

# 【AWS就業研修】Webサーバ構築＋負荷分散

## このハンズオンのゴール

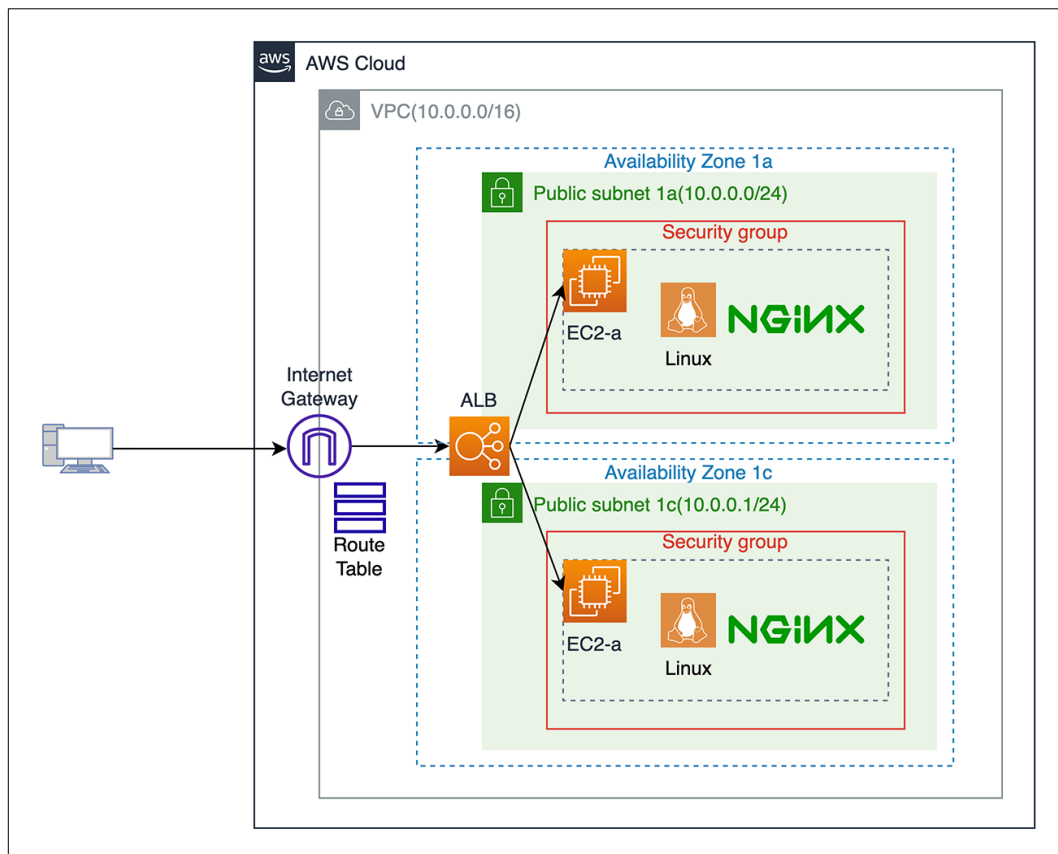
- ・ AWSの概要の理解
- ・ AWS EC2を利用したクラウド環境構築の理解
- ・ LinuxサーバによるWebサーバの構築
- ・ ロードバランサーによる負荷分散の仕組み構築

## 前提条件

- ・ ローカル端末のOS: Windows
- ・ 研修用AWSアカウント: 作成済み
- ・ 研修用IAM User: 作成済み
- ・ 推奨環境: デュアルモニター

## 構成イメージ図

今回のハンズオンでAWS上で構成するイメージとなります。



# 1.AWSコンソールへログイン

1.連携済みのURLからAWSに接続

2.IAMユーザーとしてサインインの画面でユーザ名とパスワードを入力して[サインイン]をクリック



3.リージョンが[東京▼]以外になっている場合は、検索窓の右側にある[リージョン名▼]をクリックし、[アジアパシフィック(東京) ap-northeast-1]を選択



