@TK1989
投稿日 2016年10月24日

AWS NAT構成の作り方(NATゲートウェイ編)

AWS

AWSでのNAT接続を実現する方法を備忘を兼ねて記載。

NATインスタンス編はこちら

NAT構成の必要性

簡単にいうと、
インターネットから接続される必要のないインスタンスについて、
インターネットからの接続を遮断しつつ、
自身はインターネットに接続を出来るようにするため。
外部から接続される危険性を減らすことと、
ライブラリの取得などで必要になる外部への接続の両立が可能となる。

参考

シナリオ 2: パブリックサブネットとプライベートサブネットを持つ VPC（NAT）
http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/VPC_Scenario2.html

NATゲートウェイ
http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/vpc-nat-gateway.html

注意

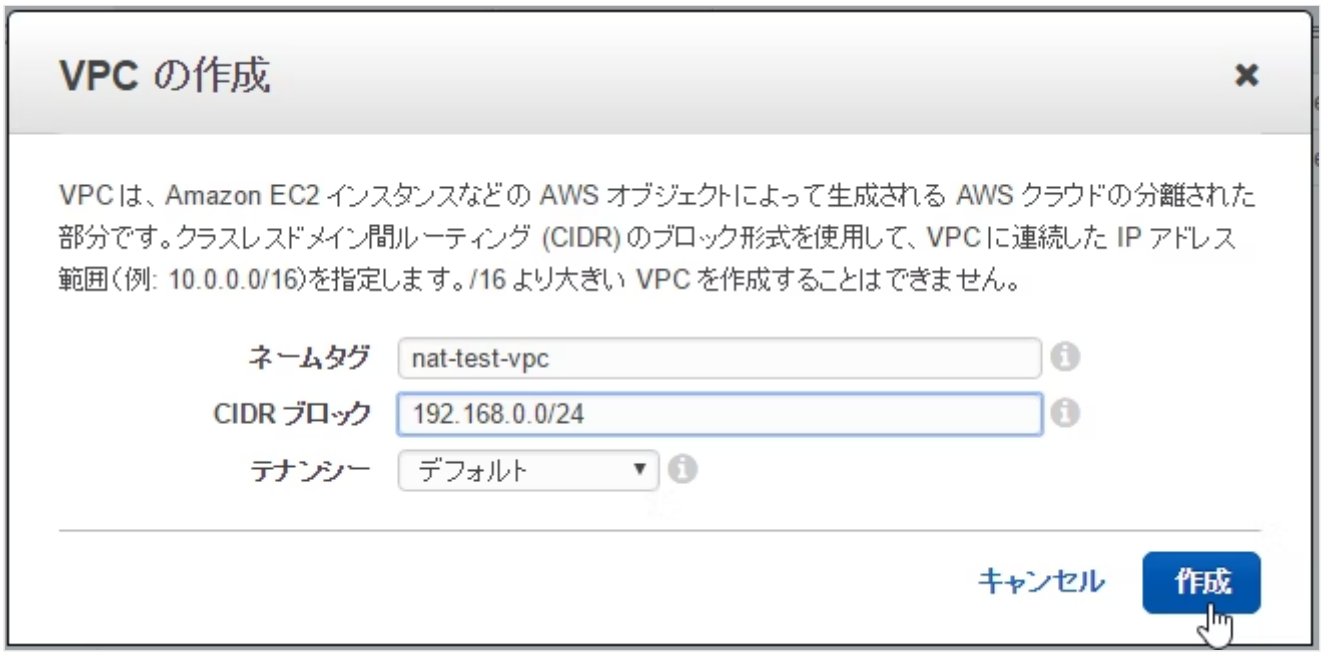
NATゲートウェイはAWSの12ヶ月無料利用枠の対象外です。(2016/10/23 現在)

NAT構成の作り方(NATゲートウェイ)

