

Tái cấu trúc Ứng dụng Web với RDS và S3

Mục tiêu: Thực hành việc tách cơ sở dữ liệu và tài sản tĩnh ra khỏi máy chủ ứng dụng, sử dụng các dịch vụ được quản lý của AWS.

Phần A: Di chuyển Cơ sở dữ liệu sang RDS

1. Tạo một RDS DB Instance:

- Trong AWS Console, tìm và truy cập dịch vụ **RDS**.
- Nhấp vào **"Create database"**.
- Chọn **"Standard create"** và **"MySQL"**.
- Trong mục **"Templates"**, chọn **"Free tier"**.
- **Settings:**
 - `DB instance identifier`: Đặt tên, ví dụ: `webapp-db`.
 - `Master username`: `admin`.
 - `Master password`: Đặt một mật khẩu mạnh và ghi nhớ nó.
- **Connectivity:**
 - Đảm bảo RDS được tạo trong cùng **VPC** với EC2 instance.
 - Trong mục **"Public access"**, chọn **"No"**. Đây là phương pháp bảo mật tốt nhất.
- **Security group**: Chọn Security Group mặc định hoặc tạo một cái mới. Cách cấu hình sẽ được hướng dẫn thực hiện ở bước tiếp theo.
- Nhấp **"Create database"**. Quá trình này có thể mất vài phút để thực hiện

2. Cấu hình Security Group cho RDS:

- Trong khi RDS đang được tạo, đi đến bảng điều khiển **EC2** -> **Security Groups**.
- Tìm Security Group mà RDS instance đang sử dụng.
- Chọn nó, vào tab **"Inbound rules"** và nhấp **"Edit inbound rules"**.
- Nhấp **"Add rule"**:
 - `Type`: `MYSQL/Aurora`.
 - `Source`: Chọn `Custom`, và trong ô tìm kiếm, chọn Security Group của **EC2 instance** của bạn.
- Lưu lại. Quy tắc này chỉ cho phép EC2 instance kết nối đến database.

3. Lấy thông tin Endpoint của RDS:

- Quay lại bảng điều khiển **RDS**. Khi database xuất hiện trạng thái là **"Available"** thì nhấp vào nó.

- Trong tab **"Connectivity & security"**, sao chép giá trị **Endpoint**. Nó sẽ trông giống như:
`webapp-db. xxxxxxxx.us-east-1.rds.amazonaws.com`

4. Cập nhật Mã nguồn Ứng dụng:

- SSH vào EC2 instance.
- Mở file mã nguồn có chứa chuỗi kết nối database (ví dụ: `server.js`).
- Thay đổi các thông tin kết nối đến RDS:
 - `db_host`: Dán **Endpoint** của RDS vào đây.
 - `db_user`: `admin`.
 - `db_pass`: Mật khẩu bạn đã tạo cho RDS.
 - `db_name`: Tên database bạn muốn sử dụng (bạn có thể cần kết nối và tạo nó trước).
- Khởi động lại ứng dụng (ví dụ: `pm2 restart app`).

5. Kiểm tra Kết nối:

Tải lại trang web. Nếu nó vẫn hoạt động, có nghĩa là ứng dụng đã kết nối thành công tới RDS.

Phần B: Phục vụ File tĩnh từ S3

1. Tạo một S3 Bucket:

- Trong AWS Console, tìm và truy cập dịch vụ **S3**.
- Nhấp **"Create bucket"**.
- **Bucket name**: Đặt một tên duy nhất (ví dụ: `my-unique-webapp-assets-12345`).
- **Region**: Chọn Region gần với người dùng của bạn nhất.
- Trong **"Block Public Access settings for this bucket"**, **bỏ tick** ở ô `Block all public access`. Đánh dấu vào ô xác nhận. Đây là bước cần thiết để phục vụ file cho website.
- Nhấp **"Create bucket"**.

2. Tải File tĩnh lên S3:

- Nhấp vào tên bucket vừa tạo.
- Nhấp **"Upload"**, sau đó **"Add files"**.
- Chọn một vài file tĩnh từ dự án (ví dụ: một file ảnh, file CSS).
- Tải nội dung lên.

3. Tạo một Bucket Policy để Cho phép Đọc Công khai:

- Trong bucket của bạn, đi đến tab **"Permissions"**.
- Cuộn xuống **"Bucket policy"** và nhấp **"Edit"**.

- Dán vào đoạn JSON sau. **Hãy thay thế YOUR_BUCKET_NAME bằng tên bucket của bạn.**

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PublicReadGetObject",
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::YOUR_BUCKET_NAME/*"
    }
  ]
}
```

4. Cập nhật Mã nguồn để sử dụng URL của S3:

- Trong S3 console, nhấp vào một file đã tải lên (ví dụ: logo.png).
- Sao chép giá trị **Object URL**.
- Mở mã nguồn frontend (ví dụ: một component React) và thay thế đường dẫn tương đối của file tĩnh bằng URL S3 đầy đủ.

i. **Trước:** ``

ii. **Sau:** ``

- Triển khai lại mã nguồn đã cập nhật lên EC2 của bạn.

5. Kiểm tra cuối cùng: Tải lại trang web. Mở công cụ phát triển của trình duyệt (Developer Tools), vào tab "Network" và xác nhận rằng các file tĩnh đang được tải từ S3, không phải từ EC2 instance.