

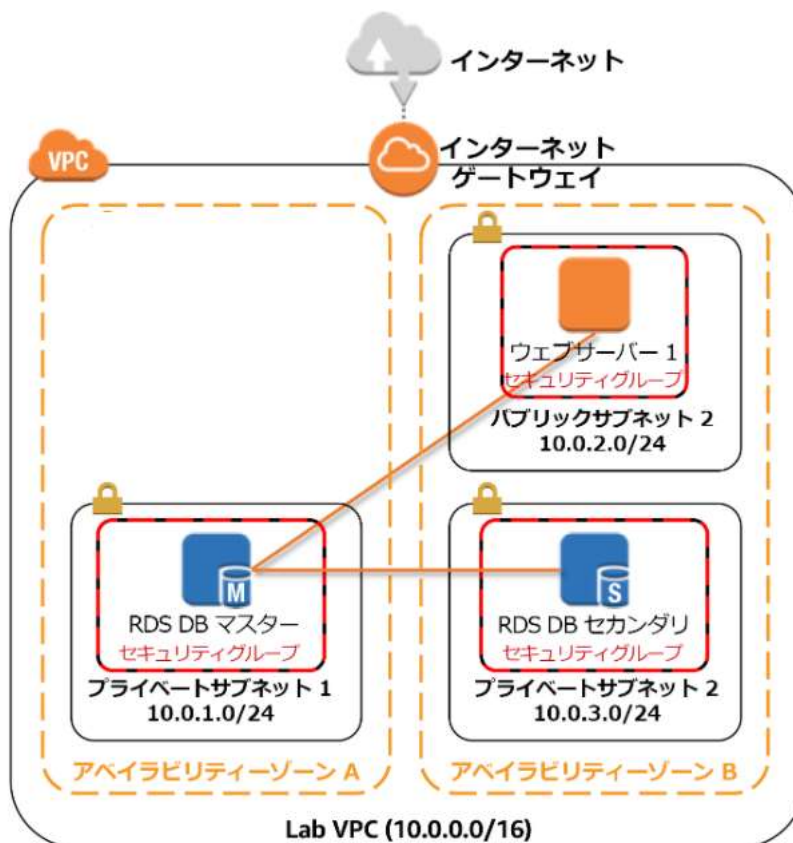
【AWS マネジメントコンソールへログイン】

[AWS Educate へログイン] ※実習後に自動で削除希望の場合は、Academy の Vocareum のサンドボックスを使用すること。以下は、ログアウト後も残る環境で作成する場合は。

1. <https://www.awseducate.com>
2. Educate へサインイン
3. AWS Account をクリック
4. AWS Educate Starter Account をクリック
5. 画面右上の My Class (私のクラス) をクリックして、3 Department ~ 又は Starter Account をクリック
6. AWS Console をクリック

