグ

▼ ルートデバイスの詳細

ルートデバイス名
/dev/xvda

ルートデバイスタイプ
EBS

EBS 最適化
有効

▼ ブロックデバイス

| ブロックデバイスのフィルター | | | | | | |
|-----------------------|-----------|-------------|-------------|------------------------|-------|-----------|
| ボリューム ID | デバイス名 | ボリュームサイズ... | アタッチメントの... | アタッチ時刻 | 暗号化済み | KMS キー ID |
| vol-0aeaf13744d6821ab | /dev/sdf | 8 | ☑ アタッチ済み | 2023/09/06 13:29 GMT+9 | いいえ | - |
| vol-0099bf30cc92ee4c0 | /dev/xvda | 8 | ☑ アタッチ済み | 2023/09/06 13:37 GMT+9 | いいえ | - |

▼ 最近のルートボリュームの置き換えタスク

| タスクをフィルタリング | | | | | ルートボリュームを置き換える |
|------------------------------|--------|----------------------|----------------------|----|----------------|
| タスク ID | タスクの状態 | 開始時刻 | 完了時間 | タグ | |
| replacevol-0fa91385261f10a66 | ☑ 成功 | 2023-09-06T04:34:37Z | 2023-09-06T04:35:03Z | - | |
| replacevol-00856bc4e34f72af2 | ☑ 成功 | 2023-09-06T04:37:16Z | 2023-09-06T04:37:35Z | - | |

ルートボリュームの詳細

インスタンス ID

i-0cdf301160629218c (lab2-ec2)

ルートボリューム ID

vol-0099bf30cc92ee4c0

最近のルートボリュームの置き換えタスク

▶ replacevol-0fa91385261f10a66  成功

復元

インスタンスのルートボリュームを初期起動状態に復元するか、特定のスナップショットから、または AMI から復元するかを指定します。

起動状態

スナップショット

イメージ

スナップショット

スナップショットを指定して、ルートボリュームをその状態に復元します。

snap-0cce6c516a347e089  

置き換えたルートボリュームを削除

元のルートボリューム (vol-0099bf30cc92ee4c0) が正常に置き換えられたら、削除します。

RDSバックアップ

2023年9月6日 13:11

スナップショットの移行・・・mysqlauroraに変更する場合
スナップショットを復元・・・新しいインスタンスが作成される

RDS > スナップショット > スナップショットの取得

DB スナップショットの取得

設定

DB スナップショットを作成するには、DB インスタンスを選択し、DB スナップショットに名前を付けます。

DB インスタンス
DB インスタンス識別子。これは、DB インスタンスを識別する一意のキーです。

lab1-rds ▾

スナップショット名
DB スナップショットの識別子。

mysnapshot

スナップショット識別子は大文字と小文字が区別されませんが、「mysnapshot」のようにすべて小文字で保存されます。null (0)、空、または空白にすることはできません。1~255 文字の英数字またはハイフンを使用してください。先頭文字は英字にする必要があります。ハイフンで終わるか、2 つ連続してハイフンを使用することはできません。

キャンセル **スナップショットの取得**

復元すると同じデータが入った

DynamoDBバックアップ

2023年9月6日 14:05

The screenshot shows the AWS DynamoDB console interface. The top navigation bar includes tabs for Overview, Indices, Monitoring, Global Tables, Backups (which is selected), and Export/Import. On the left, a sidebar lists tables: gakuen_timetable and lab1-kadai5, with lab1-kadai5 currently selected. The main content area displays information about PITR (Point-in-time Recovery) protection for the selected table. It notes that data is protected for 35 days and provides a link to the Amazon DynamoDB Pricing page. Below this, a section titled 'Backup Creation' is shown, featuring a search bar and a table with columns for Name, Status, Creation Time, ARN, and Size. A red box highlights the 'Create Backup' button at the top right of this section.

CloudFormation

2023年9月6日 13:52



example

jsonかyml形式で作成
する

リソースタブで現在の状況を確認することができる

出力タブでコードに設定しておけばエンドポイントなどを出力することも可能

ドリフトの検出で変更点を確認できる

IN_SYNC・・・同期している(変更されていない)

DELETED・・・変更されている(変更された)

Day3

2023年9月6日 15:26

画面録画しているので課題文をスクロールしておく

ヒントを使用すると解けた際に獲得できる点数が減る

チャレンジ1

2023年9月6日 15:32

AWS Jam - チャレンジ 1

まとめ昨夜、口

グシステムがクラックされ、楽しみのためにいくつかの変更が加えられました。その結果、誰もログ システムの中核である OpenSearch ドメインにアクセスできなくなりました。

システム ログ収集インフラストラクチャを迅速に修復して、安全なアクセスを備えた通常の動作に戻します。

インベントリ

• OpenSearch ドメイン •

[checkip.amazonaws.com](#)を使用する現在使用しているCIDRを識別するため

インベントリで使用するサービスを確認できる

OpenSearch

https://blog.serverworks.co.jp/tech/2016/03/08/elasticsearch_policy/

・ OpenSearchを変更する

作成時にアクセスポリシーを設定することができるのでIPを確認してそのIPを許可する

Amazon OpenSearch Service

マネージド型クラスター ダッシュボード ドメイン リザーブドインスタンスのリース パッケージ VPC エンドポイント

サーバーレス ダッシュボード コグジョン 新規 セキュリティ SAML 認証 データアクセスポリシー 暗号化ポリシー ネットワークポリシー VPC エンドポイント

取り込み バイオペイン 通知

セキュリティ設定 クラスターのヘルス インスタンスのヘルス オフピーウィンドウ 自動調整 ログ インデックス タグ

セキュリティ設定

| きめ細かなアクセスコントロール | OpenSearch Dashboards/Kibana の認証 | 暗号化 |
|-----------------|----------------------------------|-----------------|
| 有効 はい | SAML が有効になっています いいえ | 必要な HTTPS はい |
| マスターアクセス | Cognito が有効になっています いいえ | ノード間の暗号化 はい |
| 内部ユーザーデータベース | リージョン アジアパシフィック(東京) | 保存時の暗号化 はい |

AWS KMS キー
arn:aws:kms:ap-northeast-1:600728620263:key/25052c13-ae78-4207-b36d-58bc802605b8

アクセスポリシー 情報

ポリシー

```
{ "Version": "2012-10-17", "Statement": [ { "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": "es:*" } ] }
```

ポリシーコードをコピー

チャレンジ 2

2023年9月6日 15:40

AWS Jam – チャレンジ 2

まとめ

あなたは開発チームに新しいチームメンバーとして加わりました。このチームは Web アプリケーション開発に取り組んでいます

開発チームは、Amazon EC2 Linux インスタンスに Web アプリケーションをインストールして設定しました。この Amazon EC2 インスタンスは、AWS Application Load Balancer (ALB) の背後で実行されます。

開発チームはウェブ アプリケーションへのアクセス中に問題が発生しています。ウェブ アプリケーションが完全に正しく設定されていると確信していましたが、AWS マネジメント コンソールの設定を見逃していた可能性があります。

あなたの目的は、問題の根本原因を特定し、この課題に解決策を適用することです。
幸運を！

在庫

- EC2
- RDS
- ALB

- ・ ウェブにアクセスできない → セキュリティグループなど
- ・ RDSに接続できない → セキュリティグループなど
- ・ ターゲットグループが異常 → とりあえず curl -I で ポートやパスが間違っていないか確認する

