はじめに

Amazon Web Services(以降、AWS)の東京リージョンと大阪リージョンを利用したマルチリージョンのHAクラスター構築を試してみました。

2021/03/02にAWSの大阪リージョンが一般公開されました。
これまではAWSでマルチリージョンHAクラスターを構築する場合に、東京リージョンとシンガポールリージョンを組み合わせるなど、基本的に海外のリージョンを使用する必要がありましたが、今後は日本国内でマルチリージョンHAクラスターを構築することが可能になります。

今回はオンプレミス環境から、マルチリージョンHAクラスターにアクセスすることを想定して、以前のブログでご紹介した「VPC ピアリング接続を利用したHAクラスター」と「Route 53 リゾルバ―を使用したHAクラスターへの接続」を組み合わせて、東京リージョンと大阪リージョンで構築可能なマルチリージョンHAクラスターの構成をご紹介します。

大阪リージョンでは、AWS Transit Gatewayのピアリングアタッチメントがサポートされていないため、以前のブログでご紹介した「VIP制御によるマルチリージョンHAクラスター」は、大阪リージョンでは構築できませんのでご注意ください。(2021/04/20時点)

【参考】
AWS、国内 2 拠点目となるリージョンを開設

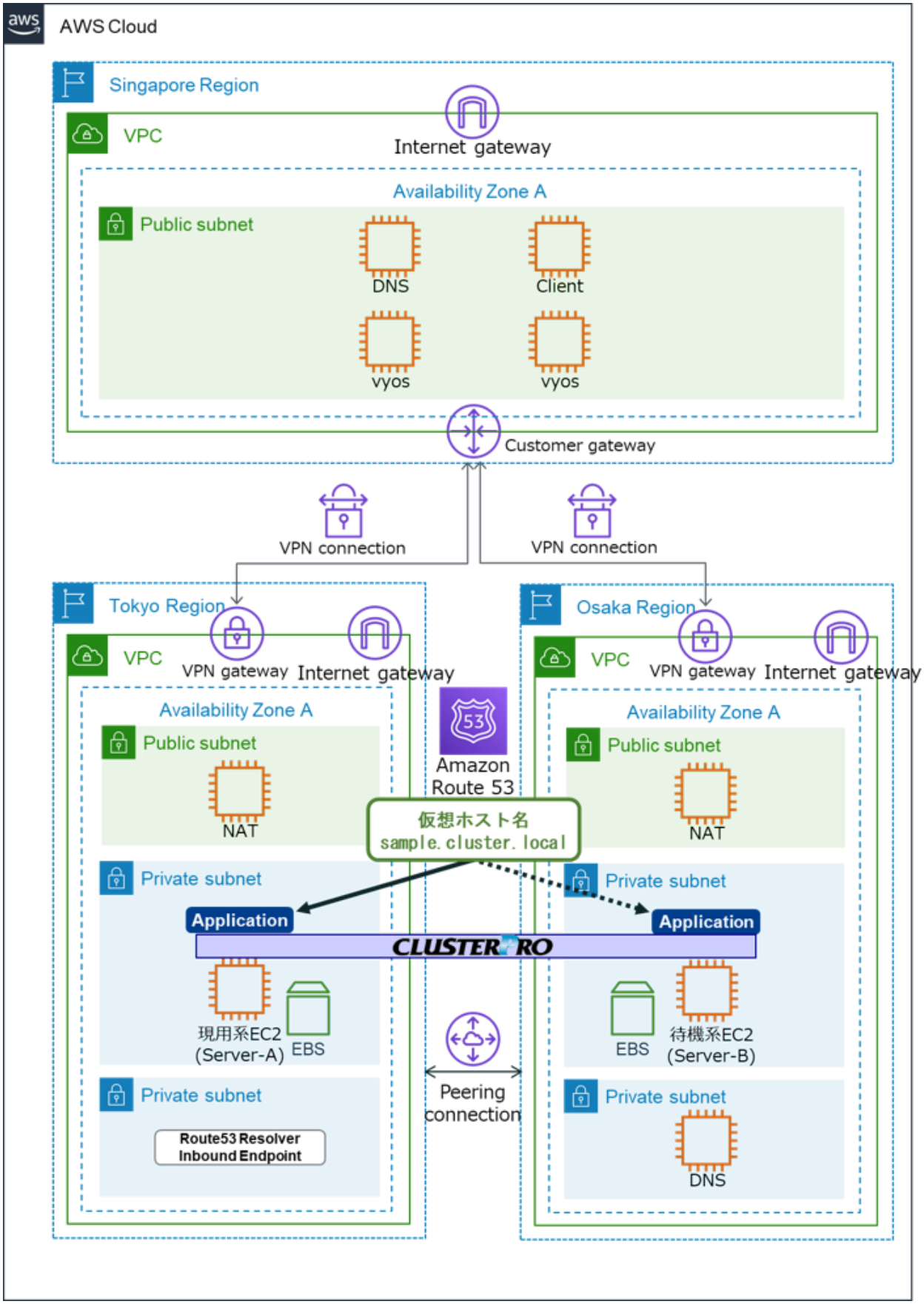
この記事の内容

1. HAクラスター構成

今回は、東京リージョンと大阪リージョンのVPC環境に「DNS名制御によるマルチリージョンHAクラスター」を構築します。
また、オンプレミス環境の疑似環境としてシンガポールリージョンにクライアントマシンを構築し、「シンガポールリージョンのVPC」から「東京リージョンのVPC」と「大阪リージョンのVPC」に対してVPNで接続します。

