# Amazon EC2:

# What:

Scalable compute capacity for running virtual servers in the cloud.

## How:

Launch instances with desired AMIs, choose instance types, configure security groups, and manage scaling.

#### **How Much:**

Pay for compute hours or seconds (depending on billing model); options include on-demand, reserved, and spot instances.

# **EC2 HANDS ON:**

Launch → Secure → Connect → Deploy Apache → Hello World

# What you'll build

A tiny web server on an EC2 Ubuntu instance serving a "Hello World" page over HTTP.

# At a glance

- EC2 launch (t2.micro / t3.micro)
- Security Group for SSH(22) & HTTP(80)
- SSH from terminal
- Install Apache
- Deploy simple index.html

#### **Step 1 — Launch an EC2 Instance**

- 1. Sign in to the AWS Management Console  $\rightarrow$  EC2  $\rightarrow$  "Launch instance".
- 2. Name: my-ec2
- 3. Select AMI: Ubuntu Server 22.04 LTS (free tier eligible).
- 4. Instance type: t2.micro or t3.micro (free tier eligible).
- 5. Key pair: Create new or reuse an existing .pem.
- 6. Network: Default VPC is fine for testing.
- 7. Security group: Allow SSH (22) from your IP and HTTP (80) from 0.0.0.0/0.
- 8. Storage: Default (8-10 GB gp3) is okay.
- 9. Launch instance

#### Step 2 — Connect via SSH

From your terminal:

```
chmod 400 ~/keys/my-key.pem
ssh -i ~/keys/my-key.pem ubuntu@<PUBLIC_IP_ADDRESS>

If SSH fails, check your Security Group (port 22), and ensure the instance is running.
```

#### Step 3 — Update & Install Apache

```
sudo apt update -y
sudo apt install -y apache2
sudo systemctl start apache2
sudo systemctl enable apache2
systemctl status apache2

Test in your browser: http://<PUBLIC_IP_ADDRESS> should show Apache's default page.
```

### Step 4 — Deploy Hello World

```
echo 'Hello World from EC2 ' | sudo tee /var/www/html/index.html
echo 'Deployed by <b>Your Name</b>' | sudo tee -a
/var/www/html/index.html
Pro tip: Use EC2 User Data to automate Apache installation & page setup.
```

## Step 5 — Costs & Cleanup

When done testing:

- 1. Stop or terminate the instance.
- 2. Delete unattached Elastic IPs and unused volumes.

- 3. Remove Security Groups and Key Pairs.
- 4. Use AWS Cost Explorer to track usage.

⚠ Stopping an instance retains the EBS volume (storage costs). Terminate if no longer needed.

# **Troubleshooting**

Issue Fix

Port 22/80 timeouts Check Security Group rules, network

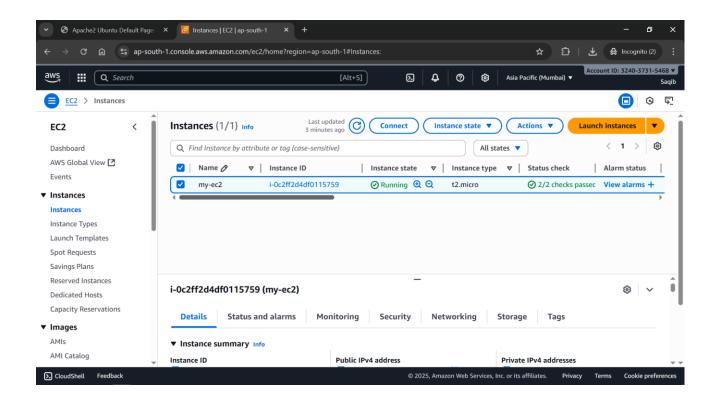
ACLs, and public IPv4.

Permission denied (publickey) Ensure correct key, chmod 400, user =

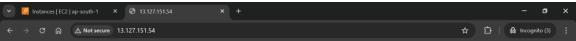
ubuntu.

Apache not serving Check apache2 service & logs.

Page shows old content Clear browser cache or reload.







Hello World from EC2

Deployed by Saqib Reza