チャレンジ 3

2023年9月6日 15:48

AWS Jam - チャレンジ 3

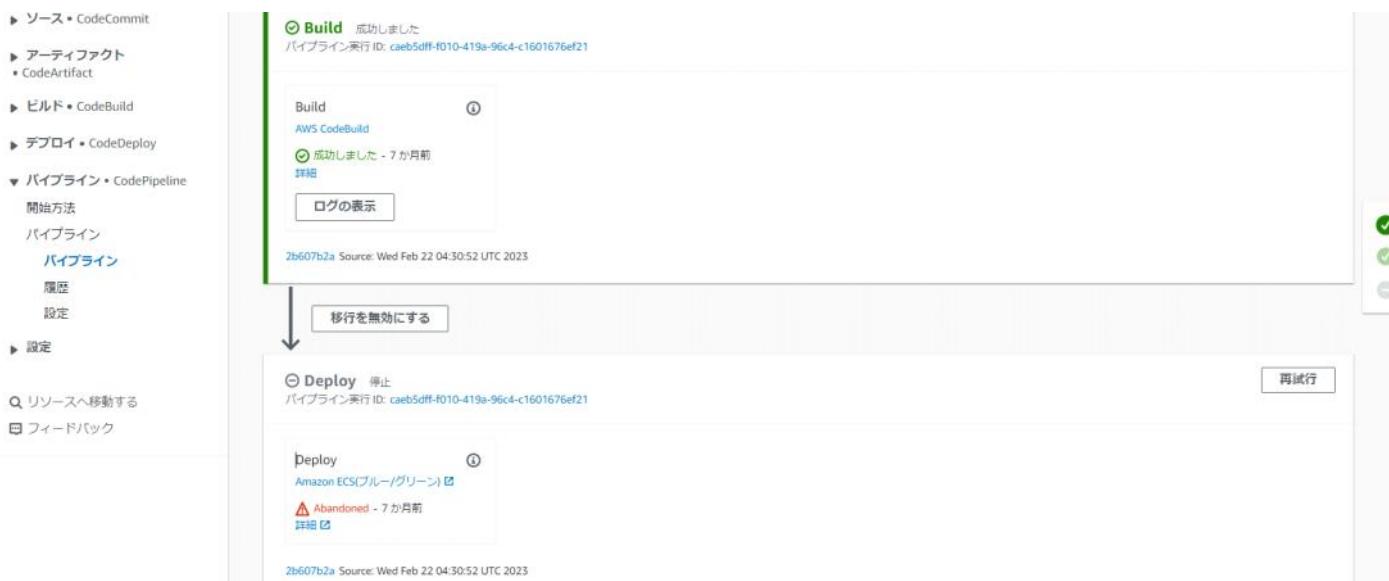
まとめ

DevOps エンジニアは、ソースコードに AWS CodeCommit、ビルトの実行に AWS CodeBuild、アプリケーションを EC2 インスタンスにデプロイするために AWS CodeDeploy を使用する AWS CodePipeline を使用してデモ パイプラインをセットアップしました。しかし、パイプラインはソース段階で障害が発生しており、Devops エンジニアは個人的な緊急事態のため休暇中です。チームはあなたが問題を見つけてパイプラインの問題を解決することを期待しています。

在庫

アプリケーションをホストする単一の Web サーバー。そのアドレスは出力プロパティにあります。

- AWS CodePipeline パイプライン: JAMPipeline
 - AWS CodeCommit リポジトリ: CDSource
 - AWS CodePipeline サービスロール: JAMCodePipelineServiceRole
-
- Pipeline のエラーを確認
 - ポリシーが必要
 - コードデプロイエージェントが必要



チャレンジ4

2023年9月8日 11:06

AWS Jam – チャレンジ4

まとめ

あなたは、AWS EKS サービス上で実行されるシステムの運用と保守を引き継ぎました。誰かが誤って EKS クラスターを削除しました。以前の運用および保守スタッフは退職し、EKS クラスターをデプロイするための yaml ファイルが 1 つだけ残されました。マネージャーは、EKS クラスターを復元し、AWS コンソールでノードの情報を利用できるようにすることを求めています。

在庫

- ・アマゾンEC2
- ・Amazon Elastic Kubernetes Service (EKS)
- ・AWS システムマネージャー

チャレンジ5

2023年9月6日 15:53

AWS Jam – チャレンジ 5

まとめ

ある会社には、仮想ホスティングを使用して 1 つの EC2 インスタンス上で提供される複数の内部アプリケーションがあります。Web サイトの 1 つはトラフィックが高かったため、EC2 インスタンスを手動で垂直方向にスケーリングし続ける必要がありました。

もう 1 つの問題は、1 つのアプリケーションの負荷が原因で他のアプリケーションが問題に直面していることでした。この問題を永久に解決するために、彼らは AWS アーキテクトに連絡を取りました。

あるアプリケーションが原因で他のアプリケーションのパフォーマンスが低下することを考慮して、アーキテクトは次のように要求しました。

1. 別々のアプリケーションには別々の EC2 インスタンスを使用します。
2. 負荷が高く予測不可能なアプリケーションには AutoScaling Group を使用します。

会社は、問題を解決するために App1 と App2 という 2 つのアプリケーションを選択しました。

- App1 については、AutoScaling グループを作成しました。
- App2 については、EC2 インスタンスを作成しました。

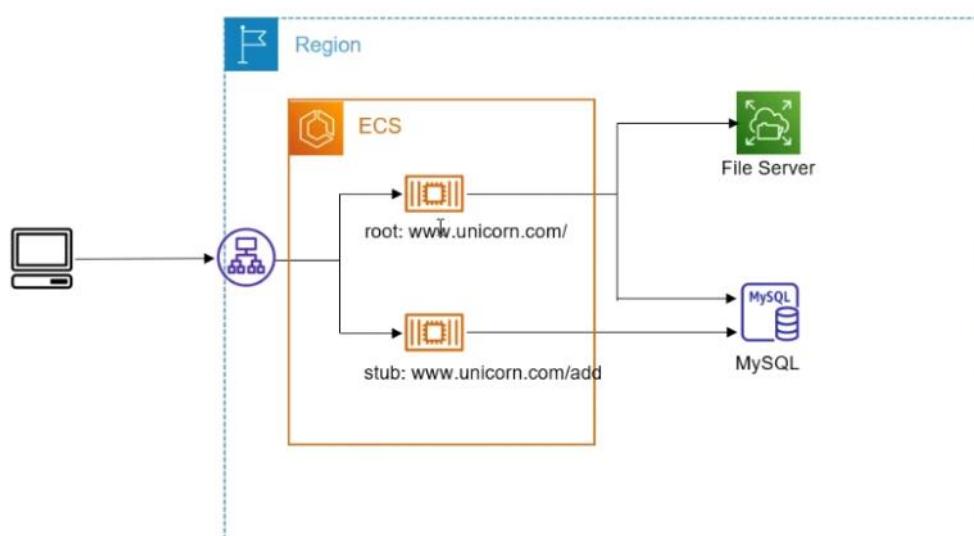
仮想ホスティングを使用する代わりに、お客様はバースペースのアプリケーション ルーティングを検討する準備ができています。