マネジメントコンソール(GUI)にて作成

VPC

項目名	設定値
ネームタグ	nat-test-vpc
CIDRブロック	192.168.0.0/24
テナンシー	デフォルト



サブネット作成

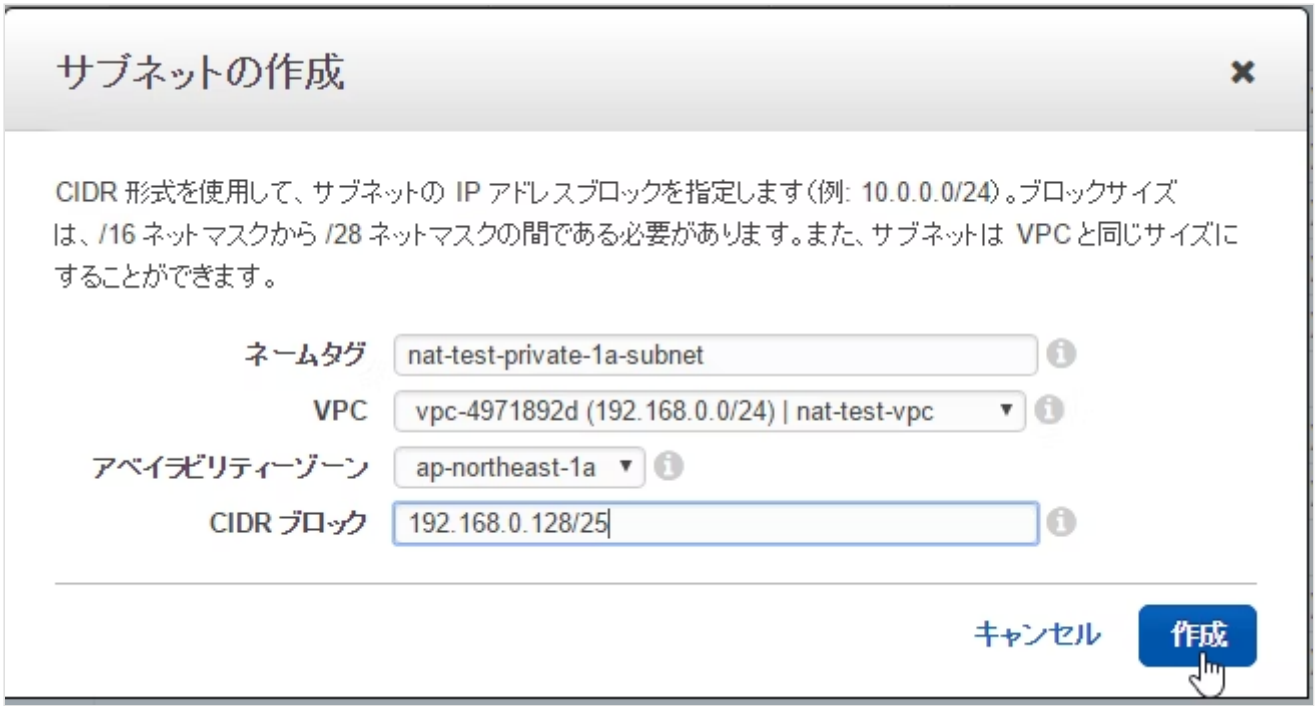
パブリック用サブネット作成

項目名	設定値

項目名	設定値
ネームタグ	nat-test-public-1a-subnet
VPC	nat-test-vpc
アベイラビリティーゾーン	ap-northeast-1a
CIDRブロック	192.168.0.0/25

プライベート用サブネット作成

項目名	設定値
ネームタグ	nat-test-private-1a-subnet
VPC	nat-test-vpc
アベイラビリティーゾーン	ap-northeast-1a
CIDRブロック	192.168.0.128/25