リージョンとはそれぞれ、地理的に離れた場所に設けられた領域であり、これにより最大限の耐障害性と安定性が確保されます。

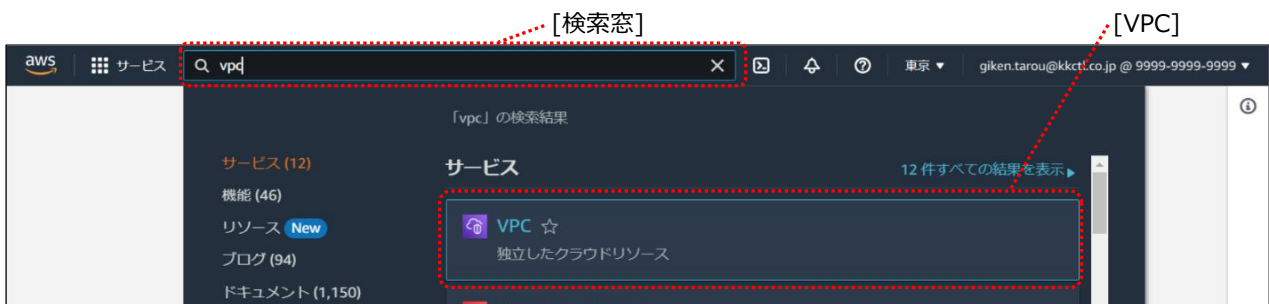
## 2.ネットワーク環境構築

### 2-1.VPC作成

ここでは1つのVPC(Amazon Virtual Private Cloud)を作成します。

Amazon VPCとは、Amazon Virtual Private Cloudの略です。Amazon VPCにより仮想ネットワークを構築し、IPアドレスおよびサブネットの管理・ルーティング・ゲートウェイの設定等一連の仮想ネットワークの制御を簡単に実現することができます。

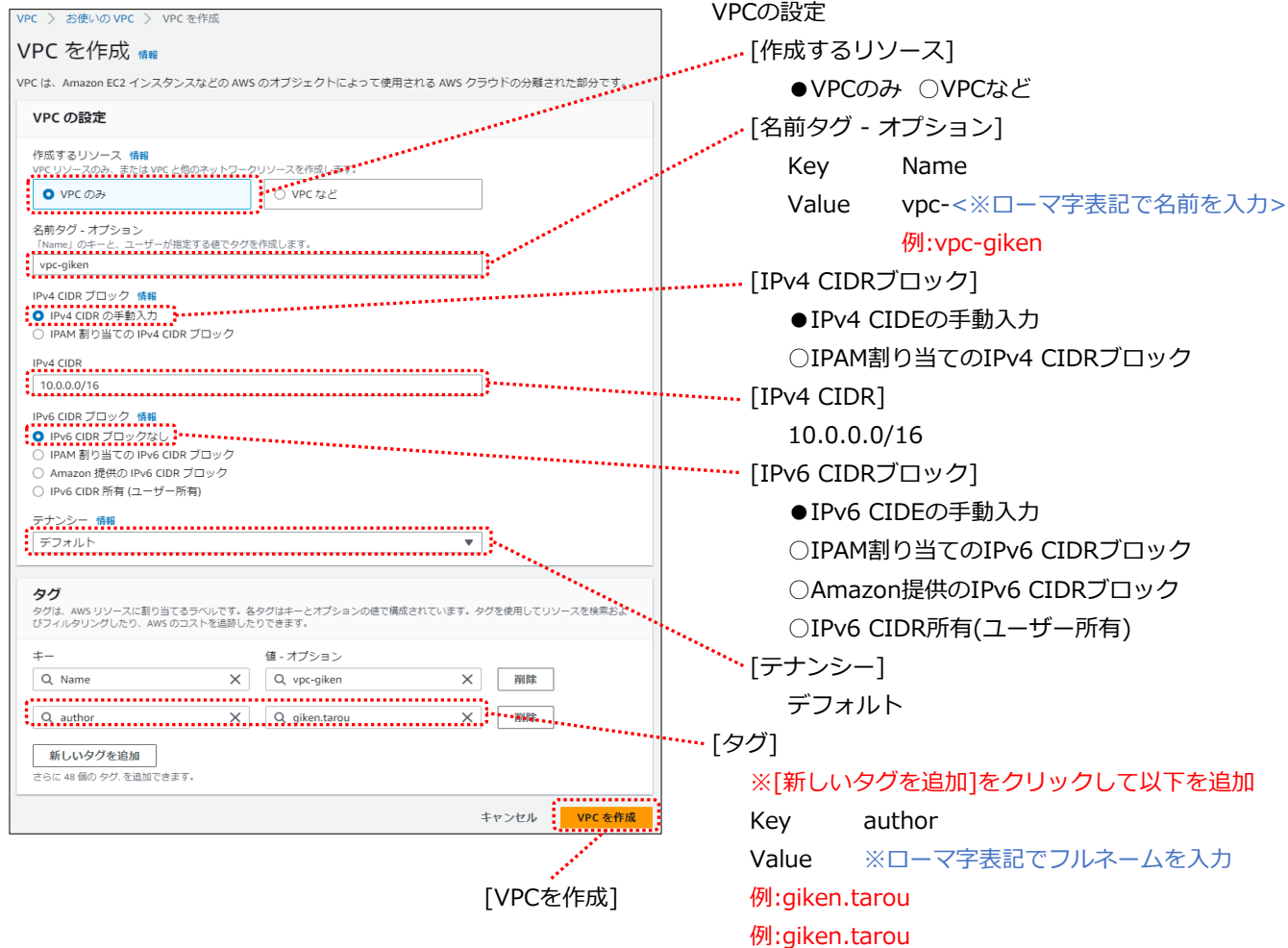
1.[AWSコンソール]画面上部の検索窓で[VPC]と入力し、検索結果の[VPC]をクリック