※ 実際にオンプレミス環境から接続する場合は、以降のシンガポールリージョンをオンプレミス環境に読み替えてください。

構築するHAクラスターの構成図は以下の通りです。



拡大表示

東京リージョンと大阪リージョンにそれぞれHAクラスターを構築するインスタンス（Server-A、Server-B）を配置します。

HAクラスターを構築するインスタンスは異なるリージョンに配置するため、VPC ピアリングを使用して、東京リージョンのVPCと大阪リージョンのVPC間を接続します。
HAクラスターを構成するインスタンスはプライベートサブネットに配置しますが、AWS CLIを利用したRoute 53の制御時にエンドポイントとの通信を行う必要があるため、それぞれのリージョンにパブリックサブネットを作成し、インターネットゲートウェイとNATインスタンスを追加しています。

接続先切り替えには、Route 53のプライベートホストゾーンと連携して仮想ホスト名による接続先切り替えを実現します。
シンガポールリージョンのクライアントマシンからRoute 53の仮想ホスト名を問い合わせるため、東京リージョンにRoute 53 リゾルバー、大阪リージョンにはDNSサーバーを作成します。
大阪リージョンでは、Route 53 リゾルバーは未対応のため(2021/04/20時点)、今回はDNSサーバーを構築して代替します。

シンガポールリージョンには、DNSサーバーを用意して、各リージョンに作成したRoute 53 リゾルバー、DNSサーバーをフォワーダーに設定します。
これにより、一方のリージョンに対して接続できなくなった場合でも名前解決が可能となります。

