This study guide is based on the video lesson available on TrainerTests.com

Installing Ubuntu in AWS EC2

This chapter guides you through setting up an Ubuntu server on Amazon's Elastic Compute Cloud (EC2) service. We'll navigate the AWS console to create a new instance, ensuring cost-efficiency and secure access.

Prerequisites

- An AWS account with appropriate permissions to launch EC2 instances.
- Basic understanding of navigating web interfaces.

Accessing the AWS Management Console

1. Navigate to the AWS Management Console at https://aws.amazon.com/console/.
2. Sign in to your AWS account using your credentials.

Launching an Instance

1. **Search for EC2:** In the console's search bar, type "EC2" and select "Amazon EC2" from the search results. This will take you to the EC2 service dashboard.
2. **Launch Instance:** Click the orange "Launch Instance" button on the EC2 dashboard.
3. **Choose an AMI (Amazon Machine Image):**
   - On the "Choose an AMI" page, select an Ubuntu image that qualifies for the free tier. You can find free-tier eligible AMIs by searching for "Ubuntu Server" and filtering by "Free tier eligible" under the "Filters" section on the left side of the page.
4. **Select Instance Type:**
   - Under "Choose an Instance Type," select a cost-effective option like the "t2.micro" instance type. This instance type is suitable for basic server tasks and falls within the free tier limitations.
5. **Configure Instance Details:**
   - You can leave most of the following options at their default settings unless you have specific requirements:
     - Number of Instances: 1
     - Storage: Defaults are sufficient for basic setups.
     - VPC (Virtual Private Cloud): Defaults are sufficient for this tutorial.
6. **Create a Key Pair:**

   - Under "Network Settings," choose "Create a new key pair." Enter a key pair name (e.g., "myUbuntuServer") and download the key pair (.pem) file securely. This key file will be used to connect to your Ubuntu server later.
7. **Security Group Configuration:**
   - Click on "Security group" and then "Create Security Group."
   - Give your security group a name (e.g., "UbuntuServerAccess") and a description.
   - Click "Edit inbound rules."
   - Add a new rule allowing SSH access:
     - Rule Type: SSH
     - Source: Custom – Choose your IP address or CIDR block to restrict access (0.0.0.0/0 allows all traffic for demonstration purposes, but restrict this for real-world scenarios).
     - Port Range: 22 (default SSH port)
   - Save the security group configuration.
8. **Launch the Instance:**
   - Review your instance configuration on the final screen.
   - Ensure you have selected the free-tier eligible AMI, t2.micro instance type, and created a new key pair.
   - Click "Launch" to start provisioning your Ubuntu server instance.
9. **Note your Instance ID:**
   - On the launch confirmation page, make note of your instance's public DNS name (e.g., [invalid URL removed]). You'll use this to connect to your server.

Connecting to your Ubuntu Server

Once your instance has launched (it may take a few minutes), you can connect to it using SSH. Here's how:

1. **Prerequisites:**
   - Downloaded key pair (.pem) file from step 6.
   - An SSH client (e.g., terminal with SSH installed for Linux/macOS or PuTTY for Windows).
2. **Connect using SSH:**
   - Open your SSH client and connect to your server using its public DNS name and username "ubuntu" (default for Ubuntu):

```
ssh -i <key_pair_file.pem> ubuntu@<public_dns_name>
* Replace `<key_pair_file.pem>` with the actual path to your downloaded key pair file.
* Replace `<public_dns_name>` with the public DNS name of your instance obtained in step
9 of the launch process.
```

3. **Enter your key passphrase (if set):**
   - If you set a passphrase while creating the key pair, you'll be prompted to enter it at this stage.
4. **Command Line Access:**
   - Upon successful authentication, you'll be granted command-line access to your Ubuntu server.