## 2.[AWS VPC]画面で[VPCを作成]をクリック



## 3.[VPCを作成]画面で以下を設定し、[VPCを作成]をクリック



## 2-2.VPCにサブネットを作成

ここではVPCに対して2つのサブネットを作成します。

サブネットとは、IPネットワークを細分化することを指します。記述方法はCIDR（サイダー）表記で、IPアドレスのプレフィックスとしてネットワークアドレスの後にスラッシュ（/）の後とネットワークアドレスビット数を記載します。使用例は、“10.0.0.0/16”等が挙げられます。

### 1.左ペインから[サブネット]をクリック

### 2.[サブネット]画面で[サブネットを作成]をクリック



### 3.[サブネットを作成]画面で、以下を設定し、[サブネットを作成]をクリック

The screenshot shows the 'Create Subnet' page in the AWS Management Console. The page is divided into two main sections: 'VPC' and 'Subnet Settings'. The 'VPC' section shows the selected VPC ID and its CIDR block. The 'Subnet Settings' section shows the subnet name, availability zone, IPv4 CIDR block, and tags. Annotations with red dotted lines point to specific fields and buttons, providing instructions in Japanese.

**[VPC]**  
vpc-<ローマ字表記の名前>  
※2-1で作成したVPCを選択

**サブネットの設定**

**[サブネット名]**  
public-subnet-1a-<※名前をローマ字で入力>  
例:public-subnet-1a-giken

**[アベイラビリティゾーン]**  
ap-northeast-1a

**[IPv4 CIDRブロック]**  
10.0.0.0/24

**[タグ - オプション]**  
※[新しいタグを追加]をクリックして以下を追加  
キー                      author  
値-オプション          ※ローマ字表記でフルネームを入力  
例:giken.tarou

**[新しいサブネットを追加]をクリック**

**[サブネット名]**  
public-subnet-1c-<※ローマ字表記で名前を入力>  
例:public-subnet-1c-giken

**[アベイラビリティゾーン]**  
ap-northeast-1c

**[IPv4 CIDRブロック]**  
10.0.1.0/24

**[タグ - オプション]**  
※[新しいタグを追加]をクリックして以下を追加  
キー                      author  
値-オプション          ※ローマ字表記でフルネームを入力  
例:giken.tarou

