DevOps Shack

Steps for Creating a Virtual Machine (VM) in AWS

[Click Here To Enrol To Batch-6 | DevOps & Cloud DevOps]

Creating a Virtual Machine (VM), also known as an EC2 instance, in AWS involves several steps. Here's a detailed guide to help you through the process.

Step 1: Sign in to AWS Management Console

- Go to the AWS Management Console.
- Sign in with your AWS account credentials.

Step 2: Navigate to EC2 Dashboard

- In the AWS Management Console, type "EC2" in the search bar and select "EC2" to go to the EC2 Dashboard.

Step 3: Launch an Instance

- Click on the "Launch Instance" button.

Step 4: Choose an Amazon Machine Image (AMI)

- Select an AMI that serves as the base image for your instance. Amazon Linux 2 is a common choice for general purposes.
- You can also choose from other options like Ubuntu, Windows, or other pre-configured AMIs.

Step 5: Choose an Instance Type

- Select an instance type based on the required vCPU and memory. The "t2.micro" instance type is a common choice for free tier eligibility.
- Click "Next: Configure Instance Details."

Step 6: Configure Instance Details

- Specify the number of instances to launch.
- Configure the network settings:
    - Network: Select the VPC (Virtual Private Cloud) where the instance will be launched.
    - Subnet: Select a subnet within the VPC.
    - Auto-assign Public IP: Enable if you want the instance to have a public IP address.
- Configure other settings as needed, such as IAM roles, shutdown behavior, and enable monitoring if required.
- Click "Next: Add Storage."

Step 7: Add Storage

- Configure the storage for your instance. By default, an EBS (Elastic Block Store) volume is attached.
    - Size (GiB): Specify the size of the root volume.
    - Volume Type: Select the volume type, such as General Purpose SSD (gp2).
- Add additional volumes if required.
- Click "Next: Add Tags."

Step 8: Add Tags

- Add tags to help manage your instances. Tags are key-value pairs.
    - Example: Key = Name, Value = MyInstance
- Click "Next: Configure Security Group."

Step 9: Configure Security Group

- Create a new security group or select an existing one.
    - Security Group Name: Enter a name for the security group.
    - Description: Enter a description for the security group.
    - Inbound Rules: Add rules to allow specific traffic to your instance.
        - Example: Allow SSH traffic on port 22 from a specific IP address.
        - Type: SSH
        - Protocol: TCP
        - Port Range: 22

        - Source: MyIP (select your IP)
    - Outbound Rules: By default, all outbound traffic is allowed.
- Click "Review and Launch."

Step 10: Review and Launch

- Review the instance details, storage, tags, and security group configuration.
- Click "Launch."

Step 11: Select a Key Pair

- Select an existing key pair or create a new one to securely connect to your instance via SSH.
    - Create a New Key Pair: If you don't have an existing key pair, create a new one, download the .pem file, and store it securely.
    - Select an Existing Key Pair: If you already have a key pair, select it from the list.
- Acknowledge that you have access to the selected key pair.
- Click "Launch Instances."

Step 12: View Instance

- Click "View Instances" to go to the EC2 Dashboard where you can see the status of your newly launched instance.
- Wait for the instance state to become "running."

Connecting to Your EC2 Instance

Once the instance is running, you can connect to it using SSH.

Step 1: Locate the Public IP

- Go to the EC2 Dashboard.
- Select the running instance and find its Public IP address.

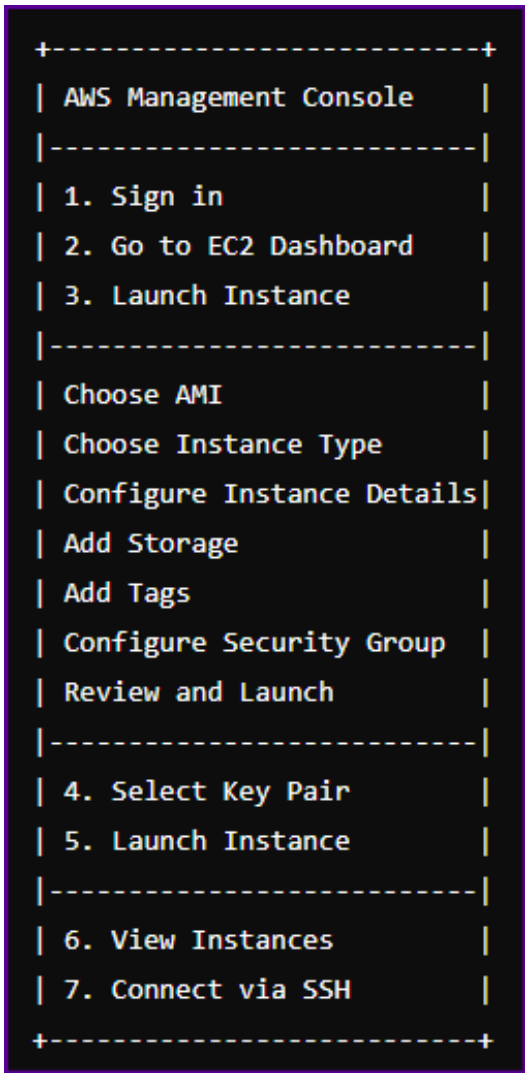
Step 2: Connect via SSH

- Open a terminal on your local machine.
- Use the SSH command to connect to the instance.

ssh -i /path/to/your-key-pair.pem ec2-user@your-instance-public-ip

Replace /path/to/your-key-pair.pem with the path to your downloaded key pair and your-instance-public-ip with the instance's public IP address.

Textual Diagram



This guide provides a comprehensive step-by-step process to create and launch a VM in AWS and connect to it using SSH.