AWSでのインスタンス構築初歩

entry point

Table of Contents

初歩的なAWS

こんにちは。 以前Qiitaにてメモ書きとして掲載(http://qiita.com/futoase/items/9c23306c1db790f35b87)していたものをまとめていたことをまとめ直してみました。

前提としていること

- すでにAWSにアカウントを作成している
- すでにAWSに登録しているアカウントに対してクレジットカードの関連付けを行っている
- AWS Management Consoleが何かわかっている
- AWS Management Consoleへのアクセスには一般のウェブブラウザを利用する
- APIを基本とした動作、設定については書いていない
- やろうとしてることは、僅かなお金とはいえ、コストがかかるということ。
- AWSのEC2インスタンスCPU利用料金の無償枠があるものの、コストがかかります！注意！

この文書で保証しないこと

- 作成したEC2インスタンス、RDSなどのAWSを利用することにかかった料金は保証しない。
- 第三者からのEC2インスタンス及びVPC内への攻撃
- 作成したEC2インスタンスの運用保守に関しての責任

VPCとは

VPCとは、EC2インスタンス、RDSなどのサービスのネットワークセグメントを 仮想的に、扱い的には物理的に切り離すことが可能になるネットワークの単位。 2012年ごろから使えるようになりました。VPC間での通信についても設定可能。 VPC内にEC2、RDS、AWS Lambdaなどのサービスを動かすようにします。

EIPを与えることでVPC外部からの通信も可能になります。 (IAMを使えばAWS API経由でインスタンス操作も可能になります)