【Web サーバ + DB サーバを構築】

下記のような構成のサービスを、構築します。



[STEP1] ネットワーク環境の構築

①VPC の作成

1. サービスから VPC を選択
2. ナビゲーションペインから、VPC を選択
3. **VPC の作成** ボタンをクリック

名前タグ：WebDB_VPC （何でも良いが、わかり易い名前）

IPv4 CIDR ブロック*：10.0.0.0 /16 （全体のネットワークアドレス）

※その他の項目は何もしない

②作成した VPC の中に、サブネットを 3 つ作成する。※今回 Public Subnet1 は作成しない

1. 10.0.1.0/24 のサブネットを作成

- 1-1. ナビゲーションペインから、サブネットを選択

- 1-2 **サブネットの作成** ボタンをクリック

名前タグ：Private Subnet1

VPC*：WebDB_VPC （①で作成した VPC）

アベイラビリティーゾーン： us-east-1a

IPv4 CIDR ブロック*：10.0.1.0/24

2. 10.0.2.0/24 のサブネットを作成

- 2-1. ナビゲーションペインから、サブネットを選択

- 2-2 **サブネットの作成** ボタンをクリック

名前タグ：Public Subnet2

VPC*：WebDB_VPC （①で作成した VPC）

アベイラビリティーゾーン： us-east-1b

IPv4 CIDR ブロック*：10.0.2.0/24

3. 10.0.3.0/24 のサブネットを作成

- 3-1. ナビゲーションペインから、サブネットを選択

- 3-2 **サブネットの作成** ボタンをクリック

名前タグ：Private Subnet2

VPC*：WebDB_VPC （①で作成した VPC）

アベイラビリティーゾーン： us-east-1b

IPv4 CIDR ブロック*：10.0.3.0/24

★[補足説明]★この段階では名前タグに Private や Public とついていますが、ただのサブネットです。ルートテーブルに関連付けて、初めて Public（外に出ていける NW）や Private（内部(10.0.0.0/16内)通信のみの NW）といった種類のサブネットになります。★

③ルートテーブルの作成

1. Private Subnet 用ルートテーブルの設定

1-1 ナビゲーションペインから、ルートテーブルを選択

★[補足説明]★ここで表示されるルートテーブルのうち、①で作成した WebDB_VPC に関するルートテーブルは、メインが「はい」のルートテーブルが一つだけ作成されています。メインが「はい」のルートテーブルは、メインルートテーブルと呼ばれ、サブネットワークの関連付けの設定をしていなくても、デフォルトで作成した全てのサブネットワークが、このメインルートテーブルへ関連付けられています。

セキュリティの観点から、メインルートテーブルはプライベートのトラフィック制御用とし、パブリック用のルートテーブルを新規に作成し、そこにパブリックサブネットを割り当てた方が、設定を忘れたときに、デフォルトでは外部アクセスできないようになるため安全です★

1-2 メインが「はい」で、VPC ID に WebDB_VPC が表示されている行を選択します。

1-3 Name の列が空白のため、マウスカーソルをポイントすると、鉛筆のアイコンが表示されるため、それをクリックして Private WebDB_VPC と入力して ☒ をクリック。

1-4 ペイン（画面下半分の表示）のルートをクリックし、10.0.0.0/16 のルートだけが表示されることを確認。★[補足説明]NAT ゲートウェイを使用しないので、1 行のみ★
さらに、サブネットの関連付けをクリックします。

★[補足説明]「サブネットの関連付けがありません。」と表示されていますが、このルートテーブルはメインルートテーブルなので、実際には現段階では作成した 3 つのサブネットが、このルートテーブルを使用する設定になっています。しかしこの状態では、使用物に解り難いので、明示的に関連付けをしてあげましょう ★

サブネットの関連付けの編集をクリックします。Private Subnet 1 と Private Subnet2 の 2 つのサブネットを選択して、保存ボタンをクリック。

2. Public Subnet 用ルートテーブルの設定

2-1 ナビゲーションペインから、インターネットゲートウェイを選択

2-2 右上にある、インターネットゲートウェイの作成 ボタンをクリック

2-3 名前タグ：IGW-WebDB と入力し、インターネットゲートウェイ作成をクリック

2-4 作成後の画面右上に、「VPC へアタッチ」というボタンが表示されるので、それをクリック。表示されない人は、その画面のアクションボタンから「VPC へアタッチ」をクリック。その後、WebDB_VPC を選択して、アタッチする。