これら 2 つが期待どおりに機能する場合は、他のアプリケーションも移植する必要があるかもしれません。

在庫

- Application Load Balancer、ターゲット グループ、AutoScaling グループ、EC2 インスタンス、セキュリティ グループ。 • 在庫の全リストは出力セクションにあります。

マイクロサービス化をしたい



ALBのルールを追加する必要がある

一つ目をautoscalling

ターゲットグループを2つ作成する

ロードバランサーのルールを確認する → パスによってターゲットグループを変更する

チャレンジ 6

2023年9月6日 16:00

AWS Jam - チャレンジ 6

まとめ

2021 年 12 月、重要な安全上の脅威が世界中でパニックを引き起こしました。 Log4Shell の侵害。

このチャレンジでは、この脅威から保護するために AWS が提供する主要なセキュリティ ツールを試します。特に、この課題では、アカウントのコンプライアンスと受信トラフィックからの保護に焦点を当てています。もちろん、

他のアクションは、発信トラフィックや他の側面（ネットワーキング、権限、検出、フィルタリングなど）に対して実行できます。

このチャレンジでは、あなたはセキュリティ チームの一員として、開発チームからのパッチを待つ間に AWS アカウントの公開を制限しようとします。

在庫

- ・企業 Web サイトを提供するパブリック ALB がアカウントにすでに存在している
- ・クラウドフロント
- ・AWS WAF ルール

設定

料金クラス 情報

- 支払う上限価格に関連付けられている料金区分を選択します。
- すべてのエッジロケーションを使用する (最高のパフォーマンス)
 - 北米と欧州のみを使用
 - 北米、欧州、アジア、中東、アフリカを使用

AWS WAF ウェブ ACL - オプション

このディストリビューションに関連付ける AWS WAF のウェブ ACL を選択します。

my-web-acl ▾

代替ドメイン名 (CNAME) - オプション

このディストリビューションで提供されるファイルの URL で使用するカスタムドメイン名を追加します。

*.ocp4.work

削除

項目を追加

① 代替ドメイン名のリストを追加するには、一括エディタ を使用します。

カスタム SSL 証明書 - オプション

AWS Certificate Manager から証明書を関連付けます。証明書は、米国東部 (バージニア北部) リージョン (us-east-1) にある必要があります。

*.ocp4.work (427fe69d-18bd-4b30-b11d-ae470b84d1f3) ▾

⟳

*.ocp4.work [証明書をリクエスト]

レガシークライアントサポート - 月額 600 USD の比例配分された料金が適用されます。ほとんどのお客様はこれを必要としません。

CloudFront は、HTTPS 経由でコンテンツを配信する、各 CloudFront エッジロケーションに専用 IP アドレスを割り当てます。

有効

セキュリティポリシー

セキュリティポリシーは、SSL または TLS プロトコルと、CloudFront がビューワー (クライアント) との HTTPS 接続に使用する特定の暗号を決定します。

TLSv1.2_2021 (推奨)

TLSv1.2_2019

TLSv1.2_2018

TLSv1.1_2016

TLSv1_2016

<https://dev.classmethod.jp/articles/aws-waf-new-rule-log4jrce/>

チャレンジ7

2023年9月6日 16:10

AWS Jam – チャレンジ7

まとめ

あなたはネットワーク アーキテクトのチームを率いています。組織内の全員が優れたパフォーマンスを発揮し、最適な管理を行ってネットワークをセッタップします。ある日、一部の AWS アカウントにセキュリティ グループがインターネットに公開されていることに気づき、「しまった! 私のスター パフォーマーにそんなことは期待していなかった」と思いました。あなたは、早期に検出して手遅れになる前に修復できるように、自動化を実装することにしました。

在庫

- Amazon EC2 セキュリティ グループ
- AWS 設定
- AWS システムマネージャーのドキュメント
- AWS IAM ロール

AWS Config でルールに違反しているものを検索でき、自動修復が可能

AWS システムマネージャー

<https://dev.classmethod.jp/articles/auto-recovery-restricted-ssh-without-lambda/>

①ポリシーを作成してロールにアタッチする

信頼されたエンティティを選択 情報

信頼されたエンティティタイプ



ユースケース

EC2, Lambda、その他の AWS のサービスがこのアカウントでアクションを実行することを許します。

一般的なユースケース

EC2
Allows EC2 instances to call AWS services on your behalf.

Lambda
Allows Lambda functions to call AWS services on your behalf.

```
1 * [
2     "Version": "2012-10-17",
3     "Statement": [
4         {
5             "Sid": "VisualEditor0",
6             "Effect": "Allow",
7             "Action": "ec2:RevokeSecurityGroupIngress",
8             "Resource": "*"
9         }
10    ]
11 ]]
```

②ルールを作成する

AWS 構成 > ルール > ルールを追加

ステップ 1
ルールタイプの指定

AWS リソースに必要な設定を定義するためのルールを追加します。ニーズに合わせて以下のルールをカスタマイズするか、カスタムルールを作成します。

ステップ 2
ルールの設定

ステップ 3
確認と作成

ルールタイプの選択

- AWS によって管理されるルールの追加
ニーズに合わせて以下のルールをカスタマイズします。
- カスタム Lambda ルールを作成
カスタムルールを作成し、AWS Config に追加します。各カスタムルールを AWS Lambda 関数に関連付けてください。この機能では、AWS リソースがルールに準拠しているかどうかを評価するログが含まれています。
- Guard を使用してカスタムルールを作成
AWS リソースがルールに準拠しているかどうかを評価する Guard カスタムルールを使用してカスタムルールを作成します。

AWS マネージド型ルール (318)

| Q ssh | | | | X | 2試合 | < | 1 | > | ④ |
|--|---|----------------|--|---|-----|---|---|---|---|
| 名前 | ▲ ラベル | サポートされている評価モード | 説明 | | | | | | |
| <input type="radio"/> nacl-no-unrestricted-ssh-rdp | NACL, SSH, RDP, RESTRICTED, NETWORK, ACL, TCP | 探偵 | Checks if default ports for SSH tcp/22 or RDP tcp/3389 ingress traffic for network access control lists (NACLs) is unrestricted. The rule is NON_COMPLIANT if a NACL inbound entry allows a source CIDR block of 0.0.0.0/0 or ::/0 for ports 22 or 3389. | | | | | | |
| <input checked="" type="radio"/> restricted-ssh | EC2 | 探偵 | Checks whether security groups that are in use disallow unrestricted incoming SSH traffic. | | | | | | |

キャンセル 次へ

評価モード

- プロアクティブ評価をオンにする
プロビジョニング前にリソースの評価を有効にする
- 検出評価をオンにする
プロビジョニングされたリソースの評価を有効にする

トリガータイプ

AWS Config は、トリガーが発生したときにリソースを評価します。

- 設定変更時
指定した AWS リソースに変更があると実行されます
- 定期的
選択した頻度で実行する

変更範囲

評価をいつ行うかを選択します。

- すべての変更
AWS Config によって記録されたリソースが作成、変更、または削除されたとき
- リソース
指定されたタイプ、またはタイプと ID に一致するリソースが作成、変更、または削除されたとき
- タグ
指定されたタグを持つリソースが作成、変更、または削除されたとき