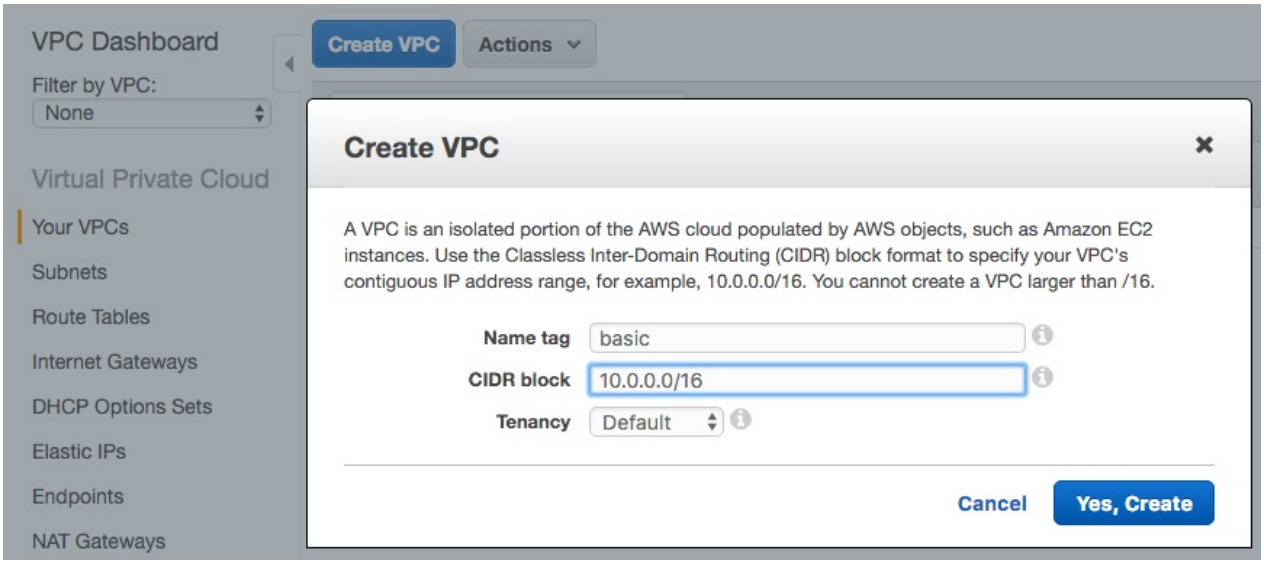
VPCについて以下に詳しく載っています。

https://aws.amazon.com/jp/vpc/

VPCを作る

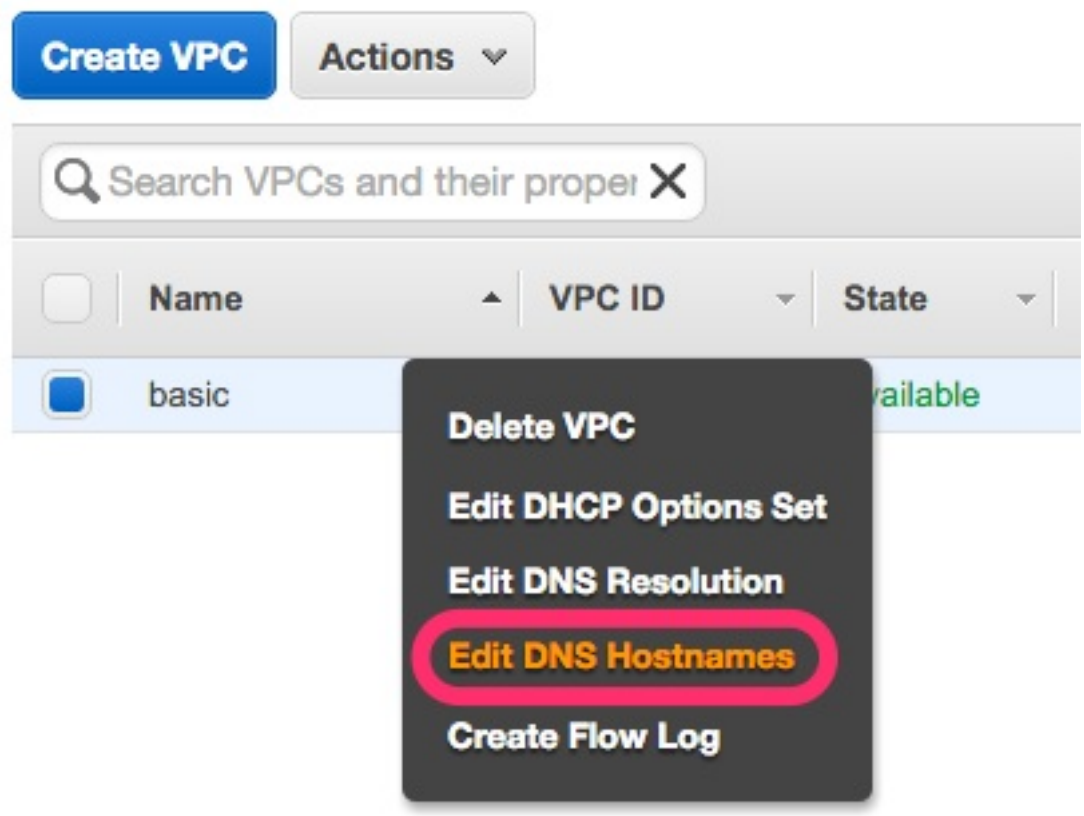
AWS Management Consoleにて、VPCを選択します。 VPCってなんぞ？と思いつつも従っていけばきっとVPCを構築できるようになっているはず！やっていきましょう。