**[サブネットを作成]**

## 2-3.インターネットゲートウェイ作成、VPCに紐付け

ここではインターネットゲートウェイを作成し、VPCと紐付けることにより、インターネットを使う準備ができます。

サブネットをパブリックサブネットとすることで、インターネットアクセスが可能です。

送信先をIPv4の場合は、0.0.0.0/0とします。

ターゲットをインターネットゲートウェイIDである、igw-XXXXXXXXXXとします。

#### 1.左ペインから[インターネットゲートウェイ]をクリック

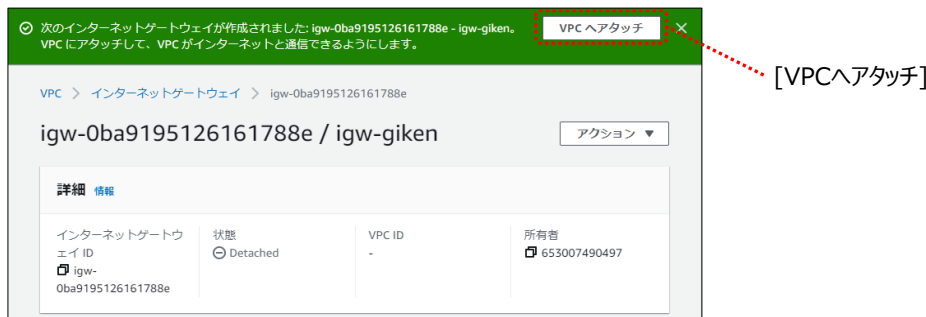
## 2.[インターネットゲートウェイ]画面で[インターネットゲートウェイの作成]をクリック



## 3.[インターネットゲートウェイの作成]画面で、以下を設定し、[インターネットゲートウェイの作成]をクリック



## 4.画面右上の[VPCへアタッチ]をクリック



## 5.[VPCにアタッチ]画面の[使用可能なVPC]で、作成したVPCを選択し、[インターネットゲートウェイのアタッチ]をクリック



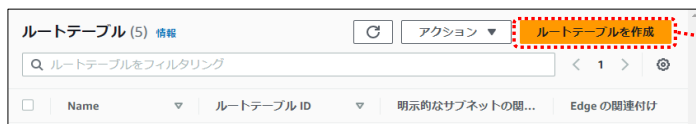
## 2-4.ルートテーブル作成

### 2-4-1.ルートテーブルの設定

ここではルートテーブルを設定することによりインターネットと通信をできるようにします。  
VPCルートテーブルでは、ルートの送信先ならびにターゲットを関連付けします。

1.左ペインから[ルートテーブル]をクリック

2.[ルートテーブル]画面で[ルートテーブルを作成]をクリック



[ルートテーブルを作成]

3.[ルートテーブル]画面で、以下を設定し、[ルートテーブルを作成]をクリック



[名前 - オプション]

rtb-<※ローマ字表記で名前を入力>

例: rtb-giken

[VPC]

vpc-xxxxxxxxxxxxxxxx(vpc-<※ローマ字表記の名前>)

※2-1で作成したVPCを選択

例: vpc-xxxxxxxx~(vpc-giken)

[タグ]

※[新しいタグを追加]をクリックして以下を追加

キー author

値 - オプション ※ローマ字表記でフルネームを入力

例:giken.tarou

[ルートテーブルを作成]

## 2-4-2.ルートテーブルを編集

ここではルートテーブルを設定することによりインターネットと通信をできるようにします。

VPCルートテーブルでは、ルートの送信先ならびにターゲットを関連付けします。

1.作成したルートテーブル[rtb-xxxxxxxxxxxxxxxx]の画面下の[ルート]タブをクリックして、[ルートを編集]をクリック



[ルートタブ]

[ルートを編集]

2.[ルートを編集]画面で[ルートを追加]をクリック

3.追加表示された[ルートを編集]で、以下を設定し、[変更を保存]をクリック

ルートを編集

送信先: 0.0.0.0/0

ターゲット: igw-0863fbf310535866

ステータス: アクティブ

変更を保存

[送信先]

0.0.0.0/0

[ターゲット]

igw-xxxxxxxx~(igw-**<※ローマ字表記の名前>**)