リソース

このルールは、記録されたリソースが作成、編集、削除されたときにのみトリガーできます。[Settings] ページを編集して、記録するリソースを指定します。

リソースカテゴリ

リソースタイプ

すべてのリソースカテゴリ ▼ 複数選択 ▼

AWS EC2セキュリティグループ X

リソース識別子 - オプション

Q リソース識別子を入力

③自動修復の設定

AWS 構成 > ルール > restricted-ssh

制限付き SSH

| アクション ▲ |
|---------|
| 修復の管理 |
| 再評価 |
| 結果の削除 |
| ルールの削除 |

ルールの詳細

説明

使用中のセキュリティグループが無効な場合の受信 SSH トライアル

有効な評価モード

- 探偵

検出評価トリガーのタイプ

- オーバーナイミングの設定変更

編集: 修復アクション

▼ 修復方法を選択

- 自動修復
スコープ内のリソースが非準拠になると、修復アクションを自動的にトリガードします。

- 手動修復
非準拠リソースの修復を手動で選択する必要があります。

自動修復後もリソースがまだ準拠していない場合は、このルールを再試行するように設定できます。修復スクリプトの実行には費用がかかることに注意してください。

再試行まで 秒

▼ 修復アクションの詳細

修復アクションの実行は、AWS Systems Manager Automation を使用して実現されます

修復アクションを選択

AWS-DisablePublicAccessForSecurityGroup ▾

指定された IP アドレス、またはアドレスが指定されていない場合はすべてのアドレスに対して開かれた SSH および RDP ポートを無効にします。[RevokeSecurityGroupIngress]
(https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API_RevokeSecurityGroupIngress.html) API と同様に、セキュリティ グループには、特に SSH ポートと RDP ポートに関する既存のルールが必要です。無効になる。

AutomationAssumeRole に作成したロールのarnを追加する

▼ レート制限

SSM ドキュメントが一度に実行されるリソースの割合、およびバッチ全体が失敗とマークされる SSM 実行の失敗の割合を指定できます。

同時実行率 エラー率

▼ リソース ID パラメータ

ドロップダウンリストを使用して、リソースタイプに依存するパラメータを選択することで、準拠していないリソースのリソース ID を修正アクションに渡すことができます。ドロップダウンリストで使用できるパラメータは、選択した修復アクションに応じて異なります。

グループID ▾

▼ パラメータ

すべてのパラメータには、静的値または動的値のいずれかがあります。[リソース ID] ドロップダウンリストからパラメータを選択すると、選択したパラメータに RESOURCE_ID 値が渡されます。他のすべてのキーの値は入力が可能です。[リソース ID] ドロップダウンリストからパラメータを選択しない場合は、各キーの値を入力でできます。

| | | |
|----------------------|---|--|
| GroupId | > | RESOURCE_ID |
| IpAddressToBlock | > | (オプション) |
| AutomationAssumeRole | > | arn:awsiam::608728620263:role/DisablePublicA |

HTTPフルアクセスのセキュリティグループを検出して修復する方法

①ポリシー作成

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Action": [
        "ec2:RevokeSecurityGroupIngress"
      ]
    }
  ]
}
```