Create VPCボタンを押して、VPCを作成します。ここでの名前"basic"とします。

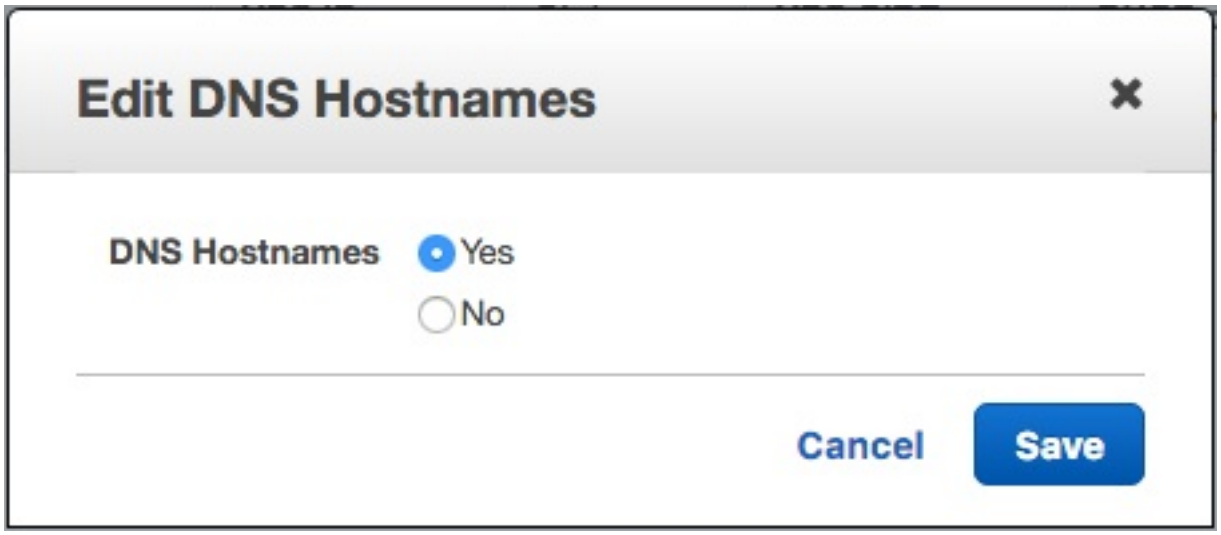


Private DNS、つまるところAWSで作成した他のホストについてから ホスト名を引けるようにするため、Private DNSを有効にします。

作成したVPCを一覧より選択し、右クリック(かそれに相当するジェスチャ)で "Edit DNS Hostnames"オプションを表示します。



"Edit DNS Hostanames"の"DNS Hostnames"が "YES"と設定し保存すればOK。



では、次に移ります(｀･ω･´)ゞ

VPCで利用するためのSubnetを切る

Subnetについてはネットワーク構成の単位として扱います。 通常のネットワーク構成を作る際に設定するサブネットマスクと同じと考えるといいかもしれません。

Subnetについて詳しく知りたい場合は、 以下の文書を参考してください。

http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Subnets.html

今回の例で切っているSubnetの注意点

1000台以上のホストを同じSubnetに属させる...とかは考えず、ミニマムな構成でSubnetを切っています。これを元にサービス用にSubnetを設計すると設定できるホストが不足します。注意！

各Subnetを切っていく

切っていきます(｀･ω･´)ゞ

作成しようとしているSubnet

テーブル構造にて今回作ろうとしているSubnetを示します。

Subnet	役割	Availability Zone
10.0.0.0/27	Managed NAT Gateway	ap-northeast-1a
10.0.5.0/27	EC2(踏み台)	ap-northeast-1a
10.0.10.0/27	ELB	ap-northeast-1a
10.0.10.32/27	ELB	ap-northeast-1c
10.0.11.0/24	EC2	ap-northeast-1a
10.0.12.0/24	EC2	ap-northeast-1c
10.0.15.0/27	RDS	ap-northeast-1a
10.0.15.32/27	RDS	ap-northeast-1c
10.0.16.0/27	ElastiCache	ap-northeast-1a
10.0.16.32/27	ElastiCache	ap-northeast-1c

Managed NAT Gatewayってなあに？

役割の中に載っているManaged NAT Gatewayについて。 以前はNATについては、EC2インスタンスから作成しなければだめでした。が、程なくAWSさんが対応するサービスを発表しました。それがManaged NAT Gatewayです。

NATをワンクリックで作成できます。ENIも作ってくれます＾＾

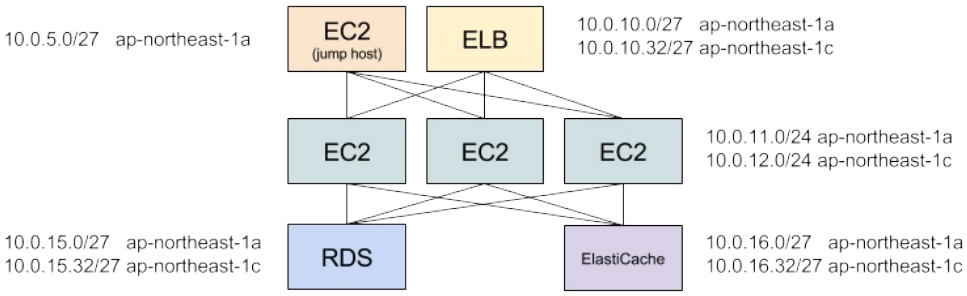
http://aws.typepad.com/aws_japan/2015/12/managednat.html

NATインスタンス自分で作りたいんだけど？

NATを自前で作りたい！という場合は 提供されているAMI(http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_NAT_Instance.html)から作成するか、VyOS(http://vyos.net/wiki/Main_Page)のAMIイメージ(コミュニティのもの)を利用して構築してみてください。個人的にVyOS構築がおすすめです。