※2-3で作成したインターネットゲートウェイを選択

ターゲットを選択は2段階になっているので、  
[インターネットゲートウェイ]を選択します。

次に[igw-]と入力すると、作成したインターネット  
ゲートウェイが表示されます。

[変更を保存]

## 2-4-3.サブネットの関連付けを編集

ここではルートテーブルを設定することによりインターネットと通信をできるようにします。

VPCルートテーブルでは、ルートの送信先ならびにターゲットを関連付けします。

1.[サブネット]タブをクリックして、[サブネットの関連付けを編集]をクリック

サブネットの関連付け

明示的なサブネットの関連付け (0)

サブネットの関連付けを編集

[サブネットの関連付けタブ]

[ルートを編集]

2.[サブネットの関連付けを編集]画面で、2-3で作成した2つのサブネットにチェックを入れ、[関連付けを保存]をクリック

サブネットの関連付けを編集

利用可能なサブネット (2/2)

名前	サブネット ID	IPv4 CIDR	IPv6 CIDR	ルートテーブル ID
public-subnet-1a-giken	subnet-0239debc2f58...	10.0.0.0/24	-	メイン@rtb-0f83cc60d071c
public-subnet-1c-giken	subnet-027f6b69fe9d0...	10.0.1.0/24	-	メイン@rtb-0f83cc60d071c

選択されたサブネット

subnet-0239debc2f58ac7a / public-subnet-1a-giken

subnet-027f6b69fe9d0034 / public-subnet-1c-giken

関連付けを保存

[サブネットのチェック]

[関連付けを保存]

## 3.Amazon EC2(仮想サーバ)作成

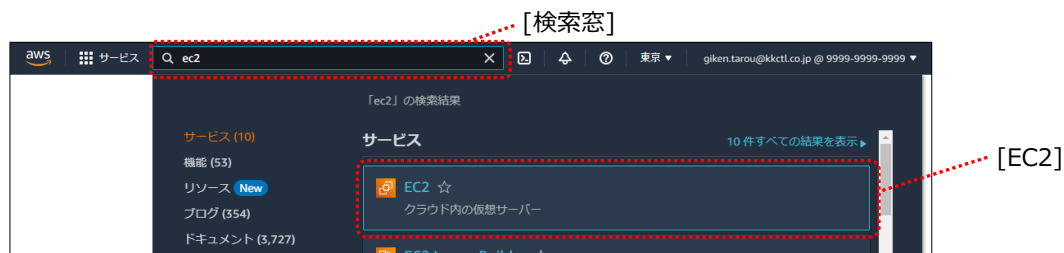
ここでは2つのEC2(Amazon Elastic Compute Cloud)を作成します。

1台目のEC2の手順項番は3a～、2台目のEC2の手順項番は3b～とします。

EC2とはAmazon Cloud上に作成可能なコンピュータであり、OSやスペック、台数などを容易に準備することができます。  
必要な時に起動したり、停止することが可能で停止中は料金が発生しないというメリットがあります。

