Jenkins installation on EC2

Introduction

Jenkins is an open-source automation server primarily used for continuous integration and continuous delivery (CI/CD) in software development. It plays a crucial role in automating various stages of the software delivery process, including building, testing, and deploying applications.

Why Jenkins is Widely Used

1. **Open Source:** Jenkins is free to use and has a large community that contributes to its development, ensuring continuous improvement and support.
2. **Extensibility:** With hundreds of plugins available, Jenkins can integrate with virtually any tool in the CI/CD toolchain, allowing teams to customize their workflows according to specific needs.
3. **Cross-Platform Support:** Jenkins can run on various operating systems, including Windows, macOS, and Linux, making it versatile for different environments.
4. **Pipeline as Code:** Jenkins supports defining build processes as code through Jenkinsfiles, which allows for version control of the CI/CD process itself. This approach enhances collaboration among team members.
5. **Scalability:** Jenkins can distribute workloads across multiple machines, improving performance and enabling parallel execution of jobs.
6. **Real-Time Feedback:** By automating the build and test processes, Jenkins provides immediate feedback on code changes, helping developers identify issues early in the development cycle.

These features make Jenkins a popular choice among development teams looking to implement efficient CI/CD practices and streamline their software delivery processes.

Step 1: Launch an EC2 Instance

1. **Log in to AWS Management Console**.
2. **Navigate to EC2** and click on **Launch Instance**.
3. **Choose an Amazon Machine Image (AMI)**:
   ○ Select an Amazon Linux 2 AMI or Ubuntu Server (LTS version).
4. **Choose an Instance Type**:○ Select a type (e.g., t2.micro for free tier).
5. **Configure Instance Details**: (optional)
   ○ Modify settings as needed.
6. **Add Storage**: (default settings are usually sufficient)

7. **Configure Security Group**:
    - Create a new security group or select an existing one.
    - **Add Rules**:
        - HTTP: Port 80
        - HTTPS: Port 443 (optional)
        - Custom TCP Rule: Port 8080 (default Jenkins port)
    - **Review and Launch** the instance.
    - **Select or create a key pair** to access the instance.

Step 2: Connect to Your EC2 Instance

Connect using SSH:

```
ssh -i path_to_your_key.pem ec2-user@your_ec2_public_dns
```

Step 3: Update your local package index and Install Java

```
sudo apt update
sudo apt install openjdk-11-jdk -y
```

Step 4: Add Jenkins Repository

Import the GPG Key:

For Ubuntu:

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

Then add a Jenkins apt repository entry:

```
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
```

Step 5: Update your local package index, then finally install Jenkins:

```
sudo apt-get update
sudo apt-get install fontconfig openjdk-11-jre
sudo apt-get install jenkins
```

Step 6: Start and Enable Jenkins

```
sudo systemctl start jenkins
sudo systemctl enable jenkins
```

Step 7: Access Jenkins

Open your web browser.

http://your_ec2_public_dns:8080

Step 8: Unlock Jenkins**Get the initial admin password**:

```
sudo cat /var/lib/jenkins/secrets/initialAdminPassword
8a0c1015d8d145048d4258e5063d8ca1
```

Copy the password and paste it into the Jenkins setup page.

Step 9: Complete Setup

1. Follow the prompts to customize Jenkins installation. Install suggested plugins or select specific ones.2. Create your first admin user or continue as admin.

References

Debian Jenkins Packages

Linux