作ったSubnetってどういう感じで使われるの？

について図を作りました！ こんな感じで各ホストを繋いでいきます！



ap-northeast-1a, ap-northeast-1c 2つのAvailability Zoneを指定し、冗長化を図っています。

AWSのSubnetでのCIDRについて最少はいくつ？

Subnetについては予約済みのアドレスが既に含まれていてSubnetについては予約済みのアドレスが既に含まれていて、AWSの場合すでに5つが予約済みとなっているためです。 /28がAWSで利用できるSubnetのCIDRでの最少数となっていることが多いです。

通常のサービスであれば/24(250台ほどのEC2ホスト)あればよく、すごく繁盛してるサービスなら/23(1010台ほどのEC2ホスト...！)が必要になったりします。

詳しくは以下のドキュメントを参考してください。

http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Subnets.html#SubnetSize

Subnetとルーティングテーブルを結びつける

作成したSubnetとルーティングテーブルを結びつけてインスタンス間・サービス間(RDS, Elasticache)で疎通が行えるようにしていきます。

ルーティングテーブルについては以下の文書を参考にしてください。

http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Route_Tables.html

Internet Gatewayを作る！

Internet Gatewayを作って、EC2 インスタンスがインターネットに接続できるようにします。また、Managed NAT Gatewayも利用できるようにし、Managed NAG Gateway経由でインターネットと隔離しているEC2 インスタンスもインターネット側と通信できるようにします。

Internet Gatewayについて詳しく知りたい人はこちらを参考にしてください。

http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Internet_Gateway.html

"basic"という名のInternet Gatewayをつくる

特に名前にこだわりは無いので、"basic"という名前のInternet Gatewayを作ります。Internet Gatewayの画面から、"Create Internet Gateway"ボタンを押して、"basic"という名前のInternet Gatewayを作ってください(｀・ω・´)ゞ

ルーティングテーブルを作成する

では、ルーティングテーブルを作っていきます！

2つのルーティングテーブル

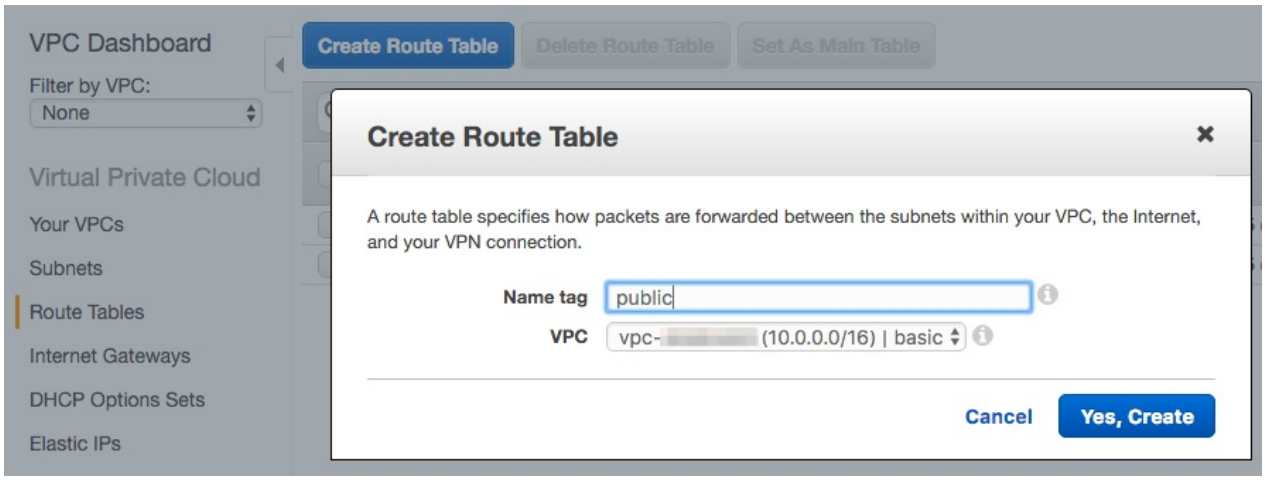
public用とprivate用、2つのルーティングテーブルを作成します。 役割は以下のとおりです