## 3a-1.AWS EC2(インスタンス)起動

- 1.[AWSコンソール]画面上部の検索窓で[EC2]と入力し、検索結果の[EC2]をクリック

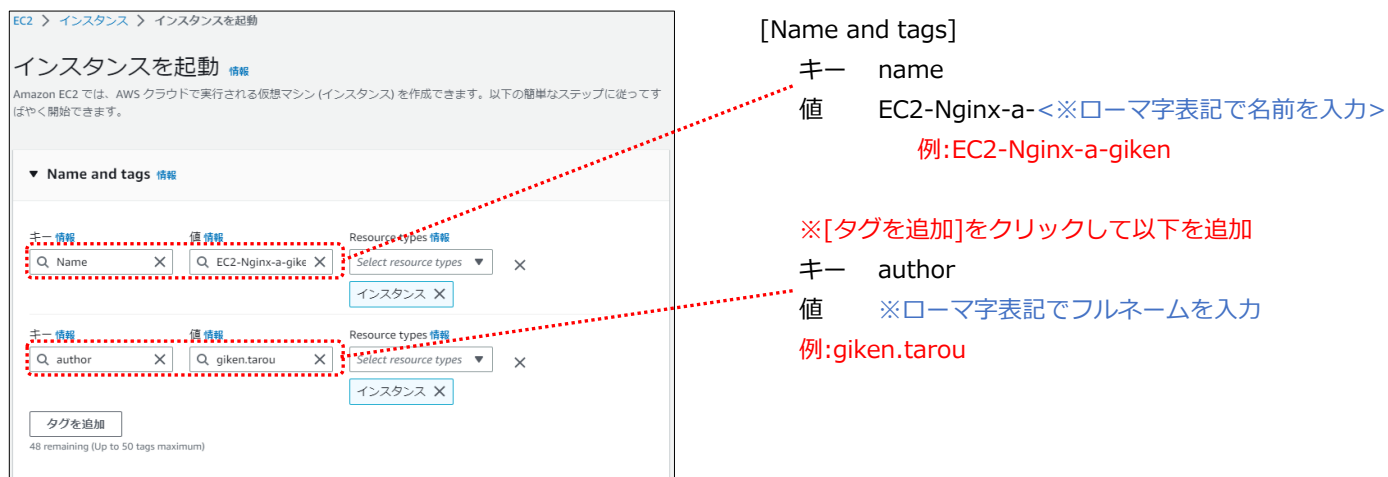


- 2.[AWS EC2]画面の[インスタンスを起動]枠の[インスタンスを起動]をクリック



## 3a-2.名前とタグの追加

- 1.[VPCを作成]画面で以下を設定





## 3a-3.アプリケーションおよびOSイメージ(AMI)の指定

1.[クリックスタート]タブをクリックし、[Amazon Linux]をクリック



## 3a-4.インスタンスタイプの指定

1.[インスタンスタイプ]の選択欄で[t2.micro]を選択



## 3a-5.キーペア(ログイン用の鍵)作成

キーペアとは、EC2インスタンスにアクセスするために必要となる「鍵」のことで、「公開鍵」と「秘密鍵」になります。キーペアを作成し、ローカルの端末の[c:¥work]などのフォルダを作成し、[公開鍵]をダウンロードします。

### 1.キーペア（ログイン）の「新しいキーペアの作成」をクリック

▼ キーペア (ログイン) 情報

キーペアを使用してインスタンスに安全に接続できます。インスタンスを起動する前に、選択したキーペアにアクセスできることを確認してください。

キーペア名 - 必須

選択

新しいキーペアの作成

[新しいキーペアの作成]

### 2.[キーペアを作成]画面で以下を設定し、[キーペアを作成]をクリック

キーペアを作成

キーペアを使用すると、インスタンスに安全に接続できます。

以下にキーペアの名前を入力してください。プロンプトが表示されたら、ご使用のコンピュータの安全でアクセス可能な場所に、そのキーペアを保存してください。後でインスタンスに接続するときに必要になります。 [詳細はこちら](#)

キーペア名

EC2-kpr-giken

キーペアのタイプ

☒ RSA  
RSA で暗号化されたプライベートとパブリックのキーペア

☐ ED25519  
ED25519 で暗号化されたプライベートキーとパブリックのキーペア (Windows インスタンスではサポートされません)

プライベートキーファイル形式

☒ .pem  
OpenSSH で使用する場合

☐ .ppk  
PuTTY で使用する場合

キャンセル キーペアを作成

[キーペア名]

EC2-kpr-<※ローマ字表記で名前を入力>

例:EC2-kpr-giken

[キーペアのタイプ]

● RSA

○ ED25519

[プライベートキーファイル形式]

● .pem

○ .ppk

[キーペアを作成]

### 3.ダウンロード用のダイアログが表示されたら、

[c:¥work]フォルダに保存する。

※事前にCドライブの直下に[work]フォルダを作成する。

### 4.ダウンロード後に元の画面に遷移するので、キーペア名の欄に作成したキーペアが設定されていることを確認

▼ キーペア (ログイン) 情報

キーペアを使用してインスタンスに安全に接続できます。インスタンスを起動する前に、選択したキーペアにアクセスできることを確認してください。

キーペア名 - 必須

EC2-kpr-giken

新しいキーペアの作成

ここでは作成するEC2インスタンスが利用するネットワークの設定を行います。

[VPC]の選択欄で2-1で作成したVPCを選択  
例:vpc-xxxxxxxxxx~(vpc-giken)