【参考】
- VPC ピアリング接続
- VPC とネットワーク間の DNS クエリの解決

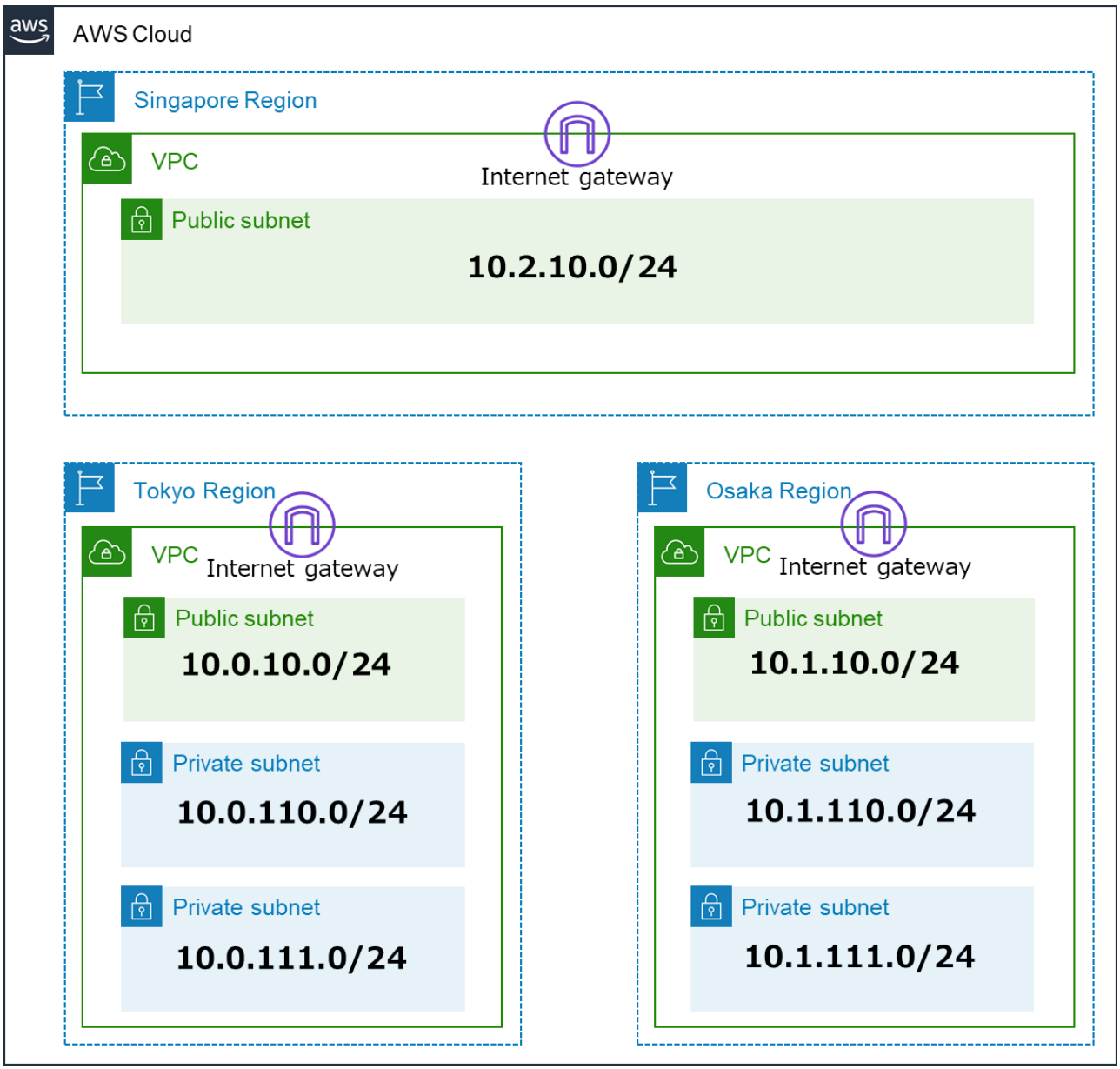
【参考】（以前のブログ）
- リージョン間 VPC ピアリング接続を利用したHAクラスターの構築を試してみました(Windows)
- Route 53 リゾルバーを使用してAWS上のHAクラスターへの接続を試してみました