- publicは、インターネットとのやり取りを行うためのもの。
- privateは、サービス内での通信を行うためのもの。

public側にはManaged NAT Gatewayや踏み台サーバ、private側にはEC2インスタンスやRDSインスタンスを配置します。

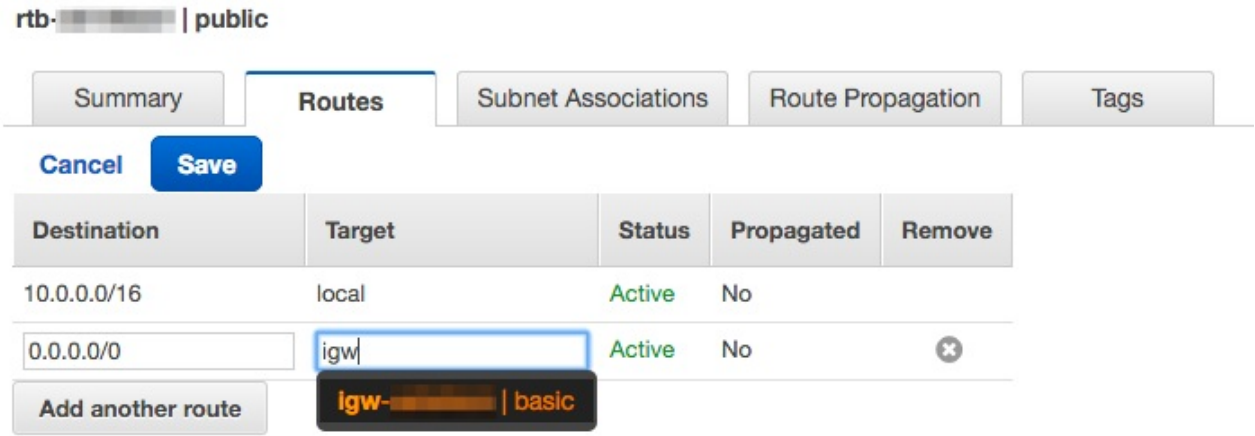
早速ルーティングテーブルを作る

ルーティングテーブルを作成します。public, privateについてそれぞれ作ります。ルーティングテーブル画面にて"Create Route Table"ボタンを押してください。VPCは先ほど作った"basic"を選んでください。



"public" Route Tableに対するInternet Gatewayの設定

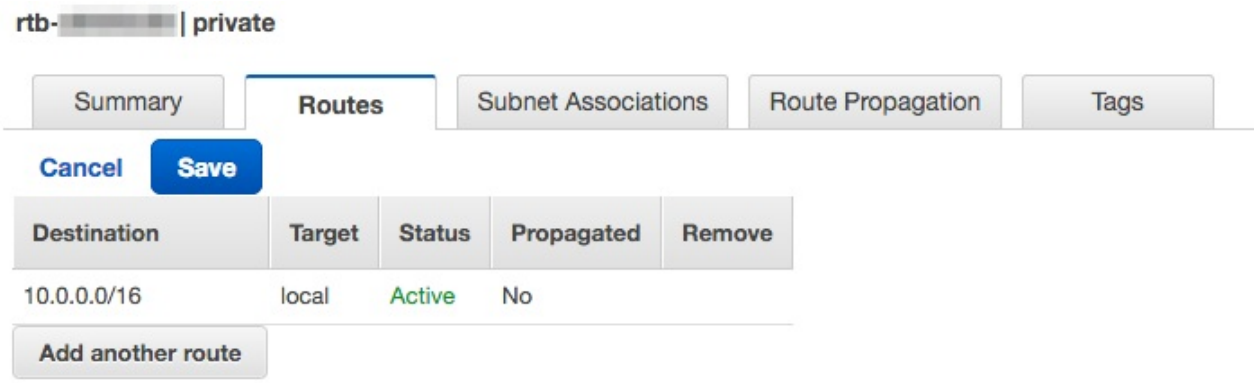
"public" Route Tableに対してInternet Gatewayの設定を行います。



