**Project Title:**

**Deployment and Configuration of Jenkins on AWS EC2 for Continuous Integration and Delivery**

**Project Description:**

In this project, I deployed and configured a Jenkins Continuous Integration (CI) server on an AWS EC2 instance to automate the build and deployment processes. The project involved launching an EC2 instance, setting up the necessary environment by installing Java and Jenkins, and configuring Jenkins to be accessible from the internet.

I also configured security groups to manage inbound and outbound traffic, ensuring that the Jenkins server was secure and accessible only through designated ports. This setup facilitates continuous integration and delivery (CI/CD) pipelines, enabling automated testing, building, and deployment of applications, thus enhancing the development workflow and improving overall efficiency.

The project demonstrates a strong understanding of cloud infrastructure management, CI/CD practices, and the deployment of scalable and reliable software solutions using AWS services.

**Your First AWS Project: Deploy a simple web application(such as jenkins) on the ec2 instance and access the application from outside AWS.**

**steps:**

**1. Create an EC2 Instance:**

* **Login to AWS Management Console:**
  + Go to the EC2 Dashboard.
* **Launch a New EC2 Instance:**
  + Click "Launch Instance."
  + Choose an Amazon Machine Image (AMI), such as Amazon Linux 2 or Ubuntu Server.
  + Select an instance type (e.g., t2.micro for free tier).
  + Configure instance details, storage, and add tags if needed.
  + **Configure Security Group:**
    - Create a new security group or use an existing one.
    - Allow inbound traffic for:
      * SSH (port 22) for remote access.
      * HTTP (port 80) and HTTPS (port 443) for web access.
      * Custom TCP Rule for Jenkins (port 8080).
  + Review and launch the instance.
  + Download the key pair (.pem file) to access the instance.

**2. Access the EC2 Instance:**

* Open a terminal or command prompt.
* Use the following command to connect to your instance:

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

* Replace /path/to/your-key.pem with the path to your key file and your-ec2-public-ip with your instance's public IP.

**3. Install Jenkins on the EC2 Instance:**

* **Update the package index:**

sudo yum update -y # For Amazon Linux 2

sudo apt-get update # For Ubuntu

* **Install Java (required for Jenkins):**

sudo yum install java-1.8.0-openjdk-devel -y # For Amazon Linux 2

sudo apt-get install openjdk-11-jdk -y # For Ubuntu

* **Add the Jenkins repository and import the GPG key:**
  + For Amazon Linux 2:

sudo wget -O /etc/yum.repos.d/jenkins.repo \

https://pkg.jenkins.io/redhat-stable/jenkins.repo

sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

* + For Ubuntu:

wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add -

sudo sh -c 'echo deb http://pkg.jenkins.io/debian-stable binary/ > /etc/apt/sources.list.d/jenkins.list'

sudo apt-get update

* **Install Jenkins:**

sudo yum install jenkins -y # For Amazon Linux 2

sudo apt-get install jenkins -y # For Ubuntu

* **Start Jenkins:**

sudo systemctl start jenkins

sudo systemctl enable jenkins

**4. Access Jenkins from Outside AWS:**

* Open a web browser.
* Enter http://your-ec2-public-ip:8080.
* You should see the Jenkins setup wizard.
* To get the initial admin password, use:

sudo cat /var/lib/jenkins/secrets/initialAdminPassword

* Copy the password and complete the Jenkins setup.

**5. (Optional) Setup a Domain Name:**

* If you want to access Jenkins via a custom domain, configure an Elastic IP and use a DNS service like Route 53 to map your domain to the EC2 instance.

**6. Security Considerations:**

* Ensure your security group rules are appropriately configured to only allow necessary access.
* Consider setting up HTTPS for secure access.