```

        ],
      "Resource": [
        "*"
      ]
    }
  }
}

```

②ルールを作成

ルールタイプの選択

- | | | |
|---|---|---|
| <input checked="" type="radio"/> AWS によって管理されるルールの追加 ニーズに合わせて以下のルールをカスタマイズします。 | <input type="radio"/> カスタム Lambda ルールを作成 カスタムルールを作成し、AWS Config に登録します。カスタムルールは AWS Lambda を呼び出すことで実行できます。このルールは AWS リソースがルールに参照しているかどうかを評価するロジックが含まれています。 | <input type="radio"/> Guard を使用してカスタムルールを作成 AWS リソースガルールに参照しているかどうかを評価する Guard カスタムポリシーを使用してカスタムルールを作成します。 |
|---|---|---|

AWS マネージド型ルール (318)

| AWS マネージド型ルール (318) | | | |
|--|---|----------------|---|
| 名前 | ラベル | サポートされている評価モード | 説明 |
| <input type="radio"/> nacl-no-unrestricted-ssh-rdp | NACL, SSH, RDP, RESTRICTED, NETWORK, ACL, TCP | DETECTIVE | Checks if default ports for SSH tcp/22 or RDP tcp/3389 ingress traffic for network access control lists (NACLs) is unrestricted. The rule is NON_COMPLIANT if a NACL inbound entry allows a source CIDR block of '0.0.0.0/0' or '::/0' for ports 22 or 3389. |
| <input type="radio"/> no-unrestricted-route-to-igw | Internet, Gateway, Public, Route | DETECTIVE | Checks if there are public routes in the route table to an Internet gateway (IGW). The rule is NON_COMPLIANT if a route to an IGW has a destination CIDR block of '0.0.0.0/0' or '::/0' or if a destination CIDR block does not match the rule parameter. |
| <input checked="" type="radio"/> restricted-common-ports | EC2 | DETECTIVE | Checks if the security groups in use do not allow unrestricted incoming Transmission Control Protocol (TCP) traffic to the specified ports for IPv4. The rule is COMPLIANT if IP addresses for inbound TCP connections are restricted to the specified ports. |
| <input type="radio"/> restricted-ssh | EC2 | DETECTIVE | Checks whether security groups that are in use disallow unrestricted incoming SSH traffic. |
| <input type="radio"/> s3-bucket-policy-grantee-check | S3, Zelkova | DETECTIVE | Checks that the access granted by the Amazon S3 bucket is restricted to any of the AWS principals, federated users, service principals, IP addresses, or VPCs that you provide. The rule is COMPLIANT if a bucket policy is not present. |

詳細

名前

一意のルール名。最大 128 文字を使用できます。特殊文字またはスペースは使用できません。

restricted-common-ports

Description - optional

ルールの評価対象と、準拠しないリソースを修正する方法を説明します。

Checks if the security groups in use do not allow unrestricted incoming Transmission Control Protocol (TCP) traffic to the specified ports for IPv4. The rule is COMPLIANT if IP addresses for inbound TCP connections are restricted to the specified ports.

管理されるルールの名前

RESTRICTED_INCOMING_TRAFFIC

評価モード

- プロアクティブ評価をオンにする
プロビジョニング前にリソースの評価を有効にする
- 検出評価をオンにする
プロビジョニングされたリソースの評価を有効にする

トリガータイプ
AWS Config は、トリガーが発生したときにリソースを評価します。

設定変更時
指定した AWS リソースに変更があると実行されます

定期的

選択した頻度で実行する

変更範囲

評価をいつ行うかを選択します。

- すべての変更
AWS Config によって記録されたリソースが作成、変更、または削除されたとき

- リソース
指定されたタイプ、またはタイプと ID に一致するリソースが作成、変更、または削除されたとき

- タグ
指定されたタグを持つリソースが作成、変更、または削除されたとき

リソース

このルールは、記録されたリソースが作成、複数、削除されたときにのみトリガーできます。[Settings] ページを複数して、記録するリソースを指定します。

リソースカテゴリ

すべてのリソースカテゴリ ▾

Multiple selected ▾

AWS EC2 SecurityGroup X

リソース識別子 - オプション

リソース識別子を入力

パラメータ

ルール/パラメータは、リソースが修復される属性を定義します。例えば、必須のタグや S3 バケットなどです。無効なままでのオプションのパラメータ(キーや値がない)は、保存されません。

| | |
|---------------------------------------|----|
| キー | 値 |
| blockedPort1 | 80 |
| <input type="button" value="削除"/> | |
| <input type="button" value="別の行を追加"/> | |

ルールタグ - オプション

ルールタグは完全にオプションです。値はオプションですが、キーのないタグは保存されないことに注意してください。

| | |
|---------------------------------------|---------|
| キー | 値 |
| キー | (オプション) |
| <input type="button" value="削除"/> | |
| <input type="button" value="別の行を追加"/> | |

③Systems Managerのドキュメントから

「AWS-DisablePublicAccessForSecurityGroup」のクローンを作成

ドキュメント

Q タグで検索する。またはクリックしてフィルタリングする
テキストを検索する: None Clear filters

ドキュメントのクローニング

| | | |
|---|--|---|
| AWS-DisablePublicAccessForSecurityGroup | awsConfigureAutomation-EnableCloudWatchLogsForSessionManager | awsConfigureAWSLogs-OperationalPracticesForSecurityServices |
| ドキュメントタイプ Automation | ドキュメントタイプ Automation | ドキュメントタイプ Automation |
| プラットフォームタイプ Windows, Linux, macOS | プラットフォームタイプ Windows, Linux | プラットフォームタイプ Windows, Linux |
| デフォルトバージョン 1 | デフォルトバージョン 1 | デフォルトバージョン 1 |

ステップ 2 移行の追加の入力のIpPermissionsの入力値のポートを80に変更する

④自動修復設定をする

▼ 修復方法を選択

自動修復
スコープ内のリソースが非準拠になると、修復アクションを自動的にトリガーリます。

手動修復
非準拠リソースの修復を手動で選択する必要があります。

自動修復後もリソースがまだ準拠していない場合は、このルールを再試行するようになります。修復スクリプトの実行には費用がかかることがあります。注意してください。

再試行まで 秒
1 60

▼ 修復アクションの詳細

修復アクションの実行は、AWS Systems Manager Automation を使用して実現されます

修復アクションを選択
lab2-AWS-DisablePublicAccessForSecurityGroup

Disable SSH and RDP ports opened to IP address specified, or to all addresses if no address is specified. Similar to the [RevokeSecurityGroupIngress](https://docs.aws.amazon.com/AWSEC2/latest/APIReference/API_RevokeSecurityGroupIngress.html) API, the security group must have existing rules specifically on the SSH and RDP ports in order for ingress to be disabled.

▼ レート制限

SSM ドキュメントが一度に実行されるリソースの割合、およびバッチ全体が失敗とマークされる SSM 実行の失敗の割合を指定できます。

同時実行率 エラー率
2 5

▼ リソース ID パラメータ

ドロップダウンリストを使用して、リソースタイプに値を指定することで、準拠していないリソースのリソース ID を修復アクションに渡すことができます。ドロップダウンリストで選択できるパラメータは、選択した修復アクションに応じて異なります。

GroupId

AutomationAssumeRoleに作成したロールのarnを追加する

▼ パラメータ

すべてのパラメータには、静的値または動的値のいずれかがあります。[リソース ID] ドロップダウンリストからパラメータを選択すると、選択したパラメータに RESOURCE_ID 値が渡されます。その他のすべてのキーの値は入力可能で、[リソース ID] ドロップダウンリストからパラメータを選択しない場合、各キーの値を入力できます。

| | |
|----------------------|---|
| GroupId | > RESOURCE_ID |
| IpAddressToBlock | > (オプション) |
| AutomationAssumeRole | > arn:aws:iam::608728620263:role/DisablePublicA |

⑤違反のあるセキュリティグループのインバウンドルールを作成すると自動で削除される

チャレンジ8

2023年9月6日 16:09

AWS Jam - チャレンジ8

まとめ

あなたは ACME 社のクラウド アーキテクトです。上司から、年次回復シナリオを実行するように頼まれました。ただし、AMI から作成したインスタンスは、理由は不明ですがすぐにシャットダウンされます。

さらに悪いことに、チームが復旧シナリオに対して十分な準備ができていることを証明するために、抜き打ち監査が行われることになります。

年次回復シナリオを成功裏に実行し、監査の観点から貴社が優良であることを証明するには、皆様のご協力が必要です。そうしないと、違反した場合に高額の罰金が科せられる可能性があります。窮地を救ってもらえますか？

この課題を解決するには、IAM ロール、ポリシー、KMS キーを活用します。

在庫

- EC2アプリケーションサーバー
- AMI イメージ
- IAM ユーザー

ユーザーに対するポリシー

KMSへのアクセス権限がないのでストレージへのアタッチが失敗する
立ち上げはできる

The screenshot shows the AWS EC2 AMI management interface. At the top, there's a breadcrumb navigation: EC2 > AMI > ami-04b4d663ee724e901. Below it is a summary card for the AMI:

| AMI ID | イメージタイプ | プラットフォームの詳細 | ルートデバイスタイプ |
|-----------------------|--------------|---|--------------|
| ami-04b4d663ee724e901 | machine | Linux/UNIX | EBS |
| AMI 名 | 所有者のアカウント ID | アーキテクチャ | 使用オペレーション |
| lab1-devsv | 608728620263 | x86_64 | RunInstances |
| ルートデバイス名 | ステータス | ソース | 仮想化タイプ |
| /dev/sda1 | 利用可能 | 608728620263/lab1-devsv | hvm |
| ブートモード | 状態の理由 | 作成日 | カーネル ID |
| uefi-preferred | - | Wed Sep 06 2023 08:18:55 GMT+0900 (日本標準時) | - |
| 説明 | 製品コード | RAM ディスク ID | 起動優先の時刻 |
| - | - | - | - |

Below the summary card, there are three tabs: 許可 (Permissions), ストレージ (Storage), and タグ (Tags). The Storage tab is selected, showing the following details:

最終起動時間: Wed Sep 06 2023 09:38:31 GMT+0900 (日本標準時)

ブロックデバイス: /dev/sda1: snap-05809ef1215dbf218:8:true:gp2:encrypted

ルートデバイスの詳細:

| ルートデバイス名 | デバイス名 | ルートデバイスタイプ |
|-----------|-----------|------------|
| /dev/sda1 | /dev/sda1 | EBS |

ブロックデバイス:

| Q. ブロックデバイスのファイル... | | | | | | |
|------------------------|-----------|-------------|-----------|-------|--------|-------------------------------------|
| デバイス ID | デバイス名 | ボリュームサイズ... | ボリュームタ... | 増分化済み | 終了時に削除 | KMSキー ID |
| snap-05809ef1215dbf218 | /dev/sda1 | 8 | gp2 | はい | はい | 539ac0ca-05ab-4884-817a-e2e66e000b8 |

IAM > ユーザー > labuser1

labuser1 詳細

概要

| | | |
|--|---------------------|----------------------|
| ARN arn:aws:iam::608728620263:user/labuser1 | コンソールを通じたアクセス 無効 | アクセスキー1 アクセスキーを作成 |
| 作成日 September 06, 2023, 09:29 (UTC+09:00) | 前回のコンソールサインイン - | - |

許可 グループ タグ セキュリティ認証情報 アクセスアドバイザー

許可ポリシー (2)

許可は、ユーザーに直接アタッチされたポリシー、またはグループを通してアタッチされたポリシーで定義されます。

| 権限 | ポリシー名 | タイプ | 次を絞り込む |
|--------------------------|---|------------|--------|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> AmazonEC2FullAccess | AWS 管理 | 直接 |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> KMS | カスタマーインライン | オンライン |

KMSのポリシーを追加する

KMS カスタマーインライン インライン

KMS

JSON をコピー **編集**