[サブネット]  
2-2で作成した[ap-northeast-1a]のサブネットを選択  
例:subnet-xxxxxxxx~(public-subnet-1a-giken)

[パブリックIPの自動割り当て]  
有効化

[ファイアウォール(セキュリティグループ)]  
●セキュリティグループを作成する  
○既存のセキュリティグループを選択する

[セキュリティグループ名]  
for-EC2-<※ローマ字表記で名前を入力>  
例:for-EC2-giken

[説明]  
for-EC2-<※ローマ字表記で名前を入力>  
例:for-EC2-giken

[インバウンドセキュリティグループのルール]  
※ここでは3つのルール(ssh/http/https)を作成

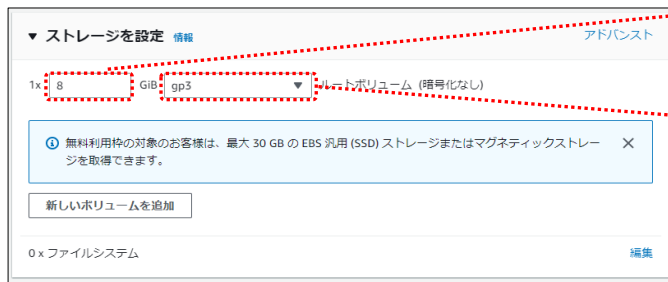
[セキュリティグループルール1]  
タイプ ssh  
ソースタイプ 任意の場所  
説明 from home

[セキュリティグループルール2]  
タイプ HTTP  
ソースタイプ 任意の場所  
説明 from Internet

[セキュリティグループルール3]  
タイプ ssh  
ソースタイプ 任意の場所  
説明 from Internet

## 3a-6-5.ストレージを設定

- 1.[ストレージを設定]枠で以下を設定



[容量(GiB)]

8 GiB

[ストレージタイプ]

gp3

## 3a-6-4.インスタンスを起動

- 1.[概要]枠のインスタンス数が[1]であることを確認し、[インスタンスを起動]をクリック



[インスタンス数]

[インスタンスを起動]

## 3b-1.AWS EC2(インスタンス)起動

- 1.[AWSコンソール]画面上部の検索窓で[EC2]と入力し、検索結果の[EC2]をクリック
- 2.[AWS EC2]画面の[インスタンスを起動]枠の[インスタンスを起動]をクリック

## 3b-2.名前とタグの追加

- 1.[VPCを作成]画面で以下を設定

### 名前とタグ

キー      name  
値      EC2-Nginx-b-**<※ローマ字表記で名前を入力>**  
例:EC2-Nginx-b-giken

**※[タグを追加]をクリックして以下を追加**

キー      author  
値      **※ローマ字表記でフルネームを入力**  
例:giken.tarou

## 3b-3.アプリケーションおよびOSイメージ(AMI)の指定

- 1.[クリックスタート]タブをクリックし、[Amazon Linux]をクリック

## 3b-4.インスタンスタイプの指定

- 1.[インスタンスタイプ]の選択欄で[t2.micro]を選択

## 3b-5.キーペア(ログイン用の鍵)の選択

1台目で作成したキーペアを利用します。

- 1.キーペア（ログイン）の「新しいキーペア名」欄で3a-5で作成したキーペアを選択

例:EC2-kpr-giken

## 3b-6.ネットワーク設定

ここでは作成するEC2インスタンスが利用するネットワークの設定を行います。

### 3b-6-1.VPC、サブネットを設定

- 1.[ネットワーク設定]枠の右側にある[編集]をクリック
- 2.[VPC]の選択欄で2-1で作成したVPCを選択  
例:vpc-xxxxxxxxxxxxxxxx(vpc-giken)
- 3.[サブネット]の選択欄で2-2で作成したアベイラビリティゾーン[ap-northeast-1a]のサブネットを選択  
例:subnet-xxxxxxxxxxxxxxxx(public-subnet-1a-giken)

