



4



1



この記事は最終更新日から1年以上が経過しています。



@hibohiboo

投稿日 2022年04月19日

# cdkで不要なルートテーブルを作らずにvpcを作成したメモ



aws-cdk

## 概要

前回作成したVPCだと、サブネットごとにデフォルトのルートテーブルができていた。プライベートは全て同じルートテーブルを使いたい、L2のクラスのプロパティにそれらしいものを見つけられなかった。

[stackoverflow](#)でもL1使うしかないと思うとあったので、あきらめてL1を使うことにした。

ソースコード

## ソース

```
import { Aspects, Stack, StackProps, Tag, Tags } from 'aws-cdk-lib';
import { Construct } from 'constructs';
import { CfnInternetGateway, CfnRoute, CfnRouteTable, CfnSubnet, CfnSubnetRouteTableAssociation, CfnVPCGatewayAttachment, Priv

interface VpcStackProps extends StackProps {

}

export class VpcRouteTestStack extends Stack {
  constructor(scope: Construct, id: string, props: VpcStackProps) {
    super(scope, id, { ...props, subnetGroupName: undefined } as StackProps);
    const cidr = '10.1.0.0/16';
    const vpc = new Vpc(this, 'VPC', {
      cidr,
      natGateways: 0, // デフォルトは1。
      subnetConfiguration: [], // サブネットの自動作成はなし。
    })
    // Tags.of(vpc).add('Stack', id);
    Tags.of(vpc).add('Name', 'vpc');

    const publicSubnet = new CfnSubnet(this, "lab-subnet", {
      vpcId: vpc.vpcId,
      cidrBlock: "10.1.0.0/24",
      availabilityZone: this.availabilityZones[0],
      tags: [{ key: "Name", value: "lab-subnet" }]
    });

    const igw = new CfnInternetGateway(this, "lab-igw", {
      tags: [{ key: "Name", value: "lab-igw" }]
    });

    const igwAttach = new CfnVPCGatewayAttachment(this, "lab-igw-attach", {
      vpcId: vpc.vpcId,
```

```
internetGatewayId: igw.ref
});

const publicRouteTable = new CfnRouteTable(this, "lab-public-route", {
  vpcId: vpc.vpcId,
  tags: [{ key: "Name", value: "public-rt" }]
});

const igwRoute = new CfnRoute(this, "lab-public-route-igw", {
  routeTableId: publicRouteTable.ref,
  destinationCidrBlock: "0.0.0.0/0",
  gatewayId: igw.ref
});

const association = new CfnSubnetRouteTableAssociation(this, "lab-public-route-association", {
  routeTableId: publicRouteTable.ref,
  subnetId: publicSubnet.ref
});

// -----
// private
const privateRouteTable = new CfnRouteTable(this, "lab-private-route", {
  vpcId: vpc.vpcId,
  tags: [{ key: "Name", value: "private-rt" }]
});

// プライベートSubnet
const privateSubnetProps: PrivateSubnetProps[] = [
  { availabilityZone: 'ap-northeast-1a', vpcId: vpc.vpcId, cidrBlock: '10.1.1.0/24', },
  { availabilityZone: 'ap-northeast-1c', vpcId: vpc.vpcId, cidrBlock: '10.1.2.0/24', },
  // ap-northeast-1b は使えない
  { availabilityZone: 'ap-northeast-1d', vpcId: vpc.vpcId, cidrBlock: '10.1.3.0/24', },
]
```

```

const subnets = privateSubnetProps.map((prop, i) => {
  const subnet = new CfnSubnet(this, `MyPrivateSubnet${i}`, {
    vpcId: vpc.vpcId,
    cidrBlock: prop.cidrBlock,
    availabilityZone: prop.availabilityZone,
    tags: [{ key: "Name", value: `MyPrivateSubnet${i}` }]
  });
  Tags.of(subnet).add('Name', `private-subnet-${i}`);
  Tags.of(subnet).add('aws-cdk:subnet-type', SubnetType.PRIVATE_ISOLATED);
  new CfnSubnetRouteTableAssociation(this, `MyPrivateSubnet-associations-${i}`, {
    routeTableId: privateRouteTable.ref,
    subnetId: subnet.ref
  })
  return subnet
});

//----- 共通設定 -----
// 作成したリソース全てにタグをつける
Aspects.of(this).add(new Tag('Stack', id));

}
}

```

## 参考

---

CDK(TypeScript)で自分好みのVPCを作りたい

[初心者向け] AWS CDKのレイヤーについて調べてみた



## 新規登録して、もっと便利にQiitaを使ってみよう

1. あなたにマッチした記事をお届けします
2. 便利な情報をあとで効率的に読み返せます

[ログインすると使える機能について](#)

新規登録

ログイン



ひぼ

@hibohiboo

Javascriptが好きです。

フォロー





## コメント

この記事にコメントはありません。

あなたもコメントしてみませんか :)

新規登録

すでにアカウントを持っている方は[ログイン](#)

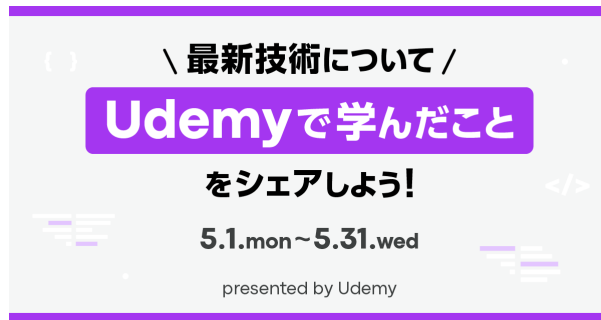
## 記事投稿キャンペーン開催中



### 【RubyKaigi 2023連動イベント】 みんなでRubyの知見を共有しよう

2023/05/01~2023/05/31

[詳細を見る](#)



### 最新技術についてUdemyで学んだことをシェアしよう！

2023/05/01~2023/05/31

[詳細を見る](#)

[すべて見る](#) ➔

# Qiita

How developers code is here.

ガイドとヘルプ	コンテンツ	SNS	Qiita 関連サービス	運営
About	リリースノート	 Qiita（キータ）公式	Qiita Team	運営会社
利用規約	公式イベント	 Qiita マイルストーン	Qiita Jobs	採用情報
プライバシーポリシー	公式コラム	 Qiita 人気の投稿	Qiita Zine	Qiita Blog
ガイドライン	募集	 Qiita（キータ）公式	Qiita 公式ショップ	
デザインガイドライン	アドベントカレンダー			
ご意見	Qiita 表彰プログラム			
ヘルプ	API			
広告掲載				