```
1 {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Sid": "VisualEditor0",  
6             "Effect": "Allow",  
7             "Action": "kms:*",  
8             "Resource": "*"  
9         }  
10    ]  
11 }
```

チャレンジ9

2023年9月6日 16:26

AWS Jam - チャレンジ9

まとめ

MegaShop は、さまざまなカテゴリの製品を販売するグローバルな電子商取引 Web サイトです。これにより、オンライン顧客は在庫内のすべての商品について製品レビューを提供できるようになります。MegaShop はグローバルに運営しているため、ユーザーはさまざまな国から来ており、さまざまな言語でレビューを書いています。MegaShop は、レビューが高く、顧客ベースの非常に高い品質要件を満たす製品のみを販売することを優先しています。

MegaShop の分析責任者は、顧客のレビューの分析について支援を求めてきました。彼は、顧客の期待を満たしていない製品を特定するために、どのレビューが否定的な感情を表現しているかを確認したいと考えています。ただし、レビューはさまざまな言語で行われるため、この作業は困難です。

さらに、レビューの数が多すぎて手動でレビューできません。

あなたのタスクは、各レビューの言語を特定し、次に意味分析を実行して、顧客によって否定的に評価された製品を特定することです。

Comprehend を使用

<https://dev.classmethod.jp/articles/comprehend-operations-using-python-boto3-ja/>

```
import json
import boto3

REGION = 'ap-northeast-1'
BUCKET = 'ahaws-bucket-01'

s3 = boto3.client('s3')
comprehend = boto3.client('comprehend', region_name=REGION)

# Function for detecting the dominant language
def detect_dominant_language(text):
    response = comprehend.detect_dominant_language(Text=text)
    return response

# Function for detecting sentiment
def detect_sentiment(text, language_code):
    response = comprehend.detect_sentiment(Text=text,
                                             LanguageCode=language_code)
    return response
```

```
objects = s3.list_objects_v2(Bucket=BUCKET, Prefix='challenge9/')
# print(objects['Contents'])
for object in objects['Contents']:
    obj = s3.get_object(Bucket=BUCKET, Key=object['Key'])
    text = json.loads(obj['Body'].read().decode('utf-8'))
    review = text['review']

    lang = detect_dominant_language(review)['Languages'][0]['LanguageCode']
    sentiment = detect_sentiment(review, lang)['Sentiment']
    # # print(object['Key'] + ':' + response['Sentiment'])

    text['sentiment'] = sentiment
    s3.put_object(Body=json.dumps(text).encode('utf-8'), Bucket=BUCKET, Key='result/' + object['Key'])
```

athenaからs3に保存したものを確認できる

```
SELECT item, COUNT(sentiment) FROM day3_challenge9 WHERE sentiment = 'NEGATIVE'
GROUP BY item, sentiment
ORDER BY COUNT(sentiment);
```

データ取得

データをでコードする

s3に保存

Day4

2023年9月7日 10:38

チャレンジ1

2023年9月7日 10:39

AWS Jam - チャレンジ 1

まとめ

このアプリケーションは、Application Load Balancer とコンテンツ配信用の Amazon CloudFront を備えた Amazon Elastic Compute Cloud (Amazon EC2) 上で実行されています。CloudFront はコンテンツを顧客に近づけ、セキュリティ体制を向上させるとともに、キャッシングによりバックエンド サーバーの負荷を軽減します。

顧客はサポートをリクエストしており、AWS の専門知識を必要としています。アプリケーションの安全性をさらに高め、エンド ユーザー エクスペリエンスを向上させるために、直ちにトラブルシューティングを行うことが求められます。アプリケーション チームはアーキテクチャ図を提供しました (以下を参照)。

Amazon CloudFront は要件ですが、このサービスは適切に保護されていません。オリジン、静的フェイルオーバー コンテンツ、およびアプリケーションのバックエンドは両方とも、パブリック インターネットから直接アクセスできます。

さらに 2 つのチームがオフィス内の AWS エキスパートに目覚めました。彼らは、問題を「すぐに」調べるように求めます (静的フェイルオーバー コンテンツをテストしている間は必ず休憩を取ってください。CloudFront の変更には時間がかかる場合があります)。興味を持ってください。興味深いトラブルシューティング セッションが予定されています。

次の AWS サービスを使用します: Amazon CloudFront、Amazon Simple Storage Service (Amazon S3)、Elastic Load Balancing、および Amazon EC2。

在庫

- CloudFront ディストリビューション
- CloudFront オリジン アクセス ID
- Jam S3 静的コンテンツ パケット
- アプリケーションロードバランサ

ALB直アクセスやs3直アクセスで見れないようにする

s3パケットのパブリックアクセスをオフにする
パケットポリシーをCloudFrontからもってくる

ALBのセキュリティグループをcloudfrontからのみにする

The screenshot shows the 'Inbound Rule' configuration for an origin access identity (OAI) in the AWS CloudFront console. The rule type is set to 'HTTP'. The source is set to 'CloudFront'. A search bar at the top right shows the term 'cloudfront'. The rule is associated with a pre-signed list named 'com.amazonaws.global.cloudfront.origin-facing | pl-58a04531'.

チャレンジ 2

2023年9月7日 10:46

AWS Jam - チャレンジ 2

まとめ

あなたは、Travelme に AI/ML エキスパートとして採用されました。同社はグローバル企業であり、多くの従業員が頻繁に出張しています。領収書を提出すると、会社は旅費を払い戻します。CFO には 10 人からなるチームがあり、各領収書を手動で調べてシステムに入力します。これには多くの時間がかかり、払い戻しプロセスが遅れる場合があります。従業員は、1週間以内に払い戻しを受けられないと苦情を言います。

新しい取り組みの一環として、CFO があなたのところに来て、請求書の入力プロセスの自動化を手伝ってほしいと依頼しました。

在庫

- ラムダ
- Textract API
- S3

Textractを使用してs3に保存する　リサイズする可能性あり

<https://laboratory.kazuuu.net/using-amazon-textract-with-boto3-in-python/>

チャレンジ 3

2023年9月7日 10:46

AWS Jam - チャレンジ 3

まとめ

この JAM チャレンジでは、複数の AZ の NAT ゲートウェイとインターネット ゲートウェイの間にファイアウォールを挿入するために、出力 VPC または検査 VPC で拡張ルーティングを構成する手順を説明します。これは、2021 年 8 月時点の新機能で、イングレストラフィックとサブネット間のトラフィックを AWS ネットワーク ファイアウォールにリダイレクトできるようになります。

あなたは金融機関の新人ネットワーク セキュリティ エンジニアです。あなたは、AWS 環境で実行されているインスタンスのストリーミング ビデオ サービスをブロックするために AWS ネットワーク ファイアウォールを実装している途中で退職した元従業員の後を引き継ぎます。あなたのタスクは、AWS ネットワーク ファイアウォール エンドポイントを実装し、エンドポイントを利用するようにルーティングを設定し、ファイアウォール ポリシーに追加のルールを追加することです。

同社の AWS ネットワークは 1 つのアカウントと VPC で構成されており、インスタンスは NAT GW から出る 2 つの AZ のプライベート サブネットにあります。AZI (アベイラビリティーゾーン独立性) の原則とベストプラクティスを使用して AWS ネットワーク ファイアウォールを実装し、この新しい機能を利用するように VPC 構成を変更します。

これは、この割り当ての開始時にデプロイされた VPC の図です。前任者は、ファイアウォールが存在するサブネットを作成し、使用する基本ルール グループを作成しました。さあ、仕事を終えるかどうかはあなた次第です。

プライベート サブネット内のテスト インスタンスにアクセスするには SSM を使用することに注意してください。

在庫

- AWS ネットワーク ファイアウォール ルール グループ: icmp-alert, domain-deny
- 検査 VPC
- サブネット: ネットワーク ファイアウォール サブネット A、ネットワーク ファイアウォール サブネット B

SSM でアクセスできないといけないのでインスタンスのサブネットに対して HTTPS(443) を許可する必要がある

※ リージョンによってアドレス帯がちがうので注意する

<https://ip-ranges.amazonaws.com/ip-ranges.json>

チャレンジ4

2023年9月7日 10:46

AWS Jam – チャレンジ4

まとめ

Sam は、クラウド ネイティブ アプリケーション開発者として IoT デバイス製造会社に入社しました。アプリケーション チームは、Amazon API Gateway と AWS Lambda を使用して、安全でない REST API をいくつか構築しました。

Sam の最初のタスクは、これらの API に対して JSON Web Token (JWT) トークンベースの承認を強制することでした。彼女は、Lambda オーソライザーを使用して API の 1 つに概念実証 (PoC) を実装しました。

ただし、PoC のデプロイメントにより API がエラーを返す結果になりました。

課題は、API が再び期待どおりに機能するように、API Gateway 構成のトラブルシューティングを行うことです。

在庫

- ・「デバイス ID ジェネレーター」REST API
- ・ラムダ関数

API Gateway に認証機能を実装する 今回は lambda

モジュールエラー → レイヤー

構文エラー → 頑張る

オーソライザーの作成

名前 *

タイプ * ⓘ

 Lambda Cognito

Lambda 関数 * ⓘ

 ap-northeast-1

Lambda 呼び出しロール ⓘ

Lambda イベントペイロード * ⓘ

 トークン リクエスト

トークンのソース * ⓘ

トークンの検証 ⓘ

認可のキャッシュ ⓘ

 有効

TTL (秒)

 300

作成 キャンセル

コードの例