IGW(インターネットゲートウェイ)作成

インターネットゲートウェイ作成

項目名	設定値
ネームタグ	nat-test-igw

インターネットゲートウェイをVPCにアタッチ

項目名	設定値
VPC	nat-test-vpc

NATゲートウェイ

項目名	設定値
サブネット	nat-test-public-1a-subnet
Elastic IP 割り当て ID	新しいEIPの作成

ルートテーブル作成

カスタムルートテーブル作成

項目名	設定値
ネームタグ	nat-test-public-rt
VPC	nat-test-vpc

メインルートテーブル編集

ルート追加

ルートにNATゲートウェイを追加

送信先	ターゲット
0.0.0.0/0	nat-test-igw

プライベート用サブネット割り当て

サブネットの関連付けで、

nat-test-private-1a-subnet

を関連付ける。

ネームタグ変更

変更ついでにネームタグを

nat-test-private-rt

に変更して、

ルートテーブルの役割を名前から判別しやすくしておきます。

カスタムルートテーブル編集

パブリック用サブネット割り当て

サブネットの関連付けで、

nat-test-public-1a-subnet

を関連付ける。

ルート追加

ルートにインターネットゲートウェイを追加

送信先	ターゲット
0.0.0.0/0	nat-test-igw

セキュリティグループ

NAT接続インスタンス用セキュリティグループ

項目名	設定値
セキュリティグループ名	nat-test-ap-sg
説明	security group for ap
VPC	nat-test-vpc

インバウンド

タイプ	プロトコル	ポート範囲	送信元
SSH	TCP	22	踏み台サーバーのIP or セキュリティグループ

EC2作成

設定値(重要な箇所のみ抜粋)

項目名	設定値
AMI	なんでも (検証ではAmazon Linux：ami-1a15c77b (2016/10時点の東京リージョン最新Amazon Linux用AMI)を使用)
ネットワーク	nat-test-vpc
サブネット	nat-test-private-1a-subnet
自動割り当てパブリックIP	無効
セキュリティグループ	nat-test-ap-sg

動作確認

踏み台サーバーからNAT接続するインスタンスにログインし、
pingやcurlでインターネットに接続できることを確認する。

```
ping google.co.jp
```

```
curl http://google.co.jp/
```

```
[ec2-user@ip-192-168-0-222 ~]$ ping google.co.jp
PING google.co.jp (216.58.221.3) 56(84) bytes of data.
64 bytes from nrt13s38-in-f3.1e100.net (216.58.221.3): icmp_seq=1 ttl=55 time=2.08 ms
64 bytes from nrt13s38-in-f3.1e100.net (216.58.221.3): icmp_seq=2 ttl=55 time=1.88 ms
64 bytes from nrt13s38-in-f3.1e100.net (216.58.221.3): icmp_seq=3 ttl=55 time=1.89 ms
^C
--- google.co.jp ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 1.889/1.957/2.087/0.104 ms
[ec2-user@ip-192-168-0-222 ~]$
```

```
[ec2-user@ip-192-168-0-222 ~]$ curl http://google.co.jp/
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.co.jp/">here</A>.
</BODY></HTML>
[ec2-user@ip-192-168-0-222 ~]$
```

最後に

NATゲートウェイを作成することで、
NATインスタンスを使用するより(若干)簡単にNAT接続を行うことができます。

またAWSが用意するサービスであるので、
デフォルトで冗長性が担保されています。

では、NAT接続をすべてNATゲートウェイで行えばいいのかというと、
そういうわけでもないようです。

NAT インスタンスと NAT ゲートウェイの比較
http://docs.aws.amazon.com/ja_jp/AmazonVPC/latest/UserGuide/vpc-nat-comparison.html
によると、監視やカスタマイズを柔軟に行おうとするとNATインスタンスのほうが良いようです。(※よくわかっていない)

結局、双方のメリット・デメリットを見て、
どちらを選択するのかということになるようです。

※記述ミス・認識違いなどがあれば、ご指摘いただけると幸いです。