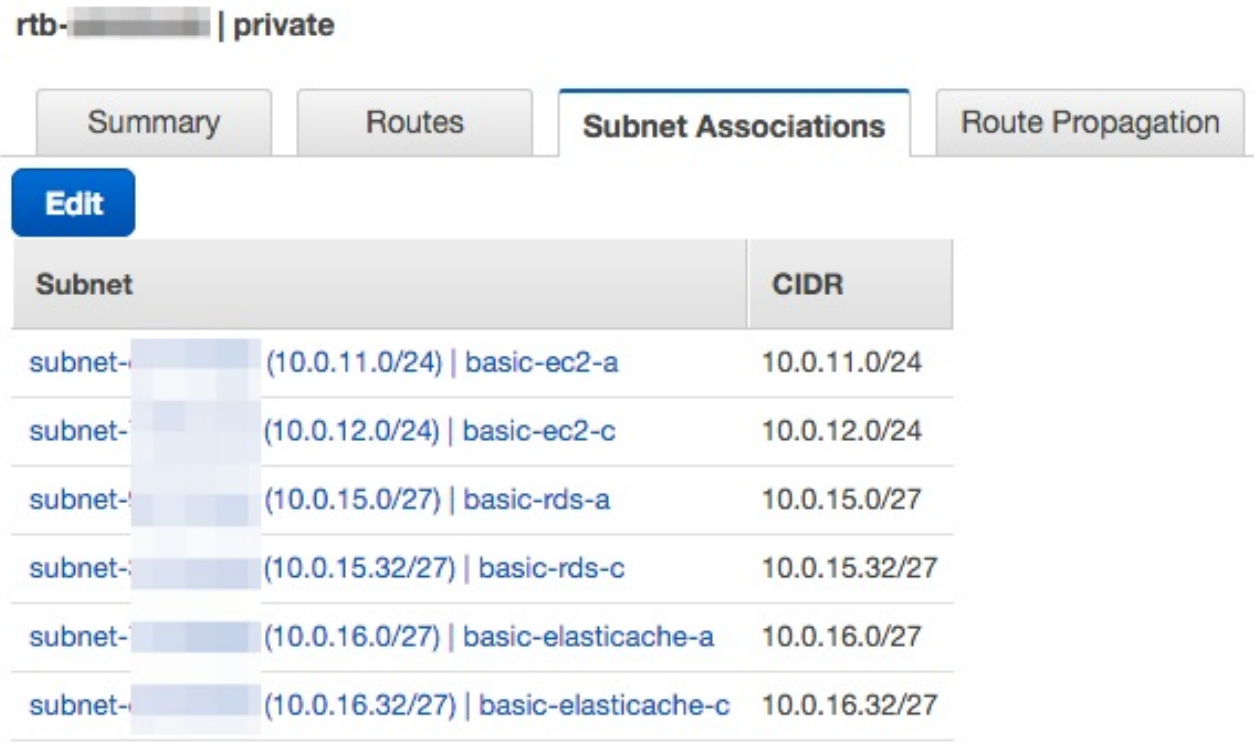
これにより、"public" Route Tableに割り当てられたEC2インスタンス(踏み台サーバ)やManaged NAT Gatewayがインターネットに直接でられるようになります。(踏み台サーバについては後述。)Subnetがどのルーティングテーブルとの結びつけについてSubnet Associationsの設定によって設定します。 Managed NAT Gatewayなどが含まれてるかよく見ましょう。

"private" Route Table

"private" 側のルーティングテーブルについてもInternet Gatewayの設定を行います。



"private" Subnet側ではAWS内でのみ通信のやり取りを行うホスト、サービス(ElastiCacheなど)のSubnetの設定を行います。サービスを稼働させるインスタンスなどは"private"側のSubnetに属させます。