```
import json
import logging
import jwt

logger = logging.getLogger()

SECRET = 'my-secret'

def lambda_handler(event, context):

    try:
        username = event['username']
        password = event['password']
        if username == 'user01' and password == 'pass01':
            header = {
                "alg": "HS256",
                "typ": "JWT"
            }
            payload = {
                "sub": "1",
                "exp": 202301101800
            }
            encode_jwt = jwt.encode(payload, SECRET, algorithm='HS256',
headers=header)
            return {
                'token': encode_jwt
            }
        except:
            pass

    return {
        'message': 'Unauthorized'
    }
```

チャレンジ 5

2023年9月7日 10:46

AWS Jam – チャレンジ 5

まとめ

電子商取引サイトを運営するあなたの会社は、先週から断続的に攻撃を受けています。

セキュリティ チームのレポートによると、インターネット用の API にいくつかの脆弱性があります。具体的には SQL インジェクションです。

CISO から緊急メールを受け取りました。

電子メールによると、彼はあなたを、この問題を解決するための当社の最高の AWS スペシャリストとして任命したことです。新しい API への移行を今すぐ実行する必要があります。

問題を迅速に解決するには、Amazon API Gateway、AWS WAF などの一部の AWS サービスを使用します。

在庫

- アプリケーションロードバランサ
- ラムダ関数

SQL インジェクション用の WAF を作成して API ゲートウェイに割り当てる

チャレンジ 6

2023年9月7日 10:46

AWS Jam – チャレンジ 6

まとめ

あなたは、会社の AWS 認定セキュリティ管理者として新たに雇用されました。

会社の AWS アカウントには多数のサービスがあり、それらすべてが 1 つの KMS キーを使用していることに気づきました。

Amazon Simple Storage Service (S3) と Amazon Simple Queue Service (SQS) が、まったく同じ Key Management Service (KMS) キーを使用してオブジェクトを暗号化していることがわかります。

あなたはセキュリティのベスト プラクティスを認識しており、異なるサービスで同じキーを使用するのをやめたいと考えています。異なるキーを使用すると、KMS キーへの不正アクセスのリスクが軽減されます。

今回のチャレンジでは

1. 1 つのサービスに対して 1 つのキーを実装することで、セキュリティの危険を制限します。
2. リソース ポリシーを編集して、公開を制限するためにキーを 1 つのアカウントのみに制限します。
3. 最後に、キーのローテーションを有効にします。

この課題を解決して、AWS アカウントのセキュリティ体制を強化してください。

在庫

- アマゾン S3
- AWS KMS
- Amazon SQS キュー

s3 デフォルトの暗号化を変更する

sqs

Amazon SQS > キュー > lab-queue-01

lab-queue-01

編集 削除 クリア メッセージを送受信 DLQ 再処理の開始

詳細 情報

| | | |
|-------------------------|--|--|
| 名前 | タイプ | ARN |
| lab-queue-01 | 標準 | arn:aws:sqs:ap-northeast-1:608728620263:lab-queue-01 |
| 暗号化 | URL | デッドレターキュー |
| Amazon SQS キー (SSE-SQS) | https://sqs.ap-northeast-1.amazonaws.com/608728620263/lab-queue-01 | - |

▶ さらに表示

SNS サブスクリプション | Lambda トリガー | デッドレターキュー | モニタリング | タグ付け | アクセスポリシー | **暗号化** | デッドレターキューの再処理タスク

暗号化 情報

Amazon SQS は、デフォルトで転送時の暗号化が実施されます。サーバー側の暗号化 (SSE, Server-Side Encryption) をキューに追加することもできます。つまり、SQS によって SQS サーバー上に保管中のすべての顧客データが暗号化されるということです。

サーバー側の暗号化は SSE-SQS によって管理されます

SQS/KMS/無効の間で使用されるサーバー側の暗号化を選択できます

編集

キーローテーションの設定

539ac0ca-05ab-4884-817a-ea2e66e008b8

一般設定

| | |
|--|-------|
| エイリアス | ステータス |
| lab1-key | 有効 |
| ARN | 説明 |
| arn:aws:kms:ap-northeast-1:608728620263:key/539ac0ca-05ab-4884-817a-ea2e66e008b8 | - |

[キーポリシー](#) | [暗号化設定](#) | [タグ](#) | [キーローデーション](#) | [エイリアス](#)

キーローデーション

この KMS キーを毎年自動的にローデーションします。詳細は[こちら](#)

チャレンジ7

2023年9月7日 10:46

AWS Jam - チャレンジ7

まとめ

あなたの会社は、開発者に IAM ロールを作成する機能を与えています。一部の異端児開発者が、開発者ロールに課された制限を回避する特権アクセスを持つロールを作成したことが判明しました。セキュリティ チームは、開発者の不適切な権限昇格を防ぐための創造的な方法を考え出すために、皆さんの協力を求めてきました。

在庫

- EC2: 管理ホスト
- IAM アクセス許可境界ポリシー: NoPrivilegeEscalation-PermissionsBoundary-REGION
- EC2 の IAM ロール: oAdminHostRoleName の出力プロパティを参照してください。
- CloudTrail: jam-iam-cloud-trail

CloudTrailを使用して作成されたロールを確認して削除する

| イベント名 | イベント時間 | ユーザー名 | イベントソース |
|---------------|------------------------------------|----------|---------------------------|
| ListResources | September 07, 2023, 14:38:53 (...) | labuser1 | ram.amazonaws.com |
| DescribeHub | September 07, 2023, 14:38:53 (...) | labuser1 | securityhub.amazonaws.com |

許可境界ポリシー：許可ポリシーよりも強い権限を設定することができる

例) 許可の境界で s3 にアクセスしないように設定したらポリシーでアクセス可能にしてもアクセスできない



<https://dev.classmethod.jp/articles/iam-policies-evaluation-logic-rikai/>

どちらでも許可されていることしかできない

▼ アクセス管理
ユーザーグループ
ユーザー
ロール
ポリシー
ID プロバイダ
アカウント設定
▼ アクセスレポート
アクセスアナライザー

AmazonSSMManagedInstanceCore

許可の境界 – (not set) 情報
許可の境界を設定して、このロールが持つことのできる許可の上限を制御します。これは一般的な設定ではありませんが、許可の管理を他のユーザーに委任するために使用できます。

許可の境界を設定

他のロールにスイッチできないように設定する必要がある



<https://www.yamamanx.com/iam-permission-boundary-role/>

ロール名 : DelegatedRoleBoundary

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "iam:DeleteRole",  
                "iam:AttachRolePolicy",  
                "iam:DeleteRolePolicy",  
                "iam:DetachRolePolicy",  
                "iam>CreateRole",  
                "iam:UpdateRole*",  
                "iam:PutRolePolicy"  
            ],  
            "Resource": "*",  
            "Condition": {  
                "StringEquals": {  
                    "iam:PermissionsBoundary":  
                        "arn:aws:iam::608728620263:policy/LabPermissionBoundaryPolicy"  
                }  
            }  
        },  
    ],  
}
```

```
{
    "Effect": "Allow",
    "Action": [
        "cloudwatch:*",
        "iam:Get*",
        "iam>CreatePolicy",
        "iam>DeletePolicy",
        "iam>DeletePolicyVersion",
        "iam>List*",
        "iam:SetDefaultPolicyVersion",
        "iam:SimulatePrincipalPolicy",
        "iam:SimulateCustomPolicy"
    ],
    "NotResource": "arn:aws:iam::608728620263:user/labuser1"
},
{
    "Effect": "Deny",
    "Action": [
        "iam>CreatePolicyVersion",
        "iam>DeletePolicy",
        "iam>DeletePolicyVersion",
        "iam:SetDefaultPolicyVersion"
    ],
    "Resource": [
        "arn:aws:iam::608728620263:policy/LabPermissionBoundaryPolicy",
        "arn:aws:iam::608728620263:policy/DelegatedRoleBoundary"
    ]
},
{
    "Effect": "Deny",
    "Action": "iam>DeleteRolePermissionsBoundary",
    "Resource": "*"
}
]
```