★[補足説明] 2-1 から 2-4 まだが、クラウド内部の NW からインターネットへ出るための

出口である、インターネットゲートウェイの準備です。★

2-5 ナビゲーションペインから、ルートテーブルを選択

2-6 ルートテーブルの作成ボタンをクリックし、下記を入力して作成ボタンをクリック。

名前タグ：Public WebDB_VPC

VPC*：WebDB_VPC

2-7 表示されているルートテーブルから、Name が Public WebDB_VPC（今作成したルートテーブル）を選択する。

2-8 ペイン（画面下半分の表示）のルートをクリックし、10.0.0.0/16 のルートだけが表示されることを確認。表示画面から、ルートの編集ボタンをクリック。

ルートの編集画面から、ルートの追加ボタンをクリックし、下記の行を追加する。

- ・送信先： 0.0.0.0/0 （インターネット全般の事）

- ・ターゲット：リストから Internet Gateway を選択し、IGW-WebDB を選択。

右下にある、ルートの保存ボタンをクリック

2-8 ペイン（画面下半分の表示）に戻ったら、右隣の「サブネットの関連付け」タブをクリック。サブネットの関連付けの編集ボタンをクリックし、Public Subnet2 だけを選択して保存ボタンをクリック。

ここまでの、ネットワークの設定です。

[STEP2] Web サーバの構築

① Web サーバのセキュリティグループを設定

1. ナビゲーションペインで [セキュリティグループ] をクリックします。

2. セキュリティグループの作成 をクリックし、以下のように設定します。

- セキュリティグループ名: Web Security Group

- 説明: Enable HTTP access

- VPC: WebDB_VPC

3. インバウンドのルールで、ルールを追加 ボタンをクリック。

- タイプ: HTTP

- ソース: 任意の場所

- 説明: Permit web requests

4. さらに、ルールを追加 ボタンをクリック。

- タイプ: SSH
 - ソース: マイ IP
 - 説明: **Permit SSH Access**
5. アウトバウンドのルールで、ルールを追加 ボタンをクリック。
- タイプ: すべてのトラフィック
 - ソース: 任意の場所
 - 説明: **Web to Internet requests**
6. セキュリティグループを作成 ボタンをクリックこのセキュリティグループは、Amazon EC2 インスタンスを作成するときに使用します。

★[補足説明] インバウンドのルールは、Web サーバ(としての EC2 インスタンス)が受信するトラフィックの許可です。セキュリティグループは、ステートフルのファイアウォールなので、HTTP Request に対応する Response は、通常アウトバウンドルールへ設定する必要はありません (Request と Response で1セット。インバウンドの許可でやってくれる)。今回は、Web サーバへ各種インストールを行うための設定として、Web サーバ発インターネット宛での通信を、アウトバウンドで許可しています。★

② Web サーバを構築

このタスクでは、Amazon EC2 インスタンスを新しい VPC に作成します。インスタンスは、ウェブサーバーとして機能するように設定します。

1. **サービス** で **[EC2]** をクリックします。
2. ナビゲーションペインからインスタンスを選択し、**インスタンスの作成** をクリックします。
3. **Amazon Linux 2** で **選択** をクリックします。
4. **t2.micro** を選択します。
5. **次の手順: インスタンスの詳細の設定** をクリックします。

次に、新しい VPC のパブリックサブネットに作成されるようにインスタンスを設定します。

6. 以下のように設定します。
 - ネットワーク: WebDB_VPC

- サブネット: **Public Subnet 2** (Private ではありません)
 - 自動割り当てパブリック IP: 有効
7. 高度な詳細 セクション (ページ下部) を展開します。
8. 以下のコードをコピーし、[ユーザーデータ] ボックスに貼り付けます。

---↓ここから↓---

```
#!/bin/bash

# Install Apache Web Server and PHP

yum install -y httpd mysql php

# Download Lab files

wget https://aws-tc-largeobjects.s3.amazonaws.com/AWS-TC-AcademyACF/acf-lab3-vpc/lab-app.zip

unzip lab-app.zip -d /var/www/html/

# Turn on web server

chkconfig httpd on

service httpd start

---↑ここまで↑---
```