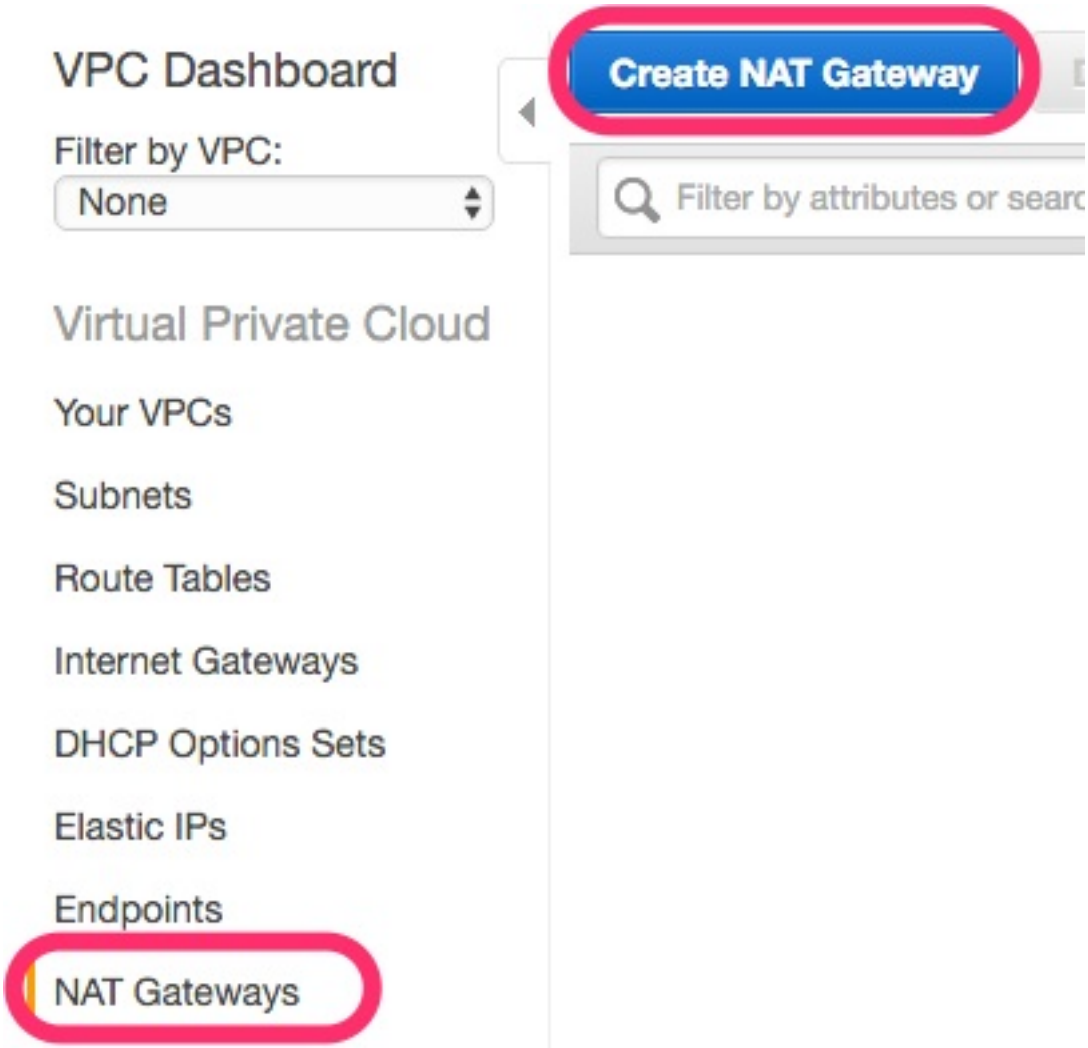
Managed NAT Gatewayを作る

Managed NAT Gatewayは、2015年12月からAWSにて利用が可能になったNATを気軽に作成できるサービスです。これでNAT インスタンスを用意するのがだいぶ楽になりました。閉じてるネットワークがインターネットにアクセスしたい場合に構築が都度必要だったので。こういうのどんどん楽になっていきますね。