2. HAクラスター構築手順

CLUSTERPRO Xを使用した「DNS名制御によるマルチリージョンHAクラスター」を構築します。
今回、VPN接続の設定手順は割愛します。

2.1 VPC、サブネットの作成

まず、VPCやサブネットなどを作成します。CIDRやサブネットアドレスなどは以下の通りです。



拡大表示

併せてインターネットゲートウェイも作成しておきます。
ルートテーブルやセキュリティグループについては後述します。

東京リージョン (ap-northeast-1)

```
VPC(VPC ID：vpc-1111aaaa)
    ーCIDR：10.0.0.0/16
    ーSubnets
      ■Subnet-A1 (Public) ：10.0.10.0/24
      ■Subnet-A2 (Private)：10.0.110.0/24
      ■Subnet-A3 (Private)：10.0.111.0/24
    ーInternet Gateway
```

大阪リージョン (ap-northeast-3)

VPC(VPC ID：vpc-2222bbbb)
　－CIDR：10.1.0.0/16
　－Subnets
　　■Subnet-B1 (Public) ：10.1.10.0/24
　　■Subnet-B2 (Private)：10.1.110.0/24
　　■Subnet-B3 (Private)：10.1.111.0/24
　－Internet Gateway

シンガポールリージョン (ap-southeast-1)

VPC(VPC ID：vpc-3333cccc)
　－CIDR：10.2.0.0/16
　－Subnets
　　■Subnet-C1 (Public) ：10.2.10.0/24
　－Internet Gateway

2.2 VPC ピアリングの作成

東京リージョンと大阪リージョンのVPC間を接続するためVPC ピアリング接続の作成を行います。

大阪リージョンでピアリング接続を作成し、リクエスタに大阪リージョンのVPC、アクセプタに東京リージョンのVPCを選択します。

※ 東京リージョンでピアリング接続を作成した場合に、アクセプタを作成するリージョンに大阪リージョンが選択できません。(2021/04/20時点)

VPC（リクエスタ）：vpc-2222bbbb ←大阪リージョンのVPC ID
VPC（アクセプタ）：vpc-1111aaaa ←東京リージョンのVPC ID

拡大表示

2.3 ルートテーブルの作成

各VPCにおいてルートテーブルを必要に応じて作成します。
シンガポールリージョンへの接続(VPN接続)に使用するルーティング設定、および、シンガポールリージョンのルートテーブルの記載は割愛します。

■ **東京リージョン (ap-northeast-1)**

Subnet-A1 (Public) 用ルートテーブル

送信先	ターゲット	備考
0.0.0.0/0	Internet Gateway	Internet通信用
10.0.0.0/16	local	VPC内通信用
10.1.0.0/16	pcx-1234abcd	VPC(大阪リージ

	(VPC ピアリング接続のID)	ョン)との通信用

Subnet-A2 (Private) 用ルートテーブル

送信先	ターゲット	備考
0.0.0.0/0	NATインスタンスのENI ID	Internet通信用(NATインスタンス作成後に設定)
10.0.0.0/16	local	VPC内通信用
10.1.0.0/16	pcx-1234abcd (VPC ピアリング接続のID)	VPC(大阪リージョン)との通信用

Subnet-A3 (Private) 用ルートテーブル

送信先	ターゲット	備考
10.0.0.0/16	local	VPC内通信用
10.1.0.0/16	pcx-1234abcd (VPC ピアリング接続のID)	VPC(大阪リージョン)との通信用

■ **大阪リージョン (ap-northeast-3)**

Subnet-B1 (Public) 用ルートテーブル

送信先	ターゲット	備考
0.0.0.0/0	Internet Gateway	Internet通信用
10.1.0.0/16	local	VPC内通信用
10.0.0.0/16	pcx-1234abcd (VPC ピアリング接続のID)	VPC(東京リージョン)との通信用

Subnet-B2 (Private) 用ルートテーブル

送信先	ターゲット	備考
0.0.0.0/0	NATインスタンスのENI ID	Internet通信用 (NATインスタンス作成後に設定)
10.1.0.0/16	local	VPC内通信用
10.0.0.0/16	pcx-1234abcd (VPC ピアリング接続のID)	VPC(東京リージョン)との通信用