9. **次の手順: ストレージの追加** をクリックします。

ストレージにはデフォルト設定を使用します。

10. **次の手順: タグの追加** をクリックします。

タグは、リソースを識別するために使用できます。タグを使用して、インスタンスに Name を割り当てます。

11. **タグの追加** をクリックし、以下のように設定します。

- キー: **Name**
- 値: **Web Server WebDB_VPC**

12. **次の手順: セキュリティグループの設定** をクリックします。

先ほど作成した **Web Security Group** を使用するようにインスタンスを設定します。

13. **既存のセキュリティグループを選択する** をクリックします。
14. **Web Security Group** を選択します。※Default と間違えないように！！
15. **確認と作成** をクリックします。
16. インスタンス情報を確認し、**起動** をクリックします。
17. 「**既存のキーペアを選択するか、新しいキーペアを作成します**」ダイアログボックスで「**新しいキーペアを作成**」を選択し、キーペア名に「WebKey」と入力したら、キーペアのダウンロード ボタンをクリックし、保存します。保存が完了したら、インスタンスの作成ボタンをクリックし、その後の画面で、**インスタンスの表示** をクリック。
18. **Web Server 1** の [ステータスチェック] 列に **2/2 のチェックに成功しました** と 表示されるまで待ちます。
19. ページの下部にある [説明] タブに表示されている [IPv4 パブリック IP] の値をコピーし、Web ブラウザからアクセスして、Web ページが閲覧できることを確認します。
20. Tera Tearn で SSH アクセスを行い、管理接続できることも確認して下さい。

EC2 へのログインは、ユーザ名：ec2-user、認証方式は RSA/DSA～鍵を使うで、秘密鍵にダウンロードしたファイル、WebKey.pem を指定します。

[STEP3] DB サーバの構築

①DB サーバのセキュリティグループを作成

1. AWS マネジメントコンソールの **サービス** で [VPC] をクリックします。
2. ナビゲーションペインで [セキュリティグループ] をクリックします。
3. **セキュリティグループの作成** をクリックし、以下のように設定します。
 - セキュリティグループ名: **DB Security Group**
 - 説明: **Permit access from Web Security Group**
 - VPC: WebDB_VPC

[インバウンドのルール] で、ルールの追加 ボタンをクリックします。

- タイプ: MySQL/Aurora (3306)
- ソース: カスタム▼ の右側で sg と入力して Web Security Group を選択※
間違えると Web サーバから DB へ接続できないので要注意

Web Security Group に関連付けられている EC2 インスタンスからのインバウンドトラフィックをポート 3306 で許可するデータベースセキュリティグループが設定されます。

4. セキュリティグループの作成 をクリック。

このセキュリティグループは、Amazon RDS データベースを作成するときに使用します。

②データベースのサブネットグループを作成

このタスクでは、データベースに使用できるサブネットを RDS に通知するために使用される _DB サブネットグループ_を作成します。各 DB サブネットグループでは、少なくとも 2 つのアベイラビリティゾーンでサブネットが必要です。

1. [サービス]メニューから [RDS] をクリックします。
2. ナビゲーションペインで [サブネットグループ] をクリックします。
3. [DB サブネットグループの作成] をクリックし、以下のように設定します。
 - 名前: DB-Subnet-Group
 - 説明: DB Subnet Group
 - VPC: WebDB_VPC
 - サブネットを追加 の[アベイラビリティゾーン] で、最初の 2 つのゾーン (us-east-1a と us-east-1b) を選択します。
 - [サブネット] で、10.0.1.0/24 および 10.0.3.0/24 のサブネットを選択します。

