Project Report: Path-Based Routing using Application Load Balancer in AWS

Project Title

Path-Based Routing with AWS ALB and NGINX

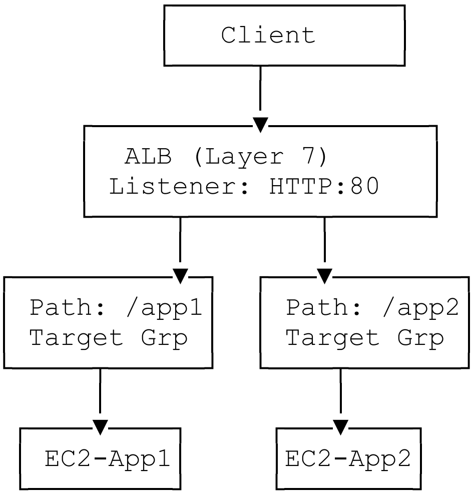
Objective

To implement a scalable web architecture using an AWS Application Load Balancer that forwards requests to different target groups based on URL paths (e.g., `/app1`, `/app2`) using NGINX as the backend server.

Key Concepts & Services Used

- **Application Load Balancer (ALB)**
- **Target Groups**
- **Listeners (HTTP 80)**
- **Listener Rules (Path-based)**
- **NGINX Servers**
- **Security Groups (NSG-equivalent)**
- **EC2 Instances (web backend)**

Architecture Overview



Steps Implemented

1. Launch EC2 Instances

- Deployed two EC2 instances with NGINX installed.
- Configured different content/pages served at root `/` (for visibility in tests).

2. Security Groups

- Allowed HTTP (port 80) inbound from ALB SG.
- Restricted SSH and other access to prevent unwanted traffic.

3. Create Target Groups

- **Target Group 1**: Targets EC2 instance serving App1 (`/app1`)
- **Target Group 2**: Targets EC2 instance serving App2 (`/app2`)
- Health checks set to `/` for both

4. Create Application Load Balancer

- Internet-facing ALB with at least two public subnets
- Listener on port 80
- Associated with Security Group allowing HTTP traffic

5. Configure Listener Rules

- Rule 1: If path is `/app1` → forward to **Target Group 1**
- Rule 2: If path is `/app2` → forward to **Target Group 2**
- Default: Return 404 or custom fallback page

6. Test Routing Behaviour

- Visit `http://<alb-dns>/app1` → App1 content
- Visit `http://<alb-dns>/app2` → App2 content

NGINX Configuration (on EC2 instances)

Sample for App1 Instance:

```
sudo apt update && sudo apt install nginx -y
echo "Welcome to App1" | sudo tee /var/www/html/index.html
sudo systemctl start nginx
```

Repeat with different content for App2.
