AWS EC2 Instance Setup and Jenkins Installation Guide

Step 1: AWS Account Creation and EC2 Instance Setup

1. **Create an AWS Account.**
2. **Navigate to EC2 Service:**

    - Create an AWS account with Free tier eligibility.
    - Navigate to AWS Dashboard.
    - Go to Services --> EC2.
    - Click on Launch Instance.

3. **Configure and Launch Instance:**

    - Configure the instance as required.
    - Recommended:

        - Name Instance name as our wish.
        - Select operating system as ubuntu.
        - We can use 30gb of storage freely for upto 1 year.

    - Create and download the key pair (.pem file).

Step 2: Connecting to the EC2 Instance

1. **Open Command Prompt:**

    - Navigate to the directory where the .pem file is downloaded.

2. **Connect to Instance:**

- In the instance console, click on Connect.
- Navigate to Connect --> SSH Client.
- Copy and paste the provided SSH command (Example**: ssh -i "jenkins-node.pem" ubuntu@ec2-16-171-129-214.eu-north-1.compute.amazonaws.com**).

Step 3: Install Node.js and npm

1. Update and Install Node.js:

- Execute:

```
sudo apt update
sudo apt install nodejs
sudo apt install npm
node -v
npm -v
```

Step 4: Install Jenkins

1. Follow Instructions from Jenkins Official Site:

   - Visit https://pkg.jenkins.io/debian-stable/.

2. **Add Jenkins Key and Repository:**

   - Execute:
     - **sudo wget -O /usr/share/keyrings/jenkins-keyring.asc https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key**

   Then,

     - **echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/" | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null**

3. **Update Packages and Install Jenkins:**

   Execute:

   - **sudo apt-get update**
   - **sudo apt-get install fontconfig openjdk-17-jre**
   - **sudo apt-get install Jenkins**

4. **Start Jenkins:**

Execute:

- **sudo systemctl enable jenkins**
- **sudo systemctl start jenkins**
- **sudo systemctl status jenkins**

Step 5: Initial Jenkins Setup

Access Jenkins:

- Navigate to your public IP + :8080 to access Jenkins.

Unlock Jenkins:

- Find the initial admin password at **/var/jenkins_home/secrets/initialAdminPassword.**
- Paste the password, install default plugins, and create a user account.

Step 6: Create a Jenkins Pipeline

Create New Pipeline Project:

- **Click on New Item and select Pipeline Project.**

Configure Secret Text and SSH key:

- **Environmental Credentials**

1. Select Manage Jenkins and click on credentials.
2. Click on global on the credentials click on add credentials.
3. Add environmental variables(use if your project has the env. Variables) in terms of secret text format on the dialog box.

- **Generation of SSH key:**

1. Open Terminal and connect to ec2 instance using key pair ssh key.
2. Generate ssh-keygen -t rsa -b 4096 -C ‘ your_email@example.com’.

- **Add the SSH Key to Your SSH Agent**

1. Start the SSH agent in the background: eval "$(ssh-agent -s)".
2. Add your SSH private key to the SSH agent:" ssh-add ~/.ssh/id_rsa".
3. Add the SSH Key to Your GitHub/GitLab/Bitbucket Account
4. Copy the SSH public key to your clipboard: cat ~/.ssh/id_rsa.pub.
5. Log in to your repository hosting service (e.g., GitHub, GitLab, Bitbucket).
6. Navigate to the SSH keys section (usually found under settings).
7. Add a new SSH key by pasting your copied public key.

- **Add the SSH key to Jenkins Credentials:**

1. Go to Manage Jenkins > Manage Credentials.
2. Select the Global domain.
3. Click on Add Credentials.
4. For Kind, select SSH Username with private key.
5. Username: Enter the username that will use the SSH key.
6. Private Key: Choose Enter directly and paste your private key content.

- **Using Pipeline Syntax for Build:**

1. Click on Pipeline Syntax.
2. Generate checkout pipeline script with deployment SSH key.
3. Click on Apply and Save the Configurations.

Step 8:Build the project:

1. Login to the Jenkins Dashboard.
2. Click on project and Select Build now.
3. The Build will be Scheduled .
4. The output is displayed on the build console.

Step 9: Run the Project

1. **Navigate to Project Directory:**

- **Example:** cd /var/lib/jenkins/workspace/your_pipeline_project_name.

2. **Add .env File (if applicable):**

- **Execute:** echo "*__File_content__*" | sudo tee .env > /dev/null
- **Verify**: ls -la

3. **Remove and Install Dependencies:**

   - If node_modules already exists, execute:rm -rf node_modules

     npm install

4. **Start the Node.js Project:**

   - **Execute:**sudo npm install

     sudo node server.js

5. **Access the Website:**

   - Navigate to your public IP address with the specified port (**e.g.,** http://16.171.129.214:3000/).