これで、これらのサブネットが [選択されたサブネット] のテーブルに表示されます。

4. [作成] をクリックします。

この DB サブネットグループは、次のタスクでデータベースを作成するときに使用します。

③DB サーバの構築

1. ナビゲーションペインで [ダッシュボード] をクリックします。
2. **データベースの作成** をクリックします。

画面上部に **Switch to the new database creation flow** が表示された場合、それをクリックしてください。

3. データベース作成方法を選択は、標準作成のままで、**MySQL** を選択します。
4. テンプレートは、**無料利用枠** を選択します。
5. [設定] で以下のように設定します。
 - DB インスタンス識別子: **lab-db**
 - マスターユーザー名: **master**
 - マスターパスワード: **lab-password**
 - パスワードを確認: **lab-password**
6. [DB インスタンスサイズ] で以下のように設定します。
 - バースト可能クラス (t クラスを含む) を選択
 - **db.t2.micro** を選択
7. [ストレージ] で以下のように設定します。
 - ストレージタイプ: **汎用 (SSD)**
 - ストレージ割り当て: **20**
 - ストレージの自動スケーリングを有効にするのチェックを**外す**
8. [接続] で以下のように設定します。
 - **Virtual Private Cloud (VPC): WebDB_VPC**
9. **追加の接続設定** を展開し、以下のように設定します。
 - **既存の VPC セキュリティグループ: DB Security Group** をクリックして青色の強調表示にする **※Default は×ボタンで削除しておく**
 - アベイラビリティゾーンは、**us-east-1b** を選択
10. **追加設定** を展開し、以下のように設定します。
 - **最初のデータベース名: lab**
 - **自動バックアップの有効化** チェックボックスをオフ
 - **拡張モニタリングの有効化** チェックボックスをオフ
 - **マイナーバージョン自動アップグレードの有効化** チェックボックスをオフ

11. **データベースの作成** をクリックします。
これでデータベースが作成されます。
12. **lab-db** をクリックします（リンク自体をクリックします）。

4 分ほど待つと、データベースが利用可能になります。デプロイプロセスでは、2 つの異なるアベイラビリティゾーンにデータベースがデプロイされます。
13. **[情報]** が **[変更中]** または **[利用可能]** になるまで待ちます。
14. **[接続とセキュリティ]** セクションが表示されるまで下にスクロールし、**[エンドポイント]** フィールドの値をコピーします。

この値は **lab-db.cggq8lhnxvkv.us-west-2.rds.amazonaws.com** のような形式で表示されます。
15. エンドポイントの値をテキストエディタに貼り付けます。この値はラボで後ほど使います。

④ DB サーバを操作する

1. ウェブブラウザで新しいタブを開き、Web サーバ(EC2 インスタンス)の IP アドレスを貼り付けて **Enter** キーを押します。
2. ページの上部に表示されている **RDS** リンクをクリックします。

次に、データベースに接続するようアプリケーションを設定します。
3. 以下のように設定します。
 - **Endpoint:** 先ほどコピーしたエンドポイントの値をテキストエディタに貼り付け
 - **Database:** **lab**
 - **Username:** **master**
 - **Password:** **lab-password**
 - **[Submit]** をクリック
4. **今回は何も表示されません。画面が切り替われば OK です。**

⑤ Web サーバ(EC2 インスタンス)を踏み台にして、DB サーバへ接続し操作する

1. Tera Tearn から SSH へ Web サーバ(EC2 インスタンス)へ接続

2. 下記コマンドを打って、データベースサーバへ接続

```
mysql -h エンドポイント -P 3306 -u master -p
```

※エンドポイントは、④で使用したエンドポイントです

3. パスワードの入力が求められるので、lab-password (DB の作成で設定したパスワード) と入力します。パスワードは入力中に表示されないので、慌てないでください。

4. 接続ができれば、データベースの操作ができることを確認しましょう。画面表示のプロンプトが、MySQL~> のようになっていれば、接続成功です。

【データベースの選択】

use lab (DB 名) (※DB 作成時に指定したデータベースの名前 : 手順書どおりなら lab)

【テーブルの作成】

```
create table TEST(id int(5), name varchar(20));
```

【テーブルへ行の挿入】

```
insert into TEST values(1, 'Sato');
```

```
insert into TEST values(2, 'Takei');
```

【テーブル内容の表示】

```
select * from TEST;
```

※MySQL の実行は、; セミコロンで終了なので、最後に付け忘れると実行できません。