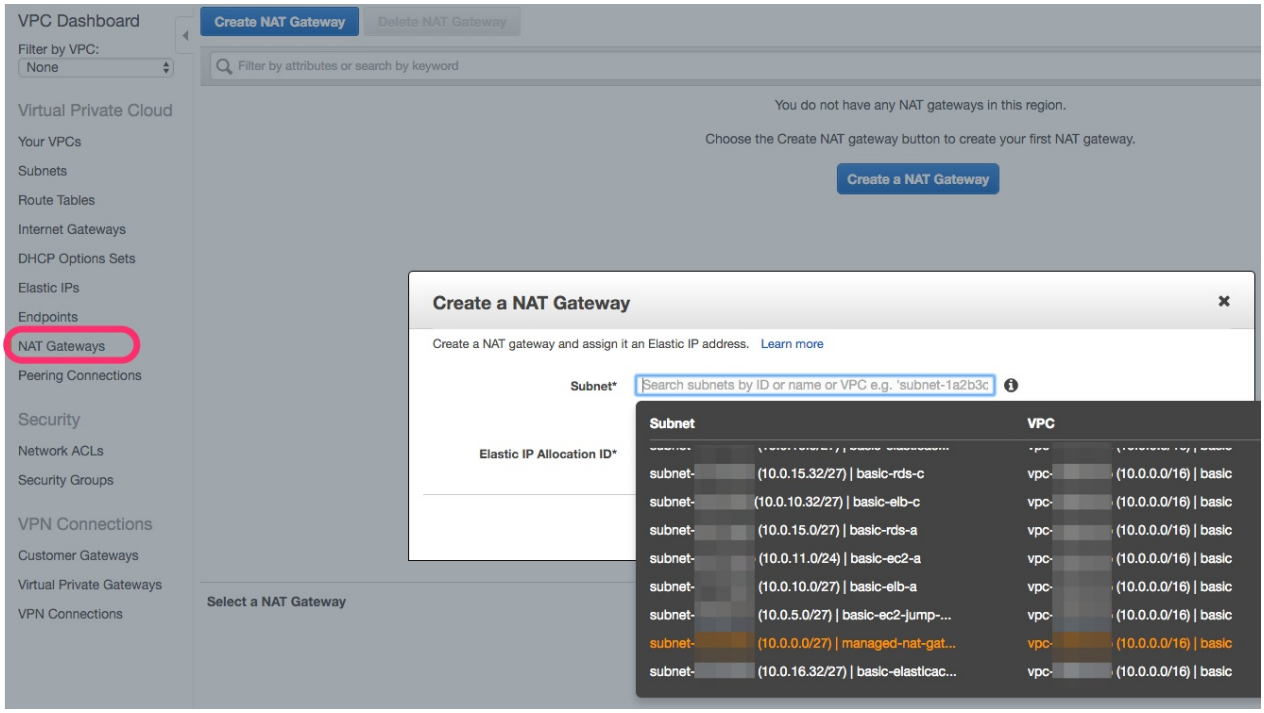
早速作っていきましょう。 以下の文書を参考に作ってます。

http://aws.typepad.com/aws_japan/2015/12/managednat.html

VPC Dashboardより、NAT Gatewaysを選択し、"Create NAT Gateway"ボタンを押します。

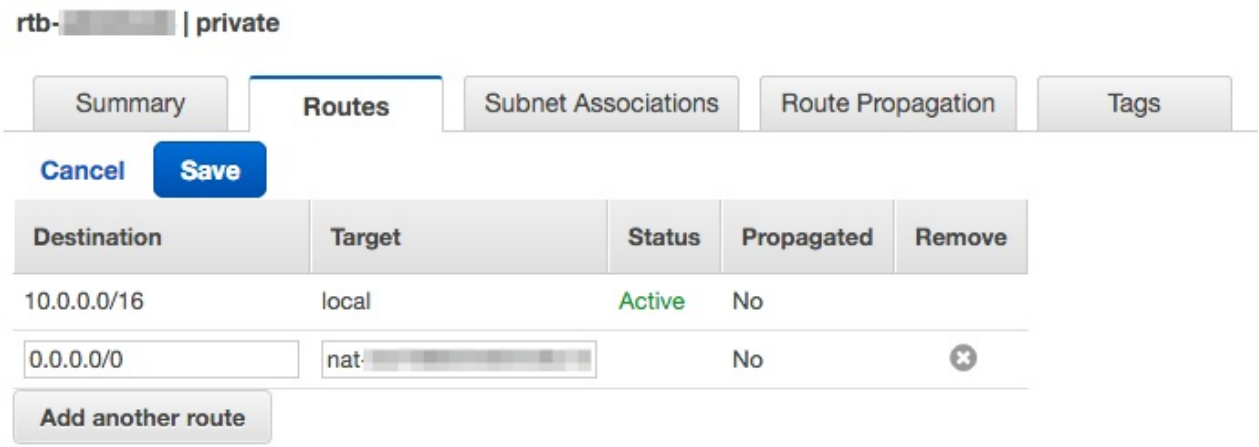


内容に問題がないのであれば、"Create a NAT Gateway"ボタンを押して作成します。その後、選択するSubnetについて"managed-nat-gateway"を選択してください。



"private" Route tableのRoutesに Managed Nat Gatewayを設定

"private" Route tableが利用するためのNATインスタンスを先ほど作成した"private" Route TableのRoutesに設定を行います。



これでManaged NAT Gatewayを作成できました。