### 3b-6-2.パブリックIP無効設定

- 1.[パブリックIPの自動割り当て]で[有効化]を選択

### 3b-6-3.セキュリティグループを選定

1台目で作成したセキュリティグループを利用します。

- 1.[ファイアウォール(セキュリティグループ)]枠で以下の設定をする

- セキュリティグループを作成する
- 既存のセキュリティグループを選択する

共通のセキュリティグループ

for-EC2-<※ローマ字表記の名前>

例:for-EC2-giken

## 3b-6-5.ストレージを設定

- 1.[ストレージを設定]枠で以下を設定  
1x 8 GiB  
gp2 ルートボリューム

## 3b-6-4.インスタンスを起動

- 1.[概要]枠のインスタンス数が[1]であることを確認
- 2.[インスタンスを起動]をクリック

## 3-7.作成されたEC2の確認

何か打ち込む予定

## 4.Webサーバ構築

ネットワークとサーバの準備が整いましたので、ローカル端末からAmazon EC2に接続してWebサーバを構築していきます。

### 4-1.Amazon EC2にSSH接続

ローカル端末からAmazon EC2に接続するための設定をしていきます。

#### 4-1-1.キーペアのアクセス権の変更

EC2作成時にダウンロードしたキーペアのアクセス権を変更していきます。

- 1.Powershellを起動し、以下のコマンドを実行

```
cd c:\work
dir
$path = ".\EC2-kpr-<※ローマ字表記の名前>.pem
icacls.exe $path /reset
icacls.exe $path /GRANT:R "$($env:USERNAME):(R)"
icacls.exe $path /inheritance:r
```

#### 4-1-2.Amazon EC2にSSHで接続

SSHクライアントでEC2サーバに接続します。

- 1.コマンドプロンプトを起動し、以下のコマンドを実行

```
ssh -i "EC2-kpr-<※ローマ字表記の名前>.pem" ec2-user@パブリックIPアドレス
```

例 : `ssh -i "EC2-kpr-giken.pem" ec2-user@xxx.xxx.xxx.xxx`

※EC2のグローバルIPアドレスはAWSコンソールのEC2のインスタンスの一覧を表示後、確認したいインスタンスIDをクリックすることにより、パブリックIPv4アドレスを確認することができます。

2.以下のメッセージが表示されたら、[yes]と入力し[Enter]キーを押下

SSHクライアントがインストールされていない場合は、sshコマンドがエラーになりますので、[Windows]の[設定]の[オプション機能]から以下の2つのオプションを追加する必要があります。

- ・OpenSSHクライアント
- ・OpenSSHサーバー

## 4-2.Webサーバ(Nginx)をインストール

---

1.以下のコマンドを実行し、インストール可能なパッケージを検索

```
which amazon-linux-extras
amazon-linux-extras
```

2.以下のコマンドを実行し、nginxのパッケージをインストール

```
sudo amazon-linux-extras install nginx1
```

3.以下のコマンドを実行し、インストールされたNginxのバージョンを確認

```
nginx -v
```

4.以下のコマンドを実行し、Nginxの初期設定ファイルをバックアップ

```
sudo cp -a /etc/nginx/nginx.conf /etc/nginx/nginx.conf.back
```

## 4-3.Nginxを起動

---

1.以下のコマンドを実行し、Nginxを起動

```
sudo systemctl start nginx
```

2.以下のコマンドを実行し、EC2インスタンスが起動したタイミングでNginxも起動する設定をする

```
sudo systemctl enable nginx
```

3.以下のコマンドを実行し、正しく設定されていることを確認

```
systemctl status nginx
```

## 4-4.ブラウザでWebサーバに接続

---

1.ローカル端末のブラウザにEC2のパブリックIPを入力

## 4-5.表示されるWebページを変更

---

Nginxのコンテンツファイルを変更して、表示されるページ内容を変更します。

## 5.ALB(アプリケーションロードバランサー)作成

---

### 5-1.ターゲットグループの設定

---

### 5-2.ターゲットの登録

---

### 5-3.ロードバランサーとリスナーの設定

---

### 5-4.ロードバランサーのテスト

---