Denyで自分でポリシーや許可境界ポリシーを削除できないようにしている

チャレンジ8

2023年9月7日 10:47

AWS Jam - チャレンジ8

まとめ

Github リポジトリでアプリケーション コードの脆弱性を確認していたときに、プレーン テキストで IAM アクセス キーのペアを見つけました。どのようにしてそこに到達したかの調査を開始するには、IAM ユーザーとそれに関連付けられている AWS アカウント番号を特定する必要があります。

在庫

- IAM キーペア

特定する必要がある

Trusted Advisor から

The screenshot shows the Trusted Advisor interface with the following details:

- Left Sidebar:** Includes categories like 優先度, レコメンデーション, コスト最適化, パフォーマンス, **セキュリティ** (which is selected), 耐障害性, and サービスの制限.
- Engage Section:** Contains links for Trusted Advisor を管理 and 通知.
- Top Right Panel:** Title "セキュリティチェック". Filter sections for タグキー and タグ値, and buttons for リセット and 送信. Search fields for キーワードで検索 (with 情報 link) and ソース (with 表示 link). A dropdown for 全てのソース and another for 全てのチェック.
- Findings List:**
 - IAM アクセスキーローテーション:** Description: 過去 90 日間ローテーションが行われていないアクティブな IAM アクセスキーを 2 個のアクティブアクセスキーのうち 1 個が過去 90 日間ローテーションされてい
 - 漏洩したアクセスキー:** Description: 一般的に漏洩してしまったアクセスキーおよび悪用されたアクセスキーが原因で発生するAmazon Elastic Compute Cloud (Amazon EC2) の不規則な使用状況について、頻繁に利用されているコードリポジトリをチェックします。

チャレンジ 9

2023年9月7日 10:47

AWS Jam - チャレンジ 9

まとめ

あなたは最近、Best FinServe Corp (大手金融サービス会社) にデータ サイエンティストとして採用されました。Best FinServe はデジタル変革に着手しており、2 年以内に手動プロセスの 80% を自動化するという目標を設定しています。あなたの上司は、SEC Form S-1 提出書類から情報を自動的に抽出する POC ソリューションを開発するようにあなたに依頼しました。SEC フォーム S-1 は、米国での新規株式公開 (IPO) 前に企業が証券取引委員会 (SEC) に提出する登録届出書です。SEC フォーム S-1 には、売り出し株式数や株式数などの情報が含まれています。公開価格は投資家にとって非常に貴重な情報です。

締め切りが厳しいため、チームは注釈付きのドキュメントを 300 件しか取得できませんが、モデルを最初からトレーニングするには十分ではない可能性があります。自然言語処理 (NLP) ドメインの経験は限られているため、データ サイエンティスト向けに NLP を簡素化する AI サービスから大きなメリットを得ることができます。

在庫

- SageMaker ノートブック インスタンス
- 部分的に埋められた Jupyter Notebook

S3バージョニング

2023年9月7日 14:58

上書きされた時に前の情報を確認できる

Amazon S3 > バケット > ahaws-bucket-01 > バケットのバージョニングを編集

バケットのバージョニングを編集 情報

バケットのバージョニング

バージョニングは、オブジェクトの複数のバリアントを同じバケット内に保持する手段です。バージョニングを使用すると、Amazon S3 バケットに格納されているすべてのオブジェクトのすべてのバージョンを保存、取得、復元できます。バージョニングを使用すると、意図しないユーザーアクションと意図しないアプリケーション障害の両方から簡単に復旧できます。[詳細](#)

バケットのバージョニング

停止

これにより、すべてのオペレーションに対してオブジェクトバージョンの作成が停止されますが、既存のオブジェクトバージョンはすべて保持されます。

有効にする

Multi-Factor Authentication (MFA) の削除

バケットのバージョニング設定を変更し、オブジェクトバージョンを完全に削除するために多要素認証を必要とする追加のセキュリティレイヤーです。MFA の削除設定を変更するには、AWS CLI、AWS SDK、または Amazon S3 REST API を使用します。[詳細はこちら](#)

無効

[キャンセル](#)

[変更の保存](#)

暗号化

2023年9月11日 15:24

EC2を暗号化

2023年9月7日 15:00

スナップショットを作成する

スナップショットをコピーし、暗号化にチェックをいれる

コピーしたものから立ち上げる

EFSの暗号化

2023年9月7日 15:05

backup vault