AWS EC2 Getting Started Guide

This guide walks you through creating, connecting to, and managing your first EC2 instance on AWS.

Prerequisites

- AWS account with appropriate permissions
- Basic familiarity with terminal/command line

Step 1: Login to AWS Console

1. Navigate to the AWS Console:

   ```
   https://eu-north-1.console.aws.amazon.com/console/home?region=eu-north-1
   ```

2. In the search bar, type `ec2` and press Enter
3. This will take you to the EC2 Overview page:

   ```
   https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#Overview:
   ```

Step 2: Launch New Instance

1. From the left menu, choose **Instances**:

   ```
   https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#Instances:
   ```

2. Click **Launch instances** from the top left corner:

   ```
   https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-
   1#LaunchInstances:
   ```

Step 3: Configure Instance Settings

Name and Tags

- **Name**: `Web1`

Application and OS Images (Amazon Machine Image)

- Keep the default **Amazon Linux** selection

Instance Type

- Select `t2.micro` or `t3.micro` (any micro instance type for free tier eligibility)

Key Pair

1. Click **Create new key pair**
2. **Name**: `key01`
3. **Type**: RSA
4. **Format**: `.pem`

Network Settings

- Under **Allow SSH traffic from**, change from "Anywhere" to **My IP**
- This creates a security group that only allows SSH access from your current IP address

Configure Storage

- No changes needed (keep defaults)

Advanced Details

- No changes needed (keep defaults)

Step 4: Launch Instance

1. Click the orange **Launch instance** button
2. Wait 1-2 minutes for AWS to launch your new EC2 instance

Step 5: Connect to Your Instance

Get Connection Instructions from AWS Console

1. After the instance is created and shows "Running" status, **select the instance line** by clicking on it
2. Click the **Connect** button at the top of the instances list
3. In the connection dialog, go to the **SSH client** tab

Copy the SSH Command from AWS

Important: AWS automatically generates the exact SSH command you need. In the SSH client tab, you'll see:

- Instructions for setting file permissions
- The complete SSH command with your instance's specific public DNS name
- Example format: `ssh -i "key01.pem" ec2-user@ec2-XX-XXX-XXX-XXX.eu-north-1.compute.amazonaws.com`

Copy this command directly from the AWS console - don't try to type it manually as the public DNS changes for each instance.

Execute the Connection

1. Navigate to your Linux download directory where your `key01.pem` file is located
2. Set proper permissions for the key file:

   bash

   ```
   chmod 400 "key01.pem"
   ```

3. **Paste and run the SSH command you copied from AWS**:

   bash

   ```
   # Example (your actual command will have different IP/DNS):
   ssh -i "key01.pem" ec2-user@ec2-16-171-112-254.eu-north-1.compute.amazonaws.com
   ```

Successful Connection

Upon successful connection, you'll see the Amazon Linux welcome message:

```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _|_|_
       _/m/'

Last login: Wed May 28 17:44:37 2025 from 79.177.133.211
```

Step 6: Clean Up Resources

Stop and Terminate Instance

1. Go back to the Instances page:

   ```
   https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#Instances:
   ```

2. **Stop** the instance first, then **Terminate** it

Check for Remaining Volumes

1. Navigate to **Elastic Block Store → Volumes**:

   ```
   https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#Volumes:
   ```

2. Verify no redundant volumes exist to avoid unnecessary charges

Important Notes

- Always terminate instances when not in use to avoid charges
- Keep your `.pem` key file secure and never share it
- The security group restricting SSH to "My IP" provides better security than "Anywhere"
- Check for leftover resources (volumes, snapshots) to prevent unexpected billing

Troubleshooting

- If connection fails, verify your security group allows SSH (port 22) from your IP
- Ensure the key file has correct permissions (400)
- Check that you're using the correct username (`ec2-user` for Amazon Linux)