Subnet-B3 (Private) 用ルートテーブル

送信先	ターゲット	備考
10.1.0.0/16	local	VPC内通信用
10.0.0.0/16	pcx-1234abcd	VPC(東京リージ

	(VPC ピアリング接続のID)	ョン)との通信用

2.4 セキュリティグループの作成

各VPCにおいてセキュリティグループを必要に応じて作成します。
セキュリティグループは、システムのポリシーに応じて適切に設定してください。

今回はさらにシンガポールリージョンのDNSサーバーからDNSクエリを受信できるように、53番ポートの通信を許可するセキュリティグループを以下の通り作成します。

東京リージョン (ap-northeast-1)

Security Group
■InboundEndpoint (Group ID：sg-0000aaaa)
インバウンドのルール01：
・タイプ　　：DNS(UDP)
・プロトコル：UDP
・ポート範囲：53
・ソース　　：10.2.10.0/24 ← **シンガポールリージョンのDNSサーバーが存在するサブネット**
インバウンドのルール02：
・タイプ　　：DNS(TCP)
・プロトコル：TCP
・ポート範囲：53
・ソース　　：10.2.10.0/24 ← **シンガポールリージョンのDNSサーバーが存在するサブネット**

大阪リージョン (ap-northeast-3)

Security Group
■InboundEndpoint (Group ID：sg-0000bbbb)
インバウンドのルール01：
・タイプ　　：DNS(UDP)
・プロトコル：UDP
・ポート範囲：53
・ソース　　：10.2.10.0/24 ← **シンガポールリージョンのDNSサーバーが存在するサブネット**
インバウンドのルール02：
・タイプ　　：DNS(TCP)

・プロトコル：TCP
・ポート範囲：53
・ソース　　：10.2.10.0/24 ← **シンガポールリージョンのDNSサーバーが存在するサブネット**

2.5 Amazon Route 53の設定

DNS名制御によるHAクラスターを実現するため、Amazon Route 53を設定します。
プライベートホストゾーンを作成し、ホストゾーンに関連付けるVPCに東京リージョンのVPCと大阪リージョンのVPCをそれぞれ指定します。

Domain Name：cluster.local
Hosted Zone ID：123456789abcde
Domain Name：Type：Private Hosted Zone for Amazon VPC
Domain Name：Associated VPCs：vpc-1111aaaa、vpc-2222bbbb

2.6 Route 53 リゾルバーの作成

インバウンドエンドポイントの作成手順は、下記サイトを参考にしました。

【参考】
VPC へのインバウンド DNS クエリの転送

東京リージョンに、Route 53 リゾルバーのインバウンドエンドポイントを作成します。
セキュリティグループには、シンガポールリージョンからのDNSクエリの受信用に作成したセキュリティグループを指定します。

インバウンドエンドポイント

エンドポイント名　　：InboundTest
当該リージョンのVPC　：vpc-1111aaaa
セキュリティグループ　：sg-0000aaaa
IP アドレス #1：
　・アベイラビリティーゾーン：ap-northeast-1a
　・サブネット：subnet-A3
　・IPアドレス：10.0.111.100
IP アドレス #2：
　・アベイラビリティーゾーン：ap-northeast-1a

・サブネット：subnet-A3
・IPアドレス：10.0.111.101

2.7 EC2インスタンスの作成

東京リージョンのSubnet-A2 (Private)、大阪リージョンのSubnet-B2 (Private)のそれぞれに、HAクラスター用インスタンスを作成します。
また、東京リージョンのSubnet-A1 (Public)、大阪リージョンのSubnet-B1 (Public)にNATインスタンスを作成します。

大阪リージョンのSubnet-B3にDNSサーバー用インスタンスを構築し、DNSフォワーダーとして利用します。
DNSサーバー用インスタンスには、シンガポールリージョンからのDNSクエリの受信用に作成したセキュリティグループを設定します。

シンガポールリージョンのSubnet-C1にクライアント用インスタンスとDNSサーバー用インスタンスを作成します。
今回は、BINDでDNSサーバーを構築し、フォワーダーに「東京リージョンのRoute 53 リゾルバー」と「大阪リージョンのDNSサーバー」を指定します。
これにより、東京リージョンと大阪リージョンのどちらかのリージョンへの接続が途絶えた場合でも、Route 53による名前解決が可能になります。

2.8 CLUSTERPROによるHAクラスターの構築

「DNS名制御によるHAクラスター」を構築します。CLUSTERPROの構成は以下の通りです。
CLUSTERPROのフェールオーバーグループには「AWS DNSリソース」、「ミラーディスクリソース」の2つを登録します。

CLUSTERPRO

ーフェールオーバーグループ（failover）
■AWS DNSリソース
・ホストゾーン：123456789abcde ←**「2.5 Amazon Route 53の設定」で設定したホストゾーンID**

・リソースレコードセット名：sample.cluster.local.

■ミラーディスクリソース(Windows版)

・データパーティション：M:¥

・クラスタパーティション：R:¥

■ミラーディスクリソース(Linux版)

・データパーティション：/dev/nvme1n1p2

・クラスタパーティション：/dev/nvme1n1p1

【参考】

CLUSTERPRO X ソフトウェア構築ガイド

Windows > クラウド > Amazon Web Services > HAクラスタ構築ガイド

Linux > クラウド > Amazon Web Services > HAクラスタ構築ガイド

3. 動作確認

クライアントマシンから、仮想ホスト名(sample.cluster.local)を使用して、東京リージョンのEC2インスタンス（Server-A）、大阪リージョンのEC2インスタンス（Server-B）にアクセスできることを確認します。

稼働系のEC2インスタンスを停止して、正常にフェールオーバーができることを確認します。
また、併せてVPN接続を停止することで、一方のリージョンへ接続できない場合でもRoute53による名前解決ができることを確認します。

1. Server-Aでフェールオーバーグループを起動します。
2. シンガポールリージョンのクライアントマシンから、仮想ホスト名(sample.cluster.local)にアクセスし、Server-Aに接続できることを確認します。
3. Cluster WebUIから、Server-Aをシャットダウンして、フェールオーバーグループをServer-Bにフェールオーバーします。
   また、東京リージョンとシンガポールリージョンのVPN接続を停止します。
4. シンガポールリージョンのクライアントマシンから、仮想ホスト名(sample.cluster.local)にアクセスし、Server-Bに接続できることを確認します。
5. AWS マネジメントコンソールからServer-Aを起動し、Server-AがHAクラスターに復帰するまで待ち合わせます。
   また、東京リージョンとシンガポールリージョンのVPN接続を再開します。
6. Cluster WebUIから、Server-Bをシャットダウンして、フェールオーバーグループをServer-Aにフェールオーバーします。
   また、大阪リージョンとシンガポールリージョンのVPN接続を停止します。

7. クライアントマシンから、仮想ホスト名(sample.cluster.local)にアクセスし、Server-Aに接続できることを確認します。

DNS名制御によるマルチリージョンHAクラスターについて、現用系、待機系の切り替えが行えることを確認できました。

まとめ

今回は東京リージョンと大阪リージョンを使用した、マルチリージョンHAクラスターを構築しました。
大阪リージョンの登場により、より気軽にマルチリージョンHAクラスターが構築できるようになったため、今後利用機会が増えていくのではないかと思います。
大阪リージョンを利用したマルチリージョンHAクラスターの一例として是非ご検討ください。

今後も東京リージョンと大阪リージョンを連携したHAクラスターをご紹介していきたいと思います。

本記事の構成をご検討の際は、CLUSTERPROの 試用版を用いて検証した後、ご提案・構築ください。